Convergence Study of Iterative Multi-user Detection for Turbo-coded CDMA

Behrooz Hamidian

A Thesis

In

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Applied Science(Electrical Engineering) at
Concordia University
Montreal, Quebec, Canada

February 2006

Your file Votre référence
ISBN: 0-494-14260-X
Our file Notre référence
ISBN: 0-494-14260-X

ABSTRACT

Convergence Study of Iterative Multi-user Detection for Turbo-coded CDMA

Behrooz Hamidian

Convergence study is a crucial issue for any iterative structure. In fact, it focuses on the main feature of such structures which is iteration, and traces the behavior of the system through iterations. Iterative Multi-user Detection (IMUD) turbo-coded CDMA is an iterative structure that includes another embedded iterative structure, which is the turbo decoder. This makes the convergence study of such a system more interesting.

In this thesis, we mainly focus on the convergence of IMUD turbo-coded systems. The main interest is the possibility that the system converges to the near-single-user performance and the second important concern is the speed of the successful convergence.

The convergence thresholds of IMUD turbo-coded CDMA systems have been studied in [11] using conventional BER curves. However, BER curves do not have the ability to sufficiently describe the convergence behavior of iterative structures. Therefore in this thesis, we use the Variance Exchange Graph (VEG) as the methodology of analysis. VEG is a method that uses variance error of soft symbols as a measure to characterize the components in the IMUD structure. VEGs surpass BER curves for the purpose of

convergence study by providing the possibility to analyze the system convergence from different angels. They also present a more detailed description of the iterative process comparing with BER curves.

In this thesis, convergence thresholds and convergence speed of the proposed system are analyzed. Guidelines are proposed based on VEG methodology to design the system that successfully convergences to near-single-user performance. A novel technique, called smart iteration schedule, is also presented that speeds up the convergence process significantly.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor Dr. Yousef R. Shayan for his enthusiastic supervision, support and assistance in developing this thesis. He has been an invaluable source of knowledge and his helpful comments have helped to inspire many of the ideas in this thesis.

Also, I would like to thank Dr. Xiaofeng Wang for promptly answering my questions and for providing useful information regarding my thesis. I would also like to thank the rest of the faculty members of the Department of Electrical and Computer Engineering at Concordia University. I also appreciate all my friends who helped me and encouraged me throughout my studies, especially Mr. Robert Zadeh.

Finally, I am forever indebted to my parents for their understanding, endless encouragement when it was most required.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AWGN .... Additive White Gaussian Noise

BER .... Bit Error Rate

CDMA .... Code Division Multiple Access

DSSS .... Direct Sequence Spread Spectrum

EXIT .... Extrinsic Information Transfer

FDMA .... Frequency Division Multiple Access

FEC .... Forward Error Control

IMUD .... Iterative Multi-User Detection

LLR .... Logarithm Likelihood Ratio

MAI .... Multiple Access Interference

MAP .... Maximum A Posteriori

MUD .... Multi-User Detection

PCCC .... Parallel Concatenated Convolutional Codes

PIC .... Parallel Interference Canceller

PPIC .... Partial Parallel Interference Canceller

SCCC .... Serial Concatenated Convolutional Codes

SIC .... Serial Interference Canceller

SISO .... Soft-Input Soft-Output

SNR .... Signal to Noise Ratio

SOVA .... Soft Output Viterbi Algorithm

TDMA .... Time Division Multiple Access

VEG .... Variance Exchange Graph

VT..Variance Transfer

1 Introduction

The channel in a communication system is typically used by more than a single user. There are various ways in which multiple users can access a shared channel to transmit information in an efficient and reliable manner. The three main methods include Frequency-Division Multiple Access (FDMA), Time-Division Multiple Access (TDMA), and Code-Division Multiple Access (CDMA), among which there are many variations and combinations possible.

In FDMA systems, each user is assigned a different carrier frequency, so the whole bandwidth consists of non overlapping slots of frequency and each slot is assigned to one user. Therefore in this method, users are separated from each other in frequency domain. For TDMA systems, time is partitioned into slots with each user being assigned to a particular slot. So in this method, users are separated from each other in time domain. The main drawback of FDMA and TDMA is that they have hard capacity limit. This means that if the number of users equals to the number of frequency or time slots, adding one more user will be impossible. This is due to the fact that all the frequency and time slots have already been allocated to existing users.

A CDMA system incorporates spread spectrum techniques to allow each user access the entire spectrum, making better use of the limited bandwidth and time resources available to the entire system. In this method, users are separated from each other in code domain and one unique codeword is assigned to each user. The capacity of CDMA system has a soft limit in the sense that we can add one additional user and tolerate a slight degradation in signal quality. CDMA systems are grouped to different categories depending on spreading techniques, such as: Direct-Sequence (DS), Frequency-Hopping (FH) and Time-Hopping (TH). Most of recent research has focused on DS-CDMA and this thesis will only focus on DS-CDMA. Thus, CDMA will refer to DS-CDMA throughout this thesis. An overview of more general spread spectrum techniques can be found in [1], [2],[3] and [4].

In multi-user environment, for each user, the interference of other users is an unwanted phenomena called Multiple Access Interference (MAI). MAI is a factor which degrades the performance of CDMA systems. Conventional receivers do not take the existence of MAI into account and follow a single-user detection strategy in which each user is detected separately, irrespective of other users. Obviously, when communication channel is a multi-user environment, single-user approaches cannot be the best choice and have to be modified. A proper detection strategy for multi-user systems is multi-user detection in which the information of all users are used jointly to better detect each individual user. The optimum multi-user detector or maximum likelihood sequence detector was presented by Verdu in 1986 [5]. Unfortunately, this detector is much too complex to implement for practical purposes. Therefore since then, most of the research has focused on finding sub-optimal multi-user detectors which are more proper to implement. Recent research has focused on designing efficient multi-user receivers which operate in conjunction with forward error control (FEC) coding. These sub-optimal receivers are all arranged in an iterative structure, i.e. the multi-user detector and a bank of decoders form a closed loop where data values are updated iteratively. The ideal scenario for iterative multi-user detection is that the system constantly makes improvement through iterations until the MAI is totally removed and the system approaches the single-user performance. Iterative multi-user detection schemes can approach single-user performance if the components in the iterative loop are well-matched to each other. It is important to investigate whether or not an iterative multi-user detection is able to approach the single-user performance throughout iterations. This question directs us to "convergence behavior study" which is a fundamental issue in any iterative architecture. It gives us a deep insight of the iterative detection process from the beginning to the end, and the interaction between the components in the iteration loop can be clearly observed.

Mainly, there are two methodologies to investigate the convergence behavior of an iterative multi-user detection scheme: EXtrinsic Information Transfer (EXIT) chart and Variance Exchange Graph (VEG). These two analysis methods are similar in some aspects and different in others. They both consist of two constituent characteristics, each of which characterizes one of the components in the iteration loop. EXIT charts visualize the convergence behavior of iterative systems using mutual information that are exchanged between components in the receiver side [8]. On the other hand, VEGs analyze the system convergence using the evolution of the error variances of bits. VEGs have two main advantages that EXIT charts lack. Firstly, it is much easier to connect the system parameters to error variances rather than to mutual information. Secondly, the behavior of widely used multi-user detectors can be analytically described from the error variance transfer perspective. This brings us an important advantage for the system investigation. The analytical formulation allows us to avoid simulating the multi-user detector each time it is called on. Therefore, the VEG can be considered as a semi-analytical methodology for the convergence study of IMUD systems. It enables us to predict the behavior of the system in terms of the parameters that exist in the obtained formula. It also simplifies the analysis of iterative detection process to a great extent.

Based on the two mentioned advantages, VEGs show us how well an iterative multi-user detection architecture can converge to the single-user performance. Hence, we use the VEG as an analysis method instead of the EXIT chart.

1.1 Literature Review

A few papers have addressed the convergence behavior of iterative multi-user detection systems so far. The iterative loop consists of a multi-user detector, a bank of single-user independent SISO decoders with interleavers (π) and deinterleavers (π^{-1}) between them. The summary of the iterative systems which convergence behaviors have been investigated is shown in table 1. In this table, the main features of the studied systems are also mentioned.

Table 1: The summary of previous research on the convergence of IMUD schemes

Ref.	Methodology	MUD	Channel Coding	Synchronization
[6]	EXIT Chart	MMSE	MAP Convolutional Coding	Synchronous
[7]	EXIT Chart	PIC	MAP Convolutional Coding	Synchronous & Asynchronous
		PIC & MMSE		
[8]	EXIT Chart	PIC	MAP Convolutional Coding	Synchronous
		Optimum Joint Detection		
[9]	VEG	PIC	MAP Convolutional Decoder	Synchronous
[10]	VEG	PIC	Serial Concatenated Convolutional Codes	Asynchronous
[11]	BER curves	PIC	Turbo codes	Asynchronous
This thesis	VEG	PIC	Turbo codes	Asynchronous

All the iterative multi-user receivers investigated in the literature have a general structure as depicted in Figure 1. As Figure 1 shows, most of the literatures have used MAP convolutional decoding as channel coding; PIC and MMSE are the multi-user detectors that are mainly used in IMUD architectures because of their compatibility for iterative structures.

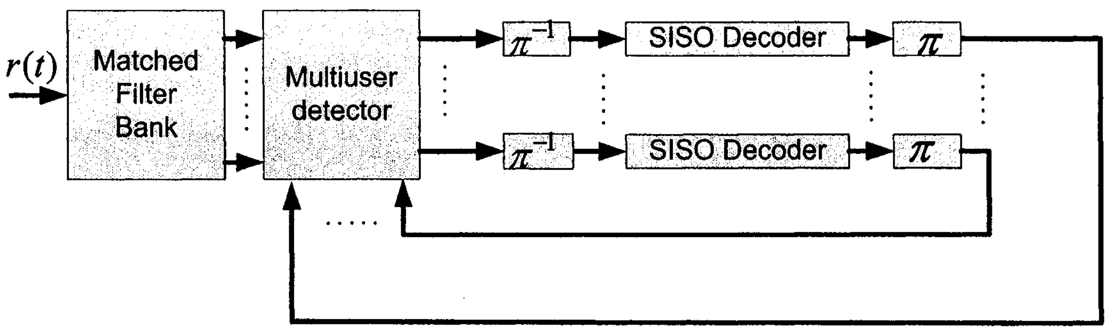

Figure 1: Typical structure of an iterative multi-user detector

Among the mentioned literature in Table 1, [7] is a massive research that has been done on the convergence of IMUD in AWGN and multi-path fading channel by EXIT charts. It compares the performance of PIC-based with PIC & MMSE-based convolutional-coded IMUD systems. The impacts of different system parameters are studied, paying special attention to the frame length.

This thesis can be considered similar to [10], except that we use PCCC (Turbo codes) instead of SCCC. To make our analysis more comprehensive, we include the influence of frame length in this thesis similar to work in [7]. This is because of the fact that the frame length determines the size of the embedded interleaver in the turbo decoders and the performance of turbo decoding is highly affected by it [12].

The studied system in this thesis is the same as in [11]. However in [11], BER curves are used to analyze the convergence of the system, which are not as efficient as VEGs. For instance, it is very complicated to connect the system parameters to BER curves because it requires dealing with Q-Function. In addition, BER curves are only able to present the performance of the whole system as one block, whereas as VEGs not only visualize the convergence of the whole system, but also present the dynamics inside the iterative structure; and the gain of each component inside the iterative loop is individually apparent in VEGs.

In this thesis, the turbo coders and decoders parameters are kept constant, except for frame length. However, other parameters of the system such as the number of users and signal to noise ratio are varied to analyze their influence on the system convergence. Then, we find thresholds for the possibility of the system convergence in terms of these parameters.

Another aspect of our interesting in this thesis is the convergence speed. This term refers to the number of iterations needed to reach the near-single-user performance. The impact of the mentioned parameters on the convergence speed can also be predicted and investigated efficiently by VEGs.

Another parameter of interest is the number of iterations inside the turbo decoders. To avoid terminology problem we call this “inner iteration”. The tradeoff between the inner and outer iterations brings us to a subject called “iteration schedule”, which will be investigated using VEGs.

1.2 Motivations

Due to the nature of iterative multi-user detection, convergence study shows the compatibility of components in the iterative loop to one another. Hence in any IMUD system, convergence analysis is crucial and has to be considered to obtain guidelines to design.

The IMUD turbo-coded CDMA system is an iterative structure that contains turbo decoders as another embedded iterative structure. This makes the convergence study of such a system more interesting. An extensive research about the convergence behavior of such a system has not been done yet, that motivates us to deeply explore in this subject. It is necessary to trace the system behavior in each iteration to figure out what happens in the whole process. In addition, the system has lots of different parameters on which its convergence depends. The best method to investigate the convergence behavior of iterative systems is a graph that visualizes the whole iterative process, and on which the impacts of the system parameters are apparent.

1.3 Objectives

The objective of this research is to investigate the convergence behavior of IMUD turbo-coded systems. The four questions of interest throughout this thesis are as follows:

1. On what conditions, in terms of system parameters, an iterative multi-user turbo coded CDMA system can approach to the single-user performance?

2. In case of convergence, how fast does the system approach to the convergence point?
3. What is the optimum iteration pattern between outer iterations and inner iterations?
4. What is the difference between IMUD-SCCC systems and IMUD-PCCC systems from the convergence point of view?

1.4 Contributions

The main contribution of this thesis can be summarized as follows:

- Extensive analysis of convergence behavior of IMUD turbo-coded CDMA systems for the first time to the best knowledge of the author,
- Using VEG to predict the behavior of the proposed system for the first time,
- Partitioning the IMUD structure in a way that VEG can be used to predict the system behavior,
- Analysis of convergence thresholds of the proposed system from an error variance viewpoint,
- Analysis of convergence speed of the system from an error variance perspective,
- Analysis of turbo decoding iteration patterns in order to find the best iteration schedule for the system,
- Development of the novel idea of smart iteration schedule, in order to make perfect use of the iterative property of turbo decoding,
- Analysis of the convergence behavior of the system in terms of frame length,

- Comparing the convergence of IMUD-PCCC structures and IMUD-SCCC structures.

1.5 Thesis Outline

This thesis is organized in five chapters as shown in the diagram of Figure 1. The introduction chapter of this thesis gives us a general overview of the subsequent chapters. It also provides a brief review of previous research done on the convergence behavior of IMUD systems.

Chapter 2 provides the background knowledge that is useful in the subsequent chapters. This chapter is organized in two independent sections. The first section contains a general description of asynchronous DS-CDMA systems, and presents the basic elements in a DS-CDMA system such as transmitters and receivers. The second section introduces the turbo coding and turbo principle which have an important role in the structure of this research work.

The two independent fields that are presented in chapter 2 will join together in the context of an iterative multi-user detection scheme in chapter 3. A general overview of MUD is presented and various MUD techniques are reviewed and compared, leading to the introduction to iterative MUD systems.

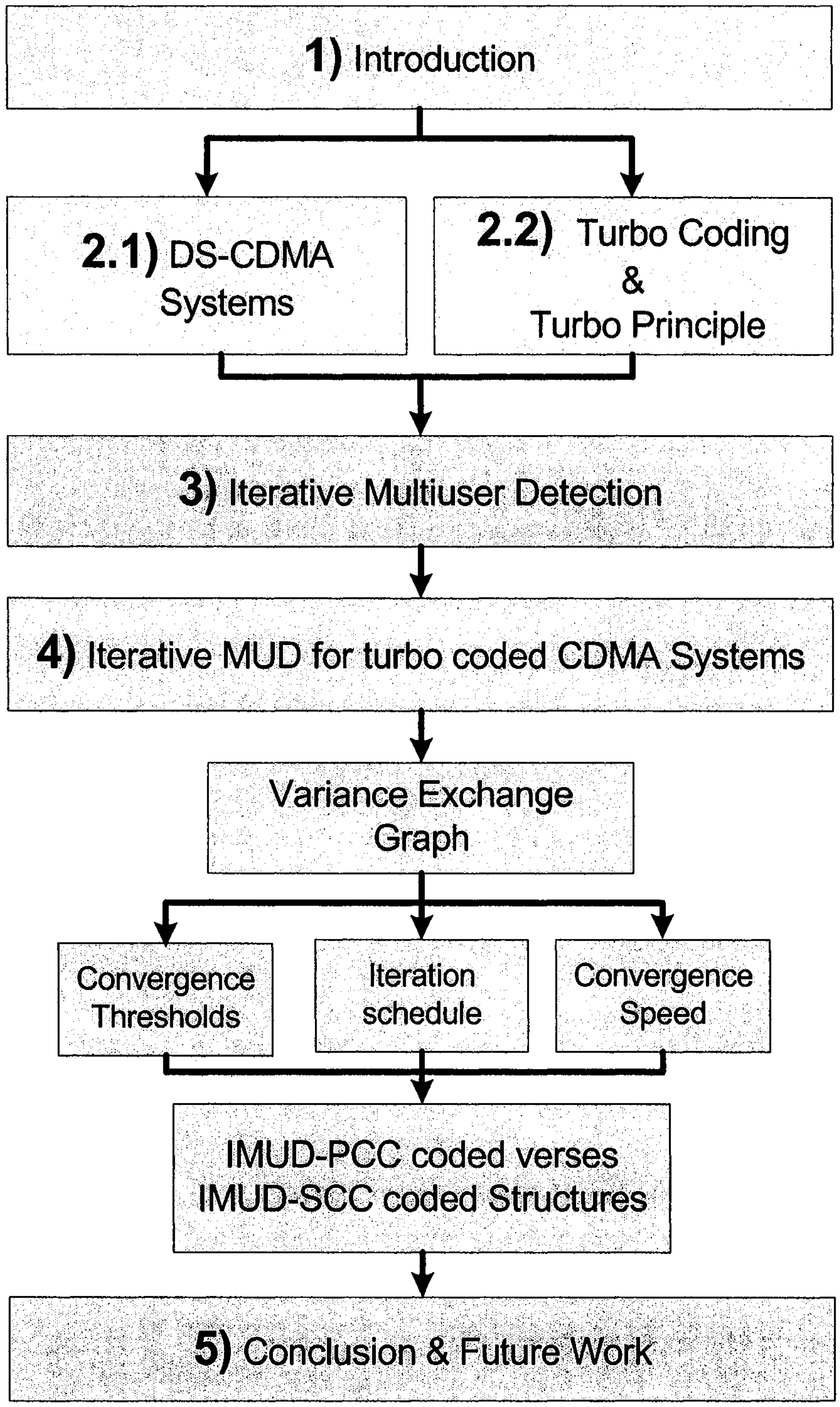

Figure 2: Thesis organization

In chapter 4, an iterative multi-user detection turbo-coded DS-CDMA system is introduced. Subsequently, The VEG is described in detail as the analysis methodology for the convergence behavior of the system in question. Consequently, the convergence of the proposed system is investigated from three different angles: convergence thresholds, convergence speed and iteration schedule. Finally, a comparison between IMUD-SCCC structures and IMUD-PCCC structures is presented.

