Concatenated Space-Time Block Codes and Turbo Codes with Unstructured Interference

by

William Chow

B.Eng, University of Victoria, 2002

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

© William Chow, 2004

University of Victoria

All rights reserved. This thesis may not be reproduced in whole or in part by photocopy or other means, without the permission of the author.
ABSTRACT

The performance of space-time block codes in providing transmit diversity is severely degraded when strong localized interference is present. This problem is addressed by investigating a recently proposed coherent space-time block code decoding algorithm for unknown interference suppression. The algorithm assumes a Gaussian noise and interference approximation and is based on a cyclic-based maximum-likelihood estimation technique (CML). In this thesis, simulations are done applying CML in a coherent system with unstructured interference to validate previous work. An extension of these results is obtained by examining factors that affect CML performance and modifying CML for use in a non-coherent system. To improve bit error rate performance, a turbo code for channel coding was added to both systems. This addition required the development of reliability metrics for soft-information transfer between the space-time block code detector and the turbo code decoder. Significant coding gains exceeding 8dB at a bit error rate of $10^{-3}$ are achieved for the turbo-coded system when compared to that of an uncoded system.
# Table of Contents

Abstract ii

Table of Contents iv

List of Tables vii

List of Figures viii

List of Abbreviations x

Acknowledgement xi

1 Introduction 1

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

1.2 Brief Background in Traditional Noise Cancellation Techniques ........ 3

1.3 Robust Space-Time Receiver Designs for Unknown Interference ........ 4

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

1.5 Outline of Thesis .......................................... 6

2 Background 7

2.1 MIMO Wireless System Architecture ............................... 7

2.2 Orthogonal Space-Time Block Coding ............................. 13

2.2.1 OSTBC Encoder ........................................... 13

2.2.2 OSTBC Detection ........................................ 15

2.3 Differential Space-Time Modulation ................................ 16

2.3.1 DSTM Encoder ........................................... 16
# Table of Contents

2.3.2 DSTM Detection ........................................ 17

2.4 Cyclic-Based Maximum-Likelihood Estimation .................. 19
   2.4.1 Exact ML Approach with Unknown Channel and Noise .... 19
   2.4.2 Initial Training-Based ML Approach .................... 20
   2.4.3 Cyclic-Based ML Approach ............................... 21
   2.4.4 Modifying CML for DSTM ................................. 22

2.5 STBC Soft-Information Transfer ............................... 23
   2.5.1 Soft Output OSTBC Detection ............................ 23
   2.5.2 Soft Output DSTM Detection .............................. 24

2.6 Turbo Coding .............................................. 25
   2.6.1 Turbo Code Encoder ...................................... 26
      2.6.1.1 State and Trellis Representation of RSC Encoders 29
      2.6.1.2 Trellis Termination ................................. 30
   2.6.2 Turbo Code Decoder ...................................... 31
      2.6.2.1 The SOVA Algorithm for Iterative Decoding .... 33

3 Performance of CML in OSTBC and DSTM ......................... 39
   3.1 Validating CML Performance in OSTBC and DSTM ............. 40
   3.2 Impact of the Interference Power ........................... 41
   3.3 Effect of the Number of Training Matrices .................. 42
   3.4 Effect of Doppler Fading ................................... 42
   3.5 Applying Different Interference Models ..................... 43
   3.6 Summary .................................................. 44

4 Performance of CML and Turbo-Coded MIMO Wireless System .... 55
   4.1 Coding Gain with Turbo Codes in an OSTBC System .......... 56
   4.2 Coding Gain with Turbo Codes in a DSTM System ........... 56
   4.3 Effect of Constraint Length on Turbo-coded Coherent System Performance 57
   4.4 Effect of Puncturing on Turbo-coded Coherent System Performance ... 57
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.5 Comparing Turbo-Coded Coherent and Noncoherent System</td>
<td>58</td>
</tr>
<tr>
<td>5 Conclusion and Future Work</td>
<td>64</td>
</tr>
<tr>
<td>References</td>
<td>65</td>
</tr>
</tbody>
</table>
List of Tables

Table 2.1 Decoder complexity comparison [37] .......................... 32
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.1</td>
<td>The MIMO wireless system architecture</td>
<td>8</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Block diagram of a 4-state turbo code encoder</td>
<td>26</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>State diagram for (5,7) RSC encoder [36]</td>
<td>29</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>Trellis diagram for (5,7) RSC encoder [36]</td>
<td>30</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Block diagram of generic turbo code decoder [37]</td>
<td>31</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Performance of CML in an OSTBC system with no interference</td>
<td>45</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Performance of CML in an OSTBC system with one strong interferer.</td>
<td>45</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>Performance of CML in a DSTM system with one strong interferer.</td>
<td>46</td>
</tr>
<tr>
<td>Figure 3.4</td>
<td>Performance of CML in an OSTBC system with $p_e = 10dB.$</td>
<td>46</td>
</tr>
<tr>
<td>Figure 3.5</td>
<td>Performance of CML in a DSTM system with $p_e = 10dB.$</td>
<td>47</td>
</tr>
<tr>
<td>Figure 3.6</td>
<td>Performance of CML in an OSTBC system with $p_e = 20dB.$</td>
<td>47</td>
</tr>
<tr>
<td>Figure 3.7</td>
<td>Performance of CML in a DSTM system with $p_e = 20dB.$</td>
<td>48</td>
</tr>
<tr>
<td>Figure 3.8</td>
<td>Effect of varying $p_e$ on CML performance in an OSTBC system.</td>
<td>48</td>
</tr>
<tr>
<td>Figure 3.9</td>
<td>Effect of varying $p_e$ on CML performance in a DSTM system.</td>
<td>49</td>
</tr>
<tr>
<td>Figure 3.10</td>
<td>Effect of the number of training matrices on CML performance in an OSTBC system</td>
<td>50</td>
</tr>
<tr>
<td>Figure 3.11</td>
<td>Effect of the number of training matrices on CML performance in a DSTM system</td>
<td>51</td>
</tr>
<tr>
<td>Figure 3.12</td>
<td>Effect of Doppler fading on CML performance in an OSTBC system.</td>
<td>52</td>
</tr>
<tr>
<td>Figure 3.13</td>
<td>Effect of Doppler fading on CML performance in a DSTM system.</td>
<td>52</td>
</tr>
<tr>
<td>Figure 3.14</td>
<td>Effect of co-channel interference on CML performance in an OSTBC system.</td>
<td>53</td>
</tr>
</tbody>
</table>
Figure 3.15 Effect of periodic interference on CML performance in an OSTBC system. ............................. 53
Figure 3.16 CML performance in an OSTBC system with a mixture of correlated Gaussian and sinusoidal interference. ..................... 54
Figure 4.1 Performance of a concatenated 4-state turbo-coded and OSTBC systems with one strong interferer. .................................. 59
Figure 4.2 Performance of a concatenated 4-state turbo-coded and DSTM systems with one strong interferer. ................................. 60
Figure 4.3 Effect of constraint length on turbo-coded and OSTBC system performance with one strong interferer. ................................. 61
Figure 4.4 Performance comparison between a 4-state punctured and unpunctured turbo-coded systems with one strong interferer. .................. 62
Figure 4.5 Performance comparison between turbo-coded coherent and noncoherent systems with one strong interferer. ............................ 63
### List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANC</td>
<td>Active Noise Cancellation</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>BPSK</td>
<td>Binary Phase Shift Keying</td>
</tr>
<tr>
<td>CML</td>
<td>Cyclic Maximum Likelihood</td>
</tr>
<tr>
<td>DCML</td>
<td>Differential CML</td>
</tr>
<tr>
<td>DPSK</td>
<td>Differential Phase Shift Keying</td>
</tr>
<tr>
<td>DSCM</td>
<td>Differential Space-Code Modulation</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Array</td>
</tr>
<tr>
<td>ISI</td>
<td>Inter-Symbol Interference</td>
</tr>
<tr>
<td>LLR</td>
<td>Log-Likelihood Ratio</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum A Posterior</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple-Input Multiple-Output</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum-Likelihood</td>
</tr>
<tr>
<td>RSC</td>
<td>Recursive Systematic Convolutional</td>
</tr>
<tr>
<td>SIR</td>
<td>Signal to Interference Ratio</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>STBC</td>
<td>Space-Time Block Code</td>
</tr>
<tr>
<td>STTC</td>
<td>Space-Time Trellis Code</td>
</tr>
<tr>
<td>SOVA</td>
<td>Soft Output Viterbi Algorithm</td>
</tr>
<tr>
<td>TC</td>
<td>Turbo Code</td>
</tr>
<tr>
<td>T-R</td>
<td>Transmit-Receive</td>
</tr>
<tr>
<td>VA</td>
<td>Viterbi Algorithm</td>
</tr>
<tr>
<td>WML</td>
<td>White-based Maximum-Likelihood</td>
</tr>
</tbody>
</table>
Acknowledgement

I want to give a special thank you to my family and friends for their encouragement along the way. I am grateful for all the enduring support and care given to me throughout my life. To Dr. T. Aaron Gulliver, I am thankful for the opportunity to have worked on this thesis. Your support, guidance, and patience as my supervisor has made my stay at the University of Victoria all the more enjoyable. To Dr. Robert Schober, I wish to say thanks for your assistance and technical advice. Lastly, I wish to acknowledge NSERC and Sierra Wireless, Inc., for without their financial sponsorship this thesis would not be possible. In particular, much thanks is given to Mr. Peter McConnell, the representative for Sierra Wireless, Inc. who motivated this work.
Chapter 1

Introduction

1.1 Motivation

In the near future, we can expect widespread deployment of wireless communication systems incorporating multi-transmit and multi-receive antenna arrays (MIMO systems) to enable high data rate wireless applications \([1][13]\). This high data rate will be possible with the development of advanced modulation and coding techniques. Space-time coding is one such modulation technique that has received much attention recently due to its ability to exploit both spatial and temporal diversity.

Diversity is used in a wireless communication system to provide the receiver with multiple independent copies of the source data. This improves the system performance of a wireless link because the probability of all signals being faded is far less than the probability that a single signal is faded. In wireless systems, spatial diversity is obtained using multiple antennas spaced sufficiently apart with appropriate signal processing. More specifically, the terms transmit and receive spatial diversity are used respectively, to indicate the use of multiple antennas located at the transmitter or receiver. To achieve independent copies, the signals must be received from antennas spatially separated greater than or equal to one half of the carrier wavelength \([26]\). Temporal diversity, or time diversity, is obtained by repeatedly transmitting multiple copies of the same data at multiple time spacings. Independent copies will be received and exploited provided the bandwidth of the signal is less than the bandwidth of the channel.
1.1 Motivation

At present, the two most common choices of codes for space-time coding are space-time trellis codes (STTC) and space-time block codes (STBC). STTC is designed using the combination of trellis coded-modulation and transmit diversity principles. This provides both coding and transmit diversity gains. On the other hand, STBC exploits transmit diversity only.

Coding gain can be defined as the power reduction achievable to obtain an equivalent target bit error rate (BER) in a system using channel coding compared to a system without channel coding. BER, in turn, is used as a metric in determining system performance. Channel coding provides this gain by adding redundant (parity) information into the data stream. This implies a cost of higher system bandwidth due to the increased amount of data that needs to be transferred.

