Coded Parity Packet Transmission Method for Two Group Resource Allocation

By Hadhrami Ab. Ghani
Supervised by: Dr. M.K. Gurcan
Intelligent Systems and Networks Research Group

A thesis submitted for a PhD degree of Imperial College London

Department of Electrical and Electronic Engineering
Imperial College London

September 15, 2011

Abstract

Gap value control is investigated when the number of source and parity packets is adjusted in a concatenated coding scheme whilst keeping the overall coding rate fixed. Packet-based outer codes which are generated from bit-wise XOR combinations of the source packets are used to adjust the number of both source packets. Having the source packets, the number of parity packets, which are the bit-wise XOR combinations of the source packets can be adjusted such that the gap value, which measures the gap between the theoretical and the required signal-to-noise ratio (SNR), is controlled without changing the actual coding rate. Consequently, the required SNR reduces, yielding a lower required energy to realize the transmission data rate. Integrating this coding technique with a two-group resource allocation scheme renders efficient utilization of the total energy to further improve the data rates. With a relatively small-sized set of discrete data rates, the system throughput achieved by the proposed two-group loading scheme is observed to be approximately equal to that of the existing loading scheme, which is operated with a much larger set of discrete data rates. The gain obtained by the proposed scheme over the existing equal rate and equal energy loading scheme is approximately 5 dB. Furthermore, a successive interference cancellation scheme is also integrated with this coding technique, which can be used to decode and provide consecutive symbols for inter-symbol interference (ISI) and multiple access interference (MAI) mitigation. With this integrated scheme, the computational complexity is significantly reduced by eliminating matrix inversions. In the same manner, the proposed coding scheme is also incorporated into a novel fixed energy loading, which distributes packets over parallel channels, to control the gap value of the data rates although the

SNR of each code channel varies from each other.

Acknowledgments

First and foremost, I would like to express my utmost gratitude to my Creator who gave me the chance to complete my thesis. Deep in my heart, I wish to acknowledge and thank Dr. M.K. Gurcan for his continuous guidance and patience with fruitful ideas in helping me to complete this project up to this stage. I am also very grateful to be accompanied with my beloved wife and little son who are so patient supporting myself morally. Throughout my study duration, Yayasan Telekom Malaysia Berhad (YTM) has been continuously supporting myself and family with a full scholarship, without which it is near impossible for me to complete my PhD. The support received from YTM, financially or morally, is very much appreciated. To all my friends and colleagues, your kind help will always be remembered and may I wish you a meaningful and peaceful life.

I hereby confirm that the content of this thesis is original and with that I am ready to hold full responsibility for all results produced to complete this thesis. All related work mentioned and discussed in this thesis, to the best of my knowledge, have been properly cited.

List of Publications

The content of this thesis is mainly written based on the following publications and submitted manuscripts

Published Journal Paper

- M. Gurcan, H. Ab Ghani, J. Zhou, and A. Chungtragarn, "Bit Energy Consumption Minimization for Multi-path Routing in Ad Hoc Networks," The Computer Journal, 2011.

Submitted Manuscripts for Journal Publication

- M. K. Gurcan and Hadhrami Ab. Ghani, "Coded Parity Packet Approach for Small-sized Packet Transmission in Wireless Ad-hoc Networks," submitted to European Transactions on Communications, Dec. 2010.
- M. K. Gurcan and Hadhrami Ab. Ghani, "Generalized Coded Parity Packet Transmission for Gap Value Control over Parallel Channels," submitted to IEEE Communications Letters, June 2011.
- M. K. Gurcan, Hadhrami Ab. Ghani and Irina Ma, "The Interference-reduced Energy Loading for Multi-code Coded Packet Transmission," submitted to IEEE Journal on Selected Topics in Communications, June 2011.
- M. K. Gurcan, Irina Ma and Hadhrami Ab. Ghani, "The System Value Approach for Interference Cancellation in Multi-channel Systems," submitted to IEEE Transactions on Vehicular Technology, June 2011.

Published Conference Papers

- H. Ghani and M. Gurcan, "Rate multiplication and two-group resource allocation in multi-code CDMA networks," IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, 2009, pp. 551-555.
- A. Ghani, M. Gurcan, and Z. He, "Two-Group Resource Allocation with Channel Ordering and Interference Cancellation," IEEE Wireless Communications and Networking Conference (WCNC), 2010, pp. 1-6.
- M. Gurcan and H. Ab Ghani, "Small-sized packet error rate reduction using coded parity packet approach," IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), 2010, pp. 419-424.
- Z. He, M. Gurcan, and H. Ghani, "Time-efficient resource allocation algorithm over HSDPA in femtocell networks," in IEEE 21st International Symposium on Personal, Indoor and Mobile Radio Communications Workshops (PIMRC Workshops), 2010, pp. 197-202.

Accepted Manuscripts for Conference Publication

- Mustafa Gurcan, Irina Ma, Hadhrami Ab Ghani, and Zhenfeng He,"Complexity Reduction for Multi-hop Network End-to-End Delay Minimization," accepted for 26th International Symposium on Computer and Information Sciences, 2011.
- Hadhrami Ab Ghani, Mustafa Gurcan, and Zhenfeng He,"Cross-layer Optimization with Two-group Loading for Ad-hoc Networks," accepted for 26th International Symposium on Computer and Information Sciences, 2011.
- M. K. Gurcan, Hadhrami Ab. Ghani and Irina Ma, "Successive Interference Cancellation for Two-group Resource Allocation," accepted for IEEE 22nd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), Mar. 2011.

Nomenclature

General Notation

A	Constant
$\mathbf{A}$	Matrix
$\vec{a}$	Vector
a	Index or variable
$P(.)$	Probability of the argument
$\min(.,.)$	The minimum of the two arguments
$\min_x f(x)$	The minimum value of $f(x)$ for all values of x
$\max(.,.)$	The maximum of the two arguments
$\max_x f(x)$	The maximum value of $f(x)$ for all values of x
$(.)^T$	Transpose
$(.)^H$	Hermitian transpose
$\mathbf{I}_K$	A $K \times K$-dimensional identity matrix
$a_{i,j} = [\mathbf{A}]_{i,j}$	The entry of matrix $\mathbf{A}$ at row i and column j
$E\{.\}$	Expectation of the argument
$\text{diag}(a_1, a_2, \ldots, a_K)$	A diagonal matrix with the scalar input arguments, $a_1, a_2, \ldots, a_K$ as the diagonal elements

Roman Symbols

B	The number of bits in a Turbo-coded parity packet
E_T	The total energy available per symbol
E_s	The energy allocated per symbol
E_k	The energy allocated per symbol at channel k
J	The number of symbols in a Turbo-coded parity packet

K	The number of parallel coded-channels employed
M	The number of constellation points in a modulation scheme
N	The spreading factor
$\frac{N_0}{2}$	The single-sided noise power spectral density
N_U	The length of the source or parity packet
N_P	The number of the cyclic redundancy check bits
N_{p1}	The number of the first parity bits produced by a Turbo encoder in a packet
N_{p2}	The number of the second parity bits produced by a Turbo encoder in a packet
$N_{s,e,l}$	The total number of ones in vector $\vec{h}_{e,l}$
N_{re}	The maximum number of permissible retransmissions
N	The spreading factor of the employed signature sequences
P_b	The probability of packet error when one external extrinsic LLR is used
V	The number of source packets
Z	The maximum number of realizable parity packets using the coded parity packet approach
b_p	The p-th data rate in bits per symbol realizable by a modulation and coding scheme
d_{min}	The minimum distance between two constellation points
p	The index of the data rate chosen using an equal energy loading or equal signal-to-noise ratio loading
par	The number of parity bits generated per information bit
m	The number of channels loaded with a higher data rate when the two-group resource allocation is in place
r_{in}	The inner code rate
r_{out}	The outer code rate
y_k	The data rate in bits per symbol for channel k
$\vec{c}_e$	The e-th outer encoded vector
$\vec{d}_e$	The e-th inner encoded vector
$\vec{u}_v$	The v-th information source vector
$\vec{h}_{e,l}$	A vector which identifies the l-th unique subset of the *extrinsic* LLRs
$\vec{n}_e$	The e-th noise vector
$\vec{r}_e$	The e-th received symbol vector
$\vec{x}_e$	The e-th coded symbol vector

$\mathbf{A}$	The diagonal amplitude matrix for the transmitted symbols
$\mathbf{A}_{ext}^{e,l}$	The matrix whose columns represent the extrinsic log-likelihood ratio vectors corresponding to the l-th subset for the e-th parity packet
$\mathbf{C}$	The covariance matrix
$\mathbf{C}_k$	The covariance matrix for channel k when successive interference cancellation is implemented
$\mathbf{C}_{out}$	The outer encoded matrix
$\mathbf{G}$	The generator matrix for the coded parity packet approach
$\mathbf{H}_{ext}$	The extended parity check decoding matrix
$\mathbf{H}_{base}$	The base parity check matrix
$\mathbf{H}_e$	The recovery matrix used in hard decoding the failed packets
$\mathbf{J}$	The upshift/downshift matrix
$\mathbf{J}^{sel}$	The parity selector matrix to produce $\mathbf{G}$
$\mathbf{N}$	The additive white Gaussian noise matrix
$\mathbf{P}$	The parity generator matrix
$\mathbf{P}_{ord}$	The channel ordering matrix
$\mathbf{Q}_e$	The receiver signature sequence matrix which includes the inter-symbol interference components
$\mathbf{R}$	The received signal matrix
$\mathbf{U}$	The source information data matrix
$S_{m_i}^b$	A set of symbols whose the m_i-th bit that forms the symbol equals to b, where $b \in 0, 1$

Greek Symbols

β	The bit granularity
Γ	The gap value
$\Lambda()$	The soft channel output of the argument
$\Lambda_{apri}()$	The apriori log-likelihood ratio of the argument
$\Lambda_{ext}()$	The extrinsic log-likelihood ratio of the argument
$\Lambda_{Ext}()$	The external extrinsic log-likelihood ratio of the argument
$\Lambda_{apos}()$	The aposteriori log-likelihood ratio of the argument
ε	The number of parity packets
σ^2	The white Gaussian noise variance
$\vec{\nu}_v$	The v-th information source vector appended with a cyclic redundancy check byte
$\mathbf{\Lambda}_{ext}$	The extrinsic log-likelihood ratio matrix whose columns represent all extrinsic log-likelihood vectors

Superscript and Subscript

e	The parity packet index
j	The symbol index in the transmitted symbol packet
k	The index referring to the k-th channel
l	The external extrinsic log-likelihood ratio vector index
n	The information bit index in the source packet
p	The data rate index
v	The source packet index
z	The index of the decimal values converted to binary to form the parity selector matrix, $\mathbf{P}$

Acronyms

AMC	Adaptive modulation and coding
AWGN	Additive white Gaussian noise
BER	Bit error rate
BICM	Bit interleaved coded modulation
CDMA	Code division multiple access
CPP	Coded parity packet
CRC	Cyclic redundancy check
ER	Equal rate margin adaptive loading scheme
EREE	Equal rate and equal energy loading scheme
HDPIC	Hard decision parallel interference cancellation
HSDPA	High speed downlink packet access
ISI	Intersymbol interference
LC	Levin-Campello
LDPC	Low density parity check
LLR	Log-likelihood ratio
LTE	Long term evolution
MAI	Multiple access interference
MCS	Modulation and coding scheme
MIMO	Multiple-input and multiple-output
MMSE	Minimum mean square error
MTF	Multiple turbo-fountain
PER	Packet error rate
PIC	Parallel interference cancellation
QAM	Quadrature amplitude modulation
QoS	Quality of service
SIC	Successive interference cancellation
SNR	Signal-to-noise ratio
SINR	Signal-to-interference and noise ratio
UBRT	Union Bhattacharyya rate threshold
WCDMA	Wideband code division multiple access

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Background

High speed downlink packet access (HSDPA) [1], which is implemented over the wideband code division multiple access (WCDMA) networks, is one of the most widely adopted high speed broadband standards. Various features are provided to enhance the system throughput including adaptive and modulation coding (AMC) and multi-code transmission. To increase the practical achievable system throughput in order to improve performance of HSDPA systems is one of the main focal areas in the research community. It has been indicated in [2] that there is still a substantial room for improvement to bring the practical achievable system throughput closer to the theoretical upper bound. With the incorporation of additional features such as multiple antennas at both the transmitter and the receiver ends [1], any proposed designs to further improve performance of HSDPA systems should be computationally efficient whilst keeping the achievable system throughput relatively high. In this thesis, an integrated multi-code transmission system model, which incorporates a capacity-approaching channel coding scheme, an efficient resource allocation scheme and a successive interference cancellation scheme, is designed to address these challenges. In the next section, these main challenges as well as other related

research questions addressed in this thesis are described.

1.2 Research Problem Description

Consider a multi-code HSDPA communication system incorporated with a modulation and coding scheme (MCS) configurable to operate at a set of realizable data rates. For each of the rates, this research project attempts to design a modulation and coding method such that the gap value [3], which is the ratio between the required signal-to-noise ratio (SNR) and the minimum theoretical SNR to hit a target packet error rate, can be controlled by adjusting the number of alternative additional input information values for the decoding process without changing the original rate. This is realized by generating an adjustable number of source and parity packets at the transmitter end without changing the rate. The external extrinsic log-likelihood ratios, which are the additional input information values, are alternately fed into the decoding process until the received packet is successfully decoded or all external extrinsic log-likelihood ratios have been tested or fed into the decoder. For brevity, this MCS is called the coded parity packet (CPP) scheme. Reduced gap values yield lower energy requirement to hit a low target packet error rate, which is essential to reduce the number of retransmissions, hence the end-to-end delay, as well as the energy consumption in wireless ad-hoc networks. An example of a reduced gap value is shown in Figure 1.2. The gap value is reduced by more than 2 dB using the CPP coding scheme configured at 1.5 bits per symbol as opposed to the Turbo coding scheme configured at 1.33 bits per symbol, where both coding schemes were run with 16-QAM scheme. The gap value is reduced by increasing the adjustment parameter, α, which increases the number of alternative additional input information values for the iterative decoding process at the receiver end. Further explanation on this adjustment parameter and its effects on improving the decoding process and reducing the gap value is given on page 19 and page

25(The iterative decoding process).

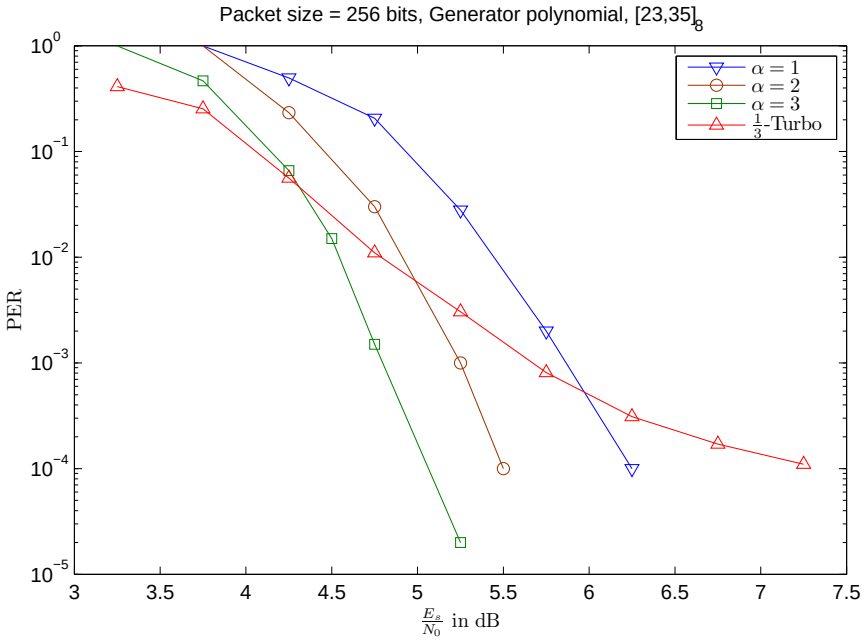

Figure 1.1: The gap value reduction using the CPP coding scheme

This proposed coding technique is integrated with a fixed signal-to-noise ratio (SNR) loading scheme which divides transmitting channels into two groups, each loaded with a distinct fixed SNR. As it loads two adjacent data rates to the employed code channels, this loading scheme is referred to as a two-group resource allocation scheme. With two groups of channels loaded with different fixed SNRs to realize two distinct data rates, the total transmission data rate is increased since a larger amount of energy is utilized to realize these data rates. When a set of realizable discrete data rates with relatively large bit granularities, which are the differences between two adjacent data rates, is provided, the two-group resource allocation scheme has been observed to produce an approximately equal system throughput as compared to that of the existing loading schemes, which are provided with a set of discrete data rates of large bit granularities.

The two-group resource allocation scheme is further improved by eliminating

any inverse matrix operations by employing a successive interference cancellation (SIC)-based energy calculation method. With this SIC-based energy calculation approach, the aforementioned CPP scheme can be applied to provide symbols including the past and the future symbols for the interference cancellation process. Eliminating matrix inversions is essential for practical implementation in order to further enhance performance of multi-code HSDPA systems. This computationally efficient SIC-based two-group resource allocation scheme may also be considered to be implemented in the multiple-input multiple-output (MIMO) HSDPA systems [1], which is more complicated than the existing single-input single-output (SISO) multi-code HSDPA systems. Apart from allocating two adjacent data rates over two groups of code channels, the total energy may also be equally distributed over each channel while loading an equal data rate to each channel corresponding to the average SNR of all channels. In order to control the gap value, the CPP scheme is applied to realize this equal data rate on each channel. Apart from controlling the gap value, this equal energy loading method is also computationally efficient as it requires no energy calculation.

In the next section, the research methodology adopted to solve these research problems is briefly described.

1.3 Research Methodology

The key steps applied to solve the aforementioned research problems are summarized as follows:

Research Focus

With the problems described in the previous section, the main focus of this research project is the reduction in the gap value achieved in the downlinks of

multi-code HSDPA communication systems. The reduced gap value is essential in order to minimize the required energy to transmit the data, hence enhance the total transmission data rate for a given total constrained energy.

Assumptions

In Chapter 2, a single-link white Gaussian channel is assumed for the implementatio of the proposed channel coding scheme. In Chapter 3, it is assumed that the channel side information including the channel impulse response and the noise variance is known to both the transmitter and the receiver. The noise is assumed to be the additive white Gaussian noise. To demonstrate the feasibility of implementing the proposed system design in practical systems, ISI and MAI components, which severely degrade the quality of the received signal, are also considered. As the channel side information might not be perfectly known in practice, the capacity-approaching CPP scheme is implemented with the proposed two-group resource allocation scheme in Chapter 3 and 4 to ensure that the received data are successfully decoded and detected although the received SNRs vary due to the imperfect knowledge of the channel side information, including the received SNR, and the noise variance. An alternative fixed energy loading scheme with the CPP method is also implemented in Chapter 4 when no channel side information is known.

All code channels in the multi-code HSDPA systems are dedicated to a single user as the focus of this thesis is the gap value reduction and not the resource scheduling. However, the design developed in this thesis may be extended for solving other research problems including the resource scheduling.

Solution Methods

Knowing the research problems, solution methods are developed based on the assumptions made. The solution methods include the CPP coding scheme, the

two-group resource allocation method and the SIC-based loading technique. In multi-code HSDPA systems, reducing the gap values and consumed energies as well as improving the total transmission data rates using these solution methods should consider the presence of both ISI and MAI components, which have been assumed earlier, in order to render the proposed design feasible in practice. Besides improving performance, the required computational complexity should also be monitored. The solution methods described in this thesis are also compared to the existing methods in solving the research problems addressed.

Results Production and Verification

The solution methods proposed in this thesis have been continually tested by different people under distinct assumptions to produce and verify the desired results. Furthermore, these results have also been compared to those of the existing and latest schemes. By doing multiple tests under different assumptions, the application of the proposed solution methods may be generalized to a wider range of areas in practice especially the improvement in the total transmission data rate over the downlinks of the multi-code HSDPA systems in the presence of ISI and MAI.

Revision Process

Apart from repetitively testing the proposed solution methods or designs, conference and journal publications have significantly contributed towards improving and rectifying the problems encountered in designing the proposed solution methods. The comments received during the paper revision process have given useful feedbacks to improve and verify the solutions.

1.4 Thesis Outline

In short, this thesis consists of the following chapters

1. Chapter 1: The first chapter describes the background, the research problems and the research methodolgy taken to complete the thesis.

2. Chapter 2: The second chapter gives an account of the CPP scheme, which reduces the gap values without changing the data rate. This reduction is further applied for improving the two-group resource allocation scheme described in Chapter 3.

3. Chapter 3: The third chapter presents a two-group resource allocation scheme which is integrated with the CPP scheme to minimize the required energy to load the data rates, and hence increase the total transmission data rate.

4. Chapter 4: This chapter enhances the two-group loading method in the previous chapter by further reducing the required energy when successive interference cancellation is implemented to load the data rates whilst keeping the computational complexity low.

5. Chapter 5: The last chapter concludes the thesis and highlights a number of future directions to further improve and apply the solution methods developed in the thesis.

1.5 List of Contributions

The contributions made by completing this research project are listed as follows

1. A capacity-approaching coding scheme referred to as CPP coding has been developed, which is able to reduce the gap value achieved using

the existing Turbo codes by more than 2dB when small packet sizes of around 100 bits are transmitted at data rates between 1 to 2 bits per symbol. (Chapter 2)

2. A two-group resource allocation scheme, which is implemented with large bit granularities, that produces an approximately equal system throughput to that of the existing equal rate margin adaptive loading scheme, which is implemented with small bit granularities. (Chapter 3)

3. Improvement of approximately 5 dB achieved by the two-group resource allocation scheme as compared to that of the equal rate and equal energy loading scheme. (Chapter 3)

4. Reduction in the amount of energy to realize the target total data rate using an SIC-based two-group resource allocation scheme is used. (Chapter 4)

5. Reduction in the computational complexity using the SIC-based energy calculation method, which requires no matrix inversions. (Chapter 4)

6. Reduction in the gap value when a fixed energy loading is incorporated into the CPP coding scheme for multi-code systems. When compared with existing schemes, the achieved gap value is observed to be lower by up to 2 dB. (Chapter 4)

The final aim of this thesis is to design a scalable and computationally efficient multi-code transmission system model, which is proposed as an alternative to further improve HSDPA systems. Since the current practical achievable system throughput is still lower than the theoretical upper bound, an integrated transmission system which combines the proposed packet-based channel coding scheme with the two-group resource allocation technique and the SIC scheme has been developed in this thesis as a possible solution which might be considered to be incorporated in the future version of the multi-code HSDPA systems.

Realizing that the number of matrix inversions required is a serious issue in the resource allocation as well as the signal detection process, the proposed integrated system, which employs the SIC-based energy calculation technique, eliminates these matrix inversions whilst maintaining and improving the achievable system throughput. This computationally-efficient system may also be considered as a solution for improving the latest broadband technologies such as MIMO HSDPA and long term evolution (LTE) systems.

CHAPTER 2

Coded Parity Packet

2.1 Introduction

The increasing demand for high-speed wireless broadband technologies stimulates much interest in both the research community as well as industry. Various communication models and products have been developed and proposed to improve system performance for serving this growing need. One of the key measures to gauge system performance is the gap value [3], which is defined as the difference in dB between the theoretical signal-to-noise ratio (SNR) achievable from Shannon's equation [4] for a given data rate in bits per symbol and the experimental SNR required by a system under consideration in order to achieve the same data rate at a target error rate.

For reducing the gap value at a given data rate, the transmission system model must be designed with the capability of successfully transmitting and receiving data at the particular data rate for a given target error rate. This must be achieved by using as low an SNR as possible, which is lower bounded by the theoretical SNR calculated from Shannon's equation.

One of the commonly adopted methods used to improve performance of the transmission system model by reducing the gap value is channel coding. Various

channel coding techniques are proposed in literature, which mainly classified as block and convolutional codes [5]. The block codes are produced from linear combinations of the information sequences, whereas the convolutional codes are generated from the sequences of codes in both the current and the previous information sequences. However, it has been observed that most of the existing codes perform well at certain range of code rates. Turbo codes [6], which are commonly regarded as capacity-approaching convolutional codes, have been demonstrated to reduce gap values at low code rates although they suffer from residual error floor problems. At high code rates, block codes such as low density parity check (LDPC) codes [7] are reported to perform better whilst reducing computational costs. To reap the benefits of both block and convolutional codes, concatenated codes are also designed in literature, such as in [8].

Reduced gap values at various data rates are essential to ensure that low target packet error rates are achievable with relatively low required energies. In wireless ad-hoc networks, a low target packet error rate is important to reduce retransmissions, which will shorten the end-to-end delay. Furthermore, the low energy requirement renders lower energy consumption, which can also be reduced as the number of retransmissions decreases.

In this chapter, a concatenated channel coding scheme which combines a simple block coding technique with a standard Turbo coding scheme is developed to significantly reduce the gap values, rendering lower energy requirements for packet transmission in wireless ad-hoc networks.

2.2 Related Work

Some of the most commonly studied and implemented capacity-approaching coding schemes which are Turbo codes [6], LDPC codes [7] and repeat accumulate (RA) codes [9]. Most of these codes are chosen as the standard forward error correction (FEC) techniques and incorporated into modulation schemes

such as quadrature amplitude modulation (QAM) schemes in current wireless communication systems including the wireless broadband code division multiple access (CDMA) systems.

In this chapter, channel coding schemes especially concatenated coding techniques [8] based on Turbo codes are studied in order to improve coding performance and reduce the gap value. Concatenating Turbo codes with other channel codes is mainly motivated by the need for eliminating the error floor problem experienced by the standard Turbo coding schemes at considerably high SNRs. Since the focus of this chapter is on Turbo-based coding schemes, less emphasis is given on other channel coding schemes such as LDPC and RA codes. Turbo codes have been extensively studied in literature, yielding numerous variants of modulation and coding schemes (MCSs) developed and proposed to improve system performance. A commonly adopted design of Turbo-based MCSs was developed by concatenating a Turbo coding scheme with a modulation scheme, such as in [10], for instance. These MCSs can be further upgraded by incorporating interleavers to avoid burst errors on the received bits, as presented in a system known as the bit-interleaved coded modulation (BICM) scheme [11–15]. Alternatively, Turbo codes can also be combined with modulation schemes, as implemented by Ungerboeck [16, 17], which are further modified and enhanced in [18–21].

In addition to the aforementioned designs, the applications of Turbo codes are also extended in developing multi-level coding schemes [22, 23] such as in [24]. Turbo codes are reported to produce low gap values especially at low code rates, but suffer from a residual error floor problem when the SNR is considerably high.

Using a random puncturing method, a Turbo-based design known as variable-rate Turbo-BICM scheme was developed in [14]. The corresponding achievable capacity by this scheme was modelled using a Union Bhattacharyya Rate

Threshold (UBRT) equation, which was generically produced from [3] based on the Shannon's equation [4].

The concept of message passing or belief propagation, as commonly employed in LDPC codes, was implemented in [25] to improve the corresponding iterative soft decoding performance on the receiver end in order to reduce the gap value. The 'messages' or the extrinsic log-likelihood ratios passed during the decoding process are generated from every received parity bits as well as the received systematic bit on the receiver end. Although satisfactory performance is reported, this so-called Multiple Turbo-Fountain (MTF) coding scheme is limited to low code rates, in the form of $1/(1+par)$, where par denotes the number of parity bits generated along with an information bit during the encoding operation. Furthermore, the problem of residual error floor commonly observed in Turbo codes is also inherent in this coding scheme.

A possible alternative of realizing multiple distinct discrete data rates is by concatenating different codes [8], which generally produces longer codes. The simplest form of this type of coding is a concatenation of two codes known as outer and inner codes. Capacity-approaching codes such as Turbo codes can be used as one of the codes being concatenated. Turbo-Hadamard codes developed in [26–28] for example, were implemented by concatenating Hadamard and Turbo codes. This scheme, although achieving low gap values, generates very long codes, which requires high computational loads. Another variant of Turbo-based concatenated codes is proposed in [29], which also achieves low gap values. However, the code rates realized are also limited and the required computational cost to perform the iterative decoding process is high.

In this chapter, a new method of realizing multiple distinct data rates at relatively low gap values is proposed. An efficient parity packet generation approach is developed to produce an adjustable number of source and parity packets, each of which is further inner encoded using a standard Turbo encoder. With-

out changing the code rate, the number of these packets can be adjusted using an adjustment parameter α to control the number $L_\varepsilon = 2^{\alpha(\varepsilon_o - V_o)-1}$ of subsets of the associated packets, given the initial number of source and parity packets as V_o and $\varepsilon_o - V_o$ respectively. These subsets of associated packets, which are received on the receiver end, are used to produce a number of alternative additional input information values known as external extrinsic log likelihood ratios (LLRs) for the iterative decoding process. Furthermore, a simple hard decoding technique can also be invoked based on these subsets to recover the failed source packets. Tests run at small-sized packets using this proposed coding scheme demonstrate that significant gap value reduction is achieved. This reduction is highly applicable to reduce energy consumption while keeping the packet error rate low [30] to reduce the number of retransmissions and end-to-end delay in wireless ad-hoc networks.

The rest of this chapter is organized as follows: the next section describes the system model in which the proposed MCS is implemented. A detailed account of the encoding scheme developed is given in Section 2.4.

2.3 System Model

Consider a model of a single-link communication system implemented between two interacting nodes in a wireless ad-hoc network, where a packet-based concatenated coding scheme comprising an outer and inner coding scheme is incorporated, as appears in Figure 2.1. The information sequences intended for transmission are generated in blocks of data represented as a $(K \times 1)$-dimensional vector $\vec{u}$, each element of which is denoted as $u_k \in \{\pm 1\}$ for $k = 1, \ldots, K$. Each of the data blocks is further segmented into $((N_U - N_P) \times 1)$-dimensional vectors $\vec{u}_v$ for $v = 1, \ldots, V$. Then, an outer encoder is used to encode these packets to yield ε packets, which are denoted as $((N_U - N_P) \times 1)$-dimensional vectors $\vec{c}_e$, before being appended with cyclic redundancy check (CRC) bytes,

each of which having N_P bits, to yield $(N_U \times 1)$-dimensional vectors, $\vec{\nu}_e$. These CRC-appended packets are then inner-encoded to produce $(B \times 1)$-dimensional vectors $\vec{d}_e$. Each of these so-called coded parity packets is modulated to generate $(J \times 1)$-dimensional packets of $\vec{x}_e$, which are transmitted over an additive white Gaussian noise channel, rendering a noise-corrupted received packet of $\vec{r}_e$ for $e = 1, \ldots, \varepsilon$.

On the receiver end, the received packet, $\vec{r}_e$ is first demodulated before being passed on to the decoder. Since a soft-in hard-out decoding process is used, a $(B \times 1)$-dimensional channel soft output vector, $\Lambda(\vec{d}_e)$, needs to be determined from each received packet $\vec{r}_e$ to be fed into the decoding process. The channel soft output is a type of log-likelihood ratio, representing the reliability of the received packet. This soft information is fed into the soft-in-hard-out iterative soft decoding process to perform the actual decoding operation, from which the estimate of the $(N_U \times 1)$-dimensional cyclic redundancy check (CRC)-appended parity packet $\hat{\vec{\nu}}_e$ is made. Hard decoding is carried out based on the succesful packets to recover the failed packets, if any.

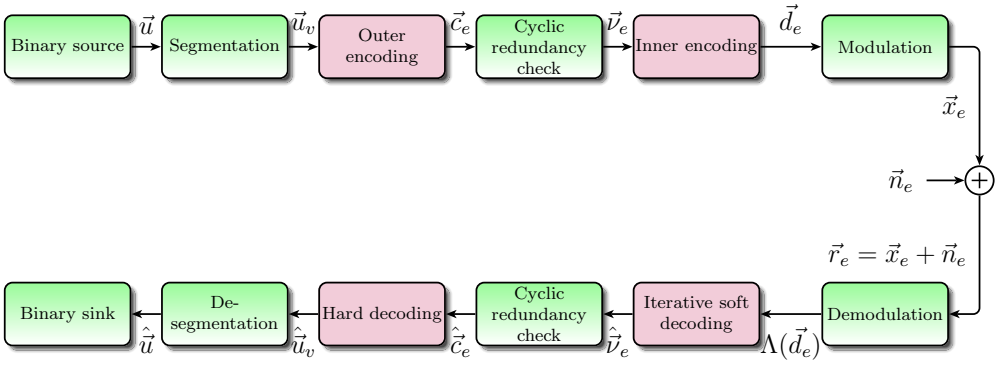

Figure 2.1: The proposed MCS model

As highlighted in Figure 2.1, the focus of this chapter is to describe the encoding and decoding designs developed with the aim of achieving reduced gap values. In the next section, the design of the proposed encoding process is presented.

2.4 The Encoding Process

The methods presented in this section and most of the following sections are based on a manuscript presented in [31], which has been submitted to European Transactions on Telecommunications and refined from the original document submitted to IEEE Transactions on Communications. As can be seen in Figure 2.1, the proposed encoding scheme consists of two components; the outer encoding module and the inner encoding module. The outer encoding module is a packet-based encoding scheme which generates parity packets by combining the source packets, as will be described in the following section. In this section, a technique of selecting and generating $\varepsilon - V$ parity packets from V source packets is developed such that $2^{\varepsilon-V-1}$ distinct subsets of associated packets can be generated to produce $L_\varepsilon = 2^{\varepsilon-V-1}$ external extrinsic LLRs for each source packet received at the receiver end. Then, an adjustment parameter α is introduced such that the number of subsets is controlled to be $L_\varepsilon = 2^{\alpha(\varepsilon_o-V_o)-1}$, where V_o and $\varepsilon_o - V_o$ are the initial number of source and parity packets respectively and $V = \alpha V_o$, $\varepsilon = \alpha\varepsilon_o$. As the number L_ε of external extrinsic LLRs increases, the probability of successfully decoding a received packet also increases as each of these LLRs can be alternately fed into the decoding process to recover the received packet.

Figure 2.2 shows the design of the transmitter, developed on the basis of the proposed encoding scheme. The account of this encoding scheme, which is indicated inside a closed dotted line in the figure, will be presented in the next subsection.

2.4.1 The Outer Encoding Module

The proposed outer encoder is designed to generate ε packets $[\vec{c}_1, \ldots, \vec{c}_e, \ldots, \vec{c}_\varepsilon] = \mathbf{C}_{out}$, each of which with a dimension of $((N_U - N_P) \times 1)$, from V source pack-

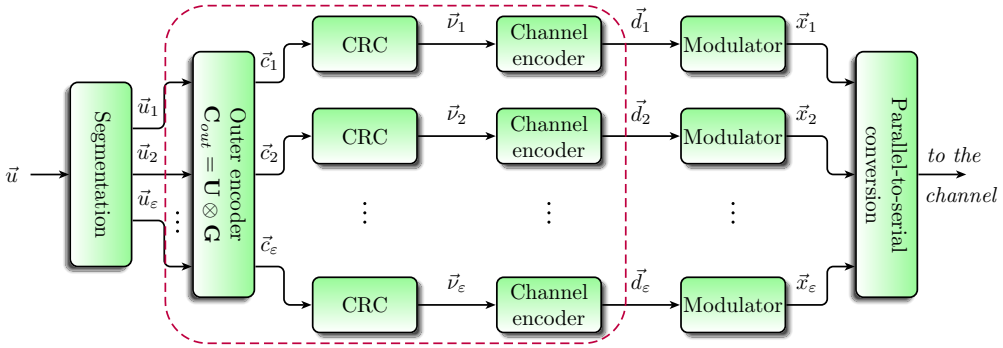

Figure 2.2: The transmitter design with the proposed encoding scheme

ets $[\vec{c}_1, \ldots, \vec{c}_v, \ldots, \vec{c}_V]$ as follows

$$\mathbf{C}_{out} = \mathbf{U} \otimes \mathbf{G}, \tag{2.1}$$

where $\mathbf{U} = [\vec{u}_1, \ldots, \vec{u}_v, \ldots, \vec{u}_V]$ is an $((N_U - N_P) \times V)$-dimensional matrix representing the block of input information data and $\otimes$ is the element-by-element vector or matrix multiplication and bit-wise modulo-2 addition operation. The $(V \times \varepsilon)$-dimensional matrix $\mathbf{G}$ is the CPP generator matrix used to produce $\varepsilon - V$ desired parity packets by bit-wise XOR combining two or more of the V source packets, in addition to the existing V source packets. When a parity packet is generated from d source packets, this parity packet is referred to as a packet of degree d. In simpler terms, the degree d of a particular packet signifies the number of packets called neighbours which form the packet through a bit-wise XOR combination between all of them.

With V source packets, there are a maximum of $2^V - V - 1$ parity packets which can be generated through this encoding technique, in addition to the source packets. Therefore, a generator matrix which can produce a maximum of $2^V - 1$ parity packets including the source packets can be defined as

$$\mathbf{G}^{(m)} = \begin{bmatrix} \mathbf{I}_V & \mathbf{P} \end{bmatrix}, \tag{2.2}$$

where $\mathbf{I}_V$ is the $(V \times V)$-dimensional identity matrix which reproduces all the source packets and $\mathbf{P}$ is the $(V \times Z)$-dimensional matrix that generates all $Z = 2^V - V - 1$ parity packets using Eq(2.1). This matrix, $\mathbf{P}$, is called a parity generator matrix, and has columns defined as follows

$$\vec{p}_z = \text{bin}_V(D), \tag{2.3}$$

where bin_V is a decimal to binary operator which converts a decimal number to its equivalent $(V \times 1)$-dimensional binary column vector where the first or the uppermost element is the least significant bit and the last element is the most significant bit. Variable D is a decimal number that takes values from 3 to $2^V - 1$, except those which are equivalent to the power of two. This is because all decimal numbers which are a power of two have been converted to binary and form the identity matrix, $\mathbf{I}_V$.

In this research project, a novel technique is proposed to adjust the number $\varepsilon - V$ of parity packets generated by selecting $\varepsilon - V$ columns in the parity generator matrix, $\mathbf{P}$. For this reason, a $(Z \times (\varepsilon - V))$-dimensional parity selector matrix, $\mathbf{J}^{(sel)}$, is introduced, where each column in this matrix has elements equal to zero except for one element, the index of which equals to the index of the desired column in $\mathbf{P}$. As a result, the generator matrix, $\mathbf{G}$ on the right hand side of Eq(2.1) can now be written as

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_V & \mathbf{P}\mathbf{J}^{(sel)} \end{bmatrix}, \tag{2.4}$$

where $1 < V < \varepsilon \leq Z + V$. The minimum value ε can take is $V + 1$ in order to ensure that at least one parity packet is generated. The $\varepsilon - V$ parity packets are essential in providing alternative additional input information values for decoding the V source packets. If the parity packets are properly selected and generated using the $\mathbf{P}\mathbf{J}^{(sel)}$, all ε outer-encoded packets can be grouped into L_ε distinct subsets, from which the additional input information values can be

generated.

The number L_ε of the packet subsets produced for each received source packet on the receiver end is equal to $L_\varepsilon = 2^{\varepsilon - V - 1}$ for a given number V of source packets from which $\varepsilon - V$ parity packets are selected and generated such that the weight of each row in the matrix product $\mathbf{PJ}^{(sel)}$ is greater than zero. In order to control L_ε, which affects the decoding process, an adjustment parameter is introduced as

$$\alpha = \frac{a}{g_o}, \tag{2.5}$$

to yield the new number of source and parity packets as $V = \alpha V_o$ and $\varepsilon - V = \alpha(\varepsilon_o - V_o)$ such that $L_\varepsilon = 2^{\alpha(\varepsilon_o - V_o) - 1}$, where g_o is the greatest common divisor between V_o and ε_o while a is any integer larger than or equal to g_o, given the initial number of source and parity packets as V_o and $\varepsilon_o - V_o$ respectively.

Therefore, the significance of adjusting the parameter α is that it controls the number, $\varepsilon - V = \alpha(\varepsilon_o - V_o)$, of the parity packets, from which more alternative additional input information values are produced and fed into the iterative Turbo decoders to improve the decoding process when these parity packets are received at the receiver end.

Since V and ε are integers, representing the packet numbers, the adjustment parameter α should be determined such that αV_o and $\alpha \varepsilon_o$ take all possible integers. If the simplest fractional form of the outer code rate is given as $r_{out} = \frac{n_o}{d_o}$, then the initial numbers of the packets can be expressed as $V_o = g_o n_o$ and $\varepsilon_o = g_o d_o$ since g_o is the greatest common divisor, where the initial number V_o of the source packets must satisfy

$$2^{V_o} - 1 \geq \varepsilon_o, \tag{2.6}$$

$$2^{V_o} - 1 \geq \frac{d_o}{n_o} V_o, \tag{2.7}$$

$$2^{V_o} - \frac{d_o}{n_o} V_o \geq 1. \tag{2.8}$$

With this condition and without loss of generality, the outer code rate can be further expressed as

$$r_{out} = \frac{g_o n_o}{g_o d_o} = \frac{\left(\frac{g_o+1}{g_o}\right) g_o n_o}{\left(\frac{g_o+1}{g_o}\right) g_o d_o} = \frac{\left(\frac{g_o+2}{g_o}\right) g_o n_o}{\left(\frac{g_o+2}{g_o}\right) g_o d_o} = \cdots = \frac{\alpha V_o}{\alpha \varepsilon_o} = \frac{V}{\varepsilon}, \tag{2.9}$$

where the adjustment parameter is deduced as

$$\alpha = \frac{a}{g_o}, \tag{2.10}$$

for all integer values of a given as $a = g_o, g_o + 1, g_o + 2, \ldots$.

For a special case of integer-valued α, the possible values of L_ε are produced and tabulated in Table 2.1 for different initial numbers of source and parity packets, V_o and $\varepsilon_o - V_o$ respectively.

α	The number of subsets, L_ε				
	$\varepsilon_o - V_o = 1$	$\varepsilon_o - V_o = 2$	$\varepsilon_o - V_o = 3$	$\ldots$	$\varepsilon_o - V_o > 0$
1	1	2	4	$\ldots$	$2^{\varepsilon_o - V_o - 1}$
2	2	8	32	$\ldots$	$2^{2(\varepsilon_o - V_o)-1}$
3	4	32	256	$\ldots$	$2^{3(\varepsilon_o - V_o)-1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ldots$	$\vdots$
α	$2^{\alpha-1}$	$2^{2\alpha-1}$	$2^{3\alpha-1}$	$\ldots$	$2^{\alpha(\varepsilon_o - V_o)-1}$ $= 2^{\varepsilon - V - 1}$

Table 2.1: The adjustment of the number of the subsets of the parity packets

Therefore, the $\varepsilon - V$ parity packets generated in addition to the V source packets should be selected such that every row in the matrix product, $\mathbf{PJ}^{(sel)}$, has a

minimum weight of one. As the number of columns in the parity generator matrix, $\mathbf{P}$, grows when the number of source packets, V, increases; the number of possible options in selecting $\varepsilon - V$ parity packets which satisfy this condition also increases. In order to facilitate the selection process, a new $(Z \times (\varepsilon - V))$-dimensional parity selector matrix, $\mathbf{J}^{(sel)}$, is defined as follows.

$$\mathbf{J}^{(sel)} \triangleq \begin{bmatrix} \mathbf{0}_{(Z-(\varepsilon-V))\times(\varepsilon-V)} \\ \mathbf{I}_{\varepsilon-V} \end{bmatrix}, \tag{2.11}$$

where $\mathbf{0}_{(Z-(\varepsilon-V))\times(\varepsilon-V)}$ is a $((Z-(\varepsilon-V)) \times (\varepsilon-V))$-dimensional zero matrix and $\mathbf{I}_{\varepsilon-V}$ is an $((\varepsilon-V) \times (\varepsilon-V))$-dimensional identity matrix.

This parity selector matrix, $\mathbf{J}^{(sel)}$, as defined in Eq(2.11), selects the last $\varepsilon - V$ columns in the parity generator matrix, $\mathbf{P}$. This is because matrix $\mathbf{P}$ is produced from a sequence of increasing decimal numbers, D, as defined in Eq(2.3), where the last column vector, $\vec{p}_Z$, is a column vector having all elements equal to one, $\vec{p}_Z = [1, 1, \ldots, 1]^T$. Since $\vec{p}_Z$ is a one or unity vector, the weight of each row in $\mathbf{PJ}^{(sel)}$ is now greater than or equal to one. Therefore, the column vectors from which the remaining $\varepsilon - V - 1$ parity packets are generated can be selected from the rest of the column vectors in matrix $\mathbf{P}$. One of the most convenient ways of choosing these column vectors is by selecting the column vectors next to $\vec{p}_Z$, as defined in Eq(2.11). As a result, the resulting matrix product $\mathbf{PJ}^{(sel)}$ in Eq(2.4) can be simplified as follows

$$\mathbf{PJ}^{(sel)} = \begin{bmatrix} \vec{p}_{Z-(\varepsilon-V-1)} & \cdots & \vec{p}_{Z-1} & \vec{p}_Z \end{bmatrix}. \tag{2.12}$$

It can be seen in Eq(2.11) that if all possible parity packets are to be generated, the parity selector matrix $\mathbf{J}^{(sel)}$ becomes a $(Z \times Z)$-dimensional identity matrix, where $\varepsilon - V = Z$. Consequently, the matrix product resulted is $\mathbf{PJ}^{(sel)} = \mathbf{P}$, which is equal to the original parity generator matrix.

