Improved SOVA and APP Decoding Algorithms for Concatenated Codes

Chuanxiu Huang

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Application Science

Concordia University

Montreal, Quebec, Canada

November 2004

Your file Votre référence
ISBN: 0-494-10236-5
Our file Notre référence
ISBN: 0-494-10236-5

Abstract

Improved SOVA and APP Decoding Algorithms for Concatenated Codes

Chuanxiu Huang

In this thesis, we propose a novel and simple approach for dealing with the exaggerated *extrinsic* information produced by the soft-output Viterbi algorithm (SOVA). We identify the reason behind producing these exaggerated values and propose a simple remedy for it. We argue that what leads to these optimistic extrinsic information is the *inherent* strong correlation between the *intrinsic* information (input to the SOVA) and extrinsic information (output of the SOVA). The proposed remedy is based on mathematical analysis and it involves using two attenuators, one applied to the immediate output of the SOVA and another applied to the extrinsic information before it is passed to the other decoder (assuming iterative decoding). We examine the modified SOVA (MSOVA) on various channel models including additive white Gaussian noise (AWGN) channels, flat fading channels, and storage channels for both parallel concatenated codes (PCCs) and serial concatenated codes (SCCs). We show that the MSOVA provides substantial performance improvements over these channels. For example, it provides improvements of about 0.8 to 1.0 dB on AWGN channels,

about 1.4 to 2.0 dB on fading channels, and up to 1.6 dB on storage channels, all at bit error rate 10^{-5}. We also show that there are cases where the MSOVA is superior to the *a posteriori* probability (APP) algorithm. With this motivation, we extend the proposed modification to the APP algorithm on AWGN and fading channels with favorable results. We demonstrate that the modified APP (MAPP) provides performance improvements between 0.3 to 0.6 dB at bit error rate10^{-5} relative to the APP. We lastly mention that the proposed modifications, while they provide considerable performance improvements, add only two multipliers to the complexity of the conventional SOVA, which is remarkable.

Dedicated to my dearest husband, parents, and sister.....

Acknowledgments

First of all, I would like to express my sincere appreciation to my academic supervisor Dr. Ali Ghrayeb for giving me invaluable support, constructive guidance and helpful discussion. I am grateful for his patience and kindness in answering my questions and revising my reports. Without his continuous encouragement and stimulating suggestions, I would not have succeeded.

I would also like to thank Xiang Nian Zeng, Hao Shen, Mohamed Abou-Khousa and Abdollah Sanei in our research group. They generously provided me with a lot of suggestions and help in my research and thesis. I also want to thank all the people who helped me throughout my study at Concordia University.

Finally, I would like to thank my husband and my parents for their love, trust and encouragement. Without their support, it would have been impossible for me to accomplish what I have accomplished.

Contents

List of Figures ix

List of Tables xiii

1 Introduction 1

1.1 PCC and SCC Codes 3

1.2 Iterative Decoding Algorithms 5

1.2.1 The APP Algorithm 6

1.2.2 Overview of the SOVA Algorithm 11

1.3 Thesis Outline 13

1.4 Thesis Contribution 14

2 Improved SOVA and APP Decoding Algorithms for AWGN and Fading Channels 16

2.1 Introduction 16

2.2 System Model 18

2.3 Modified SOVA Algorithm 20

2.3.1 Proposed Modifications 20

2.3.2 Extension to the SOVA in the SCC Case 26

2.3.3 Extension to the APP Algorithm 28

2.3.4 Computation of the Coefficients c and d 29

2.4 Performance Analysis 32

2.4.1 AWGN Channels 33

2.4.2 Fading Channels 34
2.5 Simulation Results 36
2.5.1 AWGN Channel 37
2.5.2 Fading Channel 40
2.6 Concluding Remarks 43

3 Improvements in SOVA-based Decoding for Turbo-coded Data Storage Channels 44
3.1 Introduction 44
3.2 System and Channel Model 46
3.3 Modified SOVA Algorithm 49
3.3.1 The SOVA for PR Trellises 50
3.3.2 Proposed Modifications 55
3.3.3 Computation of the Coefficients c and d 60
3.4 Simulation Results 63
3.4.1 Idealized PR4 Channel 63
3.4.2 Equalized PR4 Channel 67
3.5 Concluding Remarks 70

4 Conclusions and Future Work 71
4.1 Conclusions 71
4.2 Future Work 72

Bibliography 73

List of Figures

1.1 Encoder (top) and decoder (bottom) structure of a PCC system. [E denotes encoder component, D denotes decoder component, Π, Π^{-1} denote interleaver and de-interleaver, respectively, P denotes puncturing.] 3

1.2 Encoder (top) and decoder (bottom) structure of a SCC system. [E denotes encoder component, D denotes decoder component, Π, Π^{-1} denote interleaver and de-interleaver, respectively, P denotes puncturing.] 4

1.3 A diagram on the trellis of the APP decoder. The top part of the figure depicts the forward recursion, the middle part of the figure depicts the backward recursion, and the bottom part depicts the computation of $L(\hat{u}_k)$. 10

2.1 System model with an AWGN channel (top) and a flat fading channel (bottom) [Π denotes interleaver.] 18

2.2 Implementation of the proposed MSOVA algorithm. 26

2.3 Comparison between the extrinsic information supplied by the SOVA (top), the SOVA-SA (middle), and the MSOVA (bottom), and that supplied by the MAPP (for PCC and AWGN using the same bit and noise sequences.) 31

2.4 Comparison between the extrinsic information supplied by the SOVA (top), the SOVA-SA (middle), and the MSOVA (bottom), and that supplied by the MAPP (for SCC and AWGN using the same bit and noise sequences.) 31

2.5 Comparison between the extrinsic and intrinsic information for the MAPP and APP algorithms. (The top figure is for the PCC case, and the bottom one is for the SCC case.) 32
2.6 Bit error rate performance comparison between the SOVA, SOVA–SA, MSOVA, APP, and MAPP decoders in AWGN channels. [PCC scheme with 4-state RSC encoders, rate 1/2, $N = 512$, and 8 decoder iterations.] 37
2.7 Bit error rate performance comparison between the SOVA, SOVA–SA, MSOVA, APP, and MAPP decoders in AWGN channels. [PCC scheme with 16-state RSC encoders, rate 4/5, $N = 512$, and 8 decoder iterations.] 38
2.8 Bit error rate performance comparison between the SOVA, SOVA–SA, MSOVA, APP, and MAPP decoders in AWGN channels. [SCC scheme with 4-state outer code, differential encoder for the inner code, overall rate 1/2, $N = 512$, and 8 decoder iterations.] 39
2.9 Bit error rate performance comparison between the SOVA, MSOVA and APP decoders in flat fading channels. [PCC scheme with 16-state RSC encoders, rate 4/5, $N = 512$, and 8 decoder iterations.] 40
2.10 Bit error rate performance comparison between the SOVA, MSOVA, APP, and MAPP decoders in flat fading channels. [SCC scheme with 4-state outer code, differential encoder for the inner code, overall rate 1/2, $N = 512$, and 8 decoder iterations.] 41
2.11 Bit error rate performance comparison between the SOVA, MSOVA, and APP decoders in flat fading channels. [SCC scheme with 4-state outer code, differential encoder for the inner code, overall rate 8/9, $N = 1024$, and 8 decoder iterations.] 42
3.1 System and channel model. 47
3.2 The SOVA output for bit u_k in PR trellises 54
3.3 Implementation of the MSOVA algorithm 60

3.4 Comparison between the extrinsic information supplied by the SOVA and MSOVA, and that supplied by the APP on the idealized PR4 channel. Left plots for the SCC system; right plots for the PCC system. (All results for the same concatenation system were obtained using the same bit and noise sequences.) 63
3.5 Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [PCC system with 16-state RSC encoders, rate 4/5, $N = 2048$, and 5 decoder iterations.] 64
3.6 Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [SCC system with 16-state outer RSC encoder, rate-1 inner code, overall rate 4/5, $N = 2048$, and 5 decoder iterations.] 65
3.7 Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [SCC system with 16-state outer RSC encoder, rate-1 inner code, overall rate 16/17, $N = 2048$, and 5 decoder iterations.] 66
3.8 Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [SCC system with 16-state outer RSC encoder, rate-1 inner code, overall rate 64/65, $N = 2048$, and 5 decoder iterations.] 66
3.9 Bit error rate performance comparison between the SOVA, MSOVA, and APP decoders on the equalized PR4 channel. [PCC scheme with 16-state outer RSC encoder, code rate 16/17, $N = 2048$, $S_u = 2.0$, and 5 decoder iterations.] 68
3.10 Bit error rate performance comparison between the SOVA, MSOVA, and APP decoders on the equalized PR4 channel. [SCC scheme with 16-state outer RSC encoder, rate-1 inner code, overall rate 16/17, $N = 2048$, $S_u = 2.0$, and 5 decoder iterations.] 68

3.11 Comparison between the extrinsic information supplied by the SOVA and MSOVA, and that supplied by the APP on the equalized PR4 channel. Left plots for the SCC system; right plots for the PCC system. (All results for the same concatenation system were obtained using the same bit and noise sequences.) 69

List of Tables

2.1 Correlation coefficient between the intrinsic and extrinsic information supplied by the MAPP, APP, MSOVA, SOVA-SA and SOVA in the PCC scheme. 22

2.2 Correlation coefficient between the intrinsic and extrinsic information supplied by the MAPP, APP, MSOVA, SOVA-SA and SOVA in the SCC scheme. 27

2.3 Values of c and d for various PCC and SCC codes for the MSOVA algorithm on AWGN and fading channels. 30

3.1 Correlation coefficient for the PCC and SCC codes on the equalized PR4 channels, Rate=16/17 56

3.2 Values of c and d for various PCC and SCC codes for the MSOVA algorithm on PR4 channels. 61

Chapter 1

Introduction

The main challenge of communication systems is to recover the original information at the receiver as reliably as possible. To improve the transmission quality, channel coding has become an indispensable tool. It involves the addition of some redundant symbols to a group of source symbols, which helps to correct errors in the received signals corrupted by a noisy channel. In 1948, Shannon demonstrated in his landmark paper [1] that by proper encoding of the information, errors caused by a noisy channel or storage medium can be reduced to any desired level if the data transmission rate R (in bits/sec) from the source encoder is smaller than the channel capacity C, where $C = B\log_2(1 + S/N)$, where B is the system bandwidth and S/N is the *signal-to noise ratio* (SNR).

Since then, great efforts have been made to design codes that would approach the Shannon limit. An important step towards achieving this goal was made by Forney [2], where he proposed to cascade relatively simple codes to obtain powerful overall

codes. One of the most prominent concatenation schemes is the serial concatenation of an inner convolutional code and a powerful nonbinary Reed-Solomon (RS) outer code. With this scheme, a nearly error-free performance is achievable (bit error rate on the order of 10^{-10}.) The astonishing performance of these concatenated codes resulted in a surge in the research on concatenated codes.

In 1993, Berrou, *et al.* [3] introduced a new coding/decoding technique referred to as *turbo coding.* A turbo codes normally consists of two recursive convolutional codes concatenated in parallel and is decoded using an iterative algorithm consisting of two maximum *a posteriori* probability (MAP) decoders. Performance less than 1.0 dB away from capacity is reported with a short constraint length, very long block length, and $10-20$ iterations of decoding. Later, Benedetto, *et al.* introduced in [4] a coding scheme that involves concatenating two convolutional codes in a serial fashion with a random interleaver separating them. It was shown that serial concatenated convolutional codes normally outperform their counterpart parallel concatenated convolutional codes over an additive white Gaussian noise (AWGN) channel. Due to their excellent error correcting capability, these codes are being considered for the 3rd Generation (3G) mobile communication standards, 3GPP, UTMS, and CDMA.

The idea of concatenated codes can be extended to various codes and code combinations. In this thesis, our interest is focused on the concatenation of two convolutional codes because of their impressive performance and wide range of applications. For simplicity, we refer to the parallel concatenated code as the PCC code and the serial concatenated code as the SCC code.

1.1 PCC and SCC Codes

Fig. 1.1 depicts a PCC system. As shown in the top part of the figure, the encoder consists of two recursive systematic convolutional (RSC) encoders (denoted by E1, E2) arranged in a parallel fashion, and separated by a random interleaver (denoted by Π). A puncturing mechanism (denoted by P) is also employed to obtain various code rates.

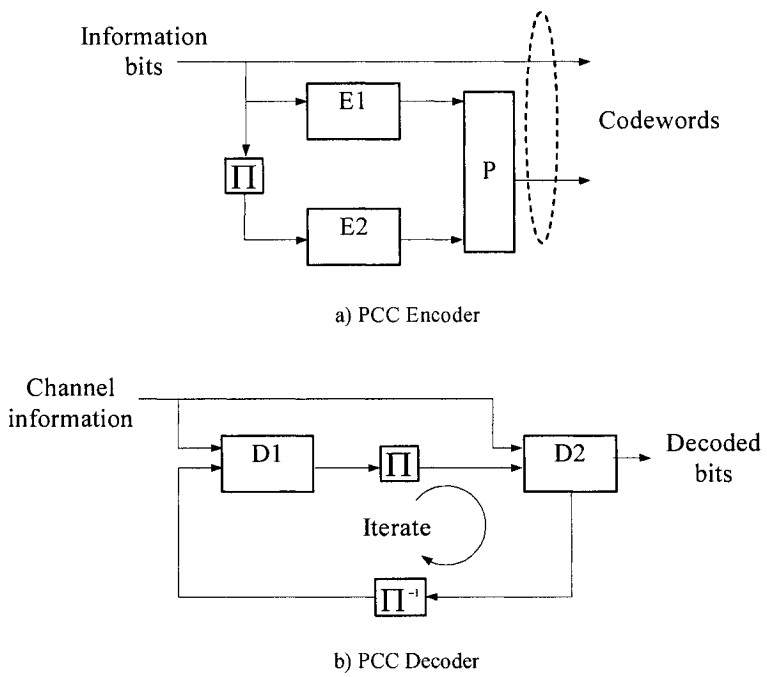


Figure 1.1: Encoder (top) and decoder (bottom) structure of a PCC system. [E denotes encoder component, D denotes decoder component, Π, Π^{-1} denote interleaver and de-interleaver, respectively, P denotes puncturing.]

Without loss of generality, we assume that the constituent codes are identical. A corresponding PCC decoder is shown in the bottom part of the figure. The input to each decoder component (denoted by D1, D2) includes information from the channel

and extrinsic information delivered from the other decoder component. This *soft-in/soft-out* (SISO) decoder works in an iterative fashion, and is known to converge at low bit error rates as the number of iterations increases [5].

The basic structure of a SCC code is shown in Fig. 1.2. As seen in the top part of the figure, the outer encoder (E1) and the inner encoder (E2) are serially concatenated with an interleaver inserted between them. The overall code rate r of the SCC code is the product of the outer code rate r_o and the inner code rate r_i, i.e., $r = r_o \cdot r_i$. In the bottom part of the figure, we present the iterative decoder structure for a SCC code. This iterative SCC decoder employs two SISO decoding components, one matched to the trellis of the outer encoder and another matched to the trellis of the inner encoder.

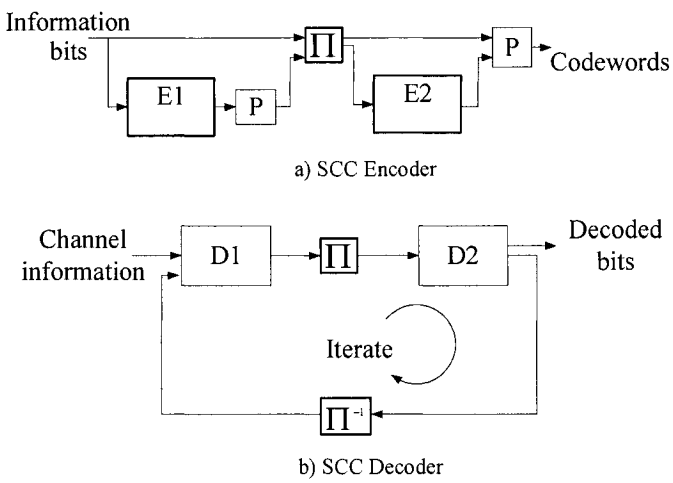


Figure 1.2: Encoder (top) and decoder (bottom) structure of a SCC system. [E denotes encoder component, D denotes decoder component, Π, Π^{-1} denote interleaver and de-interleaver, respectively, P denotes puncturing.]

1.2 Iterative Decoding Algorithms

An iterative decodernormally consistes of two SISO decoder components[1]. The goal of the decoder component is to iteratively estimate the *a posteriori* probabilities of each information bit u_k given the observation of the received sequence $\mathbf{y}$. In the logarithm domain, the log-likelihood ratio (LLR) of the *a posteriori* probability is defined as

$$L(\hat{u}_k) \triangleq \log \left[\frac{P(u_k = +1 \mid \mathbf{y})}{P(u_k = -1 \mid \mathbf{y})}\right]. \tag{1.1}$$

A SISO decoder component produces soft-output values of the LLR. This LLR can be represented as the sum of three independent terms [6]: the channel information $L_c y_k$ ($L_c \triangleq 4rE_b/N_0$, is called the reliability value of the channel), the *a priori* values $L(u_k)$ delivered from the other SISO decoder component, and the extrinsic information $L_e(\hat{u}_k)$, i.e.,

$$L(\hat{u}_k) = L_c y_k + L(u_k) + L_e(\hat{u}_k). \tag{1.2}$$

The extrinsic information $L_e(\hat{u}_k)$, is obtained by subtracting $L_c y_k + L(u_k)$ from $L(\hat{u}_k)$ (see 1.2), and then passed on to the other decoder component. This extrinsic information serves as *a prior* information for the receiving decoder component. For SCC

[1]It is possible for the iterative decoder to consist of more than two SISO decoders, depending on the number of concatenated codes. However, for practical reasons, only two SISO decoders are used.

codes, the iterative decoding proceeds as D2, D1, D2, *etc.*, with the previous decoder passing extrinsic information along to the next decoder at each half iteration. In the case of PCC codes, for hardware implementations, D1 and D2 can operate simultaneously by initializing the *a prior* information to zero in the first iteration.