Chapter 5 concludes this thesis. It highlights the contributions and achievements of this thesis. Some suggestions for the future work are also discussed in this final chapter.

2 General Background

The purpose of this thesis is to analyze the convergence behavior of iterative multi-user detection for turbo-coded DS-CDMA systems. In order to discuss such a sophisticated system, it is essential to present a brief description of DS-CDMA systems and turbo coding independently. This chapter gives a review of key concepts used in following chapters.

Section 1 presents the fundamentals of asynchronous DS-CDMA systems. This section is organized in two subsections that discuss transmitters and receivers. Turbo coding and decoding scheme will be discussed in detail in section 2.

2.1 Direct Sequence CDMA Systems

A Direct Sequence Spread Spectrum (DSSS) technique is performed by spreading the data signal with a unique spreading sequence which is independent of the data signal. The receiver, which is synchronized with the transmitter and with knowledge about the code, is then capable of despreading and recovering the desired signal [13], [14].

Figure 3 shows a basic Direct Sequence Spreading Spectrum (DSSS) system for both the transmitter and the receiver.

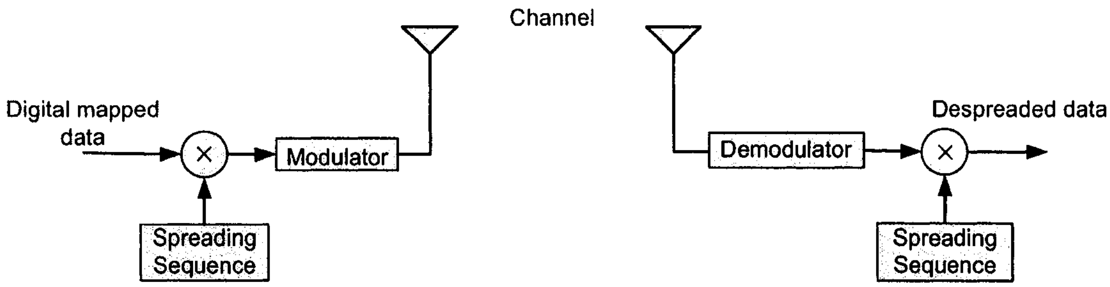

Figure 3: Basic DSSS system

In DSSS systems, spreading data is performed by multiplication of the data signal by a spreading sequence in the time domain, which results in a signal with a frequency spectrum similar to the spectrum of the spreading sequence. This is due to the fact that $T_c \prec T$, where T_c and T represent the duration of one chip in the spreading sequence and one symbol in the data signal respectively. Therefore, the effects of increasing the data rate from R (symbol level) to R_c (chip level) are a reduction in the amplitude spectrum (from T to T_c) and an expansion of the signal in the frequency domain. Since the wide bandwidth of the spreading sequence allows us to reduce the amplitude spectrum to noise levels, the generated signals appear as background noise in the frequency domain. From another perspective, the bandwidth of the data signal is basically spreaded by a factor of $N = T / T_c$, which corresponds to the processing gain in direct sequence systems. In this type of systems, the processing gain is the same as the length of the spreading sequence. A DSSS receiver multiplies received signal by a synchronized replica of the spreading sequence used for a desired transmitter. This operation will cause the desired signal to recover, and other undesired signals to remain spread over the frequency domain which results in minimal interference to the desired signal.

2.1.1 DS-CDMA Transmitter

In DS-CDMA systems, all users share the same communication channel overlapping transmitted signal in the time and frequency domain. To separate users from one another, a unique spreading sequence is assigned to each user in code domain. CDMA systems can be either synchronous or asynchronous. In synchronous systems the received data of all users are synchronous i.e. the symbols of all users are aligned in time domain, whereas they may not be in asynchronous systems.

A general scheme of a asynchronous coded CDMA system model is depicted in Figure 4.

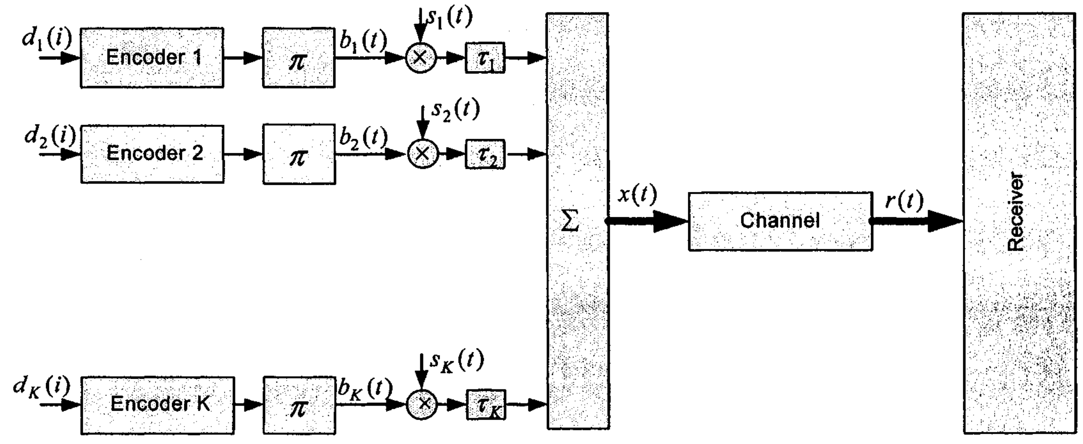

Figure 4: A general scheme of coded asynchronous CDMA tramsmitter

In this model, each of K users generates an input binary data stream. Then the data bits of each user are encoded seprately. To reduce the negative effect of error bursts of channel, interleaving is applied to the output of encoders. At the interleavers output, each user transmits interleaved coded bits in the frames of length L. The interleaved coded bits of each user are then spreaded by a unique spreading sequence, $s_k(t)$, which is unit energy and zero outside $[0,T]$, i.e.

$$\|s_k\|^2 = \int_0^T s_k^2(t)dt = 1 .$$

A mathematical representation of transmitted signal of the k^{th} in time domain can be expressed as:

$$x_k(t) = \sum_{i=0}^{L-1} \sqrt{P_k}\, b_k(i) s_k(t - iT - \tau_k) \qquad (2.1)$$

where P_i denotes the transmitted power of the i^{th} user and τ_k is the delay of the k^{th} user.

2.1.2 DS-CDMA Receiver

The received signal in a CDMA system is a superposition of the individual transmitted signals and channel noise. The received signal in Figure 1 can be expressed as:

$$r(t) = \sum_{k=1}^{K} x_k(t) + n(t) \qquad (2.2)$$

where $n(t)$ is the Additive White Gaussian Noise (AWGN) with zero mean and the variance of $N_0/2$.

For each user, the received signal includes two undesired components: the interference from other users (MAI) and channel noise. We have to fight against these two destructive phenomenon on the receiver side. Extracting the desired signal for each user from received signal mainly can be done in either of the two ways: Conventional receiver or Multi-user receiver.

2.1.2.1 Conventional Receiver

A conventional detector is generally a bank of K parallel correlators. For the received signal in (2.2), it becomes a bank of K matched filters; each one is matched to the spreading sequence of one user. The outputs of matched filters are sampled at bit times, yielding soft estimates of transmitted symbols as shown in Figure 5.The output of the conventional detector for the k^{th} user can be expressed as:

$$y_{MF,k}(i) = \int_{iT+\tau_k}^{(i+1)T+\tau_k} r(t)s_k(t-iT-\tau_k)dt$$

$$= \sqrt{P_k}b_k(i) + \underbrace{\sum_{l=1}^{L}\sum_{j\neq k}^{K}\sqrt{P_j}b_j(l)h_{jk}(l,i)}_{MAI} + \underbrace{n_k(i)}_{Noise} \tag{2.3}$$

where $n_k(i) = \frac{1}{T}\int_{iT+\tau_k}^{(i+1)T+\tau_k} s_k(t-iT-\tau_k)n(t)\ dt$ is the sampled noise with the variance of σ_0^2, and

$h_{jk}(l,i) = \int_{iT+\tau_k}^{(i+1)T+\tau_k} s_j(t-lT-\tau_j)s_k(t-iT-\tau_k)dt$ is the asynchronous correlation value between

the l^{th} symbol of the j^{th} user and the i^{th} symbol of the k^{th} user.

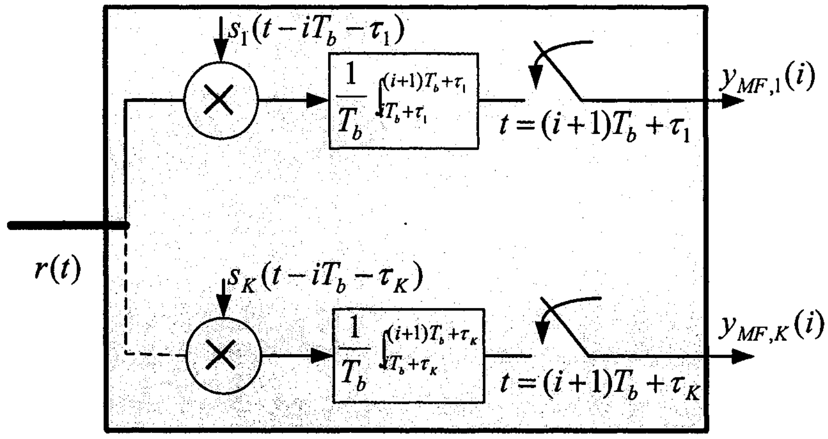

Figure 5:Conventional detector

Here, we assume that the receiver has perfect knowledge of spreading sequences and of timing information to generate a synchronized replica of spreading sequence for each user. Acquisition of timing information is a non-trivial issue in the asynchronous CDMA systems, but this is not we will concentrate on in this thesis [15].

Conventional detectors detect each user regardless of other users and treat the interference of other users as additive noise. They follow a single-user strategy because there is neither sharing of multi-user information, nor joint signal processing.

In particular cases, when received signals of the users are orthogonal, conventional detectors are optimum [16]. This happens when the orthogonal spreading sequences are used and there is no time delay caused by channel. When received signals of the users are not orthogonal, conventional detectors are not optimum anymore; and the quality of their performance strongly depends on the correlation between the received signals of users.

2.1.2.2 Multi-user Receiver

For non-orthogonal received signals, the performance of CDMA systems can be greatly enhanced by joint detection of all users instead of detecting them separately as done in conventional detectors. The task of these detectors, referred to as multi-user detectors, is to reliably detect the information signal for each user using joint processing techniques.

The most widely recognized multi-user receiver so far, is optimum multi-user detector or maximum likelihood (ML) detector introduced by Verdu [5]. Unfortunately, its prohibitive computational complexity, which grows exponentially with the number of users, makes this structure too complex for practical CDMA systems. Hence, most of the recent research has addressed the designing of sub-optimal multi-user detectors for practical systems. There are wide-range tradeoffs between performance and complexity for sub-optimal multi-user detectors.

In practical systems, sub-optimal multi-user detection is often used in conjunction with error control decoding to obtain a powerful detection scheme [17], [18], [19] and [20]. Error control coding increases system performance and makes it possible to form an iterative structure as shown in Figure 6 .

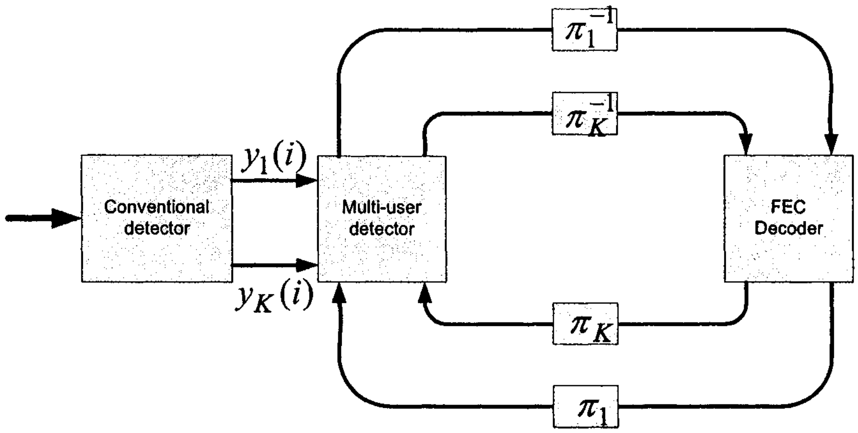

Figure 6: Typical iterative multi-user detection

As Figure 6 shows, the iterative multi-user receiver includes a conventional detector at the first stage. It operates on the correlated soft outputs of the conventional detector, and tries to reduce the correlation between users' data throughout iterations.

In this thesis, we will use turbo decoding in a structure similar to Figure 6. Hence, an introduction of turbo coding is essential in two aspects: In one aspect, we discuss turbo coding as a FEC coding scheme that will be employed in the context of the proposed IMUD system; from another angle, we present it as an iterative decoding scheme that gives us a general insight into iterative structures.

2.2 Turbo Coding and Turbo Principle

Turbo codes have received considerable attention since their introduction in 1993 [21]. This is because of their exceptional performance at very low signal to noise ratios, reaching close to the Shannon limit of reliable communication. Because of their outstanding performance, research work in turbo codes and their application has never diminished. Today, turbo codes have been accepted in the third generation mobile communication standard (3G) as one of the channel coding methods for wireless data service [22].

Turbo decoding process involves Soft-Input Soft-Output (SISO) decoding of concatenated codes separated by an interleaver. Depending upon the codes used for concatenation, turbo codes are classified as convolutional turbo codes and block turbo codes. Although it is theoretically possible to derive the optimal decoder for these codes, the complexity of optimal decoding makes it practically impossible. Therefore, the suboptimal iterative decoding algorithms are used for turbo codes in practice [23]. An iterative decoder is a suboptimal decoder that provides very good performance while requiring a modest level of complexity. An iterative decoder operates on the received data in a repetitive manner utilizing feedback to form estimates of the transmitted data.

2.2.1 Turbo Encoder

The original turbo encoder consists of a parallel concatenation of two recursive systematic convolutional (RSC) encoders that receive the same information bits in different orders. A more general structure is extended to include more than two convolutional codes either in a parallel or serial concatenation [24], [25], [26] and [27].

In Figure 7, we present a generic turbo encoder. The two constituent RSC encoders are identical with the generator polynomial of $G(D) = (1,5/7)$. The first block of data will be encoded by the first encoder. The same block of information bits is interleaved and encoded by the second encoder. The parity bits can be "punctured" in order to obtain higher coding rates. A ½ turbo code can be implemented by alternatively selecting the outputs of the two encoders [28].

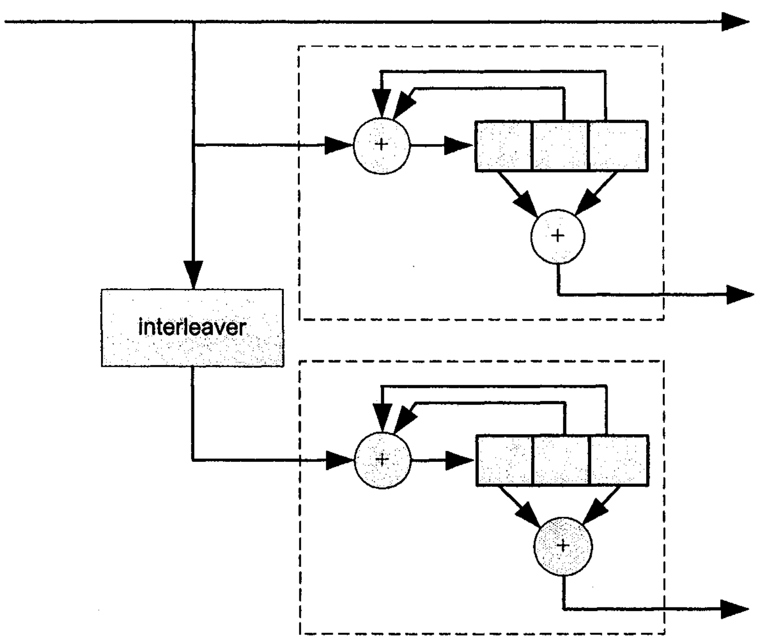

Figure 7: Turbo encoder

The output of a RSC encoder usually has a fairly high hamming weight due to its infinite impulse response nature but there are some input sequences that result in low Hamming weight outputs. By passing the same input through both RSC encoders in different orders, the probability that both encoders simultaneously produce low-weight outputs is reduced. Therefore, the embedded interleaver in turbo encoder plays a key role in the performance of such a coding scheme.

2.2.1.1 Interleaving

Interleaving is a permutation ($i \rightarrow \pi(i)$) that changes the order of a data stream. Any interleaver has its corresponding deinterleaver (π^{-1}) that is able to restore interleaved data stream to its original order.

