Iterative Detection/Equalization with Frequency Domain Filtering for MIMO Channels

by

JIE LIN

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Master of Applied Science

Ottawa-Carleton Institute for Electrical and Computer Engineering
Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada

May 2004

Abstract

In this thesis, we proposed a novel iterative receiver for multiple-input multiple-output (MIMO) channels, which is comprised of a bank of multi-user frequency domain detectors and soft-input soft-output decoders. The filter coefficients of the MIMO detectors are determined according to the MMSE criterion under the assumption that the *a priori* probabilities of the transmitted symbols are arbitrary. Frequency domain filtering provide much lower implementation complexity than time-domain filtering, while being at least as effective at eliminating intersymbol and multiple access interference. Simulation results show that the proposed MIMO detector has excellent performance when used in space division multiple access (SDMA) environment. Using multiple antennas in the receiver not only gives diversity gain, but also allows the system to handle more users without any performance loss. With QPSK the system can support twice as many users as the number of receive antennas, and with 16-QAM can support the same number of users as receive antennas. We also investigate the effects of symbol interleaving and bit interleaving. Symbol interleaving is less complex than bit interleaving, but when QPSK is used, the performance of symbol interleaving is essentially the same as that of bit interleaving.

Table of Contents

List of Figures

List of Symbols

$\underline{a}$	Message word
n_a	Number of bits in one message symbol
N_a	Number of message symbols
$\underline{c}$	Code word
n_c	Number of bits in one code symbol
N_c	Number of code symbols/code word
L_c	Constraint length of the convolutional code
$\underline{d}$	Interleaved symbols
n_d	Number of bits in one interleaved symbol
N_d	Number of interleaved symbols per code word
$\underline{v}$	Transmitted symbols
N	Number of transmitted symbols per transmitted block
$\underline{r}$	Received samples
M	Number of points in signal constellation
L	Channel delay spread (in symbols)
$\underline{h}$	Channel coefficients
$\underline{w}$	AWGN noise
N_0	Single-sided noise power spectral density
N_S	Number of transmit antennas
N_A	Number of received antennas
$\underline{v}^{(s)}$	User *s* transmitted symbols
$\underline{r}^{(a)}$	Antenna *a* received samples
$\underline{h}^{(s,a)}$	The impulse response of the channel between user *s* and receive antenna *a*
$w_n^{(a)}$	AWGN noise at receive antenna *a*
$R_k^{(a)}$	The DFT at frequency *k* of the samples received from antenna *a*

$\sigma_{h,l}^2$	$\mathrm{E}\left[\left\|h_l\right\|^2\right]$
$H_k^{(s,a)}$	The DFT of the discrete-time channel impulse response between user *s* and receive antenna *a*
$V_k^{(s)}$	The DFT of transmitted samples for user *s* at frequency *k*
$W_k^{(a)}$	The DFT of AWGN at frequency *k* and antenna *a*
$P_k^{(s,a)}$	Frequency domain filter coefficient
$q_0^{(s)}$	Another filter coefficient for bias term
$b_n^{(s)}$	An estimation anti-biasing term
$z_n^{(s)}$	The detector output for user *s* at time *n*
$J^{(s)}$	Mean square value of the difference between the actual and desired equalizer outputs for user *s*
$\overline{R}_k^{(a)}$	The expected value of $R_k^{(a)}$
$\sigma_{v,n,s}^2$	The variance of the n^{th} transmitted symbol from user *s*
$\overline{\sigma}_{v,s}^2$	The time-averaged variance of all the symbols transmitted by user *s*
$\mu_n^{(s)}(d)$	The mean of $z_n^{(s)}$ conditioned on $d_n^{(s)} = d$
$\sigma_{z,n,s}^2$	The variance of $z_n^{(s)}$ conditioned on $d_n^{(s)} = d$
$\overline{\sigma}_{z,s}^2$	The time-average variance of $z_n^{(s)}$ conditioned on $d_n^{(s)} = d$
Γ	$N_A \times N_A$ matrix
$\underline{\gamma}$	Metric values
$\underline{\chi}$	De-interleaver metric values
$\underline{\hat{a}}$	Estimation of the source symbols
$\underline{\overline{v}}$	Mean value of the feedback symbols
$\underline{\psi}$	Soft feedback information
$\underline{\phi}$	Interleaved soft feedback information
$\mathrm{B_i(d)}$	The i^{th} bit of code symbol d
T	Symbol period

List of Acronyms

APP	*A Posteriori* Probability
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
dB	DeciBel
LE	Linear Equalizer
DFE	Decision Feedback Equalizer
DFT	Discrete Fourier Transform
MLSE	Maximum Likelihood Sequence Estimation
IC	Interference Canceller
ISI	Intersymbol Interference
MAI	Multiple Access Interference
MAP	Maximum *A Posteriori* probability
MMSE	Minimum Mean Square Error
QPSK	Quadrate Phase Shift Keying
QAM	Quadrature Amplitude Modulation
TCM	Trellis Coded Modulation
FFT	Fast Fourier Transform
IFFT	Inverse FFT
SNR	Signal-to-Noise Ratio
MIMO	Multiple-Input Multiple-Output
MCU	Metric Calculation Unit

Chapter 1 Introduction

In recent years, cellular telephony, personal computing and the Internet have become more and more popular. These trends suggest that high-speed, high-capacity and high-quality wireless communication will grow to be a major area of the telecommunication services. A fundamental challenge in transmitting at high data rates over radio channels is to overcome intersymbol interference (ISI) in data transmission, which is caused by multipath propagation.