With the expression in Eq(2.12), the generator matrix equation given in Eq(2.4)

can also be further simplified and rewritten as

$$\mathbf{G} = \left[\begin{array}{ccccccccc} \vec{g}_1 & \dots & \vec{g}_{V-1} & \vec{g}_V & \vec{p}_{Z-(\varepsilon-V-1)} & \dots & \vec{p}_{Z-1} & \vec{p}_Z \end{array} \right], \tag{2.13}$$

where

$$\left[\begin{array}{cccc} \vec{g}_1 & \dots & \vec{g}_{V-1} & \vec{g}_V \end{array} \right] = \mathbf{I}_V. \tag{2.14}$$

Therefore, with the generator matrix G, as redefined in Eq(2.13), the ε parity packets, $\vec{\nu}_1, \dots, \vec{\nu}_\varepsilon$ are generated using Eq(2.1). These parity packets are then passed on to the inner encoders, which are the Turbo encoders, as described in the next subsection.

2.4.2 The Inner Encoder

Each of the ε parity packets is inner encoded by a standard Turbo encoder [6] of rate, $r_{in} = N_U/B$, where N_U is the length of the source or parity packet and B is the length of the Turbo coded packet. For notational convenience, these Turbo-coded packets are termed as coded parity packets and denoted as $(B \times 1)$-dimensional vectors, $\vec{d}_e$. The two recursive systematic code (RSC) encoders, which are the principal components in a standard Turbo encoder, have a constraint length of $K_m = 5$ with the polynomial generator of $[23, 35]_8$. The output coded bit vector of this encoder, $\vec{d}_e$, comprises of two main types - the systematic bits and the parity bits. Each systematic bit, which is the actual information bit, is denoted as $d_{e,n}^{(s)} = d_{e,n} = \nu_{e,n}$. Parity bits are formed by the first and the second parity bits, represented as $d_{e,n_{p1}}^{(p1)}$ and $d_{e,n_{p2}}^{(p2)}$ respectively, where $n = 1, \dots, N_U$, $n_{p1} = 1, \dots, N_{p1}$, $n_{p2} = 1, \dots, N_{p2}$ and $N_U + N_{p1} + N_{p2} = B$.

The number of both parity bits, N_{p1} and N_{p2}, determined to be transmitted along with the systematic bits depends on the puncturing operation performed to vary the code rate. A standard puncturing operation for Turbo codes as

given in [10] was employed in this research project in order to realize a number of typical standard Turbo code rates including $1/2, 2/3$ and $3/4$.

A summarized list of steps taken to produce these coded parity packets, $\vec{d}_1, \ldots, \vec{d}_\varepsilon$ is given in the flow chart shown in Figure 2.3. The last process

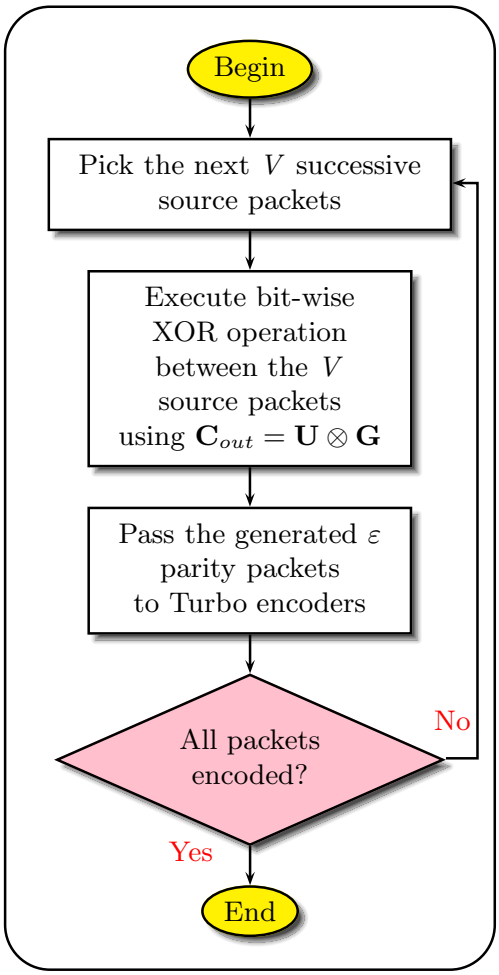

Figure 2.3: The summarized algorithm for the proposed CPP encoding scheme

required before transmission is modulation, which is briefly explained next.

2.4.3 Modulation

As can be observed in Figure 2.2 on page 17, an M-Quadrature Amplitude Modulation (QAM) scheme is used to modulate the coded parity packets produced from the proposed coding scheme. The length of the coded parity packets, B, is

set such that $\text{mod}(N_U, \log_2 M) = 0$, where the coded parity packets can be divided into J strings of coded bits, $\vec{d}_{e,j} = \left[d_{e,j,1}, \ldots, d_{e,j,\log_2 M}\right]^T$ for $j = 1, \ldots, J$ and $J = B/\log_2 M$. Hence, each coded parity packet can be represented as $\vec{d}_e = \left[\vec{d}_{e,1}^T, \ldots, \vec{d}_{e,j}^T, \ldots, \vec{d}_{e,J}^T\right]^T$.

These strings of coded parity bits are Gray mapped [11, 21] to choose the appropriate constellation points in order to generate the modulation symbols, $\vec{x}_e = [x_{e,1}, \ldots, x_{e,j}, \ldots, x_{e,J}]^T$, for transmission. A simple method of Gray mapping, as shown in [32] for instance, can be used for this purpose. Figure 2.4 shows an example of Gray-mapping performed for 16-QAM symbols.

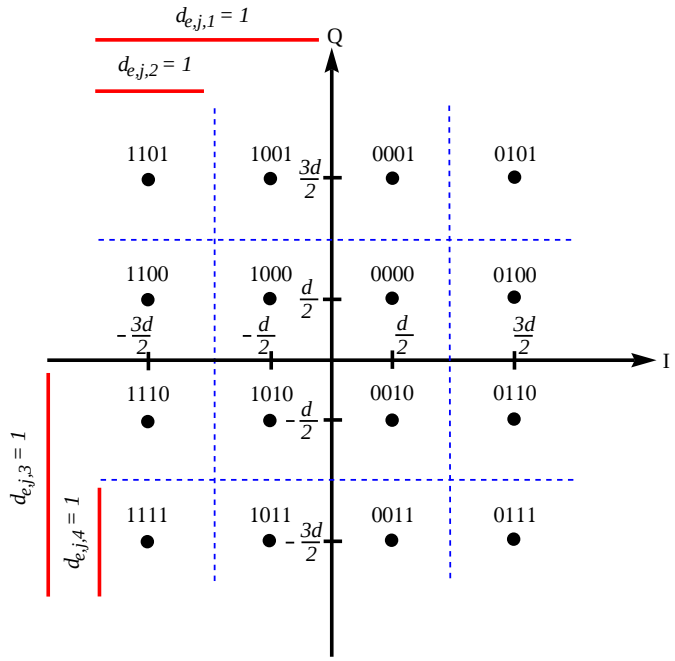

Figure 2.4: Gray mapping for 16-QAM symbols

From Figure 2.4 it can be seen that bit d_{e,j,m_i} of string $\vec{d}_{e,j}$, which forms symbol $x_{e,j}$, is equal to one if it falls below the red bar indicated in this figure, where $m_i = 1, \ldots, \log_2 M$. Otherwise, d_{e,j,m_i} will be equal to zero. The same pattern or method can be extended and applied to the other constellation sizes such as $M = 64$ and $M = 256$.

It can also be noted that there is no interleaving performed between the coding and the modulation processes of the proposed scheme. This is different to the Turbo-BICM approach [13], which performs interleaving on the coded bits before the modulation process. Based on the results produced and shown in Figure 2.14 on page 47, the packet error rates produced by the proposed CPP scheme are almost as good as those of the Turbo-BICM approach.

Once the modulation process is completed, the symbols are ready for transmission. Based on the encoding process presented in this current section, its decoding counterpart at the receiver end will be fully described in the next section.

2.5 The Decoding Process

At the receiver end, each transmitted packet, $\vec{x}_e$, is corrupted by an additive white Gaussian noise vector $\vec{n}_e$, which has a zero mean and a variance of $\sigma^2 = N_0$. The noise-corrupted received packet is then represented as

$$\vec{r}_e = \vec{x}_e + \vec{n}_e. \tag{2.15}$$

With ε received packets $\vec{r}_1, \ldots, \vec{r}_e, \ldots, \vec{r}_\varepsilon$, this section presents an account of a novel decoding process which utilizes ε extrinsic LLRs generated from decoding each of the ε packets to produce $2^{\varepsilon - V - 1}$ distinct external extrinsic LLRs for improving the input of the decoding process for each received packet.

If the average symbol energy used to transmit a CPP coded M-QAM symbol is denoted as

$$E_s = \frac{d^2 (M-1)}{6}, \tag{2.16}$$

then the received SNR can be expressed as $\text{SNR}_R = \frac{E_s}{N_0}$, where d_{min} is the minimum distance between the M constellation points. It is assumed that no

intersymbol interference (ISI) is present for this case. A general block diagram of the receiver model, with the proposed iterative soft decoding scheme, is given in Figure 2.5.

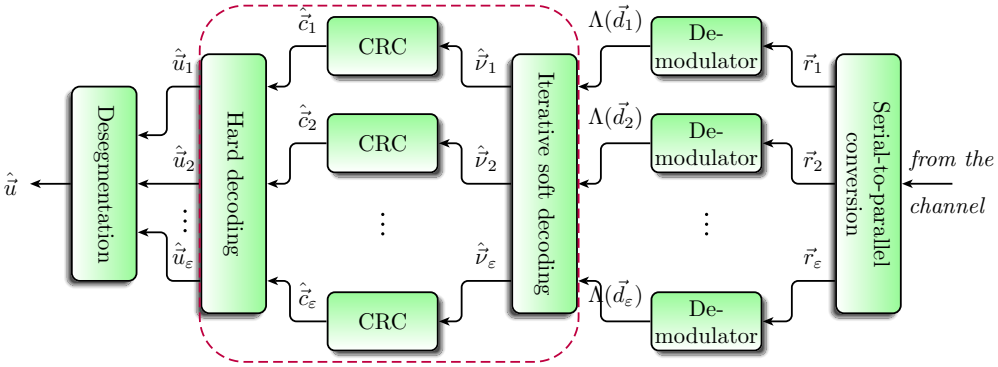

Figure 2.5: The block diagram of the proposed receiver model

Upon receiving the ε packets, the first operation carried out after separating these ε packets in order to process each packet in parallel is to measure the reliability of each bit which is mapped to each transmitted symbol. For QAM symbols, the well-established log-likelihood ratio approach [32] can be used to calculate this parameter. For notational convenience, this reliability measurement is referred to in this paper as the channel soft output, $\Lambda(d_{e,j,m_i})$, which is written as follows [32]

$$\Lambda(d_{e,j,m_i}) = \frac{1}{N_0}\left(\min_{\varsigma_0 \in S^0_{m_i}} \|r_{e,j} - \varsigma_0\|^2 - \min_{\varsigma_1 \in S^1_{m_i}} \|r_{e,j} - \varsigma_1\|^2\right), \tag{2.17}$$

where $S^0_{m_i}$ represents a set of symbols with $d_{e,j,m_i} = 0$ and $S^1_{m_i}$ denotes another set of symbols with $d_{e,j,m_i} = 1$ for $m_i = 1, \ldots, \log_2 M$. The demodulation process, which produces the channel soft output, is indicated in the second block diagram from the right of Figure 2.5. With Eq(2.17), the soft channel output vector can be represented as $\Lambda(\vec{d}_e) = \left[\Lambda(\vec{d}_{e,1})^T, \ldots, \Lambda(\vec{d}_{e,j})^T, \ldots, \Lambda(\vec{d}_{e,J})^T\right]^T$.

As can be seen in Figure 2.5, the channel soft output vectors are required in the proposed iterative soft decoding process to decode the received packets. For

each of the ε received packets, a standard Turbo decoder which uses the same constraint length and the polynomial generator, as mentioned in the previous section is employed to decode and detect the received packet. This method is different from the technique implemented in [33], which employs several Turbo decoders to decode each received packet subject to the number of parity packets generated and transmitted.

In order to process by using a standard Turbo decoder which consists of two component decoders, a soft channel output vector is demultiplexed into two main parts. The first part corresponds to a sequence of the soft channel output of the systematic bits, $\Lambda(d_{e,n}^{(s)})$, which are passed on to both component decoders. The second part comprises of two sequences. The first is a sequence of the soft channel output of the first parity bits, $\Lambda(d_{e,n}^{(p1)})$, which is passed on to the first component decoder. The second sequence is the soft channel outputs of the second parity bits, $\Lambda(d_{e,n}^{(p2)})$, which is passed on to the second component decoder. As the number of parity bits, either the first or the second, may not necessarily be similar to the number of systematic bits due to the puncturing operation during the Turbo encoding process, the soft channel output bits for the parity bits are defined as follows

$$\Lambda(d_{e,n}^{(pc)}) = \begin{cases} 0 & \text{if the } n\text{th parity bit } pc \text{ is punctured,} \\ \Lambda(d_{e,n_{pc}}^{(pc)}) & \text{if } n_{pc} + nop_{pc}(n) = n, \end{cases} \tag{2.18}$$

where $pc \in \{p1, p2\}$ denotes the type of the parity bit and $nop_{pc}(n)$ is the total number of parity bit pc punctured after the Turbo encoding process for the first n systematic bits.

2.5.1 *Apriori* LLR Generation

The soft channel outputs of the systematic bits are added to the extrinsic LLR as well as the external extrinsic LLR to produce the *apriori* LLR as follows

$$\Lambda_{apri}(d_{e,n}^{(1)}) = \Lambda_{ext}(d_{e,n}^{(2)}) + \Lambda(d_{e,n}^{(s)}) + \Lambda_{ext}^{\varepsilon}(d_{e,n}), \quad (2.19)$$

$$\Lambda_{apri}(d_{e,n}^{(2)}) = \Lambda_{ext}(d_{e,n}^{(1)}) + \Lambda(d_{e,n}^{(s)}) + \Lambda_{ext}^{\varepsilon}(d_{e,n}), \quad (2.20)$$

where the superscripts (1) and (2) refer to the component decoders, one and two, respectively. It can be seen that the soft channel output of the systematic bit $\Lambda(d_{e,n}^{(s)})$ is needed for both component decoders. The extrinsic LLRs, $\Lambda_{ext}(d_{e,n}^{(2)})$ and $\Lambda_{ext}(d_{e,n}^{(1)})$, as well as the external extrinsic LLR $\Lambda_{ext}(d_{e,n}^{\varepsilon})$ are produced from the previous iteration of the Turbo decoders. Therefore, these two types of LLR are only available after the first decoding iteration is completed and hence are zero at the first iteration.

When the *apriori* LLRs are fed as the input into each of the component decoders in the Turbo decoder, a soft decoding process is internally performed inside the component decoders to estimate the soft bit of the actually transmitted bits. A typical algorithm used to perform this task is the log-MAP algorithm [34], which produces two outputs - the *aposteriori* LLR and the extrinsic LLR. The *aposteriori* LLR represents the actual estimate of the actually transmitted bit, $d_{e,n}$. However, this estimation will normally be considered as the output of the Turbo decoder after the number of iterations reaches the maximum number of iterations set to stop the decoding process. This type of LLR can be written as follows

$$\Lambda_{apos}(d_{e,n}^{(1)}) = \log\left(\frac{P(d_{e,n}=+1\left|\Lambda(d_{e,n}^{(p1)}),\Lambda_{apri}(d_{e,n}^{(1)})\right.)}{P(d_{e,n}=-1\left|\Lambda(d_{e,n}^{(p1)}),\Lambda_{apri}(d_{e,n}^{(1)})\right.)}\right), \quad (2.21)$$

which is produced by the first component decoder and

$$\Lambda_{apos}(d_{e,n}^{(2)}) = \log\left(\frac{P(d_{e,n}=+1\left|\Lambda(d_{e,n}^{(p2)}),\Lambda_{apri}(d_{e,n}^{(2)})\right.)}{P(d_{e,n}=-1\left|\Lambda(d_{e,n}^{(p2)}),\Lambda_{apri}(d_{e,n}^{(2)})\right.)}\right), \quad (2.22)$$

produced by the second component decoder.

2.5.2 *Aposteriori* LLR Generation

With both of these *aposteriori* LLRs calculated in the previous subsection, the extrinsic LLRs, which are one of the input components forming the *apriori* LLRs as given in Eq(2.19) and Eq(2.20), can be generated for both of the component decoders as follows

$$\Lambda_{ext}(d_{e,n}^{(1)}) = \Lambda_{apos}(d_{e,n}^{(1)}) - \Lambda_{apri}(d_{e,n}^{(1)}), \quad (2.23)$$

$$\Lambda_{ext}(d_{e,n}^{(2)}) = \Lambda_{apos}(d_{e,n}^{(2)}) - \Lambda_{apri}(d_{e,n}^{(2)}). \quad (2.24)$$

Both of these extrinsic LLRs are exchanged between the two component decoders, as can be observed from Eq(2.19) and Eq(2.20) before the next decoding iteration is run. This exchange process takes place after every iteration until the maximum number of iteration is reached.

After completing this decoding iteration within the Turbo decoder, the extrinsic LLR, $\Lambda_{ext}(d_{e,n}^{(2)})$ is taken as the extrinsic LLR produced for decoding the corresponding parity bit $\nu_{e,n}$, where $\nu_{e,n} = d_{e,n}^{(s)} = d_{e,n}^{(2)} = d_{e,n}^{(1)}$. This extrinsic LLR is required to generate the external extrinsic LLR, which will be explained in the next subsection.

2.5.3 External Extrinsic LLR Generation

The generation of external extrinsic LLRs provides alternative and additional sources of input for improving the decoding process. One of the $L_\varepsilon = 2^{\varepsilon-V-1}$

external extrinsic LLRs is fed into the Turbo decoding process, by updating the *apriori* LLRs as given in Eq(2.19) and Eq(2.20), until the received packet is successfully decoded or all external extrinsic LLRs have been tested or fed to the decoding process.

The external extrinsic LLRs are generated by combining certain extrinsic LLRs. Therefore, it is essential to identify the particular extrinsic LLRs from ε extrinsic LLRs generated by the ε Turbo decoders to produce each of the extrinsic LLRs. To facilitate this combination process, the ε extrinsic LLRs are represented in the form of matrix $\mathbf{\Lambda}_{ext}$ as follows

$$\mathbf{\Lambda}_{ext} = [\Lambda_{ext}(\vec{\nu}_1), \ldots, \Lambda_{ext}(\vec{\nu}_e), \ldots, \Lambda_{ext}(\vec{\nu}_\varepsilon)]\,. \tag{2.25}$$

These extrinsic LLRs can be identified from the extended parity check decoding matrix, $\mathbf{H}_{ext}$ which can be generated from the generator matrix, $\mathbf{G}$, as will be explained as follows.

2.5.4 Extended Parity Check Decoding Matrix, $\mathbf{H}_{ext}$

In this section, the final aim is to produce a $\big((V+Z)\times(2^(\varepsilon-V)-1)\big)$-dimensional extended parity check decoding matrix, $\mathbf{H}_{ext}$, which is essential for generating the external extrinsic LLRs to improve the decoding process. In order to produce this matrix, the following matrices are required:

- $\mathbf{H}_{base}$, a $((V+Z)\times(\varepsilon-V))$-dimensional base parity check matrix.
- $\mathbf{Y}$, a $\big((\varepsilon-V)\times(2^{\varepsilon-V}-(\varepsilon-V)-1)\big)$-dimensional matrix used to generate the columns of matrix $\mathbf{H}_{ext}$.
- $\mathbf{J}^{(check)}$, a $(\varepsilon\times(V+Z))$-dimensional matrix applied to select the nonzero rows of matrix $\mathbf{H}_{ext}$.

- $\mathbf{J}^{(sel)}$, a $(Z \times (\varepsilon - V))$-dimensional matrix, as defined in Eq(2.11) on page 21.
- $\mathbf{P}$, a $(V \times Z)$-dimensional parity generator matrix as defined on page 18.
- $\mathbf{I}_Z, \mathbf{I}_{\varepsilon - V}$, a $(Z \times Z)$ and $((\varepsilon - V) \times (\varepsilon - V))$-dimensional identity matrices.

Using the parity generator matrix, $\mathbf{P}$ and the parity selector matrix, $\mathbf{J}^{(sel)}$, which are also used to form the generator matrix in Eq(2.13), the extended parity check decoding matrix $\mathbf{H}_{ext}$ is expressed as follows

$$\mathbf{H}_{ext} = \mathbf{H}_{base} \otimes \begin{bmatrix} \mathbf{I}_{(\varepsilon - V)} & \mathbf{Y} \end{bmatrix}, \tag{2.26}$$

where the $(Z \times Z)$-dimensional matrix $\mathbf{I}_Z$ is an identity matrix and matrix $\mathbf{H}_{base}$ is the base parity check matrix defined as

$$\mathbf{H}_{base} \triangleq \begin{bmatrix} \mathbf{P} \\ \mathbf{I}_Z \end{bmatrix} \mathbf{J}^{(sel)}, \tag{2.27}$$

The elements in the $(V + z)$-th row in matrix $\mathbf{H}_{ext}$ are all zero if the z-th parity packet is not chosen to be generated from all of the Z possible distinct parity packets, where $z \in \{1, \ldots, Z\}$. This extended parity check decoding matrix, $\mathbf{H}_{ext}$ is produced such that the following equation is satisfied

$$\mathbf{G} \otimes \mathbf{J}^{(check)} \mathbf{H}_{ext} = \mathbf{0}, \tag{2.28}$$

where $\mathbf{J}^{(check)}$ is an $(\varepsilon \times (Z + V))$-dimensional matrix composed to select only the non-zero rows of the extended parity check decoding matrix $\mathbf{H}_{ext}$ defined as follows

$$\mathbf{J}^{(check)} \triangleq \begin{bmatrix} \mathbf{I}_V & \mathbf{0}_{V \times Z} \\ \mathbf{0}_{(\varepsilon - V) \times V} & \left(\mathbf{J}^{(sel)}\right)^T \end{bmatrix}, \tag{2.29}$$

with $\mathbf{I}_V$ as a $(V \times V)$-dimensional identity matrix, $\mathbf{0}_{V\times Z}$ as a $(V \times Z)$-dimensional zero matrix and $\mathbf{0}_{(\varepsilon-V)\times V}$ as an $((\varepsilon - V) \times V)$-dimensional zero matrix.

The equation in Eq(2.28) can be employed for error detection. If all of the ε packets are received and successfully decoded, which should be equal to $\mathbf{C}_{out}$, then the following check equation

$$\begin{aligned} \mathbf{C}_{out} \otimes \mathbf{J}^{(check)}\mathbf{H}_{ext} &= [\mathbf{U} \otimes \mathbf{G}] \otimes \mathbf{J}^{(check)}\mathbf{H}_{ext}, & (2.30) \\ &= \mathbf{0}. & (2.31) \end{aligned}$$

should result in zero. Each column $\vec{y}_f$ of the extended parity check decoding matrix is a bit-wise XOR combination of two or more other columns in this same matrix. In order to simplify the process of generating all of these columns, Eq(2.26) is defined by generating all possible sets of these bit-wise XOR combinations using an identity and a binary matrix $\mathbf{Y}$, whose columns are produced as follows

$$\vec{y}_f = \text{bin}_{\varepsilon-V}(F), \tag{2.32}$$

where $\text{bin}_{\varepsilon-V}(F)$ represents an $((\varepsilon - V) \times 1)$-dimensional binary vector, where the first element is the least significant bit, converted from an equivalent decimal number F that takes any values from 3 to $\left(2^{(\varepsilon-V)} - 1\right)$ excluding the values equivalent to a power of two for $f = 1, \ldots, (2^{(\varepsilon-V)} - (\varepsilon - V) - 1)$. This is true for all values of ε and V such that $\varepsilon - V > 1$. When $\varepsilon - V = 1$, the matrix $\mathbf{Y}$ becomes a null matrix since there will be only one subset of associated packets required as $L_\varepsilon = 2^{\varepsilon-V-1} = 1$ in this case. Consequently, the extended parity check decoding matrix $\mathbf{H}_{ext}$ becomes equal to the base parity check matrix $\mathbf{H}_{base}$, which will be a single-column matrix.

Based on Eq(2.26), it may be observed that $2^{\varepsilon-V} - 1$ columns of the extended parity check decoding matrix were generated. For notational convenience, the

extended parity check decoding matrix can also be written as

$$\mathbf{H}_{ext} = \left[\vec{h}_1, \ldots, \vec{h}_{2^{\varepsilon-V}-1}\right]. \tag{2.33}$$

This is because all combinations of the $\varepsilon - V$ columns in the base parity check matrix are generated in order to produce the extended parity check decoding matrix using the formula given in Eq(2.26). From the equation given in Eq(2.31), it can be seen that each column, $\vec{h}_z$ in the extended parity check decoding matrix can be used to identify a subset of associated packets such that

$$\mathbf{C}_{out} \otimes \mathbf{J}^{(check)}\vec{h}_z = \mathbf{0}, \tag{2.34}$$

for $z = 1, \ldots, 2^{\varepsilon-V} - 1$. Likewise, each column $\vec{h}_z$ can also be used to represent a subset of extrinsic LLRs as each packet received at the receiver end will be decoded by a Turbo decoder, which produces a unique extrinsic LLR. In the next section, the method of choosing and generating the corresponding subsets of extrinsic LLRs from each received source packet for producing the external extrinsic LLRs is described.

2.5.5 Associated External Extrinsic LLR Matrices for Encoded Packets, $\vec{\nu}_e$

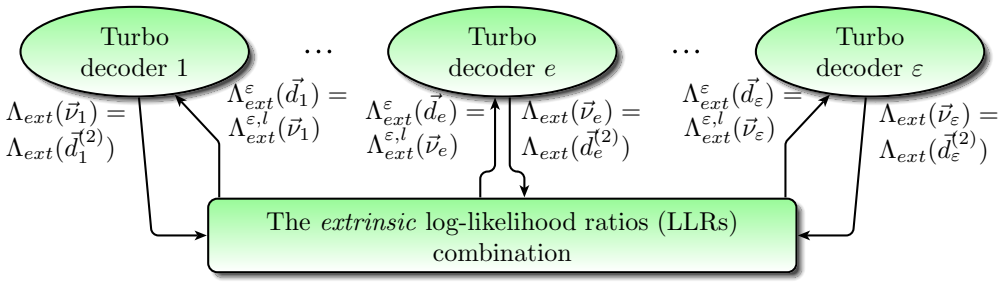

Figure 2.6: The illustration of the external extrinsic LLR generation process

In order to generate the external extrinsic LLRs to improve the decoding pro-

cess, a technique of identifying the subsets of extrinsic LLRs, from which the external extrinsic LLRs are generated, is presented in this subsection. The extended parity check decoding matrix $\mathbf{H}_{ext}$, which is used to identify the subsets of the associated packets, can also be used to identify the subsets of the extrinsic LLRs, which are produced by each Turbo decoder decoding each received packet as shown in Figure 2.6. For the given V source packets and ε outer-encoded packets, all subsets of extrinsic LLRs for packet $\vec{\nu}_e$ can be identified by first introducing a special matrix $\mathbf{H}_e$ called recovery matrix as follows

$$\mathbf{H}_e = \left[\vec{h}_{e,1}, \ldots, \vec{h}_{e,l}, \ldots, \vec{h}_{e,L_\varepsilon}\right], \tag{2.35}$$

where each column $\vec{h}_{e,l}$ represents an identifier of the l-th unique subset of the extrinsic LLRs and the total number of distinct subsets is $L_\varepsilon = 2^{\varepsilon - V - 1}$, for $l = 1, \ldots, L_\varepsilon$.

A column vector $\vec{h}_{e,l}$ is equivalent to the $n_{c,e,l}$-th column, $\vec{h}_{n_{c,e,l}}$, in the extended parity check decoding matrix except that the e-th element in the column vector $\vec{h}_{e,l}$ is equal to zero. The weight or the total number of ones in this vector $\vec{h}_{e,l}$ is represented as $N_{s,e,l}$. Using these identifier vectors, the l-th subset of the extrinsic LLRs for packet $\vec{\nu}_e$ is found in a matrix form as follows

$$\begin{aligned} \mathbf{A}_{ext}^{(e,l)} &= \mathbf{\Lambda}_{ext}\mathbf{J}_{e,l}^{(sel)}, & (2.36)\\ &= \left[\vec{a}_{ext,1}^{(e,l)}, \ldots, \vec{a}_{ext,i}^{(e,l)}, \ldots, \vec{a}_{ext,N_{s,e,l}}^{(e,l)}\right], & (2.37) \end{aligned}$$

where each column $\vec{j}_{e,l,i}$ in the subset selector matrix $\mathbf{J}_{e,l}^{(sel)}$ has zero elements except when the index of the element is equal to the index of the i-th one of column $\vec{h}_{e,l}$. The n-th component of the l-th extrinsic LLR vector $\vec{a}_{ext,i}^{(e,l)}$ for the e-th packet can be represented as $\left[\vec{a}_{ext,i}^{(e,l)}\right]_n$ for $n = 1, \ldots, N_U$, $l = 1, \ldots, L_\varepsilon$. With this LLR component defined, the n-th component of the external extrinsic

LLR vector, $\left[\Lambda_{ext}^{\varepsilon,l}\left(\vec{\nu}_e\right)\right]_n$, can be expressed as [35]

$$\left[\Lambda_{ext}^{\varepsilon,l}\left(\vec{\nu}_e\right)\right]_n = \min_i \left|\left[\vec{a}_{ext,i}^{(e,l)}\right]_n\right| \prod_{i=1}^{N_{s,e,l}} \text{sign}\left(\left[\vec{a}_{ext,i}^{(e,l)}\right]_n\right), \tag{2.38}$$

from which the external extrinsic LLR vectors $\Lambda_{ext}^{\varepsilon,l}\left(\vec{\nu}_e\right)$ are known, where the function sign() finds the sign of the input argument or the value inside the close bracket.

After completing this external extrinsic calculation process, the Turbo decoding operation is reinvoked with an updated *apriori* LLR as given in Eq(2.19) and Eq(2.20) for both component decoders in the Turbo decoder used for the e-th packet, where the external extrinsic LLR in these equations are updated as $\Lambda_{ext}^{\varepsilon}\left(\vec{d}_e\right) = \Lambda_{ext}^{\varepsilon,l}\left(\vec{\nu}_e\right)$. Then, the Turbo decoding process is run for a number of iterations while updating this external extrinsic LLR using a new updated subset of the extrinsic LLRs. At the same time, the integrity of the packet being decoded is regularly checked. If the packet is successfully decoded, the decoding process for the particular packet is terminated. Otherwise, the next $(l+1)$-th external extrinsic LLR vector will be generated using the same process mentioned earlier on.

Therefore, as the number L_ε of unique external extrinsic LLRs increases, the probability of correctly decoding the received packet increases. In addition to this iterative soft decoding operation, a simple hard decoding technique can also be implemented based on L_ε different subsets of associated packets, as can be identified using matrix $\mathbf{H}_e$, if some of the received source packets are still unsuccessfully decoded. In the following section, an account of the proposed hard decoding operation is given.

2.5.6 The CPP Hard Decoding Technique

The proposed CPP hard decoding technique is a simple yet effective tool to recover the failed received source packets by combining some of the successfully detected received packets. The output of each Turbo decoder after the iterative soft decoding process is hard decided and CRC-checked. This vector is then represented as $\tilde{\vec{c}}_e$ for $e = 1, \ldots, \varepsilon$. Then, a diagonal $(\varepsilon \times \varepsilon)$-dimensional matrix $\mathbf{I}_c$ is composed as follows

$$\mathbf{I}_c = \text{diag}(\vec{h}_c), \tag{2.39}$$

having all diagonal elements equal to zero except those which have the same indices as the indices of the successfully decoded received packets. These diagonal elements are represented as an $\varepsilon \times 1$-dimensional vector $\vec{h}_c$, which is termed as the successful packet vector. An example of this vector is for $\varepsilon = 7$ is, $\vec{h}_c = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}^T$, which means that the successfully decoded packets are the first, the sixth and the seventh received packets, as indicated by the indices of ones in this vector.

Matrix $\mathbf{I}_c$ can be used to check the feasibility of successfully hard decoding the failed source packets using one of the columns in the recovery matrix, $\mathbf{H}_e$. If column $\vec{h}_{e,l}$ is feasible to be used for hard decoding a failed e-th packet, then the following relationship

$$\mathbf{I}_c \times \vec{h}_{e,l} = \vec{h}_{e,l}, \tag{2.40}$$

for $l \in \{1, \ldots, L_\varepsilon\}$ must be satisfied.

If a column vector $\vec{h}_{e,l}$ satisfies the condition given in Eq(2.40), then the failed packet, $\tilde{\vec{c}}_e$ can be successfully hard decoded using the following equation

$$\tilde{\vec{c}}_e = \tilde{\mathbf{C}_{out}} \otimes \vec{h}_{e,l}, \tag{2.41}$$

where

$$\tilde{\mathbf{C}}_{out} = \left[\tilde{\vec{c}}_1, \ldots, \tilde{\vec{c}}_\varepsilon\right] \tag{2.42}$$

is the $(N_U \times \varepsilon)$-dimensional matrix representing the hard-decided vectors generated from the soft output vectors of the iterative soft decoding process run previously.

Therefore, it may be concluded that this proposed hard decoding process will improve as the number of columns in matrix $\mathbf{H}_e$ increases, yielding more options of hard decoding the failed packets. Not only does the increase in this number, L_ε, improve the iterative soft decoding process, but it also improves performance of the hard decoding process.

A summary of the proposed integrated decoding technique, which combines the soft and the hard decoding processes, is written in the flow chart given in Figure 2.7.

2.6 Numerical Results

In this section, a series of tests have been carried out to observe the performance of the proposed MCS in reducing the gap value when the number L_ε of associated packets is increased by controlling the adjustment number α, given the initial number of source and parity packets as V_o and $\varepsilon_o - V_o$ respectively. The gap value for a target packet error rate can be determined from a plot of packet error rates vs the input SNRs. The gap value obtained for the proposed MCS is then compared to those of the existing MCSs, which are also tested to measure their gap values. This gap value investigation is done for different data rates and packet sizes to observe performance of these MCSs when data rates and packet sizes are changed.

With the reduced gap values, a further investigation was carried out to find the minimum energy required in order to realize the desired discrete data rate when

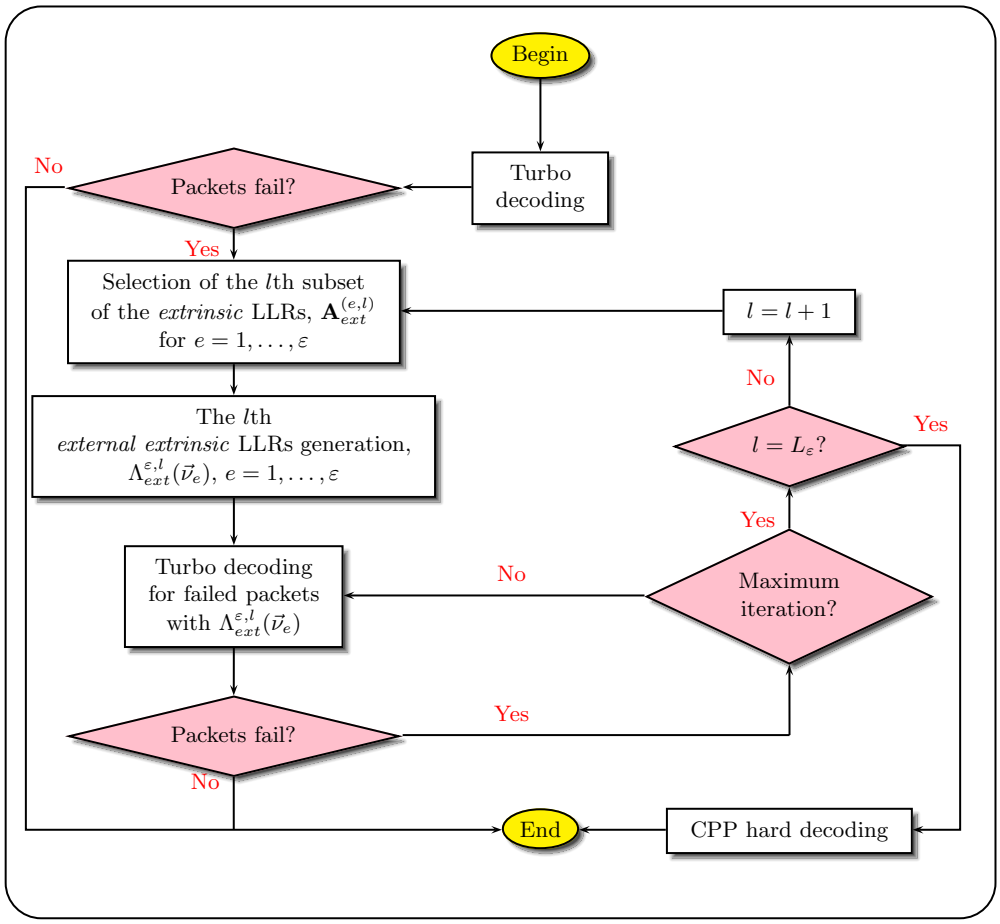

Figure 2.7: The summarized algorithm for the proposed CPP decoding scheme

retransmissions are permissible, such as in wireless ad-hoc networks. This test was repeated for a number of data rates realized with different MCSs including the proposed MCS for comparison study. Reduction in energy consumption is essential for wireless ad-hoc networks to ensure that the lifetimes of the energy-limited nodes are increased since less amounts of energy are consumed or drained. Finally, the number of iterations taken, on average, to decode a packet are measured as a function of the input SNR values in order to observe the computational complexity incurred by the proposed scheme. The result demonstrates the feasibility of incorporating the proposed MCS in practical transmission systems.

To investigate the gap values achieved by the proposed technique as well as the existing MCSs, tests are run by transmitting packets using these MCSs at several input SNR values for a number of data rates, which are defined as

$$\begin{aligned} b_p &= r_{out} r_{in} \log_2 M, \\ &= \frac{V}{\varepsilon} \frac{N_U}{B} \log_2 M. \end{aligned} \tag{2.43}$$

The packet error rates are then measured at each of the input SNRs during the transmission process. If the target packet error rate of 1×10^{-4} is not yet achieved at a given input SNR, the test will be continued at a higher input SNR until the target packet error rate can be achieved.

In order to test the proposed channel coding scheme, short packet sizes of 84 and 256 bits are chosen. There are a number of reasons why these packet sizes are chosen. The first reason is the short time duration taken to produce the packet error rates for the tested schemes when these short packet sizes are used. The second reason, which is the main reason, is that the gap values when long packets are transmitted using capacity-approaching channel coding schemes such as Turbo and LDPC codes are almost equal to zero dB. But, the gap values become larger when short packet sizes are transmitted [36].

Therefore, reducing gap values at short packet sizes are challenging and should be solved. However, several longer packet sizes are also considered such as $84 \times 2, 84 \times 3, 256 \times 2, 256 \times 3, etc$ for the existing schemes under consideration in order to have a fair comparison when these schemes with the proposed scheme.

At a packet size of 84 bits, the proposed MCS is first run at two different data rates, which are $b_p = 1.00$ and $b_p = 1.33$ bits per symbol. The former rate is generated using an outer code rate of $r_{out} = \frac{1}{2}$, an inner code rate of $r_{in} = \frac{1}{2}$ and a 16-QAM modulation scheme, where the outer code rate is realized with $V_o = 3$ and $\varepsilon_o = 6$ and the inner code rate is realized with a standard half-rate Turbo code configuration, which is a punctured rate from the original $\frac{1}{3}$ code rate. The second data rate is realized using an outer code rate of $r_{out} = \frac{2}{3}$, an inner code rate of $r_{in} = \frac{1}{2}$ and a 16-QAM modulation scheme, where the outer code rate is produced from $V_o = 2$ and $\varepsilon_o = 3$ and the inner code rate is produced from a half-rate Turbo code configuration. The generator polynomial used for Turbo codes in both data rates, $b_p = 1.00$ and $b_p = 1.33$ bits per symbol, is $[7, 5]_8$. To observe the effect of increasing the adjustment parameter α in reducing the gap value, a number of packet error rate curves are produced with different α values, as can be seen in Figure 2.8 and 2.9.

It can be seen in Figure 2.8 that two packet error rate curves are plotted at $\alpha = 1$ and $\alpha = \frac{4}{3}$ for the data rate of $b_p = 1.00$ bits per symbol, where the greatest common divisor is $g_o = 3$. As the initial number of parity packets is $\varepsilon_o - V_o = 3$, the number of subsets of the associated packets for $\alpha = 1$ is $L_\varepsilon = 4$ and the number of subsets of the associated packets for $\alpha = \frac{4}{3}$ is $L_\varepsilon = 8$. Therefore, the packet error rate curve corresponding to the case of $\alpha = \frac{4}{3}$ is lower than that of $\alpha = 1$. This is expected since the number L_ε of subsets of the associated packets, from which the alternative additional input information values are generated, is higher. In order to run a comparison between the proposed scheme and the existing schemes, another test is run using a capacity-approaching Multiple Turbo-Fountain (MTF) scheme [25], which is

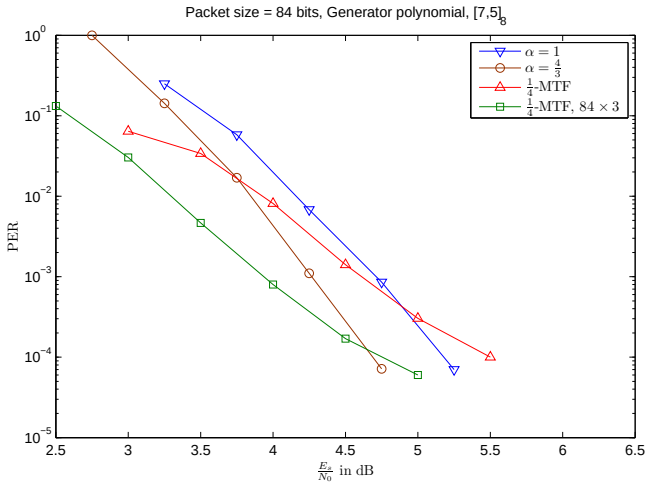

Figure 2.8: The packet error rates recorded when the proposed scheme is implemented at a packet size of 84 bits for $b_p = 1.00$ bits per symbol, $V_o = 3$, $\varepsilon_o = 6$

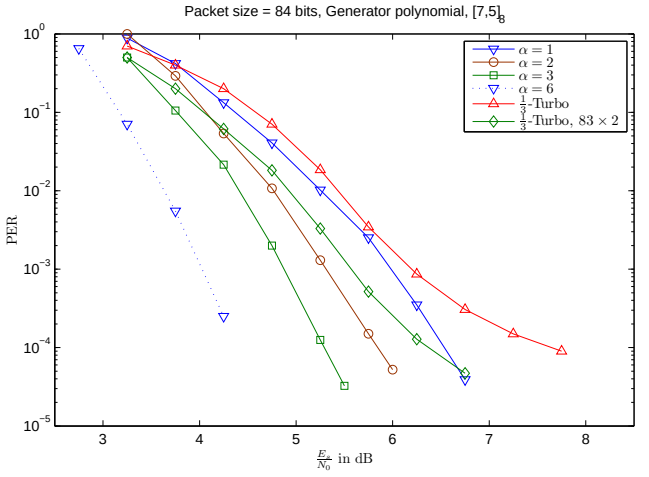

Figure 2.9: The packet error rates recorded when the proposed scheme is implemented at a packet size of 84 bits for $b_p = 1.33$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$

configured at an effective code rate of $\frac{1}{4}$. When operating at a code rate of $\frac{1}{par}$, an MTF scheme is run using three parallel concatenated recursive systematic code (RSC) encoders to produce three streams of parity bits, in addition to the systematic bits. In principle, this MTF scheme is an extended version of a standard Tubo encoder as more RSC encoders are added in parallel to operate this MTF scheme. The same modulation scheme, which is 16-QAM, renders an effective data rate of $b_t = 1.00$ bits per symbol. In order to calculate the gap value for each of the tested data rates, which are realized with different MCS configurations, the SNRs required by these MCSs to achieve a target packet error rate of 1×10^{-4} must be acquired from the packet error rate curves plotted in the figure such that the gap value can be computed as [3]

$$\Gamma = \frac{\text{SNR}_{re}}{2^{b_p} - 1}, \tag{2.44}$$

Using Eq(2.44) the gap value for $b_p = 1.00$ bits per symbol at $\alpha = \frac{4}{3}$ is $\Gamma = 4.7$ dB, where the required SNR, which is denoted by SNR_{re}, is found to be 4.7 dB from Figure 2.8 on page 41. In the same way, the gap value for the data rate of $b_p = 1.00$ bits per second at $\alpha = 1$ is found as $\Gamma = 5.25$ dB and the gap value for the data rate of $b_t = 1.00$ bits per symbol realized by the MTF scheme is calculated as $\Gamma = 5.50$ dB. Clearly, based on these measured gap values, the smallest gap value is achieved by the proposed scheme at $b_p = 1.00$ bits with $\alpha = \frac{4}{3}$ and $\varepsilon_o - V_o = 3$.