In turbo encoder, similar to any concatenated coding structure, an interleaver is needed between the two constituent encoders to prevent the phenomena of error propagation at the decoding process. The performance of turbo codes highly depends on the interleaver size and the interleaver design [29]. A possible choice for this purpose can be a simple block interleaver, i.e. to write by row and read by column. However, some input of low weight would give very unfortunate patterns in this interleaver. A better choice is using a pseudo-random interleaver, i.e. to read the information bits to the second encoder in a random (but fixed) order. The probabilities of having a critical output is reversely proportional with interleaver size and typically much lower than block interleaver with the same size [30]. Therefore, pseudo-random interleaver is superior to block interleaver and is standard for turbo coding.

2.2.2 Turbo Decoder

The iterative turbo decoder consists of two soft-input/soft-output (SISO) decoders, concatenated serially via an interleaver identical to the one used in the turbo encoder. The decoding process is performed in an iterative way: The two constituent decoders exchange information between each other until the desired degree of convergence is achieved.

The output of one decoder is used as a-priori information by the other decoder. In order to take advantage of this iterative decoding scheme, it is necessary for each decoder to produce soft-bits estimation in the form of log-likelihood ratio (LLR). The LLR of a bit is a signed real number defined as follows:

$$L(d_k) = \log \frac{p(d_k = 1 \mid Y)}{p(d_k = 0 \mid Y)} \tag{2.3}$$

where Y is the received coded data frame; i.e. $Y = (x_1', p_{1_1}', p_{1_2}', \ldots\ldots, x_L', p_{L_1}', p_{L_2}')$. If the LLR of a bit is positive, the bit most likely is "1". Conversely, if the LLR of a bit is negative, the bit most likely is "0". Therefore, we can make a hard-decision based on LLR value of a bit:

$$d_k = \begin{cases} 0 & if \quad L(d_k) \prec 0 \\ 1 & if \quad L(d_k) \succ 0 \end{cases} \tag{2.4}$$

The magnitude of $L(d_k)$ shows the reliability of the hard decision. Considering the LLR definition, we can also obtain a soft estimation of a mapped bit using its LLR as follows:

$$\tilde{d} = \tanh(\frac{L(d)}{2}) \tag{2.5}$$

The input of the SISO decoders consists of three components: systematic information, parity information and a-priori information from previous decoding stage. The output of the SISO decoders can be divided into three following components [31]:

$$L(d_k) = L_e(d_k) + L_a(d_k) + L_c x_k \tag{2.6}$$

$L_e(d_k)$ is the extrinsic information which is newly gained from current decoding stage and replaces the $L_a(d_k)$ of the next decoding stage. $L_a(d_k)$ is called the a-priorio information for bit d_k which is provided by the other decoder. $L_c x_k$ is the intrinsic information that originates from systematic bits. L_c is called the reliability of channel value and is calculated as follows:

$$\log \frac{p(x_k \mid d_k = +1)}{p(x_k \mid d_k = -1)} = \frac{2}{\sigma^2} x_k = \frac{4E_s}{N_0} x_k = L_c x_k \tag{2.7}$$

To prevent positive feedback, only the extrinsic information must be exchanged between the two decoders. Otherwise, the performance will not be improved through the iterative process. The block diagram of the iterative turbo decoder is shown in Figure 8. Each iteration involves two decoding stages. The decoding process is as follows:

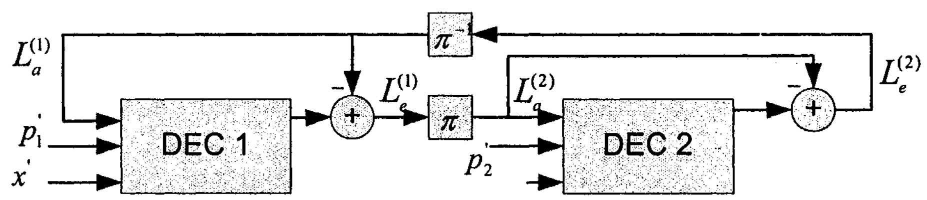

Figure 8: Turbo decoder

The first decoder receives the systematic input ($x^{'}$) the parity input ($p_1^{'}$) and the de-interleaved a priori information ($L_a^{(1)}$). The a-priori information is produced by second decoder and it is initialized to zero for the first iteration. When the first decoder finished decoding process, it generates the extrinsic information ($L_e^{(1)}$) and feeds it to the interleaver. Then the second decoder starts decoding process based on the parity bits ($p_2^{'}$) and interleaved extrinsic information of the first decoder which is a priori information for the second decoder($L_a^{(2)}$). The extrinsic information of the second decoder is fed back to the first decoder after deinterleaving, and the first decoder repeats the same process.

The above process can be iterated many times until eventually all the errors are corrected or there remains an error pattern that cannot be corrected. In general, the performance of turbo decoding improves as the number iteration increases. However, the decoding gain from one iteration to the next one decreases.

2.2.2.1 Map Algorithm

As mentioned before, the two constituent decoders in the turbo decoder have to be SISO decoders. Two suitable candidate algorithms for this purpose are Soft Output Viterbi Algorithm (SOVA) and maximum a posteriori (MAP) algorithm.

SOVA is a Maximum Likelihood (ML) Algorithm that finds the most probable information sequence that was transmitted [32], whereas MAP algorithm attempts to find most probable information bits [33]. MAP algorithm and SOVA are both trellis-based algorithms. MAP algorithm takes all the possible paths into consideration to estimate the soft value of a bit, whereas SOVA generates output considering only two paths: the maximum likelihood (ML) path and its strongest competitor. In contrast to SOVA, the information bits decoded by MAP algorithm do not necessarily form a connected path through the trellis.

The error performance of SOVA and MAP algorithm are not much different under high Eb/N0. However at low Eb/N0, MAP algorithm is found to outperform SOVA by quite a margin [34]. Therefore, MAP algorithm is usually preferred to SOVA in turbo structures. MAP Algorithm involves lots of multiplications and computations which are complicated in implementation. In order to avoid non-linear calculations, modified versions of the MAP algorithm are used such as Log-MAP algorithm and Max-log-MAP algorithm. The main idea behind them is using logarithms of data in order to convert multiplication to addition, which is easier to implement. A detailed description of MAP algorithm and its derivations can be found in [35].

2.2.3 Trellis Termination

The MAP decoding algorithm is such that each decoder requires knowledge of the first and the last state of the trellis. So, each decoder has to start and terminate at all zero state. Since the component encoders of a turbo code are recursive, it does not suffice to set the last $\nu - 1$ information bits to zero in order to drive the encoder to all zero state, where ν is constraint length of convolution coder. Note that, due to interleaving, the encoders are not in the same state at the end of the input sequence. Therefore, different termination sequences are required for terminating the encoders. The most straightforward solution for the problem of trellis termination is given in [36]. This is shown in Figure 9 , where the switch is in position "A" for the first $L - \nu - 1$ clock cycles and in position "B" for the $\nu - 1$ additional cycles. This method is used for both of encoders and it is easy to verify that it will terminate them simultaneously.

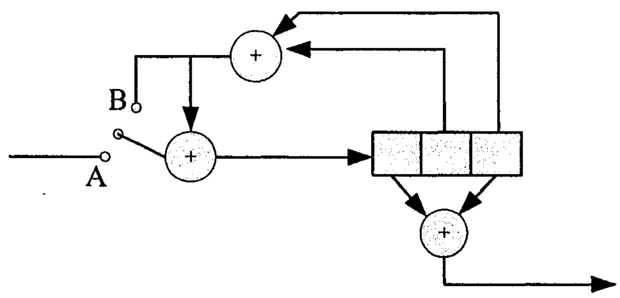

Figure 9: Trellis termination

2.2.4 The Turbo Principle

From the discussion, it is obvious that in fact the term "turbo" in turbo codes does not relate to the code itself but refers to the iterative method of decoding. The title "turbo" is taken from the principle of the turbo engine. Inside a turbo engine, exhaust gases are used to blow more air into the engine. Together with more fuel this gives more power to the engine. In the turbo engine, however the output of the engine is given as an input back to it.

The turbo concept can be used when two or more sources of information about the data are available. The information contained in these two information sources should be as uncorrelated as possible to obtain the largest improvement in the decoding process. The obtained information from each of these sources is then iterated between the decoders of the information sources, until the correlation between the information reaches to its maximum.

2.3 Summary

This chapter provided a general background for understanding coded DS-CDMA systems. A coded asynchronous CDMA system was described in the transmitter side. Different possible types of receiver were introduced leading to introduction of sub-optimal iterative multi-user (IMUD) receiver. IMUD receivers will be discussed further in the next chapter.

Turbo coding and turbo principle were described in the second section of this chapter. Turbo codes are a class of concatenated codes that provide very good performance in low signal to noise ratios. In this thesis, turbo codes are used as error control coding in the structure of iterative multi-user receiver.

The turbo principle is applicable in a situation when two or more information sources are available. In turbo codes, these two sources of information are original coded datawords and interleaved coded datawords.

3 Iterative Multi-user Detection

The Multi-user Detection (MUD) technique jointly processes the data of all users to obtain a better detected data for each individual user in the presence of MAI. Conventional detectors based on single-user approaches treat MAI as Additive White Gaussian Noise (AWGN).However, in contrast to AWGN, MAI has a perfect correlative structure that is quantified by the cross-correlation matrix of spreading sequences. Hence, the detectors that take this correlation into account perform better than the conventional detectors. MUD requires knowledge of all users, including the desired user and the interferers. Hence, it cannot be used to combat unknown interference but when all the users are desired and their information is known, we can perform MUD to achieve the most desirable detection performance.

Many types of MUD techniques have been developed so far. The optimal MUD based on maximum likelihood (ML) algorithm can approach the single-user bound. However, its high complexity, which exponentially grows with the number of users, makes it not practical. Therefore, some sub-optimal MUD algorithms which have lower complexity have been investigated. A class of sub-optimal MUD is referred as decision-feedback MUD or interference cancellation, such as successive interference cancellation (SIC) and parallel interference cancellation (PIC). The PIC detector can be used in iterative structures to perform as many iterations as needed. This feature enables us to combine the PIC-based

MUD and error control decoding in an iterative structure to achieve better performance. This structure is called iterative multi-user detection (IMUD).

Since the introduction of turbo codes and turbo principle, iterative multi-user detection began to be called "Turbo Multi-user Detection". However, turbo multi-user detection does not necessarily involve a turbo decoder.

3.1 Multi-user Detection Schemes

A general classification of sub-optimal MUD schemes is shown in Figure 10.

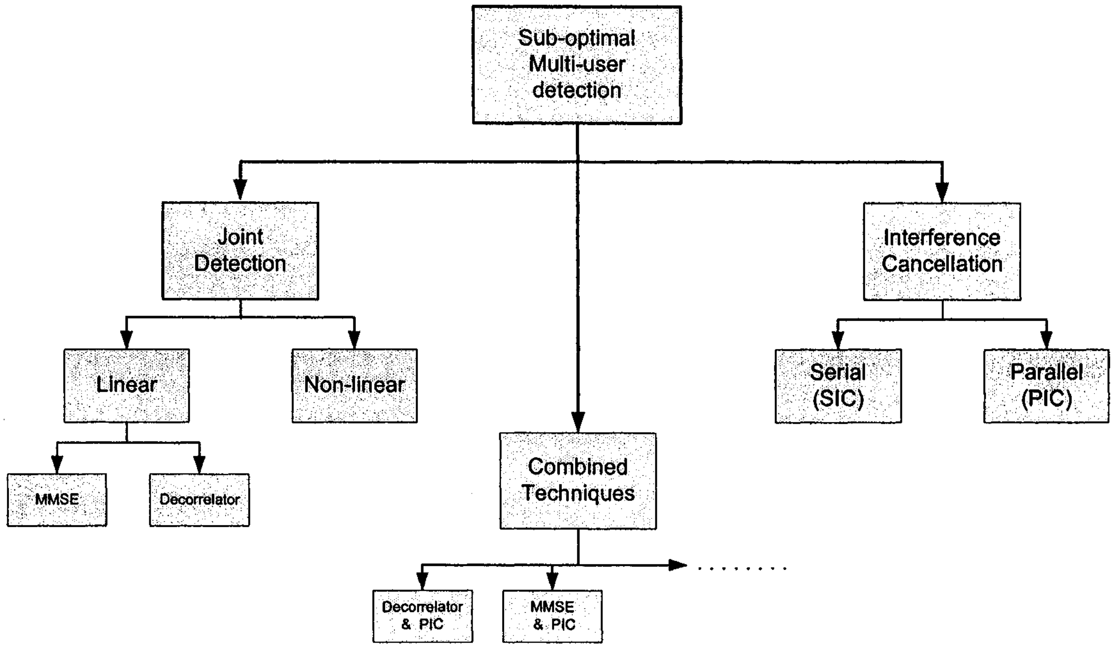

Figure 10: A general classification of sub-optimal multi-user detection schemes

This research area can be divided into three categories: Joint detection schemes, interference cancellation schemes and structures with combined schemes. Joint detection schemes generally consist of a bank of conventional receivers followed by a linear or non linear filter. Interference cancellation schemes are mainly characterized by data estimation and subtraction of interference from the output of conventional detectors, and finally the third category can be defined as the multi-user detection structures that use a combination of above mentioned techniques.

3.2 Joint Detection

Joint detection structures are divided into two main classes, which are called linear and non-linear detection methods. Non-linear detectors are used when the desired signal is not linearly separable. Such a situation can happen when there is a high level of MAI and/or ISI in the system [37]. The most important structures of category are M-algorithm and T-algorithm detectors [38], pre-selection maximum likelihood (PSML) multi-user detector [39] and Greedy detector [40].

Linear detectors mitigate interferences by applying a linear transformation to the output of conventional detectors. We present the output of the conventional detector (bank of k matched filter) as follows:

$$y(t) = RAb(t) + n'(t) \tag{3.1}$$

where R is the correlation matrix, which is invertable. A is $diag[\sqrt{P_1}$,........, $\sqrt{P_k}]$ and $n'(t)$ is the correlated noise at the output of the conventional detector.

In this chapter, we review the two most common linear detectors: Decorrelator detectors and Minimum Mean Squared Error (MMSE) detectors.

3.2.1 Decorrelator Detector

Considering that the correlation matrix R is invertible, we can multiple both sides of (3.1) by R^{-1}.

$$R^{-1}y(t) = Ab(t) + R^{-1}n'(t) \tag{3.2}$$

So, the output of the decorrelator detector can be written:

$$\hat{b}(t) = Ab(t) + n^{dec}(t) \tag{3.3}$$

where $n^{dec}(t)$ is the noise vector with zero mean and the covariance matrix of $\sigma^2 R^{-1}$.As we can see in (3.3), the MAI is completely eliminated at the output of the decorrelator without need to estimate the users' amplitude. Since the power levels of users are independent from one another, decorrelators have the best near-far resistance.

Unfortunately, this detector has two significant disadvantages. Firstly, it intensifies the noise by the term $n^{dec}(t)$ in (3.3). It has been shown in [40] that the noise power associated with the noise term $n^{dec}(t)$ is always greater or equal to the noise term at the output of conventional detector. Secondly, decorrelators need to invert the matrix R which is a complex operation.

3.2.2 Minimum Mean Square Error (MMSE) Detector

MMSE detectors apply a liner transformation to minimize the mean square error between their output and input, i,e.

$$\min E[\| b(t) - Ty(t) \|^2] \tag{3.4}$$

where T is the transformation matrix of MMSE filter.

MMSE filters can be regarded as improved decorrelators that can solve the problem of noise enhancement at low signal to noise ratios. The main distinction between MMSE filters and decorrelators is that, MMSE filters take the background noise into account and utilize the knowledge of received power. Taking the background noise into consideration avoids the noise enhancement problem of decorrelators. The soft output of MMSE detectors is given as [42]:

$$\tilde{b}(t) = (R + \sigma_0^2 A^{-2})^{-1} y(t) \tag{3.5}$$

As we can observe from (3.5), MMSE becomes decorrelator when no noise is present $(\sigma_0^2 = 0)$. The performance of MMSE filters is similar to decorrelators at high SNR; whereas, it surpasses decorrelators at low SNR.

Two main drawbacks of MMSE filters are that they require an estimation of received amplitudes and their performance depends on the power of interfering users. In terms of complexity, they face the complexity of matrix inversion implementation similar to decorrelators.

3.3 Interference Cancellation Detectors

In this class of detectors, the MAI affecting each user is estimated, and then cancelled from that user. Two main detectors of this class are serial interference cancellation (SIC) and parallel interference cancellation (PIC) detectors.

3.3.1 Serial Interference Canceller (SIC)

SIC detectors take a serial approach for interference cancellation [43]. The first operation in a SIC detector is sorting the users in descending order according to their received powers. The cancellation process starts with generating the transmitted signal of the strongest user in terms of power. This regenerated signal provides an estimate of the MAI caused by the strongest user, which is then subtracted from the total received signal, yielding a partially cleaned version of the received signal. If the estimated data of the user is accurate, the remaining users see less MAI in the next stage. Thus, this new version of received signal can be used to detect the next stronger user in the system. This process is repeated until all the users are detected. Note that in each stage the estimation of the users are obtained by making a decision at the output of the conventional detector. The benefit of sorting the signals out in a descending order is because the strongest user can give the most accurate estimation and consequently the removal of this signal will provide the most benefit to the remaining users.

Serial cancellation approach is efficient in a situation where all of the users are received with different signal strength. However, the delay of such an approach suffers from extended delay which is proportional with the number of users.

3.3.2 Parallel Interference Canceller (PIC)

PIC detectors try to simultaneously remove the MAI from all users. In this type of detector, each user receives equal treatment in the attempt to cancel its MAI. Compared with SIC, since the interference cancellation is performed in parallel for all the users, the process delay is dramatically reduced.

Figure 11 shows one stage of a PIC detector for K users. The initial bit estimates, $y_k(i)$, are usually provided by the output of a bank of conventional detectors. The parallel interference cancellation can be performed iteratively by using the increasingly reliable data provided by the last iteration.

$$z_{k,l}(i) = y_k(i) - \sum_{j \neq k} \rho_{jk} \tilde{b}_{j,l-1}(i) \qquad (3.6)$$

where $z_{k,l}$ is the l^{th} iteration PIC detection result and $\tilde{b}_{j,l-1}$ is the data decision made by the PIC detector in the $l-1^{th}$ iteration (stage).An example of implementation of (3.6) is multistage PIC, where several stages of PIC are cascaded.