Recently multiple-input-multiple-output (MIMO) systems have attracted a great deal of attention because the system capacity can be significantly improved if multiple transmit and receive antennas are used. In MIMO systems, the received data is composed of four parts: the desired transmitted data, the AWGN, the sum of the interference (ISI) from neighboring symbols and multiple access interference (MAI) from other users. The receiver has to detect each user's signals in the presence of MAI and ISI. The focus of this thesis is on the implementation of a MIMO detector to eliminate the ISI and MAI.

1.1 Techniques for Combating Intersymbol Interference

There are various equalization methods for combating intersymbol interference. Linear equalization (LE), decision feedback equalization (DFE), maximum likelihood sequence estimation (MLSE) and maximum *a posteriori* probability (MAP) equalization are popular techniques.

Linear equalizers use a simple linear filter to suppress the ISI while maximizing the signal-to-noise ratio (SNR). However, they tend to perform poorly in ISI channels with spectral nulls, because compensating for the large attenuation caused by the nulls results in large noise enhancement at the output of the equalizer. Decision feedback equalizers consist of two filters, a feed forward filter and a feedback filter. They have similar complexity to linear equalizers, but can provide a significant improvement in performance. MLSE is an ever more effective equalization technique, which is based on modeling the channel as a finite state machine. An alternative to MLSE is maximum *a posteriori* (MAP) equalization, which provides soft output that can be passed to the channel decoder. However, the MLSE and MAP equalizers require extremely high computational complexity. When the transmitted symbol alphabet or the ISI delay spread is large, the number of states in the trellis for these equalizers becomes enormous. This complexity limits the application of these equalizers.

In 1995, Douillard *et al.* introduced the concept of turbo equalization for systems that employ error control coding at the transmitter [1]. The receiver consists of an equalizer to combat the ISI and a decoder to decode the error correcting channel code, connected as shown in Figure 1-1. When a block of channel samples is received, the equalizer tries its best to eliminate the ISI, under the assumption that no priori knowledge about the probability distribution of the transmitted data is available. The equalizer output is passed to the decoder where the channel code is decoded. After decoding the code the receiver should have a much better idea of the transmitted symbols. Information is therefore passed back to the equalizer, where it is exploited to better equalize the originally received block of channel samples. This process is repeated several times in an

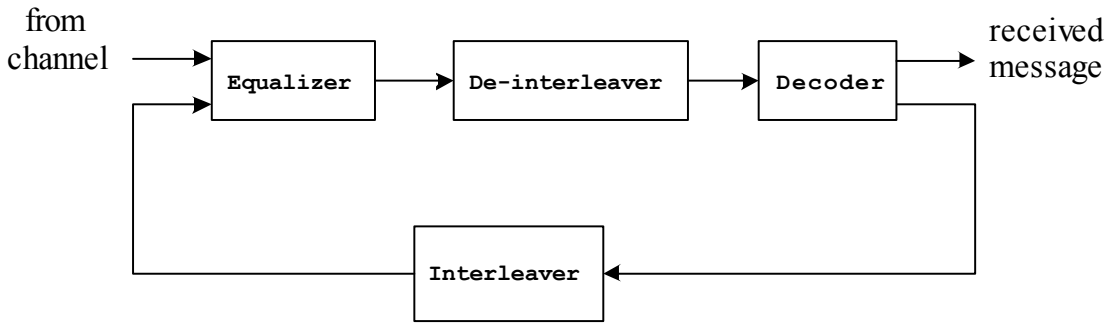


Figure 1-1. Structure of a turbo equalizer.

iterative fashion, and with each iteration the feedback information becomes more reliable, so the equalizer becomes more effective, leading in turn to ever more reliable feedback information. Therefore, with each iteration the system bit error rate (BER) is reduced, until ideally the performance of an ISI-free system is achieved.

The turbo equalizer proposed by Douillard *et al.* was based on the use of a MAP equalizer for the equalizer and a convolutional code for the channel code, decoded with a MAP decoder [2]. Although very effective, the complexity of the MAP equalizer was a serious drawback, so research shifted to a search for lower-complexity equalizers that could be used in a turbo equalization framework.

In 1997, an approach based on using an interference canceller (IC) instead of the MAP equalizer was proposed by Glavieux, Laot and Labat [3]. An IC consists of a matched filter which maximizes the SNR but increases the ISI. Then based on knowledge of the transmitted symbols, a replica of the interference is generated locally at the receiver and is subtracted from the filter output. With this structure it is possible to perfectly eliminate the ISI. Of course, since knowledge of the transmitted symbols is not available, turbo equalization is used. In the first iteration a linear equalizer is used instead

of an IC, because there is no knowledge of the transmitted symbols at this time. The equalizer output is passed to the MAP decoder where the convolutional code is decoded. After decoding the code the receiver should have a much better idea of the transmitted symbols. This information can be used by an IC to re-equalize the received samples, more effectively eliminating the ISI. Although very effective, IC based turbo equalization is nonetheless noticeably worse than MAP-based turbo equalization in terms of BER performance. This is because the IC filter coefficients are derived under the assumption that perfect knowledge of all past and future transmitted symbols are available. Since this assumption is severely violated, particularly during the early iterations, performance is limited.