STTC is advantageous over STBC for coding gain is provided as well as transmit diversity gain. However, this performance gain comes at the cost of increased decoding complexity. STBC is attractive due to its ability to provide full transmit diversity in a MIMO system with a simple receiver design. Its drawback is that no coding gain is offered. However, this can be remedied by adding an outer channel code to the system. Past research by T. H. Liew et al. [19] has compared the performance of STTC to that of STBC combined with an outer channel code. Among the two systems of similar complexity, STBC combined with an outer channel code offered superior performance.

Turbo coding, a relatively new and powerful channel coding scheme for forward error correction (FEC), was recently introduced by Berrou et. al. [3]. In [3], results were documented that showed turbo coding performance within 0.7dB of channel capacity, provided long codes are used. Channel capacity is the fundamental limit of transmission rate achievable in a digital communications system based on system bandwidth and signal-to-noise ratio (SNR). Current and future wireless systems will incorporate turbo coding [1].

The initial research on space-time coding focused on coherent modulation. In coherent space-time coding techniques, the decoder assumes or acquires knowledge of the channel under consideration. However, there are times when channel estimates cannot be reliably
obtained or are not desired. This could be due to factors such as fast fading or implementation cost. Hence, a noncoherent form of STBC would be favorable.

Pioneering work in the development of noncoherent STBC was performed separately by Hochwald et al. [9] and Hughes [11]. In their work, the concept of differential phase-shift keying (DPSK) was extended and generalized to a multi-dimensional space suitable for transmission over multi-transmit antenna arrays. The end result was the development of a class of noncoherent STBC referred to as differential space-time modulation (DSTM). DSTM performs similarly to coherent STBC, offering full transmit diversity with no coding gain. In contrast, however, a theoretical 3dB penalty in SNR performance is incurred due to DSTM decoding with no channel estimates.

In conjunction with the development of advanced modulation and coding techniques, wireless products are becoming smaller and faster to meet portability and usage requirements. This makes noise and interference a major concern. Industry measurements have shown that the surrounding electronics contribute greatly to the ambient noise power, and can often lead to an unacceptable noise floor in the receiver [5][18].

Localized noise and interference is a significant problem in MIMO wireless systems because it spatially correlates the received data resulting in a loss in receiver performance. The development of robust space-time receivers is highly desirable to mitigate this loss.

1.2 Brief Background in Traditional Noise Cancellation Techniques

The classical approach to combat localized noise in a single antenna wireless system is to employ active noise cancellation (ANC). ANC techniques are based on principles of optimal filtering [40]. Its strength is that no a priori knowledge of the signal or noise statistics is required. The cost is the requirement of a reference input. In a wireless system, this reference input is another antenna.
Attempts to eliminate the cost of this reference source is highly desirable and not new, as the development of blind ANC techniques is an active area of research. Proposed solutions include using blind source separation or signal subspace estimation for ANC [8], [6], [22]. The drawback with these solutions, however, is that they are generally computationally expensive or require expert knowledge in complex higher-order statistics or neural network theory. In wireless systems and applications, computationally and power efficient solutions are desired. Further, the use of space-time coding would be preferred to exploit multiple antennas that will be a part of the new and future wireless devices under development. Thus, a combined noise cancellation scheme with space-time coding would be ideal for improving wireless system performance.

1.3 Robust Space-Time Receiver Designs for Unknown Interference

Although much work has been done on interference cancellation in space-time receivers, the main focus has been on multi-user interference cancellation schemes. These schemes assume and exploit prior knowledge of statistics of other users [27][29][30]. There are times when this assumption may not be valid, however, in which case the interference should be assumed unknown. An example is when multiple wireless devices operate simultaneously using different standards. In this scenario, each wireless device would be contributing unknown interference with respect to one another.

Approaches specific to a single-user MIMO wireless system have been recently proposed within the last few years. Liu et al. [17] has proposed differential space-code modulation (DSCM), which combines direct-sequence spread spectrum modulation with that of the design of STBC. Although remarkable performance with DSCM was shown in [17], the drawback with this approach is its computational complexity and difficulty of integration with current systems. In Larsson et. al. [20], STBC detectors for unknown interference
suppression are analyzed and developed based on a deterministic channel model as opposed to the stochastic model commonly employed. This approach proved favorable as improved BER performance under different interference conditions was achieved for the deterministic-based STBC detectors over the conventional stochastic-based STBC detectors. Larsson et. al. expanded the results established in [20]. This extension resulted in [21], where the theory developed in [20] was applied to develop low complexity iterative STBC detectors for interference suppression based on a Gaussian noise and interference approximation. In [21], the STBC detector acquires and refines channel and noise covariance matrices based on cyclic maximization theory and a minimal amount of training data. This scheme shall be referred to as the “cyclic-based maximum likelihood technique” (CML). The proposed STBC detectors are optimal in a maximum-likelihood sense. Further investigation in [21] revealed a favorable tradeoff between design complexity and BER performance.

1.4 Thesis Contributions

This thesis reviews and extends results obtained in a recently proposed space-time receiver using CML for interference suppression. The major contributions of this thesis are two-fold: 1) The application of CML to coherent and noncoherent STBC coded systems and evaluation of the effects of CML parameters on system performance, and 2) The development of STBC soft-information transfer in the STBC detector and evaluation of the performance of CML-based space-time receivers concatenated with turbo codes. To this end, system simulations were developed using Matlab and performance results were analyzed.

The results demonstrate the robustness of the proposed STBC receivers in suppressing unknown interference under different interference conditions. This is in contrast to the conventional receivers designed under the additive white Gaussian noise assumption. Remarkable performance gains are achieved by concatenating turbo codes with the existing STBC system. Coding gains exceeding 8dB were achieved at a target bit error rate of $10^{-3}$.
and $10^{-2}$ for the coherent and noncoherent STBC system, respectively.

1.5 Outline of Thesis

The remainder of this thesis is organized as follows. Chapter 2 provides a description of the system components and the necessary theory for performance analysis. Chapter 3 presents results pertaining to the performance of CML applied to a two transmit and two receive multiple antenna wireless system with coherent and noncoherent STBC coding. Chapter 4 demonstrates the coding gains achievable by concatenating the proposed space-time receivers with turbo codes. Chapter 5 provides conclusions and suggestions for future work.
Chapter 2

Background

This chapter provides the requisite background on the MIMO wireless system under consideration. First, the overall system architecture describing the transfer of data from end-to-end is presented. This is followed with a detailed description of the subsystem components.

2.1 MIMO Wireless System Architecture

The MIMO wireless system is depicted in Figure 2.1. For simplicity, but noting that the system can be generalized to higher dimensions, we examine a wireless system with $M = 2$ transmit and $N = 2$ receive antennas (2x2 MIMO system), with signals belonging to a finite constellation of $|C| = 2$ elements.

As shown in Figure 2.1, the source signal is represented by data bits $(b_1 \ldots b_t)$ transmitted in each time interval $t$ with each bit $b_t \in \{0, 1\}$. This binary data is first transformed in one of two ways. If turbo coding is required, redundant data is first added by the turbo code encoder and then mapped into bipolar binary phase-shift keyed (BPSK) symbols $(c_1, \ldots, c_t)$. with each symbol $c_t \in \{-1, 1\}$. The data to symbol mapping equation is given by $c_t = 2 \cdot b_t - 1$. However, if turbo coding is not required, then the data bits are directly mapped into BPSK encoded symbols. These BPSK symbols are then fed into a STBC encoder to enable transmission of data over a 2x2 MIMO system.

The received signal is a noisy superposition of independently faded signals. In the discrete baseband domain with symbol-spaced sampling, the received signal at time $t$ and
antenna $n$, $n \in 1, 2$, is

$$r_{t,n} = \sum_{m=1}^{M} h_{m,n} x_{t,m} + n_{t,n} + e_{t,n}$$

$$= h_{1,n} x_{t,1} + h_{2,n} x_{t,2} + n_{t,n} + e_{t,n},$$

where $x_{t,m}$ denotes the STBC encoded symbol transmitted from antenna $m$ at time $t$ and $h_{m,n}$ describes the channel between transmit antenna $m$ with receive antenna $n$. $e_{t,n}$ and $n_{t,n}$ are the localized interference and the additive white Gaussian noise (AWGN), respectively, at receive antenna $n$ in time $t$. $n_{t,n}$ is assumed temporally and spatially white and is modeled as a complex Gaussian random variable with zero mean and variance $\sigma^2$ denoted by $CN(0, \sigma^2)$. The spatial whiteness of $n_{t,n}$ is with respect to the remaining $N - 1$ antennas.
For a 2x2 system with transmit signal power $\sqrt{\rho_2}$, (2.1) can be written as

$$ r_{t,1} = \sqrt{\rho_2} h_{1,1} x_{t,1} + \sqrt{\rho_2} h_{2,1} x_{t,2} + n_{t,1} + e_{t,1} $$

$$ r_{t,2} = \sqrt{\rho_2} h_{1,2} x_{t,1} + \sqrt{\rho_2} h_{2,2} x_{t,2} + n_{t,2} + e_{t,2} $$

(2.2)

for the signals at receiver antenna 1 and antenna 2, respectively. Here the transmit signal power is normalized by the number of transmit antennas ($M = 2$) in the system. Therefore the received energy per symbol period at each receive antenna is $\rho$.

In digital wireless communications, data received through a single-antenna wireless channel can be modeled by multiplicative fading. Depending on the bandwidth of the channel and the data rate of the system, fading will either be flat or frequency-selective.

In flat-fading channels, the received signal strength changes with time due to fluctuations in channel gain caused by multipath [26]. Assuming no line-of-sight exists between each transmit-receive (T-R) antenna pair, the instantaneous gain of a flat-fading channel can be modeled based on a Rayleigh distribution.

For frequency-selective channels, the received signal strength includes multiple versions of the transmitted signals with each copy attenuated and delayed in time. This in turn introduces intersymbol interference (ISI) into the wireless system. This ISI is usually mitigated with a wireless channel equalizer.

With the proposed interference suppression scheme, the channel is assumed to be flat-fading. The wireless channels connecting each T-R antenna pair is modeled in one of two ways.

The first model assumes a quasi-static Rayleigh flat fading channel. The term “quasi-static” implies that the wireless channel coefficients for each T-R antenna pair are constant during the time interval required to transmit a frame of STBC encoded data, but vary independently from frame to frame. This model is used in many wireless systems (e.g. indoor channels) where the coherence time of the channel is much longer than one symbol interval. The coherence time of the channel is the time it takes for the channel’s impulse response to change significantly. The channel coefficients consist of spatially and temporally white
entries \( h_{m,n} \) that are \( CN(0,1) \).

The second model also assumes a Rayleigh flat fading channel, however, the quasi-static assumption is dropped and Doppler fading is incorporated according to Clarke’s fading model [14]. With Doppler fading, the channel coefficients, \( h_{m,n} \), now contain spatially white but temporally colored coefficients. The channel coefficients change continuously in time with a temporal autocorrelation function given by

\[
R_{h_{m,n}}(\tau) = J_0(\omega_m \tau),
\]  
where \( \omega_m \) is the maximum doppler radio frequency shift and \( J_0 \) is the zero-order Bessel function of the first kind. \( \tau \) denotes the relative time shift between sets of samples used in the function time autocorrelation function.

For channel and noise estimation, we reformulate (2.2) into matrix form such that the received signal is represented by a received matrix \( R[l] \) of dimension \( M \times T \). \( M \) and \( T \) denote the number of transmit antennas and the time length of the STBC code, respectively. \( R[l] \) contain entries \( r_{t,n} \) that represent groups of \( T \) STBC encoded data that were received at antenna \( n \) in each time interval \( t \). Similarly, the transmit signal is now defined by a STBC codeword matrix \( X[l] \) with entries \( x_{t,m} \) that represent a group of \( T \) STBC encoded data symbols transmitted from antenna \( m \) at time \( t \).