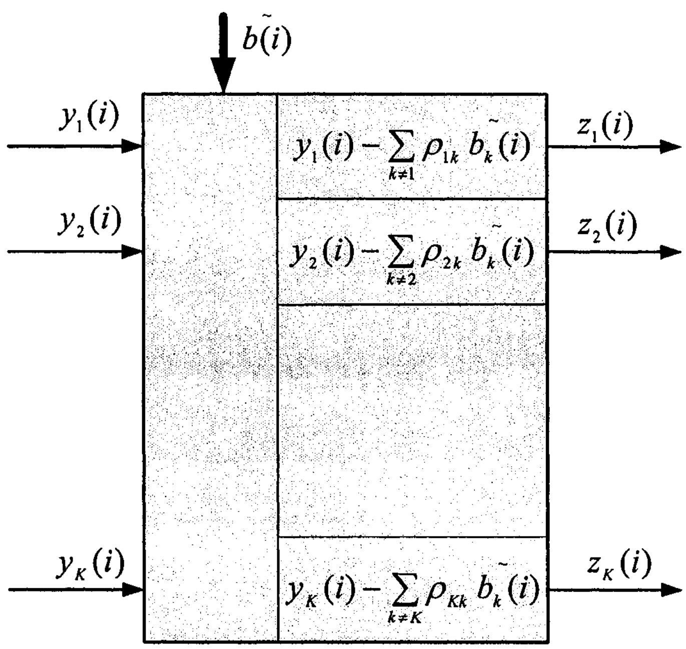

Figure 11: Parallel interference canceller (PIC)

Generally, PIC detectors can avoid the long delay derived from serial processing in SIC detectors. The PIC detection is preferred to the SIC detection when power-control techniques are employed. The iterative PIC can perform many iterations to achieve the best possible performance. Hence, PIC detectors are commonly used in IMUD structures [44], [45] and [46].

PIC detectors have the risk that the detection results in the next stage (iteration) are even worse than in the current one". This is a serious problem that degrades the whole performance of the systems including the PIC detector, and can result in failure of convergence [47]. In a PIC-based receiver, the current decisions of users' data use the reconstruction of MAI that is made by the decoders from the previous iteration. When the provided data for the PIC have lots of errors, the problem occurs. The wrong decision of a bit will result in the wrong interference reconstruction. Consequently, when the wrong value of interference is subtracted from the matched filter output, the MAI can be strengthened, instead of being cancelled. It means that the error is propagated to the corresponding bit of other users. This may result in turning the data value into wrong direction in the next iteration. This destructive phenomenon is called "error propagation", and is the reason of "Oscillatory behavior".

Typically, there are a few measures that can be taken to prevent oscillatory behavior. First approach is using a soft-in soft-out (SISO) PIC detector rather than a hard PIC detector. Giving small values to unreliable symbols will leave the chance to correct the potential errors in the next iterations. This approach requires that soft data have to be provided for the PIC.

Second approach is a similar soft decision-based approach named "partial PIC" (PPIC), which is proposed in [47], [48] and [49]. This method alleviates the error propagation by introducing a weight in each stage to reduce the cost of wrong estimation. The decision statistics at the current stage is a weighted sum of previous decision statistics and statistics resulting from MAI cancellation. Details of this can be found in [47] and [50].

Third technique is trying to improve the quality of the data provided for the PIC in order to reconstruct more accurate MAI. Using powerful decoding scheme, such as concatenated convolutional codes, significantly helps to prevent the oscillatory behavior.

3.4 Combined Schemes

Combined structures between linear detection techniques and interference cancellation schemes normally provide an enhanced receiver performance. This improvement is obtained from the combination of certain positive features from two or more of these detection techniques.

Generally, combined schemes consist of one or more stages for data estimation followed by one of interference cancellation schemes. It is clear that success of any cancellation scheme relies on the initial data estimation. If this estimation is not reliable, error propagation will occur and consequently the system will collapse.

In chapter 4 we will discuss about using FEC codes in conjunction with multi-user detection that leads to a more powerful performance. In that class of systems, decoder stage provides soft data estimation and feedback it to soft cancellation stage to refine it's output in an iterative way. This iterative cycle can repeat a few times to obtain better results.

3.5 Turbo Multi-user Detection

The iterative detection concept provides the possibility of combined error control coding and multi-user detection schemes in a way that information is exchanged and improved through iterations [51]. The receiver constructed following the "turbo principle" is referred as "iterative receiver" or "turbo receiver".

Different types of multi-user detection have been combined in an iterative structure with error correction coding. Examples of the use of the MMSE filters, decorrelators and PIC detectors in iterative structures can be found in [52], [53] and [54] . For the time being, most of the promising turbo MUD approaches are based on interference cancellation detection, especially PIC. Different iterative PIC-based structures in conjunction with different decoding schemes have been investigated so far. In such structures, mainly some of the mentioned measures are taken to avoid the error propagation phenomena.

In [55], Wang and Poor adopted another method to prevent the error propagation of PIC detectors. They employed a simultaneous MMSE filter following the PIC detector to further suppress the residual interference. After MMSE filtering, a bank of parallel convolutional decoders can achieve a satisfactory performance. In this receiver, the simultaneous MMSE filter can enhance the convergence of the whole receiver, although the convolutional decoder cannot by itself. The drawback of this approach is that the MMSE filter coefficients need to be recalculated for each coded bit in each iteration. This will be too complex for practical applications. In [56], the MMSE detector is only used in the first iteration which is supposed to be the weakest link of the whole detection process. The idea behind this approach is that if the oscillation of the PIC can be prevented in early iterations, it is unnecessary to utilize the MMSE filter for the rest of iterations.

In short, PIC detectors may fail to cancel the MAI due to the poor data estimation in the early stages of the iterative detection. In such a circumstance, decoding schemes such as e convolution decoding seems not to be able to efficiently prevent the error propagation of the PIC. However, turbo codes have such ability thanks to their power error-correction capability. Turbo decoding can enhance the convergence of the receiver if the PIC detector fails to converge in early iterations. This shows that the turbo decoder can effectively solve the problem of error propagation. This structure is also called turbo multi-user detection. Obviously it employs not only the "turbo principle", but also the turbo decoder itself.

Turbo decoders utilizing MAP algorithm return the log likelihood ratio (LLR) of bits that can be used to judge the reliability of the data. Before LLRs are fed back to the PIC to reconstruct the MAI, they have to be converted to proper values using either soft or hard conversion as discussed in the section 2.4. Considering the uncertainty of data, hard decision method is not appropriate because the LLR of a bit contains the reliability information as well as its value, and hard decisions definitely discard the reliability value. Instead of hard decision, tangent-hyperbolic function is used to map LLR values inside the [-1, 1] interval. A relatively reliable data leads to a value approaching 1 or -1, whereas an unreliable data will lead to a value close to zero. Figure 12 shows the tangent hyperbolic transformation curve.

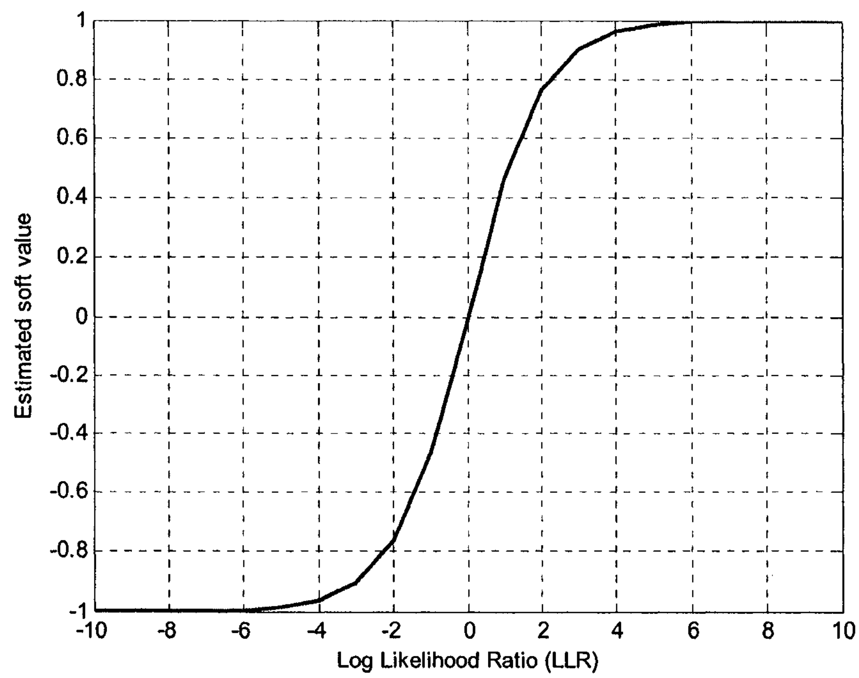

Figure 12: The transformation used to covert LLR to soft estimation value.

Here, an approach is used for turbo decoding to calculate the LLRs of not only information bits, but also parity bits. Because the PIC needs the soft estimation of the whole coded bit frame to reconstruct the interference estimation. The LLRs of parity bits can be calculated by the similar approach used for the LLRs of information bits [57].

3.6 Summary

In this chapter, we presented the structures that are named "turbo multi-user detectors". Typically, iterative MUD systems include the PIC as the proper interference canceller for iterative structures. This motivates us to seek measures that protect PIC-based systems against "error propagation". Mainly, iterative MUD structures including PICs can be of two kinds from convergence viewpoint.

In the first structure, the PIC and the MMSE form a powerful multi-user detection scheme, which is in an iterative loop with convolutional decoding. In this system MMSE has the key role in the convergence of the system. In the second structure, iterative loop consists of a PIC as multi-user detector and a powerful decoding scheme such as turbo decoding. In this structure, error control coding plays the prominent role in the system convergence.

4 Convergence Behavior of Iterative Multi-user Detection for Turbo-coded CDMA

Iterative (turbo) multi-user architectures are difficult to analyze by traditional BER curves. A recently proposed method to visualize and analyze the convergence behavior of turbo architectures is the Variance Exchange Graph (VEG). VEGs provide us useful information about convergence behavior of the system and clearly portray the role of the components in the entire picture of iterative multi-user detection. They make clear that to what level the components in a turbo structure are well matched to one another other.

In order to make use of VEG, we first introduce an empirical parameter, variance, which has been shown to be an accurate measure [58] and [59]. The use of variances makes it possible to exactly quantize the relationship between the input and the output statistics of some linear multi-user detectors with close-form formulas, and hence simplifies the analysis of the iterative structure to a great extent. The function of each component in the iterative structure can be clearly shown through the observation of the variance evolution. Moreover, the impact of system parameters such as signal to noise ratio and system load can be observed visibly.

4.1 System Model

The proposed iterative multi-user detection system is illustrated in Figure 13.

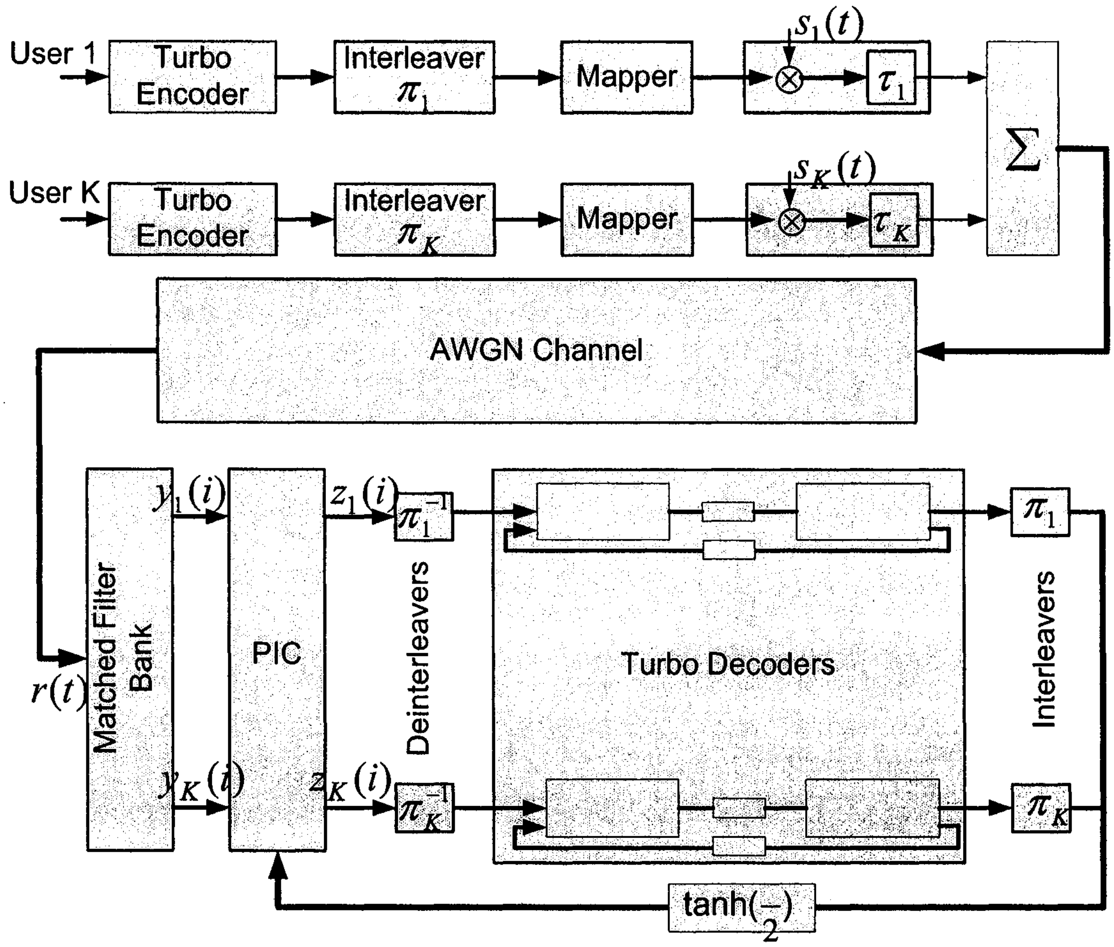

Figure 13 : The iterative multi-user detection turbo-coded CDMA system

The K data sources generates independent binary information bits. These bits are written into frames and then fed into K parallel one third rate turbo encoders. Each of the turbo encoders consists of two recursive systematic convolution (RSC) encoders with the generator polynomial $G = (1,5/7)$ in octal notation and a random interleaver between them. For each user, a different coded bit interleaver is introduced to reduce the influence of error bursts at the input of each channel decoder. The interleaved coded bits of the k^{th} user are BPSK mapped, yielding data symbols of duration T. Each data symbol is then modulated by a spreading waveform, $s_k(t)$, and transmitted through AWGN channel. The transmitted signal of the k^{th} user can be expressed as:

$$x_k(t) = \sum_{i=0}^{L-1} A_k b_k(i) s_k(t - iT - \tau_k) \tag{4.1}$$

where $\tau_k \prec T$ is the time delay of the k^{th} user and L is the length of coded frame. The received signal can be expressed as:

$$r(t) = \sum_{k=1}^{K} x_k(t) + n(t) \tag{4.2}$$

where $n(t)$ is zero-mean white Gaussian noise with variance of σ_0^2 . The received signal, $r(t)$, passes through a bank of matched filters. The output of the k^{th} matched filter can be expressed as:

$$y_k(i) = A_k b_k(i) + \sum_{l=1}^{L} \sum_{j \neq k}^{K} A_j b_j(l) h_{jk}(i,l) + n_k(i) \tag{4.3}$$

$$\text{where } n_k^{'}(i) = \frac{1}{T} \int_{iT+\tau_k}^{(i+1)T+\tau_k} s_k(t - iT - \tau_k) n(t) dt \tag{4.4}$$

and $$h_{jk}(i,l)=\frac{1}{T}\int_{iT+\tau_k}^{(i+1)T+\tau_k} s_k(t-iT-\tau_k)s_j(t-lT-\tau_j)dt\,. \tag{4.5}$$

After retrieving data from the output of matched filters, the SISO multi-user detector and the SISO decoder form a loop and exchange information iteratively. The system contains a Parallel Interference Canceller (PIC) as multi-user detector and a bank of K parallel single-user turbo decoders as channel decoder. The PIC tries to reduce the MAI by using the matched filters outputs and the soft estimations provided by the turbo decoders. It reconstructs an estimation of the MAI for each user and then subtracts it from the matched filter output. The output of the PIC for the k^{th} user can be expressed as:

$$z_k(i)=A_k b_k(i)+\sum_{l=1}^{L}\sum_{j\neq k}^{K}A_j(b_j(l)-\tilde{b}_j(l))h_{jk}(i,l)+n_k(i) \tag{4.6}$$

where $\tilde{b}_j(l)$ is the soft estimation of $b_j(l)$ which is provided by the j^{th} turbo decoder from previous iteration. In the first iteration, the PIC is passive because no soft estimations are available by turbo decoders, and the matched filters outputs are directly fed to the turbo decoders. Turbo decoders decode each user's information independently and provide improved soft estimations of each user's data. The normalized decoders' outputs are fed back to the PIC.

In summary, the PIC and the turbo decoders form a closed loop where data values are updated iteratively. Depending on the system parameters, such as the system load and signal to noise ratio, the dynamics of the system move towards an ultimate point. In other words, the system will converge to an equilibrium point at which there will be no further improvements through iterations. The ideal scenario for the proposed system is when the system completely eliminates MAI, and reaches single-user performance.