Tuchler *et al.* generalized the concept of the IC when they proposed minimum mean square error equalization using *a priori* information [4][5]. Traditional linear equalizers are designed under the assumption that the transmitted symbols have an equal *a priori* probability distribution. Unlike traditional linear equalizers, with Tuchler's equalizer the transmitted symbols can have an arbitrary probability distribution. The probability distribution of the transmitted symbols is provided as output from the decoder, so the equalizer filter coefficients change with each iteration. On the first iteration, when no a priori information is available the equalizer is identical to a traditional LE. However, with each additional iteration the filter coefficients slowly adapt to the improving reliability of the feedback information, becoming more and more like the filter coefficients of a traditional IC. By overcoming the limitations imposed by the faulty assumption of the IC, Tuchler's turbo equalizer can provide performance much closer to that of MAP-based turbo equalization, while still maintaining much lower complexity.

Although much less complex than the MAP equalizer, both the traditional IC and Tuchler's equalizer use linear filtering. The potential barrier with these techniques is that their complexity grows linearly or worse with channel delay spread. When transmitting at high data rates over radio channels, the delay spread (in symbols) grows linearly with the signaling rate. As a result, the number of equalizer taps must be increased to compensate for the increased ISI spread. For example, the delay spread is about 15 μs over a typical urban area multipath channel [6][7], so if the symbol rate were 200 kBaud, the ISI spread would be $L = 3$ symbols. If the symbol rate were increased 20 times, to 4 MBaud, the ISI spread would increase to $L = 60$ symbols. Since the number of taps in the linear filter must be a few times larger than the ISI spread, the complexity becomes prohibilive for channels with long delay spread. So broadband channels may call for new low-complexity solutions.

Frequency domain equalization (FDE) has been proposed [7][8] for broadband systems. By exploiting the fact that convolution in the time domain is multiplication in the frequency domain, FDE has relatively low complexity growth with an increase of channel delay spread. The use of FDE in conjunction with turbo equalization was proposed by Tuchler and Hagenauer [9][10]. The authors re-derived the implementation of the minimum mean square error equalizer using *a priori* information in the frequency domain instead of the time domain. With the approach it is possible to realize all the benefits of turbo equalization, but with complexity that is only loosely dependent on the channel delay spread.

1.2 MIMO Systems

High data rate wireless access is demanded by many applications. Traditionally, higher data rate transmission requires more bandwidth. However, it is often impractical or sometimes very expensive to increase bandwidth because of spectral limitations. In this case, using multiple transmit and receive antennas for spectrally efficient transmission is an alternative solution. By using multiple transmit and receive antennas to form a MIMO channel the system capacity can be significantly improved with no bandwidth expansion or increase in transmitted power. It has been shown that MIMO channels can offer enormous channel-capacity gains over independent narrow-band channels, which have the same total bandwidth [11]-[13].