Consider the transmission of a frame of \((K+L)\) concatenated STBC codeword matrices \( X = [X^{tr}[1] \ldots X^{tr}[K], X[1] \ldots X[L]] \) of dimension \( M \times T(K+L) \). \( X \) is composed of \( K \) training matrices \( X^{tr}[k] \) and \( L \) information matrices \( X[l] \). The received frame of matrices \( R = [R^{tr}[1] \ldots R^{tr}[K], R[1] \ldots R[L]] \) can be written as

\[
R = \sqrt{\frac{\rho}{M}} HX + N + E.
\]  
\( N \), and \( E \) represent the receiver noise matrix and the interference matrix, respectively, and contain entries \( n_{t,n} \) and \( e_{t,n} \) with properties as previously described. Here the channel \( H \) is assumed to undergo quasi-static fading and is therefore constant over the interval required to transmit \( X \).
The proposed CML-based space-time receivers assume unstructured interference. Therefore the localized interference $E$ can be modeled in a number of different ways. $E$ is modeled as either spatially correlated Gaussian interference, periodic sinusoidal interference, or synchronized co-channel interference made up of $P$ interferers.

$$E = \sqrt{\frac{\rho_e}{M}} H^e W^e,$$

(2.5) if the interference is modeled as spatially correlated Gaussian interference. $\rho_e$ denotes the interference power, $H^e$ signifies the wireless channel matrix linking the interference to each receive antenna, and $W^e$ is a Gaussian noise matrix with spatially and temporally white entries that are $CN(0,I)$. $H^e$ also contains entries that are $CN(0,I)$, however, these entries are temporally uncorrelated but spatially correlated. This spatially correlated Gaussian interference model is a standard interference model relevant in a communication channel subject to interference and jamming [21]. An example where this model is applicable is when enclosed interfering electronic sources are within close proximity of each other as well as the space-time receiver. This close proximity is significant since the channels that connect the interference-receiver pair can be spatially correlated. The effect of this interference on the STBC receiver is that colored noise is introduced and overall noise power is increased, which in turn degrades conventional STBC receivers designed under the assumption of white noise.

For co-channel interference, $E$ is

$$E = \sum_{p=1}^{P} \sqrt{\frac{\rho_e}{MP}} H^p X^p.$$  

(2.6) $H^p$ and $X^p$ define the channel matrix and information matrix for the $p$th interferer, respectively, with the same description as used for $H$ and $X$.

The third interference source under consideration is periodic sinusoidal interference. In this case, $E$ is

$$E = \sqrt{\frac{\rho_e}{2}} \cos(2\pi \frac{f}{f_s} + \phi) H^s,$$

(2.7)
where \( \frac{f}{f_s} \) denotes the frequency of the interference \( f \) normalized by the sampling frequency \( f_s \), \( H^* \) defines the channel matrix between the interferer and receiver, and \( \phi \) is a random phase with a uniform distribution between \( (0, 2\pi) \). In all interference models, the transmit interference power is normalized such that the average received interference power is \( \rho_c \) at each receive antenna.

In a MIMO system with no interference, the noise is assumed to be spatially or temporally white. For a MIMO system with interference, however, the noise is assumed spatially colored but temporally white. To characterize the combined noise and interference statistics for each STBC frame, the noise covariance matrix \( Q \) is used. This matrix is \( Q = \sigma^2 I \) for spatially white noise, but in the spatially colored case, \( Q \) is an unknown positive definite matrix. \( \sigma^2 \) represents the combined noise and interference power and \( I \) denotes the identity matrix [21].

Coherent STBC detection involves \( H \) and \( Q \) estimation, while noncoherent DSTM detection involves \( Q \) estimation only. The STBC detector performs STBC decoding and makes hard or decisions using the channel \( \hat{H} \) and noise \( \hat{Q} \) estimates. In hard symbol decision decoding, the STBC detector outputs an estimate of the transmitted symbol, with the estimate \( \hat{c}_{t,l} \in C \). In soft symbol decision decoding, the STBC detector outputs a value that not only indicates the estimate of the symbol transmitted, but the value also indicates the reliability of the estimate. In the proposed CML interference suppression scheme, the channel and noise statistics are obtained through training and reiteration.

If a turbo code is connected as an outer channel code, the proposed space-time receivers transfer soft symbol decisions (soft outputs) to the turbo code decoder. The turbo code decoder in turn uses these soft outputs to perform error correction in an iterative fashion. The end result of this error correction process is the output of refined hard symbol estimates that provides coding gain.

If an uncoded system is considered, the STBC detector makes hard symbol estimates directly. These BPSK symbol estimates are then mapped back into groups of binary data bits and the calculation of the BER is performed.
2.2 Orthogonal Space-Time Block Coding

Since its introduction, there has been much interest in developing efficient and high performance STBC codes. Orthogonal STBC (OSTBC) is one such STBC code pioneered by Tarokh et. al [33]. OSTBC coding involves $M \times T$ matrix designs based on orthogonal matrix theory. $M$ represents the number of transmit antennas and $T$ is the time interval occupied by the codeword matrix.

OSTBC is attractive because of its ability to provide full transmit diversity with a simple receiver structure. The diversity order ($v$) offered by STBC is limited by the number of transmit antennas ($M$) and receive antennas ($N$) in the system with $v = M \cdot N$.

A limited number of OSTBC codes exist that provide optimal (unity) transmission rate. The transmission rate, denoted by $R_r$, is the ratio of symbol durations $T$ that are required to transmit $l$ symbols. Consider an $M = 2$ and $T = 2$ OSTBC design with a BPSK constellation. Each group of two information symbols is encoded over two time intervals giving an overall transmission rate of $R_r = l/T = 2/2 = 1$. This $R_r$ rate is possible with the use of two transmit antennas. The OSTBC codes that provide optimum transmission rate performance have been identified in [33]. Further, [33] found that these optimal codes exist for $(M = 2, 4, 8)$ and $(M = 2)$ only, for real and complex OSTBC codes, respectively. For this thesis, the optimum transmission rate ($R_r = 1$) coherent real OSTBC code design that achieves full diversity ($v = M \cdot T = 2 \cdot 2 = 4$) was considered.

2.2.1 OSTBC Encoder

Consider the transmission of a frame of $(T \cdot L = 2 \cdot L)$ BPSK information symbols denoted by $c$. This frame of data is partitioned into $L$ vectors of length $T$, $c = (c_1, \ldots, c_L)$, with each vector denoted by $c_l$. $c$ is the input into the OSTBC encoder. In OSTBC encoding, each transmitted STBC codeword matrix is formed using a one-to-one mapping from a group of $T$ symbols in $c_l = (c_{1,t}, \ldots, c_{T,t})$ into a codeword matrix $X_l$ of dimension $M \times T$. The $(m,t)$ entries of $X_l$ represent the signal transmitted by antenna $m$ in time slot $t$. 
2.2 Orthogonal Space-Time Block Coding

For a $M = 2$ and $T = 2$ complex-based OSTBC encoder, each OSTBC codeword may be formed according to [21][33]

$$X_t = \sum_{t=1}^{2} (\tilde{c}_{t,t}A_t + i\tilde{c}_{t,t}B_t),$$

(2.8)

where $(\cdot)$ and $(\cdot)$ denote the real and imaginary parts of the symbols, $i = \sqrt{-1}$ and $A_t$ and $B_t$ are fixed real-valued "elementary" code matrices of dimension 2x2. An "elementary" matrix is a matrix derived by applying an elementary row operation to the identity matrix [32].

$A_t$ and $B_t$ are chosen to satisfy the following orthogonality constraints:

$$A_tA_t' = I, \quad B_tB_t' = I,$$

$$A_tA_p' = -A_pA_t', \quad B_tB_p' = -B_pB_t', \quad k \neq p$$

(2.9)

$$A_pB_t' = B_pA_t'$$

where $'$ denotes transpose.

For the special case considered here, however, the $A_t$ are given by

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

(2.10)

and $B_1 = B_2 = 0$, leading to the 2x2 real orthogonal design as described in [33].

This selection of elementary matrices leads to a design where each group of two BPSK symbols $c_t = (c_{1,t}, c_{2,t})$ may be mapped into one of four different matrices using the following scheme,

$$(-1, -1) \rightarrow \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix}, \quad (-1, 1) \rightarrow \begin{bmatrix} -1 & 1 \\ -1 & -1 \end{bmatrix},$$

$$\quad (1, 1) \rightarrow \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad (1, -1) \rightarrow \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}.$$  

(2.11)
2.2 Orthogonal Space-Time Block Coding

2.2.2 OSTBC Detection

Coherent maximum-likelihood (ML) detection is facilitated by employing the logarithm of the likelihood function of the received data. In ML decoding, the detector chooses which codeword matrix was sent based on maximizing the squared Euclidean distance metric between the received matrix and the possibly transmitted matrix.

Consider the received frame of signal matrices $R = [R_T^r[1] \ldots R_T^r[K], R[l] \ldots R[L]]$ of dimension $2 \times 2(K + L)$ for a 2x2 MIMO wireless system employing OSTBC encoding,

$$R = \sqrt{\frac{P}{2}} H X + N + E.$$

Denote the channel and noise covariance matrices by $H$ and $Q$, respectively. The log-likelihood function for the received data stream of $(K + L)$ matrices is

$$L(R|H, Q, X) = L(R|H, Q, c)$$

$$= -(K + L) \log |Q|$$

$$- \text{Tr}\{Q^{-1}(R - HX)(R - HX)^*\}$$

where $| \cdot |$ denotes determinant, $\text{Tr}\{\cdot\}$ denotes the trace, and $^*$ denotes conjugate transpose. Conditioning on $c$ is equivalent to conditioning on $X$ because of the one-to-one mapping between the symbols in $c$ and the OSTBC concatenated codeword matrix $X$ [21].

Simplifying (2.12) is possible using the unitary and orthogonality properties of OSTBC. The end result is a simple low complexity receiver involving linear processing only. The desired ML information hard symbol detector is

$$\hat{c}_{t,l} = \arg \min_c \text{Tr}\{Q^{-1}(R - HX)(R - HX)^*\}$$

$$= \arg \max_c \text{Re}\{\text{Tr}\{Q^{-1}HXR^*\}\}$$

$$= \arg \max_c \sum_{l=1}^{L} \text{Re}\{\text{Tr}\{Q^{-1}H(c_{t,l}A_t + i\hat{c}_{t,l}B_t)R^*\}\}$$

$$= \arg \max_c \sum_{l=1}^{L} \sum_{t=1}^{2} \text{Re}\{\text{Tr}\{R[l]^*Q^{-1}HA_t\} + i\text{Im}\{\text{Tr}\{R[l]^*Q^{-1}HB_t\}\}\} \hat{c}_{t,l},$$
where \( \text{Re} \text{Tr}(\cdot) \) and \( \text{Im} \text{Tr}(\cdot) \) respectively denote the real and imaginary part of the trace. \( \hat{c}_{l,t} \) denotes the estimate of the \( t \)th information symbol in the \( l \)th vector that was used to map into a corresponding space-time codeword matrix \( X[l] \).