The *a posteriori* probability (APP) algorithm or sub-optimal algorithms, such as the $soft-output\ Viterbi\ algorithm$ (SOVA) are implemented in each decoder component to produce the soft output LLRs. In the following subsections, we give a brief introduction to the APP algorithm and the SOVA algorithm.

1.2.1 The APP Algorithm

In the symbol-by-symbol APP decoder, the decoder decides $u_k = +1$ if $P(u_k = +1 \mid \mathbf{y}) > P(u_k = -1 \mid \mathbf{y})$, and it decides $u_k = -1$ otherwise, where $\mathbf{y}$ is the noisy received sequence. In the logarithm domain, the decision $\hat{u}_k$ is given by

$$\hat{u}_k = \text{sign}\ [L(\hat{u}_k)], \tag{1.3}$$

where $L(\hat{u}_k)$ is the log *a posteriori* probability defined by (1.1). To utilize the code trellis in computing the LLRs, we rewrite $L(\hat{u}_k)$ as

$$L(\hat{u}_k) = \log \frac{\sum\limits_{u_k=+1} P\left(s_{k-1} = s', s_k = s, \mathbf{y}\right)}{\sum\limits_{u_k=-1} P\left(s_{k-1} = s', s_k = s, \mathbf{y}\right)}, \tag{1.4}$$

where s_k is the encoder state at time k, $u_k = +1$ is the set of pairs (s', s) for the state transitions $(s_{k-1} = s') \rightarrow (s_k = s)$ which corresponds to the event $u_k = +1$, and $u_k = -1$ is similarly defined. To compute (1.4), we factor $P(s_{k-1} = s', s_k = s, \mathbf{y})$ as

$$P(s_{k-1} = s', s_k = s, \mathbf{y}) = P(s', y_{j<k}) \cdot P(s, y_k|s') \cdot P(y_{j>k}|s),$$

or, in the logarithmic domain

$$\log P(s_{k-1} = s', s_k = s, \mathbf{y}) = \underbrace{\log P(s', y_{j<k})}_{\tilde{\alpha}_{k-1}(s')} + \underbrace{\log P(s, y_k|s')}_{\tilde{\gamma}_k(s', s)} + \underbrace{\log P(y_{j>k}|s)}_{\tilde{\beta}_k(s)}. \quad (1.5)$$

From (1.4) and (1.5), we have

$$\begin{aligned} L(\hat{u}_k) &= \log \left[\sum_{u_k=+1} \exp\left(\tilde{\alpha}_{k-1}(s') + \tilde{\gamma}_k(s', s) + \tilde{\beta}_k(s) \right) \right] \\ &\quad - \log \left[\sum_{u_k=-1} \exp\left(\tilde{\alpha}_{k-1}(s') + \tilde{\gamma}_k(s', s) + \tilde{\beta}_k(s) \right) \right]. \end{aligned} \quad (1.6)$$

To compute $\tilde{\gamma}_k(s', s)$, which is also named as the *branch metric*, we have

$$\begin{aligned} \tilde{\gamma}_k(s', s) &= \log \left(\frac{P(s|s')}{P(s')} \cdot \frac{p(y_k|s', s)}{P(s', s)} \right) \\ &= \log P(u_k) + \log p(y_k|u_k) \\ &= \log P(u_k) - \log(\sqrt{4\pi/L_c}) - \frac{L_c}{4} \|y_k - c_k\|^2. \end{aligned} \quad (1.7)$$

According to the definition of the extrinsic information

$$L_e(\hat{u}_k) \triangleq \log \left[\frac{P(u_k = +1)}{P(u_k = -1)}\right], \tag{1.8}$$

we have

$$\begin{aligned} \log P(u_k) &= \log \left\{\left(\frac{\exp[-L(u_k)/2]}{1+\exp[-L(u_k)/2]}\right) \cdot \exp[u_k L(u_k)/2]\right\} \\ &= A_k + \frac{1}{2}u_k L(u_k). \end{aligned} \tag{1.9}$$

Note that $L(u_k)$ is the extrinsic information produced by the other decoder. Substituting (1.9) into (1.7), we have

$$\tilde{\gamma}_k(s', s) = \frac{1}{2}u_k L(u_k) - \frac{L_c}{4}\|y_k - c_k\|^2 + A_k - \log(\sqrt{4\pi/L_c}). \tag{1.10}$$

With the knowledge of $\tilde{\gamma}_k(s', s)$ of each branch in the code trellis, we compute $\tilde{\alpha}_k(s)$ in the forward recursion by

$$\tilde{\alpha}_k(s) = \log \sum_{(s',s)} \exp[\tilde{\alpha}_{k-1}(s') + \tilde{\gamma}_k(s', s)]. \tag{1.11}$$

Since each component convolutional encoder is initialized to the zero state, $\tilde{\alpha}_k(s)$ is initialized according to

$$\tilde{\alpha}_0(s) = \begin{cases} 0, s = 0 \\ -\infty, s \neq 0 \end{cases}.$$

In the backward recursion, $\tilde{\beta}_{k-1}(s')$ is yield according to

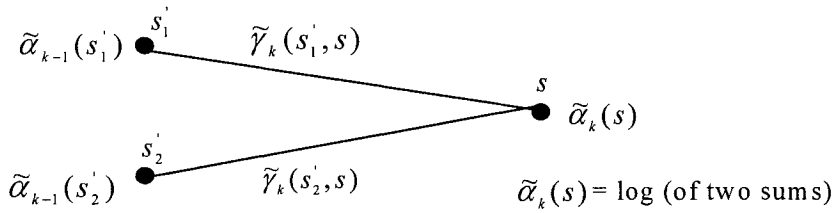
$$\tilde{\beta}_{k-1}(s') = \log \sum_{(s',s)} \exp[\tilde{\gamma}_k(s',s) + \tilde{\beta}_k(s)]. \tag{1.12}$$

For a component convolutional encoder which is terminated to the zero state, in the corresponding decoder component, $\tilde{\beta}_k(s)$ is initialized according to

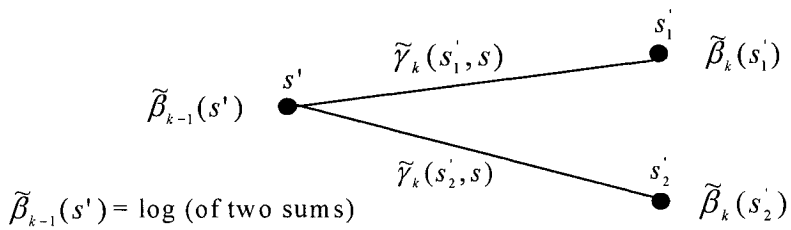
$$\tilde{\beta}_K(s) = \begin{cases} 0, s = 0 \\ -\infty, s \neq 0 \end{cases}.$$

A diagram of the computation of $\tilde{\alpha}_k(s)$, $\tilde{\beta}_k(s)$,and $L(\hat{u}_k)$ is shown in Fig. 1.3. In addition, in order to isolate the extrinsic information from $L(\hat{u}_k)$, and feed it to the other decoder as the *a priori* information, it is obtained from (1.2) as

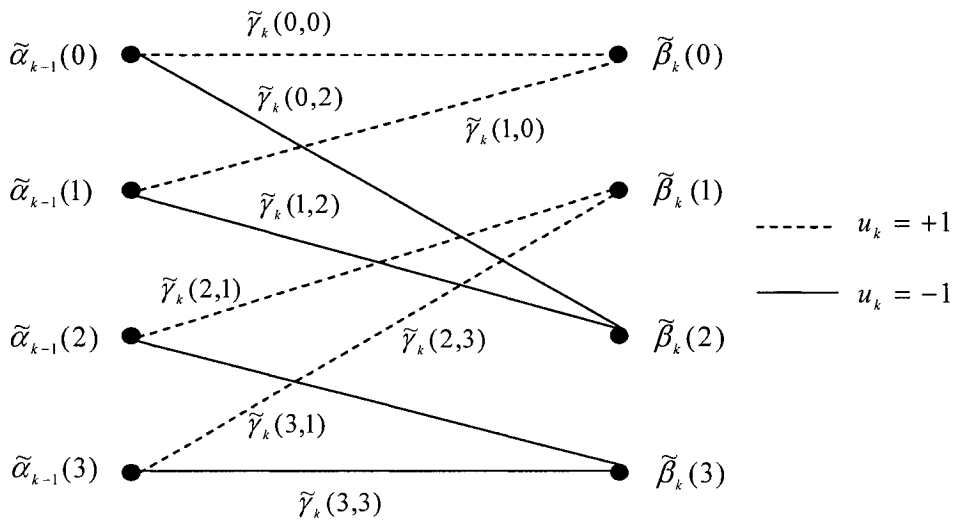
$$L_e(\hat{u}_k) = L(\hat{u}_k) - L_c y_k - L(u_k). \tag{1.13}$$



a) Forward recursion



b) Backward recursion



$$L(\hat{u}_k) = \log[\sum_{u_k=+1}(\tilde{\alpha}_{k-1} + \tilde{\gamma}_k + \tilde{\beta}_k)] - \log[\sum_{u_k=-1}(\tilde{\alpha}_{k-1} + \tilde{\gamma}_k + \tilde{\beta}_k)]$$

c) Computation of $L(\hat{u}_k)$

Figure 1.3: A diagram on the trellis of the APP decoder. The top part of the figure depicts the forward recursion, the middle part of the figure depicts the backward recursion, and the bottom part depicts the computation of $L(\hat{u}_k)$.

1.2.2 Overview of the SOVA Algorithm

The Viterbi algorithm is optimal in the sense that it maximizes the *a posteriori* probability $P(\mathbf{s} \mid \mathbf{y})$, where $\mathbf{s}$ is the state sequence representing the trellis path, and $\mathbf{y}$ is the received sequence. The SOVA finds the *most likely* (ML) path sequence as the original Viterbi algorithm, and traces back the ML path to obtain the hard decision $\hat{u}_k$. To produce the soft value as the *a posteriori* probability for each bit, the SOVA only involves the two most competitive paths, in contrast to the APP algorithm that actually considers all the trellis paths. This benefits the SOVA decoder by reducing its complexity about a factor of two relative to the APP decoder [7].

However, some performance degradation is suffered. For example, the SOVA performance was examined on AWGN channels for rate 1/2 in [7] and [8], rates that range from 1/2 to 8/9 in [6], and rates from 4/5 to as high as 64/65 in [9]. In all of these works, it was demonstrated that the SOVA algorithm is inferior to the APP algorithm by about 0.7 to 1.0 dB at bit error rate 10^{-5}. This performance gap widens considerably on fading channels and band-limited channels such as partial response (PR) channels. For instance, on fading channels, as we will demonstrate in Chapter 2, the degradation due to using the SOVA, relative to the APP, can be as large as 2.5 dB at bit error rate 10^{-5} with code rate 4/5. Larger degradations are expected at higher code rates. Also, on PR channels, as we will see later in Chapter 3, the degradation can be as large as 2.0 dB at bit error rate 10^{-5} relative to the APP, depending on the concatenation scheme and code rate.

On the other hand, the SOVA is still attractive, not only because it reduces the

complexity, but also because, unlike the APP decoder, it does not require knowledge of the channel noise variance [10], which makes it easier to implement in practice.

Several papers have recently looked into why the SOVA gives a poor performance relative to the APP [10]−[15]. It is suggested in these papers that the reason behind this degradation is that the reliability values at the output of the SOVA decoder are typically larger than those that would have been produced by the APP decoder. In [11], the authors show that the output of the SOVA typically suffers some sort of distortion, which often leads to producing optimistic reliability values at its output. With this motivation, they propose to apply an adaptive attenuator to the output of the SOVA. With this scheme, improvements of about 0.3 dB over AWGN channels can be achieved at bit error rate 10^{-4}, relative to the APP.

In [12]−[14], the authors propose to apply a fixed attenuator to the output of SOVA, where improvements of about 0.2 to 0.3 dB are shown to be possible. Another approach was introduced in [10], which involves applying a threshold to the output of the SOVA so that the reliability values are clipped if they exceed a certain value. However, in addition to being heuristic, as was pointed out in [15], this approach provides in general marginal improvements. In [15], the author proposes to apply both an attenuator and a threshold at the same time to the output of the SOVA. With these modifications, the SOVA performs within 0.1 dB from the APP at high signal-to-noise ratio (for parallel concatenated codes.) For convenience, we shall henceforth refer to the SOVA algorithm when a single attenuator (SA) is applied as SOVA-SA.

Details about the SOVA algorithm on AWGN and fading channels will be presented in Chapter 2, while the extension to the SOVA on PR channels is given in Chapter 3.

1.3 Thesis Outline

In this chapter, we presented some of the basic concepts of concatenated codes. We also reviewed the APP algorithm. We showed that performance degradation is suffered when the APP decoder is replaced by the SOVA decoder on various channels. The main advantage of employing the SOVA, however, is a reduction in complexity by about 50% relative to that of the APP. In addition, the SOVA decoder, unlike the APP decoder, does not require knowledge of the channel noise variance, which makes it even more attractive. Efforts that people have made to improve the performance of the SOVA decoder were also discussed.

In Chapter 2, after a brief introduction to the SOVA algorithm, we identify the reason behind the performance degradation when the APP algorithm is replaced with the SOVA algorithm in the iterative decoder. A simple modification based on the original SOVA is proposed. We examine the modified SOVA (MSOVA) on AWGN channels and flat fading channels, and show that the MSOVA provides substantial performance improvements over these channels. We also show that there are cases where the MSOVA is superior to the APP algorithm. With this motivation, we extend the proposed modification to the APP algorithm with favorable results.

In Chapter 3, we examine the MSOVA on turbo-coded magnetic recording channels. We consider both idealized PR channels and the Lorentzian model equalized to a PR target. We demonstrate that substantial performance improvements are possible.

In Chapter 4, conclusions are made and directions for future work are suggested.

1.4 Thesis Contribution

The contributions of this thesis can be summarized as follows:

- We pin point the reason as to why the SOVA is inferior to the APP in terms of performance. We show that this degradation is due to the large extrinsic information produced by the SOVA relative to those that would have been produced by the APP. This exaggeration in the extrinsic values is mostly due to the strong correlation between the intrinsic information (inputs to the SOVA) and extrinsic information (outputs of the SOVA).

- Based on mathematical analysis, we propose a simple remedy for these exaggerated values. The proposed remedy involves using two attenuators, one applied to the immediate output of the SOVA and another applied to the extrinsic information before it is passed on to the other decoder component (assuming iterative decoding.)

- We show that the proposed solution succeeds in reducing this inherent correlation substantially, where the resulting correlation becomes comparable to that of the APP.

- We examine the MSOVA on AWGN and fading channels. We show that substantial performance improvements can be achieved.

- We extend the proposed modifications to the APP algorithm on AWGN and fading channels. We show that performance improvements of up to 0.6 dB at $P_b = 10^{-5}$ are possible, relative to the APP.

- We examine the performance of the MSOVA on PR channels. We demonstrate that improvements of up to 1.6 dB at $P_b = 10^{-5}$ are achievable. Moreover, the performance gap between the APP and MSOVA is only around 0.4 dB.

- The above mentioned substantial improvements come at the expense of adding two multipliers to the complexity of the conventional SOVA, which is still much less complex than the APP.

Chapter 2

Improved SOVA and APP Decoding Algorithms for AWGN and Fading Channels

2.1 Introduction

In this chapter, we propose a simple approach for dealing with the exaggerated reliability values produced by the SOVA algorithm. We first pin point the reason as to why the SOVA tends to produce these exaggerated reliability values at its output, and then propose simple modifications to overcome this problem. We argue that the reason behind these large reliability values is mostly the *high correlation* between the intrinsic information (inputs to the SOVA) and extrinsic information (outputs of the SOVA). Our proposed remedy for this problem is based on mathematical analysis and

it involves using two attenuators, one applied to the immediate output of the SOVA and another applied to the extrinsic information before it is passed on to the other decoder component (assuming iterative decoding.)[1]

We examine the performance of the proposed modified SOVA (MSOVA) on AWGN and fading channels for both PCC and SCC systems. For the PCC system, the MSOVA performs as good as the APP on both AWGN and fading channels. As for the SCC case, the MSOVA outperforms the SOVA by about 1.2 dB at bit error rate 10^{-5} on both channels. It was also interesting to observe that the MSOVA outperforms the APP by about 0.2 dB on AWGN and 0.5 dB on fading channels (both at bit error rate 10^{-5}.)[2] Motivated by these promising results, we modify the APP following the same approach that we used to modify the SOVA. With these modifications, we show that the modified APP (MAPP) provides improvements by about 0.3 dB on AWGN and 0.7 dB on fading channels (for the SCC system), all relative to the APP. As for the PCC case, the MAPP provides performance improvements of about 0.2 dB relative to the APP on both channels. We will give later some insight as to why the proposed modifications are more effective for the SCC system as compared to the PCC system.