4.2 Variance Transfer Function

In the Variance Exchange Graph method of analysis, each component is characterized by its error variance transfer behavior. So for each of the components, we have to obtain an error variance transfer characteristic. This characteristic is called Variance Transfer (VT) function. The two following types of variance are introduced to keep the interfacing parameters of the PIC and turbo decoders compatible:

1. Bit variances, σ_{bit}^2, are the error variances of soft normalized bits that are provided by turbo decoders. They are calculated using "logarithm-likelihood ratios" on the coded symbol, c, as:

$$\sigma_{bit}^2(c) = E\left[|c - \tilde{c}|^2 \right] = E\left[|c - \tanh(\frac{\lambda(c)}{2})|^2 \right] \tag{4.7}$$

If the symbol c is correctly decoded, then its LLR amplitude would tend to infinity, ($|\lambda(c)| \rightarrow \infty$), and $\tilde{c}$ would tend to c. Therefore, $\sigma_{bit}^2(c)$ would tend to zero. On the other hand, if no knowledge of the symbol c is available, then $\lambda(c) = 0$. Consequently, the bit variance is $\sigma_{bit}^2(c) = E[c^2] = 1$. In the system of Figure 13, the initial value of $\sigma_{bit}^2(c)$ is unity. Then, it starts to approach zero through the successive iterations.

2. Observation variances, σ_{Obv}^2, denote the residual interference variance at the output of the PIC. If the number of users is high, according to central limit theorem, the MAI estimation generated in the PIC tends to Gaussian distribution [60], [61] and [62]. Hence for any user, each symbol at the output of the PIC can be modeled as the sum of the transmitted symbol and a Gaussian random variable of variance σ_{Obv}^2 as follows:

$$z(i) = b(i) + n'(i) \tag{4.8}$$

Here due to power control assumption, we can drop the subscript indicating user number, and (4.8) is valid for any user. The MAI term and the channel noise term in (4.8) is incorporated in $n'(i)$. Therefore, the logarithm-likelihood ratio of the PIC output symbols are expressed as:

$$\lambda(z(i)) = \log\left(\frac{P(z(i)\,|\,b(i) = +1)}{P(z(i)\,|\,b(i) = -1)}\right) = \log\left(\frac{\exp\left(-\frac{(z(i)-1)^2}{2\sigma_{obv}^2}\right)}{\exp\left(-\frac{(z(i)+1)^2}{2\sigma_{obv}^2}\right)}\right) = \frac{(z(i)+1)^2 - (z(i)-1)^2}{2\sigma_{obv}^2} = \frac{2z(i)}{\sigma_{obv}^2} \tag{4.9}$$

The LLR values of the PIC outputs are fed to the bank of single-user turbo decoders.

4.2.1 Variance Transfer Function of PIC

The VT function of the PIC depends on σ^2_{bit}, which is provided by turbo decoders from the latest iteration as following:

$$\sigma^2_{Obv}(c) = F_{MUD}(\sigma^2_{bit}(c))\,|_{\sigma_0=cte} \tag{4.10}$$

It is possible to analytically derive the F_{MUD} closed-form formula. Assuming the received powers of all the users to be unity, we can obtain the closed form of the F_{MUD} for random CDMA as [63]:

$$\sigma^2_{Obv} = \frac{K-1}{N}\sigma^2_{bit} + \sigma^2_0 \tag{4.11}$$

As (4.11) shows, the F_{MUD} is a line with a constant slope of $(K-1)/N$ and y-intercept of σ^2_0.

4.2.2 Variance Transfer Function of Turbo Decoder

In the VEG method of analysis, a decoder is characterized by the error variances of its input and its normalized output as follows:

$$\sigma^2_{bit}(c) = F_{DEC}(\sigma^2{}_{Obv}(c)) \tag{4.12}$$

In contrasts to the PIC, the VT function of a decoder has to be obtained by simulation and there is no analytical approach to obtain it. In Figure 14, the simulated system for obtaining the VT function of the turbo decoder is depicted.

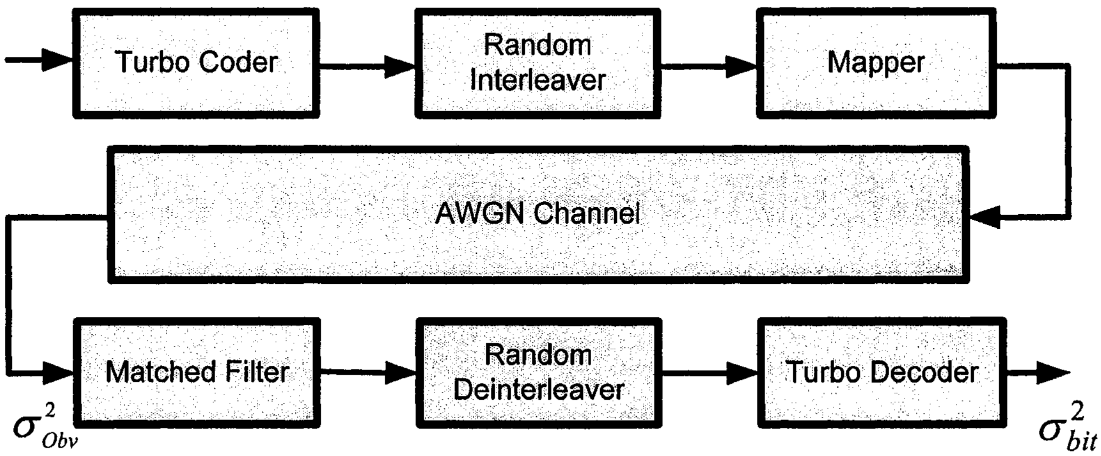

Figure 14: The simulated system for obtaining the VT function of the turbo decoder

It is worth noting that the VT function of the turbo decoder is obtained in the context of interleaved system. Therefore, the effect of interleaver and deinterleaver on the turbo decoding performance is incorporated in the obtained VT function. As we already know, interleavers and deinterleavers change the order of bits in a frame of data and do not change the variance error of their input frames. However, interleaving has a direct effect on the turbo decoder performance. By incorporation of this effect in the VT function of the turbo decoder, we can simply neglect the existence of the interleaver and the deinterleaver in the iterative loop from the error variance point of view.

The output of VT function is the error variance of normalized output of the decoder and is bounded between 0 and 1, $\sigma_{bit}^2(c) \in [0,1]$. Using MAP algorithm as SISO decoding algorithm, the proper normalization has to be done using tangent hyperbolic function.

In Figure 15, the VT function of the turbo decoder for different number of iterations is obtained by simulation. The turbo decoder decodes coded frames of length 3000. The VT curves asymptotically approaches to $\sigma_{bits}^2 = 1$ as σ_{Obv}^2 increases. The $\sigma_{bits}^2 = 1$ situation happens when the half of bits in a frame are not decoded correctly and corresponds to $BER = 0.5$.

The simulation is done for four different numbers of iterations. As the number of iteration increases, the VT curves become close to each other. For this frame length, there is almost no difference between the VT curves for more than 10 iterations.

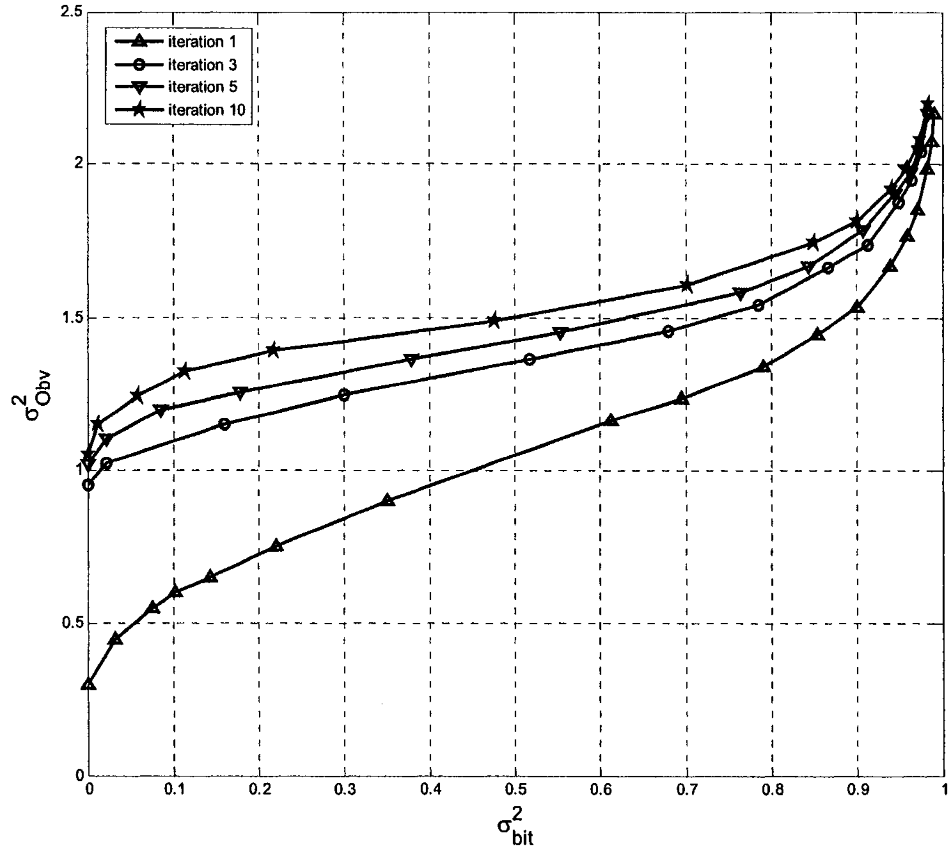

Figure 15: VT characteristic of the turbo decoder for different numbers of iteration coded frame length=3000

4.2.2.1 Frame Length Influence

Due to the usage of turbo codes in the proposed system, it is worth paying attention to the effect of frame length on the VT function of the turbo decoder. As we know, the interleaver gain is proportional with the frame size of the data. So, the frame size has a prominent effect on the performance of turbo coding. Therefore, it is worth depicting this effect by means of the VT characteristic of the turbo decoder. In Figure 16, the VT function of the turbo decoder obtained for two different frame lengths is depicted.

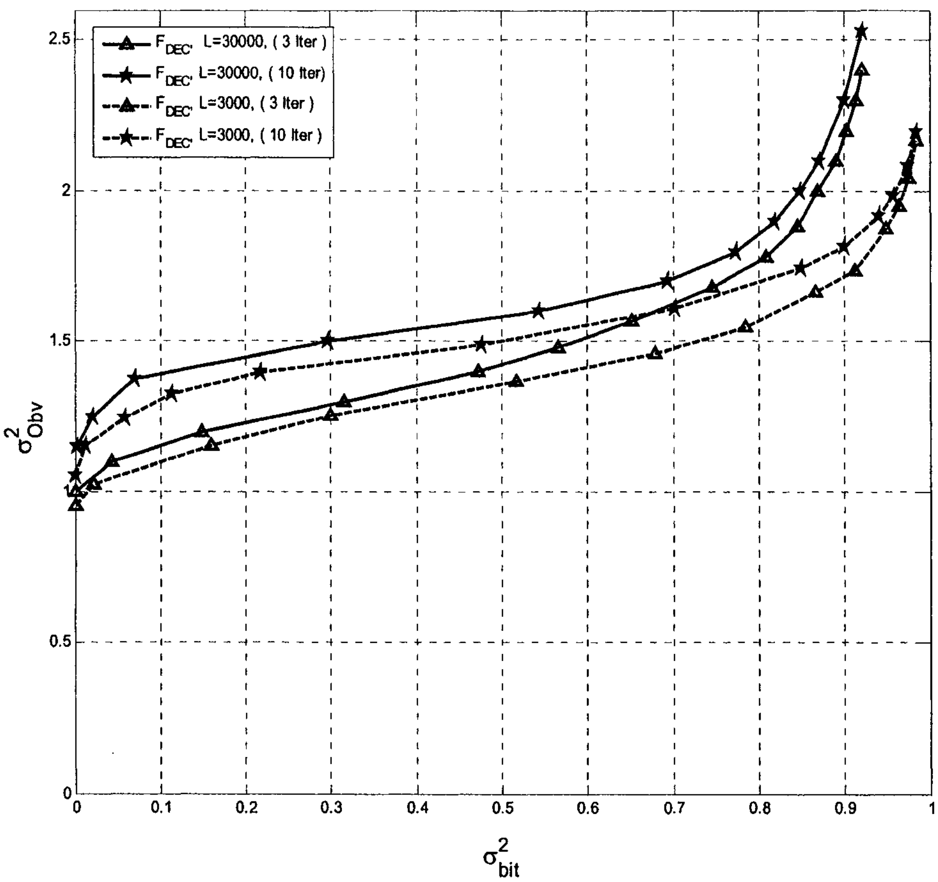

Figure 16: The influence of frame length on the VT function of the turbo decoder

Obviously, any increase in frame size makes the VT curves move upward. As we will observe in the next section, this fact indicates that long-frame systems outperform short-frame systems in the sense of convergence thresholds and convergence speed.

4.3 Variance Exchange Graph (VEG)

So far, we have constructed two functions which yield the error variances as output, given the error variances as input. Making a loop of these two functions, we can track the error variance values as a function of iterations. If the two transfer characteristics are plotted in one figure, where one of them has its axes swapped, we will have an informative graph to analyze the convergence behavior of the system, called Variance Exchange Graph (VEG). A typical VEG of the system that successfully converges to the single-user performance is shown in Figure 17 for further explanation.

The vertical axis represents σ^2_{Obv} and the horizontal axis represents σ^2_{bit}. Iterations are described by zigzag paths bouncing between the two VT characteristics, because the input of one function is the output of the other. In the zigzag path, the horizontal and vertical line segments indicate the decoder gain and the PIC gain respectively. The zigzag path starts at the initial point with the maximum value of σ^2_{Obv}, and moves forward to the equilibrium point. The ideal equilibrium point is the right vicinity of vertical axes, which is between the F_{MUD} and the F_{DEC}. At this area, σ^2_{Obv} reaches its minimum value, σ^2_0, which corresponds to the single-user performance.

The zigzag path of measured variances from an actual simulation fits between two transfer characteristics with a good approximation, and moves forward inside the convergence tunnel starting from initial point, moving forward to the equilibrium point.

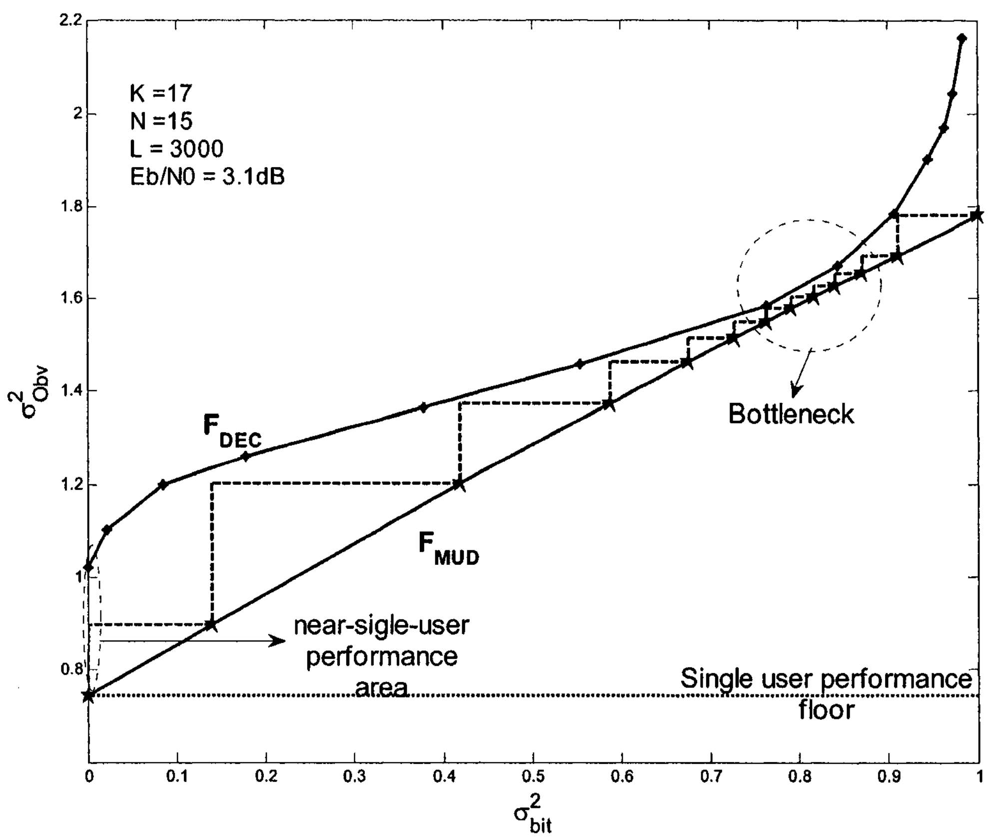

Figure 17: The VEG of the proposed system that converges to the single-user performance,

Considering the relative location of the F_{MUD} line and the F_{DEC} curve, the two characteristics can be either non-touching or intersecting. If the two characteristics intersect at some point, it means that iteration cannot improve system performance beyond that point. Conversely, if the two characteristics do not intersect each other, they form an open tunnel between themselves called "convergence tunnel". In such a case, the system can eventually reach the single-user performance after some iterations. We are mainly interested in whether or not the convergence tunnel is open because an open convergence tunnel indicates that the system is capable to approach the single-user performance.

If the system is highly loaded, the convergence tunnel narrows at a certain area. This is a very important area in the system performance, called "bottleneck". As we can see in Figure 17, the gain of the PIC and the turbo decoder decreases at the bottleneck, and more outer iterations are required to pass this area.

The relative location of the F_{MUD} line and the F_{DEC} curve contains valuable information about the convergence of the system. Considering the system parameters effect on the relative location of the two VT characteristics, we can study the possibility of the system convergence in terms of the system parameters. Moreover, we can seek proper error control coding schemes which are most compatible with other parameters of the system.

4.3.1 Convergence Thresholds

Considering the width of the bottleneck, we can define thresholds for the system parameters such as signal to noise ratio and the number of users. The threshold of a parameter is the boundary value of that parameter that keeps the convergence tunnel open.