In Figure 2.9, the next data rate of $b_p = 1.33$ bits per symbol is tested by transmitting packets at this rate before the packet error rates are measured. At the same data rate, the adjustment parameter α is varied from 1 to 3, where the initial number of parity packets is $\varepsilon_o - V_o = 1$ as given before. Having the target packet error rate of 1×10^{-4}, the gap values achieved at $\alpha = 1, 2, 3$ are $\Gamma = 4.7, 4.1, 3.6$ dB respectively. Another test is run to compare these gap values with the gap value achieved when a similar data rate of $b_t = 1.33$ bits per

symbol is realized using a $\frac{1}{3}$-Turbo and the same modulation scheme, which is 16-QAM. The packet error rate corresponding to this scheme is denoted as a red line, which is the rightmost curve in Figure 2.9. From this curve, the gap value can be measured using Eq(2.44), which renders a gap value of approximately $\Gamma = 5.7$ dB. Once again, the effectiveness of reducing the gap value using the proposed scheme in this chapter is demonstrated where the gap values achieved using the proposed scheme for $b_p = 1.33$ bits per second at $\alpha = 1, 2, 3$ are smaller than that of the $\frac{1}{3}$-Turbo and 16-QAM. The packet error rates obtained from running the proposed MCS, which was configured at $b_p = 1.33$ bits per symbol, at $\alpha = 6$ are also plotted in Figure 2.9. It is clearly shown in this figure that the packet error rates are significantly reduced at relatively low SNRs as compared to the rest. Consequently, the gap value is significantly reduced to approximately $\Gamma = 3.7$ dB for the same target packet error rate of 1×10^{-4}.

In addition to data rates of $b_p = 1.00$ and $b_p = 1.33$ bits per symbol tested at 84-bit packets, two more data rates are also tested which are $b_p = 1.50$ and $b_p = 2.25$ bits per symbol. The former data rate, $b_p = 1.50$ bits per symbol, is realized using $V_o = 3$, $\varepsilon_o = 4$, $\alpha = 1, 2, 3$ and an inner Turbo code rate of $\frac{1}{2}$. The later data rate, $b_p = 2.25$ bits per symbol, is realized using $V_o = 3$, $\varepsilon_o = 4$, $\alpha = 1, 2, 3$ and an inner Turbo code rate of $\frac{3}{4}$. Both of these data rates use 16-QAM scheme as the modulation scheme to map the coded bits to the QAM symbols for transmission.

The packet error rates, measured while running the simulation for these two data rates $b_p = 1.50, 2.25$, are plotted in Figure 2.10 and 2.11 respectively. The packet error rate curve for $b_p = 1.50$, which is realized by the proposed CPP scheme, is also compared to that of a $\frac{1}{3}$-Turbo 16-QAM scheme which produces a smaller data rate of $b_t = 1.33$ bits per symbol. The packet error rate curve for $b_p = 2.25$ is compared to that of a $\frac{1}{2}$-Turbo 16-QAM scheme which produces a smaller data rate of $b_t = 2.00$ bits per symbol. It can be observed from both Figure 2.10 and 2.11 that the proposed scheme reduces the gap values by more

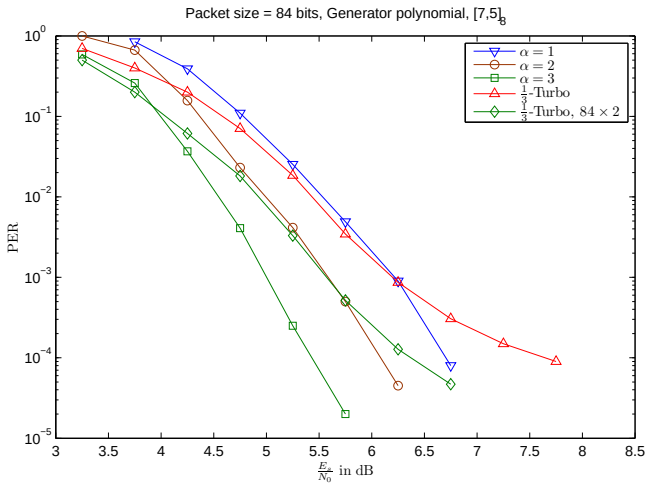

Figure 2.10: The packet error rates recorded when the proposed scheme is implemented at a packet size of 84 bits for $b_p = 1.50$ bits per symbol, $V_o = 3$, $\varepsilon_o = 4$

than 2dB as compared to those of the Turbo 16-QAM schemes. The gap value for $b_p = 1.50$ at $\alpha = 3$ is 2.88 dB and the gap value for $b_t = 1.33$ is about 5.10 dB. The gap value for $b_p = 2.25$ at $\alpha = 3$ is 2.85 dB and the gap value for the case of $b_t = 2.25$ is also larger than 5 dB.

To extend the test on the proposed MCS, the packet size is increased to 256 bits. One of the existing schemes is known as variable-rate Turbo-BICM scheme [14] is also tested before it is compared to the proposed scheme. General information about this MCS scheme is given in Section 2.2. Apart from this technique, another form of concatenated coding methods is developed and tested. This technique combines BCH codes as the outer codes, and the Turbo codes as the inner codes.

Figure 2.12 and 2.13 show the packet error rates calculated when the proposed CPP scheme is tested at $b_p = 1.33$ and $b_p = 1.50$ bits per symbol at $\alpha = 1, 3$ where $V_o = 2, \varepsilon_o = 3$ and $V_o = 3, \varepsilon_o = 4$ respectively, where 16-QAM scheme is used for both of these data rates. In addition to these data rates, a $\frac{2}{3}$-Turbo 16-QAM scheme is also tested at a data rate of $b_t = 1.33$ bits per symbol to calculate the corresponding packet error rates which are compared against those

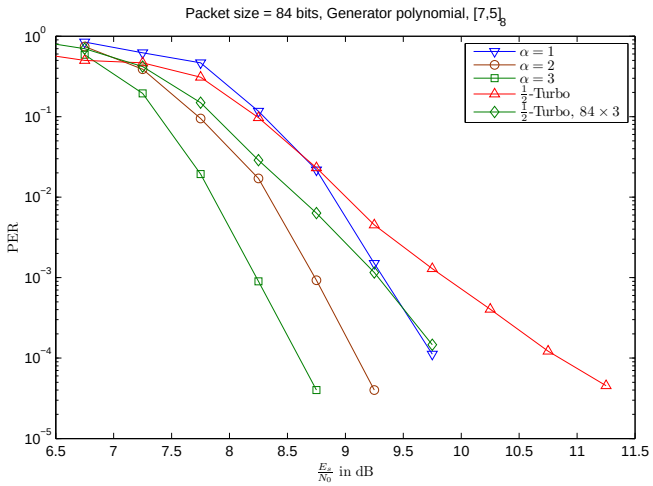

Figure 2.11: The packet error rates recorded when the proposed scheme is implemented at a packet size of 84 bits for $b_p = 2.25$ bits per symbol, $V_o = 3$, $\varepsilon_o = 4$

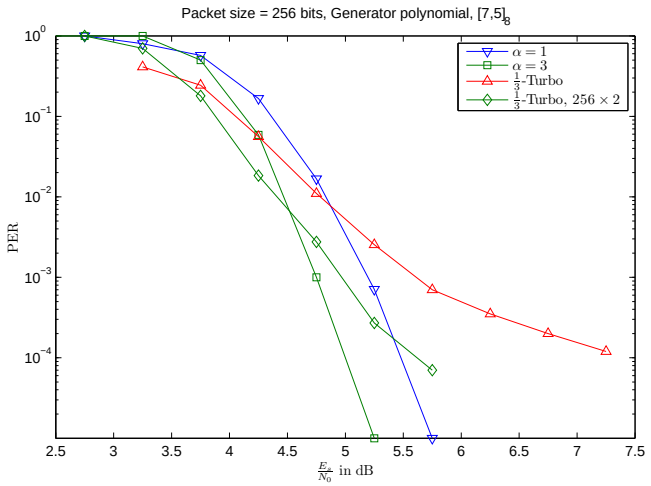

Figure 2.12: The packet error rates recorded when the proposed scheme is implemented at a packet size of 256 bits for $b_p = 1.33$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$

of the proposed CPP scheme.

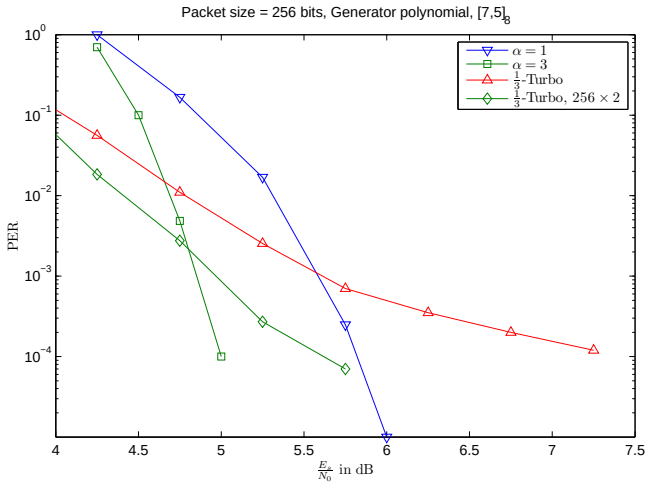

Figure 2.13: The packet error rates recorded when the proposed scheme is implemented at a packet size of 256 bits for $b_p = 1.5$ bits per symbol, $V_o = 3$, $\varepsilon_o = 4$

It can be seen in Figure 2.12, that the proposed CPP scheme configured at $b_p = 1.33$ bits per symbol reduces the gap value by more than 2 dB from that of the $\frac{2}{3}$-Turbo 16-QAM scheme, which realizes $b_t = 1.33$ bits per symbol. The gap value for $b_p = 1.33$ bits per symbol is 3.2 dB and the gap value for $b_t = 1.33$ bits per symbol is more than 5 dB. The same observation is made for the case of $b_p = 1.50$ bits per symbol, where the gap value achieved by the proposed CPP scheme at $\alpha = 3$ is about 2 dB less than that of the $\frac{2}{3}$-Turbo 16-QAM scheme, which realizes $b_t = 1.33$ bits per symbol.

Figure 2.14 shows the packet error rates obtained by simulating the variable-rate Turbo-BICM scheme at $b_t = 1.6$ bits per symbol, where the Turbo code rate is adjusted to $\frac{2}{5}$ and the same 16-QAM scheme is also used for modulation. For comparison, the proposed CPP scheme is set to run at the same data rate of $b_p = 1.6$ bits per symbol using an outer code rate of $r_{out} = \frac{2}{3}$, an inner code rate of $\frac{3}{5}$ and a modulation scheme of 16-QAM. With the adjustment parameter $\alpha = 1, 3$, the packet error rates measured from the test are also plotted in the same figure. From the gap values measured for these data rates, the proposed

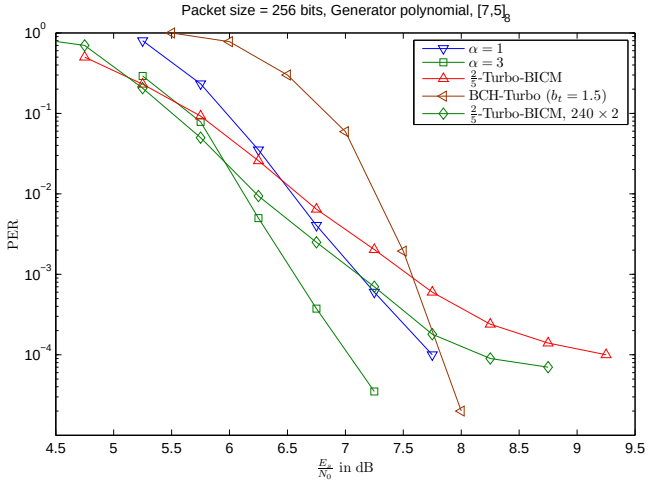

Figure 2.14: The packet error rates recorded when the proposed scheme is implemented at a packet size of 256 bits for $b_p = 1.6$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$

CPP scheme is found to reduce the gap value by approximately 2 dB from the gap value achieved by the variable-rate Turbo-BICM scheme when the target packet error rate is set at 1×10^{-4} and the adjustment parameter is $\alpha = 3$.

Similar performance of the proposed CPP scheme is also observed in Figure 2.15, where the BCH-Turbo coding scheme combined with the 16-QAM as mentioned previously is compared to the proposed CPP scheme at $b_t = 2.25$ bits per symbol. The proposed CPP scheme has achieved a gap value of approximately 4 dB less than that of the BCH-Turbo scheme when $\alpha = 3$. Furthermore, a $\frac{1}{2}$-Turbo 16-QAM scheme is run at $b_t = 2.00$ bits per symbol, which is lower than the data rate realized by the proposed CPP scheme in this case. However, the gap value achieved by the proposed CPP scheme at $\alpha = 3$ is still lower than that of the $\frac{1}{2}$-Turbo 16-QAM scheme. The difference between these two gap values is approximately 1.5 dB.

Based on the results obtained so far at various data rates and a number of small packet sizes, it can be observed that as the adjustment parameter α for the number of parity packets $\varepsilon_o - V_o$ is increased, the gap value reduces. The main reason behind this improvement is the increase in the number L_ε of

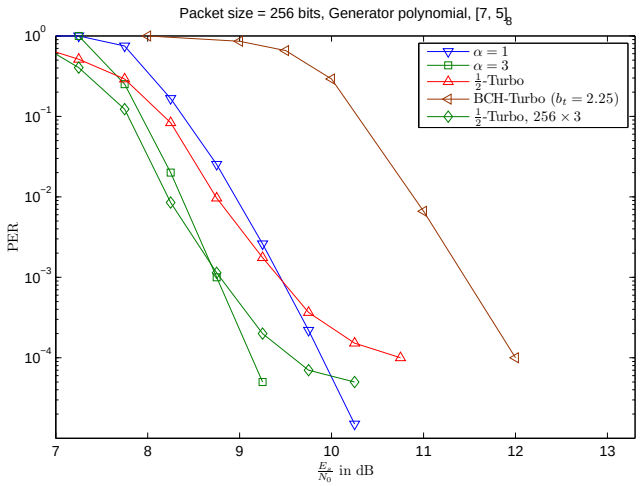

Figure 2.15: The packet error rates recorded when the proposed scheme is implemented at a packet size of 256 bits for $b_p = 2.25$ bits per symbol, $V_o = 3$, $\varepsilon_o = 4$.

subsets of the associated packets, from which the alternative additional input information values can be generated and fed into the decoding process as to improve detection performance, hence reducing the gap value. In most of the data rates tested, the gap value can be reduced by approximately 2 dB from the gap value achieved using the existing MCS scheme when the adjustment parameter α is increased for a given initial number of parity packets, $\varepsilon_o - V_o$.

When the effective block size of the source packets is increased (increased packet sizes such as $84 \times 2, 84 \times 3, 256 \times 2$, as seen in the figures) for the other capacity-approaching channel coding schemes such as Turbo, Turbo-BICM and MTF schemes, it can be observed that the gap values of these codes reduce. This gap value reduction is expected as a longer interleaver length is applied on the source packets. However, the reduced gap values of these codes are only within 0 to 0.5 dB, approximately, from those of the CPP codes.

In practice, the real-time adjustment of α is implemented by checking the received SNR, the information block size and the target packet error rate. The received SNR is obtained from a signal sent by the receiver end, which previously receives a test signal from the transmitter. If a larger information block

size is required to be transmitted, the adjustment parameter α is increased accordingly whilst ensuring that the received SNR, which is known from the receiver, is higher than the minimum required SNR for the target packet error rate.

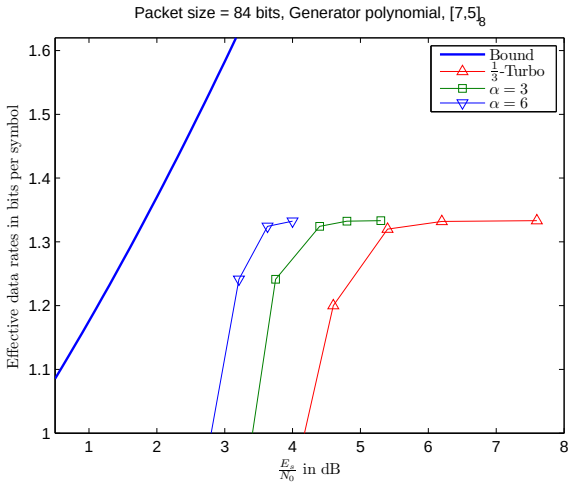

Figure 2.16: The effective data rates after retransmissions of 84-bit packets

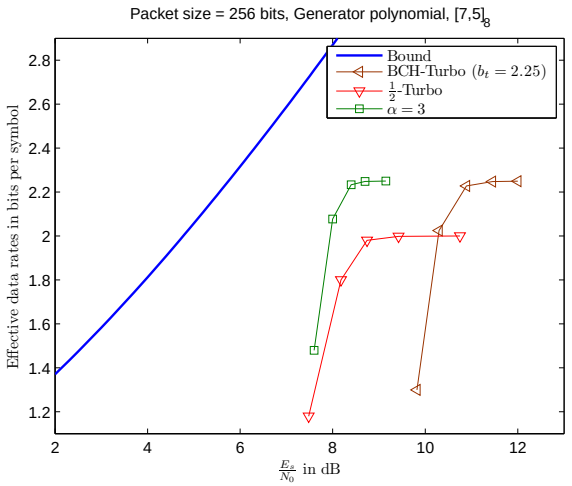

Figure 2.17: The effective data rates after retransmissions of 256-bit packets

Reduction in the gap value is essential to ensure that the SNR value required to realize a desired data rate is reduced whilst maintaining the target packet error rate low. Minimizing the required SNR value yields reduction in the amount of energy consumed in order to transmit packets at the desired data rate. When

energy is limited, especially in the case of wireless ad-hoc networks which use energy-limited nodes, energy consumption must be reduced to ensure that the node lifetime is prolonged, hence improving the network lifetime and availability. To demonstrate the effectiveness of employing the proposed CPP scheme in reducing the required SNR, hence the energy consumed, whilst ensuring that the desired data rate is achieved as high as possible, the following test is carried out.

Assuming that N_{re} retransmissions are allowed, then the expected number of packets which has been retransmitted if the packet error rate of PER($\frac{E_s}{N_0}$) at the current SNR value $\frac{E_s}{N_0}$ is given as

$$\text{Number of retransmitted packets } = V \sum_{i=1}^{N_{re}} \left(\text{PER}(\frac{E_s}{N_0}) \right)^i, \tag{2.45}$$

which leads to an effective data rate for the proposed scheme after N_{re} retransmissions as

$$b_{re} = \frac{V}{\varepsilon + V \sum_{i=1}^{N_{re}} \left(\text{PER}(\frac{E_s}{N_0}) \right)^i} \frac{N_U}{B} \log_2 M, \tag{2.46}$$

At $b_p = b_t = 1.33$ bits per symbol, the data rate as defined in Eq(2.46) is calculated using the packet error rate values PER($\frac{E_s}{N_0}$) obtained from Figure 2.9 for 84 bits per packet, where the number of permissible retransmissions is $N_{re} = 10$. The range of selected packet error rates is between 0.4 and 1×10^{-4} and the adjustment parameter values considered are $\alpha = 3$ and $\alpha = 6$. Therefore there are two new effective data rates, each corresponds to one of the α values. In addition to these two data rates, another data rate realized using a $\frac{1}{3}$-Turbo scheme implemented with 16-QAM is also calculated using an equation which is slightly modified from Eq(2.46) since $V = \varepsilon = 1$ in the Turbo case. Hence, this modified formula is written as

$$b_{re}^{(2)} = \frac{N_U}{B + B\sum_{i=1}^{N_{re}} \left(\mathrm{PER}(\frac{E_s}{N_0})\right)^i} \log_2 M, \tag{2.47}$$

All these new effective data rates are plotted against the required input SNR $\frac{E_s}{N_0}$ as can be seen in Figure 2.16 on page 49. From this figure, comparisons can be made by finding the minimum input SNR needed to realize the highest possible data rate. This minimum SNR can be identified by finding the minimum ratio between the input SNR and the effective data rate in Figure 2.16 on page 49, for a given minimum threshold of the effective data rate. The minimum threshold of the effective data rate permissible is $0.95b_p$, where b_p is the maximum achievable data rate when the packet error rate is zero. By looking at these figures, the minimum SNR can be quickly identified as follows:

As the highest possible data rate is 1.33 bits per symbol: the minimum input SNR is found from this figure in the leftmost curve, which corresponds to the case of $\alpha = 6$, at input SNR of $\frac{E_s}{N_0} = 3.2$ dB. This input SNR corresponds to the packet error rate of 1×10^{-2}, when no retransmissions are offered. Whereas, the minimum input SNR required by the $\frac{1}{3}$-Turbo scheme combined with 16-QAM is about 5.3 dB, which is about 2 dB larger than that of the proposed scheme at $\alpha = 6$. When the proposed scheme is set at $\alpha = 3$, the minimum SNR required is about 4.2 dB, which is still smaller than the required minimum input SNR by the Turbo scheme. Having minimized input SNRs is essential in reducing the energy consumption in wireless ad-hoc networks.

In the same manner, another plot of effective data rates versus input SNRs is produced in Figure 2.17 on page 49. The effective data rates are again calculated using Eq(2.46), where the packet error rates are taken from Figure 2.15, in which the packet error rates for the data rate of $b_p = 2.25$ bits per symbol are generated using the proposed scheme along with the data rate of $b_t = 2.00$ bits per symbol realized by the standard $\frac{1}{2}$-Turbo scheme with 16-QAM. In addition to these curves, another packet error rate curve is also considered,

which is produced from running BCH codes as the outer codes and Turbo codes as the inner codes, along with 16-QAM to yield a data rate of $b_t = 2.25$ bits per symbol. In this figure, the same range of packet error rates PER($\frac{E_s}{N_0}$) are considered between 0.4 to 1×10^{-4}. Likewise, another plot of effective data rates after retransmissions are given in Figure 2.17. From this figure, it can be observed that the minimum input SNR $\frac{E_s}{N_0}$ used to achieve the highest data rate of 2.25 bits per symbol is 8.5 dB, which corresponds to the proposed scheme at an adjustment parameter of $\alpha = 3$. This minimum input SNR corresponds to the packet error rate of 1×10^{-2} when no retransmissions are considered. More interestingly, this minimum input SNR is still smaller than the minimum input SNR required by the $\frac{1}{2}$-Turbo scheme with 16-QAM, which can only realize a maximum data rate of $b_t = 2.00$ bits per symbol in this case.

With these results, it can be observed that the proposed MCS can be used to minimize the required input SNR, hence the energy consumption. This is all whilst ensuring the highest possible data rate is realizable.

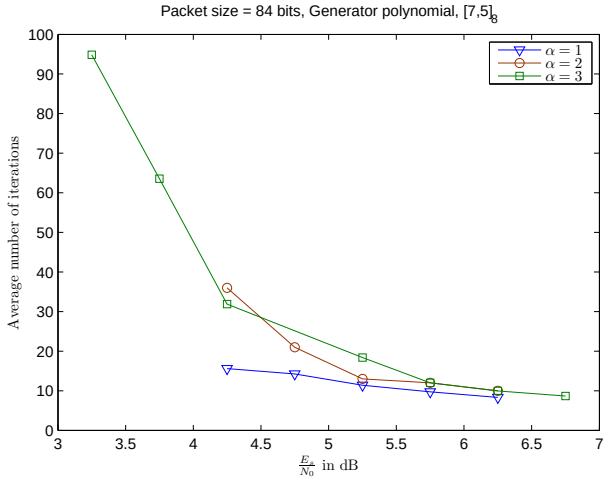

Figure 2.18: The average number of internal Turbo iterations taken to decode a received packet for $b_p = 1.33$ bits per symbol.

A further important question remains as regards the proposed MCS is the computational complexity incurred to realize these data rates. For this reason, the average number of iterations required by the Turbo decoder implemented in

the proposed MCS is measured against the input SNR. The number of iterations investigated is the average of the total number of iterations required by the two component decoders of a standard Turbo decoder when operating at a given input SNR, $\frac{E_s}{N_0}$. This test is run for different adjustment parameter values, $\alpha = 1, 2, 3$ at $b_p = 1.33$ bits per symbol at a packet size of 84 bits. The results obtained are plotted in Figure 2.18 on page 52. As can be observed from this figure, the average number of iterations required by a Turbo decoder to decode a packet decreases significantly as the input SNR is increased. This occurs due to the early stopping criteria introduced in the iterative decoding process, as given the flow chart shown in Figure 2.7 on page 38. When the input SNR is at about 5 dB, where the packet error rate is observed to be less that the target packet error rate of 1×10^{-4}, the average number of iterations falls between 10 and 16. This average number of iteration agrees with the reduced number of iterations investigated in [37], which showed that the average number of iterations is between 6 to 19.

2.7 Conclusion

This chapter gives an account of a novel channel coding scheme which reduces the gap value whilst keeping the packet error rate low by providing alternative additional input information values for the decoding process. These alternative additional input information values are generated by adjusting the number of source and parity packets during the outer encoding process. Reduction in the gap value is demonstrated to significantly reduce the required SNR, hence the energy consumption, by keeping the packet error rate low. Consequently, the number of retransmissions reduces at relatively low SNRs as compared to other existing techniques.

CHAPTER 3

Two-group Resource Allocation Implementation with Coded Parity Packet Approach

3.1 Introduction

Improvement in the transmission data rates achievable in wireless WCDMA networks is highly sought after in this rapidly advancing wired world. This is especially so, given the introduction of new technologies including multi-code transmission and adaptive modulation coding (AMC) [38]. Multi-code transmission permits parallel multi-code channels to be employed to transmit the data, the rate of which can be adaptively configured using AMC for each of the multi-code channels. Subject to the available total energy and the channel conditions, each of these channels is allocated a fraction of the total energy to load one of the realizable discrete data rates via AMC.

In the current standard of the Third Generation Partnership Project (3GPP), which is specified in Release 10 [1], more discrete data rates with smaller bit granularities are offered, where a bit granularity is defined as the difference

between two adjacent data rates. Consequently, the total data rate achieved by this current technique approaches that of the two-group resource allocation scheme as well as the optimal method. In practice, however, this technique might be computationally prohibitive as the iterative calculation of the energy to be allocated per channel requires extensive computational loads as more data rates need to be tested before finding the highest possible data rate for the given total energy. This computational load is prohibitive especially when matrix inversions are required to perform operations such as minimum mean square error (MMSE) equalization, which is typically used to minimize the required energy for realizing a target SNR.

Alternatively, this chapter provides a different transmission method to the well-established multi-code HSDPA systems [1] for improving the total data rate by using a two-group resource allocation scheme. This proposed loading scheme has been shown to produce an improvement of about 5 dB against the existing equal rate and equal energy loading scheme. To further reduce the computational complexity of the two-group resource allocation scheme whilst maitaining and improving the total data rate, the two-group resource allocation scheme was proposed as to be implemented with a smaller number of realizable data rates. From the results produced, this proposed scheme is demonstrated to be effective in maintaining high total data rates at relatively lower computational complexity. Further enhancement measures including the gap value reduction and the channel ordering methods were also incorporated and investigated as to increase the total data rate and reduce the required computational complexity. With a reduced computational load, the application of the two-group approach can be extended to the latest technologies including the femtocells, which have a limited signal processing capability but a higher received SNR due to the proximity of the femtocell node to the users.

3.2 Related Work

Resource allocation schemes in multi-code transmission systems primarily aim to efficiently utilize the limited radio resources to enhance the total transmission data rate. The available total energy can be equally allocated to each channel [39, 40] to realize an equal data rate per channel, but this approach causes a significant amount of SNR wasted. An improved scheme which is able to remove ISI effects was proposed in [1] but the amount of wasted SNR was still relatively high due to the equal energy allocation used to realize an equal data rate in each channel.

As one of the main radio resources, the total energy must be allocated to each active code channel such that the minimum SNR needed to realize the desired data rate in the channel is achieved. Although the energy allocated per channel can be increased to achieve the minimum or the target SNR, the interference components including ISI and MAI which are experienced by the channels, tend to limit the corresponding SNR. These interference components are caused by the cross-correlations between the receiver signature sequences, which are the convolution between the originally orthonormal signature sequences and the impulse response of the frequency selective channels. Furthermore, the channel gain might also decrease due to the multiple propagation paths that exist over the frequency selective channels.

One common practice applied to improve resource allocation techniques at the transmitter end is to consider and incorporate equalization techniques implemented at the receiver end. These equalizations are developed to reduce the cross-correlations aforementioned in order to lower the effects of the interference components whilst ensuring the effective gain for each channel to be equal to unity, which is the case when orthonormal signature sequences are employed. Chip-level equalization [41–43], which was implemented in a Rake

receiver structure, was demonstrated as to be effective in removing the interference components. To remove interference whilst keeping the noise level low, symbol-level equalization [44] may be integrated with the Rake receiver, which applies the matched filtering approach. The despreading filter on the receiver end is matched to the receiver signature sequence, which is produced from the convolution between the original signature sequence and the channel convolution matrix [45–47], generated from the channel impulse response vector. Due to frequency selective channels which cause multiple propagation paths, the orthonormality of the receiver signature sequences tends to disappear, which consequently increases the cross-correlations between these receiver signature sequences and hence increases MAI and ISI. Alternatively, a linear minimum mean squared error (MMSE) equalization technique can be incorporated [48] to minimize MAI and ISI, yielding a minimized energy as to achieve a target SNR in each channel. The iterative process of adjusting and minimizing the energy that should be allocated to each channel in order to achieve a target SNR for the desired data rate is considered as a margin adaptive optimization problem. For the given total of constrained energy, a margin adaptive optimization technique reduces the required amount of energy to realize a target data rate, which has a target minimum SNR. Consequently, there is a relatively significant amount of residual energy, which is unused to transmit any information, due to the minimization of the allocated energy. This residual energy should be used to further increase the data rate. Therefore the residual energy is not wasted but used to transmit the data at a higher data rate.

This residual energy can be utilized to enhance the transmission data rate using rate adaptive optimization, which adjusts and tests different combinations of data rates to be allocated over the channels such that the total transmission data rates are maximized for a given total energy. In literature, optimal rate adaptive optimization methods include water-filling algorithms [49–53], which initially find the optimal continuous data rates for each channel before convert-

ing them to data rates. These approaches however require relatively high computational complexity and result in an SNR waste problem as the continuous data rates tend to be rounded to the nearest lower data rates. Levin-Campello (LC)-algorithms [54–56] provide an alternative solution by finding the optimal data rates to be allocated to the channels. Another optimal approach, to give an example, is the branch and bound algorithm [57], which also finds the optimal total data rates for the case of ISI-free channels. These optimal methods, however, incur prohibitively high computational complexity.

Apart from the optimal methods, several sub-optimal rate adaptive loading methods are also proposed in literature such as simulated annealing [58], integer programming [59] and dynamic programming [60–62]. These approaches require considerably less computational complexity but tend to degrade in performance as the number of employed channels increases. Another widely applied loading strategy is the singular value decomposition(SVD)-based scheme, such as in [63–65], to name a few. With this approach, the channels are truncated such that the remaining channels used for transmission are orthogonal to each other. Although interference components including MAI and ISI are removed, the realized total data rate may decrease since the number of remaining channels is relatively small especially when the number of delayed multiple propagation paths is comparable to the spreading factor of the signature sequences.

In this chapter, an alternative resource allocation scheme which combines both margin and rate adaptive optimization methods is developed to improve the total data rate by utilizing the total energy including the residual energy and removing the wasted SNR. This so-called two-group resource allocation scheme, which is modified from the technique presented in [66–68], demonstrates better performance than some of the current existing margin adaptive loading schemes, such as the methods presented in [59,69,70], at a relatively lower computational complexity. When data rates with relatively large bit granularities, which are the differences between adjacent data rates, are used for the two-group resource

allocation scheme, the achievable total data rates observed were found to be at least as good as those of the existing margin adaptive equal rate loading schemes implemented using a larger number of data rates at smaller bit granularities. With lower computational loads, this proposed two-group resource allocation scheme would be suitable for application in femtocells [71, 72], which has a limited processing capability.

The rest of this chapter is organized as follows. The system models on which the proposed two-group resource allocation scheme were designed are described in Section 3.3. The design of the two-group resource allocation scheme and related existing resource allocation schemes were first described in Section 3.4, presenting the concept of margin adaptive optimization and related loading techniques in literature as well as the modifications made to develop the two-group loading method. Section 3.5 continues with the description of rate adaptive optimization methods, which is also an integral part of the two-group resource allocation scheme. This proposed two-group loading scheme is further enhanced using a number of performance enhancement measures, as presented in Section 3.6. In order to test the two-group loading method, a brief description of block transmission simulation is given in Section 3.7, before the acquired results from simulations are presented in Section 3.8. This chapter is concluded in Section 3.9.

3.3 System Model

Transmission over the downlinks of multi-code HSDPA systems can be modeled in continuous and discrete time domains, as given in Figure 3.2 and 3.3, which are briefly described as follows.

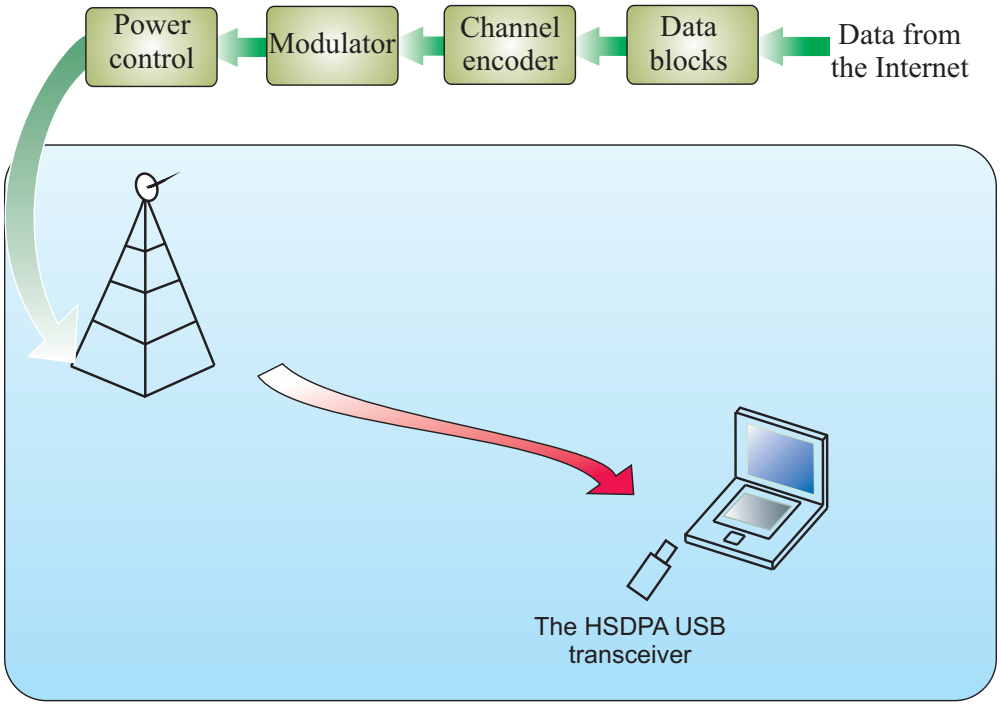

Figure 3.1: An example of multi-code transmission networks:HSDPA

3.3.1 The Chip-matched Filtering Model

Figure 3.2 shows a chip-matched filtering model for a single code channel in multi-code transmission systems. At the ρ-th symbol period, the real and imaginary components of a QAM symbol $x_k(\rho)$ are separately spread using an $(N \times 1)$-dimensional $\vec{s}_k = [s_{k,1}, s_{k,2}, \ldots, s_{k,N}]^T$ (Point 1). Convolution using a chip pulse shaping function $\varphi(t)$ (Point 2) is then run before modulation is implemented at Point 3 to yield a modulated waveform $\zeta(t) = \sqrt{\frac{2}{T_c}}\mathrm{Re}\left(\sum_{k=1}^{K}\sqrt{E_k}x_k(\rho)\sum_{n=1}^{N}s_{k,n}\varphi(t-nT_c)e^{j2\pi f_c t}\right)$. Over the transmission channel (Point 4), convolution with the impulse response function $h(t)$ and addition with the white Gaussian noise $n(t)$ take place. The received signal $r(t)$ is then down-converted and divided into two branches (Point 5). The chip-matched filtering operation at T_c seconds (Point 6) produces $r_n(\rho)$, which is despread with a complex-valued vector $\vec{w}_k$ (Point 7) to yield a despread symbol $\varkappa_k(\rho) = \vec{w}_k^H\vec{r}(\rho)$ at every $T_s = NT_c$ seconds, where

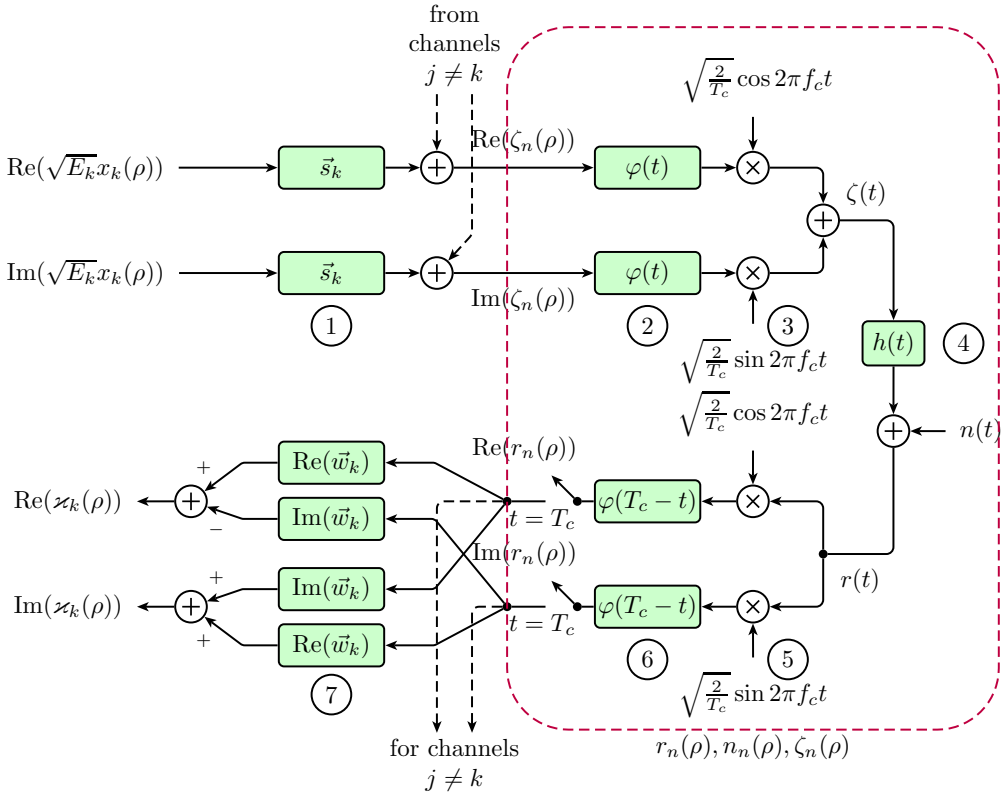

Figure 3.2: The continuous and discrete time domain model for transmission over a single channel

$\vec{r}(\rho) = [r_1(\rho), \ldots, r_{N+L-1}(\rho)]^T$ is the discrete-time received signal vector. In order to capture the ISI components from the previous and the next symbols, some elements in the received signal $\mathbf{R}$ are recorded in consecutive symbol periods, where $[r_N(\rho), \ldots, r_{N+L-1}(\rho)] = [r_1(\rho+1), \ldots, r_L(\rho+1)]$ for every symbol period ρ, where N is the spreading factor and L is the number of propagation paths. Likewise, the discrete-time noise vector $\vec{n}(\rho) = [n_1(\rho), \ldots, n_{N+L-1}(\rho)]^T$ is also represented in the same manner aforementioned such that $[n_N(\rho), \ldots, n_{N+L-1}(\rho)] = [n_1(\rho+1), \ldots, n_L(\rho+1)]$

3.3.2 The Multi-code Transmission Model

In the previous chapter, the proposed coding scheme was implemented in a single-link transmission model. Most of the current high-speed transmission

systems, however, are implemented using multi-channel transmission models which provide parallel channels to substantially enhance the total transmission data rate. In this chapter, a multi-code transmission model is chosen as the platform on which the proposed two-group allocation scheme is implemented. Multi-code transmission uses parallel code channels, each of which is allocated with an orthonormal signature sequence, in order to increase the total transmission data rate. As in the previous chapter, the model is designed as a discrete time model, the mathematical formulation of which is easier than the continuous time model.

Consider K code channels available for carrying data at b_p bits per symbol, where b_p is configured from the employed MCS and takes one of the P configurable data rates for $p = 1, \ldots, P$. Frequency selective channels are assumed with the corresponding signature sequence matrix $\mathbf{S} = [\vec{s}_1, \ldots, \vec{s}_k, \ldots, \vec{s}_K]$, the spreading factor N and the channel impulse response vector $\vec{h} = [h_0, \ldots, h_{L-1}]^T$, where $L-1$ is the number of delayed propagation paths which cause interference including the MAI and ISI on the received signal, which is also corrupted by the additive white Gaussian noise with a variance of σ^2.

For notational convenience, a channel convolution matrix [45–47] defined as

$$\mathbf{H} = \begin{bmatrix} h_0 & 0 & & 0 \\ \vdots & h_0 & & \vdots \\ h_{L-1} & \vdots & \ddots & 0 \\ 0 & h_{L-1} & & h_0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & & h_{L-1} \end{bmatrix}, \tag{3.1}$$

which is generated from the channel impulse response vector $\vec{h}$, is used to facilitate the convolution operation between the channel impulse response and the

signature sequences. These multiple delayed propagation paths render the loss of orthonormality of the signature sequences $\vec{s}_k$, which in turn causes the cross-correlations or the inner products between any two signature sequences from different channels to be nonzeros. Consequently, the total of the cross-correlations experienced by any single channel yields a relatively significant amount of interference, rendering a reduction in the received SNR, which should be kept larger than the minimum required SNR to realize the target transmission data rate. Furthermore, the inner products between two similar signature sequences are unlikely to be equal to unity, which is desirable to ensure the received SNR is close to the target SNR set during the resource allocation process on the transmitter end. If ISI is also considered, the interference effects experienced in each channel tend to deteriorate due to the loss of the orthonormality of the signature sequences at the receiver end.

Assuming that the ISI components are caused by the previous and the next symbols, the effective receiver signature sequences can be expressed as an $((N+L-1)\times 3K)$-dimensional matrix [73] as follows

$$\begin{aligned}\mathbf{Q}_e &= \left[\begin{array}{cccc}\vec{q}_1 & \vec{q}_2 & \dots & \vec{q}_{3K}\end{array}\right], \\ &= \left[\begin{array}{ccc}\mathbf{HS} & \left(\mathbf{J}^T\right)^N\mathbf{HS} & \mathbf{J}^N\mathbf{HS}\end{array}\right].\end{aligned} \tag{3.2}$$

The first, second and the third K columns in matrix $\mathbf{Q}_e$ represents the signature sequences corresponding to the symbols received in the current, previous, and next symbol periods respectively. A matrix $\mathbf{J}^T/\mathbf{J}$ is used to shift the matrix with which it left-multiplies in the upwards/downwards direction by a row, while leaving the elements in the new last/first row to become zeros.