### 2.3 Differential Space-Time Modulation

Differential phase-shift keying (DPSK) is a well-known modulation technique because it eliminates the need for channel estimation in a single-antenna wireless system. This in turn leads to a lower complexity receiver design. Noncoherent STBC is an extension of DPSK concepts to a multi-transmit wireless system as channel knowledge is not required as well. Noncoherent STBC is commonly referred to as differential space-time modulation (DSTM). DSTM is similar to OSTBC, providing full transmit diversity \( M \cdot N \) with no coding gain compared to OSTBC. The 3dB penalty is an expected result due to DSTM decoding without channel state information. A discussion and analysis between coherent and noncoherent symbol detection can be found in [25].

#### 2.3.1 DSTM Encoder

For DSTM encoding, as in OSTBC encoding, we consider the transmission of a frame of \( (T \cdot L) \) BPSK information symbols, with the frame of data \( c = (c_1, \ldots, c_L) \) also partitioned into vectors \( c_l \) of length \( T \). The first step in DSTM encoding is to map a group of \( T \) symbols in each vector \( c_l = (c_{1,l}, \ldots, c_{T,l}) \) into an information matrix \( G[l] \in \mathcal{G} \), where \( \mathcal{G} \) denotes a set of information matrices \( \mathcal{G} = (G_1, \ldots, G_j) \) determined by DSTM code construction. Each information matrix is in turn differentially encoded into a codeword matrix \( X[l] \) for transmission over the wireless channel.

The matrices in \( \mathcal{G} \) form a group with special properties that allow differential encoding. The theory of group codes is beyond the scope of this thesis. However, the definition of multichannel group codes [11] is given below to assist in the understanding of DSTM.

**DEFINITION:** [Multichannel Group Codes]
2.3 Differential Space-Time Modulation

For any $T \geq M$, where $T$ and $M$ signify time and the number of transmit antennas respectively, let $\mathcal{G}$ be any group of $T \times T$ unitary matrices ($G^*G = GG^* = I$ for all $G \in \mathcal{G}$). Further, let $D$ be a $M \times T$ matrix such that $DG \in \mathbb{C}^{M \times T}$ for all $G \in \mathcal{G}$. The collection of matrices $\mathcal{G} = (G_1, \ldots, G_J)$ is called a multichannel group code of length $T$ with elements taken from a constellation $C$. The transmission rate of this code is given by $R_r = \frac{1}{T} \log_2 |G|$ bits per second per Hertz (b/s/Hz), where $|G|$ denotes the cardinality of $\mathcal{G}$.

The following full-rate, i.e. $R_r = 1$, DSTM group code from [11] with $|G| = 4$ was considered:

$$
D = \begin{bmatrix}
1 & -1 \\
1 & 1 \\
0 & -1 \\
1 & 0
\end{bmatrix}, \quad G_1 = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix},
\quad G_2 = \begin{bmatrix}
1 & -1 \\
0 & 1
\end{bmatrix}, \quad G_3 = \begin{bmatrix}
0 & 1 \\
-1 & 0
\end{bmatrix},
\quad G_4 = \begin{bmatrix}
0 & 1 \\
-1 & 0
\end{bmatrix}.
$$

In addition, gray-code mapping was used to map each vector of $T = 2$ symbols $c_l = (c_{1,l}, c_{2,l})$ into an information matrix $G[l]$, i.e.,

$$
(-1, -1) \rightarrow G_1, \quad (-1, 1) \rightarrow G_2, \quad (1, 1) \rightarrow G_3, \quad (1, -1) \rightarrow G_4.
$$

For the transmission of $(K + L)$ concatenated DSTM codeword matrices $X = [X^{tr}[1] \ldots X^{tr}[K], X[1] \ldots X[L]]$ of dimension $2 \times 2(K + L)$, DSTM encoding is as follows. First, $K$ training matrices $[X^{tr}[1] \ldots X^{tr}[K]]$ are sent with each training matrix given by $X^{tr}[k] = D$. This is then followed by transmission of $L$ differential codeword matrices encoded according to

$$
X[1] = X^{tr}[K]G[l], l = 1
\quad X[l] = X[l - 1]G[l], l = 2, \ldots, L.
$$

2.3.2 DSTM Detection

DSTM detection is similar to OSTBC detection and involves ML decoding. The difference is that knowledge of the channel transfer matrix $H$ is not required for DSTM detection.
2.3 Differential Space-Time Modulation

For noncoherent maximum-likelihood (ML) detection, consider the reception of two consecutive blocks of received data in the \( (K + L) \) frame of received matrices. Let these two matrices of received data be denoted by \( \tilde{R}[l] = [R[l - 1], R[l]] \). When \( G[l] = G_j \), \( G_j \in \mathcal{G} \), the code matrices that affect \( \tilde{R}[l] \) are

\[
\tilde{X}[l] = [X[l - 1], X[l - 1]G_j].
\]

The conditional probability of the last two blocks of received data \( \tilde{R}[l] \) given a possibly transmitted matrix is \([11]\)

\[
\Pr(\tilde{R}[l]|G_j) = \frac{\exp(-\text{Tr}\{R[l]\Sigma^{-1}R[l]^*\})}{|\pi\Sigma|^N}
\]  

where

\[
\Sigma = I + \rho_t \tilde{X}[l]^{*}\tilde{X}[l].
\]

\( \rho_t = \sqrt{\frac{P}{2N}} \) is the amount of transmit power per antenna.

If \( X[l - 1] \) was known at the receiver, the optimal ML detector would depend only on the cross-product matrices

\[
\tilde{X}[l]^{*}\tilde{X}[l] = \begin{bmatrix}
T & T G_j \\
T G_j^{*} & T
\end{bmatrix},
\]  

(2.17)

and this detector would correspond to the following differential space-time unitary quadratic receiver written as \([9]\)

\[
\hat{j} = \arg\max_{j \in \{1, \ldots, 4\}} \Pr(\tilde{R}[l]|G_j)
\]

\[
= \arg\max_{j \in \{1, \ldots, 4\}} \text{Tr}\{\tilde{R}[l]^{*}\tilde{X}[l]^{*}\tilde{X}[l]\tilde{R}[l]^{*}\}.
\]  

(2.18)

\( \hat{j} \) is the index of the estimate of \( G_j \), with \( \hat{G}[l] = G_j \). As in OSTBC detection, given \( \hat{G}[l] \), an estimate of its corresponding group of \( T \) transmitted BPSK information symbols \( c_t = (c_{1,t}, c_{2,t}) \) may be determined.

With the group properties of the considered DSTM code, the cross-product matrices in (2.17) do not depend on the knowledge of the past matrix \( X[l - 1] \) in order to decode the
current codeword matrix $X[l]$. Thus, (2.18) can be further reduced to the following simple DSTM receiver [11]

$$
\hat{j} = \arg\max_{j \in \{1, \ldots, 4\}} \text{Re} \text{Tr} \{ R[l] \bar{X}[l]^{*} \bar{X}[l] \bar{R}[l]^{*}\}
$$

$$
= \arg\max_{j \in \{1, \ldots, 4\}} \text{Re} \text{Tr} \{ R[l] G_{j} R[l]^{*}\}
$$

$$
= \arg\max_{j \in \{1, \ldots, 4\}} \text{Re} \text{Tr} \{ G_{j} R[l]^{*} R[l - 1]\}, \tag{2.19}
$$

where the last step follows from the identity $\text{Tr} \{AB\} = \text{Tr} \{BA\}$.

The DSTM detector in (2.19) is optimal in an environment without interference. To handle interference or spatially correlated noise sources, however, prewhitening is required. It is well-known that a linear transformation of a Gaussian signal will still retain its Gaussian properties. Therefore (2.19) can be modified to account for interference using prewhitening (see Chapter 3, [24]), leading to

$$
\hat{j} = \arg\max_{j \in \{1, \ldots, 4\}} \text{Re} \text{Tr} \{ G_{j} R[l]^{*} Q^{-1} R[l - 1]\}. \tag{2.20}
$$

### 2.4 Cyclic-Based Maximum-Likelihood Estimation

The proposed CML scheme for coherent OSTBC detection acquires initial estimates of the channel $H$ and noise $Q$ matrices using training symbols. These estimates are then refined through an iterative process. For the noncoherent DSTM detection, however, only an estimate of $Q$ is required. In this section, motivation is first provided for using CML estimation to obtain $H$ and $Q$ estimates. This is followed by a description of the CML implementation.

#### 2.4.1 Exact ML Approach with Unknown Channel and Noise

Consider $H$ and $Q$ as unknown quantities. Using the frame of $(K + L)$ concatenated OSTBC codeword matrices $X$, define $\Phi_{X} = X(X^{*}X)^{-1}X^{*}$ and $\bar{\Phi}_{X} = I - \Phi_{X}$. The exact
likelihood function in a MIMO system based on an assumption of thermal noise (EWML) and colored (ECML) is [21]

\[
L_{EWML}(R|c) = \max_{H,Q} L(R|H, Q, X) = -\log \text{Tr}\{R\Phi_{X}^{\dagger}R^{\ast}\} + C,
\]

if \( Q = \sigma^2 I \), and

\[
L_{ECML}(R|c) = \max_{H,Q} L(R|H, Q, X) = -\log |R\Phi_{X}^{\dagger}R^{\ast}| + C,
\]

if \( Q \) is a general positive definite matrix. \( C \) denotes an arbitrary constant.

Evaluation of (2.21) and (2.22) is complex and undesirable for a search over the \( X^{2(K+L)} \) possible sequences is required. Given estimates \( \hat{H}, \hat{Q} \), of \( H \) and \( Q \), however, the use of (2.21) and (2.22) may be avoided. These exact likelihood functions may be approximated using ML detection (2.13) and (2.19) for OSTBC and DSTM detection, respectively. \( \hat{H} \) and \( \hat{Q} \) may be obtained with training symbols.

### 2.4.2 Initial Training-Based ML Approach

The training-based white ML (TWML) and training-based colored ML (TCML) method are now presented for obtaining initial channel \( \hat{H} \) and noise \( \hat{Q} \) estimates.

Given the received STBC training block \( R_{tr} = [R_{tr}^{tr}[1] \ldots R_{tr}^{tr}[K]] \), the ML estimate of the channel gain matrix \( \hat{H} \) is [21]

\[
\hat{H} = R_{tr}X_{tr}^{\ast}(X_{tr}X_{tr}^{\ast})^{-1}
\]

provided that the training block \( X_{tr} = [X_{tr}^{tr}[1] \ldots X_{tr}^{tr}[K]] \) has full-row rank which is easily satisfied by any STBC codeword. Let \( \Phi_{X_{tr}} = X_{tr}(X_{tr}^{\ast}X_{tr})^{-1} \) and \( \Phi_{X_{tr}}^{\dagger} = I - \Phi_{X_{tr}} \). The noise covariance estimates are given by

\[
\hat{Q}_{TWML} = \hat{\sigma}_{ML}I = \frac{1}{NK} \text{Tr}\{R_{tr}\Phi_{X_{tr}}^{\dagger}R_{tr}^{\ast}\}I
\]

\[
\hat{Q}_{TCML} = \left( R_{tr}\Phi_{X_{tr}}^{\dagger}R_{tr}^{\ast} \right) \frac{1}{K}
\]

(2.24)
under the assumptions of spatially white and spatially colored noise, respectively.

The training-based ML algorithm consists of the following steps:

Step 1) Obtain initial channel and noise estimates $\hat{H}$ and $\hat{Q}$ based on the received training block $\mathbf{R}_{tr}$ using (2.23) and (2.24).