The rest of this chapter is organized as follows. In Section 2.2, we describe the system model. In Section 2.3, we present the proposed MSOVA algorithm and discuss the rationale behind proposing these new modifications. Performance analysis is given in Section 2.4. We present and discuss the simulation results in Section 2.5. Finally,

[1]It is worth mentioning that the proposed MSOVA has almost the same complexity as that of the conventional SOVA, which makes it quite attractive.

[2]We remark that the APP algorithm when used in an iterative decoding environment is suboptimal.

Section 2.6 concludes this chapter.

2.2 System Model

The system under consideration is depicted in Fig. 2.1. In the top part of the figure, the channel is modeled as AWGN, whereas it is modeled as independent fading in the bottom part of the figure. In the fading case, the system is equipped with M_t transmit and M_r receive antennas, and it employs, in addition to the PCC (or SCC) code, a space-time block code (STBC) as described in [16] and [17]. An interleaver is inserted between the outer code and STBC code to decorrelate the coded sequence so as to ensure that the maximum diversity order is achieved.

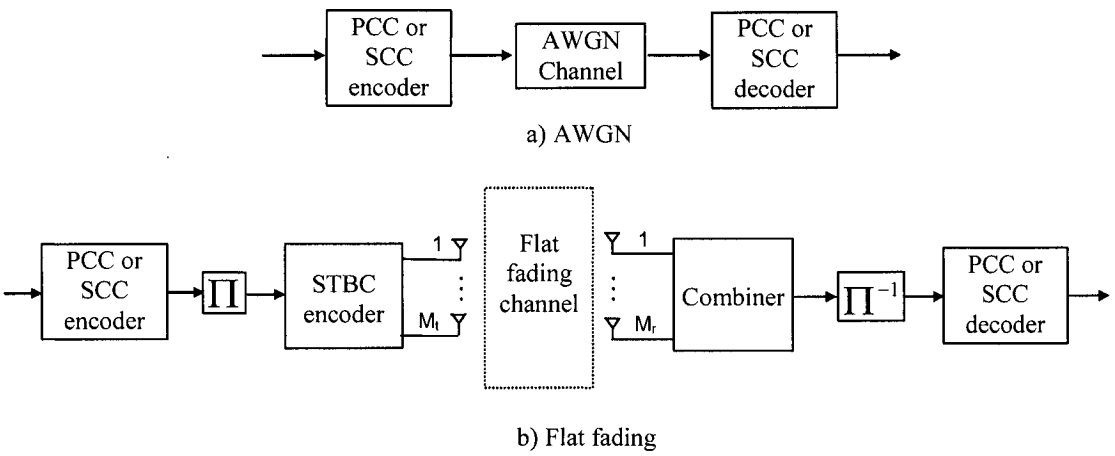


Figure 2.1: System model with an AWGN channel (top) and a flat fading channel (bottom) [Π denotes interleaver.]

When a PCC code is employed, the encoder consists of two identical RSC codes connected in parallel and separated by an interleaver. This code is decoded using

two identical iterative decoder components. In the SCC case, the encoder consists of two convolutional codes connected serially and separated by an interleaver. For the inner code, we always use a rate-1 differential encoder modeled by $1/(1+D)$. The SCC code is decoded using two iterative decoder components, one matched to the trellis of the outer encoder and another matched to the trellis of the inner encoder. For both concatenation schemes, each iterative decoder component implements the SOVA, SOVA-SA, MSOVA, APP, or MAPP. The decoder components used at a time are assumed to implement the same algorithm.

For the fading case, the signal y_k^j received by antenna j at time k received, after demodulation, matched-filtering, and sampling, is given by

$$y_k^j = \sum_{i=1}^{M_t} \alpha_{i,j}(k) c_k^i + w_k^j, \tag{2.1}$$

where c_k^i is the signal transmitted from antenna i at time k, the noise w_k^j at time k is modeled as independent samples of a zero-mean complex Gaussian random variable with variance $N_0/2$ per dimension. The coefficients $\alpha_{i,j}(k)$ model fading between the i^{th} transmit and j^{th} receive antennas at time k and are assumed to be complex Gaussian random variables with zero mean and variance 0.5 per dimension. We assume that the fading coefficients are known exactly at the receiver, and are constant over a block of consecutive M_t symbols and vary independently from one block to another. Moreover, the subchannels are assumed to fade independently. For the AWGN case, the signal model in (2.1) is modified as $y_k = c_k + w_k$.

2.3 Modified SOVA Algorithm

2.3.1 Proposed Modifications

In this section, we discuss the SOVA algorithm as an approximation to the APP algorithm for the PCC system; SOVA decoding for the SCC system is discussed in the following section. In the symbol-by-symbol APP decoder, the decoder decides $u_k = +1$ if $P(u_k = +1 \mid \mathbf{y}) > P(u_k = -1 \mid \mathbf{y})$, and it decides $u_k = -1$ otherwise, where $\mathbf{y}$ is the noisy received sequence. In the logarithm domain, the decision $\hat{u}_k$ is given by

$$\hat{u}_k = \text{sign } [L(\hat{u}_k)], \tag{2.2}$$

where $L(\hat{u}_k)$ is the log *a posteriori* probability ratio defined as

$$L(\hat{u}_k) \triangleq \log \left[\frac{P(u_k = +1 \mid \mathbf{y})}{P(u_k = -1 \mid \mathbf{y})}\right]. \tag{2.3}$$

Using Bayes' rule, (2.3) may be expressed as

$$L(\hat{u}_k) = \log \left[\frac{P(\mathbf{y} \mid u_k = +1)}{P(\mathbf{y} \mid u_k = -1)}\right] + \log \left[\frac{P(u_k = +1)}{P(u_k = -1)}\right], \tag{2.4}$$

where the second term represents the *a priori* information of bit u_k. Since equally likely inputs are typically assumed, the *a priori* term is usually zero for conventional decoders. However, in iterative decoding this *a priori* term is usually non-zero and it

represents the soft (extrinsic) information that the decoder components pass to each other.

The SOVA algorithm was first introduced in its MAP form by Forney [18]. Then Hagenauer [19] modified it so that the metric of the *Viterbi algorithm* (VA) incorporates the *a priori* or *a posteriori* information about the probability of the information bits. Without loss of generality, assuming an iterative decoder using two SOVA decoders (D1 and D2), the output of D1 at time k can be expressed as [6]

$$L(\hat{u}_k) = L_c y_k + L(u_k) + L_e(\hat{u}_k), \tag{2.5}$$

where $L_c y_k$ is the channel value ($L_c \triangleq 4rE_b/N_0$, which can be set to 1.0 for SOVA [10], r is the code rate, E_b is the average energy per information bit, y_k is the k^{th} noisy received systematic bit), $L(u_k)$ is the *a priori* values delivered from D2, and $L_e(\hat{u}_k)$ is the extrinsic information produced by D1. Now let $L_i(u_k) = L_c y_k + L(u_k)$, which represents the input (intrinsic information) to D1. It has been shown that the extrinsic information $L_e(\hat{u}_k)$ follows a Gaussian distribution [20], and so does $L_i(u_k)$ simply because $L(u_k)$ is the extrinsic information passed from D2 and $L_c y_k$ is the output of the channel.

In the conventional SOVA algorithm, it is normally assumed that the term $L_e(\hat{u}_k)$ in (2.5) is weakly correlated with the other two terms [6], and thus $L_e(\hat{u}_k)$ can be

obtained by directly subtracting $L_i(u_k)$ from $L(\hat{u}_k)$. That is

$$L_e(\hat{u}_k) = L(\hat{u}_k) - L_i(u_k). \tag{2.6}$$

However, we have observed through computer simulations that the correlation between $L_i(u_k)$ and $L_e(\hat{u}_k)$ is rather strong, which is normally ignored. We believe that this is essentially the reason behind the degradation in the SOVA performance. To elaborate on this, we have listed in Table 2.1 (last row) values for the correlation coefficient between the intrinsic and extrinsic information of the SOVA at 3 dB for different decoder iterations of the upper decoder.

Table 2.1: Correlation coefficient between the intrinsic and extrinsic information supplied by the MAPP, APP, MSOVA, SOVA-SA and SOVA in the PCC scheme.

# Iterns at 3 dB	1	3	8
MAPP	0.03	0.04	0.04
APP	0.04	0.04	0.05
MSOVA	0.04	0.14	0.20
SOVA-SA	0.20	0.22	0.25
SOVA	0.20	0.24	0.30

These values are obtained by simulating a PCC code with generator polynomials $(g_1, g_2) = (7, 5)_{oct}$, where g_1 is the feedback polynomial and g_2 is the feedforward polynomial, and code rate 1/2. We also list in Table 2.1 the correlation coefficient for the APP, SOVA-SA and two modification version algorithms proposed in this thesis later: the MSOVA and MAPP. In this table, all correlation coefficients were computed for the same noise and bit sequences. As is evident from the table, there is clearly a strong correlation between the extrinsic and intrinsic information of the

SOVA as compared to that of the APP (or MAPP). This correlation, however, is reduced significantly for the MSOVA particularly for small number of iterations.

The consequence of this correlation is that the extrinsic information estimated by the SOVA is larger than its true value. This is explained as follows. As the name suggests, the extrinsic information passed from one decoder to another should present *new* information to the receiving decoder. However, this strong correlation between the extrinsic and intrinsic information implies that part of the extrinsic information passed to a decoder is already known to this decoder. However, it is not treated as such, i.e., the receiving decoder normally regards all what it receives from the other decoder as new information. Consequently, this leads to a redundancy in the exchanged information, which tends to produce exaggerated extrinsic information at the output of the SOVA.

In light of the above discussion, it is clear that the output of the conventional SOVA is not really $L(\hat{u}_k)$ as defined by (2.4), and consequently its extrinsic information is not $L_e(\hat{u}_k)$ as defined by (2.6). Now, let us denote the actual SOVA output by $V(\hat{u}_k)$ and the corresponding extrinsic information by $V_e(\hat{u}_k)$, where the latter can be expressed as

$$V_e(\hat{u}_k) \triangleq V(\hat{u}_k) - L_i(u_k), \tag{2.7}$$

with $L_i(u_k)$ representing the intrinsic information (inputs to the SOVA). Our objective here is to modify $V(\hat{u}_k)$ and $V_e(\hat{u}_k)$ in order to obtain the true $L_i(u_k)$ and $L_e(\hat{u}_k)$,

which are supposed to be uncorrelated. As per the above discussion, the variables $L_i(u_k)$ and $V_e(\hat{u}_k)$ are correlated Gaussian random variables with means m_i, m_e, and variances σ_i^2, σ_e^2, respectively. Assuming $u_k = +1$ is transmitted, the joint conditional probability density function (pdf) $P(V_e(\hat{u}_k), L_i(u_k)|u_k = +1)$ is then given as

$$P(V_e(\hat{u}_k), L_i(u_k)|u_k = +1) = \frac{1}{2\pi\sigma_i\sigma_e\sqrt{1-\rho_+^2}} \cdot \exp\left(-\frac{1}{2\left(1-\rho_+^2\right)}\left[\frac{(V_e(\hat{u}_k)-m_e)^2}{\sigma_e^2} + \frac{(L_i(u_k)-m_i)^2}{\sigma_i^2}\right]\right) \cdot \exp\left(\frac{\rho_+(V_e(\hat{u}_k)-m_e)(L_i(u_k)-m_i)}{\sigma_i\sigma_e\left(1-\rho_+^2\right)}\right), \quad (2.8)$$

where ρ_+ is the correlation coefficient given by

$$\rho_+ = \frac{E\left[(V_e(\hat{u}_k)-m_e)\cdot(L_i(u_k)-m_i)\right]}{\sigma_i\sigma_e}. \quad (2.9)$$

where m_e and m_i are estimated as $m_e = \frac{1}{N}\sum_{k=1}^{N} V_e(\hat{u}_k)$ and $m_i = \frac{1}{N}\sum_{k=1}^{N} L_i(u_k)$, respectively, and N is the data block size. Also, σ_e and σ_i are estimated as $\sigma_e = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left[m_e - V_e(\hat{u}_k)\right]^2}$ and $\sigma_i = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left[m_i - L_i(u_k)\right]^2}$, respectively. Similarly, when $u_k = -1$ is transmitted, the joint conditional pdf $P(V_e(\hat{u}_k), L_i(u_k)|u_k = -1)$ is

given as

$$P(V_e(\hat{u}_k), L_i(u_k)|u_k = -1) = \frac{1}{2\pi\sigma_i\sigma_e\sqrt{1-\rho_-^2}} \cdot \exp\left(-\frac{1}{2(1-\rho_-^2)}\left[\frac{(V_e(\hat{u}_k)+m_e)^2}{\sigma_e^2} + \frac{(L_i(u_k)+m_i)^2}{\sigma_i^2}\right]\right) \cdot \exp\left(\frac{\rho_-(V_e(\hat{u}_k)+m_e)\cdot(L_i(u_k)+m_i)}{\sigma_i\sigma_e(1-\rho_-^2)}\right), \quad (2.10)$$

where

$$\rho_- = \frac{E\left[(V_e(\hat{u}_k)+m_e)\cdot(L_i(u_k)+m_i)\right]}{\sigma_i\sigma_e}. \quad (2.11)$$

Using Bayes' rule, the *a posteriori* log-likelihood ratio $L(\hat{u}_k)$ can be computed as

$$\begin{aligned} L(\hat{u}_k) &= \log\frac{P(u_k=+1|V_e(\hat{u}_k), L_i(u_k))}{P(u_k=-1|V_e(\hat{u}_k), L_i(u_k))} \\ &= aV_e(\hat{u}_k) + bL_i(u_k), \end{aligned} \quad (2.12)$$

where

$$a = \frac{1}{1-\rho^2}(r_e - \rho r_i\frac{\sigma_i}{\sigma_e}), \quad (2.13)$$

$$b = \frac{1}{1-\rho^2}(r_i - \rho r_e\frac{\sigma_e}{\sigma_i}), \quad (2.14)$$

$r_e = 2m_e/\sigma_e^2$, $r_i = 2m_i/\sigma_i^2$, and $\rho = \rho_+ = \rho_-$.

Substituting (2.7) into (2.12) yields

$$L(\hat{u}_k) = a\left(V(\hat{u}_k) - L_i(u_k)\right) + bL_i(u_k), \tag{2.15}$$

and, consequently, the extrinsic information to be passed to the other SOVA decoder, $L_e(\hat{u}_k)$, can be expressed as

$$\begin{aligned} L_e(\hat{u}_k) &= L(\hat{u}_k) - L_i(u_k) \\ &= (a+1-b)\left[\frac{a}{a+1-b}V(\hat{u}_k) - L_i(u_k)\right], \end{aligned} \tag{2.16}$$

which can be implemented as shown in Fig. 2.2 where c and d in the figure correspond to $a+1-b$ and $\frac{a}{a+1-b}$, respectively. This expression suggests that the immediate output of the SOVA should be scaled by d before the intrinsic information is subtracted off, and then the difference should be scaled by c.

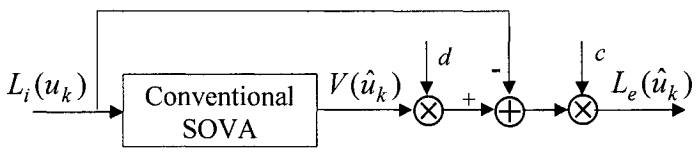


Figure 2.2: Implementation of the proposed MSOVA algorithm.

2.3.2 Extension to the SOVA in the SCC Case

Following the same experiment in the PCC case, we have studied the SCC case, and observed through computer simulations that the correlation between $L_i(u_k)$ and

$L_e(\hat{u}_k)$ is rather strong, similar to the case of PCC. We believe that this is the same reason behind the degradation in the SOVA performance in the SCC system. To elaborate on this, we have listed in Table 2.2 (last row) values for the correlation coefficient between the intrinsic and extrinsic information of the SOVA at 4 dB for different decoder iterations of the inner decoder. These values are obtained by simulating a SCC code with generator polynomials $(g_1, g_2) = (7, 5)_{oct}$, where g_1 is the feedback polynomial and g_2 is the feedforward polynomial, and code rate 1/2 (the inner code is simply a rate-1 differential code.) We also list in the table the correlation coefficient for the SOVA-SA, MSOVA, APP and MAPP (all correlation coefficients were computed for the same noise and bit sequences.) As is evident from the table, there is clearly a strong correlation between the extrinsic and intrinsic information of the SOVA as compared to that of the APP (or MAPP).

Table 2.2: Correlation coefficient between the intrinsic and extrinsic information supplied by the MAPP, APP, MSOVA, SOVA-SA and SOVA in the SCC scheme.

# Iterns at 4 dB	3	5	7
MAPP	0.03	0.03	0.03
APP	0.03	0.04	0.06
MSOVA	0.07	0.15	0.20
SOVA-SA	0.18	0.20	0.22
SOVA	0.18	0.22	0.25

In order to reduce this inherent correlation, we go through the same derivation as we have done in the PCC case, and the result is the same for both code systems, except the following two differences.