Therefore, using the receiver signature sequence matrix given in Eq(3.2) the received signal at the ρ-th symbol period which is corrupted by ISI, MAI and

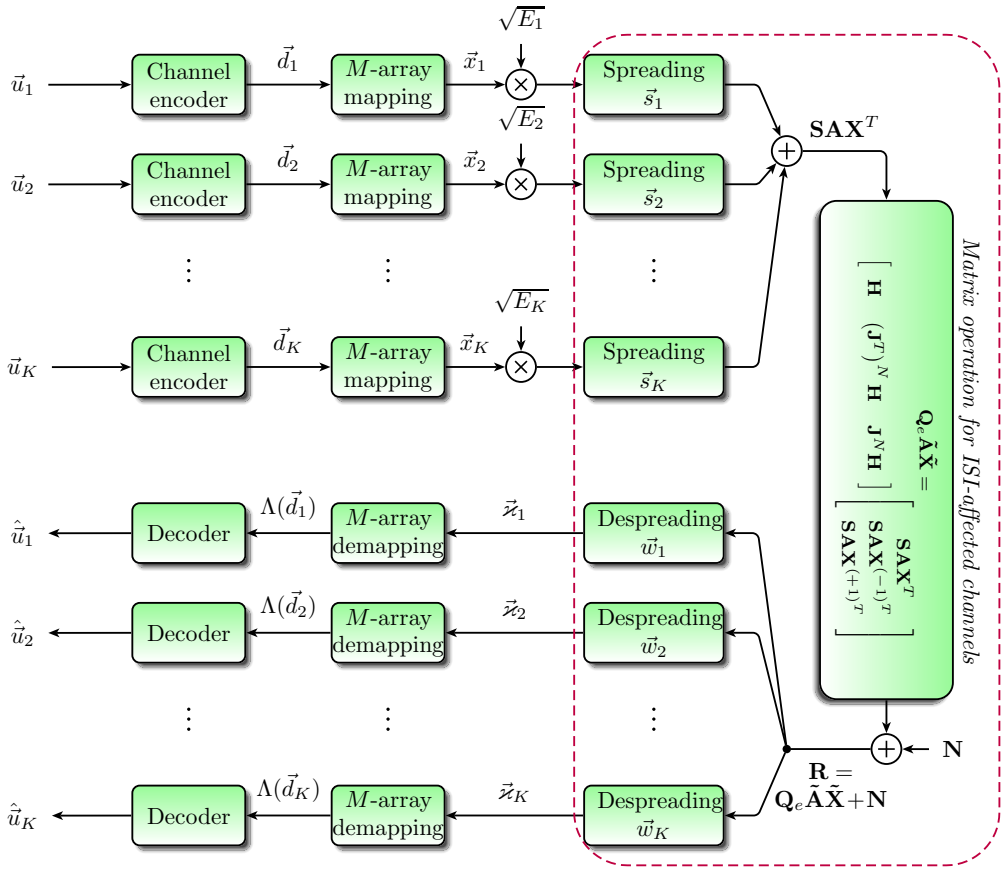

Figure 3.3: The multi-code transmission system model with matrix and vector representations

additive white noise is written as

$$\vec{r}(\rho) = \mathbf{Q}_e \tilde{\mathbf{A}} \tilde{x}(\rho) + \vec{n}(\rho), \tag{3.3}$$

with $\tilde{\mathbf{A}} = \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_1}, \sqrt{E_2}, ..., \sqrt{E_K}\right)$ is a $(3K \times 3K)$-dimensional diagonal matrix which takes the amplitudes or the square roots of the allocated energies for symbols $\tilde{x}(\rho) = \left[\vec{x}^T(\rho), \vec{x}^T(\rho-1), \vec{x}^T(\rho+1)\right]^T$ in the ρ-th, $(\rho-1)$-th and $(\rho+1)$-th symbol periods as its diagonal elements, where the symbol $\otimes$ denotes the Kronecker product and $\vec{x}(\rho) = [x_1(\rho), x_2(\rho), \ldots, x_K(\rho)]^T$. It should be noted that the ρ-th, $(\rho-1)$-th and $(\rho+1)$-th periods correspond to the current, previous and next symbols respectively.

The received signal may also be represented in a matrix form, especially when

block data transmission simulation is carried out. In this chapter, block data transmission is performed to measure the average received SNR, which is then compared to the target SNR on the transmitter end to measure performance of the resource allocation schemes under consideration. The matrix and vector representation of the proposed block data transmission model is given in Figure 3.3, where each channel employs the chip-level matched filtering model presented in Figure 3.2. With a block of information data denoted as a vector $\vec{u}$, each channel is loaded with an $((N_U - N_P) \times 1)$-dimensional data vector of $\vec{u}_k$, before it is encoded to an $(N_U \times 1)$-dimensional encoded data vector $\vec{d}_k$ and mapped to an $(N^{(x)} \times 1)$-dimensional symbol vector $\vec{x}_k = \begin{bmatrix} x_k(1) & x_k(2) & \dots & x_k(N^{(x)}) \end{bmatrix}^T$. This symbol vector, which has been allocated an amplitude of $\sqrt{E_k}$, is spread with the signature sequence $\vec{s}_k$ before it is transmitted over the transmission channel. In matrix form, the received signal is expressed as:

$$\begin{aligned} \mathbf{R} &= \begin{bmatrix} \vec{r}(1) & \vec{r}(2) & \dots & \vec{r}(N^{(x)}) \end{bmatrix}, \\ &= \mathbf{Q}_e \tilde{\mathbf{A}} \tilde{\mathbf{X}} + \mathbf{N}, \end{aligned} \tag{3.4}$$

where the $\left(3K \times N^{(x)}\right)$-dimensional matrix $\tilde{\mathbf{X}} = \begin{bmatrix} \mathbf{X} & \mathbf{X}^{(-1)} & \mathbf{X}^{(+1)} \end{bmatrix}^T$ represents the transmitted symbol matrix, each column of which contains symbols received over three consecutive symbol periods in all $3K$ channels, with $\mathbf{X} = \begin{bmatrix} \vec{x}_1 & \dots & \vec{x}_K \end{bmatrix}$, $\mathbf{X}^{(a)} = \begin{bmatrix} \vec{x}_1^{(a)} & \dots & \vec{x}_K^{(a)} \end{bmatrix}$ for $k = 1, 2, \dots, K$, $\vec{x}_k^{(a)} = \begin{bmatrix} x_k(1+a) & \dots & x_k(N^{(x)}+a) \end{bmatrix}^T$, $a \in \{+1, -1\}$ and $x_k(0) = x_k(N^{(x)}+1) = 0$. The noise signal is also represented in a matrix form as $\mathbf{N}$, each element of which has a variance of σ^2.

A typical despreading technique used to despread the received signal for multi-code receivers is matched filtering, which employs the Rake receiver to capture the received signal at different delayed propagation paths using the so-called Rake receiver fingers. In this case, the matched filter signature sequence $\vec{q}_k$ for code channel k can be used to despread the received signal $\vec{r}(\rho)$ such that the

expected received SNR for code channel k at the input of the decoding unit can be written as

$$\gamma_k = \frac{E_k \left|\vec{q}_k^H \vec{q}_k\right|^2}{\sum_{\substack{j=1 \\ j\neq k}}^{3K} E_j \left|\vec{q}_j^H \vec{q}_k\right|^2 + 2\sigma^2 \vec{q}_k^H \vec{q}_k}, \tag{3.5}$$

with $k = 1, \ldots, K$. This technique however fails to eliminate the ISI effects [44] since it does not consider the ISI components in generating the signature sequence $\vec{q}_k$. Alternatively, the MMSE equalization [74] may be implemented to produce the MMSE coefficient vector $\vec{w}_k$ for channel k as follows (See Appendix A)

$$\vec{w}_k = \frac{\mathbf{C}^{-1}\vec{q}_k}{\vec{q}_k^H \mathbf{C}^{-1}\vec{q}_k}, \tag{3.6}$$

such that $\vec{w}_k^H \vec{q}_k = 1$ and $0 \leq \vec{w}_k^H \vec{q}_j < 1$ for $j \neq k$ and the covariance matrix $\mathbf{C}$ is written as

$$\mathbf{C} = \mathbf{Q}_e \tilde{\mathbf{A}}^2 \mathbf{Q}_e^H + 2\sigma^2 \mathbf{I}_{N+L-1}. \tag{3.7}$$

The covariance matrix in Eq(3.7) consists of the ISI components caused by the previous and the next symbols. Therefore, by taking into account these components in generating the MMSE coefficient vectors using the equation given in Eq(3.6), the cross-correlation between an MMSE coefficient vector and a receiver signature sequence, as given in Eq(3.3), of different code channels can be maximized and the inner-product between an MMSE coefficient vector and a receiver signature sequence of the same code channel is made equal to unity. As a result, the received SNR, which can be written as

$$\begin{aligned}
\gamma_k &= \frac{E_k \left|\vec{w}_k^H \vec{q}_k\right|^2}{\sum_{\substack{j=1 \\ j\neq k}}^{3K} E_j \left|\vec{w}_k^H \vec{q}_j\right|^2 + 2\sigma^2 \vec{w}_k^H \vec{w}_k}, \\
&= \frac{E_k \left|\vec{w}_k^H \vec{q}_k\right|^2}{\vec{w}_k^H \mathbf{C} \vec{w}_k - E_k \left|\vec{w}_k^H \vec{q}_k\right|^2}, \\
&= \frac{E_k \vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k}{1 - E_k \vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k},
\end{aligned} \tag{3.8}$$

may be realized with a less amount of energy for all channels $k = 1, \ldots, K$.

In order to examine the quality of the received signal $\mathbf{R}$, a normalized mean square error (MSE) MSE_k is measured from the despread symbol vector $\vec{\varkappa}_k^T = \vec{w}_k^H \mathbf{R}$, where a normalized MMSE despreading filter coefficient vector $\vec{w}_k = \frac{\mathbf{C}^{-1}\vec{q}_k}{\vec{q}_k^H \mathbf{C}^{-1}\vec{q}_k}$ is applied, as follows [75]:

$$\begin{aligned} \text{MSE}_k &= \text{E}\left\{ \frac{\left(\vec{\varkappa}_k - |h|\sqrt{E_k}\vec{x}_k\right)^H \left(\vec{\varkappa}_k - |h|\sqrt{E_k}\vec{x}_k\right)}{|h|^2 E_k \vec{x}_k^H \vec{x}_k} \right\}, \\ &= 1 - E_k \vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k, \end{aligned} \tag{3.9}$$

for channel $k = 1, 2, \ldots, K$, where $|h|^2 = \vec{h}^H \vec{h}$. This normalized MSE may also be used to calculate the SNR at the output of the despreading unit as follows

$$\gamma_k = \frac{1 - \text{MSE}_k}{\text{MSE}_k}, \tag{3.10}$$

$$= \frac{E_k \vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k}{1 - E_k \vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k}, \tag{3.11}$$

which is equivalent to the SNR equation expressed in Eq(3.8). Therefore, it is essential to ensure that the SNR measured at the output of the despreading filter, using Eq(3.11), is as close as possible to the theoretical SNR given in Eq(3.8).

A computationally-efficient loading method can be implemented to realize an equal data rate on each channel by simply allocating an equal energy to each channel, yielding an SNR on each channel k as $\gamma_k = \frac{E_k/K\left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}{1 - E_T/K\left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}$ based on Eq(3.8). To determine the equal data rate that should be allocated, test packets can be transmitted over each code channel to the receiver before the SNR is measured for each channel at the receiver end and sent back to the transmitter via feedback channels. With the SNR information for each channel known, this equal rate and equal energy scheme determines the minimum SNR among all the channels before the data rate realizable with this minimum SNR

is taken as the data rate that should be equally allocated to all channels. This technique is described in more detail in the next subsection.

3.3.3 The Wasted SNR Problem: Equal Rate and Equal Energy Allocation

As the name implies, this loading technique allocates to each of the K active code channels with an equal energy $E_k = \frac{E_T}{K}$ from the total energy E_T to realize an equal data rate y. Therefore, the resultant SNR for each channel can be written from Eq(3.8) as given by

$$\gamma_k = \frac{(E_T/K)\left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}{1 - (E_T/K)\left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}. \tag{3.12}$$

The equal data rate can be determined by finding the data rate y realizable with the minimum SNR $\min_k \gamma_k$, where $y \in \{b_1, \ldots, b_P\}$. Therefore, y must satisfy the following condition

$$\gamma_k^*(y) \leq \min_k \gamma_k, \tag{3.13}$$

where $\gamma_k^*(y)$ is the target SNR to realize y bits per symbol. When the data, which are transmitted at Ky bits per symbol, are received on the receiver end, a despreading filter is used to despread the received corrupted data using a despreading filter coefficient vector $\vec{w}_k$ for channel $k = 1, \ldots, K$. This filter coefficient vector is generated by inserting the equal energy, E_T/K in Eq(3.7) before calculating the covariance matrix $\mathbf{C}$, which is required to find $\vec{w}_k$ using Eq(3.6).

Although this scheme is more computationally efficient than most of the existing schemes, it suffers from the problem of wasted SNR as the channels with high SNRs are only allocated with a low data rate determined based on the lowest

SNR. The amount of this wasted SNR can be calculated as follows

$$\text{wasted SNR}=\sum_{k=1}^{K}\gamma_k - K\gamma_k^*(y).$$

In order to eliminate this problem, an iterative energy calculation process can be used to realize an equal data rate with an equal SNR.

The energy allocated to each channel is iteratively adjusted until the SNR of the channel is equal to the target minimum SNR required for realizing a data rate of y. This iterative calculation process is repeated for all values of y until the sum of the allocated energy when Ky bits per symbol are allocated is smaller than or equal to the total energy E_T, while the sum of the allocated energy when Kb_p bits per symbol are allocated is larger than the total energy E_T. This scheme requires the knowledge of the channel impulse response $\vec{h}$ and the noise variance σ^2 available at the transmitter end.

The mathematical formulation of this iterative energy calculation process can be produced from Eq(3.17). If the target minimum SNR needed to achieve a data rate y_k in bits per symbol is set as $\gamma_k^*(y_k) = \Gamma(2^{y_k} - 1)$, the left-hand side of Eq(3.17) can be substituted with $\gamma_k^*(y_k)$ before both sides of this equation are rearranged to yield the energy update formula for channel k at the i-th iteration as

$$E_{k,i+1}(y_k) = \frac{\gamma_k^*(y_k)}{1+\gamma_k^*(y_k)}\left(\vec{q}_k^H \mathbf{C}_i^{-1}\vec{q}_k\right)^{-1}, \tag{3.14}$$

where the current covariance matrix, which is also iteratively updated, is written as

$$\mathbf{C}_i = \mathbf{Q}_e\tilde{\mathbf{A}}_i^2\mathbf{Q}_e^H + 2\sigma^2\mathbf{I}_{N+L-1}. \tag{3.15}$$

at the i-th iteration. As the diagonal matrix $\tilde{\mathbf{A}}_i^2$ contains the square roots of the currently updated energies $\sqrt{E_{1,i}(y_k)},\ldots,\sqrt{E_{K,i}(y_k)}$, the covariance matrix $\mathbf{C}_i$ is a function of the energy $E_{k,i}(y_k)$ in each channel $k = 1,\ldots,K$. Therefore the iterative process of updating the energy $E_{k,i}(y_k)$ is done by al-

ternately and iteratively updating both the energy as well as the covariance matrix. This iterative process continues until the calculated energy converges, where $E_{k,i}(y_k) \approx E_{k,i-1}(y_k)$.

The step-by-step procedures of the iterative energy calculation by which margin adaptive optimization is run are summarized as follows

3.3.4 The Iterative Energy Calculation Process

1. Initialize $i = 1$, $\gamma_k^*(y_k) = \Gamma(2^{y_k} - 1)$ and $\tilde{\mathbf{A}}_1 = \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_{1,1}}, \ldots, \sqrt{E_{K,1}}\right)$, where $E_{k,1} = \frac{E_T}{K}$ for $k = 1, 2, \ldots, K$.

2. Calculate the current covariance matrix $\mathbf{C}_i = \mathbf{Q}_e \tilde{\mathbf{A}}_i^2 \mathbf{Q}_e^H + 2\sigma^2 \mathbf{I}_{N+L-1}$.

3. Calculate the current energy $E_{k,i+1}(y_k) = \dfrac{\gamma_k^*(y_k)}{1 + \gamma_k^*(y_k)} \left(\vec{q}_k^H \mathbf{C}_i^{-1} \vec{q}_k\right)^{-1}$ for $k = 1, 2, \ldots, K$.

4. Calculate the current amplitude diagonal matrix $\tilde{\mathbf{A}}_{i+1} = \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_{1,i+1}(y_1)}, \ldots, \sqrt{E_{K,i+1}(y_K)}\right)$.

5. Set $i = i + 1$. If $E_{k,i}(y_k) \neq E_{k,i-1}(y_k)$for $k = 1, 2, \ldots, K$ or $i \leq I_{max}$, go to step (2). Otherwise $E_k(y_k) = E_{k,i}(y_k)$.

Typical margin adaptive optimization techniques are developed for realizing an equal data rate on each code channel. However, as the required energy is minimized, the margin or the residual energy which is not used to transmit any useful information is now maximized. To further utilize this residual energy for enhancing the transmission data rate improvement, rate adaptive optimization can be implemented. In this chapter, the proposed two-group resource allocation scheme is shown to be a combination of the margin adaptive and the rate adaptive optimizations which minimize the required energy per symbol and maximize the total transmission rate in bits per symbol for the given total symbol energy.

In the next section, an account of the proposed margin adaptive optimization method used as part of the two-group resource allocation scheme is given along with the descriptions of a number of existing margin adaptive optimization techniques.

3.4 Margin Adaptive Optimization

The commonly adopted margin adaptive optimization technique is the equal rate margin adaptive optimization. This loading method allocates an equal data rate y in each of the K channels such that the total data rate is Ky, which should be realized by minimizing the required energy $E_k(y)$ for all channels $k = 1, 2, \ldots, K$. For a given multi-code transmission system with K parallel code channels, each of which realizable with y bits per symbol, the margin adaptive optimization problem can be formulated as follows

$$\begin{aligned} &\min\left(\textstyle\sum_{k=1}^{K} E_k(y)\right) \leq E_T \\ &s.t. \max R_T = \max_y (Ky) \\ &and \;\; y = b_1, \ldots, b_{p+1} \end{aligned} \tag{3.16}$$

The energy minimization in this optimization problem is performed such that the expected received SNR γ_k for every channel $k = 1, 2, \ldots, K$ as given in the right-hand side of Eq(3.8) is equal to the target minimum SNR $\gamma_k^* = \Gamma(2^y - 1)$, where Γ is the gap value of the MCS used to realize y bits per symbol, rendering

$$\Gamma\left(2^y - 1\right) = \frac{E_k(y)\left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}{1 - E_k(y)\left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}. \tag{3.17}$$

To ensure that the highest possible equal data rate b_p is realized, the following condition is checked

$$\sum_{k=1}^{K} E_k(b_p) \leq E_T < \sum_{k=1}^{K} E_k(b_{p+1}), \tag{3.18}$$

where $b_p \in \{b_1, \ldots, b_P\}$ is selected as the data rate that should be equally allocated to each channel for a given total energy E_T.

As long as the sum of the allocated energy $\sum_{k=1}^{K} E_k(y_k)$ is less than the total energy E_T for $y_k = b_j$, the iterative energy calculation process explained before is then repeated for a higher data rate of $y_k = b_{j+1}$. This process is repeated until the condition or the inequality given in Eq(3.18) is satisfied. Therefore, the total amount of allocated energy when this margin adaptive equal rate optimization is implemented is $\sum_{k=1}^{K} E_k(b_p)$. This amount of energy is the minimum total amount of energy required to realize b_p bits per symbol on each channel for the given target SNR, $\gamma_k^*(b_p)$, the channel impulse response, the noise variance and the covariance matrix. Therefore, the remaining energy $E_T - \sum_{k=1}^{K} E_k(b_p)$, which is also known as the residual energy is maximized.

If the bit granularities or the differences between any two adjacent data rates are reduced, the residual energy will be reduced as more energy is utilized to allocate the data rate, hence increasing the total data rate. Having smaller bit granularities requires a larger number of data rates, which results in a relatively higher computational load. Instead of using more data rates with smaller bit granularities, a smaller number of data rates with large bit granularities can be used with the proposed two-group resource allocation scheme to avoid the increase in the computational complexity while increasing the total data rate. If these few data rates are realized at lower gap values, which yield lower required energies to realize the data rates at the target packet error rate, the total data rate can be further enhanced. For this reason, the CPP scheme incorporated with the QAM scheme is incorporated in order to control and reduce the gap value so that the total data rate can be increased.

In the next section the rate adaptive optimization, which is an integral part of the two-group resource allocation scheme, is presented.

3.5 Rate Adaptive Optimization

Rate adaptive optimization is applied to maximize the transmission data rate for a limited total energy as the constraint. Optimum techniques of this kind are also proposed in literature to ensure the maximum transmission data rate is realizable in multi-code transmission systems. This section gives a brief account of the optimum approaches which exist in literature before describing the proposed rate adaptive optimization technique to utilize the residual energy and further enhance the transmission data rate.

Before briefly describing the existing optimum techniques, this rate adaptive optimization problem is formulated in the next subsection.

3.5.1 Rate Adaptive Optimization Formulation

A multi-code CDMA system with K code channels, where each of these channels is assigned with a signature sequence $\vec{s}$ used to spread the information data which can be realized at b_p bits per symbol for $p = 1, \ldots, P$, is considered. If the total energy is E_T, the rate adaptive optimization problem for finding the maximum total data rate R_T in bits per symbol in all channels can be formulated as follows:

$$\begin{aligned} \max R_T &= \max_{k=1,2,\ldots,K} \sum_{k=1}^{K} b_{p_k} \\ s.t. E_{sum} &= \sum_{k=1}^{K} E_k\left(b_{p_k}\right) \leq E_T \end{aligned} \tag{3.19}$$

where p_k is the data rate index allocated to channel k and falls between one and P. The energy $E_k(b_{p_k})$ required to realize b_{p_k} bits per symbol at channel k is subject to the noise and the interference components, including both MAI and ISI. With Eq(3.14), this energy is formulated as $E_k(b_{p_k}) = \frac{\Gamma\left(2^{b_{p_k}}-1\right)}{1+\Gamma\left(2^{b_{p_k}}-1\right)} \left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)^{-1}$. If no interference components, neither ISI no MAI,

are taken into account, then the required energy to realize b_{p_k} is expressed as:

$$E_k(b_{p_k}) = \frac{2\sigma^2\Gamma}{|h|^2}\left(2^{b_{p_k}} - 1\right), \tag{3.20}$$

for $p_k = 1, 2, \ldots, P$ and $k = 1, 2, \ldots, K$. Apart from this formulation, rate adaptive optimization can also be formulated as an incremental energy model, in which the incremental energy is used to calculate the amount of required energy. This formulation is briefly given in the next subsection.

Incremental Energy Model

As there are a number of data rates $b_1, b_2, \ldots, b_P$ offered and which are to be realized in a multi-code CDMA system, a data rate b_{p_k} for $p_k = 1, \ldots, P$ and $k = 1, \ldots, K$ can also be written as

$$b_{p_k} = b_{p_k-1} + \beta_{p_k}, \tag{3.21}$$

where β_{p_k}, which is the difference between the two adjacent data rates, b_{p_k} and b_{p_k-1}, is known as the bit granularity. To increase the data rate b_{p_k-1} by a bit granularity of β_{p_k}, there is a requirement for an incremental energy of $e_k(b_{p_k})$.

In the presence of interference, any increase in the energy in any channel k will yield a different amount of interference experienced by other channels. Therefore, if the target SNR in each channel needs to be maintained, the energy increase in this channel will cause an increase in the energy required by other channels due to the increase in interference, which affects the other channel. Hence, calculating the incremental energy for the case of interfered channels takes into account the energy change in other channels, which actually requires the iterative energy adjustment to achieve the target SNR. Therefore, the incremental energy $e_k(b_{p_k})$ that results when a bit granularity of β_{p_k} is realized

at channel k is written as

$$e_k(b_{p_k}) = E_s(b_{p_k+1}) - E_{so}, \tag{3.22}$$

where E_{so} is the original total energy allocated when b_{p_j} is loaded to channel $j = 1, \ldots, K$ and $E_s(b_{p_k+1})$ is the new total energy allocated when the data rate at channel k is increased by β_{p_k} from b_{p_k} to b_{p_k+1} bits per symbol.

As for the interference-free channels, the incremental energy required when β_{p_k} is loaded to channel k, which is originally realized at b_{p_k}, is expressed as

$$e(b_{p_k}) = E(b_{p_k}) - E(b_{p_k-1}), \tag{3.23}$$

which is equal for any channel since the increase in energy at channel k does not affect the energy that should be allocated in other channels.

Consequently, the rate adaptive optimization can be reformulated as an incremental energy model as follows

$$\begin{aligned} &\max R_T = \max_{\{z_l\}_{l=1}^{P}} \left(\sum_{l=1}^{P} \beta_l z_l \right) \\ &s.t. E_{sum} = \sum_{l=1}^{P} e(b_l) z_l \leq E_T \\ &and \quad 0 \leq z_l \leq K \quad l = 1, 2, ..., P \end{aligned} \tag{3.24}$$

with z_l as the number of channels which are loaded with β_l bits per symbol from the lower data rate of b_{l-1} bits per symbol. Therefore, resolving this incremental energy problem requires a search for the optimum number z_l of channels which should be allocated β_l bits per symbol such that the total data rate $R_T = \sum_{l=1}^{P} \beta_l z_l$ is maximized, subject to the total energy constraint E_T.

In the next subsection, the proposed two-group resource allocation for the case of interference-free channels is described based on the incremental energy model. The results obtained from running the two-group resource allocation scheme

demonstrates that this loading method is optimum for interference-free channels and requires less computational complexity as compared to the other optimum techniques aforementioned.

3.5.2 The Two-group Resource Allocation Scheme for Interference-free Channels

The two-group resource allocation scheme is proposed to maximize the total data rate R_T by realizing two adjacent data rates into two groups of channels. It is the optimal solution of the rate adaptive optimization problem formulated in the incremental energy model, which is given in Section 3.5.1. It can be shown that the optimal solution for this rate adaptive optimization problem achieved when

$$z_l = \begin{cases} K & \text{if } l = 1, ..., p, \\ m & \text{if } l = p+1, \\ 0 & \text{if } l = p+2, ..., P, \end{cases} \tag{3.25}$$

where $0 \leq m < K$, yielding a maximum total data rate as

$$\begin{aligned} R_T &= K \sum_{l=1}^{p} \beta_l + m\beta_{p+1}, \\ &= (K-m) \sum_{l=1}^{p} \beta_l + m \sum_{l=1}^{p+1} \beta_l, \\ &= (K-m) b_p + m b_{p+1}. \end{aligned} \tag{3.26}$$

A graphical illustration of the resource allocation process based on the incremental energy model from which the two-group resource allocation approach can be deduced is shown in Figure 3.4, where $e_1(b_l) = e_2(b_l) = \cdots = e_K(b_l)$ and $e_k(b_l) \geq e_k(b_{l-1})$ for $l = 1, 2, \ldots, P$ and $k = 1, 2, \ldots, K$. Since the incremental energy, increases with respect to l, the energy allocation begins from $l = 1$ and proceeds so long as the sum of the allocated energy $\sum_{l=1}^{P} e(b_l) z_l$ is less than the total constraint energy E_T. It can be seen from this figure that for the

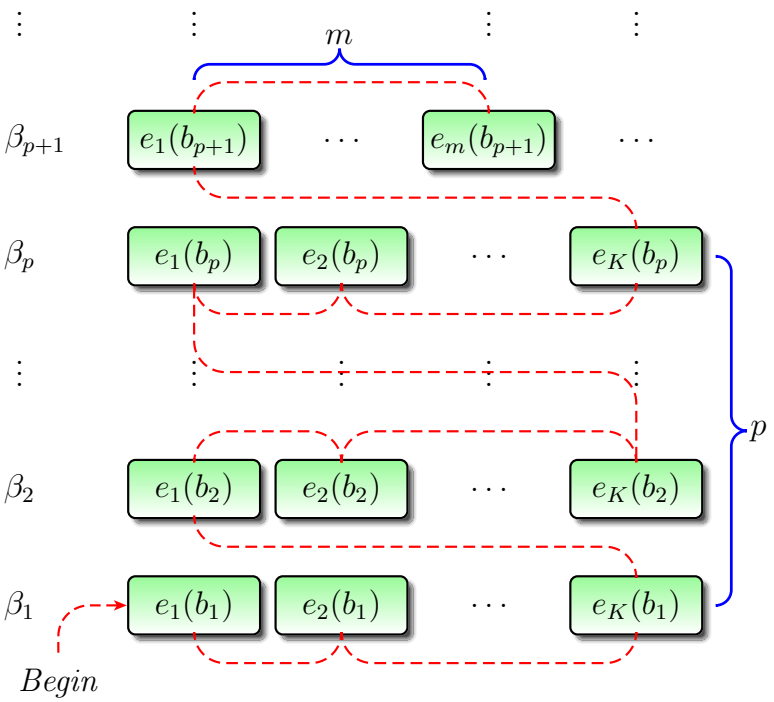

Figure 3.4: A graphical illustration of the resource allocation based on the incremental energy model

interference-free case, the maximum data rate R_T is equivalent to the equation given in Eq(3.26), where there are only two variables needed: p and m.

The first variable, p, is the lower data rate b_p, which is realized on each of the $K-m$ channels in the first group. The second group of channels contains the remaining m channels, each of which is loaded with the higher data rate of b_{p+1}. To ensure that the total data rate R_T is maximized, p needs to be maximized as $b_1 < b_2 < \cdots < b_P$, where p may take any integer values between 1 and P. This is subject to the total energy E_T, the channel gain $|h|^2$, the noise variance σ^2 and the gap value Γ. Therefore, this data rate index p can be determined by satisfying the following condition

$$\begin{aligned} K\sum_{l=1}^{p} e(b_l) &\leq E_T < K\sum_{l=1}^{p+1} e(b_l), \\ KE(b_p) &\leq E_T < KE(b_{p+1}). \end{aligned} \tag{3.27}$$

From the condition given in Eq(3.27) and the definition of the energy $E(b_p) = \frac{2\sigma^2\Gamma}{|h|^2}\left(2^{b_p}-1\right)$, as given in Eq(3.20), this lower data rate index p can be determined as:

$$p = \min\left(\left\lfloor \frac{\log_2\left(1+\frac{|h|^2E_T}{2K\sigma^2\Gamma}\right)}{\beta}\right\rfloor, P\right), \tag{3.28}$$

assuming that $\beta_l = \beta$ for $l = 1, 2, \ldots, P$.

Having the value of p, the number m of channels in the second group of channels, each of which is loaded with the higher data rate b_{p+1}, can be now determined. This number should also be maximized to ensure that the total data rate R_T is likewise maximized. For this reason the difference between the total amount of allocated energy $KE(b_p) + me(b_p)$ should be minimized to be less than the incremental energy $e(b_p)$ so that the energy can be maximally used for transmitting the data. In other words, this number, m, should satisfy the following condition or inequality:

$$\begin{aligned} KE(b_p) + me(b_{p+1}) &\leq E_T < KE(b_p) + (m+1)e(b_{p+1}), \\ 0 &\leq E_T - (KE(b_p) + me(b_{p+1})) < e(b_{p+1}). \end{aligned} \tag{3.29}$$

Therefore, with the knowledge of the lower data rate index p and the energy equation given in Eq(3.20), the number m of channels in the group loaded with the higher data rate b_{p+1} can be determined as follows:

$$m = \left\lfloor \frac{E_T - KE(b_p)}{e(b_{p+1})} \right\rfloor. \tag{3.30}$$

Having the values of the lower data rate index p and the number m of the channels in the group loaded with the higher data rate b_{p+1}, the necessary adaptive modulation and coding (AMC) can be carried out to generate the data at b_p bits per symbol in each channel in the first group of $K-m$ channels and b_{p+1} bits per symbol in each channel in the second group of m channels.

In the next subsection, the rate adaptive optimization methods for the case of interfered channels are described including the proposed two-group resource allocation scheme.

3.5.3 Rate Adaptive Optimization Techniques over Channels with ISI and MAI

When frequency selective channels are considered with multiple delayed propagation paths, the orthonormality of the employed signature sequences is lost, rendering non-zero cross-correlations which give rise to interference, hence increasing the energy required to realize the data rates. Therefore, allocating energy over these interfered channels is challenging as to ensure that the energy can be properly allocated in order to maximize the total data rate.

One possible way of optimizing the total data rate is by modifying the Levin-Campello algorithm [54–56], which is optimal in the case of interference-free channels, in order to be implemented over frequency selective channels, as will be explained next.

Modified Levin-Campello Algorithm

Unlike the case of interference-free channels, the incremental energy required when the data rate is increased by β_{p_k} bits per symbol over channel k depends on the change in the energy allocated to all channels, as given in Eq(3.22). This is because any increase in the energy of a channel k will increase the effect of interference caused by this same channel to the rest of the channels, rendering an increase in the energy allocated to these channels to maintain their target SNRs. In other words, the iterative energy calculation must be invoked whenever there is any change in the energy allocated to any channel in order to ensure that the target SNR is achieved in all channels.

In order to select the channel that should be loaded with the additional data

rate of β_{p_k}, which is the bit granularity, this modified Levin-Campello algorithm measures the incremental energy per bit

$$e_k^{(b)} = \frac{E_s(b_{p_k+1}) - E_{so}}{b_{p_k+1} - b_{p_k}}, \tag{3.31}$$

for channel $k = 1, \ldots, K$ before selecting the channel having the minimum incremental energy per bit $\min_{k=1,\ldots,K} e_k^{(b)}$ to be allocated with the additional data rate aforementioned. This process continues until the difference between the total constrained energy and the total allocated energy per bit is less than the minimum incremental energy per bit, which is required to increase the total allocated energy with the additional data rate. This process will also terminate if the highest realizable data rate b_P has been allocated to all channels.

Although this allocation scheme is optimum, the extensive iterative energy calculation required causes a significant increase in the computational complexity, which in turn, is prohibitive in practical systems.

The next section gives the full account of the implementation of the proposed two-group resource allocation scheme over the channels with ISI and MAI. This proposed method has been shown to have relatively low computational complexity whilst enhancing the realizable total data rate.

Two-group Resource Allocation Scheme over Channels with ISI and MAI

When resource allocation is implemented over frequency selective channels which cause ISI and MAI, the energy for each channel needs to be iteratively updated in order to achieve the target SNR needed for realizing the desired data rate. The same situation applies to the two-group resource allocation scheme. To realize the total data rate of $R_T = (K-m)b_p + mb_{p+1}$, this proposed method

first finds the lower data rate index p by satisfying the following inequality:

$$\sum_{k=1}^{K} E_k^{(0)}(b_p) \leq E_T < \sum_{k=1}^{K} E_k^{(0)}(b_{p+1}). \tag{3.32}$$

The process of finding p using the condition in Eq(3.32) is actually the margin adaptive optimization approach, in which the energy required to realize the desired data rate b_p is minimized using the iterative energy calculation process mentioned in Section 3.3.4. The remaining residual energy $E_T - \sum_{k=1}^{K} E_k^{(0)}(b_p)$ is then utilized to realize an additional data rate of $m\beta_{p+1}$ bits per symbol using m channels, each of which is loaded with β_{p+1} bits per symbol in addition to the current b_p bits per symbol. In order to find m, the following inequality should be satisfied:

$$0 \quad \leq \quad E_T - \sum_{k=1}^{K} E_k^{(0)}(b_p) - \sum_{j=0}^{m} e_j(b_{p+1}) < e_{m+1}(b_{p+1}), \tag{3.33}$$

with the incremental energy $e_j(b_{p+1})$ now redefined as:

$$\begin{aligned} e_j(b_{p+1}) = &\left(\sum_{k=0}^{j} E_k^{(j)}(b_{p+1}) + \sum_{k=j+1}^{K} E_k^{(j)}(b_p) \right) \\ &- \left(\sum_{k=0}^{j-1} E_k^{(j-1)}(b_{p+1}) + \sum_{k=j}^{K} E_k^{(j-1)}(b_p) \right), \end{aligned} \tag{3.34}$$

for $j = 0, 1, \ldots, K-1$, where $E_0^{(j)}(b_{p+1}) = 0$ and $e_0(b_{p+1}) = 0$.

The new energy notation $E_k^{(j)}(y_k)$ represents the energy allocated to channel k to realize a data rate of y_k when j channels are each allocated the higher data rate of b_{p+1} bits per symbol, where $y_k = b_p$ for $k = m+1, \ldots, K$ and $y_k = b_{p+1}$ for $k = 0, 1, \ldots, m$. The reason for this new energy definition has been previously alluded to, as any change in the energy allocated to any channel will change the energy allocated to the rest of the channels if the target SNR on each channel is to be maintained to realize the desired data rate. With this new definition of incremental energy, the decision rule to determine m given in

Eq(3.33) can be further simplified as follows:

$$0 \leq E_T - \sum_{k=1}^{K} E_k^{(0)}(b_p) - \sum_{j=0}^{m} e_j(b_{p+1}) < e_{m+1}(b_{p+1}),$$
$$0 \leq E_T - \left(\sum_{k=0}^{m} E_k^{(m)}(b_{p+1}) + \sum_{k=m+1}^{K} E_k^{(m)}(b_p) \right) < e_{m+1}(b_{p+1}),$$
$$\sum_{k=0}^{m} E_k^{(m)}(b_{p+1}) + \sum_{k=m+1}^{K} E_k^{(m)}(b_p) \leq E_T < \sum_{k=0}^{m+1} E_k^{(m+1)}(b_{p+1}) + \sum_{k=m+2}^{K} E_k^{(m+1)}(b_p),$$

which can be further simplified to

$$\sum_{k=1}^{K} E_k^{(m)}(y_k) \leq E_T < \sum_{k=1}^{K} E_k^{(m+1)}(y_k^{(m+1)}), \tag{3.35}$$

where $y_k^{(m+1)} = b_{p+1}$ for $k = 0, 1, \ldots, m$ and $y_k^{(m+1)} = b_p$ for $k = m+1, \ldots, K$.

When both p and m are known, the AMC, energy allocation and spreading processes are then implemented in order to produce the data symbols before they are transmitted over the channels.

Figure 3.5 shows the implementation of the proposed two-group resource allocation scheme over frequency selective channels with ISI and MAI in a multi-code transmission system. The K code channels in this figure are divided into two groups, where the first m channels are loaded with the higher data rate b_{p+1} bits per symbol and the remaining channels are loaded with the lower data rate b_p bits per symbol. After p and m are discovered, the AMC, energy allocation and spreading operations are carried out accordingly before the symbols are transmitted to the receiver over the transmission channel.

On the receiver end, the despreading operation is carried out on the received signal before it is passed on to the demodulation and decoding units. In Figure 3.5, the soft-output demodulation operation [32], which produces the channel soft output as a measure of the reliability of the received symbol, is considered as one of the components employed in the decoding process.

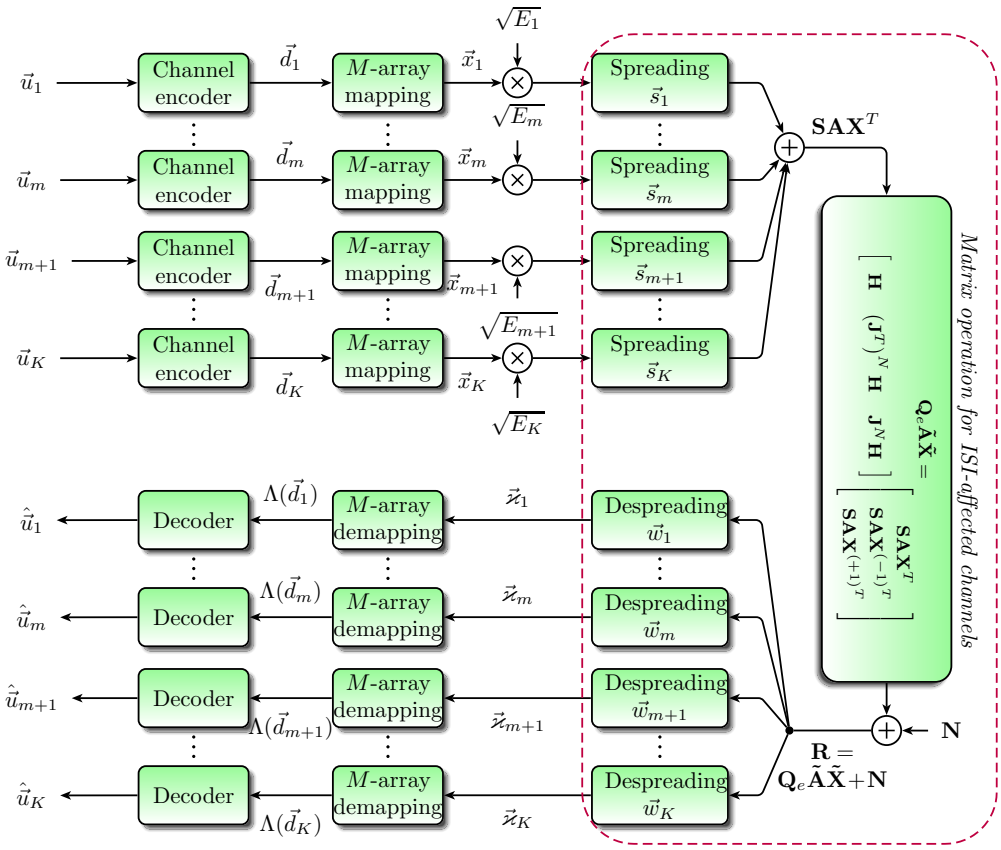

Figure 3.5: A multi-code transmission system in which the two-group resource allocation scheme is incorporated

Although the two-group resource allocation is considered optimal in the case of interference-free channels, performance in frequency selective channels might be affected by the presence of ISI and MAI components which severely degrade the SNR in the channels. The varying nature of these SNRs should be addressed as to further enhance its performance. In some cases, these SNRs have a considerably large variance, hence needs further enhancement measures such as optimal signature sequences employment, channel removal and channel ordering to maximize the total data rate allocated by the two-group resource allocation scheme. Optimal signature sequence design and channel removal methods have been currently considered and studied in the research group [75] as the future focus of the group. Hence these enhancement methods are beyond the scope of this thesis. However, one enhancement measure is proposed and discussed in the next section with the aim to improve performance of the two-group resource allocation scheme in the case of frequency selective channels having ISI and MAI.

3.6 Performance Enhancement for The Two-group Resource Allocation Scheme

A performance enhancement measure is proposed to enhance performance of the two-group resource allocation scheme. This technique is referred to as channel ordering, which orders the channels such that the channel which requires the least amount of energy to realize the data rate will be regarded as the first channel and the last channel will be the channel which requires the largest amount of energy to realize the same data rate. The following subsection briefly describes this enhancement technique.

3.6.1 Channel Ordering

Channel ordering [76] is performed so that the number m of channels allocated the higher data rate b_{p+1} is increased to improve the total data rate $R_T = (K-m)b_p + mb_{p+1}$ (see Section 3.5.2 (Eq(3.26)) and Section 3.5.3 for further description on the realization of this total data rate). There are two approaches which are investigated. The first approach is termed as a double-loaded channel ordering scheme and the second method is referred to as a single-loaded channel ordering scheme.

Double-loaded Channel Ordering Scheme

Double-loaded channel ordering orders the channels based on the incremental energy $e_j(b_{p+1})$, which is expressed in Eq(3.34), for $j = 1, \ldots, K-1$. The channel order is determined by identifying j channels which results in the least amount of incremental energy $e_j(b_{p+1})$ among all possible selections of j channels. Therefore, these j channels will be regarded as the first j channels in the new channel order. This channel ordering procedure is continued until $j = K-1$, where the remaining K-th channel will become the last channel in the new channel order.

In the two-group resource allocation scheme implementation, this channel order identification process is run after the lower data rate index b_p has been identified. It is one of the procedures taken by the two-group resource allocation scheme when it is identifying the number m of channels which will be loaded with the higher data rate b_{p+1}. Algorithm 1 shows the step-by-step procedures of determining the channel order as to maximize the number m of channels loaded with the higher data rate b_{p+1}.

With all parameters $\mathbf{H}, \vec{s}_k, \mathbf{Q}_e, \varepsilon_\tau, \sigma^2, K, E_T, \Gamma, b_p, b_{p+1}, I_{max}, E_k(b_p)$ for $k = 1, \ldots, K$ initialized, the algorithm starts by choosing the first channel to be loaded with the additional data rate β_{p+1} from b_p as shown in line 10 where the

for loop starts from channel $z = 1$, the rest of the channels are set to realize the same data rate b_p. Then, the iterative energy calculation is run until the target SNR, $\gamma_k^*(y_l)$ is achieved in all channels. This iterative energy calculation is done in the while loop at line 17. After completing this iterative process for $z = 1$, the algorithm continues again for higher values of z whilst recording the current minimum total allocated energy $E_{TK} = \sum_{k=1}^{K} E_k^{(j)}$, which is executed in line 24. The minimum total allocated energy should correspond to the total energy allocated when the channel j_{temp}, which requires the least amount of additional energy, is loaded with an additional data rate of β_{p+1} from b_p to b_{p+1} bits per symbol. Therefore, channel j_{temp}, after completing the *while* loop in line 17, is considered as the new first channel.

The algorithm checks the current total allocated energy $\sum_{k=1}^{K} E_k^{(j)}(y_k)$ as given in line 7. If it is less than the total constrained energy E_T, then the algorithm continues to find the next channel after channel j_{temp} which requires the least amount of incremental energy to increase the data rate. The same procedures described in the previous paragraph are followed. If the current total allocated energy $\sum_{k=1}^{K} E_k^{(j)}(y_k)$ is larger than the total constrained energy E_T, the *while* loop which starts from line 7 terminates. The number m of channels is then taken as $j - 1$ and the allocated energy is $E_k^{(m)}(y_k) = E_k^{(j-1)}(y_k)$, where $y_k = b_{p+1}$ for $k = 1, \ldots, m$ and $y_k = b_p$ for $k = m + 1, \ldots, K$.