Step 2) Apply the estimates obtained in Step 1 to the coherent ML detector given by (2.13) to determine the estimated STBC codeword matrices $\hat{X} = [\hat{X}[1]...\hat{X}[L]]$.

Step 3) Map each estimated STBC codeword matrix $\hat{X}[l]$ back to its associated $T$ information symbols to obtain an estimate of the transmitted BPSK symbol stream $\hat{c}$.

2.4.3 Cyclic-Based ML Approach

The channel and noise estimates obtained with the training-based ML algorithm can be improved using the CML technique. This improvement is achieved by exploiting the knowledge provided in the symbol decisions obtained through the training-based ML detector.

The CML iterative algorithm proceeds as follows:

Step 1) Obtain initial estimates, $\hat{H}$ and $\hat{Q}$ based on the training block $\mathbf{R}_{tr}$ using (2.23) and (2.24).

Step 2) Apply the estimates obtained in Step 1 to the coherent ML detector (2.13) to determine the estimated STBC codeword matrices $\hat{X}$.

Step 3) Re-estimate the channel and noise covariance using formulas (2.23) and (2.24). In this instance, however, $\hat{H}$ and $\hat{Q}$ is re-estimated using $\hat{X}$ in place of the original training matrices.

Step 4) Iterate until convergence or until a predefined number of steps have been carried out.

Step 5) Map each estimated STBC codeword matrix $\hat{X}[l]$ back to
its associated $T$ information symbols to obtain an estimate of the transmitted BPSK data stream $\hat{c}$.

The CML algorithm is essentially a cyclic maximizer of the likelihood function in (2.12) [21]. A cyclic maximizer maximizes a two-variable function by first maximizing the function with respect to the first variable while keeping the second variable fixed. The maximizer then maximizes the function with respect to the second variable while keeping the other variable fixed. These two steps can then be iterated until convergence occurs. As explained in [21], conditions for convergence are satisfied for the MIMO system under consideration.

### 2.4.4 Modifying CML for DSTM

[21] states that for optimal training, an arbitrary OSTBC matrix or a concatenation of several such matrices can be used. Further, given “reasonably” accurate symbol estimates, the accuracy of the re-estimated channel in Step 3 of the CML algorithm is optimal. An examination of “reasonable” accuracy in CML implementation led to the conclusion that a minimum of two training matrices is required in the MIMO system for valid CML estimation. This is because results showed that the use of one training matrix in conjunction with the CML algorithm provided no performance improvement due to the poor channel estimate obtained $\hat{H}$.

In a noncoherent DSTM system, CML implementation with only one training matrix is possible because only $\hat{Q}$ is required. This is done by assuming an initial unknown knowledge of noise (i.e. $\hat{Q} = I$) in the initial training stage. For the special case where only one training matrix is used, we assume an initial $\hat{Q} = I$. The performance is denoted by (DTML) regardless of the spatial white or colored noise assumption replacing TWML and TCML. The DTML noise estimates are then used for the CML iterations. Results in Section 3 indicate that the $\hat{Q}$ obtained using DTML is “reasonable” for a performance improvement can be achieved through the CML iterations.
2.5 STBC Soft-Information Transfer

This section outlines the transfer of soft-information from the OSTBC and DSTM detectors. Soft-information transfer is required in order to employ turbo codes for outer channel coding. This is in contrast to a hard decision STBC detector that yields a binary decision for each BPSK information symbol transmitted. Soft-information is the transfer of reliability information in the form of soft BPSK symbol decisions (soft outputs) using multi-bit quantized data from the STBC detector.

For the OSTBC detector, soft-information transfer is developed based on [21]. For the DSTM detector, however, a new soft reliability metric was developed based on the principles in [12]. In both cases, the log-likelihood values are calculated from the STBC detector and these values are treated as observations from BPSK modulation over an AWGN channel in the turbo code decoder. Although this approximation is not optimal, it provides a reasonable trade-off between complexity and performance as the optimum solution gives rise to a complex metric computation [37]. Results are given in Section 4 to demonstrate the suitability of this approach.

2.5.1 Soft Output OSTBC Detection

Soft-information transfer from the OSTBC detector to the turbo decoder is considered by first evaluating the likelihood function for the received data. Recall the OSTBC ML hard decision symbol detector output is given by

$$\hat{c}_{t,l} = \arg \max_{c_{t,l}} \sum_{t=1}^{L} \sum_{l=1}^{2} \text{Re} \left( \text{Re} \text{Tr} \{ R[l]^* Q^{-1} H A_t \} + i \text{Im} \text{Tr} \{ R[l]^* Q^{-1} H B_t \} \right) c_{t,l}.$$

The hard decision detector can be modified to facilitate soft decisions by defining the following $T \cdot L = 2 \cdot L$ complex quantities representing the information symbols transmitted in a frame of data [21]

$$\phi_{t,l} \triangleq \gamma \left[ \text{Re} \text{Tr} \{ R[l]^* Q^{-1} H A_t \} + i \text{Im} \text{Tr} \{ R[l]^* Q^{-1} H B_t \} \right] c_{t,l}, \quad (2.25)$$
for \( t = 1, 2 \) and \( l = 1, 2, \ldots, L \). \( \gamma \) is defined as

\[
\gamma \triangleq \text{Tr}\{H^*Q^{-1}H\}. \tag{2.26}
\]

Equipped with knowledge of \( H, Q \) and 2.12, let

\[
L(R|H, Q, X) = -(K + 2 \times L) \log |Q| - \text{Tr}\{Q^{-1}(R - HX)(R - HX)^*\} \\
= -(K + 2 \times L) \log |Q| - \text{Tr}\{Q^{-1}(RR^* + H(XX^* + 2 \times LI)H^*)\} \\
+ 2\gamma \sum_{t=1}^{T} \sum_{l=1}^{L} \text{Re}(\phi_{t,l}c_{t,l}),
\]

which shows that \( \phi_{t,l} \) can be interpreted as likelihood values and represent sufficient statistics for the elements in each group of \( T = 2 \) information symbols \( c_t = (c_{1,t}, c_{2,t}) \) representing the OSTBC codeword transmitted. Therefore \( \phi_{t,l} \) can be used as an input to a turbo code decoder.

Note that the value defined in (2.25) differs slightly from that given in [21] by the factor \( \gamma \). In [21], the observed quantities were assumed circularly symmetric complex Gaussian random variables with mean \( c_{t,l} \) and variance \( 1/\gamma \). Here an equivalent representation is used with observations assumed similarly distributed but with mean \( c_{t,l} \) and variance \( \gamma \) as in [9][11].

### 2.5.2 Soft Output DSTM Detection

Assume that the transmission of the information matrices \( G[l] \in \mathcal{G} \) are identically and independently distributed over the four values given in \( \mathcal{G} \). Conditioned on the reception of the last two received blocks of data \( \bar{R}[l] \), the likelihood that the first transmitted symbol \( c_{1,l} \)
2.6 Turbo Coding

in a $T = 2$ symbol group $c_t = (c_{1,t}, c_{2,t})$ is equal to one can be written as [12][15]

$$
Pr(c_{1,t} = 1 | \bar{R}[l]) = \frac{Pr(c_{1,t} = 1)Pr(\bar{R}[l]|c_{1,t} = 1)}{\sum_{j=1}^{4} (Pr(G_j)Pr(\bar{R}[l]|G_j))} = \frac{Pr(c_{1,t} = 1)Pr(\bar{R}[l]|c_{1,t} = 1)}{Pr(\bar{R}[l])}
$$

$$
\propto Pr(\bar{R}[l]|c_{1,t} = 1)
\propto Pr(\bar{R}[l]|G_3) + Pr(\bar{R}[l]|G_4),
$$

where $c_{1,t} = 1$ maps directly into two of four possible information matrices in the DSTM code. For the DSTM group code considered, $c_{1,t} = 1$ maps to $G_3$ and $G_4$. These likelihood values are then used as suboptimum soft reliability metrics in an outer channel code. Using the same principle, we obtain

$$
Pr(c_{2,t} = 1 | \bar{R}[l]) \propto Pr(\bar{R}[l]|G_2) + Pr(\bar{R}[l]|G_3)
$$

In the outer channel code decoder, the likelihood values calculated in (2.28) and (2.29) are in turn treated as observations from BPSK modulation over an additive white Gaussian noise channel. Note that only a minor modification is required to obtain soft outputs in this DSTM detector as $Pr(\bar{R}[l]|G_j)$ with $j \in (1, \ldots, 4)$ is necessarily calculated in a DSTM hard-decision detector.

2.6 Turbo Coding

Turbo coding is a relatively new and powerful channel coding technique first introduced by Berrou et al. [3]. In [3], performance to within fractions of a decibel (dB) of Shannon capacity was demonstrated using turbo codes. The turbo code encoder consists of concatenating two identical component codes in parallel and separating the codes by an interleaver of length $N$. The code rate $R$ for a turbo code is determined by the number of input data streams $d$ and the number of output data streams $q$ used in turbo encoding with $R = d/q$. It is possible to increase the code rate of a turbo code by puncturing the parity data, but this
increased rate is achieved with a loss in BER performance [38]. This encoder structure is desirable for it allows the use of a soft input soft output (SISO) decoder implementation with an efficient iterative decoding algorithm. The end result is that turbo codes provide a low error rate with an overall decoding complexity much less than that required for a single code of relative performance [37].

2.6.1 Turbo Code Encoder

The block diagram of a typical turbo code encoder is shown in Figure 2.2. A turbo code encoder consists of two recursive systematic convolutional (RSC) codes separated by an interleaver of size $N$. Note that although the turbo code encoder in Figure 2.2 indicates four output data streams, the systematic output in RSC encoder 2, denoted by $u'_4$, is not
used. Thus the code rate for this turbo code is $\mathcal{R} = 1/3$ since the number of input data streams and the number of output data streams used are $d = 1$ and $q = 3$, respectively.

The use of RSC codes are desirable for [3] has shown that these class of convolutional codes provide the best BER performance at any SNR for high code rates. As shown in Figure 2.2, the RSC code can be designed through a serial connection of one or more shift registers and one or more exclusive or's (+). The RSC codes used in a turbo code are identical and can be represented by a generator matrix

$$G(D) = \begin{bmatrix} 1, & \frac{g_1(D)}{g_0(D)} \end{bmatrix},$$

for encoding. The generator matrix is useful for turbo encoding as it provides a description in how to attain the current coded bit based on the current input bit and the state of the RSC encoder. The fundamentals behind the generator matrix description is extensive so for further information the reader is referred to [37][39].

The RSC encoders can be further described by its code rate and its memory length. The code rate for a RSC code, similar to the code rate for a turbo code, is also determined by the number of parallel input data streams $b_{RSC}$ and the number of output data streams $q_{RSC}$ in the RSC encoder. The RSC code rate is $\mathcal{R}_{RSC} = b_{RSC}/q_{RSC}$. The memory length, denoted by $\nu$, of a RSC code indicates the number of shift registers used to implement the RSC code. A parameter related to the memory length and commonly used is the constraint length $K_l$ of a RSC code, given by $K_l = \nu + 1$.

An interleaver in a turbo code is used to permute the data such that the input sequences to each component code are approximately uncorrelated, thus allowing for extraction of new information in iterative decoding. Its length $\mathcal{N}$ determines the frame length for data transmission. This is because the interleaver operation is performed by reading a block of $\mathcal{N}$ bits and writing out these same bits in a pseudo-random manner. Here $\mathcal{N}$ describes the frame length for turbo coding which is different from the STBC coding frame length of $2(K + L)$. $\mathcal{N}$ is chosen to be an integer multiple of $2(K + L)$.