In MIMO communication systems, multiple users use the same time and

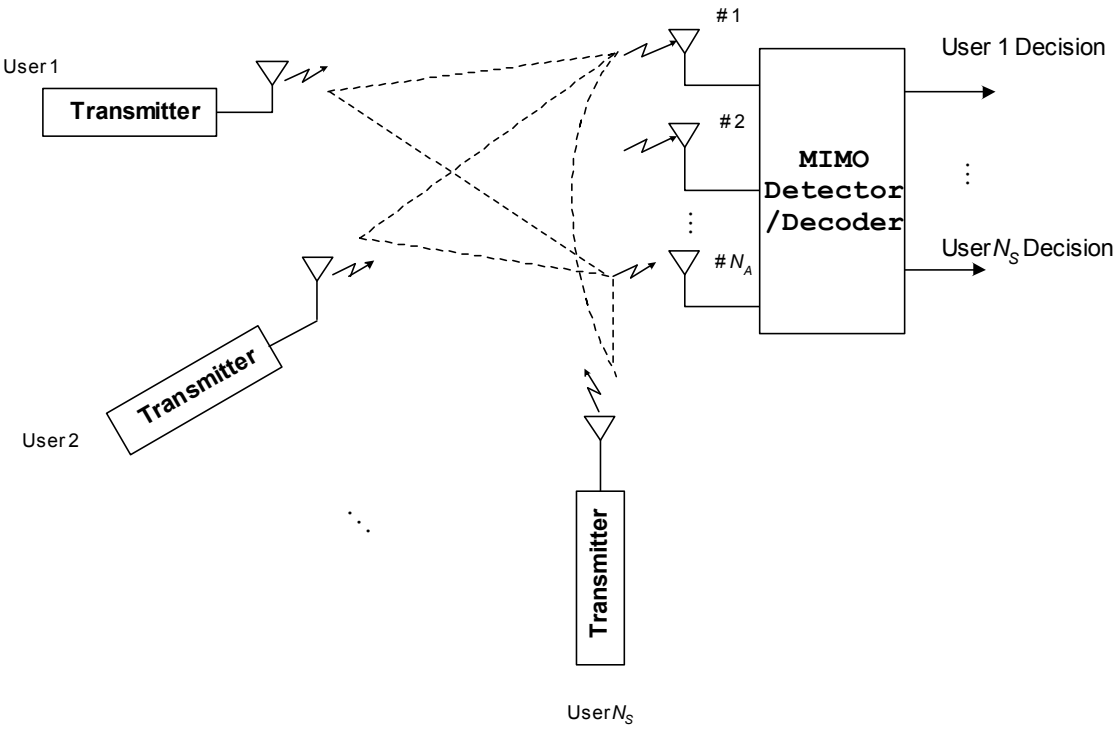


Figure 1-2. Block diagram of the MIMO system.

frequency slots to send information sequences. Each user creates multiple access interference (MAI) for all other users. The logical structure of a system uplink is shown in Figure 1-2. Each user uses a single antenna to transmit its signal, and the receiver uses multiple antennas to detect the multiple signals in the presence of MAI and ISI.

Abe and Matsumoto proposed an iterative ISI and MAI cancellation technique for frequency-selective MIMO channels in the time domain [14]. The iterative receiver, as shown in Figure 1-3, consists of a bank of multi-user signal detectors followed by soft-input-soft-output decoders whose outputs are used as feedback information for improving the detector's estimates in the next iteration. This process is repeated in an iterative manner to enhance signal-detection performance. It is shown that with this scheme the system capacity can be significantly improved by using multiple transmit and receive antennas.

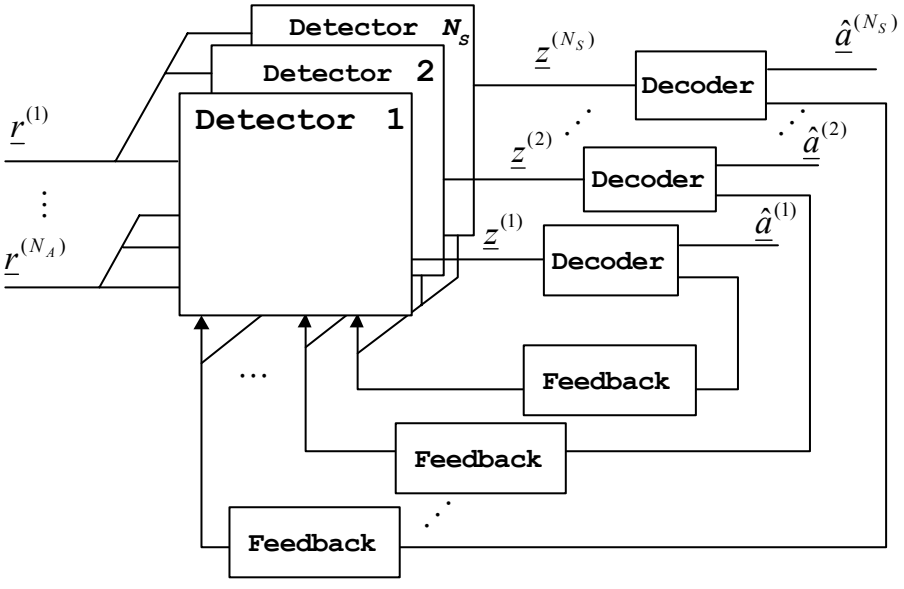


Figure 1-3. Block diagram of iterative detection.

1.3 Thesis Motivation

IC-based turbo equalization was originally proposed by Glavieux *et al.* in 1997 [3]. Tuchler *et al.* proposed minimum mean square error equalization using *a priori* information which was originally implemented in time domain [4], and later extended to the frequency domain [9]. The frequency domain approach provides further complexity reductions compared to the time domain. These approaches are all for single-input-single-output channels.

Abe and Matsumoto proposed a turbo detector/equalizer for MIMO channels [14]. This approach was shown to be very effective at eliminating MAI and ISI, but because time domain processing is used it is not suitable for channel with long delay spreads.

In this thesis we attempt to overcome the limitations of Abe and Matsumoto's scheme by implementing the detector/equalizer in the frequency domain. We therefore introduce turbo frequency domain detection/equalization for MIMO channels. The overall structure of the receiver is the same as in Figure 1-3, but the implementation of the detectors is drastically different, since we use frequency domain filters instead of time domain. The resulting system is even more effective at eliminating the interference, while at the same time requiring lower implementation complexity particularly for channels with long delay spreads.

1.4 Thesis Contributions

The main contributions of this thesis are:

1. Introducing turbo frequency domain detection/equalization for frequency selective MIMO channels.

2. Deriving the optimal detector filter coefficients by minimizing the mean square error. For simplicity, we assume all users in the system are synchronous and have the same average power and the fading between different users and different antennas is uncorrelated.
3. Evaluating the receiver performance by computer simulation. We investigate system performance with different channel models, different modulation schemes, and compare symbol interleaving and bit interleaving. The effects of using different modulation schemes for different users are also investigated. The simulation result shows the MIMO detector can completely cancel MAI and ISI.