It can be clearly seen from the procedures shown in Algorithm 1 that this double-loaded channel ordering approach requires significantly high computational complexity. Whereas, a very computationally-efficient channel ordering method known as single-loaded channel ordering scheme is developed and described next.

Algorithm 1: The Double-loaded Channel Ordering Scheme

Data: $\mathbf{H}, \vec{s}_k, \mathbf{Q}_e, \varepsilon_\tau, \sigma^2, K, E_T, \gamma_k^*(b_p), \gamma_k^*(b_{p+1}), b_p, b_{p+1}, I_{max}, E_k(b_p), E_k(b_{p+1})$ for $k = 1, 2, \ldots, K$

Result: $m, E_k^{(m)}(y_k)$ for $k = 1, 2, \ldots, K$

begin
- $j \longleftarrow 0$
- $j_{temp} \longleftarrow 0$
- $\mathcal{K} \longleftarrow \emptyset$
- $y_k \longleftarrow b_p$ for $k = 1, 2, \ldots, K$
- $E_k^{(j)}(y_k) \longleftarrow E_k(b_p)$ for $k = 1, 2, \ldots, K$
- **while** $\sum_{k=1}^{K} E_k^{(j)}(y_k) \leq E_T$ **do**
  - $E_{TK} \longleftarrow \sum_{k=1}^{K} E_k(b_{p+1})$
  - $j \longleftarrow j + 1$
  - **for** $z = 1, 2, \ldots, K$ **do**
    - **if** $z \notin \mathcal{K}$ **then**
      - $i \longleftarrow 1$
      - $y_k \longleftarrow b_{p+1}$ for all $l \in \mathcal{K}$ and $l = z$
      - $y_k \longleftarrow b_p$ for all $l \notin \mathcal{K}$ and $l \neq z$
      - $E_{k,i-1}^{(j)}(y_k) \longleftarrow 0$ for $k = 1, 2, \ldots, K$
      - $E_{k,i}^{(j)}(y_k) \longleftarrow E_T/K$ for $k = 1, 2, \ldots, K$
      - **while** $\left(\sum_{k=1}^{K} \left(E_{k,i}^{(j)}(y_k) - E_{k,i-1}^{(j)}(y_k) \right)^2 > \varepsilon_\tau \right)$ **or** $(i < I_{max})$ **do**
        - $\tilde{\mathbf{A}}_i \longleftarrow \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_{1,i}^{(j)}(y_1)}, \ldots, \sqrt{E_{K,i}^{(j)}(y_K)} \right)$
        - $\mathbf{C}_i \longleftarrow \mathbf{Q}_e \tilde{\mathbf{A}}_i^2 \mathbf{Q}_e^H + 2\sigma^2 \mathbf{I}_{N+L-1}$
        - $E_{k,i+1}^{(j)}(y_k) \longleftarrow \frac{\gamma_k^*(y_k)}{1+\gamma_k^*(y_k)} \left(\vec{q}_k^H \mathbf{C}_i^{-1} \vec{q}_k \right)^{-1}$ for $k = 1, \ldots, K$
        - $i \longleftarrow i + 1$
      - **end**
      - **if** $\sum_{k=1}^{K} E_{k,i}^{(j)}(y_k) \leq E_{TK}$ **then**
        - $E_{TK} \longleftarrow \sum_{k=1}^{K} E_{k,i}^{(j)}(y_k)$
        - $E_k^{(j)}(y_k) \longleftarrow E_{k,i}^{(j)}(y_k)$ for $k = 1, 2, \ldots, K$
        - $j_{temp} \longleftarrow j$
      - **end**
    - **end**
  - **end**
  - $\mathcal{K} \longleftarrow \{\mathcal{K}\ j_{temp}\}$
- **end**
- $m \longleftarrow j - 1$
- $E_k^{(m)}(y_k) = E_k^{(j-1)}(y_k)$ for $k = 1, 2, \ldots, K$

end

Single-loaded channel ordering

A single-loaded channel ordering scheme orders the channel based on the difference between two energies, $E_k(b_p)$ and $E_k(b_{p+1})$ required to realize two data rates b_p and b_{p+1} bits per symbol, for every channel k. Both $E_k(b_p)$ and $E_k(b_{p+1})$ are iteratively calculated when each channel is equally loaded with b_p and b_{p+1} bits per symbol respectively. These energies have been calculated during the process of finding the lower data rate index p, which is the margin adaptive optimization process, as the first major step of the two-group resource allocation scheme before determining the number m of the channels that should be allocated with the higher data rate b_{p+1}. The channel which has the minimum energy difference will be regarded as the first channel that should be loaded with the higher data rate b_{p+1}.

Therefore, unlike the double-loaded channel ordering scheme, there are no further iterative energy calculations required to order the channels using this single-loaded channel ordering scheme. Once the channel order is known, the process of determining the number m of the channels can be run based on this same channel order.

After presenting two methods of channel ordering, a block data transmission model is described in the next section. This model is designed to evaluate the two-group resource allocation scheme along with the performance enhancement measure by transmitting blocks of symbols, which are generated according to the data rate and the energy determined by the two-group scheme.

3.7 Two-group Resource Allocation Scheme: Block Data Transmission Simulation

The block data transmission simulation is important in terms of demonstrating the feasibility of implementing the proposed two-group resource allocation

scheme in practice. One of the major output expected from this simulation is the measured received SNR, which is determined at the input of the decoding unit, or before the actual decoding process begins on the receiver end. This value is measured using Eq(3.11), which is expressed in terms of the normalized MSE given in Eq(3.9). The received SNR for each symbol is the symbol energy divided by the measured interference and noise component. To obtain a better estimate of the received SNR, a very large number of symbols is transmitted so that the measured SNR can be computed by adding all received SNRs before dividing this total average received SNR with the number of the symbols.

This measured SNR is then compared against the target SNR set during the resource allocation process on the transmitter end. The aim is to ensure that the measured received SNR is obtained as close as possible to the target SNR for each channel. If the measured SNR is found to be very close to the expected target SNR, the decoding or detection process is expected to recover the received data successfully whilst ensuring that the desired data rate y_k on channel $k = 1, \ldots, K$ is achieved.

Before running the simulation, it is essential to clearly state all assumptions made.

- The received SNR, the channel impulse response and the additive white noise variance are known to both transmitter and receiver ends.
- The channel impulse response and the noise variance remain unchanged throughout the transmission duration in the simulation.
- Inter-symbol interference (ISI) exists besides multiple-access interference (MAI) and additive white noise.

The link simulation, with the given assumptions, is carried out in four major steps:

1. Initialization and definitions. All required parameters need to be initialized and defined before running the simulation.

2. The resource allocation process. With all parameters initialized and defined, the resource allocation process is run to find the data rate as well as the corresponding energy required to realize it.

3. The transmission process. In this chapter, the transmission process is referred to as the process of coding, modulating, spreading and transmitting the desired information bits at the data rate and the energy determined by using the employed resource allocation scheme.

4. The despreading process. The received data is despread using a despreading filter designed based on a symbol-based MMSE equalization. The received SNR is measured at this stage.

In short, the simulation performed for this chapter includes the proposed resource allocation, coding and modulation, transmission via the added white Gaussian noise channel and finally despreading operation. No decoding is performed in this simulation as the results of interest are the received SNRs which can be measured at the output of the despreading unit.

The next subsection presents all parameters initialized and defined for the simulation.

3.7.1 Initializations and Definitions

To simulate the two-group resource allocation scheme as well as other existing techniques selected for comparison study, the following parameters need to be initialized:

- The realizable data rates b_p for $p = 1, 2, \ldots, P$.
- The gap value to realize these data rates Γ_p for $p = 1, 2, \ldots, P$.

- The channel impulse response vector $\vec{h} = [\ h_0 \quad h_1 \quad \dots \quad h_{L-1}\ ]^T$ and the noise variance σ^2.
- The total energy E_T, the number K of orthonormal signature sequences $\vec{s}_1, \dots, \vec{s}_K$, with a spreading factor of $N = 16$, generated based on Hadamard codes.

With the channel impulse response vector $\vec{h}$ and the signature sequences $\vec{s}_1, \dots, \vec{s}_K$, the receiver signature sequence matrix $\mathbf{Q}_e$, as formulated in Eq(3.2), which takes into account the ISI components apart from the MAI and noise components, can be determined. This is essential to observe the performance of the two-group resource allocation scheme over frequency selective channels with the presence of ISI. In addition to this, a $(3K \times 3K)$-dimensional amplitude diagonal matrix $\tilde{\mathbf{A}}$ is also defined as $\tilde{\mathbf{A}} = \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_1}, \dots, \sqrt{E_K}\right)$ in order to take into account the symbols received in each of the K channels over three symbol periods. These definitions have been briefly described in Section 3.3. With these initialized and defined parameters, the two-group resource allocation scheme implementation for finding the lower data rate index p and the number m of the channels in the group of channels loaded with the higher data rate b_{p+1} is described next.

3.7.2 The p and m Calculation

In order to realize the total data rate $R_T = (K - m)b_p + mb_{p+1}$ bits per symbol (see Section 3.5.2 (Eq(3.26)) and Section 3.5.3 for further description on the realization of this total data rate) using the proposed two-group resource allocation scheme, two important parameters which should be determined are p and m. The step-by-step procedures to find the lower data rate index p are given in Algorithm 2 and the corresponding step-by-step procedures to determine the number m of the channels in the higher data rate group are summarized in Algorithm 3.

Algorithm 2: The p Finder Algorithm

Data: $\mathbf{H}, \vec{s}_k, \mathbf{Q}_e, \varepsilon_\tau, \sigma^2, K, E_T, I_{max}, \gamma_k^*(b_j), b_j$ for $j = 1, 2, \ldots, P$
Result: $p, E_k(b_p), E_k(b_{p+1})$ for $k = 1, 2, \ldots, K$

begin
  $j \longleftarrow 0$
  $b_j \longleftarrow 0$ for $k = 1, 2, \ldots, K$
  $E_k(b_j) \longleftarrow 0$ for $k = 1, 2, \ldots, K$
  while $\sum_{k=1}^{K} E_k(b_j) \leq E_T$ **and** $j \leq P$ **do**
    $j \longleftarrow j + 1$
    $i \longleftarrow 1$
    $E_{k,i-1}(b_j) \longleftarrow 0$ for $k = 1, 2, \ldots, K$
    $E_{k,i}(b_j) \longleftarrow \frac{E_T}{K}$ for $k = 1, 2, \ldots, K$
    while $\left(\sum_{k=1}^{K} \left(E_{k,i}(b_j) - E_{k,i-1}(b_j)\right)^2 > \varepsilon_\tau\right)$ **or** $(i < I_{max})$ **do**
      $\tilde{\mathbf{A}}_i \longleftarrow \mathbf{I}_3 \otimes \text{ diag}\left(\sqrt{E_{1,i}(b_j)}, \sqrt{E_{2,i}(b_j)}, \ldots, \sqrt{E_{K,i}(b_j)}\right)$
      $\mathbf{C}_i \longleftarrow \mathbf{Q}_e \tilde{\mathbf{A}}_i^2 \mathbf{Q}_e^H + 2\sigma^2 \mathbf{I}_{N+L-1}$
      $E_{k,i+1}(b_j) \longleftarrow \frac{\gamma_k^*(b_j)}{1+\gamma_k^*(b_j)} \left(\vec{q}_k^H \mathbf{C}_i^{-1} \vec{q}_k\right)^{-1}$ for $k = 1, 2, \ldots, K$
      $i \longleftarrow i + 1$
    end
    $E_k(b_j) \longleftarrow E_{k,i}(b_j)$ for $k = 1, 2, \ldots, K$
  end
  $p \longleftarrow j - 1$
  $E_k(b_p) \longleftarrow E_k(b_{j-1})$ for $k = 1, 2, \ldots, K$
  $E_k(b_{p+1}) \longleftarrow E_k(b_j)$ for $k = 1, 2, \ldots, K$
end

It can be seen in Algorithm 2 that there are two major parts or groups of steps used to determine p. Each of these two parts is executed by one of the *while* loops in the algorithm. The first part, which corresponds to the second inner *while* loop, consists of the steps from line 11 to line 14. These steps are used to perform the iterative energy calculation, which minimizes the required energy to achieve the target SNR $\gamma_k^*(b_j)$ of data rate b_j. The iteration continues until the energy is converged to $E_k(b_j) = E_{k,i}(b_j) \approx E_{k,i-1}(b_j)$ or the maximum iteration I_{max} is reached. This iterative process is an equal data rate implementation of the generalized iterative energy calculation process summarized in Section 3.3.4.

The second part of this algorithm is executed by the outer *while* loop in Algorithm 2, which starts from line 6 and ends at line 16. This part of the algorithm feeds different data rates into the iterative energy calculation process, which is implemented within this loop, from b_1 to b_{p+1}, so long as the current total allocated energy $\sum_{k=1}^{K} E_k(b_j)$ does not exceed the total energy E_T, where $j = 1, \ldots, p+1$.

Algorithm 3 shows a list of steps required to find m, which is the number of the channels loaded with the higher data rate b_{p+1}. After knowing the lower data rate index p, this algorithm is then invoked to determine m with the knowledge of the energy $E_k(b_p)$ and $E_k(b_{p+1})$ allocated to realize b_p and b_{p+1} respectively besides p and other parameters used in the p finder algorithm. The core process in Algorithm 3 is again the iterative energy calculation process, which starts from line 14 to line 17. This iterative energy process is repeated in a larger *while* loop, which starts from line 7, for a different target data rate $y_{\mathcal{K}_k} = b_{p+1}$ for $k = 1, \ldots, m$, while the rest of the channels are targetted as to be realized with b_p. When the total energy allocated $\sum_{k=1}^{(j)}(y_k)$ is greater than the total constrained energy E_T, then the number of the channels in the higher data rate group should be $m = j - 1$.

Algorithm 3: The m Finder Algorithm

Data: $\mathbf{H}, \vec{s}_k, \mathbf{Q}_e, \varepsilon_\tau, \sigma^2, K, E_T, I_{max}, \gamma_k^*(b_p), \gamma_k^*(b_{p+1}), b_p, b_{p+1}, E_k(b_p), E_k(b_{p+1})$ for $k = 1, 2, \ldots, K$

Result: $m, E_k^{(m)}(y_k)$ for $k = 1, 2, \ldots, K$

begin
- $j \longleftarrow 0$
- $y_k \longleftarrow b_p$ for $k = 1, 2, \ldots, K$
- $E_k^{(j)}(y_k) \longleftarrow E_k(b_p)$ for $k = 1, 2, \ldots, K$
- $[\mathcal{K}_1, \ldots, \mathcal{K}_K] \longleftarrow \text{sort}\left(E_k(b_{p+1}) - E_k(b_p)\right)$
- **while** $\sum_{k=1}^{K} E_k^{(j)}(y_k) \leq E_T$ **do**
  - $j \longleftarrow j + 1$
  - $i \longleftarrow 1$
  - $y_{\mathcal{K}_k} \longleftarrow b_{p+1}$ for $k = 1, \ldots, j$
  - $y_{\mathcal{K}_k} \longleftarrow b_p$ for $k = j+1, \ldots, K$
  - $E_{k,i-1}^{(j)}(y_k) \longleftarrow 0$ for $k = 1, 2, \ldots, K$
  - $E_{k,i}^{(j)}(y_k) \longleftarrow \frac{E_T}{K}$ for $k = 1, 2, \ldots, K$
  - **while** $\left(\sum_{k=1}^{K} \left(E_{k,i}^{(j)}(y_k) - E_{k,i-1}^{(j)}(y_k)\right)^2 > \varepsilon_\tau\right)$ **or** $(i < I_{max})$ **do**
    - $\tilde{\mathbf{A}}_i \longleftarrow \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_{1,i}^{(j)}(y_1)}, \sqrt{E_{2,i}^{(j)}(y_2)}, \ldots, \sqrt{E_{K,i}^{(j)}(y_K)}\right)$
    - $\mathbf{C}_i \longleftarrow \mathbf{Q}_e \tilde{\mathbf{A}}_i^2 \mathbf{Q}_e^H + 2\sigma^2 \mathbf{I}_{N+L-1}$
    - $E_{k,i+1}^{(j)}(y_k) \longleftarrow \frac{\gamma_k^*(y_k)}{1+\gamma_k^*(y_k)} \left(\vec{q}_k^H \mathbf{C}_i^{-1} \vec{q}_k\right)^{-1}$ for $k = 1, \ldots, K$
    - $i \longleftarrow i + 1$
  - **end**
  - $E_k^{(j)}(y_k) \longleftarrow E_{k,i}^{(j)}(y_k)$ for $k = 1, 2, \ldots, K$
- **end**
- $m \longleftarrow j - 1$
- $E_k^{(m)}(y_k) \longleftarrow E_k^{(j-1)}(y_k)$ for $k = 1, 2, \ldots, K$

end

3.7.3 The Transmission Process

When the resource allocation scheme for finding the energy that should be allocated to each channel is completed, the system then runs the adaptive modulation and coding (AMC) to generate the symbols from the information bits. These information bits can be randomly generated before being divided into each channel, as packets represented as a $(1 \times N_k)$-dimensional vector $\vec{u}_k = [u_k(1), \ldots, u_k(N_k)]^T$ for $k = 1, \ldots, K$, where N_k is the number of bits in channel k. The corresponding coding scheme is run on each of these packets to yield a coded vector $\vec{d}_k = [d_k(1), \ldots, d_k(N_k^{(x)})]^T$, which is then modulated as $\vec{x}_k = \left[x_k(1), \ldots, x_k(N_k^{(x)})\right]^T$, where $N_k^{(x)} = \frac{N_k}{y_k}$ for $k = 1, \ldots, K$. This modulation scheme is assumed to be an M-QAM modulation scheme, which produces symbols with an average energy of one. The energy $E_k(y_k)$ calculated from the resource allocation previously is then ready to be allocated to each symbol on channel k, yielding a new symbol vector of $\sqrt{E_k(y_k)}\vec{x}_k$.

These coded QAM symbols are then spread using the signature sequences $\vec{s}_k$ for transmission over the frequency selective channels which have ISI and MAI effects. On the receiver, the received signal is represented as $\mathbf{R} = \mathbf{Q}_e\tilde{\mathbf{A}}\tilde{\mathbf{X}} + \mathbf{N}$, as defined in Eq(3.4).

3.7.4 The Despreading Process

The received signal $\mathbf{R}$ is despread with a despreading filter coefficient vector $\vec{w}_k$, which is obtained from the symbol-based MMSE equalization which generates $\vec{w}_k = \frac{\mathbf{C}^{-1}\vec{q}_k}{\vec{q}_k^H\mathbf{C}^{-1}\vec{q}_k}$. This despreading filter coefficient vector is expressed in Eq(3.6) to obtain a despread received symbol vector $\vec{w}_k^H\mathbf{R}$ for $k = 1, 2, \ldots, K$ where the covariance matrix is $\mathbf{C} = \mathbf{Q}_e\tilde{\mathbf{A}}^2\mathbf{Q}_e^H + 2\sigma^2\mathbf{I}_{N+L-1}$ and the amplitude diagonal matrix is $\tilde{\mathbf{A}} = \mathbf{I}_3 \otimes \text{diag}\left(\sqrt{E_1}, \ldots, \sqrt{E_K}\right)$. This despreading operation is carried out on the assumption that the channel impulse response $\vec{h}$ and the noise variance σ^2 are known to the receiver. The received signal is then despread before

the received SNR for each channel $k = 1, 2, \ldots, K$ is measured using Eq(3.10). This measured SNR can then be compared against the target SNR $\gamma_k^*(y_k)$ for the same channel as to observe how close the measrued received SNR is to the target SNR. It is essential to achieve the received SNR close to the target SNR so that the desired data rate y_k, which typically requires an SNR value equal to the target SNR value, is realizable.

In the next section, some results are presented and discussed to demonstrate and evaluate performance of the proposed two-group resource allocation scheme as well as the existing schemes described in this chapter.

3.8 Numerical Results

In this section, a number of tests were run to observe performance of the existing resource allocation schemes and the proposed two-group resource allocation scheme. In most of the figures produced to plot the results, the optimal values plotted from the waterfilling algorithm [77] is also plotted to compare the results with the optimal case. This optimal case is produced by assuming that the data rates have a zero bit granularity and a zero gap value. For a practical implementation, the proposed two-group resource allocation scheme and other schemes under consideration are only implemented using data rates with non-zero bit granularities.

Each of the schemes under consideration was integrated with the MMSE equalization technique, which is important to perform margin adaptive optimization to minimize the required energy to transmit the data. Different configurations of the MMSE equalization technique were set up to further investigate performance of these resource allocation schemes in the presence of ISI components, besides the MAI and noise components. The total average received SNR, which is the sum of average received SNR in all channels, was measured as the performance parameter against the total input SNR.

After producing separate results of these resource allocation schemes with different MMSE equalization settings, a comparison was made between different resource allocation schemes including the proposed two-group resource allocation scheme. Based on this comparison study, which will be further elaborated at the end of this section, it was observed that the proposed scheme performs better than most of the existing schemes under consideration with relatively low computational complexity. However, it was also found that the equal rate equal SNR loading scheme also performs very closely to that of the proposed scheme if the number of realizable data rates is high and the bit granularities, which are the differences between any two adjacent data rates, are small. Although the system throughput of this scheme was increased, the computational complexity incurred is also increased. It is interesting to note that the proposed two-group resource allocation can be run using a significantly less number of realizable data rates with large bit granularities to produce better performance whilst keeping the computational complexity relatively lower than that of the equal rate equal SNR loading scheme. This improvement is further studied with the implementation of the proposed CPP, which is presented in the previous chapter, along with the two-group resource allocation scheme.

Before presenting the results, the parameters used for the simulation are given in Table 3.1. These parameters are used in all tests run for this chapter, unless otherwise stated.

3.8.1 Implementation of The Two-group and Existing Resource Allocation Schemes with and without ISI

In the first test, the two-group resource allocation scheme and the equal rate margin adaptive loading scheme were run to observe the achievable total average received SNRs at the presence of ISI and without ISI. These total average received SNRs are then compared to the total input SNRs set at the transmitter

Parameter	Value
The number of code channels, K	15
The spreading factor, N	16
The number of propagation paths, L	4
The chip rate, $\frac{1}{T_c}$	3.84 Mchips/s
The additive white Gaussian noise variance, $\sigma^2 = \frac{N_0}{2}$	0.02
Signature sequence matrix, $\mathbf{S}$	Orthonormal
Channel model	ITU Vehicular Channel Model A
The data rates	$0.5, 0.75, 1.0, 1.25, 1.5, 1.75$ $2.0, 2.25, 2.5, 2.75, 3.0, 3.75, 4.0, 4.25$

Table 3.1: The parameters used for the simulation

end. The total average received SNR is calculated from Eq(3.10), which is the average received SNR, γ_k, for channel k, to yield

$$\gamma_T = \sum_{k=1}^{K} \gamma_k. \tag{3.36}$$

For notational convenience, the two-group resource allocation scheme is now represented as o-TG scheme. Likewise, the equal rate margin adaptive loading scheme is now referred to as equal rate equal SNR (ER) loading scheme since both the rate and the SNR is loaded and made equal to each channel. This total average received SNR measurement was run using the block data transmission model described in Section 3.7 using the parameters given in Table 3.1.

Another measurement parameter used in this chapter is the system throughput, which is measured as

$$C_T = \frac{R_T}{NT_c}, \tag{3.37}$$

where $N = 16$ is the spreading factor of the employed orthonormal signature sequences and $\frac{1}{T_c} = 3.84$ Mchips/s is the chip rate.

To begin the test, the two-group resource allocation (o-TG) and the equal rate margin adaptive loading (ER) schemes have been configured to run with a set of data rates $\{b_p\}_{p=1}^{P}$, which are assumed to have zero gap values, over $K = 15$

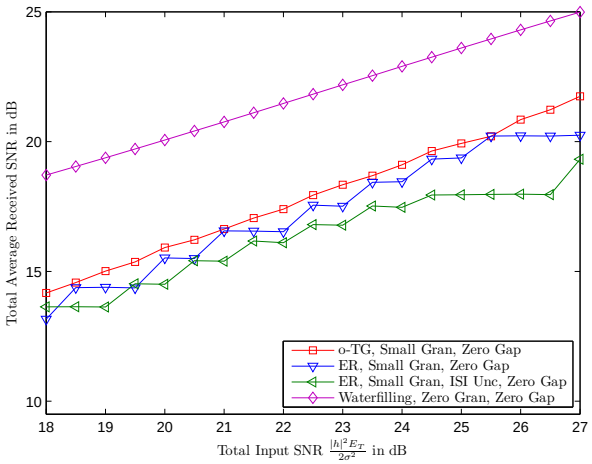

Figure 3.6: The total received SNRs achieved at the input of the decoding units when a range of total input SNRs are fed into the two-group loading (o-TG) scheme and the equal rate margin adaptive loading (ER) scheme with discrete data rates of zero gap values.

parallel channels. These data rates are given in Table 3.1. The total input SNRs, $\frac{|h|^2 E_T}{2\sigma^2}$, have been provided, ranging from 18 to 27 dB. The resultant total average received SNRs, λ_T, measured after running these loading schemes for the given total input SNRs are plotted in Figure 3.6. As observed in this figure, the total average received SNRs achieved by the o-TG scheme are higher than those achieved by the ER scheme. The measurement of the total average received SNRs is important to observe whether the target total SNR, which is determined at the transmitter end during the resource allocation process, is achieved by having the total average received SNR equal to it. This total average received SNR is measured using Eq(3.36), which measures the average SNR of each code channel before adding the average SNR of each code channel to find the total. This is expected as the two-group resource allocation scheme is designed to maximize the utilization of the total constrained energy E_T to improve the total transmission data rate. Consequently, the total average received SNRs for this proposed scheme are expected to increase as more energy is allocated and used. For the ER scheme, there are two sets of total average received SNRs plotted. The first set is produced after running the ER scheme

whilst taking into consideration the ISI components to improve the energy allocation and ensure that the total average received SNR is not far from the target SNR at the transmitter end. The second set contains the total SNRs produced when the ISI components are not considered during the ER loading process. When the ISI components are not considered, the energy equation in Eq(3.14) is solved using a different covariance matrix $\mathbf{C}_i = \mathbf{Q}\mathbf{A}_i^2\mathbf{Q}^H + 2\sigma^2\mathbf{I}_{N+L-1}$, which excludes the ISI components from the previous and the next symbol periods. The results, as seen in this figure, show that the total average received SNRs reduce especially at high total input SNRs when the ISI components are ignored. Therefore, most of the resource allocation schemes implemented in this thesis consider the presence of ISI to closely emulate the practical scenario of radio networks in real life.

The total average received SNR when MMSE equalization is implemented at the receiver end is increased if ISI caused by multiple propagation paths of the symbols adjacent to the current symbol is considered during the energy calculation process. When ISI is considered, the covariance matrix will change, yielding a change in the calculated energy which reflects the presence of the ISI components. Furthermore, the MMSE despreading filter coefficient vector at the receiver end is also adjusted accordingly, as it depends on the covariance matrix, to better despread the received signal.

In Figure 3.7, the system throughput is plotted after running the o-TG and the ER scheme using the data rates aforementioned at zero gap values. As in the case of the total average received SNR plotted in the previous figure, the system throughput in Figure 3.7 is observed to be the highest when the o-TG scheme is implemented especially at high total input SNRs. The ER scheme, as expected, has produced a lower system throughput than that of the o-TG scheme. The third scheme, which is the equal energy and equal rate loading (EREE) scheme, is observed to produce the lowest system throughput. Based on this result, it is clear that the o-TG scheme produces the highest

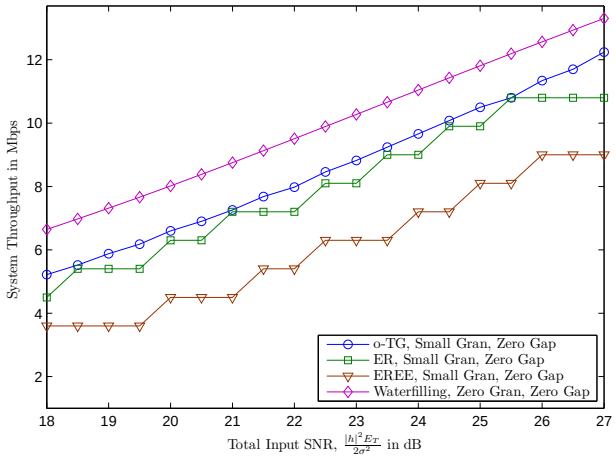

Figure 3.7: The system throughputs realizable when a range of total input SNRs were provided for the two group (o-TG), the equal rate margin adaptive loading (ER) and the equal energy and equal rate loading (EREE) schemes

system throughput as compared to the rest. Therefore, this proposed scheme is recommended for practical implementation such as in femtocells [71] to improve system performance.

In Figure 3.8, the total average received SNRs, γ_T, are plotted after running the two-group resource allocation (o-TG) and the equal rate margin adaptive loading (ER) scheme using the same set of data rates but with nonzero gap values. These data rates $\{b_p\}_{p=1}^{P}$ and the corresponding gap values $\{\Gamma_p\}_{p=1}^{P}$, which are given in Table 3.1, have been used to be allocated over $K = 15$ parallel code channels for a range of total input SNR. As in the previous figure, the total average received SNRs plotted in Figure 3.8 are observed to be the highest when the o-TG scheme is applied. When the ISI components are unconsidered during the energy allocation process, the total average received SNRs reduce for both schemes, as expected. This result further confirms the advantage of implementing the o-TG scheme over the existing ER scheme.

In the next subsection, a further investigation is carried out on these loading schemes by examining the bit granularity factor on system performance, which is measured in terms of the total average received SNR and the system

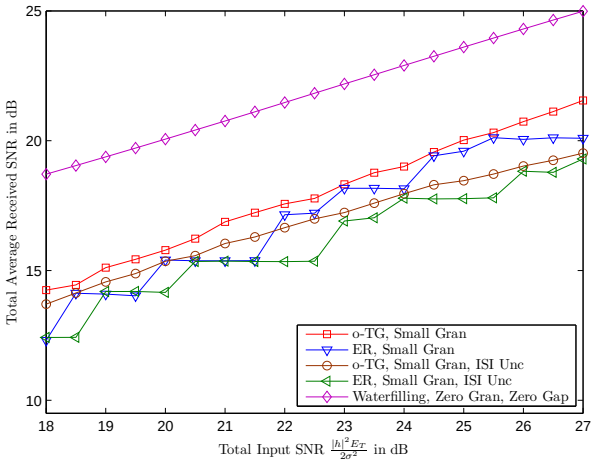

Figure 3.8: The total average received SNRs achieved at the input of the decoding units when a range of total input SNRs were fed into the two-group loading (o-TG) scheme and the equal rate margin adaptive loading (ER) scheme

throughput.

3.8.2 The Bit Granularity Factor in Performance of Resource Allocation Schemes

The bit granularity shows the closeness between two adjacent data rates from a set of data rates provided for a communication system. As the bit granularity decreases, the difference between two adjacent data rates decreases. As a result, the difference between the target minimum SNRs of these two adjacent data rates also reduces, rendering a lower difference in energy required to realize these data rates. Therefore, having a small bit granularity tend to yield a lower residual energy, which is upper bounded by the difference between the two energies required to realize the two adjacent data rates.

In this subsection, the bit granularity is varied when running the loading schemes to observe the impact of this parameter on these loading schemes. It will be shown that the two-group resource allocation scheme is capable of producing relatively higher total average received SNRs although a considerably larger bit granularity is used as compared to the equal rate margin adaptive

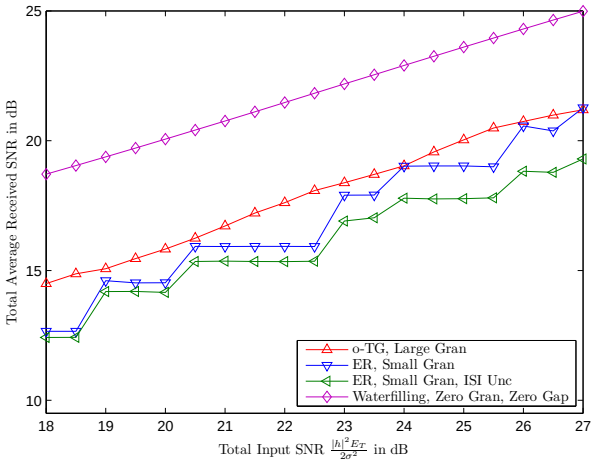

Figure 3.9: The total average received SNRs achieved at the input of the decoding units when a range of total input SNRs were fed into the two-group loading (o-TG) scheme with large bit granularities, along with the equal rate and equal SNR allocation (ER) scheme

loading scheme, which is implemented with a small bit granularity.

Figure 3.9 shows the total average received SNRs when the ER scheme was implemented with small bit granularities as opposed to the o-TG scheme, which was implemented at larger bit granularities. The small bit granularities for the ER scheme are mostly 0.25 bits per symbol, where the data rates are $b_p = 0.5$, $0.75, 1.0, 1.25, 1.5, 1.75, 2.0, 2.25, 2.5, 2.75, 3.0, 3.2, 3.75, 4.0$ bits per symbol and the bit granularities for the two-group loading are either 0.5 or 1 bits per symbol, where the corresponding data rates used are $b_p = 0.5, 1.0, 1.5, 2.0, 3.0, 4.0$ bits per symbol. The total average received SNRs, when the two-group loading was used, were plotted as the uppermost blue curve while the total average received SNRs when the ER scheme was implemented with smaller bit granularities were plotted as a green curve, just below the blue curve aforementioned.

As can be seen in Figure 3.9, the total average received SNRs achieved by the o-TG scheme, although run with larger bit granularities, are still higher than those of the ER scheme. This result demonstrates the effectiveness and practicality of implementing the proposed o-TG scheme, which may be operated with

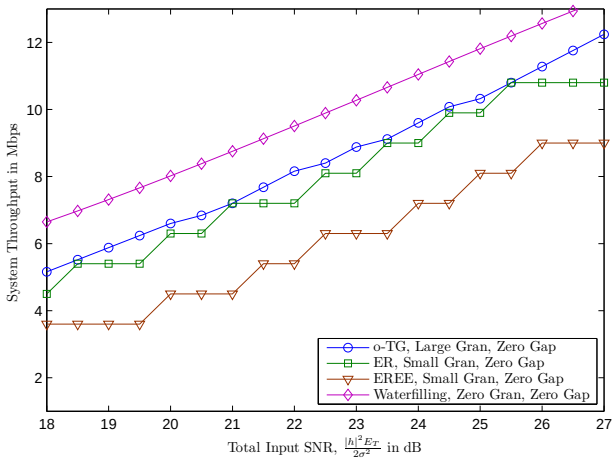

Figure 3.10: The system throughputs realizable when a range of total input SNRs were provided to the two group (o-TG), the equal rate and equal SNR (ER) and the equal rate and equal energy (EREE) schemes at zero gap values

the large bit granularities whilst maintaining considerably higher total average received SNRs as opposed to those of the ER scheme. Although the current high speed multi-code systems [1] offer a larger set of realizable discrete data rates with small bit granularities, implementing the o-TG scheme with larger bit granularities is a good alternative as the required computational complexity to operate the energy loading process with a larger number of discrete data rates tend to increase. This is because each energy iteration requires an inverse matrix operation when the equal rate margin adaptive loading scheme is used to minimize the energy required for loading the data rates. Therefore, the number of energy iterations increases, rendering increased computational complexity, when the number of discrete data rates which are provided increases to find the appropriate discrete data rate.

In Figure 3.10 and 3.11, the system throughputs in Mbps were plotted against the total input SNRs, where ISI was considered during the loading and MMSE equalization processes for the case of zero (Figure 3.10) and nonzero gap values (Figure 3.11) in dB. The two-group loading scheme was implemented with data rates in bits per symbol having large bit granularities, which have been

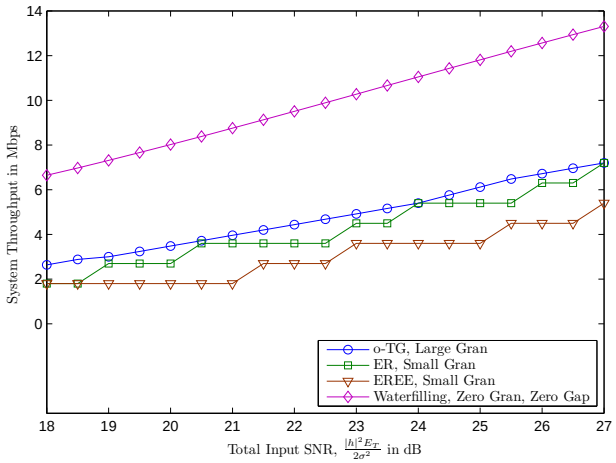

Figure 3.11: The system throughputs realizable when a range of total input SNRs were provided to the two group (o-TG), the equal rate and equal SNR (ER) and the equal rate and equal energy (EREE) schemes

mentioned earlier on. The ER scheme was implemented with small bit granularities, which have also been given earlier. Finally, the EREE scheme was also tested using the same set of data rates as in the case of the ER scheme. It can be seen that the system throughputs achieved by the two-group loading scheme are slightly higher than those of the equal rate and equal SNR loading scheme. The lowest system throughputs are realized when the equal rate and equal energy loading scheme is implemented.

In the next subsection, these loading schemes are further run to evaluate performance of the o-TG scheme when it is separately run using the two different channel ordering schemes.

3.8.3 Channel Ordering Schemes Evaluation

Channel ordering is implemented to ensure that the channel requiring the least amount of energy to realize the target SNRs is ordered as the first channel. This is implemented by calculating the required energy in each channel and order the channel according the energy value in ascending order. There are two channel ordering schemes presented in this chapter, as described in Section 3.6.1. Both

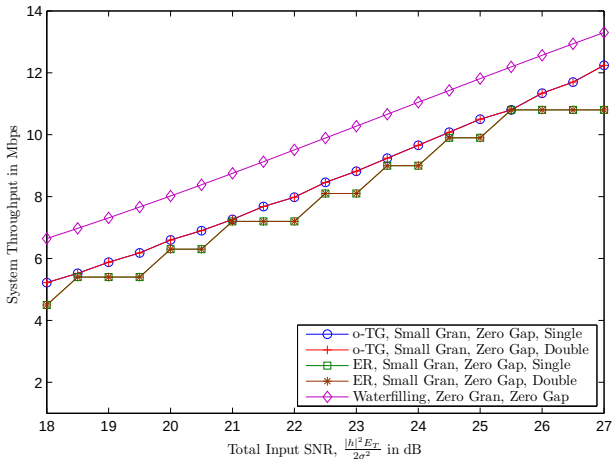

Figure 3.12: The system throughputs realizable when a range of total input SNRs were provided for the two group (o-TG) and the equal rate margin adaptive loading (ER) schemes using the single and double-loaded channel ordering schemes.

of these channel ordering schemes are separately run for the o-TG and the ER scheme to measure the achievable system throughputs over a range of total input SNRs. Figure 3.12 plot the measured system throughputs after running the loading schemes with the two channel ordering techniques. As clearly seen in this figure, both channel ordering methods are equally good as the system througputs produced after running these schemes are approximately equal to each other. Since the single-loaded channel ordering scheme requires no energy calculation iterations (see Section 3.6.1 on page 88) to determine the channel order, the single-loaded channel ordering scheme requires a lower computational load. Therefore this channel ordering scheme is more preferable to ensure that the practical implementation of the two-group resource allocation scheme is feasible.

The next subsection briefly investigates the computational complexity of the proposed SIC-based two-group resource allocation scheme and other loading schemes including the standard two-group resource allocation scheme, which is configured without SIC.

3.8.4 Computational Complexity Comparison

As the margin adaptive and the two-group loading schemes performs the iterative energy calculation process which requires matrix inversion operations, the required computational complexity by these loading schemes should be monitored. A matrix inversions requires a significantly high computational load, which might not be feasible to be implemented in practice. Assuming that each iterative energy calculation process requires a maximum of I_{max} iterations, each of which computes one matrix inversion to calculate the inverse covariance matrix $\mathbf{C}_i^{-1}$ to update the energy using Eq(3.14), the maximum numbers of matrix inversions required by these schemes are given in Table 3.2.

Loading scheme	Number of matrix inversions
ER	PI_{max}
o-TG	$(P+K-1)I_{max}$
o-TG, large granularities	$(P_{small}+K-1)\,I_{max}$
o-TG, large granularities and m estimate	$(P_{small}+2)\,I_{max}$

Table 3.2: Comparison of the number of matrix inversions

As can be seen in this table, there are two loading schemes of interest, which are the ER scheme (represents the equal rate margin adaptive loading scheme) and the two-group loading scheme. The maximum number of matrix inversions required by the ER scheme is PI_{max}, since there are at most P different data rates $\{b_p\}_{p=1}^{P}$ which should be examined before selecting the best one which maximizes the total data rate Kb_p for a given total energy E_T.

On the other hand, the maximum number of matrix inversions required by the two-group loading schme is $(P+K-1)I_{max}$ if the same number P of realizable data rates is assumed. The additional $(K-1)I_{max}$ matrix inversions are required at most to determine the number m of channels to be loaded with the higher data rate b_{p+1}. However, as can be seen in Figure 3.8, the number P of the data rates for the two-group loading scheme may be reduced significantly

to have larger bit granularities whilst keeping the largest data rate b_P and the smallest data rate b_p the same. The reduced number, which is denoted as P_{small} in the table, for the two-group loading scheme reduces the number of required matrix inversions whilst maintaining the realizable throughput, which is still relatively higher than that of the ER scheme. Figure 3.8 was produced with $P_{small} = 6$ for the two-group loading scheme while the number of data rates for the ER scheme is $P = 14$.

A further improvement is made by estimating the number m of the higher data rate b_{p+1} before determining the actual one using the iterative energy calculation process. This estimate is produced by from the single-loaded channel ordering scheme, which sorts the channels based on the energy differences, $E_k(b_{p+1}) - E_k(b_p)$, in an ascending order. Both of these energies have been determined when finding the lower data rate index p. The number m is estimated by examining the following condition $0 \leq \sum_{k=1}^{K} E_k(b_p) + \sum_{k=1}^{m} \left(E_{\kappa_k}(b_{p+1}) - E_{\kappa_k}(b_p)\right) < \sum_{k=1}^{K} E_k(b_{p+1})$, where $\left(E_{\kappa_k}(b_{p+1}) - E_{\kappa_k}(b_p)\right) < \left(E_{\kappa_{k+1}}(b_{p+1}) - E_{\kappa_{k+1}}(b_p)\right)$ and $\kappa_k \in \{1, 2, \ldots, K\}$ for $k = 1, 2, \ldots, K-1$. By estimating m, the average number of matrix inversions required to determine the actual desired m is $2I_{max}$, which renders a reduced total number $(P_{small} + 2)\, I_{max}$ of the matrix inversions. For $P_{small} = 6$, the maximum number of matrix inversions required by the two-group loading scheme is $8I_{max}$ while the maximum number of matrix inversions required by the ER scheme is $14I_{max}$.

Loading scheme	Matrix inversions	Discrete rates
ER	1400	14
o-TG	2800	196
o-TG, large granularities	2000	76
o-TG, large granularities and m estimate	800	76

Table 3.3: The number of matrix inversions and realizable discrete total data rates for $P = 14$, $P_{small} = 6$, $K = 15$ and $I_{max} = 100$

Table 3.3 shows the maximum number of matrix inversions and the total number

of total discrete data rates realizable when the ER scheme and the two-group loading schemes are run. The maximum number of matrix inversions is calculated using the formula tabulated in Table 3.3 and the total number of discrete data rates is determined as follows: P for the ER scheme and $K(P-1)+1$ for the two-group loading scheme. The total number of discrete data rates which can be realized by using the ER scheme is equal to the total number P of the data rates $\{b_p\}_{p=1}^{P}$ because every channel is allocated an equal data rate b_p. As for the two-group resource allocation scheme, there are $K(P-1)+1$ distinct total discrete data rates because there are $K-1$ different total discrete data rates which may be realized between any two adjacent data rates, b_p and b_{p+1}. Therefore, there are $K(P-1)+1$ distinct discrete rates realizable including the last and the highest total data rate, Kb_P bits per symbol. As can be seen in Table 3.3, the proposed two-group loading scheme, when implemented with large bit granularities and provided with the estimate of m, produces a total number of discrete data rates more than five times the total number of discrete data rates produced by the ER scheme for the given $P_{small} = 6$, $P = 14$, $K = 15$ and $I_{max} = 100$. Furthermore the maximum number of matrix inversions that might be used is about half of that of the ER scheme.