First, in the PCC case, due to symmetry, both iterative decoder components will

have the same structure, which is described by (2.16). Both decoders will even have the same values of c and d. However, this is not the case in the SCC case. That is, the inner decoder receives its inputs from the channel as well as the outer decoder, whereas the outer decoder receives all its inputs from the inner decoder. As such, for (2.16) to hold for the SCC case, the intrinsic information $L_i(u_k)$ in (2.16) should be replaced by $L(u_k)$ for both inner and outer decoders.

Secondly, the pair of values (c, d) is always the same for both decoder components in the PCC system, whereas they may not be the same in the SCC system. The reason being is that these values depend on the structure of the code, as well as the puncturing scheme. Therefore, when selecting the values of c and d, the inner and outer decoders in the SCC scheme should be treated separately. (More detail on this point is given later.)

2.3.3 Extension to the APP Algorithm

The changes applied to the SOVA algorithm, as outlined above, extend in a straightforward manner to the APP algorithm (for both PCC and SCC schemes.) The MAPP will also have the same structure as that of the MSOVA. Extending these modifications to the APP algorithm was actually inspired after we have observed that the MSOVA substantially outperforms the APP algorithm in the SCC system. As we have shown in Table 2.1, and Table 2.2, for both PCC and SCC schemes, the correlation coefficient of the MAPP is slightly decreased relative to the APP. Actually, modifying the APP algorithm with the aim of improving its performance has been

looked into before (see [3], as an example.)

2.3.4 Computation of the Coefficients c and d

In general, as mentioned above, to obtain the values of c and d, we need to compute the means and variances of $V_e(\hat{u}_k)$ and $L_i(u_k)$, and the correlation coefficient between them for every received data block and every decoding iteration.[3] However, this method has two drawbacks. First of all, these computations increase the complexity of the decoder due the additional processing delay. Also, the extrinsic information $V_e(\hat{u}_k)$ may not be strictly Gaussian [21], especially when the data block is relatively short, e.g., $N = 512$. As a matter of fact, we examined the performance of the MSOVA with the parameters c and d computed for data block length $N = 512$ and 1024 with the assumption that $V_e(\hat{u}_k)$ is Gaussian, but not much improvement was observed relative to the conventional SOVA.

As an alternative, we perform a computer search in an effort to find the pair (c, d) that would give the best performance. Since the values of c and d are dependent upon the correlation between the extrinsic and intrinsic information, they are affected by the code structure and the puncturing mechanism. Through simulations we found that once the code generator polynomials and puncturing mechanism are determined, there is a fixed pair (c, d) that can be used for any received data block and every decoding iteration. Moreover, this fixed pair of values works well for both channel

[3] In the unlikely event that when the correlation coefficient is very small, i.e., $\rho \approx 0$, and the output of the SOVA follows a Gaussian distribution, i.e., $r_i = r_e = 1$ [11], we have $c = d = 1$. In such cases, the proposed modifications are no longer needed.

models. We list in Table 2.3 the values of c and d that we found through Monte Carlo simulations for various PCC and SCC codes, and for various code rates. Each pair of (c, d) was obtained by averaging over 3×10^6 sample points (i.e., received symbols).

Table 2.3: Values of c and d for various PCC and SCC codes for the MSOVA algorithm on AWGN and fading channels.

PCC Code	(c, d)
$(7,5)_{oct}, r = 1/2$	$(0.9, 0.8)$
$(31,33)_{oct}, r = 4/5$	$(0.8, 0.7)$

SCC Code		(c, d)	
Outer	**Inner**	**Outer**	**Inner**
$(7,5)_{oct}$, $r = 1/2$	$\frac{1}{1+D}$, $r = 1$	$(0.7, 0.8)$	$(0.8, 1.0)$
$(7,5)_{oct}$, $r = 8/9$	$\frac{1}{1+D}$, $r = 1$	$(0.9, 0.8)$	$(1.0, 0.9)$

To illustrate the efficacy of the proposed MSOVA, we perform another experiment in which we compare the actual extrinsic information supplied by the SOVA, SOVA-SA, MSOVA, and APP algorithms with that supplied by the MAPP (for the same noise and bit sequences.)

In Fig. 2.3, we plot the extrinsic information of the SOVA, SOVA-SA, and MSOVA against that of the MAPP (for the PCC case.) Each part of the figure includes 2048 data points obtained after the first iteration at SNR = 3 dB (on AWGN).We observe from the figure that the extrinsic information of the SOVA is too optimistic relative to that of the MAPP, whereas the extrinsic information of the MSOVA matches very well that of the MAPP.

The same experiment was repeated for the SCC system and the results are scattered in Fig. 2.4.

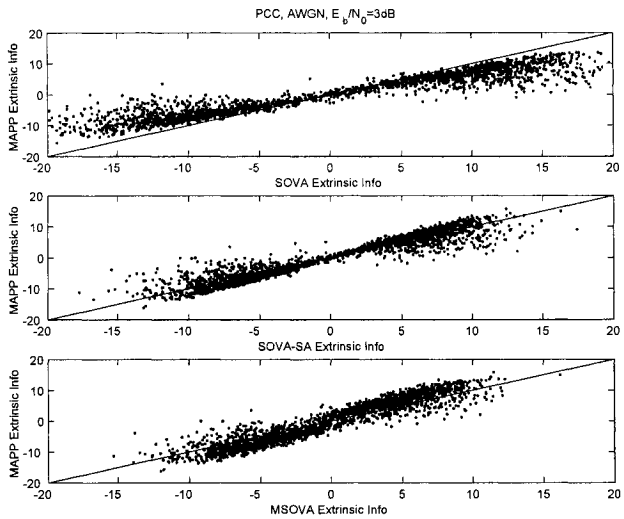


Figure 2.3: Comparison between the extrinsic information supplied by the SOVA (top), the SOVA-SA (middle), and the MSOVA (bottom), and that supplied by the MAPP (for PCC and AWGN using the same bit and noise sequences.)

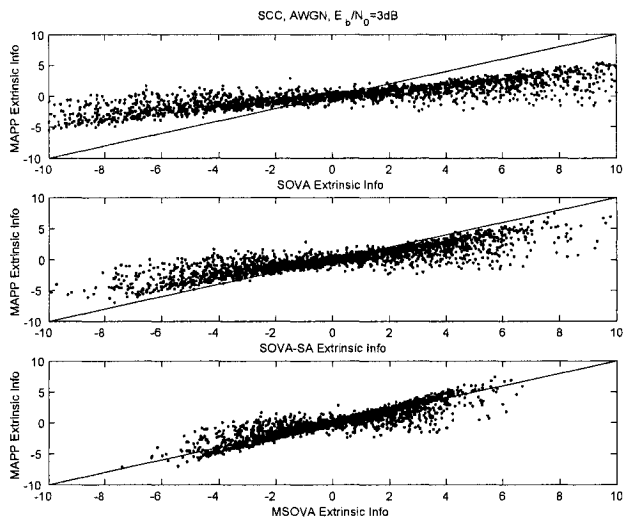


Figure 2.4: Comparison between the extrinsic information supplied by the SOVA (top), the SOVA-SA (middle), and the MSOVA (bottom), and that supplied by the MAPP (for SCC and AWGN using the same bit and noise sequences.)

In Fig. 2.5, we compare the extrinsic information of the APP with that of the MAPP for both concatenated coding schemes. It is also evident from these figures how effective the proposed modifications are.

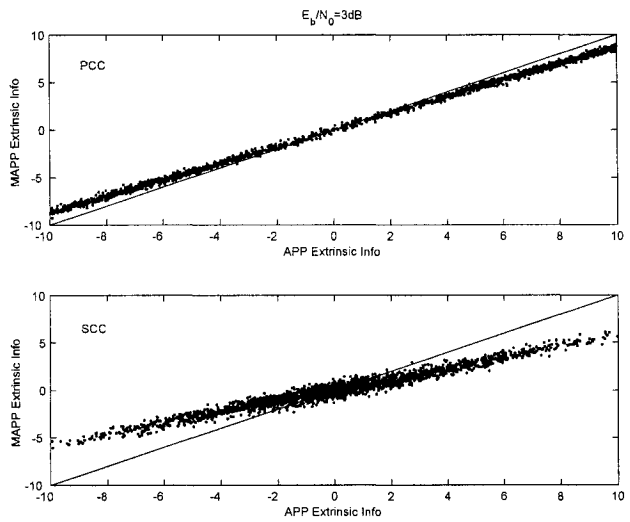


Figure 2.5: Comparison between the extrinsic and intrinsic information for the MAPP and APP algorithms. (The top figure is for the PCC case, and the bottom one is for the SCC case.)

2.4 Performance Analysis

For the purpose of making this thesis self-contained, we review in this section some of the main results, related to our work, on the maximum likelihood (ML) performance analysis of the above concatenation schemes. We use these results as a benchmark for the performance of the various iterative algorithms discussed in this chapter.

2.4.1 AWGN Channels

The ML analysis of turbo and turbo-like codes involves the union bound argument, which applies in the floor region only. Let us assume that the all-zero codeword was transmitted. Also, let the data block length be N. Assuming binary phase-shift keying (BPSK), the probability of codeword error (i.e., the decoder picking any codeword other than the all-zero codeword) is given by

$$P_{cw} = \Pr\left(\text{choosing any } k \in \left\{1, 2, \ldots, 2^N - 1\right\} \mid 0\right),$$

where the all-zero codeword is indexed as 0. This can be upper bounded using the union bound argument as

$$P_{cw} \leq \sum_{k=1}^{2^N-1} P_{cw}\left(k \mid 0\right) \tag{2.17}$$

$$= \sum_{k=1}^{2^N-1} Q\left(\sqrt{\frac{d_k r E_b}{N_0}}\right), \tag{2.18}$$

where $P_{cw}\left(k \mid 0\right)$ is the probability of choosing the k^{th} codeword given the all-zero codeword was sent, d_k is the Hamming weight of the k^{th} codeword, r is the overall code rate, and E_b is the average energy per information bit.

At high SNR (i.e., in the floor region), the probability of bit error P_b can then be approximated as [22]

$$P_b \approx \sum_{w=2}^{t} \frac{w n_w}{N} Q\left(\sqrt{\frac{d_{w,\min} r E_b}{N_0}}\right), \tag{2.19}$$

where n_w is the number of weight w input information sequences that result in the lowest weight codeword, denoted by $d_{w,\min}$. Thus, (2.19) suggests that, to find an upper bound on P_b in the floor region, it suffices to find the minimum distance $d_{w,\min}$ and its multiplicity n_w for all weight-t or lower input patterns. Typically, $1 < t \leq 3$. Note that weight-1 input patterns are excluded due to the recursive nature of the constituent encoders as they result in very high-weight codewords. Equation (2.19) holds for both of the PCC and SCC codes, except that the parameters n_w and $d_{w,\min}$ may differ due to the different structures these codes have.

2.4.2 Fading Channels

When an outer channel code is concatenated with a STBC code over fading channels, both time and space diversity gains can be achieved [23]. In such cases, the maximum diversity order that can be achieved is $M_t M_r d_{\min}$, where $d_{\min}$ denotes the minimum *Hamming distance* of the outer channel code. Such diversity gains may be achieved when the underlying STBC code is orthogonal and the channel is fully interleaved. The latter condition is normally referred to as *ideal interleaving*, which can be accomplished by using proper interleaving between the two codes (see Fig. 2.1b.)

Assuming that the channel state information (CSI) is perfectly known at the receiver, the conditional pairwise error probability that the receiver will select codeword

k over the all-zero codeword conditioned on the channel gains is given by

$$\begin{aligned} P_2\left(d_k \mid \boldsymbol{\alpha}\right) &= Q\left(\sqrt{2\frac{r}{N}\frac{E_b}{N_0}\sum_{n=1}^{d_k}\sum_{j=1}^{M_r}\sum_{i=1}^{M_t}\left|\alpha_{n,i,j}\right|^2}\right) \\ &= Q\left(\sqrt{2\frac{r}{N}\frac{E_b}{N_0}\sum_{n=1}^{M_t M_r d_k}\left|\alpha_n\right|^2}\right). \end{aligned} \tag{2.20}$$

To compute the average pairwise error probability, we average the expression in (2.20) with respect to the probability density function (pdf) of the random variables $|\alpha_n|^2$ for $n = 1, \ldots, M_t M_r d_k$. To simplify the analysis, we first introduce an auxiliary random variable that we denote by X, defined as

$$X = \sum_{n=1}^{M_t M_r d_k}\left|\alpha_n\right|^2.$$

Note that X is Chi-square distributed with $2M_t M_r d_k$ degrees of freedom and whose pdf is given as [23]

$$f_X(x) = \frac{1}{(M_t M_r d_k - 1)!}x^{M_t M_r d_k - 1}e^{-x}, \quad x \geq 0.$$

Consequently, the average pairwise error probability can be shown to be [23]

$$\begin{aligned} P_2(d_k) = &\left[\frac{1}{2}\left(1 - \sqrt{\frac{\gamma_s}{1+\gamma_s}}\right)\right]^{M_t M_r d_k} \\ &\cdot \sum_{n=0}^{M_t M_r d_k - 1}\binom{M_t M_r d_k - 1 + n}{n}\left[\frac{1}{2}\left(1 + \sqrt{\frac{\gamma_s}{1+\gamma_s}}\right)\right]^n, \end{aligned} \tag{2.21}$$

where $\gamma_s = \frac{r}{N}\frac{E_b}{N_0}E[|\alpha_n|^2] = \frac{r}{N}\frac{E_b}{N_0}$ and $E[\cdot]$ denotes the expectation operator. When the SNR is sufficiently large, the expression in (2.21) can be approximated by [23]

$$P_2(d_k) \approx \binom{2M_tM_rd_k - 1}{M_tM_rd_k}\left(4\frac{r}{N}\frac{E_b}{N_0}\right)^{-M_tM_rd_k}. \tag{2.22}$$

Clearly, at high SNR, the performance is dominated by the minimum Hamming distance of the outer code, which we denote by $d_{\min}$. Consequently, the maximum diversity order is $M_tM_rd_{\min}$. The average probability bit error rate (Pb) for this scheme is then upper bounded as

$$P_b \leq \sum_{d_k=d_{\min}}^{\infty} \beta_{d_k}P_2(d_k), \tag{2.23}$$

where β_{d_k} is the multiplicity corresponding to distance d_k.

2.5 Simulation Results

The simulation model is depicted in Fig. 2.1. The PCC scheme uses two identical RSC encoders, where in one case each RSC encoder employs the generator polynomials $(g_1, g_2) = (7, 5)_{oct}$ (4 states), whereas in the other case each RSC encoder employs the generator polynomials $(g_1, g_2) = (31, 33)_{oct}$ (16 states). The SCC scheme uses an outer RSC code whose encoder employs the generator polynomial $(g_1, g_2) = (7, 5)_{oct}$, and a rate-1 inner code employing a differential encoder with transfer function $\frac{1}{1+D}$. All interleavers used in these simulations are randomly generated. In the fading channel

case, we consider the cases of $M_t = 2$, and $M_r = 1, 2$.

2.5.1 AWGN Channel

In Fig. 2.6, we present the bit error performance (P_b) simulation results on an AWGN channel for the PCC system with the 4-state code, code rate 1/2, data block length $N = 512$, and 8 decoder iterations. The PCC decoder uses two iterative MAPPs, two APPs, two conventional SOVAs, two SOVA-SAs, or two MSOVAs.

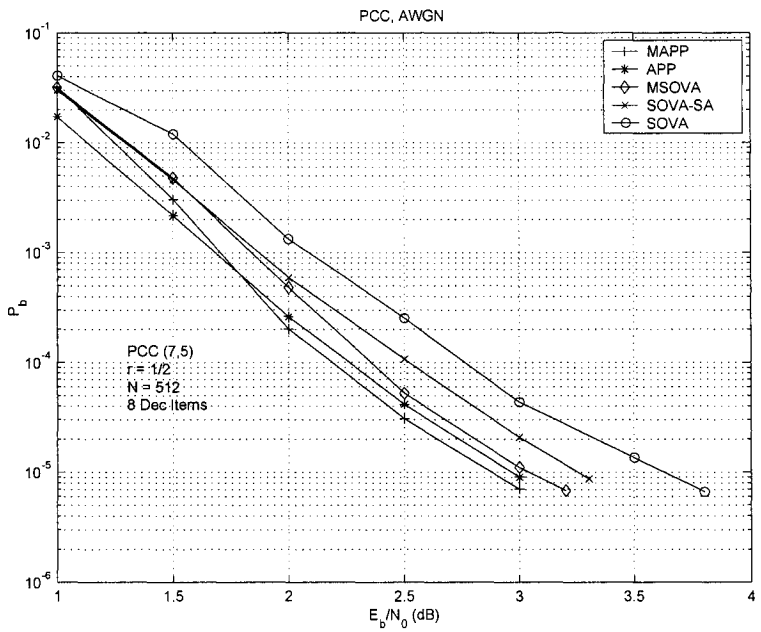


Figure 2.6: Bit error rate performance comparison between the SOVA, SOVA–SA, MSOVA, APP, and MAPP decoders in AWGN channels. [PCC scheme with 4-state RSC encoders, rate 1/2, $N = 512$, and 8 decoder iterations.]