1.5 Organization of the thesis

This thesis is organized as follows. Chapter 2 contains a complete description of the system model, including the transmitter, MIMO channel, and receiver. Chapter 3 shows the result of computer simulations to verify the effectiveness of the algorithm and performance comparison under different cases. Finally, conclusions and possibilities for future work are presented in Chapter 4. A detailed description of the MIMO turbo equalizer can be found in the appendix.

Chapter 2 Description of the System Model

Signals transmitted through Multiple-Input-Multiple-Output (MIMO) channels suffer from MAI, multipath propagation and additive noise. For such channels, we present the complete structure of an iterative receiver with frequency domain filtering, which comprises of a bank of multi-user detectors and soft-input-soft-output (SISO) decoders. The soft decoders' outputs are used as feedback information for improving the detector's estimates in the next iteration.

All users transmit message sequence simultaneously in the same frequency band, so each user creates MAI for all the other users. The logical structure of a system uplink is shown in Figure 2-1. Each user uses a single antenna to transmit its signal and the receiver uses multiple antennas to detect the multiple signals.

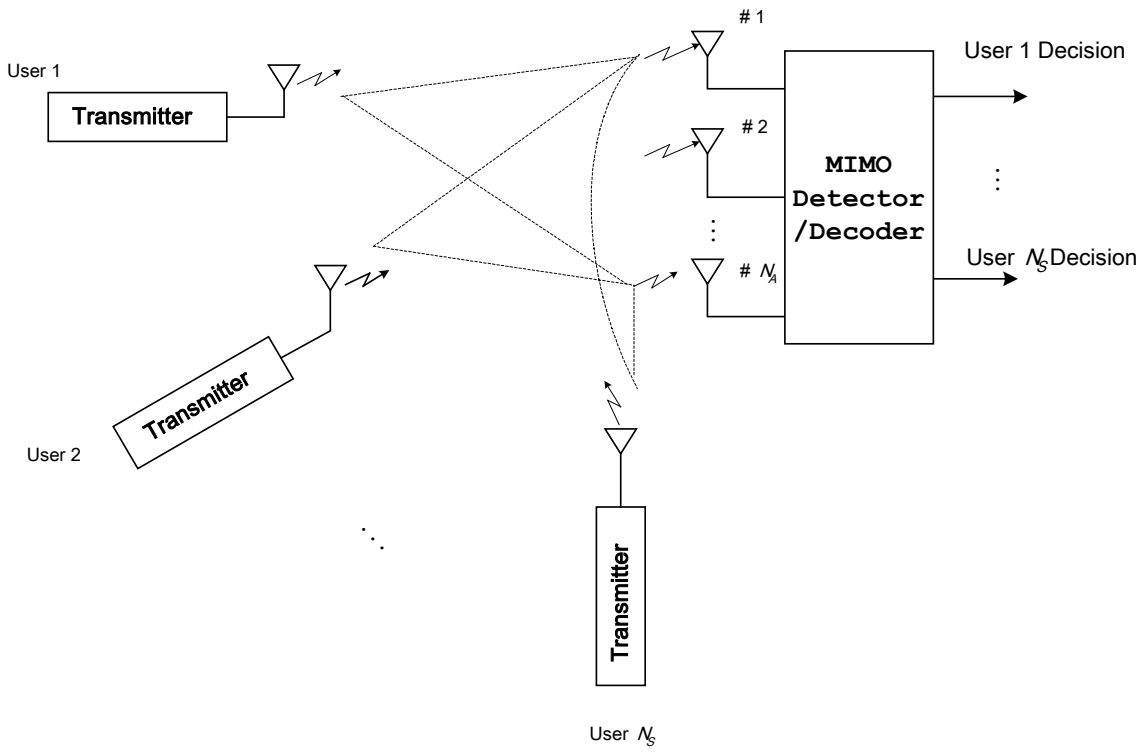


Figure 2-1. Block diagram of the MIMO system.

The number of remote users is N_S and the number of receiver antennas at the base station is N_A. The message sequence from each user is encoded by a convolutional encoder, then the coded symbols are interleaved and modulated. So the symbols are mapped onto complex symbols and transmitted over MIMO channels where distortion occurs.

The MIMO channel output is a sum of all the received signals transmitted from all the users. Then the output signals are passed through a MIMO detector/equalizer to detect multiple users' signals. Finally, the equalized samples are fed into the corresponding MAP decoder and decision device to estimate the transmitted messages. Each component of this system is described in detail in this chapter.

2.1 Transmitter

Figure 2-2 shows the block diagram of a transmitter. It includes four parts: the message source, the convolutional encoder, the interleaver and the signal mapper.

Figure 2-2. Structure of the transmitter.