3.9 Conclusion

In this chapter, a two-group resource allocation scheme is described as an alternative loading scheme for HSDPA systems [1], which have been widely implemented and studied. When this two-group algorithm is run with discrete data rates of large bit granularities, the resultant system throughput is observed to have achieved a gain of 5 dB over the existing equal rate and equal energy loading scheme, which is implemented with smaller bit granularities, for the case of nonzero gap values. The achievable system throughput is further improved when the CPP coding scheme is incorporated into the proposed system

model. It is also observed that channel ordering schemes may be implemented in different methods in order to enhance the system throughput.

Therefore, the proposed two-group resource allocation scheme offers an alternative approach for the existing HSDPA systems. If the number of matrix inversions required during the energy calculation process is reduced, this proposed two-group algorithm may also be implemented for multiple-input multiple-output (MIMO) HSDPA systems in order to further enhance the system throughput whilst keeping the computational complexity relatively low.

CHAPTER 4

Simplified Loading Schemes with Interference Cancellation and Coded Packet Transmission

4.1 Introduction

The advent of high-speed multi-channel communication schemes has been substantially improving the system throughput. Various techniques have been developed to efficiently allocate the radio resources over the parallel channels in order to optimize the total transmission data rate whilst fully utilizing the limited resources. Although advanced transmission schemes have been incorporated into the current high-speed networks, such as multiple-input multiple-output (MIMO) HSDPA [1], practical measurement-based performance evaluation carried out in [2] has shown that the achievable data throughput in practical systems is still far lower than the theoretical upper bound. Therefore, optimization methods are still one of the major interests in the research community as well as the industry to practically improve and optimize system performance.

Various optimization techniques for multi-channel transmission systems have been investigated and described in the previous chapter. From the many schemes discussed in Chapter 3, there are two major types of optimization techniques discussed; the margin adaptive optimization method and the rate adaptive optimization method. The margin adaptive optimization scheme is useful for fully utilizing the SNR in each of the parallel channels. The rate adaptive optimization method is mainly developed to maximize the total transmission by maximally utilizing the constrained radio resources especially the total energy. These optimization techniques, however, may require prohibitively high computational complexity to search for the optimal data rate and energy allocation from P available data rates to be allocated over the K parallel channels. In order to reduce the computational load required for finding the optimal or near-optimal data rate allocation, the two-group resource allocation scheme has been proposed in Chapter 2 to consider only two adjacent discrete data rates for the parallel channels whilst attempting to maximize the total transmission data rate. Although the number of searches for the appropriate and near optimal data rate allocation has been significantly reduced, this resource allocation method requires a relatively significant number of iterations to determine the energy as well as the data rate that should be loaded to each channel. Computationally efficient optimization or loading schemes which maintain relatively high performance are preferable for practical systems such as femtocells [71].

In this chapter, two main improvements have been developed to improve the computational complexity of the two-group resource allocation scheme and provide an alternative method which requires a relatively low computational complexity. The first approach is to incorporate a successive interference cancellation (SIC) scheme into the energy calculation method of the two-group loading scheme in order to remove inverse matrix operations and significantly reduce the number of iterations to complete the energy calculation process. The second approach benefits from the adaptive coded parity packet transmission, which is

mainly developed from the CPP scheme presented in the first chapter, to load a fixed and equal amount of energy to each channel and maintain a relatively low gap value.

4.2 Related Work

Multi-code transmission models, although designed to significantly improve the total transmission data rates, should be developed with relatively low computational complexity especially when these models are proposed for the current wireless broadband technologies such as femtocells [78, 79], which are limited in signal processing capability. Recent standards in the Third Generation Partnership Project, such as in [1], provide more total discrete data rates for the users with different SNRs as compared to the previous standards, such as in [39]. Although more realizable discrete data rates are offered, these recent developments focus one equal rate loading, which allocates an equal data rate to each of the code channels. Furthermore, determining the right data rate to be allocated from a larger number of realizable data rates may render a higher computational complexity.

Instead of allocating an equal data rate to each code channel from a large number of realizable data rates, two adjacent data rates from a smaller number of realizable data rates may be realized over two groups of code channels [29, 74, 80–82] to provide a larger number of realizable total discrete data rates, which are the sum of data rates loaded to all code channels. The smaller set of data rates may be formed from a number of data rates with relatively large bit granularities, or the differences between two adjacent data rates. It has been shown in the previous chapter that this smaller set of data rates with relatively large bit granularities may be provided to the two-group resource allocation scheme to realize a larger number of total discrete data rates with a relatively lower computational complexity.

However, the iterative energy calculation method employed in the two-group resource allocation scheme to calculate the energy that should be allocated to each code channel such that the target minimum SNR is met based on the MMSE approach [83] requires a relatively high computational complexity due the matrix inversions involved in the iterations. Although this MMSE-based energy calculation approach is effective in reducing the noise and interference components, the incurred computational load is relatively high and may not be feasible for practical systems with limited processing capability such as femto-cells.

Another widely adopted technique for removing interference components is SIC [84–86], which removes interference successively from one channel to another. Various SIC-based energy calculation methods have been proposed to load the energy based on different target bit error rates (BERs) on the channels, such as in [87–90], to name a few. Unlike these techniques, an SIC-based energy calculation which sets only two different target SNRs for transmitting the data over the employed channels is presented in this chapter. Another SIC-based energy allocation scheme was proposed for multi-code CDMA systems in [91]. This scheme, which was developed based on a V-BLAST system [92], assumed that the cross-correlations between different receiver signature sequences were approximately equal to each other. This approximation might be the case in practice especially when multi-path propagation environment due to the frequency selective channels is considered.

In this chapter, an SIC-based two-group resource allocation scheme, which is developed based on a computationally-efficient SIC-based energy calculation method presented in [93], is proposed to further enhance the transmission total data rate achievable in the two-group resource allocation scheme [29, 74, 80–82] with a relatively lower computational complexity as no matrix inversions are required. To consider and remove the ISI and MAI components, the proposed CPP scheme [29,30], which has been presented in Chapter 2, is also implemented

to detect and remove the symbols received in the previous and the next symbol period using a packet-based coding technique. Another loading scheme is also developed based on the CPP scheme. Instead of allocating two groups of data rates to the code channels, an equal data rate is loaded to each code channel with an equal energy.

The rest of this chapter is organized as follows. The research problem addressed in this chapter is described in the next section. The system model of the modified two-group resource allocation scheme integrated with the SIC scheme is described in Section 4.4. Then the proposed SIC-based energy calculation method is presented in Section 4.4.1 before the corresponding receiver structure with SIC is presented in Section 4.4.4. In Section 4.5, a different loading approach which applies the CPP scheme and fixed energy loading is described. Finally the results obtained from running these proposed schemes are given and discussed in Section 4.5.4. Section 4.6 concludes this chapter.

4.3 Problem Formulation

In this chapter, the main objective is to improve the total transmission data rate whilst minimizing the amount of energy consumed by using an SIC-based energy calculation method. It will be shown that by applying SIC, the total SNR, hence the energy, required to transmit information data for a given total constraint energy can be reduced.

In the previous chapter, a two-group resource allocation scheme is proposed with the CPP scheme to improve the total transmission data rate and provide a greater number or realizable total discrete data rates with a relatively lower computational complexity compared to some existing schemes. In this chapter, the following research problems are formulated and addressed:

1. *Interference cancellation for energy minimization.* As the two-group re-

source allocation scheme is proposed for multi-code CDMA systems at the presence of MAI and ISI, the energy required $E_k(y_k)$ to achieve the target minimum SNR γ_k^* tend to increase as the interference components increase. Therefore, this chapter modifies the two-group resource allocation scheme by implementing an SIC scheme to remove the interference components. By assuming that the channel side information including the channel impulse response and the noise variance are known to the transmitter as well as the receiver, an SIC-based two-group resource allocation scheme is designed in this chapter to remove interference and hence minimize the energy $E_k(y_k)$ required to achieve the target minimum SNR γ_k^* for a given data rate y_k in bits per symbol.

2. *Total transmission data rate enhancement.* When the energy $E_k(y_k)$ required to transmit a data rate y_k in bits per symbol is minimized, the total transmission data rate R_T may be maximized for a given total energy E_T since more energy is available to realize higher data rates.

3. *Computational complexity reduction.* The two-group resource allocation scheme presented in the previous chapter iteratively calculates the energy as $E_{k,i+1}(y_k) = \frac{\gamma_k^*(y_k)}{1+\gamma_k^*(y_k)} \left(\vec{q}_k^H \mathbf{C}_i^{-1} \vec{q}_k\right)^{-1}$, using Eq(3.14) until the iteration reaches the maximum iteration $i = I_{max}$ or the energy is settled or converged, $E_k(y_k) = E_{k,i+1}(y_k) \approx E_{k,i}(y_k)$. It is clear from this energy equation that matrix inversions, which are computationally expensive, are needed. In addition to this, the iterative process required to update the energy and the covariance matrix which depends on each other further increases the number of matrix inversions, and hence increases the overall computational complexity. Therefore, a computationally efficient SIC-based energy calculation method is developed which requires no matrix inversions to calculate the energy.

4.4 An SIC-based Two-group Resource Allocation Scheme

Before describing the proposed SIC-based two-group resource allocation scheme, the system model in which the SIC scheme is implemented along with the two-group resource allocation scheme and the CPP scheme is described in the next subsection. It is assumed that the SIC operation at the receiver end has no propagation error and the channel side information is known to both the transmitter and the receiver.

4.4.1 System Model

Figure 4.1 shows the model of the proposed integrated system which incorporates the SIC, the CPP and the two-group resource allocation schemes. In general, this system model is very similar to the model shown in Figure 3.5 except that the two-group resource allocation scheme is now implemented with SIC. The binary data intended for transmission are encoded and mapped to an $(N^{(x)} \times 1)$-dimensional M-array symbol vector $\vec{x}_k = \begin{bmatrix} x_k(1) & \dots & x_k(N^{(x)}) \end{bmatrix}^T$ over each code channel k according to the data rate b_{p_k} which have been determined from the resource allocation process. This symbol vector is then allocated the energy calculated from the resource allocation process to yield $\sqrt{E_k}\vec{x}_k$ before it is spread with the corresponding ordered signature sequence $\vec{s}_k = \mathbf{S}\vec{\psi}_k^T$, where $\vec{\psi}_k$ is the k-th row vector of the channel ordering matrix $\mathbf{P}_{ord}$ having all elements equal to zero except for the κ_k-th element which is equal to unity. After the spreading and modulation process, the transmission begins. Assuming that multi-path propagation exists due to the frequency selective channels, which cause ISI and MAI, the received signal is represented as an $\big((N + 2\alpha_N) \times N^{(x)}\big)$-dimensional matrix $\mathbf{R} = \mathbf{Q}_e\tilde{\mathbf{A}}\tilde{\mathbf{X}} + \mathbf{N}$, as given in Eq(3.4). This received signal matrix is then despread with the MMSE despreading fil-

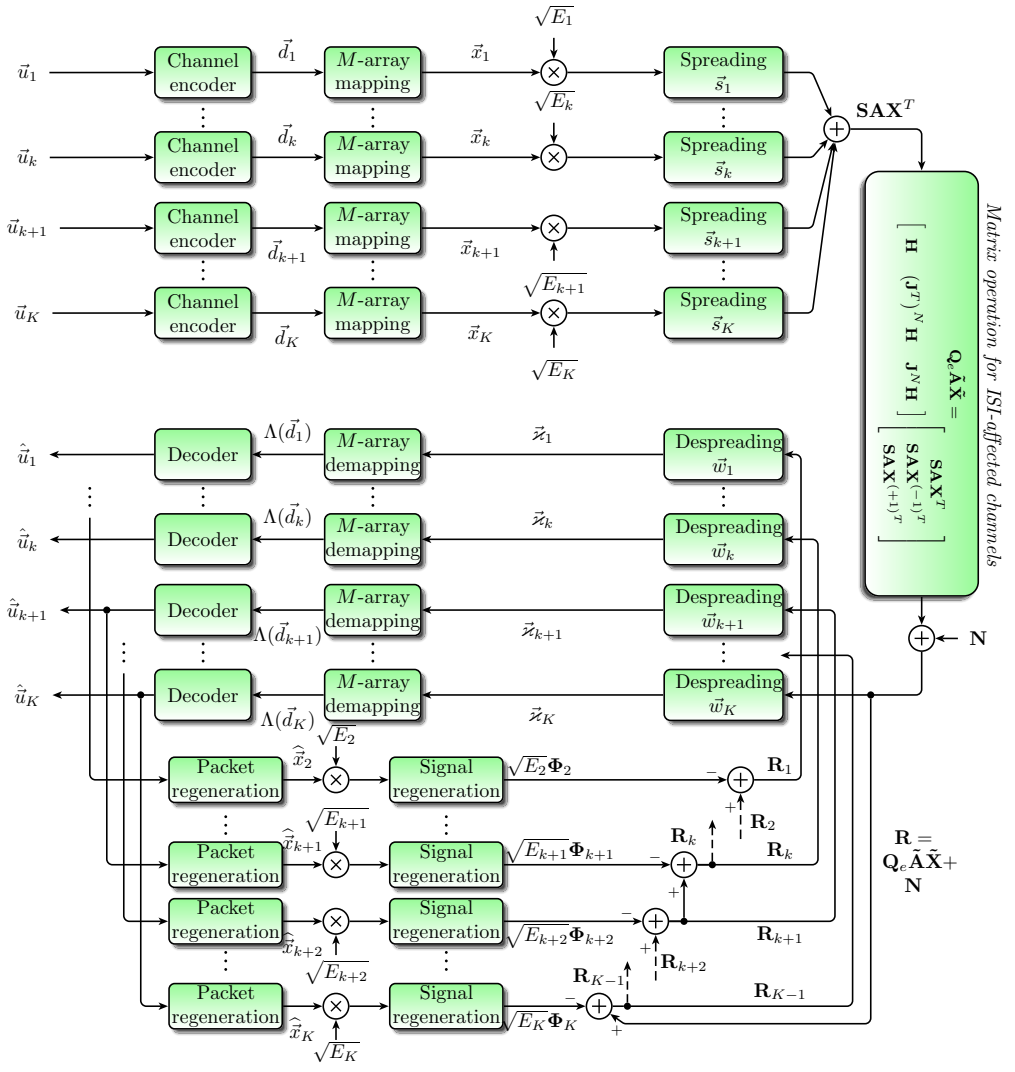

Figure 4.1: The system model

ter coefficient vector $\vec{w}_K$ at channel K, which is calculated using the SIC-based energy calculation method that will be described in the next subsection, assuming that the channel side information including the channel impulse response and the noise variance are known to the receiver. Consequently, the despread signal $\vec{\varkappa}_K^T = \vec{w}_K^H \mathbf{R}_K$ is then passed on to the demodulator to produce a soft-bit information vector $\Lambda(\widehat{\vec{d}_K})$ before it is fed into the decoder to produce $\widehat{\vec{u}}_K$ as the estimate of the actual transmitted data vector $\vec{u}_K$.

This decoded data vector is then reprocessed to produce a regenerated packet $\widehat{\vec{x}}_K$, before the energy is reallocated and the signal matrix, containing the regenerated packet and the corresponding ISI components, is regenerated as $\sqrt{E_K(y_K)}\mathbf{\Phi}_K$, where $\mathbf{\Phi}_K = \vec{q}_K \widehat{\vec{x}}_K^T + \vec{q}_{K,1} \widehat{\vec{x}}_K^{(-1)^T} + \vec{q}_{K,2} \widehat{\vec{x}}_K^{(+1)^T}$, while both $\widehat{\vec{x}}_K^{(-1)}$ and $\widehat{\vec{x}}_K^{(+1)}$ refer to the packet of symbols received in the previous and the next symbol period respectively. The reduced received signal matrix $\mathbf{R}_{K-1}$ is then despread for channel $K-1$ and the whole process aforementioned is repeated for this channel and the rest of the channels until all received signals for each channel have been detected or decoded.

Hence, the similar process of regenerating the decoded packets $\widehat{\vec{u}}_k$ to produce $\widehat{\vec{x}}_k$ using the CPP scheme applies for all $k = K, K-1, \ldots, 2$. These regenerated packets are defined as follows

$$\widehat{\vec{x}}_k = \begin{bmatrix} \widehat{x}_k(1) & \widehat{x}_k(2) & \ldots & \widehat{x}_k(N^{(x)}) \end{bmatrix}^T, \tag{4.1}$$

$$\widehat{\vec{x}}_k^{(-1)} = \mathbf{J}_{N^{(x)}} \widehat{\vec{x}}_k, \tag{4.2}$$

$$\widehat{\vec{x}}_k^{(+1)} = \mathbf{J}_{N^{(x)}}^T \widehat{\vec{x}}_k, \tag{4.3}$$

for $k = K, K-1, \ldots, 2$ where $\widehat{\vec{x}}_k$ corresponds to the regenerated packet of symbols received in the current symbol period from the discrete symbol period 1 to the discrete symbol period $N^{(x)}$. Both $\widehat{\vec{x}}_k^{(-1)} = \begin{bmatrix} 0 & \widehat{x}_k(1) & \ldots & \widehat{x}_k(N^{(x)}-1) \end{bmatrix}^T$ and $\widehat{\vec{x}}_k^{(-1)} = \begin{bmatrix} \widehat{x}_k(2) & \ldots & \widehat{x}_k(N^{(x)}) & 0 \end{bmatrix}^T$ are the regenerated packets corre-

spond to the symbols received in the previous and the next symbol periods.

With the knowledge of these regenerated packets, the regenerated signal matrix $\mathbf{\Phi}_k$, which is going to be removed from the current received matrix $\mathbf{R}_k$, is written as

$$\mathbf{\Phi}_k = \vec{q}_k \widehat{\vec{x}}_k^T + \vec{q}_{k,1} \widehat{\vec{x}}_k^{(-1)^T} + \vec{q}_{k,2} \widehat{\vec{x}}_k^{(+1)^T}, \tag{4.4}$$

for $k = K, K-1, \ldots, 2$. As the signature sequence $\vec{s}_k = \mathbf{S}\vec{\psi}_k^T$ has been ordered before the packet transmission begins, the receiver signature sequence $\vec{q}_k$ is also ordered as follows:

$$\vec{q}_k = \tilde{\mathbf{H}}\mathbf{S}\vec{\psi}_k^T, \tag{4.5}$$

for $k = 1, 2, \ldots, K$, where the channel ordering matrix $\mathbf{P}_{ord}$ is assumed to be known at the receiver end.

If the signal $\mathbf{\Phi}_{k+1}$ has been regenerated to be removed from the previously reduced received signal matrix $\mathbf{R}_{k+1}$ at channel $k+1$ and the energy $E_{k+1}(y_{k+1})$ has been recalculated using the SIC-based energy calculation method described in Section 4.4.1, the reduced received signal matrix $\mathbf{R}_k$ for channel k may be expressed as follows:

$$\mathbf{R}_k = \mathbf{R}_{k+1} - \sqrt{E_{k+1}(y_{k+1})}\mathbf{\Phi}_{k+1}, \tag{4.6}$$

for $k = K-1, K-2, \ldots, 1$, where $\mathbf{R}_K = \mathbf{R} = \mathbf{Q}_e\tilde{\mathbf{A}}\tilde{\mathbf{X}} + \mathbf{N}$ as the original received signal matrix before any interference cancellation. It is assumed that the channel side information is known to the receiver end in order to calculate the energy.

After knowing the reduced signal matrix $\mathbf{R}_k$, the despreading process then takes place using the MMSE despreading filter coefficient vector $\vec{w}_k$ to yield a despread symbol vector of

$$\vec{x}_k^T = \vec{w}_k^H \mathbf{R}_k. \tag{4.7}$$

The system model described above is essential to give an overwiew of the SIC scheme before presenting the proposed SIC-based energy calculation method. In order to describe this energy calculation method, the covariance matrices which are required for this calculation method are defined. Then, the new MMSE despreading filter coefficient vector $\vec{w}_k$ as well as the new SNR equation $\gamma_k(y_k)$, which are closely related and required for the energy equation, are given before the energy equation is expressed. With the energy equation, the SIC-based energy calculation is then presented in the following subsection.

Covariance Matrices

Without interference cancellation, the SNR in channel k can be written as $\gamma_k(y_k) = \frac{E_k(y_k)\vec{q}_k^H \mathbf{C}^{-1}\vec{q}_k}{1-E_k(y_k)\vec{q}_k^H \mathbf{C}^{-1}\vec{q}_k}$, which is given in Eq(3.8), where the covariance matrix $\mathbf{C}$ is given in Eq(3.7) and $\vec{q}_k = \tilde{\mathbf{H}}\vec{s}_k$. This covariance matrix is used in the SNR equation aforementioned to find the SNR for all channels.

When SIC is in place, the covariance matrix differs from one channel to another. Therefore it is convenient to introduce a new covariance matrix which reflects the change in its value due to interference cancellation in each channel. Assuming that the SIC operation is perfect with no propagation error and is performed from the channel $K-1$ to channel 1, the covariance matrix $\mathbf{C}_k$ for channel k can be written as follows [93, 94]:

$$\begin{aligned} \mathbf{C}_k &= 2\sigma^2\mathbf{I}_{N+2\alpha} + \sum_{j=1}^{k} E_j(y_j)\left(\vec{q}_j\vec{q}_j^H + \vec{q}_{j,1}\vec{q}_{j,1}^H + \vec{q}_{j,2}\vec{q}_{j,2}^H\right), \\ &= \mathbf{C}_{k-1} + E_k(y_k)\left(\vec{q}_k\vec{q}_k^H + \vec{q}_{k,1}\vec{q}_{k,1}^H + \vec{q}_{k,2}\vec{q}_{k,2}^H\right). \end{aligned} \tag{4.8}$$

for $k = 1, 2, \ldots, K$ where $\mathbf{C}_0 = 2\sigma^2\mathbf{I}_{N+2\alpha}$, $\vec{q}_{k,1} = (\mathbf{J}^T)^N\vec{q}_k$ and $\vec{q}_{k,2} = \mathbf{J}^N\vec{q}_k$. Both $\vec{q}_{k,1}$ and $\vec{q}_{k,2}$ are the receiver signature sequences corresponding to the previous and the next symbol period respectively. This covariance matrix is then used to compute the MMSE despreading filter coefficient vector $\vec{w}_k$ from

the original equation given in Eq(3.6). Since the effective covariance matrix $\mathbf{D}_{2,k}$ for each channel k excludes the components contributed by the energy and the receiver signature sequence of channel k, this covariance matrix $\mathbf{D}_{2,k}$ may be represented as

$$\begin{aligned} \mathbf{D}_{2,k} &= \mathbf{C}_{k-1} + E_k(y_k)\left(\vec{q}_{k,1}\vec{q}_{k,1}^{H} + \vec{q}_{k,2}\vec{q}_{k,2}^{H}\right), \\ &= \mathbf{C}_k - E_k(y_k)\vec{q}_k\vec{q}_k^{H}. \end{aligned} \tag{4.9}$$

A New MMSE Despreading Filter Coefficient Vector

By computing the inverse of the effective covariance matrix $\mathbf{C}_{2,k}^{-1}$ given in Eq(4.9) using the matrix inversion lemma as

$$\begin{aligned} \mathbf{D}_{2,k}^{-1} &= \mathbf{C}_k^{-1} - \mathbf{C}_k^{-1}\vec{q}_k\left(E_k(y_k)^{-1} + \vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k\right)^{-1}\vec{q}_k^{H}\mathbf{C}_k^{-1}, \\ &= \mathbf{C}_k^{-1} - \frac{E_k(y_k)\mathbf{C}_k^{-1}\vec{q}_k\vec{q}_k^{H}\mathbf{C}_k^{-1}}{1 + E_k(y_k)\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k}, \end{aligned} \tag{4.10}$$

the MMSE despreading filter coefficient vector $\vec{w}_k$ is then derived as follows

$$\vec{w}_k = \frac{\mathbf{D}_{2,k}^{-1}\vec{q}_k}{\vec{q}_k^{H}\mathbf{D}_{2,k}^{-1}\vec{q}_k},$$

$$= \frac{\left(\mathbf{C}_k - E_k(y_k)\vec{q}_k\vec{q}_k^{H}\right)^{-1}\vec{q}_k}{\vec{q}_k^{H}\left(\mathbf{C}_k - E_k(y_k)\vec{q}_k\vec{q}_k^{H}\right)^{-1}\vec{q}_k}, \tag{4.11}$$

$$= \frac{\mathbf{C}_k^{-1}\vec{q}_k - \dfrac{E_k(y_k)\mathbf{C}_k^{-1}\vec{q}_k\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k}{1 + E_k(y_k)\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k}}{\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k - \dfrac{E_k(y_k)\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k}{1 + E_k(y_k)\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k}}, \tag{4.12}$$

$$= \frac{\mathbf{C}_k^{-1}\vec{q}_k}{\vec{q}_k^{H}\mathbf{C}_k^{-1}\vec{q}_k}. \tag{4.13}$$

A New SNR Equation

The despreading filter coefficient vector $\vec{w}_k$ is essential to ensure that the target SNR $\gamma_k^*(y_k)$ can be achieved at the receiver end. With the new definition of the

covariance matrix given in Eq(4.8), the corresponding SNR equation can now be determined as follows:

$$\begin{aligned}\gamma_k(y_k) &= \frac{E_k(y_k)|\vec{w}_k^H\vec{q}_k|^2}{\vec{w}_k^H\mathbf{C}_k\vec{w}_k - E_k(y_k)|\vec{w}_k^H\vec{q}_k|^2}, &(4.14)\\ &= \frac{E_k(y_k)\vec{q}_k^H\mathbf{C}_k^{-1}\vec{q}_k}{1 - E_k(y_k)\vec{q}_k^H\mathbf{C}_k^{-1}\vec{q}_k}. &(4.15)\end{aligned}$$

A New Energy Calculation Method

Using the expression in Eq(4.11), the MMSE despreading filter coefficient vector may be expressed as,

$$\begin{aligned}\vec{w}_k &= \frac{\left(\mathbf{C}_k - E_k(y_k)\vec{q}_k\vec{q}_k^H\right)^{-1}\vec{q}_k}{\vec{q}_k^H\left(\mathbf{C}_k - E_k(y_k)\vec{q}_k\vec{q}_k^H\right)^{-1}\vec{q}_k},\\ &= \frac{\left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}\vec{q}_k}{\vec{q}_k^H\left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}\vec{q}_k}, &(4.16)\end{aligned}$$

which is written as a function of the covariance matrix $\mathbf{C}_{k-1}$ for channel $k-1$. With this new expression of the MMSE despreading filter coefficient vector $\vec{w}_k$, the SNR equation may also be expressed, from Eq(4.14), may also be written as follows

$$\begin{aligned}\gamma_k(y_k) &= \frac{E_k(y_k)}{\vec{w}_k^H\mathbf{C}_k\vec{w}_k - E_k(y_k)},\\ &= \frac{E_k(y_k)}{\vec{w}_k^H\left(\mathbf{C}_k - E_k(y_k)\vec{q}_k\vec{q}_k^H\right)\vec{w}_k},\\ &= \frac{E_k(y_k)}{\vec{w}_k^H\left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)\vec{w}_k},\\ &= \frac{E_k(y_k)}{\vec{q}_k^H\left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}\vec{q}_k}. &(4.17)\end{aligned}$$

The inverse matrix, $\left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}$, may be solved

using the matrix inversion lemma before a new energy calculation method is derived from Eq(4.17) as follows [93]:

$$E_{k,i}(y_k) = \frac{\gamma_k^*(y_k)}{\xi - E_{k,i-1}\left(\frac{|\xi_3|^2}{1+E_{k,i-1}\xi_1} + \frac{|\xi_4|^2}{1+E_{k,i-1}\xi_2}\right)}, \tag{4.18}$$

for a target minimum SNR of $\gamma_k^*(y_k) = \Gamma(2^{y_k} - 1)$,where the new vectors and constants are defined as follows [93]:

- *Distance vectors*:

$$\vec{d} = \mathbf{C}_{k-1}^{-1}\vec{q}_k, \tag{4.19}$$

$$\vec{d_1} = \mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}, \tag{4.20}$$

$$\vec{d_2} = \mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}. \tag{4.21}$$

- *Weighting factors*:

$$\xi = \vec{d}^H\vec{q}_k, \tag{4.22}$$

$$\xi_1 = \vec{d_1}^H\vec{q}_{k,1}, \tag{4.23}$$

$$\xi_2 = \vec{d_2}^H\vec{q}_{k,2}, \tag{4.24}$$

$$\xi_3 = \vec{d}^H\vec{q}_{k,1}, \tag{4.25}$$

$$\xi_4 = \vec{d}^H\vec{q}_{k,2}. \tag{4.26}$$

A more detailed derivation for the energy equation in Eq(4.18) is given in Appendix B, which is produced from the method presented in [93]. The advantage of using the new energy calculation method given in Eq(4.18) is its independency on the standard inverse matrix operation, which is computationally expensive. The energy calculation iteration only uses a number of constants and vectors given above. The only variable that needs to be changed and updated in calculating the energy $E_{k,i}(y_k)$ is only $E_{k,i-1}(y_k)$, which is itself. The inverse

covariance matrix $\mathbf{C}_{k-1}^{-1}$ has already been determined in the previous channel, ie. channel $k-1$, and hence needs no adjustment. Therefore, it is expected that the maximum number I_{max} of iterations required to calculate the energy using this method is relatively low. Based on the simulation results obtained when testing this energy calculation method, the maximum number I_{max} in order to obtain $E_{k,I_{max}}(y_k) - E_{k,I_{max}-1}(y_k) \leq 1 \times 10^{-4}$ is roughly 10.

After knowing the energy $E_k(y_k) = E_{k,I_{max}}(y_k)$ at channel k, the inverse covariance matrix $\mathbf{C}_k^{-1}$ needs to be determined in order to calculate the MMSE despreading filter coefficient vector $\vec{w}_k = \frac{\mathbf{C}_k^{-1}\vec{q}_k}{\vec{q}_k^H \mathbf{C}_k^{-1}\vec{q}_k}$. This inverse covariance matrix is also necessary for calculating the energy $E_{k+1}(y_{k+1})$ in channel $k+1$. In order to calculate this inverse covariance matrix, the following matrix weighting factors are defined:

$$\zeta = \frac{E_k(y_k)}{1+\gamma_k^*(y_k)}, \tag{4.27}$$

$$\zeta_1 = \frac{E_k(y_k)}{1+E_k(y_k)\xi_1}, \tag{4.28}$$

$$\zeta_2 = \frac{E_k(y_k)}{1+E_k(y_k)\xi_2}. \tag{4.29}$$

With these matrix weighting factors as well as the distance vectors and the weighting factors given in Eq(4.19), Eq(4.20), Eq(4.21), Eq(4.22), Eq(4.23), Eq(4.24), Eq(4.25), Eq(4.26) respectively, the inverse covariance matrix $\mathbf{C}_k^{-1}$ is calculated in a simplified manner as follows:

$$\begin{aligned}\mathbf{C}_k^{-1} = {} & \mathbf{C}_{k-1}^{-1} - \zeta\vec{d}\vec{d}^H - \left(\zeta_1 + \zeta\zeta_1^2\left|\xi_3\right|^2\right)\vec{d}_1\vec{d}_1^H - \left(\zeta_2 + \zeta\zeta_2^2\left|\xi_4\right|^2\right)\vec{d}_2\vec{d}_2^H \\ & + \zeta\zeta_1\left(\xi_3\vec{d}\vec{d}_1^H + \xi_3^*\left(\vec{d}\vec{d}_1^H\right)^H\right) + \zeta\zeta_2\left(\xi_4\vec{d}\vec{d}_2^H + \xi_4^*\left(\vec{d}\vec{d}_2^H\right)^H\right) \\ & - \zeta\zeta_1\zeta_2\left(\xi_3\xi_4^*\vec{d}_2\vec{d}_1^H + (\xi_3\xi_4^*)^*\left(\vec{d}_2\vec{d}_1^H\right)^H\right),\end{aligned} \tag{4.30}$$

The proof for this new inverse covariance matrix equation is provided in Appendix C, which is produced from the derivations given in [93]. After calculating

$\mathbf{C}_k^{-1}$, the MMSE despreading filter coefficient vector $\vec{w}_k$ may now be calculated and the energy calculation process continues for channel $k+1$ until all energies and MMSE despreading filter coefficient vectors have been calculated for all channels.

The next section summarizes this SIC-based energy calculation method.

An SIC-based Energy Calculation Algorithm

With the new energy formula, the step-by-step procedures to calculate the energy may be summarized as follows:

1. Calculate the initial inverse matrix, $\mathbf{C}_0^{-1} = \frac{1}{2\sigma^2}\mathbf{I}_{N+2\alpha_N}$, and the receiver signature sequences, $\vec{q}_k = \tilde{\mathbf{H}}\vec{s}_k$, $\vec{q}_{k,1} = \left(\mathbf{J}^T\right)^N \vec{q}_k$ and $\vec{q}_{k,2} = \mathbf{J}^N\vec{q}_k$ for $k = 1, 2, \ldots, K$.

2. Set $E_{k,1} = \frac{E_T}{K}$ for $k = 1, 2, \ldots, K$.

3. Set $k = 1$.

4. Calculate the distance vectors $\vec{d} = \mathbf{C}_{k-1}^{-1}\vec{q}_k$, $\vec{d}_1 = \mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}$, $\vec{d}_2 = \mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}$ and the weighting functions $\xi = \vec{d}^H\vec{q}_k$, $\xi_1 = \vec{d}_1^H\vec{q}_{k,1}$, $\xi_2 = \vec{d}_2^H\vec{q}_{k,2}$, $\xi_3 = \vec{d}^H\vec{q}_{k,1}$, $\xi_4 = \vec{d}^H\vec{q}_{k,2}$.

5. Set the target minimum SNR as $\gamma_k(y_k) = \gamma_k^*(y_k) = \Gamma_p(2^{y_k} - 1)$.

6. Calculate the energy iteratively for $i = 1, 2, \ldots, I_{max}$ using Eq(4.18)

7. Calculate the matrix weighting factors using Eq(4.27), Eq(4.28) and Eq(4.29).

8. Calculate the inverse covariance matrix $\mathbf{C}_k^{-1}$ using Eq(4.30).

9. Calculate the MMSE despreading filter coefficient vector $\vec{w}_k$ using Eq(4.13).

$$\vec{w}_k = \frac{\mathbf{C}_k^{-1}\vec{q}_k}{\vec{q}_k^H\mathbf{C}_k^{-1}\vec{q}_k},$$

10. With $\vec{w}_k$, the SNR is calculated using Eq(4.15) as follows

$$\gamma_k(y_k) = \frac{E_k(y_k)\left(\vec{q}_k^H \mathbf{C}_k^{-1} \vec{q}_k\right)}{1 - E_k(y_k)\left(\vec{q}_k^H \mathbf{C}_k^{-1} \vec{q}_k\right)},$$

to verify the energy calculation by comparing $\gamma_k(y_k)$ to $\gamma_k^*(y_k)$.

11. If $k = K$, terminate the steps. Otherwise set $k = k + 1$ and go to Step 4.

From the step-by-step procedures given above, it is clear that this energy calculation method requires no matrix inversion calculation in the iterations of calculating the energy $E_k(y_k)$. The inverse matrix $\mathbf{C}_{k-1}^{-1}$ does not require any update as it depends only on the energies, $E_1(y_1), E_2(y_2), \ldots, E_{k-1}(y_{k-1})$, which have been calculated previously. This is because the SIC scheme implemented in this research project is assumed to start cancelling the interference components from channel $K-1$ to channel 1. Therefore, any channel k only experiences interference from channel 1 to channel k (excluding its own energy component). The interference components from channel $k+1, \ldots, K$ are assumed to be removed by the SIC scheme.

Furthermore, the calculations required in this iterative energy update are only multiplications and divisions of some constants, which have been calculated before the iteration starts, as well as the energy calculated in the previous iteration. Hence, the overall computational complexity is further reduced. In the next subsection, two channel ordering methods are introduced to enhance this SIC-based energy calculation technique.

It should be noted that Step 9 and 10 in the SIC-based energy calculation method above are included to compute the SNR, $\gamma_k(y_k)$ in order to be compared with the target SNR $\gamma_k^*(y_k)$ for verification. Therefore these steps are optional and may not be included to further increase the speed of this energy calculation algorithm. Apart from these two steps, another method is also carried out to verify the calculation method. This verification method is based on a block

data transmission simulation, which transmits blocks of data and measure the average received SNR at the receiver end before it is compared to the target minimum SNR $\gamma_k^*(y_k)$.

If the received matrix at the receiver end is represented as $\mathbf{R}$, as given in Eq(3.4) and the despread signal is denoted as $\vec{\varkappa}_k^T = \vec{w}_k^H \mathbf{R}_k$, where $\mathbf{R}_k$ is the interference-reduced received signal matrix at channel k excluding the interference components in channel $k+1, \ldots, K$ and $\mathbf{R}_K = \mathbf{R}$, the normalized mean square error (MSE) MSE_k for channel k may be calculated as follows [75]:

$$\text{MSE}_k = \text{E}\left\{ \frac{\left(\vec{\varkappa}_k - |h|\sqrt{E_k(y_k)}\vec{x}_k\right)^H \left(\vec{\varkappa}_k - |h|\sqrt{E_k(y_k)}\vec{x}_k\right)}{|h|^2 E_k(y_k)\vec{x}_k^H \vec{x}_k} \right\}, \tag{4.31}$$

where $\vec{x}_k$ is the symbol vector transmitted over code channel k, as described in Section 3.3.2 on page 61, and $|h|^2 = \vec{h}^H\vec{h}$ is the channel gain. If a normalized MMSE despreading filter coefficient vector $\vec{w}_k = \frac{\mathbf{C}_k^{-1}\vec{q}_k}{\vec{q}_k^H \mathbf{C}_k^{-1}\vec{q}_k}$ is applied the above equation for the normalized MSE is further simplified to yield:

$$\text{MSE}_k = 1 - E_k \vec{q}_k^H \mathbf{C}_k^{-1} \vec{q}_k. \tag{4.32}$$

Therefore, the total normalized MSE is written as

$$\text{MSE}_T = \sum_{k=1}^{K} \text{MSE}_k. \tag{4.33}$$

As in Chapter 3, the average received SNR at each channel k may also be expressed in terms of the normalized MSE MSE_T, as given below

$$\gamma_k = \frac{1 - \text{MSE}_k}{\text{MSE}_k}, \tag{4.34}$$

except that this average received SNR measurement is based on the MSE equation in Eq(4.31), which is written for the case of SIC. These parameters, which

include the average received SNR, are measured at the output of the despreading filter at the receiver end. They are essential to evaluate performance of the proposed scheme as well as other schemes under consideration in Section 4.4.5. In the next section, channel ordering schemes, which may be used to enhance this energy calculation method, are described.

4.4.2 Channel Ordering Schemes

The performance of this SIC and CPP-based two-group resource allocation scheme can be further enhanced by first ordering the channels. The first method is named as the SIC channel ordering scheme and the second method is known as the equal energy channel ordering scheme. The output of these channel ordering algorithms is a matrix defined as $\mathbf{P}_{ord} = \begin{bmatrix} \vec{\psi}_1^T & \dots & \vec{\psi}_K^T \end{bmatrix}^T$, which is formed by K rows, where each row is denoted by a row vector $\vec{\psi}_k = [\psi_{k,1}, \psi_{k,2}, \dots, \psi_{k,K}]$. The κ_k-th element ψ_{k,κ_k} of this vector is equal to unity and the remaining elements are all zero.

SIC Channel Ordering

This channel ordering method applies the SIC-based energy calculation method to determine the new channel order. Initially, the SIC-based energy calculation process is run for a single channel. This single channel, as the first channel, is changed from $k = 1$ to $k = K$. Then, the channel κ_1 having the largest energy is chosen as the first newly ordered channel. In order to determine the next channel κ_2, the same process is repeated but excluding channel κ_1, which has been identified before. The inverse covariance matrix $\mathbf{C}_{\kappa_1}^{-1}$ calculated when determining channel κ_1 is kept and used for determining the second channel, ie. channel κ_2. In the same manner, the rest of the channels are determined until the new channel order is completely known.

The following step-by-step procedures are given to summarize this channel or-

dering scheme.

1. Initialize $S^{(\kappa)} = \{1, 2, \ldots, K\}$, $\mathbf{C}_{\kappa_0} = 2\sigma^2\mathbf{I}_{N+2\alpha}$, $\mathbf{C}_{\kappa_0}^{-1} = \frac{1}{2\sigma^2}\mathbf{I}_{N+2\alpha}$, $\mathbf{C}_{temp}^{-1} = \mathbf{0}_{N+2\alpha}$.

2. Set $\mathbf{P}_{ord} = \mathbf{I}_K$, $E_{sum} = 0$, $E_{max} = 0$ and $k = 1$.

3. While $j \in S^{(\kappa)}$

   (a) Compute the distance vectors, $\vec{d} = \mathbf{C}_{\kappa_{k-1}}^{-1}\vec{q}_j$, $\vec{d}_1 = \mathbf{C}_{\kappa_{k-1}}^{-1}\vec{q}_{j,1}$, $\vec{d}_2 = \mathbf{C}_{\kappa_{k-1}}^{-1}\vec{q}_{j,2}$.

   (b) Compute the weighting functions, $\xi = \vec{d}^H\vec{q}_j$, $\xi_1 = \vec{d}_1^H\vec{q}_{j,1}$, $\xi_2 = \vec{d}_2^H\vec{q}_{j,2}$, $\xi_3 = \vec{d}^H\vec{q}_{j,1}$, $\xi_4 = \vec{d}^H\vec{q}_{j,2}$.

   (c) Compute the target signal-to-noise ratio (SNR), $\gamma_j^*(y_j) = \Gamma_j\,(2^{y_j} - 1)$ for a given target data rate y_j and a gap value Γ_j.

   (d) Set $E_{j,1}(y_j) = 0$ and update the energy $E_j(y_j)$ iteratively using Eq(4.18) until the maximum iteration I_{max} is reached.

   (e) Compute the weighting functions, ζ, ζ_1 and ζ_2 by using Eq(4.27), Eq(4.28) and Eq(4.29) respectively.

   (f) Update the covariance matrix $\mathbf{C}_{\kappa_k}^{-1}$ using Eq(4.30).

   (g) If $E_{max} < E_j(y_j)$

      i. Set $E_{max} = E_j(y_j)$.

      ii. Set $\kappa_k = j$.

      iii. Set $\mathbf{C}_{temp}^{-1} = \mathbf{C}_{\kappa_k}^{-1}$.

4. Compute $\mathbf{C}_{\kappa_k}^{-1} = \mathbf{C}_{temp}^{-1}$, remove κ_k from $S^{(\kappa)}$ and update $E_{sum} = E_{sum} + E_{max}$.

5. Set $\mathbf{P}_k = \mathbf{I}_K$ and swap the k-th column with the κ_k-th column.

6. Update the channel ordering matrix $\mathbf{P}_{ord} = \mathbf{P}_k\mathbf{P}_{ord}$.

7. If $k = K$, go to the next step. Otherwise, set $k = k + 1$ and go to Step 2.

8. Output the channel ordering matrix $\mathbf{P}_{ord}$.

Equal Energy Channel Ordering

The implementation of this channel ordering technique is more straightforward as compared to the previous technique. By equally loading each code channel with $\frac{E_T}{K}$, the SNRs are measured for all channels before they are ordered in ascending order. Therefore, the new order of the channels follow the order of the SNRs.

This channel ordering technique is summarized as follows:

1. Intialize the channel ordering matrix $\mathbf{P}_{ord} = \mathbf{I}_K$.

2. Compute the covariance matrix $\mathbf{C} = \mathbf{Q}_e \tilde{\mathbf{A}}^2 \mathbf{Q}_e^H + 2\sigma^2 \mathbf{I}_{N+2\alpha}$ and initialize $S^{(k)} = \{1, 2, \ldots, K\}$.

3. Calculate the SNR, $\gamma_k = \dfrac{E_T/K \left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}{1 - E_T/K \left(\vec{q}_k^H \mathbf{C}^{-1} \vec{q}_k\right)}$ for $k = 1, 2, \ldots, K$.

4. Sort the calculated SNRs in ascending order such that channel $\kappa_1 = \arg\min_{k \in S^{(k)}} \gamma_k$ is the first newly ordered channel and channel $\kappa_K = \arg\max_{k \in S^{(k)}} \gamma_k$ is the last channel.

5. Reorder the rows of the channel ordering matrix $\mathbf{P}_{ord}$ from the κ_1-th row as the first row, κ_2-th row as the second row, until κ_K-th row as the last row.

6. Output the channel ordering matrix $\mathbf{P}_{ord}$.

In the next section, the SIC-based energy calculation method as well as these channel ordering schemes are incorporated into the two-group resource allocation scheme along with the CPP scheme.

4.4.3 An SIC-based Two-group Rate and Energy Calculation Method

In the previous chapter, two algorithms, which are Algorithm 2 (page 92) and 3 (page 94), are given in Section 3.7.2 to find p and m respectively. The following subsection presents the algorithms used to find these two variables when SIC is considered in the implementation of the two-group resource allocation scheme.

In order to realize the two-group total data rate of $R_T = (K-m)b_p+mb_{p+1}$, the index p of the lower data rate b_p and the number m of the channels loaded with the higher data rate b_{p+1}. Algorithm 4 shows the step-by-step mathematical procedures to find p using the SIC-based two-group resource allocation scheme.