We observe from the figure that the MSOVA achieves a performance improvement of about 0.6 dB relative to the SOVA, and is only about 0.2 dB away from the MAPP, all at $P_b = 10^{-5}$. In contrast, the SOVA-SA achieves a performance improvement of

only 0.3 dB relative to the SOVA at $P_b = 10^{-5}$. It is also interesting to observe that the MAPP is superior to the APP by about 0.2 dB at at $P_b = 10^{-5}$.

In Fig. 2.7, we repeat the same experiment mentioned above except now we use the 16-state code and code rate 4/5. We observe from the figure performance improvements similar to those reported in Fig. 2.6. We also plot on the same figure the ML bound for this code using (2.19) with the following parameters: $n_2 = 9$, $d_{2,\min} = 5$; and $n_3 = 1$, $d_{3,\min} = 3$. These parameters were obtained using semi-analytical techniques [22].

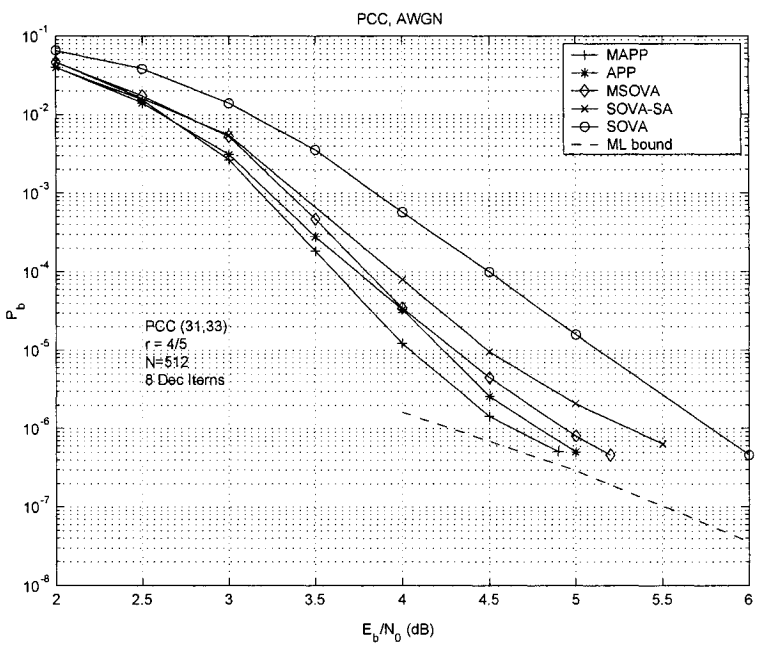


Figure 2.7: Bit error rate performance comparison between the SOVA, SOVA–SA, MSOVA, APP, and MAPP decoders in AWGN channels. [PCC scheme with 16-state RSC encoders, rate 4/5, $N = 512$, and 8 decoder iterations.]

In Fig. 2.8, we plot simulation results for the SCC system with following parameters: 4-state outer encoder with rate 1/2, 2-state differential encoder with rate 1, interleaver size $N = 512$, and 8 decoder iterations. As shown in the figure, the MSOVA improves the performance by about 1.2 dB relative to the SOVA at $P_b = 10^{-5}$. Moreover, the MSOVA is superior to the APP by about 0.2 dB. It is also observed from the figure that the improvement provided by the MAPP, relative to the APP, is about 0.3 dB at $P_b = 10^{-5}$. The values of c and d used for the MAPP were $(1.0, 1.1)$ and $(0.7, 0.8)$ for the inner and outer decoders, respectively.

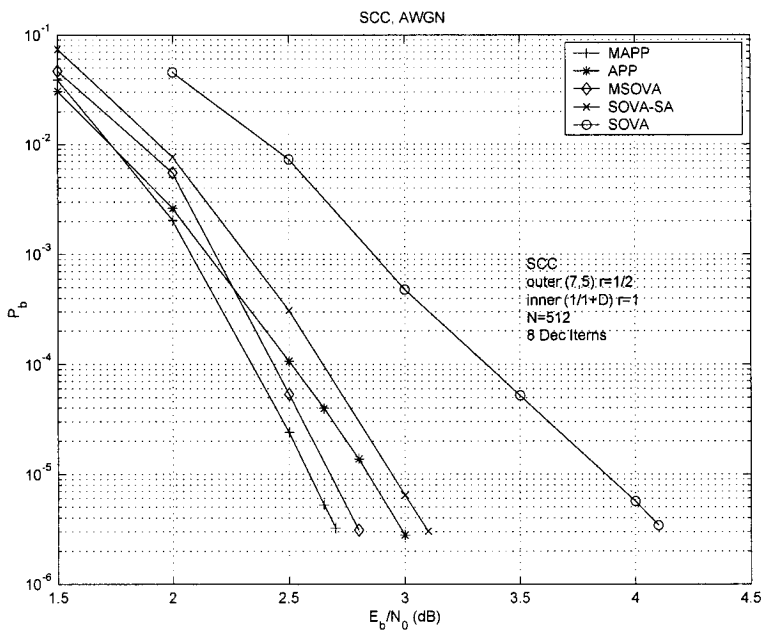


Figure 2.8: Bit error rate performance comparison between the SOVA, SOVA–SA, MSOVA, APP, and MAPP decoders in AWGN channels. [SCC scheme with 4-state outer code, differential encoder for the inner code, overall rate 1/2, $N = 512$, and 8 decoder iterations.]

2.5.2 Fading Channel

In Fig. 2.9, we plot the P_b performance of the SOVA, MSOVA and APP for the PCC system on the flat fading channel. The simulated system parameters are: 16-state RSC encoders, $r = 4/5$, $N = 512$, 8 decoder iterations, and $M_t = 2, M_r = 1, 2$.

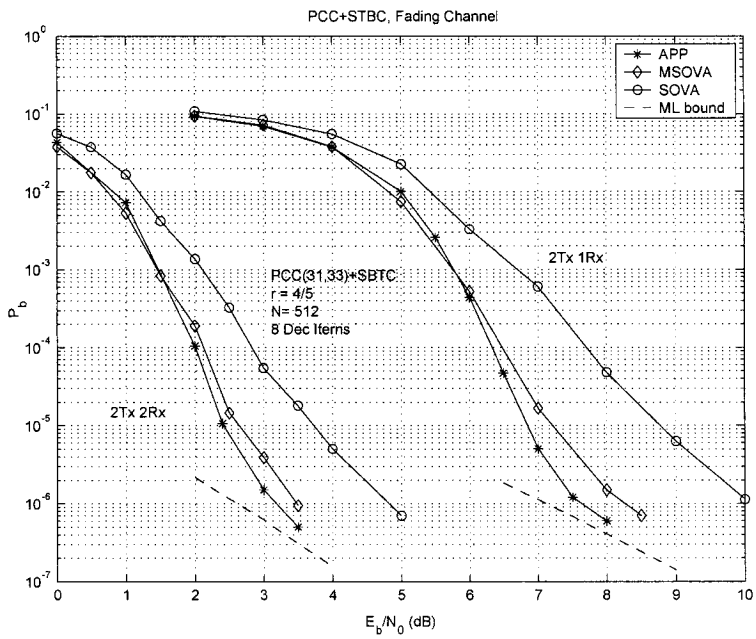


Figure 2.9: Bit error rate performance comparison between the SOVA, MSOVA and APP decoders in flat fading channels. [PCC scheme with 16-state RSC encoders, rate 4/5, $N = 512$, and 8 decoder iterations.]

We observe from the figure the substantial improvements achieved by the MSOVA over the conventional SOVA. In the case of $M_r = 1$, the SOVA is inferior to the APP by about 2.0 dB at $P_b = 10^{-5}$, where most of degradation is recovered by the MSOVA. In the $M_r = 2$ case, the MSOVA performs within 0.2 dB at $P_b = 10^{-5}$ from the APP, whereas the SOVA is about 1.6 dB away from the APP. Furthermore, we plot on

the same figure the ML performance bounds for the cases $M_r = 1, 2$ in the floor region. The diversity order achieved by this code is 6 and 12 for the $M_r = 1$, and 2, respectively.

We repeat the above experiment with the PCC code replaced by the SCC code. In this experiment we only consider the case $M_t = 2, M_r = 1$. The simulation results are plotted in Fig. 2.10. We observe from the figure that the MSOVA improves the performance by about 1.2 dB and 0.4 dB at $P_b = 10^{-5}$ relative to the SOVA and APP, respectively. Furthermore, the MAPP provides a performance improvement of about 0.6 dB over the APP, and is better than the MSOVA by only 0.2 dB at $P_b = 10^{-5}$.

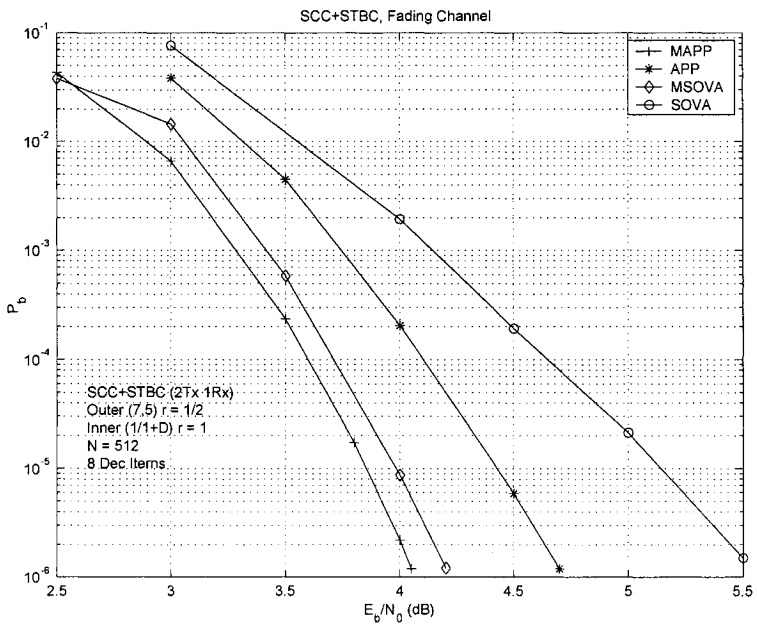


Figure 2.10: Bit error rate performance comparison between the SOVA, MSOVA, APP, and MAPP decoders in flat fading channels. [SCC scheme with 4-state outer code, differential encoder for the inner code, overall rate 1/2, $N = 512$, and 8 decoder iterations.]

The experiment whose results are reported in Fig. 2.10 is repeated with code rate 8/9 and interleaver size 1024. The simulation results are plotted in Fig. 2.11. We also plot on the same figure the corresponding ML bound using (2.23) with the following parameters: $n_2 = 1$, $d_{2,\min} = 2$; and $n_3 = 1$, $d_{3,\min} = 4$. We observe from the figure that the MSOVA and APP give essentially identical performances, and both perfectly agree with the ML bound. Furthermore, the gap between the MSOVA and SOVA is about 1.8 dB at $P_b = 10^{-5}$, which is a remarkable improvement. Simulation results for the MAPP are not included because very marginal improvements were achieved by the MAPP over the APP. This is attributed to the fact that the APP already achieves the ML bound where there is no room for further improvement.

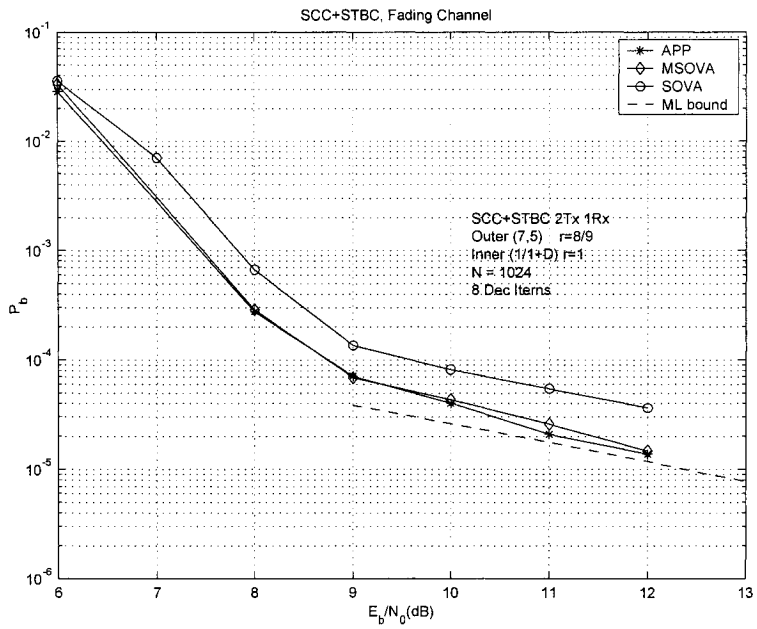


Figure 2.11: Bit error rate performance comparison between the SOVA, MSOVA, and APP decoders in flat fading channels. [SCC scheme with 4-state outer code, differential encoder for the inner code, overall rate 8/9, $N = 1024$, and 8 decoder iterations.]

It is clear from the above examples that the proposed modifications are more effective for the SCC scheme as compared to the PCC scheme. We now give some insight for this phenomenon. By looking closely at Figs. 2.3 and 2.4, we observe that the extrinsic values produced by the SOVA for the SCC scheme are much more exaggerated than their counterparts for the PCC scheme. The same phenomenon is experienced in Fig. 2.5 for the APP decoder. This suggests that there is more room for improvement in the SCC than in the PCC case.

2.6 Concluding Remarks

In this chapter, we have introduced simple modifications to the conventional SOVA in an effort to attenuate the optimistic extrinsic information at its output. We showed that the reason behind producing these exaggerated values is the strong correlation between the input and output of the SOVA. The proposed modifications were aimed at reducing this inherent correlation, which would ultimately lead to producing more realistic extrinsic information. We have examined the performance of the MSOVA on both AWGN and fading channels with favorable results. We have also extended the proposed modifications to the APP algorithm where significant performance improvements were shown possible, particularly for the SCC scheme.

Chapter 3

Improvements in SOVA-based Decoding for Turbo-coded Data Storage Channels

3.1 Introduction

Considerable research has been performed over the last few years [24]−[29] on the application of turbo coding to partial response channels. In [24] and [25], a parallel concatenated (turbo) coded partial response Class IV (PR4) system is considered where the precoded PR4 channel is treated as an inner code. The PR4 channel is detected using an APP detector matched to the precoded PR4 trellis, and the turbo code is decoded using two APP decoders matched to the two RSC encoders. The PR4 detector may or may not share soft information with the turbo decoding, depending

on performance/complexity requirements [24], [25]. The turbo code may be replaced by a single convolutional encoder as in [26]−[29], effecting a serial concatenated code system with the precoded PR4 channel acting as the inner code. Obviously, three APP decoders are required in the PCC system whereas two APP decoders are required in the SCC system. In both cases, it was shown via simulations that a substantial performance improvement is achieved over the uncoded system [24]−[29] (see also references therein.)

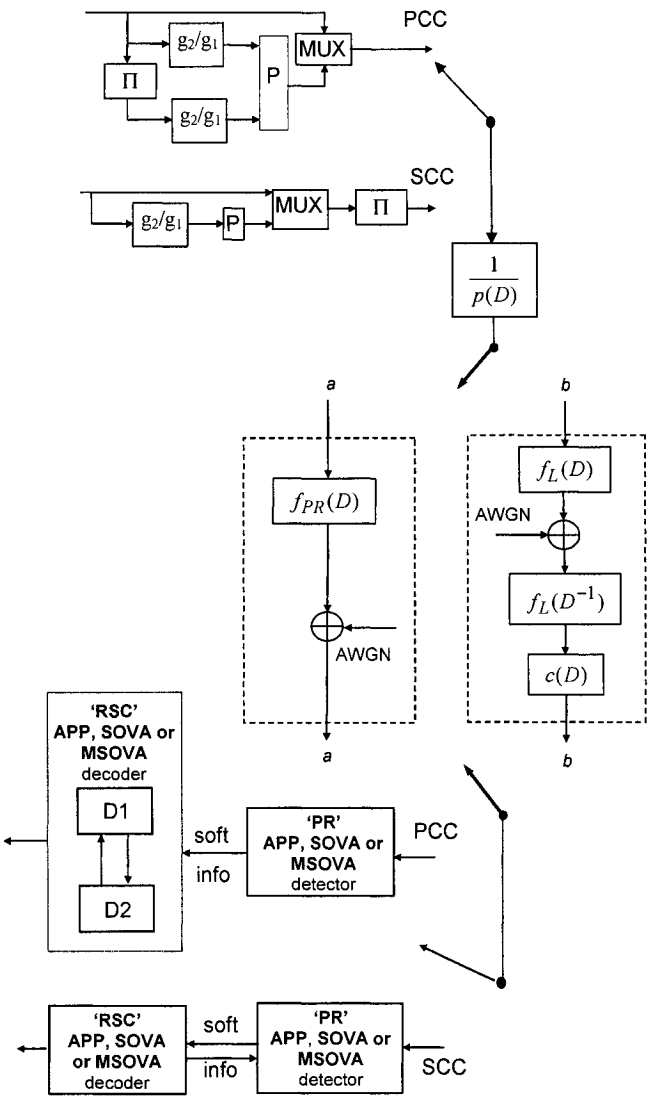
Some degradation is suffered when the APP algorithm is replaced with the SOVA on band-limited channels such as PR channels. For instance, as we will see later in this chapter, the degradation due to using the SOVA can be as large as 2.0 dB at bit error rate 10^{-5} relative to the APP. Inspired by the substantial improvements of the MSOVA relative to the SOVA algorithm on AWGN and fading channel in Chapter 2, in this chapter, we propose a simple approach for dealing with the exaggerated reliability values produced by the SOVA algorithm on partial response channels. We first pin point the reason as to why the SOVA tends to produce these exaggerated reliability values at its output, and then propose simple modifications to overcome this problem. We argue that the reason behind these large reliability values is mostly the *high correlation* between the intrinsic information (inputs to the SOVA) and extrinsic information (outputs of the SOVA). Our proposed remedy for this problem is based on mathematical analysis and it involves using two attenuators, one applied to the immediate output of the SOVA and another applied to the extrinsic information before it is passed on to the other decoder component (assuming iterative decoding.)