2.1.1 Message Source

In this system there are N_S users simultaneously transmitting in the same frequency band. Each user transmits a message word of N_a symbols, with n_a bits per message

symbol. The message word from one of the users is denoted by $\underline{a} = [a_0, a_1, ..., a_{N_a-1}]$, where $a_n \in \{0, 1, ..., 2^{n_a} - 1\}$ is the n^{th} message symbol. All message symbols within a word are independent of each other, and the message word from each user is independent of the words from the other users. The *a priori* probability distribution of the message symbols is given by

$$\Pr\{a_n = a\} = \frac{1}{2^{n_a}} \text{ for } a \in \{0, 1, ..., 2^{n_a} - 1\}. \tag{2-1}$$

2.1.2 Convolutional Encoder

The message word for each user is first encoded with a convolutional code. Convolutional codes are a basic type of error control code, which can protect digital data from errors. At each encoder clock cycle, one message symbol of n_a message bits is passed into the encoder, which produces one code symbol of n_c bits. The resulting code word consists of N_c symbols, with each symbol composed of n_c bits, and is denoted by $\underline{c} = [c_0, c_1, ..., c_{N_c-1}]$. The n^{th} code symbol is $c_n \in \{0, 1, ..., 2^{n_c} - 1\}$. The code word length is $N_c = N_a + (L_c - 1)$ symbols, where L_c is the constraint length of the code. The additional $(L_c - 1)$ symbols result from the termination of the trellis while the encoder is forced back to the all-zero state.

2.1.3 Channel Interleaver

The convolutional encoder is followed by an interleaver whose aim is to reorder the encoded data, so that error bursts occurring in the channel or caused by the detector in the

receiver can be dispersed by a de-interleaver at the receiver. A random interleaver is used in this system, which reorders the code symbols using a psuedo-randomly selected permutation, which is assumed to be known both at the transmitter and the receiver. Interleavers can operate on a bit-by-bit basis or a symbol-by-symbol basis. For symbol interleavers, the bits within each symbol aren't changed, just the order of the symbols is rearranged. With bit interleavers, the order of each individual bit is changed.

As will be discussed in the following, symbol interleaving has the advantages of lower implementation complexity at the receiver, but, as will be shown in Chapter 3, better performance may be possible with bit interleaving. However, regardless of whether bit of symbol interleaving is used, it is important that each user uses a different permutation.

2.1.3.1 Symbol interleaving

The interleaved sequence consists of N_d symbols, with n_d bits in each symbol. It is denoted by $\underline{d} = \left[d_0, d_1, ..., d_{N_d-1}\right]$, with the n^{th} symbol given by $d_n = c_{\text{IL}[n]}$, where IL[·] is the interleaver mapping, which is a psuedo-randomly selected permutation of the numbers $\{0,1,...,N_d - 1\}$. For symbol interleaving, the number of bits per symbol at the input and output of the interleaver must be the same, so $n_d = n_c$. This also implies $N_d = N_c$.

2.1.3.2 Bit interleaving

To implement bit interleaving, the output of encoder needs to be multiplexed to a sequence of bits by a symbol-to-bit conversion, then interleaved by a bit interleaver and demultiplexed into symbol by a bit-to-symbol conversion, so the output of bit interleaver is still symbol-based. Figure 2-3 shows a block diagram of a bit interleaver.

The input of the symbol-to-bit converter is $\underline{c}$ and the output is $\underline{c}' = \left[c'_0, c'_1, ..., c'_{N'_c-1}\right]$, which consists of $N_{c'} = N_c n_c$ bits. The m^{th} code bit is $c'_m \in \{0,1\}$ for $m \in \{0,1,..., N_{c'} - 1\}$. The corresponding relationship is represented by $c'_{nn_c+i} = c_{n,i}$, where $c_{n,i}$ is the i^{th} bit of the n^{th} code symbol.

The interleaver reorders $\underline{c}'$ on a bit-by-bit basis. The interleaved sequence consists of $N_{d'} = N_{c'}$ bits. It is denoted by $\underline{d}' = \left[d'_0, d'_1, ..., d'_{N_d-1}\right]$, with the m^{th} code bit given by $d'_m = c'_{\text{IL}[m]}$, where $d'_m \in \{0,1\}$, $m \in \{0,1,..., N_{d'} - 1\}$ and $\text{IL}[m] \in \{0,1,..., N_{d'} - 1\}$.

After interleaving, the sequence $\underline{d}'$ is demultiplexed into symbols by a bit-to-symbol converter. The output $\underline{d} = \left[d_0, d_1, ..., d_{N_d-1}\right]$ consists of N_d symbols, with n_d bits per symbol, where $d_n \in \{0,1,..., 2^{n_d} - 1\}$ is the n^{th} symbol. Clearly $N_d n_d = N_{d'} = N_{c'} = N_c n_c$. The corresponding relationship between the input and the output of the bit-to-symbol conversion is represented by $d_{n,i} = d'_{nn_d+i}$, where $d_{n,i}$ is the i^{th} bit of the n^{th} interleaved symbol.

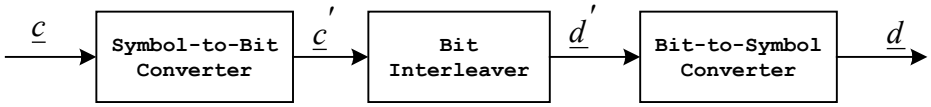


Figure 2-3. Block diagram of bit interleaver.