As in Algorithm 2 in the previous chapter, that there are two major parts in Algorithm 4 applied to find p. The first part, which corresponds to the second *while* loop from the top, consists of only one single operation, which is in line 14. This step is used to perform the iterative energy calculation for channel k, which minimizes the required energy to achieve the target SNR $\gamma_k^*(b_j)$ of data rate b_j.

The implementation of the first part is different than that of Algorithm 2 since the energies are calculated using different formula when SIC is incorporated. Since no matrix inversions are required, the high computational complexity typically required to perform matrix inversion is reduced. The calculation process must be done from channel 1 to channel K as the inverse covariance matrix of channel k depends on the inverse covariance matrix of channel $k-1$ for $k = 1, \ldots, K$, where $\mathbf{C}_0^{-1} = \frac{1}{2\sigma^2}\mathbf{I}_{N+2\alpha}$.

The first part, which calculates the energy, is iterated until the energy is converged to $E_k(b_j) = E_{k,i}(b_j) \approx E_{k,i-1}(b_j)$ or the maximum iteration I_{max} is reached. This iterative process is an equal data rate implementation of the iterative energy calculation process summarized in Section 4.4.1.

Algorithm 4: A p Finder Algorithm for The SIC-based Two-group Resource Allocation Scheme

Data: $\tilde{\mathbf{H}}, \mathbf{S}, \mathbf{P}_{ord}, \varepsilon_\tau, \sigma^2, K, E_T, I_{max}, \gamma_k^*(b_j), b_j$ for $j = 1, 2, \ldots, P$
Result: $p, E_k(b_p), E_k(b_{p+1})$ for $k = 1, 2, \ldots, K$

begin
  $j \longleftarrow 0$, $\mathbf{S} \longleftarrow \mathbf{SP}_{ord}$
  $b_j \longleftarrow 0$, $E_k(b_j) \longleftarrow 0$ for $k = 1, 2, \ldots, K$
  $\vec{q}_k = \tilde{\mathbf{H}}\vec{s}_k$, $\vec{q}_{k,1} = \left(\mathbf{J}^T\right)^N \vec{q}_k$, $\vec{q}_{k,2} = \mathbf{J}^N \vec{q}_k$ for $k = 1, 2, \ldots, K$
  $\mathbf{C}_0^{-1} \longleftarrow \frac{1}{2\sigma^2}\mathbf{I}_{N+2\alpha_N}$
  while $\sum_{k=1}^{K} E_k(b_j) \leq E_T$ **and** $j \leq P$ **do**
    $j \longleftarrow j + 1$
    for $k \leftarrow 1$ **to** K **do**
      $i \longleftarrow 1$, $E_{k,i-1}(b_j) \longleftarrow 0$, $E_{k,i}(b_j) \longleftarrow \frac{E_T}{K}$
      $\vec{d} \longleftarrow \mathbf{C}_{k-1}^{-1}\vec{q}_k$, $\vec{d}_1 \longleftarrow \mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}$, $\vec{d}_2 \longleftarrow \mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}$
      $\xi \longleftarrow \vec{d}^H\vec{q}_k$, $\xi_1 \longleftarrow \vec{d}_1^H\vec{q}_{k,1}$, $\xi_2 \longleftarrow \vec{d}_2^H\vec{q}_{k,2}$, $\xi_3 \longleftarrow \vec{d}^H\vec{q}_{k,1}$,
      $\xi_4 \longleftarrow \vec{d}^H\vec{q}_{k,2}$, $\gamma_k^*(b_j) \longleftarrow \Gamma_j(2^{b_j} - 1)$
      while $\left((E_{k,i}(b_j) - E_{k,i-1}(b_j))^2 > \varepsilon_\tau\right)$ **or** $(i < I_{max})$ **do**
        Update $E_{k,i+1}(b_j) \longleftarrow \frac{\gamma_k^*(b_j)}{\xi - E_{k,i}(b_j)\left(\frac{|\xi_3|^2}{1+E_{k,i}(b_j)\xi_1} + \frac{|\xi_4|^2}{1+E_{k,i}(b_j)\xi_2}\right)}$
        $i \longleftarrow i + 1$
      end
      $E_k(b_j) \longleftarrow E_{k,i}(b_j)$
      $\zeta \longleftarrow \frac{E_k(b_j)}{1+\gamma_k^*(b_j)}$, $\zeta_1 \longleftarrow \frac{E_k(b_j)}{1+E_k(b_j)\xi_1}$, $\zeta_2 \longleftarrow \frac{E_k(b_j)}{1+E_k(b_j)\xi_2}$
      Update $\mathbf{C}_k^{-1}$ using Eq(4.30)
    end
  end
  $p \longleftarrow j - 1$
  $E_k(b_p) \longleftarrow E_k(b_{j-1})$ for $k = 1, 2, \ldots, K$
  $E_k(b_{p+1}) \longleftarrow E_k(b_j)$ for $k = 1, 2, \ldots, K$
end

The second part of this algorithm is executed by the outer *while* loop in Algorithm 4, which starts from line 7 and ends at line 16. It is exactly equal to the second part or the outer *while* loop of Algorithm 2 on page 92. It chooses different data rates and runs the iterative energy calculation process, which is implemented within this loop, from b_1 to b_{p+1}, so long as the current total allocated energy $\sum_{k=1}^{K} E_k(b_j)$ does not exceed the total energy E_T, where $j = 1, \ldots, p+1$.

Algorithm 5 shows a list of steps required to find m, which is the number of the channels loaded with the higher data rate b_{p+1} when SIC is incorporated with the two-group resource allocation scheme. After knowing the lower data rate index p, this algorithm is then invoked to determine m with the knowledge of the energy $E_k(b_p)$ and $E_k(b_{p+1})$ allocated to realize b_p and b_{p+1} respectively besides p and other parameters used in the p finder algorithm. The core process in Algorithm 5 is again the iterative energy calculation process, which is implemented by a single line 17. This iterative energy process is repeated in a larger *while* loop, which starts from line 7, for a different target data rate $y_k = b_{p+1}$ for $k = 1, \ldots, m$, while the rest of the channels are targetted to be realized with b_p. When the total energy allocated $\sum_{k=1}^{(j)}(y_k)$ is greater than the total constraint energy E_T, then the number of the channels in the higher data rate group should be $m = j - 1$.

The next section presents the proposed receiver structure, into which the SIC scheme is incorporated into a CPP-based receiver, to complete the proposed transmission model.

4.4.4 The Step-by-step SIC and CPP-based Receiver Implementation

Upon receiving the signal, which is denoted as matrix $\mathbf{R}$, as seen in Figure 4.2, the receiver begins to carry out the despreading, the decoding, the regeneration

Algorithm 5: An m Finder Algorithm for The SIC-based Two-group Resource Allocation Scheme

Data: $\tilde{\mathbf{H}}, \mathbf{S}, \mathbf{P}_{ord}, \varepsilon_\tau, \sigma^2, K, E_T, I_{max}, \gamma_k^*(b_p), \gamma_k^*(b_{p+1}), b_p, b_{p+1}, E_k(b_p), E_k(b_{p+1})$ for $k = 1, 2, \ldots, K$

Result: $m, E_k^{(m)}(y_k)$ for $k = 1, 2, \ldots, K$

begin
- $j \longleftarrow 0$, $\mathbf{S} \longleftarrow \mathbf{SP}_{ord}$
- $y_k \longleftarrow b_p$, $E_k^{(j)}(y_k) \longleftarrow E_k(b_p)$ for $k = 1, 2, \ldots, K$
- $\vec{q}_k = \tilde{\mathbf{H}}\vec{s}_k$, $\vec{q}_{k,1} = \left(\mathbf{J}^T\right)^N \vec{q}_k$, $\vec{q}_{k,2} = \mathbf{J}^N \vec{q}_k$ for $k = 1, 2, \ldots, K$
- $\mathbf{C}_0^{-1} \longleftarrow \frac{1}{2\sigma^2}\mathbf{I}_{N+2\alpha_N}$
- **while** $\sum_{k=1}^{K} E_k^{(j)}(y_k) \leq E_T$ **do**
  - $j \longleftarrow j + 1$
  - **for** $k \leftarrow 1$ **to** K **do**
    - $i \longleftarrow 1$
    - $y_k \longleftarrow b_{p+1}$, $\Gamma_k = \Gamma_{p+1}$ if $k \leq j$
    - $y_k \longleftarrow b_p$, $\Gamma_k = \Gamma_p$ if $k > j$
    - $E_{k,i-1}^{(j)}(y_k) \longleftarrow 0$, $E_{k,i}^{(j)}(y_k) \longleftarrow \frac{E_T}{K}$ for $k = 1, 2, \ldots, K$
    - $\vec{d} \longleftarrow \mathbf{C}_{k-1}^{-1}\vec{q}_k$, $\vec{d}_1 \longleftarrow \mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}$, $\vec{d}_2 \longleftarrow \mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}$
    - $\xi \longleftarrow \vec{d}^H\vec{q}_k$, $\xi_1 \longleftarrow \vec{d}_1^H\vec{q}_{k,1}$, $\xi_2 \longleftarrow \vec{d}_2^H\vec{q}_{k,2}$, $\xi_3 \longleftarrow \vec{d}^H\vec{q}_{k,1}$, $\xi_4 \longleftarrow \vec{d}^H\vec{q}_{k,2}$, $\gamma_k^*(y_k) \longleftarrow \Gamma_k(2^{y_k} - 1)$
    - **while** $\left(\left(E_{k,i}^{(j)}(y_k) - E_{k,i-1}^{(j)}(y_k)\right)^2 > \varepsilon_\tau\right)$ **or** $(i < I_{max})$ **do**
      - Update $E_{k,i+1}(y_k) \longleftarrow \frac{\gamma_k^*(y_k)}{\xi - E_{k,i}(y_k)\left(\frac{|\xi_3|^2}{1+E_{k,i}(y_k)\xi_1} + \frac{|\xi_4|^2}{1+E_{k,i}(y_k)\xi_2}\right)}$
      - $i \longleftarrow i + 1$
    - **end**
    - $E_k^{(j)}(y_k) \longleftarrow E_{k,i}^{(j)}(y_k)$ for $k = 1, 2, \ldots, K$
    - $\zeta \longleftarrow \frac{E_k(y_k)}{1 + \gamma_k^*(y_k)}$, $\zeta_1 \longleftarrow \frac{E_k(y_k)}{1 + E_k(y_k)\xi_1}$, $\zeta_2 \longleftarrow \frac{E_k(y_k)}{1 + E_k(y_k)\xi_2}$
    - Update $\mathbf{C}_k^{-1}$ using Eq(4.30)
  - **end**
- **end**
- $m \longleftarrow j - 1$
- $E_k^{(m)}(y_k) \longleftarrow E_k^{(j-1)}(y_k)$ for $k = 1, 2, \ldots, K$

end

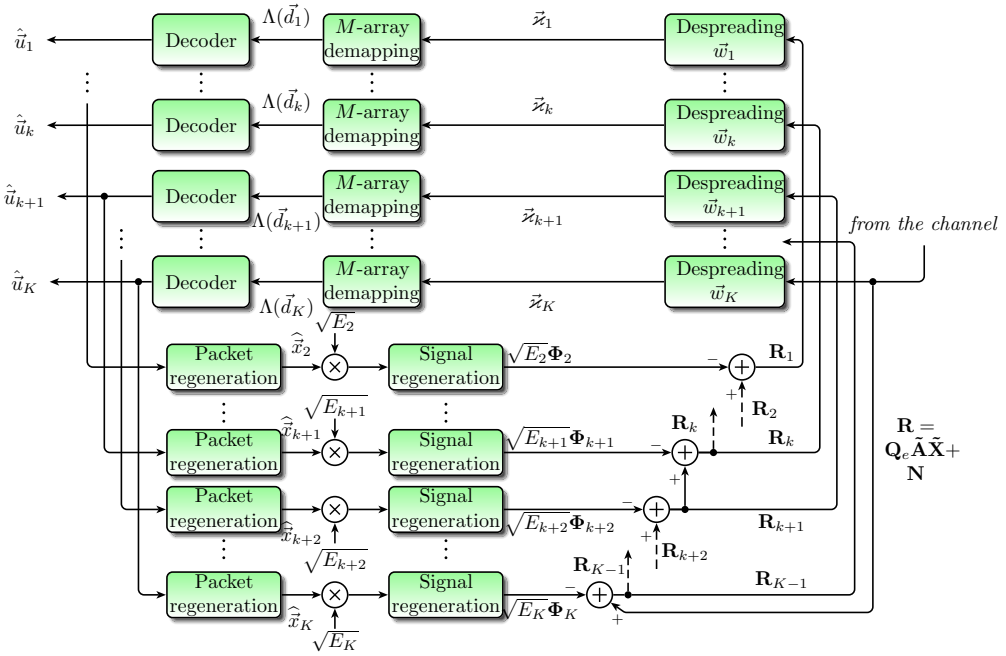

Figure 4.2: The Receiver Structure

and the interference processes from channel K to channel 1. The following list of procedures summarizes the required steps to accomplish these processes.

1. Assuming that the channel side information, including the channel impulse response $\vec{h}$ and the noise variance σ^2, is known to the receiver, the SIC-based calculation method is rerun on the receiver end to calculate the energy $E_k(y_k)$ as well as the MMSE despreading filter coefficient vector $\vec{w}_k$ for $k = 1, 2, \ldots, K$.

2. Set $k = K$ and initialize the received signal matrix as $\mathbf{R}_K = \mathbf{R} = \mathbf{Q}_e\tilde{\mathbf{A}}\tilde{\mathbf{X}} + \mathbf{N}$.

3. With the k-th MMSE despreading filter coefficient vector $\vec{w}_k$, the current received signal matrix $\mathbf{R}_k$ is despread to produce $\vec{\varkappa}_k^T = \vec{w}_k^H\mathbf{R}_k$.

4. This despread signal vector $\vec{\varkappa}_k$ is then demodulated to generate a soft-bit demodulated vector $\Lambda(\vec{d}_k)$.

5. With this soft-bit vector $\Lambda(\vec{d}_k)$, the CPP decoder is run to iteratively decode this demodulated vector to produce $\hat{\vec{u}}_k$.

6. If $k = 1$, terminate the algorithm. Otherwise go to the next step.

7. With $\hat{\vec{u}}_k$, the CPP encoder is rerun again to regenerate the symbol vector as $\widehat{\vec{x}}_k$ by recoding and remodulating $\hat{\vec{u}}_k$.

8. With the calculated energy $E_k(y_k)$ in Step 1 and the regenerated packet $\widehat{\vec{x}}_k$, the received signal to be removed is regenerated as $\sqrt{E_k(y_k)}\mathbf{\Phi}_k$. The signal $\mathbf{\Phi}_k$ is calculated using Eq(4.4).

9. Set $k = k-1$ and perform interference cancellation by producing a reduced received signal matrix as $\mathbf{R}_k = \mathbf{R}_{k+1} - \sqrt{E_{k+1}(y_{k+1})}\mathbf{\Phi}_{k+1}$.

10. Go to Step 3.

In the next section, the results obtained from running these proposed SIC-based schemes are presented and discussed.

4.4.5 Performance Evaluation

Parameter	Value
The number of code channels, K	15
The spreading factor, N	16
The number of propagation paths, L	4
The chip rate, $\frac{1}{T_c}$	3.84 Mchips/s
The noise variance	0.02
The signature sequence	orthonormal signature sequences
The data rate, b_p	$0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0,$ $2.25, 2.5, 2.75, 3.0, 3.75, 4.0, 4.25$
The gap value, Γ_p	$2.95, 2.78, 2.96, 2.98, 2.90, 3.00, 3.17,$ $2.96, 3.61, 3.72, 3.55, 3.33, 4.08, 3.75, 3.88$
The channel impulse response	$\vec{h} = \begin{bmatrix} 0.7297 & 0.5166 & 0.3657 & 0.2589 \end{bmatrix}^T$

Table 4.1: The parameters used for the simulation

Several tests have been carried out to observe performance of the proposed SIC-based two-group resource allocation scheme. The measurement parameters are the total normalized MSE, $\text{MSE}_T = \sum_{k=1}^{K} \text{MSE}_k$, the sum capacity, $C_T = \frac{R_T}{NT_c}$ and the total average received SNRs, $\gamma_T = \sum_{k=1}^{K} \gamma_k$. where γ_k is taken from Eq(4.34). After observing and analyzing the results of these tests, a comparison is made between the proposed SIC-based two-group resource allocation scheme and the existing schemes in terms of the computational complexity as well as the number of realizable total discrete rates. All these tests have been run based on the parameters defined in Table 4.1, unless otherwise stated.

4.4.6 Total Normalized Mean Square Error Observation

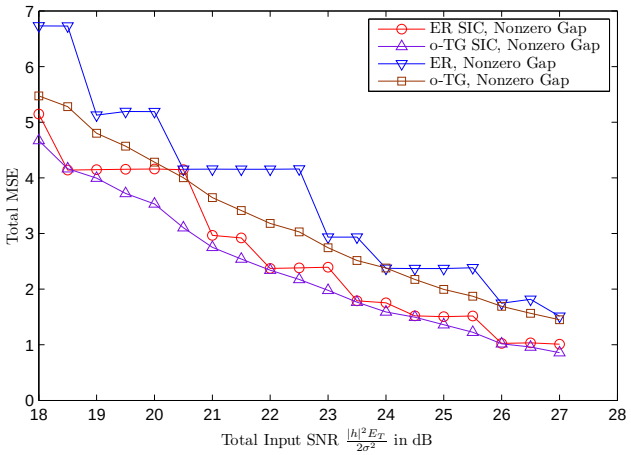

Figure 4.3: The total MSE obtained when the two-group resource allocation scheme and the equal rate margin adaptive loading scheme were implemented with SIC.

To begin the series of performance evaluation, the two-group resource allocation scheme and the equal rate margin adaptive loading scheme were implemented with and without SIC in order to observe and measure the total normalized MSE MSE_T for these schemes. The discrete data rates $\{b_p\}_{p=1}^{P}$ in Table 4.1 have been used with the corresponding nonzero gap values $\{\Gamma_p\}_{p=1}^{P}$ given in the same table. This total normalized MSE is calculated using Eq(4.33). For brevity, the two-

group resource allocation scheme is represented as o-TG when SIC is not in place and o-TG SIC when SIC is integrated with the two-group algorithm. In the same manner, the equal rate margin adaptive loading scheme is denoted as ER when SIC is not applied and ER SIC when SIC is embedded. Figure 4.3 shows the measured total normalized MSE plotted against a range of total input SNRs from 18 dB to 27 dB after running the aforementioned test. From this figure, it is clear that the total normalized MSE for the two-group loading scheme is lower than that of the equal rate margin adaptive loading scheme at most of the total input SNRs. This is because the two-group resource allocation scheme is designed to maximize the utilization of the total energy, hence minimizing the residual energy, to improve the total data rate. As a result, the allocated energy at each channel is maximized by minimizing the normalized MSE, hence maximizing the achievable target SNR for a given total energy. When SIC is integrated, the interference components, which are in the covariance matrix, are reduced to increase the SNR, hence reduce the normalized MSE. Therefore the o-TG SIC scheme is preferable for minimizing the total normalized MSE. In the next subsection, further evaluations and results are presented and discussed based on the total average received SNR and the system throughput. .

4.4.7 Total Average Received SNR and System Throughput Observation

Figure 4.4 shows the total average received SNR γ_T measured over a range of total input SNRs when the two-group resource allocation scheme and the equal rate margin adaptive loading scheme were tested. The total average received SNR γ_T was measured using Eq(3.36). The discrete data rates and the corresponding gap values are also taken from Table 4.1. As seen in Figure 4.4, the total average received SNR achieved by the equal rate margin adaptive loading (ER) scheme, which is configured without SIC, is the lowest. This is expected because the ER scheme is run to achieve the same SNR on each

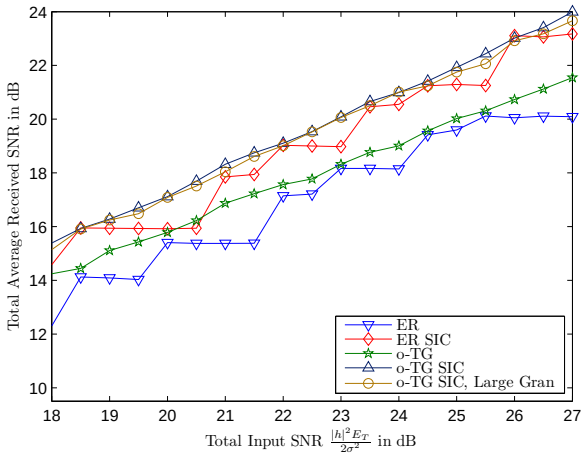

Figure 4.4: The total average received SNR obtained when the two-group resource allocation scheme was implemented with and without SIC

channel, leaving an amount of residual energy which may actually be further utilized to increase the total average received SNR. As for the o-TG scheme, the total average received SNR is slightly higher than the ER scheme at almost all total input SNRs provided during this test. When the o-TG scheme is integrated with SIC, the total average received SNR is significantly increased as the amount of interference is reduced. This o-TG SIC scheme was also run with a smaller set of discrete data rates with larger bit granularities or differences between two adjacent data rates. These data rates are $0.5, 1.0, 1.5, 2.0, 3.0, 4.0$, which are a subset of the set of discrete data rates given in Table 4.1. The total average received SNR measured from running this o-TG SIC scheme with large bit granularities is seen to be approximately equal to that of the o-TG SIC scheme, run with the original set of discrete data rates, which have smaller bit granularities. Therefore, this result demonstrates the advantage of running the two-group resource allocation scheme, which may apply only a small set of discrete data rates with large bit granularities to achieve the same level of performance as in the case of small bit granularities.

in Figure 4.5, the system throughput in Mbps is plotted after running the two-

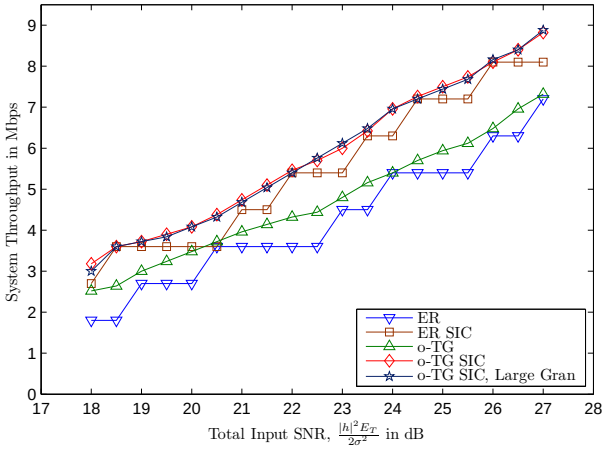

Figure 4.5: The system throughput obtained when the SIC-based two-group (o-TG SIC) loading was implemented

group resource allocation scheme and the equal rate margin adaptive loading scheme with the set of data rates and gap values given in Table 4.1. The same schemes used in the previous test are also examined in this test, which are the ER, the ER SIC, the o-TG and the o-TG SIC schemes. As expected, the o-TG SIC scheme performed better than the rest of the schemes as the measured system throughput for this scheme is the highest. Likewise, the system throughput measured from running the o-TG SIC scheme with the discrete data rates of large bit granularities is also very close to that of the original o-TG SIC scheme, which was run using the original set of discrete data rates in Table 4.1.

Another investigation on the system throughput was made on the SIC-based two-group resource allocation scheme and other schemes under consideration. The result of this test is plotted in Figure 4.6. In this test, all schemes were run with the same set of discrete data rates given in Table 4.1 but at zero gap values in dB. The schemes tested are the o-TG SIC scheme with two different channel ordering schemes and the equal rate and equal energy loading (EREE) scheme. When comparing the system throughput achieved by the two-group resource allocation scheme after it was separately run with two different chan-

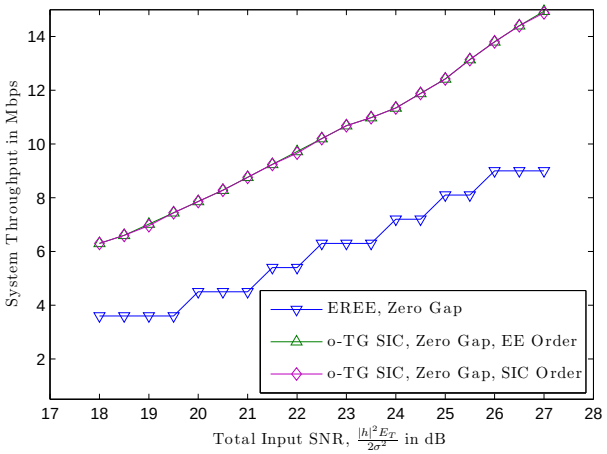

Figure 4.6: The system throughput obtained when the SIC-based two-group (o-TG SIC) loading was implemented

nel ordering schemes, it can be seen that the system throughput for each of the two channel ordering schemes is approximately equal to other. Therefore, this result confirms that both channel ordering schemes are good at enhancing performance of the two-group resource allocation scheme. However, the equal energy channel ordering scheme is more preferable as it is computationally more efficient. Another scheme tested is the EREE scheme, which is observed to have the lowest system throughput as it only allocates the same data rate to each channel. This equal data rate is chosen based on the channel with the lowest SNR, rendering the chosen data rate to be relatively small. Therefore the resultant system throughput becomes small.

Loading scheme	Matrix inversions	Discrete rates
ER	PI_{max}	P
o-TG	$(P+K-1)I_{max}$	$K(P-1)+1$
ER SIC	0	P
o-TG SIC	0	$K(P-1)+1$

Table 4.2: Comparison of the number of matrix inversions and realizable discrete rates

In order to examine the feasibility of implementing the proposed SIC-based

two-group loading scheme in practical systems, a brief analysis on the computational complexity required by the loading schemes under consideration, with and without the SIC-based energy calculation method, is carried out. The result of this analysis is tabulated in Table 4.2. The computational complexity is measured in terms of the number of matrix inversions required to complete the resource allocation process. It should be noted that I_{max} is the maximum number of iterations of the energy calculation process. Apart from the number of matrix inversions, the number of realizable total discrete rates is also shown on the third column for each scheme. As this number is higher, the corresponding loading scheme is more preferable as more realizable total discrete rates are possible for a range of total input SNRs. To ensure that a fair comparison is made, it is ensured that the lowest and the highest realizable total discrete rates for all loading schemes are equal to each other. In this table, the lowest total discrete rate in bits per symbol is Kb_1 and the highest total discrete rate in bits per symbol is Kb_P.

From this table, it can be seen that the o-TG SIC and the ER SIC schemes are better than the rest of the schemes in terms of the number of required matrix inversions since both of these schemes require no matrix inversions. As shown in Section 4.4.1 on page 126, there are only a number of scalar and vector multiplications and divisions required for the SIC-based energy calculation method. When comparing the o-TG SIC and the ER SIC in terms of the number of realizable total discrete rates, it can be seen in this table that the number of realizable total discrete rates by the o-TG SIC scheme is much higher than that of the ER SIC scheme, assuming that all K channels are used.

In the next section, another version of SIC and CPP-based resource allocation scheme which distribute the coded packet packets produced from the CPP coding scheme over the parallel channels with an equal amount of energy is described.

4.5 The CPP-based Fixed Energy Loading Technique

The results obtained from running the proposed SIC-based two-group resource allocation scheme, as presented in the previous section, demonstrate the effectiveness of implementing this loading scheme in minimizing the energy required to transmit the data and improving the total transmission rate. Although the SIC-based two-group resource allocation scheme significantly minimizes the energy and enhances the achievable transmission data rate, this loading scheme requires the knowledge of the channel side information, which includes the channel impulse response and the noise variance.

If this information is not perfectly known, a good estimation of this knowledge is needed so that the calculation of the energy, the covariance matrix and the MMSE despreading filter coefficient is near perfect. This is essential especially when the SIC scheme is in place in order to avoid any errors which might propagate from one channel to another durring the successive process of removing the interference components as well as despreading and decoding the received signal. One possible solution to avoid the need of accurate information of the transmission channel condition is to load an equal amount of energy [1, 39] to each channel with an equal data rate. As explained in the beginning of Chapter 3, this approach may result in a significant waste of SNR as the channels with high SNRs, which are sufficiently high to load higher data rates, are also allocated the same data rate.

Eliminating this wasted SNR problem using the equal rate margin adaptive loading schemes [48, 69, 70] or the two-group resource allocation schemes [29, 74, 80–82] is possible, but these schemes unfortunately require the knowedledge of channel side information, which may not be accurately acquired in practice. Hence, a different loading scheme which requires no channel side information at

the transmitter end is proposed in this section. By using the code parity packet transmission model [29, 30], which was developed based on the CPP coding scheme, a fixed energy loading scheme is implemented such that the packets generated by the packet-based CPP scheme are allocated an equal energy before they are distributed over the parallel code channels. As the performance of the CPP scheme improves when the number of packets generated increases for a given fixed data rate, more parallel code channels may be employed to be distributed with the coded parity packets generated by the CPP scheme. The main focus of this section is to investigate the new gap between the SNR required by the CPP scheme and the theoretical minimum SNR when these coded parity packets are distributed and transmitted over the channels, which are varied in SNR values due to the fixed energy loading approach.

In the following section, a new system model in which the proposed adaptive coded parity packet transmission model is designed with the fixed energy loading scheme.

4.5.1 System Model

A multi-code CDMA system is considered with ε code channels, each of which is realizable with b_p bits per symbol where b_p is chosen from a set of P realizable data rates in bits per symbol. These data rates are produced from a combination of a channel encoder and ε M-array mappers. In this section, this data rate realization is performed by the proposed CPP scheme. When the data rate b_p is determined, the channel encoder is run on a block of data $\vec{u} = \begin{bmatrix} \vec{u}_1 & \dots & \vec{u}_V \end{bmatrix}$ comprising of V source packets, each of which denoted as a column vector $\vec{u}_v$ for $v = 1, 2, \dots, V$. There are ε packets generated from the channel encoding process which are denoted as column vectors $\vec{d}_1, \vec{d}_2, \dots, \vec{d}_\varepsilon$. Each of these channel encoded packets are distributed over ε channels, where $\varepsilon \leq K$ and K is the maximum number of permissible code channels. Then,

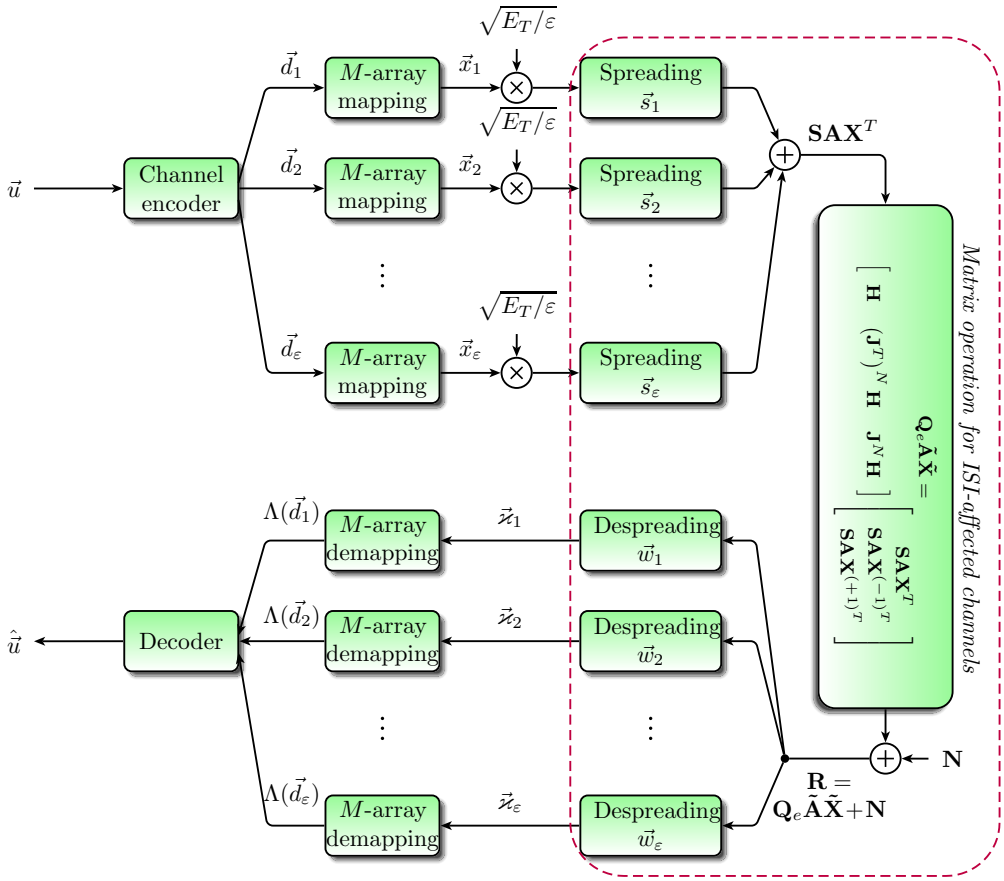

Figure 4.7: The multi-code transmission system block diagram

these packets are passed on to M-array mappers to produce unit-energy symbol vectosr $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_\varepsilon$. Each of these symbol vectors is then loaded with an equal energy $E_e = \frac{E_T}{\varepsilon}$ for $e = 1, 2, \ldots, \varepsilon$.

At the receiver end, the reverse process is run where the received signal matrix $\mathbf{R}$ is despread using a despreading filter $\vec{w}_e$ for $e = 1, 2, \ldots, \varepsilon$. Then, an M-array demapping process is run on each of the despread signal $\vec{\varkappa}_e = \vec{w}_e^H \mathbf{R}$ to produce a soft-bit vector of $\Lambda(\vec{d}_e)$. A decoding process is then invoked using this soft-bit vector as the input to produce a decoded block of data, $\hat{\vec{u}}$. In the next subsection, the problem addressed is described before the proposed fixed energy loading and the adaptive coded parity packet transmission models are presented.

4.5.2 Problem Description

In this thesis, a number of existing loading schemes have been investigated. The proposed fixed energy loading with the adaptive coded parity packet transmission model is designed to address the problems, which are found in the existing loading schemes, as follows:

Independent of The Channel Side Information

Most of the existing loading schemes such as the margin [69, 70] and rate [50, 51] adaptive loading schemes require an accurate channel side information to perform the energy calculation process. This information, which includes the channel impulse response $\vec{h}$ and the noise variance σ^2, may not be acquired accurately in practice. Therefore, the energy and data rates which have been calculated based on this inaccurate information may not be the correct values. Hence, performance of these loading schemes tend to degrade.

Wasted SNR Reduction

One of the commonly adopted loading techniques to solve the problem of acquiring accurate channel side information is fixed energy and equal rate loading [1]. With this loading technique, no energy calculation is required as the total energy E_T is loaded equally to all channels. Therefore, the channel side information is not required at the transmitter end to calculate the energy and perform the resource allocation operation. However, as the equal data rate is determined from the channel with the smallest SNR, there is a relatively high amount of SNR which is wasted. This is because the channels with high SNRs are loaded with the same equal data rate, which is relatively low for these channels. Therefore, the allocated SNR, hence the energy, on each code channel is always more than it needs (the minimum SNR, hence the energy) to send the data at the determined equal data rate. Therefore, the SNR is wasted. As a solution, the fixed energy loading technique developed and presented in this chapter takes the average SNR, and not the minimum SNR, as the measure to determine the data rate to be loaded. By doing this, the wasted SNR can be minimized.

Gap Value Control and Reduction

It will be demonstrated in this chapter, particularly in this section, that the capacity-approaching CPP scheme is able to control and reduce the gap value, Γ_p, when the number ε of generated packets, hence the number of employed code channels, is increased by an adjustment parameter α, as discussed in Chapter 2 on page 19.

Therefore, a novel fixed energy loading with an adaptive coded parity packet transmission model is proposed in this chapter to avoid the need for channel side information, which tends to be inaccurate in practice and reduce the wasted SNR problem. The next subsection describes this proposed scheme.

4.5.3 Fixed Energy Loading with Adaptive Coded Parity Packet Transmission

Consider the multi-code transmission model illustrated in Figure 4.7. The fixed energy loading considered in this chapter allocates an equal energy E_T/ε and also an equal data rate b_p to each code channel. In order to determine the equal data rate b_p at which the generated data packets should be submitted, an equal energy E_T/ε is first allocated to a test packet provided on each of the ε parallel code channels. Upon receiving these packets, the receiver measures the average SNR per symbol before sending this information via a feedback signal to the transmitter. If the SNRs over the channels at the receiver end are denoted as $\{\gamma_e\}_{e=1}^{\varepsilon}$, the average SNR among all channels is calculated as

$$\gamma^{(\varepsilon)} = \frac{1}{\varepsilon}\sum_{e=1}^{\varepsilon}\gamma_e. \tag{4.35}$$

By using this average SNR, $\gamma^{(\varepsilon)}$, the data rate b_p that should be loaded is determined by satisfying the following criterion

$$\gamma_p^* \leq \gamma^{(\varepsilon)} < \gamma_{p+1}^*, \tag{4.36}$$

where $\gamma_p^* = \gamma_p(2^{b_p} - 1)$ is the target minimum SNR to realize b_p bits per symbol for $p = 1, 2, \ldots, P$, assuming that $\gamma_{P+1}^* = +\infty$. This is the simple fixed energy loading scheme proposed for the adaptive coded parity packet transmission model, which will be briefly explained next.

Adaptive Coded Parity Packet Transmission

When the data rate b_p is known, the channel encoding and the M-array mapping processes are then run accordingly based on the CPP scheme, which produces

the data rate b_p as

$$b_p = \left(\frac{V_o}{\varepsilon_o} \frac{N_U}{B} \log_2 M \right)_p , \tag{4.37}$$

where there is a specific combination of an outer code rate $r_{out} = \frac{V_o}{\varepsilon_o}$, an inner code rate $r_{in} = \frac{N_U}{B}$ and a QAM constellation size M for a data rate b_p.

Both V_o and $\varepsilon_o - V_o$ are the initial numbers of source and parity packets generated by the CPP outer encoding process, as explained on page 16. In order to control the gap value, the number of source and parity packets is increased by the adjustment parameter α to $V = \alpha V_o$ and $\varepsilon - V = \alpha (\varepsilon_o - V_o)$ respectively without changing the effective data rate b_p. It has been demonstrated in Chapter 2, Section 2.6 on page 37 that as α increases, the gap value decreases because the number $L_\varepsilon = 2^{(\alpha(\varepsilon_o - V_o)-1)}$ of alternative additional input information values for the decoding process is increased. Therefore, based on the same concept, the number of code channels is chosen to be equal to the total number generated packets $\varepsilon = \alpha \varepsilon_o$. This number is made sufficiently large to have a relatively low gap value.

Gap Value Investigation

When the CPP scheme is performed with the fixed energy loading scheme over multi-code channels, each of these channels tend to have a different SNR. As a result, the effective gap value tends to vary. Therefore, investigation is needed to observe the variation in the gap value.

When the SNR γ_e in each channel varies, the gap value will also be likely to change. Therefore, a target data rate of b_p, which originally has a gap value of Γ_p, will now has a different gap value as a function of the SNR variance, σ_ε^2, given as

$$\sigma_\varepsilon^2 = \frac{1}{\varepsilon} \sum_{e=1}^{\varepsilon} \left| \gamma^{(\varepsilon)} - \gamma_e \right|^2 . \tag{4.38}$$

Therefore, in the next section, several tests are run to examine the new gap

value $\Gamma_{p,\sigma_\varepsilon^2}$ for b_p when the fixed energy loading which results in varying SNRs of variance σ_ε^2 is implemented with this adaptive coded parity packet transmission model. This new gap value is then compared with the gap value obtained using existing modulation and coding schemes.

4.5.4 Numerical Results

In order to test the proposed fixed energy loading scheme with the adaptive coded parity packet transmission model, there are two CPP data rates used, which are $b_p = 1.33, 1.78$. The first data rate $b_p = 1.33$ bits per symbol is produced by an outer encoder rate of $r_{out} = \frac{2}{3}$, an inner encoder rate of $r_{in} = \frac{1}{2}$ and a QAM constellation size of 16. The initial number V_o of source packets is 2 and the initial number of parity packets is $\varepsilon_o - V_o = 1$. In order to control the gap value, there are $\varepsilon = 12$ packets generated from the CPP scheme. This number is produced from the CPP scheme using an adjustment parameter $\alpha = 4$, which results in $\varepsilon = \alpha\varepsilon_o = 12$.

In order to simulate the varying nature of the SNRs over the code channels, these SNRs are generated to be normally distributed with a known target variance and mean. By using the target SNR mean, the data rate is determined accordingly using Eq(4.36). Then, the CPP scheme and the corresponding modulation scheme are run to produce the coded and modulated packets based on this data rate, before they are transmitted at the normally distributed SNRs, generated earlier on. The packet error rate is then recorded for the corresponding SNR variance and SNR mean. The same process is now repeated for the same SNR variance but a higher SNR mean in order to improve the packet error rate. This repetition continues for higher SNR means until the target packet error rate of 1×10^{-4} is achieved. This simulation is made based on the following assumptions:

- The SNRs remain unchange along the duration (including the allocation,

the coding, the modulation and the transmission processes).

- A multi-code HSDPA model is used, where the transmission time delay (TTI) is reduced from 10ms to 2ms to reduce the delay for a better adaptation to the condition of the transmission channel/link.

- No outage occurs, assuming that the smallest SNR over the code channels is always larger than the minimum SNR threshold permissible for the receiver to detect the received signal.

There are four sets of generated SNRs, each of which has a different SNR mean but an equal SNR variance, σ_{ε}^2. Each set of SNRs is denoted as $S_j^{(\sigma_{\varepsilon}^2)} = \left\{\gamma_{j,1}^{(\sigma_{\varepsilon}^2)}, \ldots, \gamma_{j,\varepsilon}^{(\sigma_{\varepsilon}^2)}\right\}$, which contains ε SNRs for each of the ε code channels and has an SNR mean of $\gamma_{E,j}^{(\sigma_{\varepsilon}^2)}$. With these sets of SNRs, the adaptive parity packet transmission model is run to observe the packet error rates for each of the SNR sets. From these packet error rates, the new gap value $\Gamma_{p,\sigma_{\varepsilon}^2}$ is measured at a target packet error rate of 1×10^{-4}.

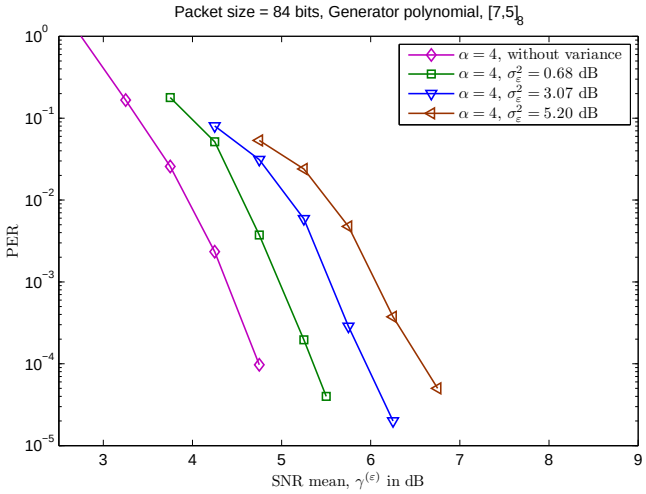

Figure 4.8: The packet error rates for the proposed CPP scheme at $b_p = 1.33$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$, and $\alpha = 4$ at varying SNRs in the channels with different means and variances.

Figure 4.8 shows the packet error rates measured when the adaptive coded parity packet transmission model was run at $b_p = 1.33$ bits per symbol, where

packets of size 84 bits were transmitted over each code channel and fed with a different SNR value, $\gamma_{j,e}^{(\sigma_\varepsilon^2)}$, which was mainly characterized by the corresponding target SNR variance σ_ε^2 for $e = 1, 2, \ldots, \varepsilon$. When the target SNR variance is $\sigma_\varepsilon^2 = 0.68$ dB, it can be seen from this figure that the new gap value $\Gamma_{p,\sigma_\varepsilon^2}$ measured is $\Gamma_{p,\sigma_\varepsilon^2} = 1.1\Gamma_p$, which is about 0.5 dB larger than the original gap Γ_p value for the case of zero SNR variance. Almost the same pattern can be seen for the case of other SNR variances, $\Gamma_{p,\sigma_\varepsilon^2} = 3.07$ dB and $\Gamma_{p,\sigma_\varepsilon^2} = 5.20$ dB, where the increase in the gap value is roughly between 0.5 to 0.7 dB. Despite the increase, this gap value is still smaller than the gap value recorded when a Turbo-QAM system having the same data rate of 1.33 bits per symbol is tested, as given in Figure 4.10. At a target SNR variance of $\sigma_\varepsilon^2 = 0.68$ dB, the gap value of the proposed adaptive coded parity packet transmission model is about 2 dB lower than that of the Turbo-QAM system. When the target SNR variance is set at $\sigma_\varepsilon^2 = 5.20$ dB, the gap value achieved by the proposed model is still lower than that of the Turbo-QAM scheme by roughly 0.7 dB. This result demonstrates the gap control and reduction advantage of the proposed adaptive coded parity packet transmission model when operated with the fixed energy loading scheme, besides its robustness to operate at varying SNRs over the parallel code channels.