In this thesis, the following generator matrices describing 4-state and 16-state RSC
2.6 Turbo Coding

Encoders, respectively, were considered:

\[ g^{(4)}_1(D) = (1 + D^2) \]
\[ g^{(4)}_0(D) = (1 + D + D^2) \]

(2.31)

\[ g^{(16)}_1(D) = (1 + D + D^2 + D^3 + D^4) \]
\[ g^{(16)}_0(D) = (1 + 0 + 0 + 0 + D^4), \]

where \( D \) denotes the delay operator. These RSC encoders were chosen because of the excellent BER performance achievable as described in [31] [36] [41].

The input to each RSC encoder at time \( t \) is a bit \( b_t \) and the corresponding three bit codeword (\( R=1/3 \)) for \( b_t \) is the multiplexed output \( c_t = (u_t, v_t, z_t) \), where

\[ u_t = b_t \]
\[ a_t = u_t + \sum_{i=1}^{(K-1)} g_{1i} \mod 2 \]
\[ v_t = \sum_{t=0}^{(K-1)} a_t \mod 2 \]
\[ u'_t = b'_t \]
\[ a_t = u'_t + \sum_{i=1}^{(K-1)} g_{1i} \mod 2 \]
\[ z_t = \sum_{t=0}^{(K-1)} a_t \mod 2. \]

\( z_t \) is the output from the second RSC encoder which takes as input a re-ordered data stream \( b'_t \) as determined by the interleaver. \( g_{1i} \) denote the values in the shift registers with taps that extends to an exclusive or gate in the feedforward direction. In the following, without loss of generality, we limit the turbo coding description to the 4-state turbo code considered in (2.31).
Figure 2.3. State diagram for (5,7) RSC encoder [36]

2.6.1.1 State and Trellis Representation of RSC Encoders

RSC encoders can be described in equivalent forms using a state or trellis diagram. The state diagram interprets the RSC encoder as a finite state machine, while the trellis diagram represents the outputs and state transitions of the RSC encoder based on the input bit from one time interval to another. These representations are important because of their use in explaining decoding algorithms such as the soft output Viterbi algorithm (SOVA) in the turbo code decoder.

For the state diagram description, the encoder state is defined as its memory content. Given an RSC encoder with $v$ memory elements, the total number of possible states is $2^v$. The current state and the output of the RSC encoder are determined by the RSC encoder's previous state and its current input. The state diagram for the 4-state RSC encoder is shown in Figure 2.3.

From the state diagram, a trellis diagram can easily be derived by tracing all possible input/output sequences and state transitions versus time.

The following are some interesting properties of the trellis [37]:

1. Every codeword in a RSC code is associated with a unique path. If proper trellis termination is used to ensure that the each frame begins in the all zero-state, then this unique path begins and stops at state $S_0$. 


2. After $\nu$ time instants, each trellis diagram will have $2^{\nu} = 2^2 = 4$ nodes at each time step.

3. Each node has $2^n = 2^1 = 2$ branches leaving each node, and after $m$ time instants, each node will have $2^n = 2^1 = 2$ branches entering each node.

4. For $d = 1$, given an input sequence of $dN = N$ bits, and assuming $\nu$ state transitions are necessary to ensure the trellis is terminated to state $S_0$, the trellis diagram must have $N + \nu$ stages.

2.6.1.2 Trellis Termination

Trellis termination is necessary at the end of each frame to ensure that the initial state for the next frame is the all-zero state. Due to the recursive nature of the RSC encoder, however, the tail bits required for termination will depend on the current state of the component encoder which is based on the input data. A solution to this problem would be to ensure that the encoder is reset to the all-zero state after $\nu$ cycles [35][37]. However, since each RSC encoder takes in the data stream in different order, terminating one RSC encoder will not necessarily terminate the other. In practice, only the first RSC encoder is terminated to the all-zero state [35]. Negligible performance degradation is produced by an unknown
2.6 Turbo Coding

Figure 2.5. Block diagram of generic turbo code decoder [37]

final state of the second encoder for sufficiently large interleaver size [10].

2.6.2 Turbo Code Decoder

A turbo code decoder is based on iterative information transfer between constituent decoders serially concatenated by the same interleaver used in the turbo code encoder. The information sharing between the two decoders leads to improved performance because the interleaver was used to create two weakly correlated codewords. This allows for the extraction of refined data estimates in each iteration. The turbo code decoder is shown in Figure 2.5.

Turbo decoding begins with the estimated codeword BPSK symbols from the STBC detector being fed into SOVA DECODER 1. SOVA DECODER 1 outputs the data’s associated extrinsic information \( L_{e1,t} \) which in turn is permuted by the same random interleaver as was used in turbo encoding. This permuted sequence of \( L_{e1,t} \) is fed into SOVA DE-
The SOVA algorithm was implemented in this work because a good tradeoff between performance and complexity is provided. SOVA requires less than half the computational complexity than the log-MAP algorithm at the expense of a minor penalty of approximately 1dB in extra power [41].
2.6 Turbo Coding

2.6.2.1 The SOVA Algorithm for Iterative Decoding

To understand SOVA, let us consider the well-known Viterbi algorithm. We first assume the transmission of a frame of turbo encoded data of $N$ symbols, denoted by $c = (c_1, \ldots, c_N)$. Note that $c$ as defined here differs, by the frame length, from the definition given previously in STBC coding. Here $c$ represents the symbol sequence of turbo encoded data of frame length $N$ for turbo coding. This is different from the symbol sequence of STBC encoded of frame length $2(K + L)$. Recall $N$ is chosen to be an integer multiple of $2(K + L)$.

To determine the corresponding bit sequence $b = (b_1, \ldots, b_N)$, the VA algorithm computes the most probable estimated code sequence conditioned on the received symbol sequence transferred from STBC detector. To assist in the explanation of the VA, and subsequently SOVA, we redefine the notation used for the estimated symbols from the STBC detector, i.e. $c$ as referred to in Sections 2.2.2 and 2.3.2. In the following discussion on SOVA, $c$ now refers to the ML estimated code sequence as determined from SOVA decoding. The $N$ estimated symbols representing the transmitted turbo encoded sequence that were STBC detected are now defined as

$$c = (\hat{c}_1, \ldots, \hat{c}_N) \rightarrow r = (\hat{r}_1, \ldots, \hat{r}_N)$$

(2.33)

where an estimated symbol $\hat{c}_t$ at time $t$ from the STBC detector is now redefined as a received symbol $r_t$ that is input into the turbo code decoder at time $t$. This notation is used to avoid confusion in the SOVA explanation.

The most probable estimated code sequence $\hat{c}$ conditioned on the received symbol sequence $r$ from the STBC detector can be written as

$$\hat{c} = \arg \{ \max_c \Pr(r|c) \},$$

(2.34)

where the probability $\Pr(r|c)$ of the received sequence $r$ of length $N$ conditioned on the transmitted code sequence $c$ is [37]

$$\Pr(r|c) = \prod_{t=1}^{N} \prod_{i=0}^{qRSC-1} \frac{1}{\sqrt{(2\pi\sigma^2)}} e^{-\frac{(r_{i,t} - c_{i,t})^2}{2\sigma^2}}.$$
\( q_{RSC} \) is the number of coded bits generated by an RSC code for each message bit transmitted. For a rate \( q_{RSC} = 1/2 \) RSC code, \( q_{RSC} = 2 \) as previously described.

Simplifying this expression is possible by using the log likelihood of \( \Pr(r|c) \)

\[
\log \Pr(r|c) = \sum_{t=1}^{N} \log \Pr(r_t|c_t).
\]  

(2.36)

For the Gaussian channel, (2.36) is

\[
\log \Pr(\hat{r}|c) = \sum_{t=1}^{N} \log \left( \prod_{i=0}^{q_{RSC}-1} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(r_{t,i} - c_{t,i})^2}{2\sigma^2}} \right) \\
= -\frac{q_{RSC}N}{2} \log(2\pi) - q_{RSC}N \log \sigma - \sum_{t=1}^{N} \sum_{i=0}^{q_{RSC}-1} \frac{(r_{t,i} - c_{t,i})^2}{2\sigma^2}.
\]  

(2.37)

Note that maximizing \( \Pr(r|c) \) is equivalent to minimizing the Euclidean difference between the received sequence and the sequences in the trellis diagram i.e. (2.37).

\[
\hat{c} = \arg\max_c \Pr(r|c)
\]

\[
\hat{c} = \arg\min_c \sum_{t=1}^{N} \sum_{i=0}^{q_{RSC}-1} \frac{(r_{t,i} - c_{t,i})^2}{2\sigma^2}
\]  

(2.38)

(2.38) can be rewritten in an equivalent form by maximizing the expression

\[
\hat{c} = \arg\max_c \sum_{t=1}^{N} \sum_{i=0}^{q_{RSC}-1} r_{t,i}c_{t,i}
\]  

(2.39)

To determine the ML path, each branch at time \( t \) on the path is assigned a Euclidean distance called a branch metric. With (2.39), the branch metric is

\[
V_t^c = \sum_{i=0}^{q_{RSC}-1} r_{t,i}c_{t,i},
\]  

(2.40)

where the path metric corresponding to path \( cp \) on the trellis is

\[
p_t^{(cp)} = \sum_{t'=1}^{t} V_{t'}^{(c)}
\]

\[
= p_{t-1}^{(cp)} + V_t^{(c)}
\]  

(2.41)

A summary of the VA is outlined below [37][39]:
2.6 Turbo Coding

1. Set initial values: \( t=0; S_0 = 0; p_{0}^{(cp)}(S_0 = 0) = 0; p_{0}^{(cp)}(S_0 \neq 0) = \infty; \)

2. At time \( t \), compute the partial path metrics for all paths entering each node using equation (2.41).

3. Compare the path metrics for all paths entering a node and find the survivor for each node. Ties can be broken by flip of a coin. Nonsurviving branches are deleted from the trellis, while the surviving path and its metric is saved.

4. If \( t \leq N + \nu \), increment \( t \) and return to step 2.

5. If \( t = N + \nu \), trace back all survivor paths using the trellis beginning from node with largest path metric at \( t = N + \nu \).

The surviving path identified through traceback corresponds to the ML code sequence [37][39].

Iterative SOVA decoding is a modified VA algorithm with two significant changes. First, SOVA takes into account \textit{a priori} information when selecting the ML path. Second, SOVA produces soft outputs as opposed to the hard symbol estimates generated in VA. With these two changes, the use of SOVA in a turbo code decoder is made possible.

The SOVA algorithm estimates soft outputs for each transmitted binary symbol in the form of

\[
L(c_t) = \log \frac{\Pr(c_t = 1)}{\Pr(c_t = -1)},
\]

where \( L(\cdot) \) defines the log-likelihood ratio (LLR). This soft output \( L(c_t) \) is passed to the subsequent SOVA decoder for use as \textit{a priori} information in the next iterative decoding stage. Once the soft outputs have been refined using a predefined number of iterations in iterative decoding, the SOVA decoder makes a hard decision on these soft outputs. The hard symbol estimates are the ML estimates of the \( N \) bits \( c = (c_1, \ldots, c_N) \) originally transmitted.