As we will see later, theoretically speaking, these attenuators should be updated for every received data block as they depend on the means and variances of the intrinsic and extrinsic information of the SOVA, as well as the correlation between them. However, performing this update at the block rate is computationally complex. As an alternative, we use Monte-Carlo simulations to determine the values of these attenuators that would give the best performance. We then use these values for all cases, including all range of SNR, and all iterations. While fixing these attenuators, we examine the performance of the modified SOVA (MSOVA) on idealized PR channels and the Lorentzian channel equalized to a PR target for both PCC and SCC systems. We demonstrate that improvements of up to 1.6 dB at $P_b = 10^{-5}$ are possible, relative to the SOVA.

The rest of the chapter is organized as follows. In Section 3.2, we describe the system model. In Section 3.3, we present the proposed MSOVA algorithm and discuss the rationale behind proposing these new modifications. We present and discuss the simulation results in Section 3.4. Finally, Section 3.5 concludes the chapter.

3.2 System and Channel Model

In this chapter, we consider two channel models: the idealized PR channel model which is modeled by the polynomial $f_{PR}(D)$ followed by a white noise process (Fig. 3.1, *a*-*a*), and the Lorentzian channel equalized to a PR target (Fig. 3.1, *b*-*b*). We now describe the Lorentzian model.



LEGEND:

Π - permuter

P - puncturer

PCC - parallel concatenation code

SCC - serial concatenation code

Figure 3.1: System and channel model.

We consider the recording head, the magnetic medium, and the read head to be a single linear system with transition response equal to the Lorentzian pulse [30]

$$h(t) = \sqrt{\frac{4\,E_i}{\pi\, pw_{50}}}\ \frac{1}{1+(2t/pw_{50})^2}, \tag{3.1}$$

where pw_{50} is the width of the pulse measured at half its height and E_i is the energy in the "isolated pulse." Thus, when an NRZ signal of the form $\sum_k a_k\, p(t-kT_c)$ is to be recorded ($a_k \in \{\pm 1\}$, $p(t) = 1$ for $t \in [0, T_c]$, and 0 elsewhere), the response of the channel is

$$r(t) = \sum_k \frac{a_k}{2}\ [h(t-kT_c) - h(t-(k+1)T_c)] + w(t). \tag{3.2}$$

In (3.2), T_c is the recorded bit duration and is related to the user bit duration T_u via the code rate r as $T_c = rT_u$; $w(t)$ is assumed to be white Gaussian noise with power spectral density $N_0/2$ for simplicity, although a more accurate model will include correlated media noise.

We assume the optimal receiver front end, which is a filter matched to the "dibit" $s(t) = h(t) - h(t-T_c)$ followed by a symbol-rate $(1/T_c)$ sampler. This leads to a discrete time equivalent channel response $f_L(D)$, where $f_L(D)f_L(D^{-1})$ is a factorization of the sampled autocorrelation function, $R_s(D)$, of $s(t)$. We use the (noncausal) factorization of Bergmans [31] for which $f_L(D) = (1-D)g(D)$ with the coefficients

of $g(D)$ given by

$$g_k = \left[\frac{E_i S_c}{2\pi}\tanh(\pi S_c/2)\right]^{1/2} \frac{k + S_c/2}{k^2 + (S_c/2)^2}\,,\ k \in \mathbf{Z}, \tag{3.3}$$

where $S_c \triangleq pw_{50}/T_c$ is defined to be the channel density and is related to the user density $S_u \triangleq pw_{50}/T_u$ via $S_c = S_u/r$.

The discrete-time magnetic recording channel (MRC) model then consists of a filter with response $f_L(D)$ and a white noise process, $w(D)$, with spectral density $N_0/2$ added to this filter's output. The front end of the discrete-time receiver is the matched filter $f_L(D^{-1})$.

3.3 Modified SOVA Algorithm

We discuss in this section the SOVA algorithm as an approximation to the BCJR-APP algorithm for the SCC system; SOVA decoding for the PCC system is similar. As mentioned earlier, the SCC system requires two SOVAs. One SOVA decoder is matched to the RSC encoder, and the second SOVA is a detector matched to the precoded PR trellis. Hagenauer in [6] and [19] derived the SOVA algorithm for binary trellises. The SOVA algorithm was modified in [32] for PR trellises. For the purpose of making this chapter self-contained, we summarize in the following section some of the main results reported in [32] that are related to our work.

3.3.1 The SOVA for PR Trellises

The branches of the trellis of a PR channel are labeled with the information bit u_k and its corresponding channel symbol c_k. Let $\mathbf{c}_1^N = [c_1\ c_2\ \ldots\ c_N]$ be the PR channel output sequence that corresponds to a block of information bits $\mathbf{u}_1^N = [u_1\ u_2\ \ldots\ u_N]$, and $\mathbf{y}_1^N = [y_1\ y_2\ \ldots\ y_N]$ be the corresponding noisy received sequence. Additive white Gaussian noise (AWGN) of variance $N_0/2$ is assumed.

At any given time k, a block-wise APP detector searches for the state sequence $\mathbf{s}_{1,m}^k = [s_{1,m}\ s_{2,m}\ \ldots\ s_{k,m}]$ that corresponds to the information sequence $\mathbf{u}_1^k$ by maximizing the *a posteriori* probability

$$M_{k,m} \triangleq P\left(\mathbf{s}_{1,m}^k \mid \mathbf{y}_1^k\right), \tag{3.4}$$

where m is an index that corresponds to the trellis path corresponding to the state sequence $\mathbf{s}_{1,m}^k$. Since $\mathbf{y}_1^k$ does not depend on m then maximizing (3.4) is equivalent to maximizing

$$P\left(\mathbf{y}_1^k \mid \mathbf{s}_{1,m}^k\right) \cdot P\left(\mathbf{s}_{1,m}^k\right). \tag{3.5}$$

Now, the term $P\left(\mathbf{s}_{1,m}^k\right)$can be expressed as

$$\begin{aligned} P\left(\mathbf{s}_{1,m}^k\right) &= P\left(\mathbf{s}_{1,m}^{k-1}\right) \cdot P\left(s_{k,m}\right) \\ &= P\left(\mathbf{s}_{1,m}^{k-1}\right) \cdot P\left(u_{k,m}\right), \end{aligned} \tag{3.6}$$

where $u_{k,m}$ is the information bit that corresponds to the state transition $s_{k-1,m} \rightarrow s_{k,m}$. We thus have from (3.4)−(3.6),

$$\max_m \{M_{k,m}\} = \max_m \begin{Bmatrix} P\left(\mathbf{s}_{1,m}^{k-1}\right) \prod_{i=1}^{k-1} p\left(y_i \mid s_{i-1,m}, s_{i,m}\right) \\ \cdot P\left(u_{k,m}\right) p\left(y_k \mid s_{k-1,m}, s_{k,m}\right) \end{Bmatrix}, \tag{3.7}$$

where $p\left(y_k \mid s_{k-1,m}, s_{k,m}\right) = p\left(y_k \mid c_{k,m}\right)$. In maximizing (3.7), it is equivalent if we take its logarithm, and add two constants that are independent of m. Let the two constants be $K_1 = -\frac{1}{2}\log\left[P\left(u_k = +1\right) \cdot P\left(u_k = -1\right)\right]$ and $K_2 = \log\sqrt{\pi N_0}$. Also, denote the logarithm of the first line in (3.7) by $M_{k-1,m}/2$ which corresponds to the cumulative metric at time $k-1$ along path m. With this, (3.7) becomes

$$\begin{aligned} \max_m \{M_{k,m}\} &= \max_m \{M_{k-1,m} + [\log P\left(u_{k,m}\right) + K_1] \\ &\quad + [\log p\left(y_k \mid s_{k-1,m}, s_{k,m}\right) + K_2]\}, \end{aligned} \tag{3.8}$$

which simplifies to

$$\max_m \{M_{k,m}\} = \max_m \begin{Bmatrix} M_{k-1,m} - \frac{1}{N_0}(y_k - c_{k,m})^2 \\ + \frac{1}{2} u_{k,m} L_a(u_{k,m}) \end{Bmatrix}, \tag{3.9}$$

where $L_a(u_{k,m}) = \log\frac{P(u_{k,m}=+1)}{P(u_{k,m}=-1)}$ is the *a priori* (extrinsic) information on bit $u_{k,m}$ that is usually obtained from another decoder assuming iterative decoding.

It follows from (3.9), after multiplying both sides by -1, that the cumulative metric $M_k(s)$ for state s at time k along some arbitrary path m is updated according

to

$$M_k(s) = \min\left\{\lambda(s', s) + M_{k-1}(s'), \lambda(s'', s) + M_{k-1}(s'')\right\}, \tag{3.10}$$

where $\lambda(s', s)$ is the branch metric for the transition from state s' to state s at time k which is defined as

$$\lambda(s', s) = \frac{1}{N_0}(y_k - c_k)^2 - \frac{1}{2}u_k L_a(u_k), \tag{3.11}$$

and c_k is the PR channel output at time k corresponding to the transition $s' \rightarrow s$. The branch metric $\lambda(s'', s)$ is similarly defined. Now define the difference metric for state s at time k as [6]

$$\Delta_k = \left|(M_{k-1}(s') + \lambda(s', s)) - (M_{k-1}(s'') + \lambda(s'', s))\right|. \tag{3.12}$$

In [6], Δ_k was also shown to be approximated as

$$\Delta_k \approx \log \frac{P(\text{correct})}{1 - P(\text{correct})}, \tag{3.13}$$

where $P(\text{correct})$ is the probability that the path decision of the survivor at time k was correct. Therefore, Δ_k represents the reliability that the path ending at state s at time k was correct.

In the conventional SOVA, to obtain the soft output for bit u_k, we first obtain the

hard decision $\hat{u}_k$ after a delay δ (i.e., at time $k+\delta$) , where δ is the *decoding depth.* At time $k+\delta$, we select the surviving path that ends at the state that has the lowest metric and the selected path is considered to be the *maximum-likelihood* (ML) path. We trace back the ML path to obtain the hard decision $\hat{u}_k$. Along the ML path, there are $\delta+1$ nonsurviving paths that have been discarded, and each nonsurviving path has a certain difference metric Δ_j where $k \leq j \leq k+\delta$. (Clearly, this is because along the ML path there are $\delta+1$ states, and each state has a difference metric that was calculated using (3.12).)

Now define

$$\Delta_k^* = \min\{\Delta_k, \Delta_{k+1}, \ \ldots \ , \Delta_{k+\delta}\}, \tag{3.14}$$

where the minimum is taken *only* over the nonsurviving paths within the time window $[k\,,\ k+\delta]$ that would have led to a different decision $\hat{u}_k$. It was shown in [6] that Δ_k^* represents the reliability of the hard decision $\hat{u}_k$, and the reliability increases with increasing δ. Given $\hat{u}_k$ and Δ_k^*, the soft output of the *Viterbi algorithm* for bit u_k is approximated by [6]

$$L_{sova}(\hat{u}_k) \approx \hat{u}_k \cdot \Delta_k^*. \tag{3.15}$$

From Fig. 3.2, and Eqns. (3.11) and (3.12) it can be shown that Δ_k^* has the

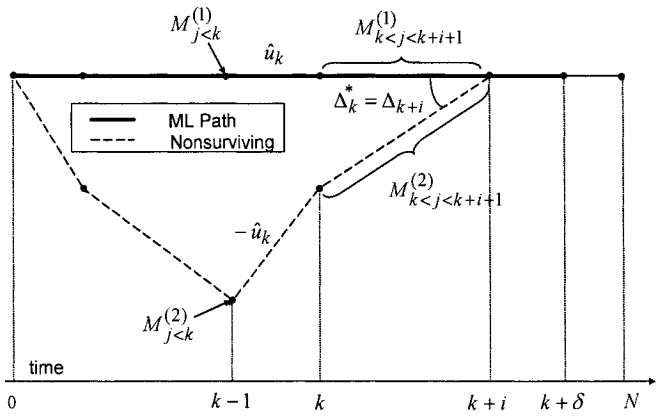


Figure 3.2: The SOVA output for bit u_k in PR trellises

following structure [6]

$$\begin{aligned}\Delta^*_k &= \left(M^{(2)}_{j<k} - M^{(1)}_{j<k}\right) + \left(M^{(2)}_{k<j<k+i} - M^{(1)}_{k<j<k+i}\right) \\ &\quad + \frac{1}{N_0}\left((y_k - c_{k2})^2 - (y_k - c_{k1})^2\right) \\ &\quad + \hat{u}_k L_a(u_k), \end{aligned} \tag{3.16}$$

where c_{k1}, and c_{k2} are the nominal channel outputs to state transitions at time k along paths 1 and 2, respectively, i is the time index at which $\Delta^*_k = \Delta_{k+i}$, and $M^{(1)}_{j<k}$ is the cumulative metric at time $k-1$ along path 1. The other terms are obviously defined. Substituting (3.16) into (3.15) yields

$$
\begin{aligned}
L_{sova}(\hat{u}_k) \approx\ & \hat{u}_k \left(M^{(2)}_{j<k} - M^{(1)}_{j<k} \right) \\
& + \hat{u}_k \left(M^{(2)}_{k<j<k+i} - M^{(1)}_{k<j<k+i} \right) \\
& + \frac{1}{N_0} \hat{u}_k \left((y_k - c_{k2})^2 - (y_k - c_{k1})^2 \right) \\
& + L_a(u_k), \qquad (3.17)
\end{aligned}
$$

where the first three terms represent the extrinsic information, which we denote by $L_e(\hat{u}_k)$, i.e., $L_e(\hat{u}_k) = L_{sova}(\hat{u}_k) - L_a(u_k)$. $L_e(\hat{u}_k)$ is then passed to the subsequent decoder to be used as an *a priori* information in the next iteration.

3.3.2 Proposed Modifications

It has been shown in [20] that the extrinsic information $L_e(\hat{u}_k)$ follows a Gaussian distribution, and so does $L_a(u_k)$ since it is a combination of the channel value and the extrinsic information passed from the other decoder. In the conventional SOVA algorithm, it is normally assumed that the terms $L_e(\hat{u}_k)$ and $L_a(u_k)$ are weakly correlated, and thus $L_e(\hat{u}_k)$ can be obtained by directly subtracting $L_a(u_k)$ from $L_{sova}(\hat{u}_k)$. That is

$$
L_e(\hat{u}_k) = L_{sova}(\hat{u}_k) - L_a(u_k). \qquad (3.18)
$$

However, we have observed through computer simulations that the correlation

between $L_a(u_k)$ and $L_e(\hat{u}_k)$ is rather strong. We believe that this is essentially the reason behind the degradation in the SOVA performance. To elaborate on this, we have listed in Table 3.1 values for the correlation coefficient between the intrinsic and extrinsic information of the SOVA at SNR = 12 dB for different decoder iterations.

Table 3.1: Correlation coefficient for the PCC and SCC codes on the equalized PR4 channels, Rate=16/17

Correlation Coefficient for PCC				
# Iterns at 12 dB	1	2	5	8
APP	0.02	0.02	0.03	0.03
MSOVA	0.08	0.09	0.07	0.07
SOVA	0.19	0.28	0.33	0.34

Correlation Coefficient for SCC				
APP	0.02	0.02	0.02	0.02
MSOVA	0.10	0.07	0.06	0.06
SOVA	0.20	0.20	0.28	0.31

Listed in the same table as well are the correlation coefficient values for the MSOVA and APP. In this experiment, we simulate both the PCC and SCC codes over the equalized-PR4 channel for code rate 16/17. For the same concatenation scheme, we use the same bit and noise sequences for the three decoding algorithms. As is evident from the table, there is clearly a strong correlation between the extrinsic and intrinsic information of the SOVA, and it appears that this correlation increases with the number of iterations. It is also evident from the table that this correlation is significantly reduced in the MSOVA case.[1] (Similar observations were made for

[1]Since the correlation coefficient remains small in the MSOVA case for all iterations, one may except to see constant performance improvements by increasing the number of iterations. However, this is not the case since the extrinsic information converges to some value after a certain number of iterations.

different SNRs and for the idealized PR4 channel.)

The consequence of this correlation is that the extrinsic information estimated by the SOVA is larger than the its true value. This is explained as follows. As the name suggests, the extrinsic information passed from one decoder to another should present *new* information to the receiving decoder. However, this strong correlation implies that part of the extrinsic information passed to a decoder is already known to this decoder. However, it is not treated as such, i.e., the receiving decoder normally regards all what it receives from the other decoder as new information. Consequently, this leads to a redundancy in the exchanged information, which tends to produce exaggerated extrinsic information at the output of the SOVA.