Figure 4.9 shows another plot of packet error rates when the proposed transmission model was run at $b_p = 1.78$ bits per symbol with the same packet size of 84 bits. The similar test was run, where sets of SNRs with different means but an equal variance were generated and fed to the system. From the figure, it can be observed that the increase in the gap value is smaller than that of the previous case in the previous figure. When the target SNR variance is $\sigma_\varepsilon^2 = 0.79$ dB, the increase in the gap value from the original gap value Γ_p is less than 0.25 dB. At $\sigma_\varepsilon^2 = 5.20$ dB, the increase in the gap value is about 1 dB, which is about half of the increase in the gap value for the previous case in the previous figure. Based on this result it can be concluded that the increase in the gap value in

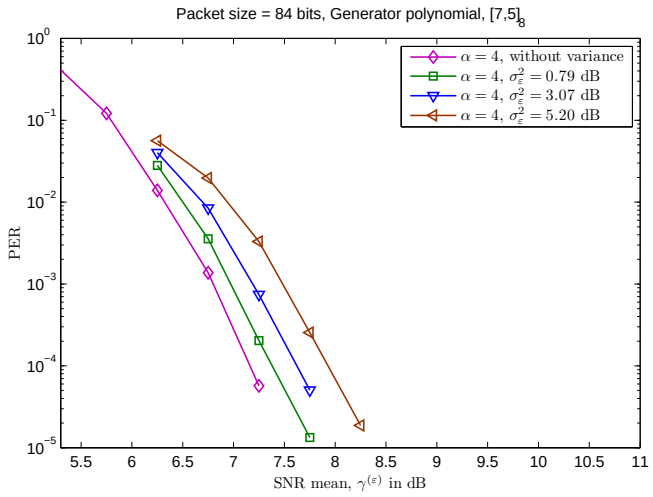

Figure 4.9: The packet error rates for the proposed CPP scheme at $b_p = 1.78$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$, and $\alpha = 4$ at varying SNRs in the channels with different means and variances.

dB, when the adaptive coded parity packet transmission model is integrated with the fixed energy loading scheme, becomes smaller as the data rate b_p is increased. This is because, as the data rate increases, the target minimum SNR to realize this data rate is also increased. Therefore if the SNR mean is also increased to be higher than this relatively large target SNR, the corresponding target SNR variance is relatively smaller as compared to this larger SNR mean. Hence, the performance of the proposed adaptive coded parity packet transmission model is not severely affected by the SNR variance.

In Figure 4.10 and 4.11, the packet error rates resulted when different modulation and coding schemes or transmission models were tested. Figure 4.10 shows the packet error rates of a Turbo-QAM system operating at $b_t = 1.33$ bits per symbol with packets of size 84 bits. As discussed earlier on, the gap value achieved by this scheme is still larger than all the new gap values achieved by the proposed adaptive coded parity packet transmission model when the SNR variances are larger than zero, as shown in Figure 4.8. There are also other packet error rates, which have been produced by the CPP scheme at different values of α but zero SNR variance, plotted in this figure. These results are also

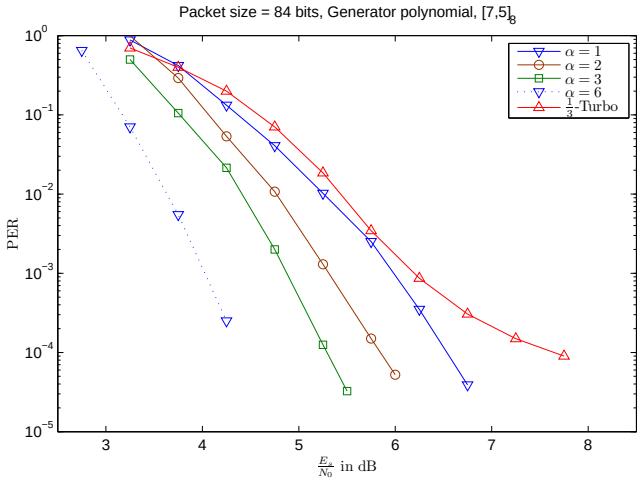

Figure 4.10: The packet error rates for the proposed CPP scheme at $b_p = 1.33$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$, with varying adjustment parameter values $\alpha = 1, 2, 3, 6$. The packet error rates for the Turbo-QAM at the same data rate are also plotted.

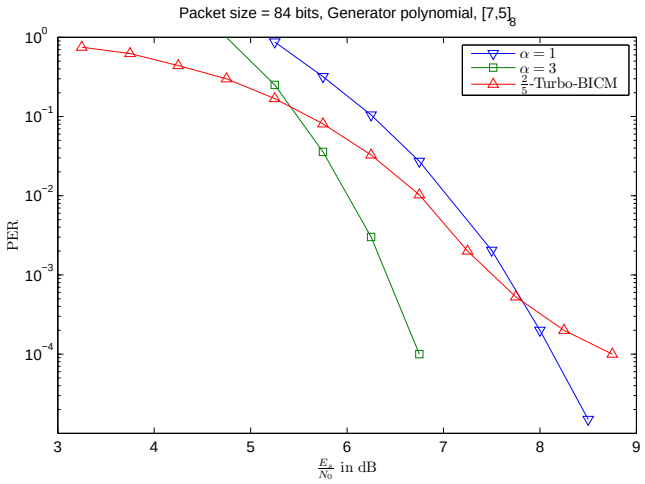

Figure 4.11: The packet error rates for the proposed CPP scheme at $b_p = 1.6$ bits per symbol, $V_o = 2$, $\varepsilon_o = 3$, with varying adjustment parameter values $\alpha = 1, 3$. The packet error rates for the Turbo-BICM QAM at the same data rate are also plotted.

better than that of the Turbo-QAM system.

Figure 4.11 shows another plot of packet error rates but at a different data rate of 1.6 bits per symbol, which was realized by a Turbo-bit interleaved coded modulation (BICM) scheme. The gap value achieved by this scheme is about 0.4 dB larger than that of the adaptive coded parity packet transmission model configured at 1.78 bits per symbol with an SNR variance of $\sigma_{\varepsilon}^2 = 5.20$ dB, which has been analyzed previously when discussing Figure 4.9. When the SNR variance is reduced, the gap value achieved by the proposed adaptive coded parity packet transmission model is further reduced.

Based on these results, it is observed that the adaptive coded parity packet transmission with the fixed energy loading offers a relatively good solution to maintain a relatively low gap value with a simple energy loading technique without requiring accurate information of the channel side information. This finding is useful for providing a computationally efficient transmission model as an alternative for practical systems.

4.6 Conclusion

By using the widely applied multi-code HSDPA system model [1], this chapter gives an accout of an SIC-based two-group resource allocation scheme, which significantly reduces the computational complexity to calculate the transmit energy for each code channel. The computational complexity of the existing two-group resource allocation scheme is reduced by eliminating matrix inversions in the energy calculation process, which is developed based on the SIC method. Without any matrix inversions, this modified two-group resource allocation scheme is proposed as an alternative practical method to further enhance performance of the well-established HSDPA system. This SIC-based two-group resource allocation scheme may also be incorporated into the MIMO HSDPA systems, which are being actively studied in the research community towards

the 3GPP Long Term Evolution (LTE) standard. Instead of assuming a perfect or accurate knowledge of the channel side information, a simple fixed energy loading which requires no channel side information has also been presented in this chapter for possible implementation in the current HSDPA systems to furthe enhance system performance. This fixed energy loading method is incorporated into an adaptive coded parity packet transmission model, which is implemented from the CPP scheme. When compared to the existing transmission models, this proposed adaptive coded parity packet transmission model has been observed to maintain a relatively low gap value at the presence of varying SNRs across the channels.

CHAPTER 5

Conclusion and Future Direction

The increasing demand for high speed communications has attracted a huge interest from the research community as well as industry. One of the most widely adopted and actively evolving high speed transmission systems is multi-code HSDPA [1], which is currently geared towards the implementation of the state-of-the art LTE system. Practical measurement-based performance evaluation carried out in [2] shows that the currently achievable system throughput by practical systems designed for MIMO HSDPA [1], which is actively studied towards the LTE standard, is still far below the upper bound. In order to offer solutions for bridging the gap between the theoretical upper bound and the practical achievable system throughput, a rigorous study on various aspects of multi-code HSDPA systems has been carried out in this thesis. Therefore, the contributions in this thesis is timely and relevant as an alternative to improve the practically achievable system throughput. These aspects include channel coding, resource allocation and interference cancellation designs. Since the number of matrix inversions has been identified as a serious problem to solve, the SIC-based two-group resource allocation scheme with the proposed channel coding technique are provided in this thesis as alternatives which might be considered as solutions for the current multi-code HSDPA systems to improve the achievable total data rate, hence the system throughput, whilst reducing the

computational complexity. With a reduced computational complexity, future incorporation into the emerging broadband technologies such as MIMO HSDPA and LTE may be considered and feasible.

In Chapter 2, a single-link transmission system is considered, where a packet-based channel coding technique is proposed to produce distinct discrete data rates with relatively low gap values. Although a single-link model is considered, the multiple distinct discrete data rates produced by the proposed channel coding scheme with low gap values is beneficial for multi-code transmission systems, such as the multi-code HSDPA system, which employs AMC to adjust the data rates according to the quality of the transmission channel.

The proposed channel coding method, which is referred to as coded parity packet scheme, also offers an alternative mechanism for controlling the gap value without changing the effective data rate. With a simple adjustment to the number of parity packets at the same data rate, the decoding performance is improved by providing a larger number of alternative additional input information values. The performance improvement is evaluated by measuring the reduction in the gap value. When compared to that of the existing schemes, the gap value achieved by this proposed coding schemes has been observed to be relatively lower.

In Chapter 3, the coded parity packet scheme is incorporated into a two-group resource allocation scheme, which is designed for multi-code HSDPA systems. This two-group resource allocation scheme loads the code channels with two adjacent data rates in order to eliminate the wasted SNR problem and maximize the utilization of the total constrained energy in order to improve the total data rate, hence the system throughput.

When this proposed loading scheme is implemented using a relatively small set of data rates, which are provided by the coded parity packet scheme, with large bit granularities, the resulted total data rate has been observed to be

approximately equal to that of the equal rate margin adaptive loading scheme, which is implemented using data rates with small bit granularities. Therefore, this proposed two-group resource allocation scheme is an alternative method to improve the total transmission data rate for multi-code HSDPA systems. Furthermore, the number of realizable total data rates is increased significantly when the same set of data rates, which are realizable in each code channel, is provided to this modified two-group resource allocation scheme.

The two-group resource allocation scheme is also observed to have a gain of approximately 5 dB over the existing equal rate and equal energy loading scheme. Therefore, this proposed loading scheme may be considered to be implemented in multi-code HSDPA systems in order to further improve performance of these widely implemented multi-code transmission systems. The two-group resource allocation scheme, however, requires an iterative energy calculation method to determine the energy, hence the data rate, that should be allocated to each code channel.

To solve the computational complexity problem, Chapter 4 presents an interference-reduced energy calculation method to eliminate inverse matrix operations in calculating the energy using the two-group resource allocation scheme. Besides requiring no matrix inversions, this scheme applies a computationally-efficient calculation method to calculate the energy.

Apart from this interference-reduced two-group resource allocation scheme, another resource allocation scheme which combines a simple fixed energy loading with an adaptive coded parity packet transmission, which is realized by the coded parity packet scheme, has been proposed. This scheme requires no channel side information to calculate the energy at the transmitter end as the energy is loaded equally to each code channel. By measuring and taking the mean of the SNRs over all code channels to determine the equal data rate for each channel, the effective gap value achieved by this scheme is observed to be slightly

smaller than that of the existing schemes.

5.1 Future Work

In the future, the proposed schemes may be further developed by considering the arrival of new technologies such as the MIMO HSDPA systems. The two-group resource allocation scheme, which requires iterative calculations to determine the total data rate as well as the energy, may be further simplified to design a closed-form rate calculation method which requires no iterative calculations. A background study with detailed mathematical footings on this specific research area has been carried out in [75], which has also discovered an upper bound on the sum capacity for a given channel impulse response with an optimum signature sequence matrix and equal energy loading. This ongoing research work aims to further improve performance of the high-speed MIMO HSDPA systems. By implementing this new approach, the constrained optimization method applied for the existing two-group resource allocation scheme can be eliminated as a closed-form rate calculation method has been designed to determine the total data rate of the two-group resource allocation scheme.

In order to further enhance this two-group loading scheme, the SIC-based energy calculation method proposed in Chapter 4 may also be integrated with the simplified rate calculation method described in the previous paragraph to further reduce the computational complexity. As the simplified rate calculation method reduces the number of energy calculation iterations, the SIC-based energy calculation method eliminates matrix inversions, which are previously required in the two-group resource allocation scheme. This SIC-based energy calculation method, which is originally developed in [93], as well as the modified two-group resource allocation scheme with the simplified rate calculation method, are proposed to further enhance and improve performance of the MIMO HSDPA systems towards the implementation of LTE systems.

APPENDIX A

The Normalized MMSE Despreading Filter Coefficient Vector

The normalized MMSE despreading filter coefficient vector, $\vec{w}_k$, which is a complex-valued column vector $\vec{w}_k \in \mathbb{C}^{N+L-1}$, is produced as follows:

$$\begin{aligned} \vec{w}_k &= \arg \min_{\vec{w} \in \mathbb{C}^{N+L-1}} \mathrm{E}\left\{ \left| \sqrt{E_k}|h|\vec{x}_k - \vec{w}^H \mathbf{R} \right|^2 \right\}, & \text{(A.1)} \\ &= \arg \min_{\vec{w} \in \mathbb{C}^{N+L-1}} \vec{w}^H \mathrm{E}\left\{ \mathbf{R}\mathbf{R}^H \right\} \vec{w} - 2\sqrt{E_k}|h|\vec{w}^H \mathrm{E}\{\mathbf{R}\vec{x}_k^H\}, & \text{(A.2)} \end{aligned}$$

As $\mathbf{C} = \mathrm{E}\left\{\mathbf{R}\mathbf{R}^H\right\}$ and $\sqrt{E_k}\vec{q}_k = \mathrm{E}\{\mathbf{R}\vec{x}_k^H\}$, the above minimum mean square error equation is then solved as follows (by substituting $\vec{w}_k = \vec{w}$)

$$\begin{aligned} 0 &= \mathbf{C}\vec{w}_k - 2E_k|h|\vec{q}_k, & \text{(A.3)} \\ \vec{w}_k &= 2E_k|h|\mathbf{C}^{-1}\vec{q}_k, & \text{(A.4)} \end{aligned}$$

from which the normalized MMSE despreading filter coefficient vector is produced such that $\vec{w}_k^H \vec{q}_k = 1$ as follows:

$$\vec{w}_k = \frac{\mathbf{C}^{-1}\vec{q}_k}{\vec{q}_k^H \mathbf{C}^{-1}\vec{q}_k} \quad \text{(A.5)}$$

APPENDIX B

The SIC-based Energy Equation Derivation

In order to proof the energy equation given in (4.18), the matrix inversion lemma may be used on the inverse matrix $\mathbf{D}_{2,k}^{-1} = \left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}$. For notational convenience, this inverse matrix may be denoted as $\mathbf{D}_{2,k}^{-1}$ and rewritten as follows:

$$\begin{aligned} \mathbf{D}_{2,k}^{-1} &= \left(\mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}, & \text{(B.1)} \\ &= \left(\mathbf{D}_{1,K} + E_k(y_k)\vec{q}_{k,2}\vec{q}_{k,2}^H\right)^{-1}, & \text{(B.2)} \end{aligned}$$

where $\mathbf{D}_{1,k} = \mathbf{C}_{k-1} + E_k(y_k)\vec{q}_{k,1}\vec{q}_{k,1}^H$.

With the matrix inversion lemma,

$$\mathbf{B}^{-1} = \left(\mathbf{C} + \mathbf{DED}^H\right)^{-1} = \mathbf{C}^{-1} - \mathbf{C}^{-1}\mathbf{D}\left(\mathbf{E}^{-1} + \mathbf{D}^H\mathbf{C}^{-1}\mathbf{D}\right)^{-1}\mathbf{D}^H\mathbf{C}^{-1},$$

the inverse matrix $\mathbf{D}_{2,k}^{-1}$, which is expressed in Eq(B.2), may be written as follows:

$$\begin{aligned}\mathbf{D}_{2,k}^{-1} &= \mathbf{D}_{1,k}^{-1} - \mathbf{D}_{1,k}^{-1}\vec{q}_{k,2}\left(E_k(y_k)^{-1} + \vec{q}_{k,2}^{H}\mathbf{D}_{1,k}^{-1}\vec{q}_{k,2}\right)^{-1}\vec{q}_{k,2}^{H}\mathbf{D}_{1,k}^{-1}, \\ &= \mathbf{D}_{1,k}^{-1} - \frac{E_k(y_k)\mathbf{D}_{1,k}^{-1}\vec{q}_{k,2}\vec{q}_{k,2}^{H}\mathbf{D}_{1,k}^{-1}}{1 + E_k(y_k)\vec{q}_{k,2}^{H}\mathbf{D}_{1,k}^{-1}\vec{q}_{k,2}}. \end{aligned} \tag{B.3}$$

As seen in Eq(B.3) above, there is another inverse matrix $\mathbf{D}_{1,k}^{-1}$, which may also be rewritten using the matrix inversion lemma as follows

$$\mathbf{D}_{1,k}^{-1} = \mathbf{C}_{k-1}^{-1} - \frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}\vec{q}_{k,1}^{H}\mathbf{C}_{k-1}^{-1}}{1 + E_k(y_k)\vec{q}_{k,1}^{H}\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}}, \tag{B.4}$$

Assuming that

$$\begin{aligned}\vec{q}_{k,2}^{H}\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1} &= 0, \\ \vec{q}_{k,1}^{H}\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2} &= 0,\end{aligned}$$

the inverse matrix $\mathbf{D}_{2,k}^{-1}$ in Eq(B.3) may be solved with Eq(B.4) to yield

$$\mathbf{D}_{2,k}^{-1} = \mathbf{C}_{k-1}^{-1} - \frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}\vec{q}_{k,1}^{H}\mathbf{C}_{k-1}^{-1}}{1 + E_k(y_k)\vec{q}_{k,1}^{H}\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}} - \frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}\vec{q}_{k,2}^{H}\mathbf{C}_{k-1}^{-1}}{1 + E_k(y_k)\vec{q}_{k,2}^{H}\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}}. \tag{B.5}$$

This approximate inverse matrix $\mathbf{D}_{2,k}^{-1}$ is then inserted into Eq(4.15), before the energy equation is derived as follows:

$$
\begin{aligned}
E_k(y_k) &= \frac{\gamma_k(y_k)}{\vec{q}_k^H\left(\mathbf{C}_{k-1}^{-1}-\frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}\vec{q}_{k,1}^H\mathbf{C}_{k-1}^{-1}}{1+E_k(y_k)\vec{q}_{k,1}^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}}-\frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}\vec{q}_{k,2}^H\mathbf{C}_{k-1}^{-1}}{1+E_k(y_k)\vec{q}_{k,2}^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}}\right)\vec{q}_k}, \\
&= \frac{\gamma_k(y_k)}{\vec{q}_k^H\mathbf{C}_{k-1}^{-1}\vec{q}_k-E_k(y_k)\left(\frac{\vec{q}_k^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}\vec{q}_{k,1}^H\mathbf{C}_{k-1}^{-1}\vec{q}_k}{1+E_k(y_k)\vec{q}_{k,1}^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}}+\frac{\vec{q}_k^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}\vec{q}_{k,2}^H\mathbf{C}_{k-1}^{-1}\vec{q}_k}{1+E_k(y_k)\vec{q}_{k,2}^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}}\right)}, \\
&= \frac{\gamma_k(y_k)}{\xi-E_k(y_k)\left(\frac{|\xi_3|^2}{1+E_k(y_k)\xi_1}+\frac{|\xi_4|^2}{1+E_k(y_k)\xi_2}\right)}, \qquad \text{(B.6)}
\end{aligned}
$$

with ξ,ξ_1,ξ_2,ξ_3 and ξ_4 as the weighting factors defined in Eq(4.22), Eq(4.23), Eq(4.24), Eq(4.25) and Eq(4.26) respectively.

APPENDIX C

The SIC-based Energy Equation Derivation

With Eq(4.17) and Eq(4.27), the inverse matrix $\mathbf{C}_k^{-1}$ given in Eq(4.10) is written and simplified as follows:

$$\begin{aligned}
\mathbf{C}_k^{-1} &= \mathbf{D}_{2,k}^{-1} - \frac{E_k(y_k)\mathbf{D}_{2,k}^{-1}\vec{q}_k\vec{q}_k^H\mathbf{D}_{2,k}^{-1}}{1+E_k(y_k)\vec{q}_k^H\mathbf{D}_{2,k}^{-1}\vec{q}_k}, \\
&= \mathbf{D}_{2,k}^{-1} - \frac{E_k(y_k)\mathbf{D}_{2,k}^{-1}\vec{q}_k\vec{q}_k^H\mathbf{D}_{2,k}^{-1}}{1+\gamma_k^*(y_k)}, \qquad \text{(C.1)} \\
&= \mathbf{D}_{2,k}^{-1} - \zeta\mathbf{D}_{2,k}^{-1}\vec{q}_k\vec{q}_k^H\mathbf{D}_{2,k}^{-1}, \qquad \text{(C.2)}
\end{aligned}$$

It should be noted that the SNR in Eq(C.1) is set to be equal to the target SNR, $\gamma_k(y_k) = \gamma_k^*(y_k) = \Gamma\left(2^{y_k}-1\right)$, which is required for calculating the energy to achieve this target SNR.

With the definitions of the distance vectors $\vec{d}_1, \vec{d}_2$ as well as the matrix weighting fectors ζ_1 and ζ_2, which are given in Eq(4.19), Eq(4.20), Eq(4.21), Eq(4.28) and Eq(4.29) respectively, the inverse matrix $\mathbf{D}_{2,k}^{-1}$ in Eq(B.5) is reexpressed as follows:

$$\begin{aligned}\mathbf{D}_{2,k}^{-1} &= \mathbf{C}_{k-1}^{-1} - \frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}\vec{q}_{k,1}^H\mathbf{C}_{k-1}^{-1}}{1+E_k(y_k)\vec{q}_{k,1}^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,1}} - \frac{E_k(y_k)\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}\vec{q}_{k,2}^H\mathbf{C}_{k-1}^{-1}}{1+E_k(y_k)\vec{q}_{k,2}^H\mathbf{C}_{k-1}^{-1}\vec{q}_{k,2}}, \\ &= \mathbf{C}_{k-1}^{-1} - \zeta_1\vec{d}_1\vec{d}_1^H - \zeta_2\vec{d}_2\vec{d}_2^H, \end{aligned} \tag{C.3}$$

before it is substituted in Eq(C.2), rendering

$$\begin{aligned}\mathbf{C}_k^{-1} &= \mathbf{D}_{2,k}^{-1} - \zeta\mathbf{D}_{2,k}^{-1}\vec{q}_k\vec{q}_k^H\mathbf{D}_{2,k}^{-1}, \\ &= \mathbf{C}_{k-1}^{-1} - \zeta_1\vec{d}_1\vec{d}_1^H - \zeta_2\vec{d}_2\vec{d}_2^H \\ &\quad -\zeta\left(\mathbf{C}_{k-1}^{-1} - \zeta_1\vec{d}_1\vec{d}_1^H - \zeta_2\vec{d}_2\vec{d}_2^H\right)\vec{q}_k\vec{q}_k^H\left(\mathbf{C}_{k-1}^{-1} - \zeta_1\vec{d}_1\vec{d}_1^H - \zeta_2\vec{d}_2\vec{d}_2^H\right). \end{aligned} \tag{C.4}$$

With all definitions of distance vectors and weighting parameters, $\vec{d}, \vec{d}_1, \vec{d}_2$, ξ_1, ξ_2, ξ_3, ξ_4, ζ, ζ_1 and ζ_2 given in Eq(4.19), Eq(4.20), Eq(4.21), Eq(4.22), Eq(4.24), Eq(4.25), Eq(4.26), Eq(4.27), Eq(4.28) and Eq(4.29), this inverse matrix $\mathbf{C}_{k-1}^{-1}$ is further simplified as follows:

$$\begin{aligned}\mathbf{C}_k^{-1} &= \mathbf{C}_{k-1}^{-1} - \zeta\mathbf{C}_{k-1}^{-1}\vec{q}_k\vec{q}_k^H\mathbf{C}_{k-1}^{-1} - \zeta_1\vec{d}_1\vec{d}_1^H - \zeta\zeta_1^2\vec{d}_1\vec{d}_1^H\vec{q}_k\vec{q}_k^H\vec{d}_1\vec{d}_1^H \\ &\quad -\zeta_2\vec{d}_2\vec{d}_2^H - \zeta\zeta_2^2\vec{d}_2\vec{d}_2^H\vec{q}_k\vec{q}_k^H\vec{d}_2\vec{d}_2^H \\ &\quad +\zeta\zeta_1\mathbf{C}_{k-1}^{-1}\vec{q}_k\vec{q}_k^H\vec{d}_1\vec{d}_1^H + \zeta\zeta_1\vec{d}_1\vec{d}_1^H\vec{q}_k\vec{q}_k^H\mathbf{C}_{k-1}^{-1} \\ &\quad +\zeta\zeta_2\mathbf{C}_{k-1}^{-1}\vec{q}_k\vec{q}_k^H\vec{d}_2\vec{d}_2^H + \zeta\zeta_2\vec{d}_2\vec{d}_2^H\vec{q}_k\vec{q}_k^H\mathbf{C}_{k-1}^{-1} \\ &\quad -\zeta\zeta_1\zeta_2\vec{d}_1\vec{d}_1^H\vec{q}_k\vec{q}_k^H\vec{d}_2\vec{d}_2^H - \zeta\zeta_1\zeta_2\vec{d}_2\vec{d}_2^H\vec{q}_k\vec{q}_k^H\vec{d}_1\vec{d}_1^H, \\ &= \mathbf{C}_{k-1}^{-1} - \zeta\vec{d}\vec{d}^H - \left(\zeta_1 + \zeta\zeta_1^2|\xi_3|^2\right)\vec{d}_1\vec{d}_1^H - \left(\zeta_2 + \zeta\zeta_2^2|\xi_4|^2\right)\vec{d}_2\vec{d}_2^H \\ &\quad +\zeta\zeta_1\left(\xi_3\vec{d}\vec{d}_1^H + \xi_3^*\left(\vec{d}\vec{d}_1^H\right)^H\right) + \zeta\zeta_2\left(\xi_4\vec{d}\vec{d}_2^H + \xi_4^*\left(\vec{d}\vec{d}_2^H\right)^H\right) \\ &\quad -\zeta\zeta_1\zeta_2\left((\xi_3\xi_4^*)^*\left(\vec{d}_2\vec{d}_1^H\right)^H + \xi_3\xi_4^*\vec{d}_2\vec{d}_1^H\right). \end{aligned} \tag{C.5}$$

Bibliography

[1] *3GPP TS 25.214: Physical Layer Procedure (FDD)*, V10.1.0 ed., 3GPP, Dec. 2010.

[2] C. Mehlfuhrer, S. Caban, and M. Rupp, "Measurement-based performance evaluation of MIMO HSDPA," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 9, pp. 4354–4367, 2010.

[3] G. Forney Jr and G. Ungerboeck, "Modulation and coding for linear Gaussian channels," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2384–2415, 1998.

[4] C. Shannon, "A mathematical theory of communications," *Bell Systems Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[5] A. Burr, *Modulation and Coding for A Wireless Communications*. Prentice Hall, 2001.

[6] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes." in *Conference Record of IEEE International Conference on Communications, 1993. ICC 93. Geneva.*, vol. 2, 23-26 May 1993, pp. 1064–1070.

[7] R. Gallager, "Low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, Jan 1962.

[8] G. Forney, *Concatenated codes*. Citeseer, 1966.

[9] D. Divsalar, H. Jin, and R. McEliece, "Coding Theorems for" Turbo-Like" Codes," in *Proceedings of The Annual Allerton Conference on Communication Control and Computing*, vol. 36. University of Illinois, 1998, pp. 201–210.

[10] S. Le Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," *IEEE International Conference on Communications, 1994. ICC'94, SUPERCOMM/ICC'94, Conference Record, Serving Humanity Through Communications*, vol. 2, pp. 645–649, 1994.

[11] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[12] A. Banerjee, J. Costello, D.J., T. Fuja, and P. Massey, "Asymmetric Turbo-like Codes for Bit-interleaved Coded Modulation," in *39th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, October 2001, p. 508.

[13] G. Caire, R. Muller, and T. Tanaka, "Iterative multiuser joint decoding: optimal power allocation and low-complexity implementation," *IEEE Transactions on Information Theory*, vol. 50, no. 9, pp. 1950–1973, Sept. 2004.

[14] J. Luo, R. Liu, and P. Spasojevic, "Variable-rate Turbo bit-interleaved coded modulation," *International Symposium on Information Theory, 2004. ISIT 2004.*, pp. 507–, June-2 July 2004.

[15] W. K. Han, S. Le Goff, and B. Sharif, "Systematic Repeat Accumulate codes for bit-interleaved coded modulation with iterative demapping over AWGN and Rayleigh fading channels," *3rd International Symposium on Wireless Pervasive Computing, 2008. ISWPC 2008.*, pp. 152–155, May 2008.

[16] G. Ungerboeck and I. Csajka, "On improving data-link performance by increasing channel alphabet and introducing sequence coding," in *IEEE International Symposium on Information Theory (ISIT)*, Ronneby, Sweden, June 1976.

[17] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Transactions in Information Theory*, vol. IT-28, no. 1, pp. 55–67, January 1982.

[18] P. Robertson and T. Worz, "A novel bandwidth efficient coding scheme employing Turbo codes," *IEEE International Conference on Communications, 1996. ICC 96, Conference Record, Converging Technologies for Tomorrow's Applications,*, vol. 2, pp. 962–967 vol.2, Jun 1996.

[19] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Parallel concatenated trellis coded modulation," *IEEE International Conference on Communications, 1996. ICC 96, Conference Record, Converging Technologies for Tomorrow's Applications*, vol. 2, pp. 974–978 vol.2, Jun 1996.

[20] C. Fragouli and R. Wesel, "Turbo-encoder design for symbol-interleaved parallel concatenated trellis-coded modulation," *IEEE Transactions on Communications*, vol. 49, no. 3, pp. 425–435, Mar 2001.

[21] S. Le Goff, "Signal constellations for bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 49, no. 1, pp. 307–313, 2003.

[22] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 371–377, May 1977.

[23] ——, "Correction to 'a new multilevel coding method using error-correcting codes'," *IEEE Transactions on Information Theory*, vol. 23, no. 6, pp. 784–784, Nov 1977.

[24] J. U. Wachsman and J. Huber, "Power and Bandwidth Efficient Digital Communication Using Turbo Codes in Multilevel Codes," *European Transactions on Telecommunications*, vol. 6, no. 5, 1995.

[25] H. Jenkac, J. Hagenauer, and T. Mayer, "The Turbo-Fountain," *European Transactions On Telecommunications*, vol. 17, no. 3, pp. 337–349, 2006.

[26] L. Ping, W. Leung, and K. Wu, "Low-rate turbo-Hadamard codes," *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3213–3224, 2003.

[27] Y. WU and L. PING, "Convergence analysis of turbo-hadamard codes using the extrinsic information transfer (EXIT) chart technique," in *IEEE International Conference on Communications*, 2004.

[28] ——, "On the limiting performance of turbo-Hadamard codes," *IEEE communications letters*, vol. 8, no. 7, pp. 449–451, 2004.

[29] H. Ghani and M. Gurcan, "Rate multiplication and two-group resource allocation in multi-code CDMA networks," in *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2009, pp. 551–555.

[30] M. Gurcan and H. Ab Ghani, "Small-sized packet error rate reduction using coded parity packet approach," in *IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2010, pp. 419–424.

[31] M. Gurcan and H. A. Ghani, "Coded Parity Packet Approach for Small-sized Packet Transmission in Wireless Ad-hoc Networks," submitted to European Transactions on Telecommunications in December 2010, Under Review.

[32] M. S. Raju, R. Annavajjala, and A. Chockalingam, "BER Analysis of QAM on Fading Channels with Transmit Diversity," *IEEE Transactions on Wireless Communications*, vol. 5, no. 3, pp. 481–486, March 2006.

[33] M. A. Imran and M. K. Gurcan, "Message passing algorithm for iterative decoding of channel codes," *HERMIS-mu-pi International Journal of Computer Mathematics and its Applications*, vol. 8, pp. 1–13, 2006.

[34] H. V. Poor, *Wireless Communication Systems: Advanced Techniques for Signal Reception*, X. Wang, Ed. Prentice Hall, 2003.

[35] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 429–445, March 1996.

[36] S. Dolinar, D. Dilvsalar, and F. Pollara, "Code performance as a function of block size," TMO Progress Report 42-133, Tech. Rep., May 1998.

[37] B. Kim and H. Lee, "Reduction of the number of iterations in turbo decoding using extrinsic information," in *TENCON 99. Proceedings of the IEEE Region 10 Conference*, vol. 1. IEEE, 1999, pp. 494–497.

[38] H. Holma, A. Toskala, T. Kolding, P. Mogensen, K. Pedersen, and J. Reunanen, *WCDMA for UMTS - HSPA Evolution and LTE*, 4th ed., H. Holma and A. Toskala, Eds. John Wiley & Sons Ltd., April 2008.

[39] *3GPP TS 25.214: Physical Layer Procedure (FDD)*, V5.11.0 ed., 3GPP, June 2005.

[40] A. Mader, D. Staehle, and M. Spahn, "Impact of HSDPA Radio Resource Allocation Schemes on the System Performance of UMTS ," University of Wurzburg, Institute of Computer Science, Germany, Research Report 410, April 2007.

[41] R. Love, K. Stewart, R. Bachu, and A. Ghosh, "MMSE equalization for UMTS HSDPA," in *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, vol. 4, Oct. 2003, pp. 2416–2420 Vol.4.

[42] K. Hooli, M. Juntti, M. J. Heikkilä, P. Komulainen, M. Latva-aho, and J. Lilleberg, "Chip-level channel equalization in WCDMA downlink," *EURASIP J. Appl. Signal Process.*, vol. 2002, no. 1, pp. 757–770, 2002.

[43] L. Fathi, G. Jourdain, and M. Arndt, "Chip-level channel equalization with rake-like structures for multicode downlink wcdma communications," in *IEEE 63rd Vehicular Technology Conference, 2006. VTC 2006-Spring.*, vol. 5, May 2006, pp. 2344–2348.

[44] G. Bottomley, T. Ottosson, and Y.-P. Wang, "A generalized RAKE receiver for interference suppression," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 8, pp. 1536–1545, Aug 2000.

[45] L. Gao and T. Wong, "Sequence optimization in cdma point-to-point transmission with multipath," in *IEEE 56th Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall.*, vol. 4, 2002, pp. 2303–2307 vol.4.

[46] G. Rajappan and M. Honig, "Signature sequence adaptation for DS-CDMA with multipath," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 2, pp. 384–395, Feb 2002.

[47] J. Concha and S. Ulukus, "Optimization of CDMA signature sequences in multipath channels," in *IEEE VTS 53rd Vehicular Technology Conference, 2001. VTC 2001 Spring.*, vol. 3, 2001, pp. 1978–1982 vol.3.

[48] B. Sayadi, S. Ataman, and I. Fijalkow, "Joint downlink power control and multicode receivers for downlink transmissions in high speed UMTS," *EURASIP J. Wirel. Commun. Netw.*, vol. 2006, no. 2, pp. 60–60, 2006.

[49] X. Qiu and K. Chawla, "On the performance of adaptive modulation in cellular systems," *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 884–895, Jun 1999.

[50] C. Chai, T. T. Tjhung, and L. C. Leck, "Combined power and rate adaptation for wireless cellular systems," *IEEE Transactions on Wireless Communications*, vol. 4, no. 1, pp. 6–13, Jan. 2005.

[51] X. Zhang and B. Ottersten, "Power allocation and bit loading for spatial multiplexing in MIMO systems," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, vol. 5, April 2003, pp. V–53–6 vol.5.

[52] S. Ulukus, E. Biglieri, and M. Z. Win, "Optimum modulation and multicode formats in CDMA systems with multiuser receivers," in *IEEE INFOCOM 2001*, 2001.

[53] D. K. Tureli, "Resource Allocation for Multicarrier Communications," Doctor of Philosophy, University of Washington, USA, 2005.

[54] H. Levin, "A complete and optimal data allocation method for practical discrete multitone systems," in *IEEE Global Telecommunications Conference, 2001. GLOBECOM '01.*, vol. 1, 2001, pp. 369–374 vol.1.

[55] J. Campello, "Optimal discrete bit loading for multicarrier modulation systems," in *IEEE International Symposium on Information Theory*, Aug 1998, pp. 193–.

[56] ——, "Practical bit loading for DMT," in *IEEE International Conference on Communications, 1999. ICC '99.*, vol. 2, 1999, pp. 801–805 vol.2.

[57] *A Tutorial on Integer Programming*, The Operations Research Faculty of GSIA.

[58] R. Kwan, M. Aydin, C. Leung, and J. Zhang, "Multiuser Scheduling in HSDPA using Simulated Annealing," in *International Wireless Communications and Mobile Computing Conference, 2008. IWCMC '08.*, Aug. 2008, pp. 236–241.

[59] R. Kwan and C. Leung, "Downlink scheduling optimization in CDMA networks," *IEEE Communications Letters*, vol. 8, no. 10, pp. 611–613, Oct. 2004.

[60] D. I. Kim, E. Hossain, and V. Bhargava, "Dynamic rate and power adaptation for forward link transmission using high-order modulation and multicode formats in cellular WCDMA networks," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2361 – 2372, sept. 2005.

[61] P. Bjorklund, P. Varbrand, and D. Yuan, "A dynamic programming technique for downlink bandwidth allocation in WCDMA networks," in *IEEE 59th Vehicular Technology Conference, 2004. VTC 2004-Spring.*, vol. 4, May 2004, pp. 2007–2011 Vol.4.

[62] K. Papadaki and V. Friderikos, "Resource management in CDMA networks based on approximate dynamic programming," in *The 14th IEEE Workshop on Local and Metropolitan Area Networks, 2005. LANMAN 2005.*, Sept. 2005, pp. 6 pp.–6.

[63] H. Sampath and A. Paulraj, "Joint transmit and receive optimization for high data rate wireless communication using multiple antennas," in *Conference Record of the Thirty-Third Asilomar Conference on Signals, Systems, and Computers,*, vol. 1, 1999, pp. 215 –219 vol.1.

[64] K. Dawui and D. Slock, "Multiuser-mimo downlink tx-rx design based on svd channel diagonalization and multiuser diversity," in *Signals, Systems and Computers, 2005. Conference Record of the Thirty-Ninth Asilomar Conference on*, october 2005, pp. 1493 – 1497.

[65] W. Liu, L. Yang, and L. Hanzo, "Svd-assisted multiuser transmitter and multiuser detector design for mimo systems," *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 2, pp. 1016 –1021, feb. 2009.

[66] Z. He and M. Gurcan, "The rate adaptive throughput maximization in PAM-modulated overloaded system," in *IEEE Wireless Communications and Networking Conference, 2009. WCNC 2009.*, April 2009, pp. 1–6.

[67] Z. He and M. K. Gurcan, "Optimizing radio resource allocation in HSDPA using 2 group allocation," in *IWCMC '09: Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing*. New York, NY, USA: ACM, 2009, pp. 1107–1111.

[68] Z. He and M. Gurcan, "Optimized resource allocation of HSDPA using two group allocation in frequency selective channels," in *IEEE WCSP 2009*, Nanjing, China, November 2009.

[69] R. Kwan and C. Leung, "Downlink Scheduling Schemes for CDMA Networks with Adaptive Modulation and Coding and Multicodes," *IEEE Transactions on Wireless Communications*, vol. 6, no. 10, pp. 3668–3677, October 2007.

[70] G. Aniba and S. Aissa, "Resource allocation in hsdpa using best-users selection under code constraints," in *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, vol. 1, May-1 June 2005, pp. 319–323 Vol. 1.

[71] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, no. 9, pp. 59–67, September 2008.

[72] T. Nihtila, "Increasing Femto Cell Throughput with HSDPA Using Higher Order Modulation," in *IEEE International Networking and Communications Conference, 2008. INCC 2008.*, May 2008, pp. 49–53.

[73] J. W. P. Ng and A. Manikas, "A Space-Diffused Spatial-Temporal ARray (STAR) Receiver for DS-CDMA System," *International Journal HERMIS-mu-pi(Mathematics and Information Science)*, vol. 4, pp. 131–152, 2003.

[74] Z. He, M. Gurcan, and H. Ghani, "Time-efficient resource allocation algorithm over HSDPA in femtocell networks," in *IEEE 21st International Symposium on Personal, Indoor and Mobile Radio Communications Workshops (PIMRC Workshops)*. IEEE, 2010, pp. 197–202.

[75] M. K. Gurcan, *Gurcan's system values and Upper bounds for total system values and sum capacity for K multicode parallel channel SISO and MIMO systems*, Imperial College London, June 8 2011.

[76] A. Ghani, M. Gurcan, and Z. He, "Two-Group Resource Allocation with Channel Ordering and Interference Cancellation," in *IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2010, pp. 1–6.

[77] J. Cioffi, "Course reader for digital communication,," 2005. [Online]. Available: http://www.stanford.edu/group/cioffi/

[78] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, no. 9, pp. 59–67, 2008.

[79] T. Nihtila, "Increasing femtocell throughput with HSDPA using higher order modulation," in *IEEE International Networking and Communications Conference, 2008. INCC 2008*, 2008, pp. 49–53.

[80] Z. He and M. Gurcan, "The rate adaptive throughput maximization in pam-modulated overloaded system," in *IEEE Wireless Communications and Networking Conference, 2009, WCNC 2009*, April 2009, pp. 1–6.

[81] ——, "Optimizing Radio Resource Allocation in HSDPA Using 2 Group Allocation," in *The 5th International Wireless Communications and Mobile Computing Conference*, Leipzig, Germany, April 2009.

[82] ——, "Optimized Resource Allocation of HSDPA Using Two-Group Allocation in Frequency Selective Channel," in *IEEE WCSP Conference*, Nanjing, China, November 2009.

[83] B. Sayadi, S. Ataman, and I. Fijalkow, "Joint downlink power control and multicode receivers for downlink transmissions in high speed UMTS," *EURASIP Journal on Wireless Communications and Networking*, vol. 2006, no. 2, p. 60, 2006.

[84] J. Chen, J. Wang, and M. Sawahashi, "MCI cancellation for multicode wideband CDMA systems," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 2, pp. 450–462, 2002.

[85] J. M. Holtzman, "Successive Interference Cancellation for Direct Sequence Code Division Multiple Access," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 697–793, October 1994.

[86] P. Patel and J. Holtzman, "Analysis of a simple successive interference cancellation scheme ina DS/CDMA system," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 5, pp. 796–807, 1994.

[87] X. Chen, D. Xu, and X. Yu, "Adaptive Transmit Antenna Selection and Power Allocation Scheme for Turbo-Blast System with Imperfect Channel State Information," *Progress In Electromagnetics Research*, vol. 10, pp. 215–230, 2009.

[88] R. Buehrer, "Equal BER performance in linear successive interference cancellation for CDMA systems," *IEEE Transactions on Communications*, vol. 49, no. 7, pp. 1250–1258, 2001.

[89] M. Tan and Y. Bar-Ness, "Equal BER power control for uplink MC-CDMA with MMSE successive interference cancellation," *IEEE Communications Letters*, vol. 8, no. 6, pp. 348–350, 2004.

[90] H. Huang, H. Viswanathan, and G. Foschini, "Multiple antennas in cellular CDMA systems: transmission, detection, and spectral efficiency," *IEEE Transactions on Wireless Communications*, vol. 1, no. 3, pp. 383–392, 2002.

[91] C. Park and K. Lee, "Transmit power allocation for successive interference cancellation in multicode MIMO systems," *IEEE Transactions on Communications*, vol. 56, no. 12, pp. 2200–2213, 2008.

[92] P. Wolniansky, G. Foschini, G. Golden, and R. Valenzuela, "V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *URSI International Symposium on Signals, Systems, and Electronics. ISSSE 98.* Ieee, 1998, pp. 295–300.

[93] M. Gurcan, *Iterative energy and inverse covariance matrix calculation for sucessive inteference calculation for WCDMA HSDPA systems*, Imperial College London, April 2011.

[94] M. K. Gurcan, *Mobile Radio Communications.* Department of Electrical and Electronic Engineering, Imperial College London, 2011.