The required modifications to adapt VA to SOVA are outlined as follows [37][41]. To obtain soft outputs, a new path metric is generated based on maximizing the logarithm of
the a posterior probability, $\Pr(c, r)$

$$\log \Pr(c, r) = \log \Pr(c) + \log \Pr(r|c). \quad (2.43)$$

Given an independent Gaussian noise channel, a suitable path metric for each path $cp$ in SOVA decoding is [37], [41]

$$p_t^{(ep)} = p_{t-1}^{(ep)} + \frac{1}{2} c_t L(c_t) + \frac{L_c}{2} V_t^c, \quad (2.44)$$

where $c_t L(c_t)$ represents the a priori information and $V_t^c$ is the branch metric as calculated in the VA algorithm. $L_c$ is the channel reliability factor serving as a scaling for the received signal based on channel conditions. $L_c$ can be written as

$$L_c = \frac{4aE_s}{N_o} \quad (2.45)$$

where $E_s$ is the signal energy, $N_o$ is the power spectral density of the noise, and $a$ is the fading amplitude of the channel. This modification of VA is the first change required in converting VA to the SOVA algorithm.

The second modification for SOVA is to generate soft outputs for use by the next component decoder in its respective path metric calculations. This is done by first examining the metric difference for each state at each time instant. Consider the binary trellis in used in the 4-state RSC encoder. Each node, $S_j(t)$, that represents a state $S_j$, $j \in \{1, \ldots, 4\}$ at time $t$, will always have two incoming paths $p_t^\varepsilon$ and $p_t^\bar{\varepsilon}$.

The path that is selected as the survivor path is the one with the higher metric. Assume that the survivor path is $p_t^\varepsilon$ and the discarded path is $p_t^\bar{\varepsilon}$. The metric difference $\Delta_t^{S_j}$ can be defined as [7][41]

$$\Delta_t^{S_j} = p_t^\varepsilon - p_t^\bar{\varepsilon} \geq 0. \quad (2.46)$$

The metric difference defined in (2.46) can be approximated by the probability that a correct decision is made in selecting the survivor path over the discarded path [7][41], given by

$$\Pr(\text{correct decision at } S_j(t) = S) = \frac{\Pr(\varepsilon)}{\Pr(\varepsilon) + \Pr(\bar{\varepsilon})}. \quad (2.47)$$
(2.47) can be simplified and rewritten as [7][41]

\[
\Pr(\text{correct decision at } S_j(t) = S) = \frac{e^{\Delta_t^{S_j}}}{e^{\Delta_t^{S_j}} + e^{-\Delta_t^{S_j}}} = \frac{e^{\Delta_t^{S_j}}}{1 + e^{\Delta_t^{S_j}}},
\]

(2.48)

thus showing that the metric difference \( \Delta_t^{S_j} \) represents the reliability that a path ending at state \( S_j \) at time \( t \) is correct. Equipped with the above knowledge, the generation of soft outputs for each symbol \( c_t \) in the SOVA algorithm may now be explained as follows outlined in the following two steps.

The first step in SOVA is to identify the ML path through the trellis as in the VA algorithm. In contrast to the VA algorithm, however, SOVA must not only obtain the hard decision for each bit, but it must also calculate and store the metric difference between the two incoming paths merging into the same state.

The second step in SOVA involves calculating the LLR of the bit sequences, (2.42), that give the reliability of the bit decisions along the ML path. This is done by tracing back along the path, using the stored difference metrics, and considering the probability that the paths merging with the ML path in the trellis were incorrectly discarded.

The LLR for each symbol can be approximated by [7]

\[
L(c_t|\mathbf{r}) \approx c_t \min_{i=t...t+\delta, c_i \neq c_t} \Delta_t^{S_j}
\]

(2.49)

where \( c_t \) is the value of the symbol given by the ML path and \( c_t' \) is the value of the symbol for the path which merged with the ML path and was discarded at trellis stage \( t \). \( \Delta_t^{S_j} \) represents the values of the metric difference for all states \( S_j \) along the ML path.

The approximated LLR for each symbol calculated in the current SOVA decoder, (2.49), is used to determine the extrinsic information used as a priori information in the next SOVA decoder. Assume that the LLRs for each symbol in SOVA DECODER 1 have been calculated. The extrinsic information for each symbol used in SOVA decoder 2 can be written as [37]

\[
L_{e1,t} = \hat{c}_t L(c_t) - 2\hat{c}_t - L_{e2,t}.
\]

(2.50)
where \( L_{e2,t} \) is assumed zero initially in the first iteration of SOVA decoding. After the desired number of iterations, the final hard decisions on the estimated symbol streams are made using the refined LLR for each bit, (i.e. \( \hat{c}_t L(\hat{c}_t) \), see Figure 2.5). The calculation of the estimated symbol streams are then followed with a conversion of these estimated symbols into an estimate of the original source data \( \hat{b} = (\hat{b}_1, \ldots, \hat{b}_N) \). For a thorough illustration and explanation of these steps, see [41].

Lastly, a note for SOVA implementation, recall that the first step in SOVA is to identify the ML path. If the sequence length is long or infinite, the survivors must be truncated to a manageable length. This length is called the decoding depth \( \delta \). At decision time \( t' = \delta + t \), the decoding depth must be long enough to ensure that all survivors will stem from the same branch in the beginning starting from time \( t \). A decoding depth of at least six times the constraint length of convolutional code is sufficient for this purpose [37].
Chapter 3

Performance of CML in OSTBC and DSTM

This chapter presents the simulation results for 2x2 MIMO wireless systems implementing STBC under different operating conditions. OSTBC and DSTM code designs were used for coherent and noncoherent STBC coding, respectively. The STBC detectors considered either incorporate no interference suppression, suppression using training data, or suppression using refined channel and noise covariance estimates based on CML. Our investigation showed that a gain in performance occurs with the color-based STBC detectors only when the detectors operate in an environment with strong interference.

For the coherent CML-based OSTBC detectors, our results agreed with [21] showing that most of the gain in performance using the CML technique occurs with one iteration. Thus, results are only presented for the training-based OSTBC detectors and the CML-based OSTBC detectors with one iteration of the CML technique. For the noncoherent CML-based DSTM detectors, our investigation showed that a gain in performance occurs using up to two iterations of the CML technique. However, no gain is further achieved after the second iteration. Thus, results are presented for the colored noise CML-based DSTM detectors with one and two iterations only, respectively denoted by (1 Iters DCML), (2 Iters DCML). Note that these findings are based on simulation results. In a practical implementation of these STBC detectors, it would be necessary to determine the optimal number of iterations that will provide the best performance based on the trade-off with computa-
tional complexity.

In the spatially correlated Gaussian interference model, the interference is assumed fully correlated. Unless specified otherwise, the signal power is equal to the interference power, implying a system operating with a signal-to-interference ratio (SIR) of 0 dB. Further, the detection of a received frame of data consisting of $L = 14$ information matrices and $K = 2$ training matrices was assumed for OSTBC, and $L = 15$ and $K = 1$ for DSTM. This represents training overheads of $2/16 = 12.5\%$ and $1/16 = 6.75\%$, respectively.

### 3.1 Validating CML Performance in OSTBC and DSTM

Figure 3.1 presents CML performance for an OSTBC system in a white noise environment with no interference. Results show that the ML detectors designed under an assumption of white noise outperform the colored noise detectors. For the CML scheme, refining the $\hat{H}$ and $\hat{Q}$ estimates using one iteration of CML for the white noise detectors (1 Iter WML) and colored noise detectors (1 Iter CML) provide a gain exceeding 1 dB and 2 dB, respectively, when compared to the initial estimates obtained in the training-based white noise (TWML) and colored noise (TCML) schemes. For reference purposes, a coherent-based single-antenna system implementing BPSK modulation is provided in Figure 3.1 to illustrate the superior performance achievable in a multiple-antenna system that is provided by diversity gain.

Figure 3.2 shows the CML performance with one strong interferer in an OSTBC system. The interferer is modeled as spatially correlated Gaussian interference. Here the results differ from [21] by a constant 3 dB penalty in SNR performance. This is due to our assumption of normalizing transmit power. This normalization results in attaining a received power per symbol period of $\rho$ as opposed to $2\rho$ in [21].

The poor performance of the receivers designed under the assumption of white noise emphasizes the need for proper interference suppression. The colored noise receivers, on the other hand, perform adequate noise interference suppression. Results show that the
colored noise receiver using CML (1 Iter CML) outperforms the colored noise receiver using the training-based method (TCML) by more than 2dB. Note that when the SNR is sufficiently low, the white noise receivers perform somewhat better. This is because the covariance matrix describing the noise and interference is close to a scaled identity matrix [21].

In Figure 3.3, a DSTM system with one strong interferer is considered. Although a relatively high error floor exists, recall that only one training matrix is employed and that $SIR = 0dB$. Performance improvements are achievable by considering a system with limited interference power or by increasing the number of training matrices. This is demonstrated in Section 3.2 and Section 3.3, respectively.

### 3.2 Impact of the Interference Power

In this section, the performance of CML is evaluated using two different approaches. In the first approach, the performance of the proposed STBC detectors is determined in an environment with limited interference power. This is done by varying the SNR and fixing the interference power $\rho_e$ in the system. Figures 3.4 and 3.6 present CML performance results in an OSTBC system by varying the SNR and fixing the interference power of the system to $\rho_e = 10dB$ and $\rho_e = 20dB$, respectively. Figures 3.5 and 3.7 also present performance results by varying the SNR with $\rho_e = 10dB$ and $\rho_e = 20dB$, respectively. However, Figures 3.5 and 3.7 refer to CML performance in a DSTM system. Here the improvement of using CML for Q estimation in a DSTM system is evident. For example, Figure 3.5 show performance gains exceeding 4dB are achievable for the CML-based colored noise receivers (1 Iter DCML) when compared to the conventional white noise receiver. However, performing an additional CML iteration (2 Iters DCML) provides a diminishing improvement of approximately 0.5dB.

The second approach in evaluating CML performance is to fix the SNR and vary $\rho_e$. Using this approach determines the asymptotic performance of the STBC detectors. Fig-
ures 3.8 and 3.9 demonstrate CML performance in an OSTBC system and a DSTM system, respectively. In both Figures, $SNR = 15dB$ and $\rho_e$ is varied. We note that even as the interference power increases asymptotically, CML mitigates interference to an error floor of $10^{-2}$ and $10^{-1}$ for the CML-based colored noise receivers in an OSTBC and a DSTM system, respectively.

### 3.3 Effect of the Number of Training Matrices

The interference suppression capability of the proposed STBC detectors can be improved if the number of training matrices is increased. For an OSTBC system, Figure 3.10 shows that $K = 4$ is sufficient to provide the best BER performance. Figure 3.11, on the other hand, shows that only $K = 3$ is required in a DSTM system to achieve the best BER performance. This improvement comes at a great cost of efficiency since the training overheads are 25% and 18.75%, respectively, for the OSTBC and DSTM system.

### 3.4 Effect of Doppler Fading

In this section, the quasi-static flat fading channel assumption is dropped and the flat fading channel model is modified to incorporate Doppler fading. Figures 3.12 and 3.13 show the robustness of DSTM over OSTBC as the channel condition changes from a slow to fast fading environment. The advantage of using noncoherent DSTM over coherent STBC modulation is evident. The crossover point in performance between these two schemes, illustrated in Figure 3.13, occurs at a low normalized Doppler frequency $\frac{f}{f_s} = 0.008$. This crossover point is the best-case for the OSTBC system under consideration, as known initial $H$ and $Q$ were assumed.
3.5 Applying Different Interference Models

In this section, the performance of CML is determined without the ideal fully spatially correlated Gaussian interference assumption. Instead, CML performance is presented with co-channel and periodic sinusoidal interference. Co-channel interference may be prevalent in a wireless communication system designed for multiple users while periodic interference sources may be encountered in enclosed electronic devices containing local oscillators or clock sources.