Threshold of E_b / N_0: For given values of K and N, the convergence tunnel is open if E_b / N_0 is over a certain threshold. For signal to noise ratios lower than that, the convergence tunnel becomes closed and the system fails to converge to the single-user performance. In Figure 18 , when $K = 15$ and the received power for each user is normalized to unity, the threshold is $\sigma_0^2 = 0.82$, which corresponds to $E_b / N_0 = 2.6$. The system can hardly pass the bottleneck through many numbers of iterations. With a slight decrease in E_b / N_0 by 0.1, the F_{MUD} line intersects with the F_{DEC} curve and convergence tunnel becomes closed.

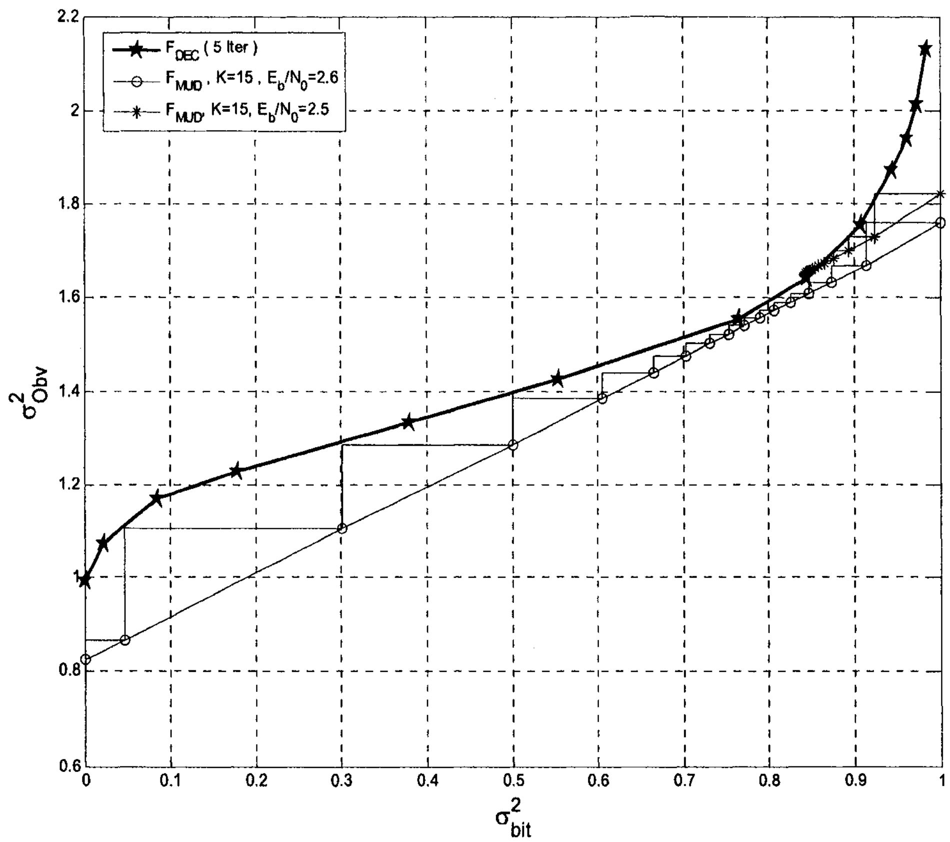

Figure 18: The threshold of signal to noise ratio

Threshold of number of users: For a given value of σ_0^2 and N, the F_{MUD} line moves upward with increasing K. We can define K^* as the maximum number of users that keeps the convergence tunnel open. In other words, K^* is the maximum number of users that the system can support at the given Eb/N0; and the value $\alpha = K^* / N$ is called "critical load" in associated literature [64]. In Figure 19, the VEG of the system with two different numbers of users is depicted.

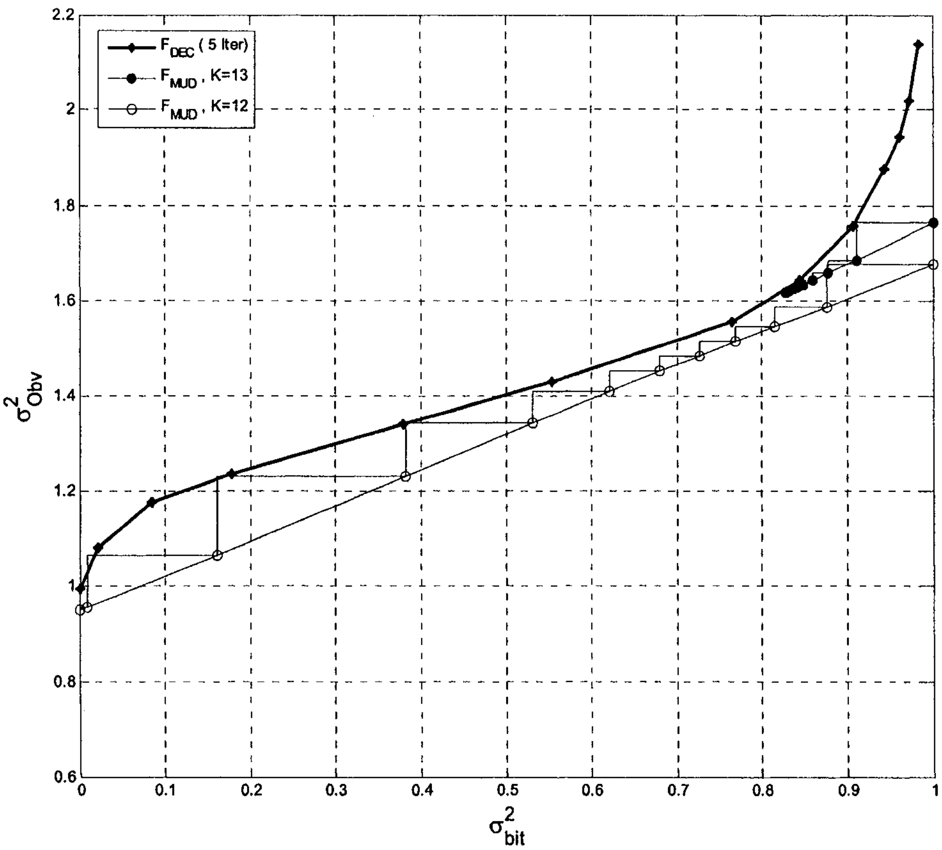

Figure 19: The threshold of number of users, Eb/N0=2.

As Figure 19 shows, when $K = 12$, the system can pass the bottleneck successfully and reach to the single-user performance after some iterations. However, when the number of users is increased by one, the F_{MUD} line intersects the F_{DEC} curve and convergence tunnel becomes closed. Therefore, for the system at $E_b / N_0 = 3$, the maximum supportable number of users is $K^* = 12$.

4.3.2 Convergence Speed

Convergence speed refers to the number of iterations required to reach the single-user performance. Drawing trajectory between two curves shows the number of required outer iteration to reach the single-user performance.

Obviously, the convergence speed highly depends on the shape of the convergence tunnel. A bottleneck in the convergence tunnel slows down the convergence of the system dramatically. The width of the convergence tunnel is dependent on the distance between the F_{MUD} and the F_{DEC}. The F_{MUD} is a line with the slope of $(K-1)/N$ and the y-intercept of σ_0^2. Any increase of either the slope or the y-intercept of the F_{MUD} makes it move upward and narrows the bottleneck. Considering the fact that bottleneck is always formed in the interval of $\sigma_{bit}^2 \in [0.7,0.9]$, we will focus on the bottleneck width for a constant initial value of σ_{Obv}^2, which originates from different values of K and σ_0^2.

Another important fact is that the gain of the PIC and the turbo decoder do not have a monotonic trend. As we can see in Figure 17 , the length of vertical and horizontal line segments zigzag path has a descending trend from the beginning to the bottleneck, whereas it turns to an ascending trend after passing the bottleneck.

Figure 20 illustrates the convergence behavior of the system with three different configurations for initial σ_{Obv}^2. For aesthetic purposes, the zig-zag paths are not shown and only the intersecting points of the zig-zag paths are shown in Figure 20 .

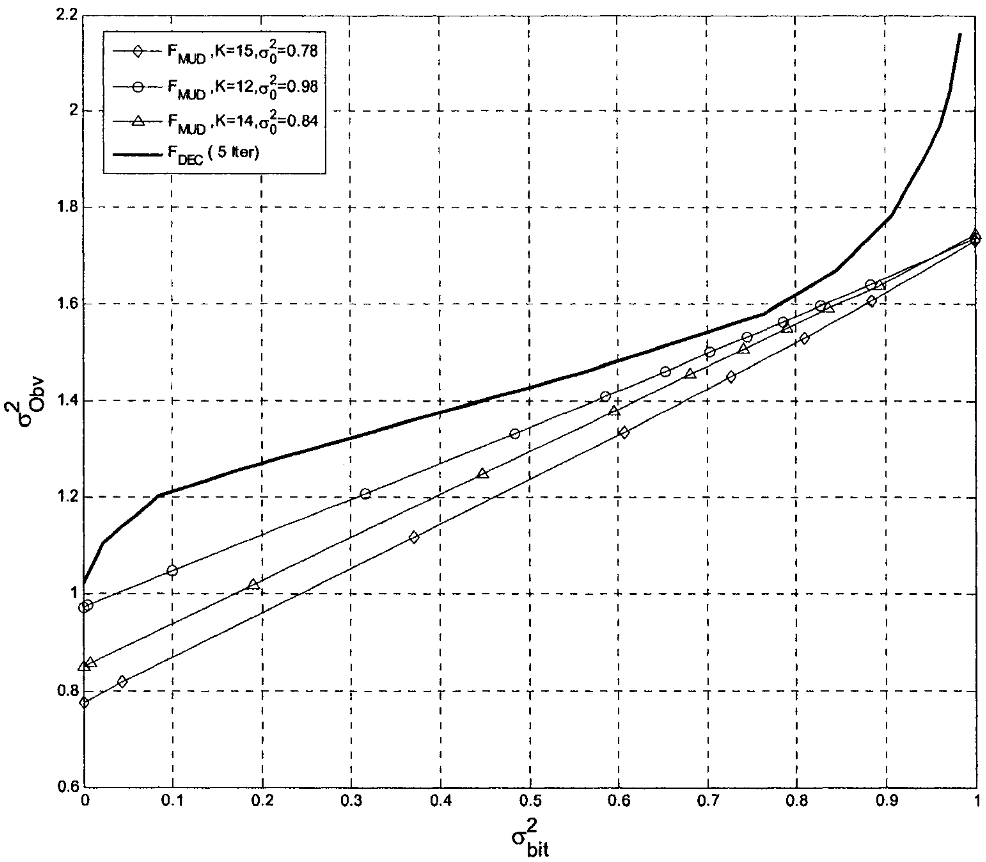

Figure 20: Convergence speed of the system with three different configurations for initial σ^2_{Obv}

As the figure shows, for an equal initial value of $\sigma^2_{Obv} = (K-1)/N + \sigma_0^2$, the convergence of the system speeds up as the contribution of MAI to initial σ^2_{Obv} increases. Conversely, it slows down as the contribution of channel noise to initial σ^2_{Obv} increases. The obtained results from simulations are summarized in the table below for comparison. Considering (4.11), the $(K-1)/N$ parameter represents MAI weight whereas σ_0^2 indicates channel noise weight.

Table 2: The effect of MAI and noise on the convergence speed of the system

Initial value of σ^2_{Obv}	Number of users (K)	$\frac{K-1}{N}$	Channel noise variance (σ_0^2)	Number of iterations
1.71	12	0.73	0.98	12
1.71	14	0.87	0.84	9
1.71	15	0.93	0.78	7

4.3.3 Iteration Schedule

In the system of Figure 13, we face with two different types of iteration in the proposed system. The first type is the iteration between the PIC and the turbo decoders, called "outer iteration". The second type is the iteration inside the turbo decoders that we refer to as "inner iteration".

In the proposed system, iteration schedule refers to the number of inner iterations per outer iteration. Iteration schedule has a prominent role in the convergence behavior of the system and has to be chosen to obtain the best possible result.

4.3.3.1 Fixed Iteration Schedule

The simplest choice is choosing a constant number of inner iterations per outer iteration. This is the choice that has been made in all the associated literature so far. Considering the VT characteristic of the turbo decoder for different numbers of iterations in Figure 15, we find out that the system can converge to the single-user performance with five and more inner iterations per outer iteration, whereas the system fails to converges with three inner iterations per outer iteration. The reason of such a failure is the fact that the data estimation provided by the decoders is not accurate enough. When such data are fed to the PIC, low quality MAI estimation is reconstructed and then subtracted from output of the matched filter. It results in propagation error of one user to the others, and makes the system fail to converge to near-single-user performance.

The task now is to find the best fixed iteration schedule between the possible choices. It is better to choose the lowest number of inner iterations that prevents a narrow bottleneck in the convergence tunnel. In Figure 21, with choosing five inner iterations, almost half of outer iterations are consumed to pass the bottleneck, whereas with choosing 10 inner iterations and widening the bottleneck, it is predictable that the system converge quicker with less number of outer iterations and total number of inner iterations.

As the VEG of the system shows in Figure 21, any increase in the number of inner iterations per outer iteration is mainly beneficial in the area of bottleneck. This fact directs us to a novel idea for iteration schedule discussed in the next section called "smart iteration schedule".

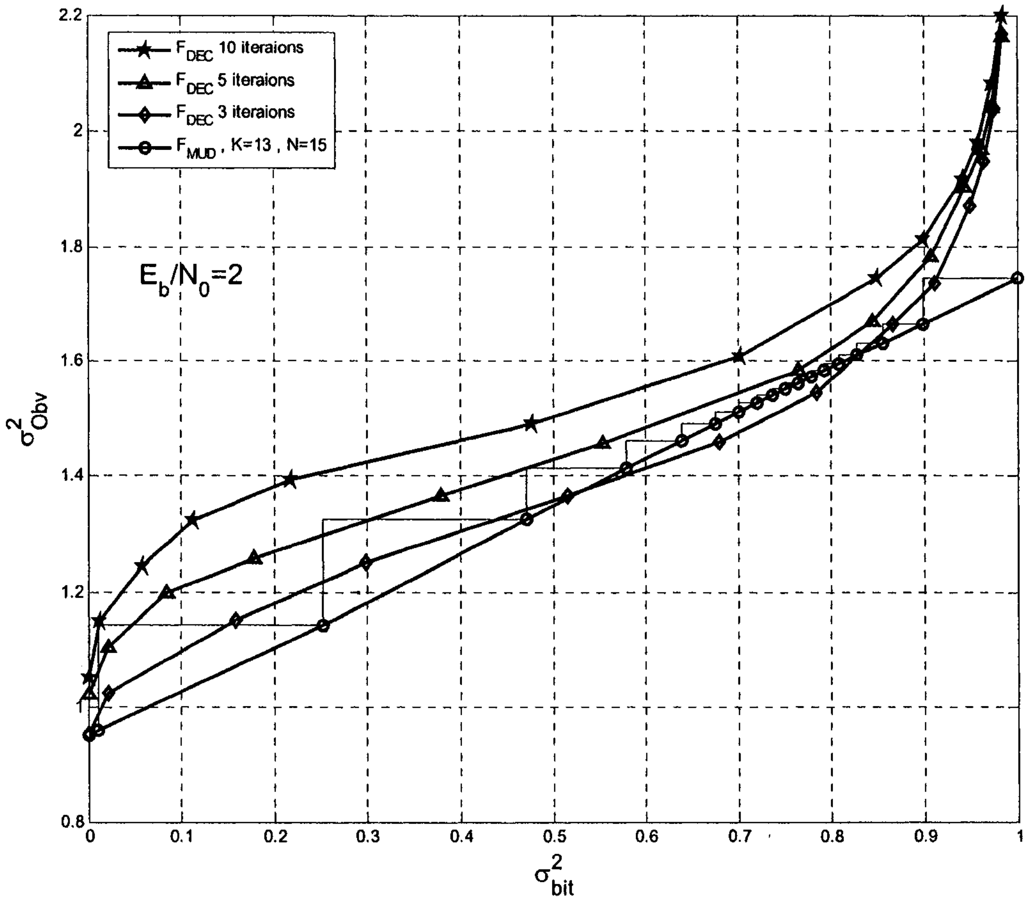

Figure 21: Convergence to near-single-user performance using fixed iteration schedule

4.3.3.2 Smart Iteration Schedule

An efficient technique to make better use of iterative turbo decoding is choosing variant number of inner iterations per outer iteration. In each outer iteration, we change the number of inner iterations per outer iteration to reach the best possible gain for the PIC and the turbo decoders. Such a smart technique speeds up the system convergence to the ideal performance with a fewer numbers of inner and outer iterations compared with fixed iteration schedule approach. Figure 22 illustrates that different iteration schedules with different convergence behaviors.

