

## Chapter 1. Introduction

Over the years, there has been a tremendous growth in digital communications especially in the fields of cellular/PCS, satellite, and computer communication. In these communication systems, the information is represented as a sequence of binary bits. The binary bits are then mapped (modulated) to analog signal waveforms and transmitted over a communication channel. The communication channel introduces noise and interference to corrupt the transmitted signal. At the receiver, the channel corrupted transmitted signal is mapped back to binary bits. The received binary information is an estimate of the transmitted binary information. Bit errors may result due to the transmission and the number of bit errors depends on the amount of noise and interference in the communication channel.

Channel coding is often used in digital communication systems to protect the digital information from noise and interference and reduce the number of bit errors. Channel coding is mostly accomplished by selectively introducing redundant bits into the transmitted information stream. These additional bits will allow detection and correction of bit errors in the received data stream and provide more reliable information transmission. The cost of using channel coding to protect the information is a reduction in data rate or an expansion in bandwidth.

### 1.1 Types of Channel Codes

There are two main types of channel codes, namely block codes and convolutional codes. There are many differences between block codes and convolutional codes. Block codes are based rigorously on finite field arithmetic and abstract algebra. They can be used to either detect or correct errors. Block codes accept a block of  $k$  information bits and produce a block of  $n$  coded bits. By predetermined rules,  $n-k$  redundant bits are added to the  $k$  information bits to form the  $n$  coded bits. Commonly, these codes are referred to as  $(n,k)$  block codes. Some of the commonly used block codes are Hamming codes, Golay codes, BCH codes, and Reed Solomon codes (uses nonbinary symbols). There are many ways to decode block codes and estimate the  $k$  information bits. These decoding techniques will not be discussed here but can be studied in courses on Coding Theory [Wic95].

Convolutional codes are one of the most widely used channel codes in practical communication systems. These codes are developed with a separate strong mathematical structure and are primarily used for real time error correction. Convolutional codes convert the entire data stream into one single codeword. The encoded bits depend not only on the current  $k$  input bits but also on past input bits. The main decoding strategy for convolutional codes is based on the widely used Viterbi algorithm.

As a result of the wide acceptance of convolutional codes, there have been many advances to extend and improve this basic coding scheme. This advancement resulted in two new coding schemes, namely, trellis coded modulation (TCM) and turbo codes. TCM adds redundancy by combining coding and modulation into a single operation (as the name implies). The unique advantage of TCM is that there is no reduction in data rate or expansion in bandwidth as required by most of the other coding schemes.

Recently, a near channel capacity error correcting code called turbo code was introduced. This error correcting code is able to transmit information across the channel with arbitrary low (approaching zero) bit error rate [Pro95]. This code is a parallel concatenation of two component convolutional codes separated by a random interleaver. It has been shown that a turbo code can achieve performance within 1 dB of channel capacity [BER93]. Random coding of long block lengths may also perform close to channel capacity, but this code is very hard to decode due to the lack of code structure. Without a doubt, the performance of a turbo code is partly due to the random interleaver used to give the turbo code a “random” appearance. However, one big advantage of a turbo code is that there is enough code structure (from the convolutional codes) to decode it efficiently.

There are two primary decoding strategies for turbo codes. They are based on a maximum a posteriori (MAP) algorithm and a soft output Viterbi algorithm (SOVA). Regardless of which algorithm is implemented, the turbo code decoder requires the use of two (same algorithm) component decoders that operate in an iterative manner. In this thesis, the SOVA will be examined, because it is much less complex than MAP and it provides comparable performance results. Furthermore, SOVA is an extension of the Viterbi algorithm, and thus has an implementation advantage over MAP.

## 1.2 Outline of Thesis

In this thesis, a new class of convolutional codes, called turbo codes, is presented. Because a turbo code is an extension of a convolutional code, the basic concepts of convolutional code are required. Chapter 2 presents the fundamentals of convolutional code. This chapter discusses the encoder structure and its many representations. Also, it discusses the primary decoding algorithm for convolutional code, namely the Viterbi algorithm. Both hard- and soft-decision Viterbi algorithms are presented in Chapter 2. Furthermore, performance issues related to convolutional code are discussed.

Chapter 3 introduces the basic turbo code encoder. The turbo code encoder is a parallel concatenation of two recursive systematic convolutional (RSC) codes, separated by an interleaver. This chapter shows the construction of a RSC encoder from a nonrecursive nonsystematic (conventional) convolutional encoder. It discusses the similarities and differences between the conventional and RSC encoders in terms of their

intrinsic properties. Furthermore, this chapter discusses the function of the interleaver and describes many different types of interleaver that are suitable for the turbo code encoder.

The turbo code decoder is described in Chapter 4. This chapter first describes the drawbacks of decoding multistage concatenated convolutional codes with the basic Viterbi algorithm. A reliability (soft output) value is introduced to combat these drawbacks. Also, integration of this reliability value to the basic Viterbi algorithm is presented. The resulting Viterbi algorithm is known as the soft output Viterbi algorithm (SOVA). The SOVA needs to be further modified for turbo code decoding. To understand this modification, the concepts of log-likelihood ratio and soft channel outputs are introduced. From these concepts, the metric for the SOVA is derived. The SOVA component decoder for turbo code is then introduced. A simple and efficient implementation of the SOVA component decoder is presented. This chapter concludes with a description of the SOVA iterative turbo code decoder (concatenated SOVA component decoders).

Chapter 5 investigates the performance of turbo codes through extensive computer simulation. The simulation setup is carefully detailed so other researchers may reproduce the simulation results. Many simulation results are then presented to show the important characteristics of turbo codes. Furthermore, this chapter discusses the relative performance and complexity of turbo codes. Chapter 6 then summarizes the important findings about turbo codes, and concludes the thesis.