In Figure 3.14, the performance of CML is presented in an OSTBC system with synchronized co-channel interference using (2.6). In this system, the SNR and total interference power are $SNR = 15dB$ and $\rho_e = 15dB$, respectively. Recall that for the co-channel interference model under consideration, the total interference power is constant and equally distributed amongst the $P$ interferers. The results show that as the number of interferers increases, the Gaussian assumption becomes more valid and performance improves.

Figure 3.15 investigates CML performance with one periodic sinusoidal interference. The interference frequency was normalized with respect to the sampling frequency $f_s = 30KHz$. We first note that the CML-based colored noise schemes provide an improvement in the space-time receiver performance, with (1 Iter CML) providing the best amongst all estimation schemes. Further, the results indicate that the BER performance varies with frequency as the colored noise receivers perform well in the range of $\frac{f_c}{f_s} = 0.15$ to $\frac{f_c}{f_s} = 0.35$. Although performance degrades significantly outside this region, recall that these results are worst-case as only one interferer is assumed present. These results are expected to improve if a more realistic scenario with a greater number of periodic interferers is assumed.

To show this, Figure 3.16 presents CML performance when the interference is modeled either as a single periodic interference source (Periodic) or as a mixture of periodic and correlated Gaussian interference (Periodic and correlated Gaussian). The frequency of the periodic noise was chosen as $\frac{f_c}{f_s} = 0.10$ and $SIR = 0dB$. With the mixed periodic and correlated interference model, the interference power was equally divided among the two
interferers. Results show that with interference modelled as purely periodic, CML performance experiences an approximate 2dB penalty in performance in the high SNR region. However, with a more realistic scenario of periodic and correlated Gaussian scenario, there is no performance penalty.

3.6 Summary

This chapter provided the simulation results for 2x2 MIMO wireless systems with white and colored noise STBC receivers implemented under several conditions. First, the effectiveness of the CML-based colored noise receivers in suppressing unknown interference was shown. Second, the effect of the number of training matrices on CML performance was evaluated. Results indicate that $K = 4$ and $K = 3$ training matrices are sufficient to provide the best BER performance for an OSTBC system and a DSTM system, respectively. Third, the effect of Doppler fading on CML performance was investigated as the channel condition was changed from a slow to a fast fading environment. The advantage of using noncoherent DSTM over coherent OSTBC modulation is evident, as the crossover point in performance between the two schemes occur at low normalized Doppler frequency of $f_d = 0.008$. Lastly, the robustness of CML-based interference suppression was determined. Results are positive as the colored noise receivers performed well when different interference models were applied.
3.6 Summary

Figure 3.1. Performance of CML in an OSTBC system with no interference.

Figure 3.2. Performance of CML in an OSTBC system with one strong interferer.
3.6 Summary

Figure 3.3. Performance of CML in a DSTM system with one strong interferer.

Figure 3.4. Performance of CML in an OSTBC system with $\rho_e = 10\,dB$. 
3.6 Summary

Figure 3.5. Performance of CML in a DSTM system with $\rho_e = 10\text{dB}$.

Figure 3.6. Performance of CML in an OSTBC system with $\rho_e = 20\text{dB}$.
3.6 Summary

**Figure 3.7.** Performance of CML in a DSTM system with $\rho_e = 20\text{dB}$.

**Figure 3.8.** Effect of varying $\rho_e$ on CML performance in an OSTBC system.
Figure 3.9. Effect of varying $\rho_e$ on CML performance in a DSTM system.
Figure 3.10. Effect of the number of training matrices on CML performance in an OSTBC system.
3.6 Summary

Figure 3.11. **Effect of the number of training matrices on CML performance in a DSTM system.**
3.6 Summary

![Figure 3.12](image1.png)

**Figure 3.12.** Effect of Doppler fading on CML performance in an OSTBC system.

![Figure 3.13](image2.png)

**Figure 3.13.** Effect of Doppler fading on CML performance in a DSTM system.
Figure 3.14. Effect of co-channel interference on CML performance in an OSTBC system.

Figure 3.15. Effect of periodic interference on CML performance in an OSTBC system.
Figure 3.16. CML performance in an OSTBC system with a mixture of correlated Gaussian and sinusoidal interference.
Chapter 4

Performance of CML and Turbo-Coded MIMO Wireless System

In this chapter, the performance of serially concatenated space-time block codes and turbo codes are investigated. For serial concatenation, the STBC receivers are modified to produce soft outputs as described in Section 2.5, with infinite-bit quantized soft outputs assumed transferred from the STBC receivers to turbo code decoder.

To determine the effect of turbo code parameters on system performance, results are presented for systems implementing concatenated OSTBC and turbo codes using different constraint lengths and code rates.

A random interleaver For data permutation in turbo coding, a random interleaver was assumed, with each interleaver independently generated on a frame by frame basis with no attempt made to optimize the interleaver design. Interleaver length of $N = 256$ and $N = 1024$ are used. These interleaver lengths are chosen to be relatively short, as compared to the original $N = 65536$ turbo code first introduced in [3], because certain wireless applications may require a limited decoding delay time (e.g. voice applications).

An additional rectangular block channel interleaver of length $N$ was added to mitigate against burst errors in the Rayleigh fading channel. This channel interleaver is inserted after the turbo code encoder and its associated channel deinterleaver is inserted right before the turbo code decoder (see Figure 2.1).

For turbo decoding with the SOVA algorithm, a decoding depth of $\delta = 30$ was chosen
to ensure proper performance [37]. Our investigation showed that the SOVA iterative decoding algorithm produced diminishing returns. Results are only presented for up to four decoding iterations because it was found that additional iterations did not produce further improvements. As noted in section 3 in the discussion of the iterative STBC detectors, the number of iterations indicated that provide maximum BER rate performance are based on simulation results. In a practical implementation, it is necessary to determine the optimal number of iterations required that will provide the best performance based on the trade-off with computational complexity.

### 4.1 Coding Gain with Turbo Codes in an OSTBC System

Remarkable performance improvement is shown in Figure 4.1 when turbo coding is incorporated into an existing OSTBC system. In the CML-based colored noise receivers with coding, a coding gain exceeding 8dB is achievable at a BER of 10^{-3} when compared to the uncoded CML-based colored noise receiver. This 8dB coding gain implies a power savings of requiring less than 1/6th the transmit power as that in an uncoded system in order to achieve a BER of 10^{-3}. The cost of this gain, however, is attributed to a loss in efficiency as the code rate is R = 1/3.

For the OSTBC systems operating under a white noise assumption with turbo coding, results indicate marginal improvement is achieved. This is due to the initial high BER in the uncoded system which, in many cases, does not allow the turbo code to convergence.

### 4.2 Coding Gain with Turbo Codes in a DSTM System

Recall the high error floor encountered for the CML-based colored noise DSTM receivers in Figure 3.3. The results in Figure 4.2 show that this high error floor can be mitigated with the use of turbo coding. Coding gains exceeding 8dB are achieved at a BER of 10^{-2} when compared to the uncoded CML-based colored noise DSTM receivers. In addition, these
results also validate the metric derived in section 2.5 as suitable for use with the DSTM receivers considered.

4.3 Effect of Constraint Length on Turbo-coded Coherent System Performance

In this section, the constraint length of the turbo code is varied in the coded CML-based MIMO wireless system. The constraint length affects turbo code performance by increasing the distance properties of the code. This in turn allows for improved error correction. Note that this improvement is expected only in the higher SNR region where the distance properties dominate performance [37].

The results in figure 4.3 confirm this theory as increasing the constraint length produces further performance gains in the high SNR region only. These performance gains are measured as approximately 0.5dB in SNR for the 16-state turbo code when compared to the 4-state turbo code. This improvement comes at a great cost, however, as decoding complexity is approximately quadrupled as SOVA decoding requires four times the number of state comparisons in finding the branch and path metrics when compared to a 4-state code.

4.4 Effect of Puncturing on Turbo-coded Coherent System Performance

The rate $R = 1/3$ turbo code investigated thus far results in a significant amount of bandwidth expansion that may not be feasible in certain applications (e.g. speech applications). For these cases, a higher rate turbo code may be desirable. The penalty for a higher rate code is a loss in BER performance. To this end, the performance of a $R = 1/2$ punctured turbo code is investigated.

A comparison was made between a punctured and unpunctured turbo coded system
employing the 2x2 CML-based colored noise OSTBC receivers (1 Iter CML). For puncturing, the turbo code encoder multiplexer punctures odd and even parity bits from RSC1 and RSC2, respectively. Figure 4.4 shows an approximate 4dB loss in coding gain at a BER of $10^{-4}$ for the punctured turbo code (1 Iter CML and 4 Iter TC-Punctured) when compared to the unpunctured code (1 Iter CML and 4 Iter TC-Punctured) using four iterations of turbo decoding. Although this loss is quite significant, (1 Iter CML and 4 Iter TC-Punctured) still provides a substantial coding gain of 5dB over an uncoded system (1 Iter CML and Uncoded) at a relatively high BER of $10^{-3}$.

### 4.5 Comparing Turbo-Coded Coherent and Noncoherent System

The relative difference in performance between a coherent and non-coherent STBC system with turbo codes is compared here. In this comparison, a minimal amount of training data of $K = 2$ and $K = 1$ was used for the coherent and noncoherent colored noise STBC receivers, respectively. Figure 4.5 shows a relative decrease of 3dB in SNR performance for the non-coherent system as opposed to the coherent system. This is not unexpected, as the noncoherent scheme can afford this penalty by not requiring channel estimates.
Figure 4.1. Performance of a concatenated 4-state turbo-coded and OSTBC systems with one strong interferer.
Figure 4.2. Performance of a concatenated 4-state turbo-coded and DSTM systems with one strong interferer.
4.5 Comparing Turbo-Coded Coherent and Noncoherent System

Figure 4.3. Effect of constraint length on turbo-coded and OSTBC system performance with one strong interferer.
Figure 4.4. Performance comparison between a 4-state punctured and unpunctured turbo-coded systems with one strong interferer.
Figure 4.5. Performance comparison between turbo-coded coherent and noncoherent systems with one strong interferer.
Chapter 5

Conclusion and Future Work

This thesis evaluated the performance of recently proposed CML-based coherent and non-coherent space-time receivers for unknown interference suppression.

We first showed that conventional space-time block code (STBC) receivers designed under the assumption of white noise operate poorly in an environment with strong interference. Thus the need for proper interference suppression is confirmed. Results with the proposed CML-based STBC receivers revealed that accurate noise and channel estimation is possible and can be used to mitigate this interference. For achieving greater gains, turbo channel codes were serially concatenated with the STBC receivers. Favorable results with coding gains exceeding 8dB at a low BER of $10^{-2}$ were achieved for the turbo-coded system over the uncoded system.

There is room for much future work. A possible research direction is to perform a formal analysis on the novel suboptimum reliability metric developed for noncoherent DSTM soft-information transfer to an outer channel code. The particular emphasis would be to derive analytical expressions that explain its effect on turbo code design and performance. Another research direction, which is natural extension of this work, is to develop a prototype of this system using either a digital signal processor (DSP) or a field-programmable gate array (FPGA). Thus, the feasibility of implementation of this algorithm, in terms of its computational complexity, can be determined for product development.
References


References


References