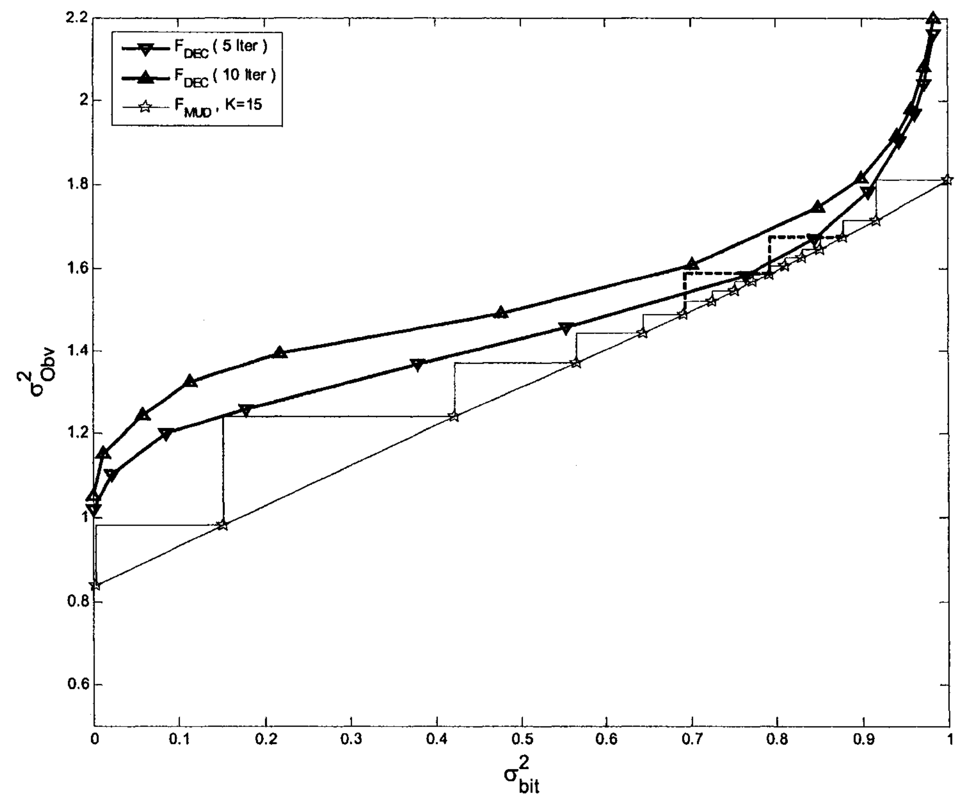

Figure 22: Convergence of the proposed system using two different iteration schedules

It is obvious that having a bottleneck in the convergence tunnel increases the number of iterations dramatically and most of the iterations will be used to pass the bottleneck. In smart iteration technique, the number of inner iterations per outer iteration is increased through the passing of the bottleneck. It means that when the system reach the bottleneck area, it switches to the next F_{DEC} curve to widen the bottleneck, and after passing the bottleneck the number of inner iterations per outer iteration will be restored. This technique increases the speed of the convergence process while passing the bottleneck, and therefore it speeds up the whole convergence process to a great extent.

As we can observe in Figure 22, using fixed iteration schedule, it takes 15 outer iterations with five inner iterations per each to reach the near-single-user performance. In the smart iteration schedule technique, we only increase the number of inner iterations from five to ten throughout the bottleneck. It makes the system pass the bottleneck by less total number of iterations than the previous choice, leading to a much quicker convergence. The comparison between the two approaches is summarized in the table below.

Table 3: Comparison between fixed and smart iteration scheduling

Iteration Schedule	Number of iterations for passing the bottleneck		Total number of iterations	
	Outer	Inner	Outer	Inner
Fixed	8	$5\times 8 = 40$	15	$5\times 15 = 75$
Smart	2	$2\times 10 = 20$	9	$(5\times 7)+(2\times 10) = 55$

Comparing the number of total inner and outer iterations shows that the smart iteration schedule can benefit the system by a dramatic decrease in the number of both inner and outer iterations.

It is worth noting that the efficiency of the smart iteration schedule is proportional with the width of the bottleneck. When the convergence tunnel contains a narrow bottleneck, increase in number of inner iterations widen the bottleneck in return. Conversely, when there is no bottleneck and convergence tunnel is already open enough, it is not worth to increasing the number of inner iteration because the achieved gain would be negligible. Figure 23 compares two different situations in which the smart iteration schedule acts differently.

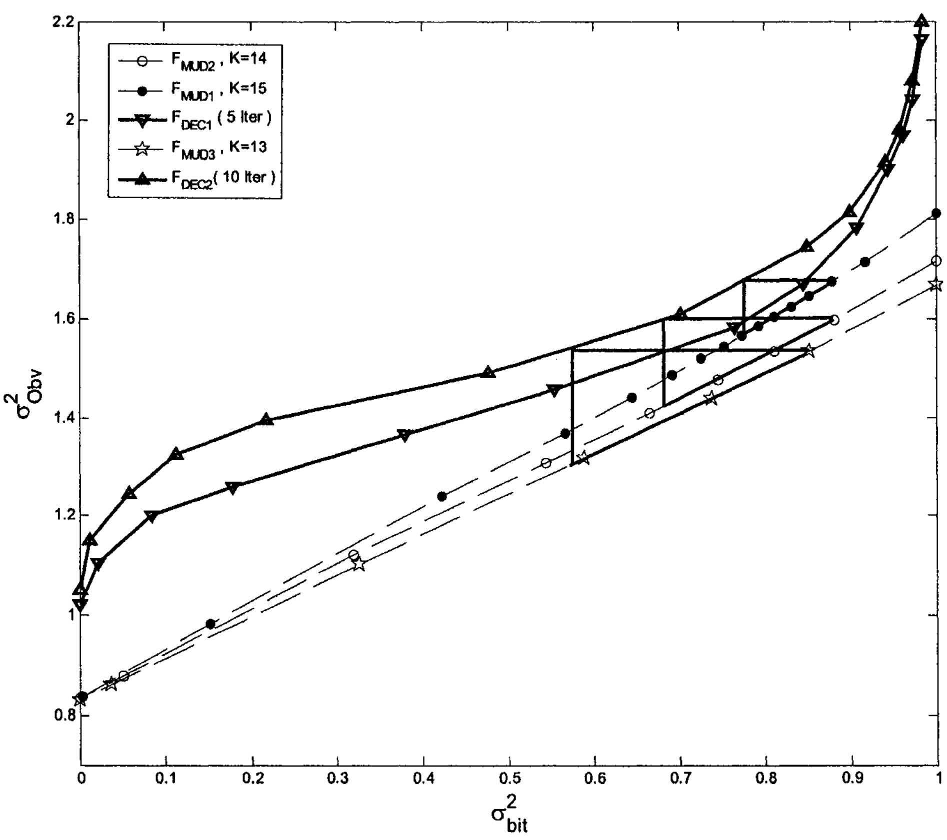

Figure 23: Smart iteration schedule for different shapes of the convergence tunnel

For the F_{MUD1}, there is a narrow bottleneck and the smart iteration schedule is very advantageous and skips 5 outer iterations. As the convergence tunnel becomes wider, the smart iteration schedule becomes less advantageous and skips only 2 outer iterations for the F_{MUD3}.

4.4 IMUD-PCCC verses IMUD-SCCC Structures

It is worth comparing the proposed system with the most similar system in previous literatures, which is an IMUD-SCCC system analyzed by Shi & Schlegel in [10]. Both systems are similar from the structure point of view and parameters. The main difference is that in Shi's system, SCCC are used, whereas in our system, PCCC (turbo codes) are employed. Both of the SCCC and PCCC include iterative decoding process, and are considered as powerful coding schemes.

In this section, we compare these two systems from two different angles. First, we compare different descriptions that the VEG provides for each of them. Second, we compare the convergence behavior of the two systems by means of the convergence tunnel shape.

In Figure 24, the structure of an IMUD-SCCC and an IMUD-PCCC are depicted for a better understanding. In [10], the iterative system is split to two loops. Loop1 consists of the PIC and the inner APP; and loop 2 is a typical SCC decoder, including the inner APP decoder and the outer APP decoder. The VT function of each loop is obtained by simulation and then plotted in one VEG.

In our approach, the iterative structure is split in a different and more efficient way. The analytically-gained VT function of the PIC forms one of the VEG constituent curves. The empirically-gained VT function of the turbo decoder with external interleaver forms the other constituent curve of the VEG. The approach used in this thesis has the following advantages to the approach used by Shi:

- In Shi's approach, both of the curves depend on the number of iterations. That complicates the design of a successful iteration schedule. Whereas, in our approach, the VT characteristic of PIC is independent of the number of iterations.

- In Shi's approach both of the curves have to be obtained by simulation. Whereas in our approach, the VT characteristic of the PIC can be analytically obtained. Therefore, it would be possible to predict the convergence behavior of the system.

- In Shi's approach both of the VT characteristics are curves. This complicates the observation of the effect of change in parameters on the VT characteristics. However in our approach, one of the VT characteristics is a straight line, and system parameters are related to its slope and y-intercept. Therefore, it is easily possible to observe the influence of parameters on the system convergence.

• In Shi's approach, the two VT functions are not independent of each other because the inner APP decoder is inside both of the loops. So, any change in this component makes both of VT functions varied, which complicates the study of the system.

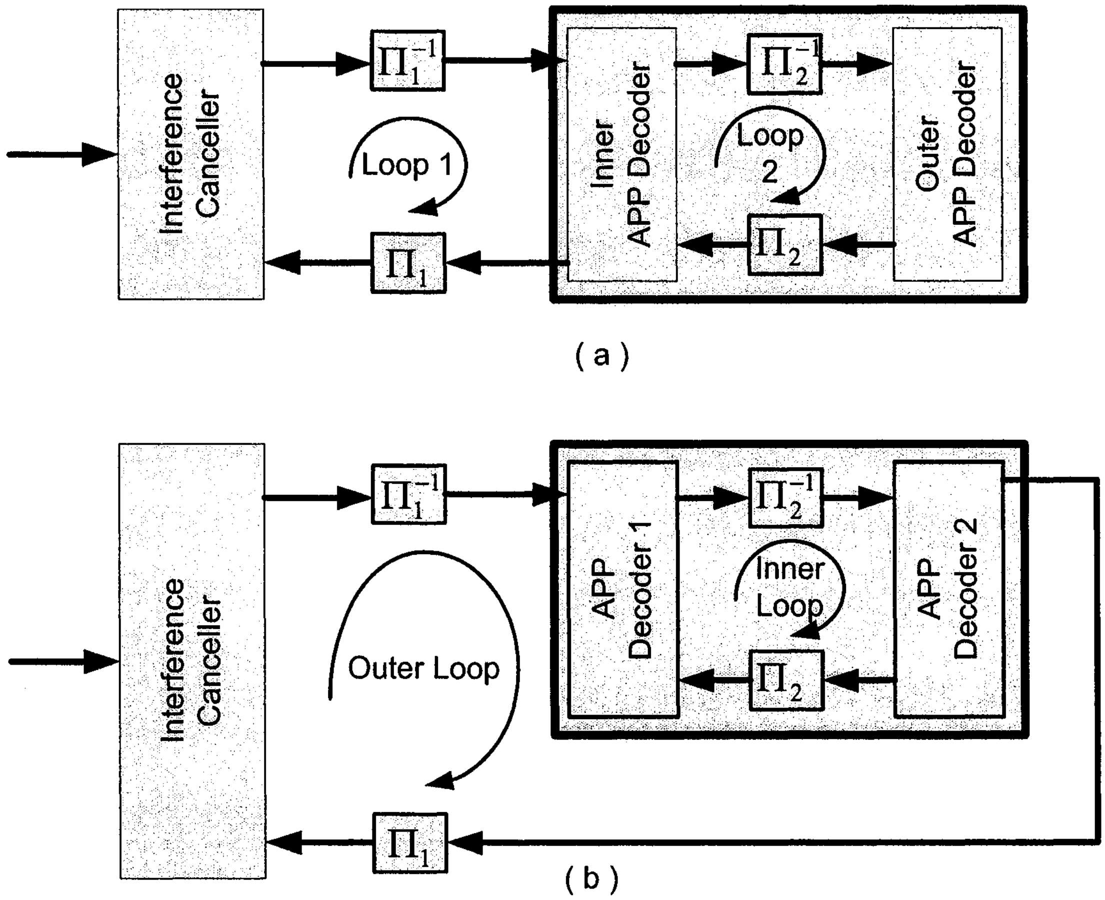

Figure 24: IMUD-SCCC (a) verses IMUD-PCCC (b) structures

In Figure 25 and Figure 27, the upper curve is the VT characteristic of loop 1 and the lower curve is the VT characteristic associated with loop 2. Note that both of the two constituent curves of the VEG are varied with changing values of K, N. We consider the two following cases for comparison between the two systems: a full-loaded system ($\alpha = 1$) at low SNR ($E_b / N_0 = 2.88$ dB), and a double-loaded system ($\alpha = 2$), at high SNR ($E_b / N_0 = 10.65$ dB).

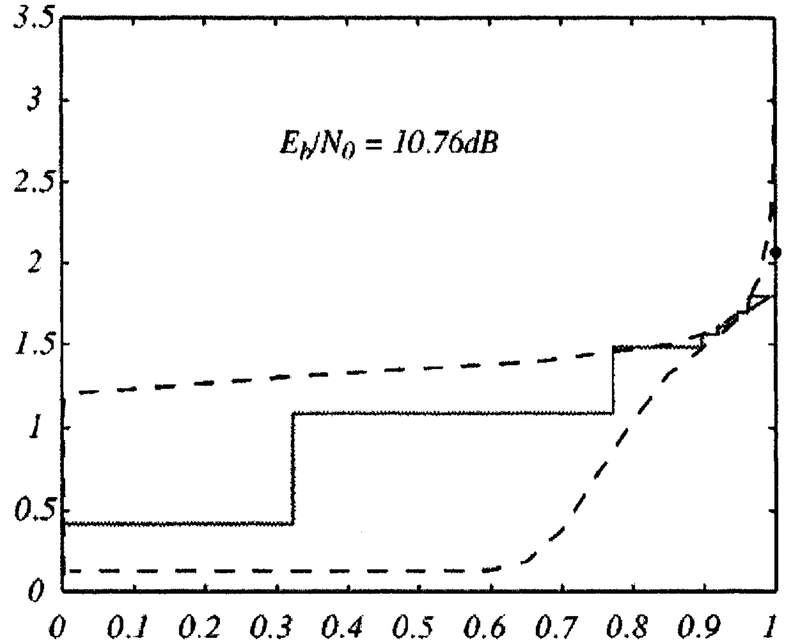

Figure 25: The VEG of the IMUD-SCCC CDMA system $K = 20, N = 10$ **[10]**

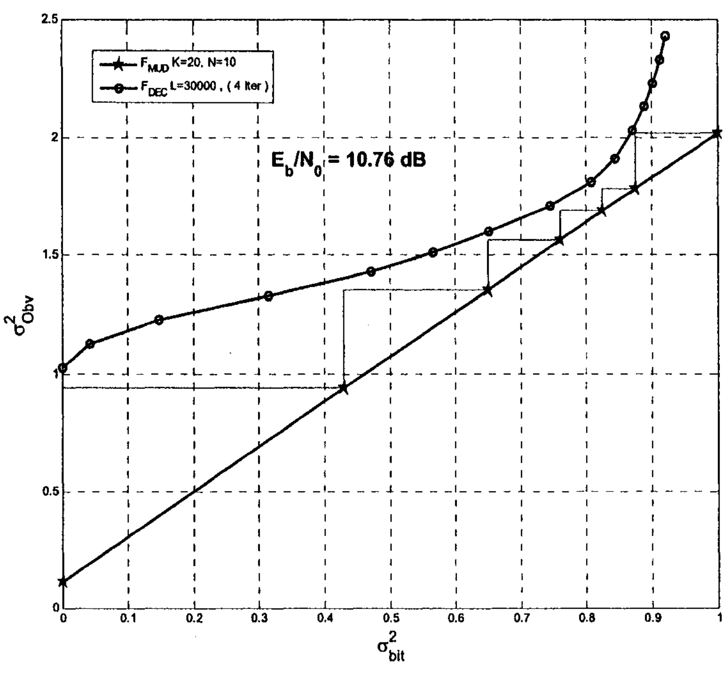

Figure 26: The VEG of the IMUD-PCCC CDMA system, $K = 20, N = 10$

In order to make a meaningful comparison between the two structures, we have to consider the shape of the convergence tunnel and the variation of the convergence tunnel shape with the change in parameter values. Considering the shape of the convergence tunnel, we will find that the systems in Figure 25 and Figure 27 have already reached narrow bottlenecks, whereas, the systems in Figure 26 and Figure 28 have rather broad bottlenecks at the same SNR level and system load value. The comparison between the results from Shi and the simulated results is summarized in the table below.

Table 4: Comparison between the IMUD-SCCC and the IMUD-PCCC structures

Structure	System load $\alpha = \frac{K}{N}$	SNR (dB)	Inner Iteration	Total number of iterations
IMUD-SCCC	$\alpha = \frac{20}{10} = 2$	10.76	4	8
IMUD-PCCC	$\alpha = \frac{20}{10} = 2$	10.76	4	6
IMUD_SCCC	$\alpha = \frac{15}{15} = 1$	2.88	4	7
IMUD-PCCC	$\alpha = \frac{15}{15} = 1$	2.88	4	5

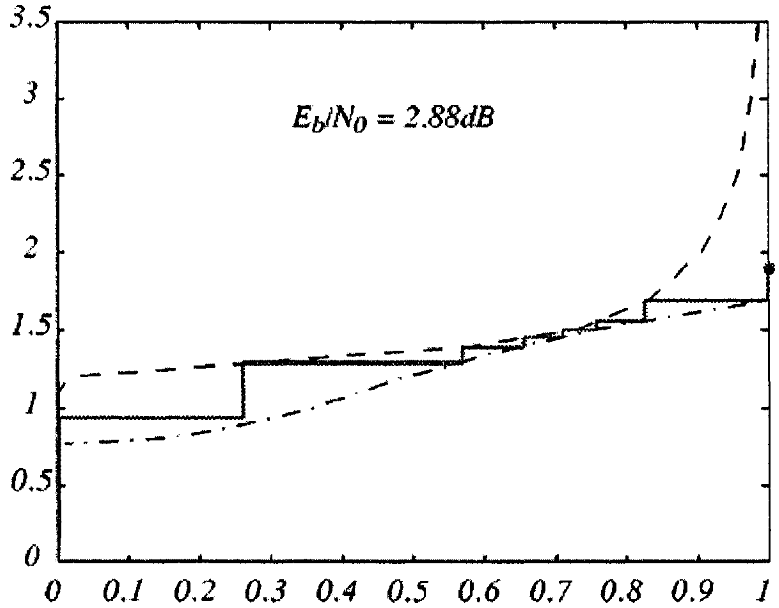

Figure 27: The VEG of the IMUD-SCCC CDMA system [10]

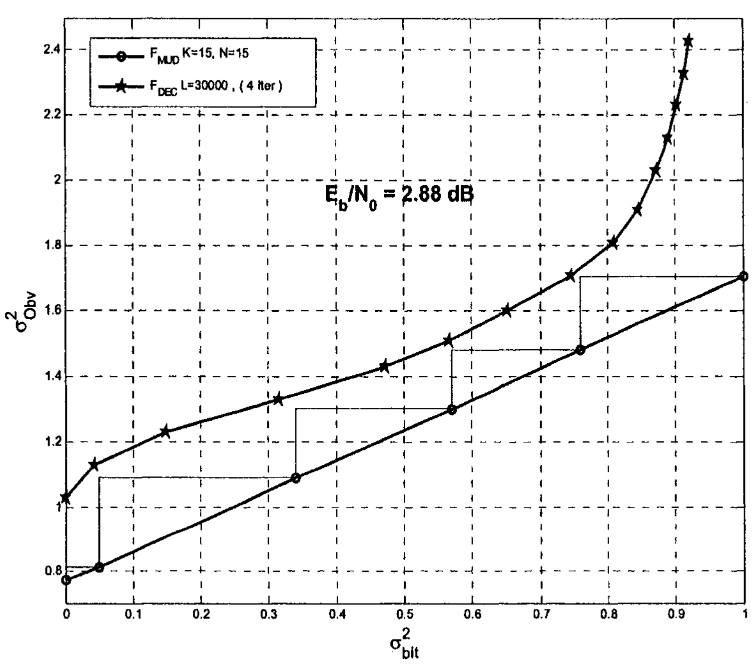

Figure 28: The VEG of the IMUD-PCCC (turbo) CDMA system

Comparing the results in table 2 and the shape of the convergence tunnel, reveals that IMUD-PCCC structures marginally outperform and converge quicker than IMUD-SCCC structures in heavily loaded systems, $(\alpha \geq 1)$.