2.1.4 Signal Mapper

The interleaved symbols, $\underline{d}$, are mapped to points in an M-ary signal constellation,

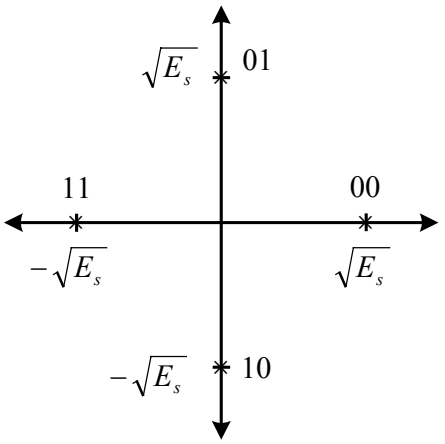
where $M = 2^{n_d}$ is the number of points in the constellation. When symbol interleaving is used, we have $n_d = n_c$, so the size of the signal constellation must be matched to the number of bits per code symbol produced by the encoder. When bit interleaving is used it is not necessary to match the encoder and constellation size, giving greater design flexibility. The signal mapper output is $\underline{v} = [v_0, v_1, ..., v_{N-1}]$, where $N = N_d$ is the number of transmitted symbols per message word. Each symbol is given by

$$v_n = \mathrm{SM}[d_n]. \tag{2-2}$$

where SM[·] is the mapping between the digital symbols and the points in the signal constellation, which depends on the modulation scheme being used.

In Chapter 3, we investigate the use of QPSK ($M = 4$) and trellis-coded 16-QAM ($M = 16$). The corresponding signal constellations along with the signal mapping scheme are shown in Figure 2-4 and 2-5, respectively. The average transmitted energy per symbol is

$$E_s = \frac{1}{M}\sum_{d=0}^{M-1}\left|\mathrm{SM}[d]\right|^2. \tag{2-3}$$



$$\mathrm{SM}[d] = \begin{cases} \sqrt{E_s}, & \text{if } d = 00 \\ j\sqrt{E_s}, & \text{if } d = 01 \\ -j\sqrt{E_s}, & \text{if } d = 10 \\ -\sqrt{E_s}, & \text{if } d = 11 \end{cases} \tag{2-4}$$

Figure 2-4. Signal constellation for QPSK with Gray mapping.

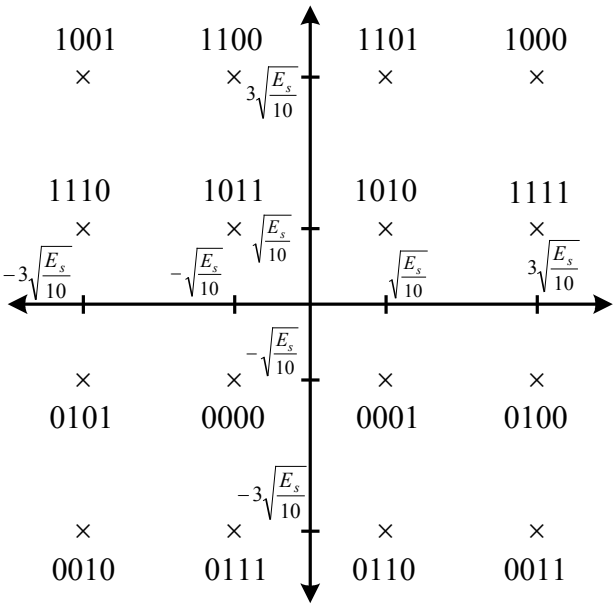


Figure 2-5. Signal constellation for 16-QAM with set partitioning.

Priori to transmission, each data block is appended with a length-L symbol cyclic prefix in order to remove the effect of interblock interference, where L= maximum channel impulse response length (in symbols). At the receiver, the cyclic prefix is removed before processing the block, so for every (N+L) received symbols, only N symbols are processed. The length of the cyclic prefix is chosen to equal the maximum expected length of the channel impulse response. The cyclic prefix at the beginning of a data block is simply a repetition of the L symbols that are at the end of the data block, so $v_{-n} = v_{N-n}$ for all $n \in \{1, 2, ..., L\}$. The benefit of using the cyclic prefix is that it prevents intersymbol interference from the previous block. Furthermore, it makes the linear convolution of the transmitted symbols with the channel impulse response that occur during transmission to be equivalent to a circular convolution. This facilitates frequency domain equalization, as discussed in Section 2.3.

2.2 Channel Model

2.2.1 ISI channel

The combination of the transmitter's pulse shaping filter and modulator, the analogue multipath fading channel, and the receiver's demodulator, matched filter (matched to the transmitted pulse shape), and symbol-rate signal sampler can be modeled as a discrete-time ISI channel.

The channel output at time *n* can be expressed as:

$$r_n = \sum_{l=0}^{L} h_l v_{n-l} + w_n \ , \tag{2-5}$$

where v_{n-l} is the symbol transmitted at time $n-l$, $\{h_l\}$ is the discrete-time channel impulse response, and w_n is additive white Gaussian noise (AWGN).