In light of the above discussion, it is clear that the intrinsic and extrinsic information of the SOVA need to be decorrelated before the latter is passed on to the second decoder. Now let us denote the actual SOVA output by $V(\hat{u}_k)$ and the corresponding extrinsic information by $V_e(\hat{u}_k)$, where the latter can be expressed as

$$V_e(\hat{u}_k) \triangleq V(\hat{u}_k) - L_a(u_k), \tag{3.19}$$

with $L_a(u_k)$ representing the intrinsic information (inputs to the SOVA). Our objective here is to modify $V(\hat{u}_k)$ and $V_e(\hat{u}_k)$ in order to obtain the true $L_a(u_k)$ and $L_e(\hat{u}_k)$, which are supposed to be uncorrelated. As per the above discussion, the variables $L_a(u_k)$ and $V_e(\hat{u}_k)$ are correlated Gaussian random variables with means m_a, m_e, and variances σ_a^2, σ_e^2, respectively. Assuming $u_k = +1$ is transmitted, the joint conditional

probability density function (pdf) $P(V_e(\hat{u}_k), L_a(u_k) \mid u_k = +1)$ is then given as

$$P(V_e(\hat{u}_k), L_a(u_k) \mid u_k = +1) = \frac{1}{2\pi\sigma_a\sigma_e\sqrt{1-\rho_+^2}} \cdot \exp\left(-\frac{1}{2\left(1-\rho_+^2\right)}\left[\frac{(V_e(\hat{u}_k)-m_e)^2}{\sigma_e^2} + \frac{(L_a(u_k)-m_a)^2}{\sigma_a^2}\right]\right) \cdot \exp\left(\frac{\rho_+(V_e(\hat{u}_k)-m_e)(L_a(u_k)-m_a)}{\sigma_a\sigma_e\left(1-\rho_+^2\right)}\right), \quad (3.20)$$

where ρ_+ is the correlation coefficient given by

$$\rho_+ = \frac{E\left[(V_e(\hat{u}_k)-m_e)(L_a(u_k)-m_a)\right]}{\sigma_a\sigma_e}. \quad (3.21)$$

where m_e and m_a are estimated as $m_e = \frac{1}{N}\sum_{k=1}^{N} V_e(\hat{u}_k)$ and $m_a = \frac{1}{N}\sum_{k=1}^{N} L_a(u_k)$, respectively, and N is the data block size. Also, σ_e and σ_a are estimated as $\sigma_e = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left[m_e - V_e(\hat{u}_k)\right]^2}$ and $\sigma_a = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left[m_a - L_a(u_k)\right]^2}$, respectively. Similarly, when $u_k = -1$ is transmitted, the joint conditional pdf $P(V_e(\hat{u}_k), L_a(u_k) \mid u_k = -1)$ is given as

$$P(V_e(\hat{u}_k), L_a(u_k) \mid u_k = -1) = \frac{1}{2\pi\sigma_a\sigma_e\sqrt{1-\rho_-^2}} \cdot \exp\left(-\frac{1}{2\left(1-\rho_-^2\right)}\left[\frac{(V_e(\hat{u}_k)+m_e)^2}{\sigma_e^2} + \frac{(L_a(u_k)+m_a)^2}{\sigma_a^2}\right]\right) \cdot \exp\left(\frac{\rho_-(V_e(\hat{u}_k)+m_e)\cdot(L_a(u_k)+m_a)}{\sigma_a\sigma_e\left(1-\rho_-^2\right)}\right), \quad (3.22)$$

where

$$\rho_- = \frac{E\left[(V_e(\hat{u}_k) + m_e)(L_a(u_k) + m_a)\right]}{\sigma_i \sigma_e}. \tag{3.23}$$

Using Bayes' rule, the *a posteriori* log-likelihood ratio $L(\hat{u}_k)$ can be computed as

$$\begin{aligned} L(\hat{u}_k) &= \ln \frac{P(u_k = +1 \mid V_e(\hat{u}_k), L_a(u_k))}{P(u_k = -1 \mid V_e(\hat{u}_k), L_a(u_k))} \\ &= k_1 V_e(\hat{u}_k) + k_2 L_a(u_k), \end{aligned} \tag{3.24}$$

where

$$k_1 = \frac{1}{1-\rho^2}(r_e - \rho r_i \frac{\sigma_a}{\sigma_e}), \tag{3.25}$$

$$k_2 = \frac{1}{1-\rho^2}(r_a - \rho r_e \frac{\sigma_e}{\sigma_a}), \tag{3.26}$$

$r_e = 2m_e/\sigma_e^2$, $r_a = 2m_a/\sigma_a^2$, and $\rho = \rho_+ = \rho_-$.

Substituting (3.19) into (3.24) yields

$$L(\hat{u}_k) = k_1 \left(V(\hat{u}_k) - L_a(u_k)\right) + k_2 L_a(u_k), \tag{3.27}$$

and, consequently, the extrinsic information to be passed to the other SOVA decoder,

$L_e(\hat{u}_k)$, can be expressed as

$$\begin{aligned} L_e(\hat{u}_k) &= L(\hat{u}_k) - L_a(u_k) \\ &= c\left[dV(\hat{u}_k) - L_a(u_k)\right]. \end{aligned} \tag{3.28}$$

where $c = k_1 - k_2 + 1$ and $d = \frac{k_1}{k_1 - k_2 + 1}$. Equation (3.28) suggests that the immediate output of the SOVA should be scaled by d before the intrinsic information is subtracted off, and then the difference should be scaled by c (see Fig. 3.3.)

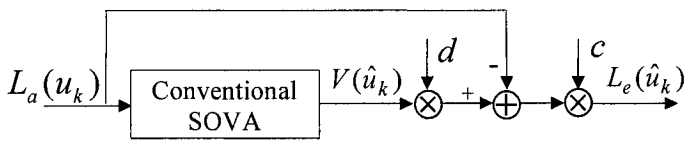


Figure 3.3: Implementation of the MSOVA algorithm

We finally remark that the pair of values (c, d) is always the same for both decoder components in the PCC system, whereas they may not be the same in the SCC system. The reason being is that these values depend on the structure of the code, as well as the puncturing scheme. Therefore, when selecting the values of c and d, the inner and outer decoders in the SCC scheme should be treated separately.

3.3.3 Computation of the Coefficients c and d

In general, as mentioned above, to obtain the values of c and d, we need to compute the means and variances of $V_e(\hat{u}_k)$ and $L_a(u_k)$, and the correlation coefficient between

them for every received data block and every decoding iteration.[2] However, this method has two drawbacks. First, these computations increase the complexity of the decoder due the additional processing delay. Second, the extrinsic information $V_e(\hat{u}_k)$ may not be strictly Gaussian [21], especially when the data block is relatively short, e.g., $N = 512$. As a matter of fact, we examined the performance of the MSOVA with the parameters c and d computed for block length $N = 512$ and 1024 with the assumption that $V_e(\hat{u}_k)$ is Gaussian, but modest improvements were observed relative to the conventional SOVA.

As an alternative, we perform a computer search in an effort to find the pair (c, d) that would give the best performance. We list in Table 3.2 the values of c and d that we found through Monte Carlo simulations for various PCC and SCC codes, and for various code rates.

Table 3.2: Values of c and d for various PCC and SCC codes for the MSOVA algorithm on PR4 channels.

PCC Code $(23, 31)_{oct}$	(c, d)
$r = 4/5$	$(0.8, 0.8)$
$r = 16/17$	$(0.7, 0.9)$
$r = 64/65$	$(0.8, 0.8)$

SCC Code		(c, d)	
Outer	**Inner**	**Outer**	**Inner**
$(23, 31)_{oct}$, $r = 4/5$	$1 - D^2$, $r = 1$	$(0.8, 0.9)$	$(0.8, 0.9)$
$(23, 31)_{oct}$, $r = 16/17$	$1 - D^2$, $r = 1$	$(0.8, 0.9)$	$(0.7, 0.9)$
$(23, 31)_{oct}$, $r = 64/65$	$1 - D^2$, $r = 1$	$(0.7, 0.8)$	$(0.7, 0.8)$

[2]In the unlikely event that when the correlation coefficient is very small, i.e., $\rho \approx 0$, and the output of the SOVA follows a Gaussian distribution, i.e., $r_i = r_e = 1$ [11], we have $c = d = 1$. In such cases, the proposed modifications are no longer needed.

Since the values of c and d are dependent upon the correlation between the extrinsic and intrinsic information, they are affected by the code structure and the puncturing mechanism. Through simulations we found that once the code generator polynomials and puncturing mechanism are determined, these fixed pairs (c, d) can be used for any received data block and every decoding iteration. Moreover, these fixed pair of values work well for both channel models.

To illustrate the efficacy of the proposed MSOVA further, we perform another experiment in which we compare the actual extrinsic information supplied by the SOVA and MSOVA algorithms with that supplied by the APP (for the same noise and bit sequences.) In Fig. 3.4, we plot the extrinsic information of the SOVA and MSOVA against that of the APP (for the PCC and SCC cases.) Each part of the figure includes 2048 data points obtained after the first iteration at SNR = 6.5 dB (on idealized PR4 channel.)

We observe from the figure that the extrinsic information of the SOVA is too optimistic relative to that of the APP, whereas the extrinsic information of the MSOVA matches very well that of the APP. It is evident from these results how effective the proposed modifications are.

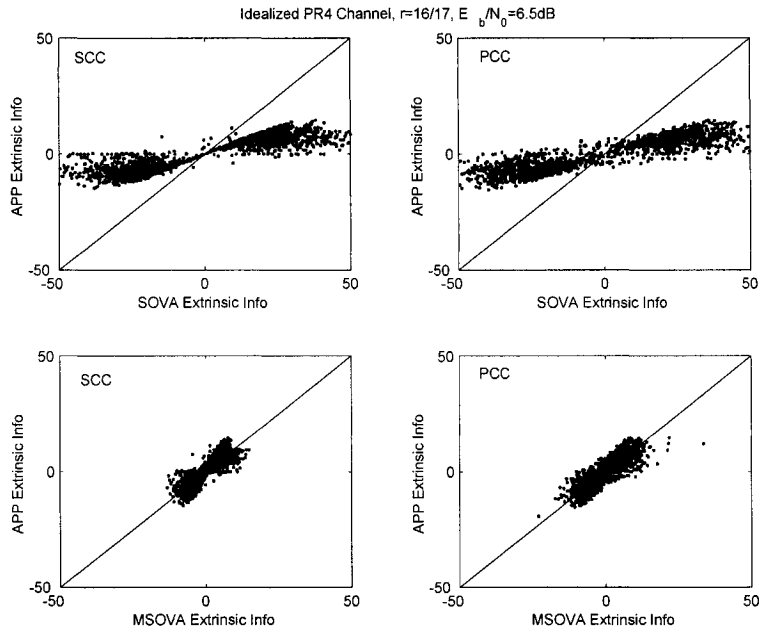


Figure 3.4: Comparison between the extrinsic information supplied by the SOVA and MSOVA, and that supplied by the APP on the idealized PR4 channel. Left plots for the SCC system; right plots for the PCC system. (All results for the same concatenation system were obtained using the same bit and noise sequences.)

3.4 Simulation Results

3.4.1 Idealized PR4 Channel

The simulation model is depicted in Fig 3.1(a-a). The PCC scheme uses two identical RSC encoders, where each RSC encoder employs the generator polynomials $(g_1, g_2) = (23, 31)_{oct}$ (16 states). As for the SCC scheme, it uses an outer RSC code whose encoder employs the generator polynomial $(g_1, g_2) = (23, 31)_{oct}$ (16 states), whereas the inner code is precoded idealized PR4 (or equalized-PR4) channel modeled by the polynomial $1 - D^2$. The values of the attenuators c and d used in the simulations are

shown in Table 3.2. These values are fixed for all range of SNR and all iterations.

For PCCs (SCCs), code rates of the form $k_0/(k_0+1)$ for $k_0 = 4, 16$, and 64, were achieved by saving the second bit in every $2k_0$-bit (k_0-bit) parity block of each RSC encoder output and puncturing the rest. In all cases, the interleaver employed are $S-$random interleavers.

In Fig. 3.5, we present the bit error performance (P_b) simulation results on the idealized PR4 channel for the PCC system for code rate 4/5 and 5 decoder iterations. The PCC decoder uses two iterative APPs, two conventional SOVAs, or two MSOVAs. We observe from the figure that the MSOVA achieves a performance improvement of about 1.2 dB relative to the SOVA, and is only about 0.4 dB away from the APP, all at $P_b = 10^{-5}$.

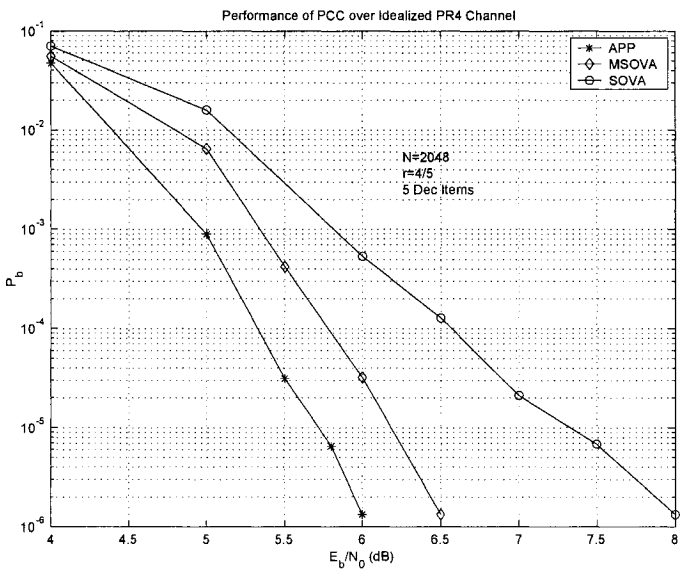


Figure 3.5: Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [PCC system with 16-state RSC encoders, rate 4/5, $N = 2048$, and 5 decoder iterations.]

In Fig. 3.6, we plot the P_b performance of the APP, SOVA and MSOVA algorithms on the idealized PR4 channel for the SCC system for code rate 4/5 and 5 decoder iterations. We observe from the figure that the MSOVA yields a performance improvement of about 1.2 dB at $P_b = 10^{-5}$ relative to the SOVA, whereas the performance gap between the MSOVA and APP is about 0.5 dB at $P_b = 10^{-5}$.

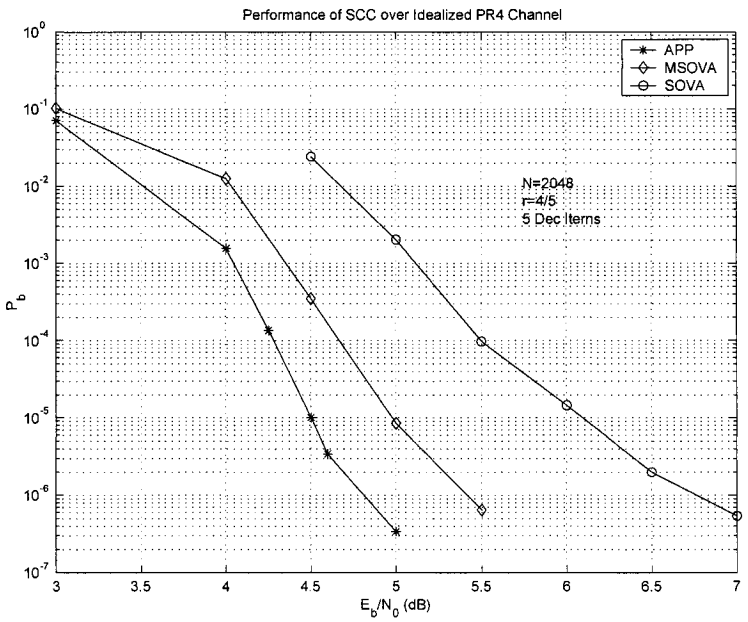


Figure 3.6: Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [SCC system with 16-state outer RSC encoder, rate−1 inner code, overall rate 4/5, $N = 2048$, and 5 decoder iterations.]

The same experiment mentioned above (for the SCC case) is repeated for rates 16/17 and 64/65 and the results are plotted in Figs. 3.7 and 3.8, respectively. As we observe from Fig. 3.7, the MSOVA improves the performance by about 1.6 dB at $P_b = 10^{-5}$ relative to the SOVA, and the performance gap between the MSOVA and APP is only 0.4 dB.

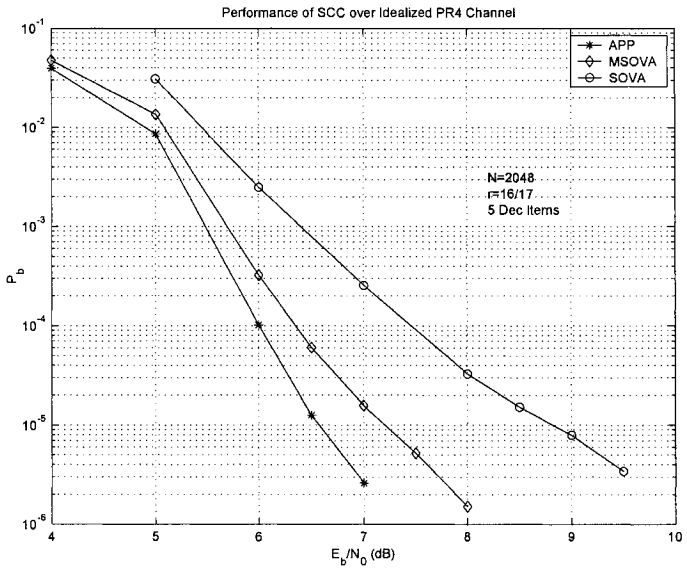


Figure 3.7: Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [SCC system with 16-state outer RSC encoder, rate−1 inner code, overall rate 16/17, $N = 2048$, and 5 decoder iterations.]