4.5 Summary

In this chapter, different types of analysis are done on iterative multi-user detection turbo-coded CDMA systems using the VEG. It is shown that the VEG can be used as a quite efficient tool. The convergence thresholds of the system to single-user performance in terms of system parameters are predicted and then validated by simulations. After studying the condition of a successful convergence, we investigated the convergence speed and the parameters that affect it.

Iteration schedule is a topic that is discussed in this chapter. The optimum fixed iteration schedule, from the convergence speed point of view, can be chosen easily by the VEG. Studying different fixed iteration schedules, the novel idea of smart iteration schedule is presented that speeds up the system convergence to a great extent.

A comparison has been done between the SCCC-based and turbo-coded based IMUD systems. It is shown that IMUD turbo-coded systems marginally outperform IMUD-SCC coded systems.

5 Conclusions and Future Work

In the introduction chapter of this thesis several questions were stated to indicate the objectives of this research work. Although the answer to all questions can be found in the pervious chapter in terms of system parameters, we generally answer them here to highlight the achievements of this thesis.

The first question was "on what conditions in terms of system parameters, an iterative multi-user turbo-coded CDMA system can approach the single-user performance?". The answer to this generic question is the description that the VEG provides for the convergence behavior. That convergence behavior highly depends on the shape of convergence tunnel. If the convergence tunnel is open, the equilibrium point will be in the near-single-user performance area. Conversely, if the convergence tunnel is closed, the equilibrium point will be the blockage point of the convergence tunnel. Therefore, the answer to this question is highly dependent on the state of the convergence tunnel (open or closed). Any choice of system parameters that leads to an open convergence tunnel on the VEG, guarantees that the system can eventually reach the single-user performance.

The second question was "in case of convergence, how fast does the system approach ideal convergence point?" The answer to this question is in the "Convergence Speed" section of chapter four. To summarize, the speed of convergence is highly dependent on how much of the observed error variance originates from MAI and how much originates from channel noise. The system tends to converge more quickly, when a high percentage of observed error variance originates from MAI rather than from channel noise.

The third question was "what is the optimum iteration pattern between outer iterations and inner iterations?" The answer to this question is the smart iteration schedule that prevents the dynamic of the system to become engaged in narrow bottlenecks. Such a schedule can dramatically speed up the system convergence.

The fourth question was "what is the difference between the IMUD-SCCC system and the IMUD-PCCC system from convergence point of view?". The answer to this question is that IMUD-PCCC structures marginally outperform the IMUD-SCCC structures in both high and low level of signal to noise ratio. We came to this conclusion comparing the convergence tunnel shape in different values of SNR.

The future research work of this thesis can be developed in following directions:

- Using a MMSE filter in concatenation with the PIC and investigating the convergence behavior of the new system for highly loaded state,

- Using Partial Parallel Interference Canceller (PPIC) instead of PIC and studying the convergence behavior of the new system,

- Applying the idea of smart iteration schedule to IMUD-SCC coded CDMA system,

- Investigating the effect of the constraint length of turbo coders on the VT function of the turbo decoder and the convergence of the system ,

- Investigating the near-far effect on the system convergence behavior,

- Applying the strategy of splitting the iterative structure in this thesis to IMUD-SCCC for obtaining a more efficient VEG for IMUD-SCC coded system than that of Shi's,

- Investigating the accuracy of the VEG analysis with respect to the data frame length.

Bibliography

[1] J. Viterbi, *CDMA: Principle of Spread Spectrum Communication*, Addison-Wesley Wireless Communication, 1995.

[2] R.L. Pickholtz, D.L. Schilling and L.B. Milstein. "Theory of Spread-Spectrum Communications-A Tutorial," *IEEE Transactions on Communication*, 30(5):855-884, May 1982.

[3] R.C. Dixon. "Spread Spectrum Systems," Wiley, New York, second edition, 1984.

[4] R. Kohno, R. Meidan, and L.B. Milstein. "Spread Spectrum Access Methods for Wire – less Communication," *IEEE Communication Magazine*, pp. 58-67, January 1995.

[5] S. Verdu. "Minimum Probability of Error for Asynchronous Gaussian Multiple-Access Channels," *IEEE Transaction on Information Theory*, IT-32:85-96, January 1986.

[6] M.H. Moghari, B. Shahrrava. "Convergence Behaviour of Iterative Multiuser Detection Algorithms," *IEEE Wireless Communication and Networking Conference*, March 2005,vol.1, pp. 326-332.

[7] K. Li, X. Wang,"EXIT chart Analysis of Turbo Multiuser Detection," *IEEE Transactions on Wireless Communication*, vol. 4, pp 300-301, Jan 2005.

[8] V. Kuhn, "Analysis of iterative Multi-user Detection Schemes with EXIT Chart," *IEEE Eighth International Symposium on Spread Spectrum Techniques and Applications,* 30 Aug. - 2 Sept. 2004, pp 535-539.

[9] P. Alexander, A. Grant and M. Reed, " Iterative Detection in Code-Division Multiple-Access with Error Control Coding," *European Transaction on Telecommunications, Special Issue on CDMA Techniques for Wireless Communications Systems*, vol. 9, pp. 419-426, Sep.-Oct.1998.

[10] Z. Shi, Ch. Schlegel, " Joint Iterative Decoding of Serially Concatenated Error Control Coded CDMA," *IEEE Journal of Selected Areas in Communications*, vol.19, no. 8, August 2001.

[11] E.O. Bejide, F. Takawira, "Convergence analysis of iterative interference cancellation receivers for turbo-coded DS-CDMA systems," *7th IEEE AFRICON Conference in Africa,* , 2004, vol. 1, pp. 197-202.

[12] S. Sadjarpour, N. J. A. Sloane, M. Salehi, G. Nebe, "Interleaver Design for Turbo Codes," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 5, May 2001.

[13] R.L. Pickholtz, D.L. Schilling and L.B. Milstein, "Theory of Spread-Spectrum Communications-A Tutorial," *IEEE Transactions on Communication*, 30(5):855-884, May 1982.

[14] R.L. Pickholtz, L.B. Milstein and D.L Schilling. "Spread Spectrum for Mobile Communications," *IEEE Transactions on Vehicular Technology*, 40(2):313-321, May 1991.

[15] H. Meyr, M. Moeneclaey and S. Fechtel, *Digital Communication Receivers*, Wiley, New York, 1998.

[16] J.G. Proakis. *Digital Communications.* McGraw Hill, fourth edition, 2001.

[17] T. R. Giallorenzi and S. G. Wilson, "Suboptimal multi-user receivers for convolutionally coded asynchronous DS-CDMA systems," *IEEE Transactions on Communications*, vol. COM-44, no. 9, pp. 1183-1196, Sept.1996.

[18] M. Moher, "An iterative multiuser decoder for near-capacity communications," *IEEE Transactions on Communications*, vol. 47, pp. 870-880, July 1998.

[19] M. C. Reed, C. B. Schlegel, P. D. Alexander and J. A. Asenstorfer, "iterative multiuser detection for CDMA with FEC: near-single-user performance," *IEEE Transactions on Communications,* vol. 46, no 12, pp. 1693-1699, Dec.1998.

[20] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on Communications*, vol. 47, no. 7, pp. 1046-1061.

[21] C. Berrou, A. Glavieux, and P. Thitimasjshima, "Near Shannon limit error- correcting coding and decoding :Turbo-codes, " *in Proceeding IEEE International Conference on Communication,*(Geneva, Switzerland) , May1993, pp. 1064-1070.

[22] C. Huang, "An analysis of CDMA 3G wireless communications standards," in *Proceedings of the 49-th Vehicular Technology Conference*, (Houston, Texas), 1999, pp. 342-345.

[23] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *European Transaction on Telecommunication.*, vol. 8, pp. 119-125, March/April 1997.

[24] D. Divsalar and F. Pollara, "Multiple Turbo Codes for Deep-Space Communications". *TDA Progress Report 42-121*, pp. 66-77, May 15 1995.

[25] S. Bendetto and G. Montorsi. "Serial concatenation of block and convlutional codes". *IEE Electronics Letters*, 32(10): 887-888, May 1996.

[26] S. Benedetto. "Serial concatenation of interleaved codes: performance analysis, design and iterative decoding," In *ISIT, Ulm Germany*, June 29-July 4 1997.

[27] S. Benedetto and G. Montorsi, "Iterative decoding of serially concatenated convolutional codes," *IEE Electronics Letters*, 32(13):1186-1188, June 1996.

[28] C. Berrou and A. Glavieux. "Near Optimum Error Correcting Coding and Decoding: Turbo-Codes," *IEEE Transactions on Communications*, 35(12):1261-1271, October 1996.

[29] S. Dolinar and D. Divsalar, "Weight distribution for turbo codes using random and nonrandom permutations," *TDA Progress report*, pp. 42-122, Aug. 1995.

[30] W. E. Ryan. "A Turbo Code Tutorial," New Mexico State University, www.eccpage.com/turbo2c.ps,

[31] B. Skalar, "Fundamentals and Applications of Digital Communications," Second edition, pp 495-497.

[32] A.J. Viterbi. "Convlutional Codes and Their Performance in Communications Systems," *IEEE Transactions on Communications Technology*, COM-19(5): 751-772, October 1971.

[33] L.R. Bahl, J. Cocke, F. Jelinek, and J. Raviv. "Optimal decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Transactions on Information Theory*, It-20:284-287, March 1974.

[34] J. Hagenauer, P. Robertson and L. Papke, "Iterative Turbo decoding of systematic convolutional codes with the MAP and SOVA algorithms," *ITG Conference on Source and Channel Coding Proceeding*, Munich, Oct. 1994.

[35] P. Robertson, E. Villeburn, P. Hoeher,"A comparison of optimal and sub-optimal MAP decding algorithms operating in the log domain," *IEEE International Conference on Communications*, Seattle, WA, June 1995, pp 1009-1013.

[36] D. Divsalar and F. Pollara, "Turbo codes for PCS applications," *Proceedings of IEEE International Conference on Communications,* ICC' 95, pp.54-59, Seattle, Washington, June 1995.

[37] R. Tanner, D.G.M Cruickshank, C.Z.W. Hassell Sweatmann, and B. Mulgrew. "Receivers for Nonlinearly Separable Scenarios in DS_CDMA," *IEE Electronics Letters*, 33(25):2103-2105, December 1997.

[38] L. Wei, L.K. Rasmussen, and R. Wyrwas. "Near Optimum Tree Search Schemes for Bit Synchronous Multiuser CDMA Systems over Gaussian and Two Path Rayleigh-Fading Channels," *IEEE Transactions on Communications*, 45:691-700, June 1997.

[39] E.A. Al-Susa and D.G.M. Cruickshank. "Pre-selection Based Reduced Complexity MLMUD for DS-CDMA Systems," *IEEE Vehicular Technology Conference, VTC 2001 Spring,* pp. 491-495.

[40] A. AlRustamani. "Greedy Multiuser Detection over Single-Path Fading Channel". In *Proceedings of the 5th IEEE International Symposium on Spread Spectrum Techniques and Applications (ISSSTA),* NY, USA, , September 2000, vol. 1, pp. 708-712.

[41] R. Lupas-Golaszewski and S. Verdu. "Asymptotic Efficiency of Linear Multi-User Detectors," In *Proc. 25th Conf. on Decision and Control,* Athens, Greece, December 1986, pp. 2094-2100.

[42] S. Verdu. *Multiuser Detection.* Cambridge University Press, 1998.

[43] J.M. Holtzman, "DS/CDMA successive interference cancellation," *IEEE International Symposium on Spread Spectrum Techniques and Applications*, July 1994, pp. 69-78.

[44] R. M.Buehrer, N.S. Correal-Mendoza, and B.D. Woerner, "A simulation comparison of multiuser receivers for cellular CDMA," *IEEE Transaction on Vehicular Technology,* vol. 49, no.4, pp.1065-1085, July 2000.

[45] D. Guo, L.K. Rasmussen, and T.J. Lim, "Linear parallel interference cancellation in long-code CDMA multiuser detection", *IEEE Journal on selected Area in Communications,* vol.17, no.12, pp.2074-2081, Dec 1999.

[46] A. Nahler, R. Irmer, and G. Fettweis, "Reduced and differential parallel interference cancellation for CDMA systems," *IEEE Journal on Selected Areas in Communications,* vol. 20, no .2, pp.237-247, Feb 2002.

[47] D. Divsalar, M.K. Simon, and D. Raphaeli, "Improved parallel interference cancellation for CDMA," *IEEE Transactions on Communications*, vol. 46, no. 2, pp.258-268, Feb 1998.

[48] G. Xue and J. Weng, " Adaptive multistage parallel interference cancellation for CDMA over multipath fading channels," *IEEE Vehicular Technology Conference (VTC 1999)*, July 1999, pp.1251-1255.

[49] R.M. Buehrer and S. P. Nicoloso, "Comments on partial parallel interference cancellation for CDMA," *IEEE Transaction on Communications*, vol. 47, pp. 658-661, May 1999.

[50] D. Divsalar, and M. K. Simon, "Improved CDMA performance using parallel interference cancellation," *Tech. Rep. 95-21, JPL Publication*, Oct. 1995.

[51] J. Hagenauer," The Turbo Principle: *Tutorials Introduction and State of the Art" International Symposium on Turbo Codes*, Breast-France, 3-5 September, 1997 pp. 1-11.

[52] H. E. Gamal, E. Geraniotis, " Iterative Multiuser Detection for Coded CDMA Signals in AWGN and Fading Channels," *IEEE Journal on Selected Areas in Communications* , vol. 18, no. 1, pp.30-41, January 2000.

[53] J.M. Hsu and C. Wang, " A low- complexity Iterative Multiuser Receiver for Turbo-Coded DS-CDMA Systems," *IEEE Journal on Selected Areas in Communications,* vol. 19, no. 9, pp. 1775-1783, September 2001.

[54] S. Marinkovic, B. Vucctic and A. Ushirokawa, "Space-Time Iterative and Multistage Receiver Structure for CDMA Mobile Communication Systems," *IEEE Journal on selected Areas in Communications*, vol. 19, no. 8, pp. 1594-1604, August 2001.

[55] X. Wang and H. V. Poor, "Iterative (Turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on Communications*, vol. 49, no. 7, pp. 1046-1061, July 1999.

[56] Jia Shen, Alister G. Burr, "Joint Turbo Decoding and MMSE-based Parallel Interference Cancellation for CDMA Systems," *EPSRC Postgraduate Research in Electronics, Photonics, Communications and Software(PREP)*, Nottingham, England, April 2002.

[57] Q. Li, "Multi-user detection for turbo coded DS/CDMA," *Ph.D. Dissertation, Texas A&M University*, Appendix B, May 2001.

[58] S. ten Brink, "Design of serially concatenated codes based on iterative decoding convergence," *Proceedings 2^nd International Symposium on Turbo Codes and Related Topic,* pp. 319-322, Brest, France, September 2000.

[59] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp.1727-1737, October 2001.

[60] P.D. Alexander, M.C. Reed, J.A. Asenstorfer and S.B. Schlegel, "Iterative multiuser interference reduction: turbo CDMA," *IEEE Trans. Commun.* , vol. 47, no 7, July 1999.

[61] M. C. Reed, C. B. Schlegel, P.D. Alexander and J. A. Asenstorfer, "Iterative multiuser detection for CDMA with FEC: near-single-user performance," *IEEE Trans Commun.*, vol. 46, no. 12, pp. 1693-1699, Dec. 1998.

[62] D. Tse and S. Hanly, "Linear multiuser receivers: effective interference, effective bandwidth and user capacity," *IEEE Trans. Inform. Theory,* pp. 641-657, March 1999.

[63] P. Alexander, A. Grant, and M. Reed, "Iterative detection in code-division multiple-access with error control coding," *Eur. Trans. Telecommun.*, vol. 9, no. 5, pp. 419-426, Sept. 1998.

[64] F. Vanhaverbeke, M. Moeneclaey, "Evaluation of critical load of parallel interference cancellation based on the PDA approach," *14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications*, vol. 2, pp. 1551-1554, 7-10 Sept. 2003.