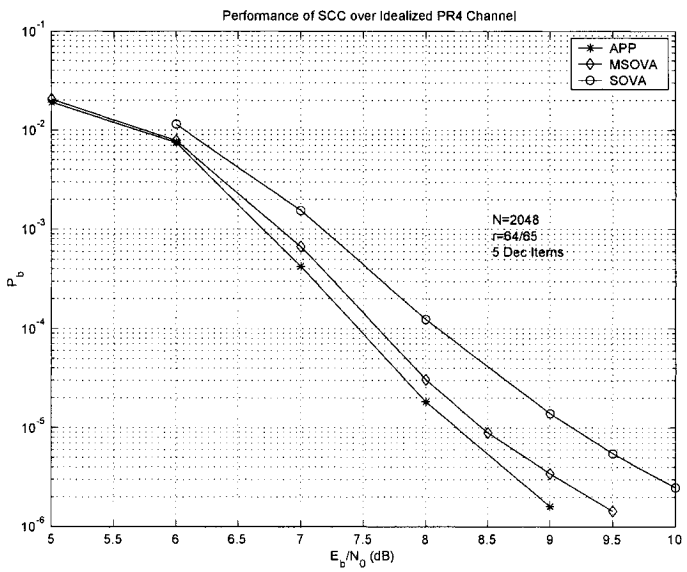


Figure 3.8: Bit error rate performance comparison between the SOVA, MSOVA and APP algorithm on the idealized PR4 Channel. [SCC system with 16-state outer RSC encoder, rate−1 inner code, overall rate 64/65, $N = 2048$, and 5 decoder iterations.]

In contrast, the performance improvement in Fig. 3.8 is about 0.7 dB at $P_b = 10^{-5}$ relative to the SOVA, whereas the MSOVA is only 0.2 dB away from the APP. It is clear from this figure that the performance improvement is modest compared to that for other code rates. This is attributed to the fact that, since the code rate is very high, the performance gap between the SOVA and APP is already small (around 0.9 dB). So there is not much room for improvements.

3.4.2 Equalized PR4 Channel

The simulation model is shown in Fig. 3.1(*b*-*b*), where the channel model is the Lorentzian model equalized to a PR4 target. The equalizer assumed is a finite impulse response (FIR) filter of length 51 whose tap weights are generated based on the minimum mean squared error (MMSE) criterion. In our simulations, a user density of $S_u = 2.0$ and 5 decoder iterations are used. Also, the PCC and SCC system parameters used here are the same as those used for the idealized PR4 channel.

The P_b simulation results on the equalized-PR4 channel for the PCC system for code rate 16/17 and 5 decoder iterations are plotted in Fig. 3.9.As observed from the figure, the MSOVA improves the performance, relative to the SOVA, by about 0.7 dB at $P_b = 10^{-5}$. Furthermore, the performance gap between the MSOVA and APP is only 0.3 dB at $P_b = 10^{-5}$.

Similar performance improvements for the SCC system were observed in Fig. 3.10.

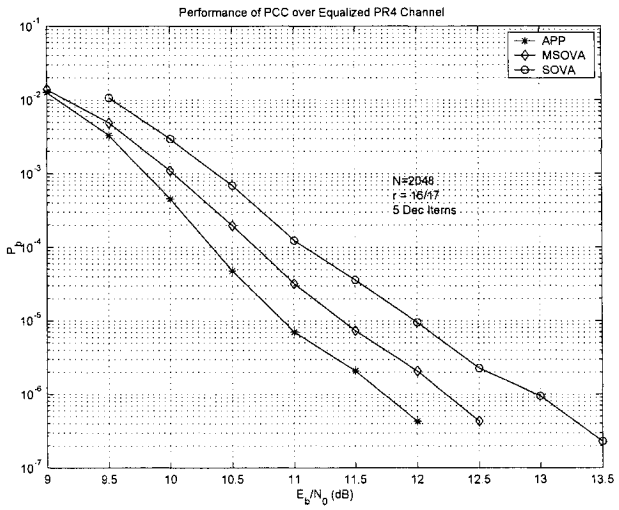


Figure 3.9: Bit error rate performance comparison between the SOVA, MSOVA, and APP decoders on the equalized PR4 channel. [PCC scheme with 16-state outer RSC encoder, code rate 16/17, $N = 2048$, $S_u = 2.0$, and 5 decoder iterations.]

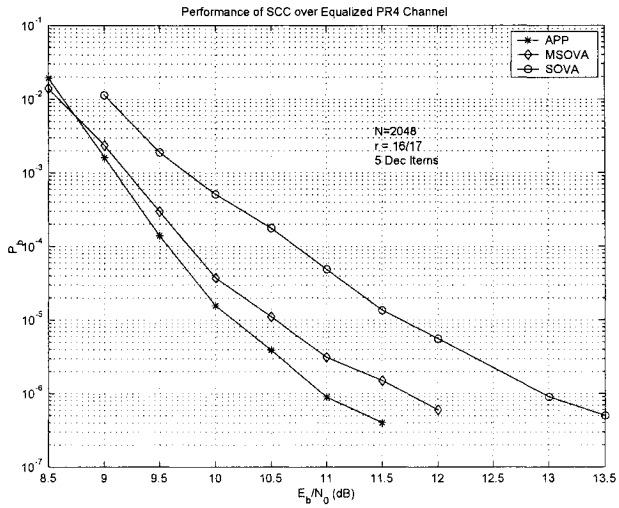


Figure 3.10: Bit error rate performance comparison between the SOVA, MSOVA, and APP decoders on the equalized PR4 channel. [SCC scheme with 16-state outer RSC encoder, rate-1 inner code, overall rate 16/17, $N = 2048$, $S_u = 2.0$, and 5 decoder iterations.]

As observed from Fig. 3.10, the MSOVA improves the performance, relative to the SOVA, by about 1.2 dB at $P_b = 10^{-5}$. Furthermore, the performance gap between the MSOVA and APP is only 0.3 dB at $P_b = 10^{-5}$.

In Fig. 3.11, we compare the actual extrinsic information supplied by the SOVA and MSOVA algorithms with that supplied by the APP (for the same noise and bit sequences.) We consider both the PCC and SCC systems. Each part of the figure includes 2048 data points obtained after the first iteration at SNR = 10 dB (on the equalized PR4 channel.) It is clear from the figure how the extrinsic information of the SOVA is too optimistic relative to that of the APP, whereas the extrinsic information of the MSOVA somewhat matches very well that of the APP.

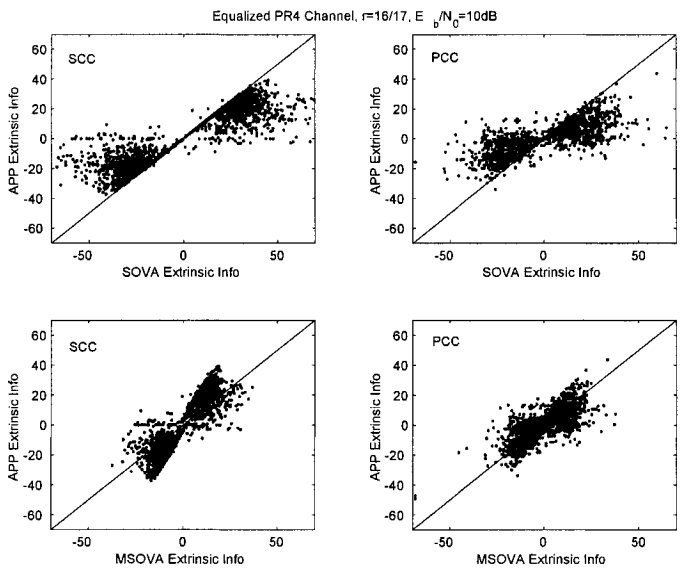


Figure 3.11: Comparison between the extrinsic information supplied by the SOVA and MSOVA, and that supplied by the APP on the equalized PR4 channel. Left plots for the SCC system; right plots for the PCC system. (All results for the same concatenation system were obtained using the same bit and noise sequences.)

3.5 Concluding Remarks

In this chapter we have introduced simple modifications to the conventional SOVA in an effort to attenuate the optimistic extrinsic information at its output. We showed that the reason behind producing these exaggerated values is the strong correlation between the input and output of the SOVA. The proposed modifications were aimed at reducing this inherent correlation, which would ultimately lead to producing more realistic extrinsic information. We have examined the performance of the MSOVA on idealized PR channels and the Lorentzian model equalized to a PR target with favorable results. We finally remark that the introduced modifications result in adding only two multipliers to the complexity of the conventional SOVA.

Chapter 4

Conclusions and Future Work

4.1 Conclusions

In this thesis, we proposed a simple modification to the conventional SOVA in an effort to overcome the optimistic extrinsic information at its output. We explained that the reason behind producing these exaggerated values is the strong correlation between the input and output of the SOVA. The proposed modifications were aimed at reducing this inherent correlation, which would ultimately lead to producing more realistic extrinsic information. Based on mathematical analysis, the MSOVA involves using two attenuators, one applied to the immediate output of the SOVA and another applied to the extrinsic information before it is passed on to the other decoder component. For PCC and SCC schemes, we examined the performance of the MSOVA on various communication channels and storage medium with favorable results. For example, it provides improvements of about 0.8 to 1.0 dB at $P_b = 10^{-5}$ on AWGN

channels, about 1.4 to 2.0 dB at $P_b = 10^{-5}$ on flat fading channels, and up to 1.6 dB at $P_b = 10^{-5}$ on storage channels. We also showed that there are cases where the MSOVA is superior to the *a posteriori* probability (APP) algorithm. With this motivation, we extended the proposed modification to the APP algorithm on AWGN and fading channels with favorable results. We demonstrated that the modified APP (MAPP) provides performance improvements between 0.3 to 0.6 dB at $P_b = 10^{-5}$ relative to the APP. We lastly mention that the proposed modifications, while they provide considerable performance improvements, keep the complexity of these decoders almost the same, which is remarkable.

4.2 Future Work

For future work, we may apply the MSOVA to MIMO systems and examine the performance over other practical channel models such as frequency-selective fading channels for W-CDMA systems. These simulations will help us to confirm that the MSOVA can be extended to any general case to which the conventional SOVA can be applied. It will be interesting to know how much improvement that the MSOVA can achieve relative to the conventional SOVA in the practical environment.

Furthermore, we can do more study on the coefficients c and d. We have shown in this thesis that the straightforward method to compute the values of c and d is to compute the means and variances of $V_e(\hat{u}_k)$ and $L_i(u_k)$, and the correlation coefficient between them for every received data block and every decoding iteration. However,

this method is not practical. Instead, we performed a computer search in an effort to find the pair (c, d) that would give the best performance. Although the number of district pairs of c and d may seem very large to search over, the range of values for these attenuators is quite limited. For example, consider Table 3.2 which gives these values for different code rates. As we can see from the table, all of the values are between 0.7 to 0.9, which limits the number of possibilities to be searched over. Nevertheless, it may be difficult to find an explicit relationship between these values and the underlying code, or code rate. It would be, however, of great interest if such a relationship can be defined explicitly.

Bibliography

[1] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, 27, Part I: 379-423, July 1948. Part II: 623-656, Oct. 1948.

[2] G. D. Forney, Jr., "Concatenated Codes," Cambridge, MA: MIT Press, 1966

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," *Proc. IEEE International Conference on Communications (ICC)*, pp. 1064-1070, 1993.

[4] S. Benedetto S, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 909-926, March 1998.

[5] H. El Gamal and A. R. Hammons, "Analyzing the turbo decoder using the Gaussian approximation," *IEEE J. Select. Areas Commun.* , vol. 47, pp. 671–686, Feb. 2001.

[6] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Info. Theory*, vol. 42, no. 2, pp. 429-445,

March 1996.

[7] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," *Proc. IEEE International Conference on Communications (ICC),* pp. 1009-1013, 1995.

[8] J. Hagenauer, P. Robertson, and L. Papke, "Iterative (TURBO) decoding of systematic convolutional codes with the MAP and SOVA algorithms," *Proc. ITG-Fachtagung 'Codierung'*, Munich, Germany, pp. 21-29, Oct. 1994.

[9] A. Ghrayeb and W. E. Ryan, "Performance of high rate turbo codes employing the soft-output Viterbi algorithm," *Proc. The 33rd Asilomar Conference on Signals, Systems, and Computer*, Pacific Grove, CA, pp. 1665-1669, Oct. 1999.

[10] L. Lin and R. S. Cheng, "Improvements in SOVA-based decoding for turbo codes," *Proc. IEEE International Conference on Communications (ICC)*, pp. 1473-1478, Montreal, Canada, June 1997.

[11] L. Papke, P. Robertson, and E. Villerbrun, "Improved decoding with the SOVA in a parallel concatenated (turbo-code) scheme," *Proc. Proc. IEEE International Conference on Communications (ICC)*, pp. 102-106, June 1996.

[12] G. Colavolpe, G. Ferrari, and R. Raheli, "extrinsic information in iterative decoding: a unified view," *IEEE Trans. Commun.*, vol. 49, pp.2088–2094, Dec. 2001.

[13] D. W. Kim, T. W. Kwon, J. R. Choi, and J. J. Kong, "A modified two-step SOVA-based turbo decoder with a fixed scaling factor," *Proc. IEEE ISCAS*, vol. 4, pp. 37–40, May 2000.

[14] S. Papaharalabos, P. Sweeney, and B. G. Evans, "Improvements in SOVA-based decoding for turbo codes," *IEE Electronics Letters*, vol. 39, no. 19, pp.1391-1392, Sept. 2003.

[15] Z. Wang and K. Parhi, "High performance, high throughput turbo/SOVA decoder design," *IEEE Trans. Commun.*, vol. 51, no.4, pp. 570- 579, April 2003.

[16] S. M. Alamouti, "A simple transmitter scheme for wireless communications," *IEEE J. Select. Areas. Commun.*, vol. 16, no. 8, pp. 1451-1458, Oct, 1998.

[17] V. Tarokh, H. Jafarkhani, and A.Calderbank, "Space–time block codes from orthogonal designs," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1456-1467, July 1999.

[18] G. D. Forney, "The Viterbi Algorithm," *Proc. IEEE*, vol. 61, pp. 268-278, March 1973.

[19] J. Hagenauer, "Source-controlled channel decoding," *IEEE Trans. Commun.*, vol. 43, no. 9, pp. 2449-2457, Sept. 1995.

[20] J. Hagenauer, and P. Hoeher, "A Viterbi Algorithm with soft-decision outputs and its applications," *Proc. IEEE Global Telecommunications Conference (Globecom),* pp. 1680-1686, 1989.

[21] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261-1271, Oct. 1996.

[22] W. E. Ryan, "Concatenated codes and iterative decoding," in *Wiley Encyclopedia of Telecommunications*, New York: Wiley Sons, 2003.

[23] J. G. Proakis, *Digital Communications*, 4th Edition, McGraw-Hill, 2001

[24] W. E. Ryan, "Performance of high rate turbo codes on a PR4-equalized magnetic recording channel," *Proc. IEEE International Conference on Communications (ICC)*, vol. 2, pp. 947-951, June 1998.

[25] W. E. Ryan, L. McPheters, and S. McLauphlin, "Combined turbo coding and turbo decoding for PR4-equalized Lorentzian channels," *Proc. of Conf. on Information Sciences and Systems*, pp. 489-493, March 1998.

[26] L. McPheters, S. McLaughlin and E. Hirsh, "Turbo Codes for PR4 and EPR4 Magnetic Recording," *Proc. the 32nd Asilomar Conference on Signals, Systems, and Computers*, vol. 2, pp. 1778-1782, Nov. 1999.

[27] L. McPheters, S. McLaughlin, and K. Narayanan, "Precoded PRML, serial concatenation, and iterative (turbo) decoding for digital magnetic recording," *IEEE Trans. Magn.*, vol. 35, pp. 2325-2327, Sept. 1999.

[28] T. Souvignier, A. Friedmann, M. Oberg, P. H. Siegel, R. E. Swanson, and J.K. Wolf, "Turbo Decoding for PR4: Parallel Versus Serial Concatenation", *Proc. 1999 Int. Conf. on Comm.*, vol. 3, pp. 1638-1642, June 1999.

[29] A. Ghrayeb and W. E. Ryan, "Concatenated system design for storage systems," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 4, pp. 709-718, April 2001.

[30] W. E. Ryan, "Optimal code rates for concatenated codes on a PR4-equalized magnetic recording channel," *IEEE Trans. Magnetics*, vol. 36, no. 6, pp. 4044-4049, Nov. 2000.

[31] J. Bergmans, "Discrete-time models for digital magnetic recording channel," *Philips J. Res.*, pp. 531-559, 1986.

[32] A. Ghrayeb, "On the SOVA for extremely high code rates for partial response channels," *Journal of Communications and Networks (JCN)*, vol. 5, no. 1, pp. 1-6, March 2003.