For any $l \in \{0,1,...,L\}$, where L is the channel delay spread, h_l has a complex Gaussian distribution, with a mean of zero and a variance of

$$\sigma_{h,l}^2 = \mathrm{E}\left[|h_l|^2\right] \tag{2-6}$$

Although the channel impulse response is randomly distributed according to the channel delay profile $\left\{\sigma_{h,l}^2\right\}$, it is assumed to remain constant for the duration of the transmission of a block of data. This corresponds to a slow fading channel. Coupled with the slow fading assumption, it is further assumed that perfect channel estimation is possible, so $\{h_l\}$ are known at the receiver.

The AWGN has a single-sided noise power spectral density of N_0, so

$$\mathrm{E}[w_n] = 0 \quad \text{and} \quad \mathrm{E}\left[w_{n-m}^* w_n\right] = N_0 \delta_m \ , \tag{2-7}$$

where $\delta_m = \begin{cases} 1 & \text{if } m = 0 \\ 0 & \text{if } m \neq 0 \end{cases}$ is the Kronecker delta function.

2.2.2 MIMO channel

To increase the system capacity, multiple transmit and receive antennas are used to form a MIMO channel. All users transmit their signals sharing the same time and frequency slot, so each user creates multiple access interference (MAI) for all other users. As a result, the receiver has to detect each user's signals in the presence of MAI and ISI.

The single link between one user and one receive antenna suffer from Rayleigh frequency selective fading. The fading between different users and different antennas is uncorrelated, and we assume all users have equal average received signal power.

Figure 2-6 shows a block diagram of a MIMO channel with N_S transmit antennas and N_A receive antennas. For $s \in \{1,2,...,N_S\}$, user s transmits the symbols $\underline{v}^{(s)} = \left[v_0^{(s)}, v_1^{(s)}, ..., v_{N-1}^{(s)}\right]$, along with its cyclic prefix. For $a \in \{1,2,...,N_A\}$, the n[th] sample from receiving antenna *a* at time *n* is

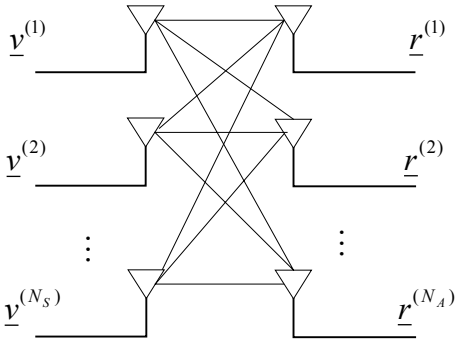


Figure 2-6. Block diagram of a MIMO channel.

$$r_n^{(a)} = \sum_{s=1}^{N_S} \sum_{l=0}^{L} h_l^{(s,a)} v_{n-l}^{(s)} + w_n^{(a)} \,, \tag{2-8}$$

where $w_n^{(a)}$ is the AWGN (with power spectral density N_0) , and $\underline{h}^{(s,a)} = \left[h_0^{(s,a)}, h_1^{(s,a)}, ..., h_L^{(s,a)}\right]$ is the impulse response of the channel between user s and antenna a.

2.3 Detector

The receiver must generate an estimate of the transmitted message word, $\hat{\underline{a}}^{(s)}$, for each user, based on the received sample sequences from all the N_A antennas, $\left[\underline{r}^{(1)}, \underline{r}^{(2)}, ..., \underline{r}^{(N_A)}\right]$. For each user, this task can be separated into two stages, a detector and a decoder. The detector tries to generate an estimate $\underline{z}^{(s)}$ of the transmitted symbol sequence $\underline{v}^{(s)}$, by combining the received samples in a manner that simultaneously suppresses the ISI and the MAI. The decoder uses the detector output to decode the convolutional code. Figure 2-7 shows the detectors/decoders of N_s users. Note that there is one pair for each user.

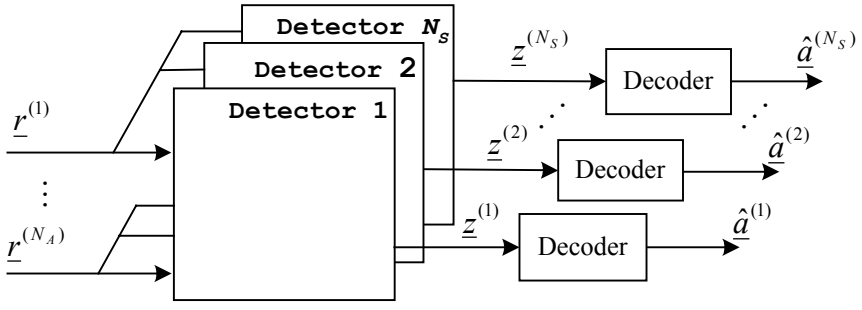


Figure 2-7. Block diagram of MIMO detector.

To suppress the interference, the detector linearly filters the signals from each of the antennas and combines the output of the filters. For computational efficiency, the filtering is done in the frequency domain, using the Fast Fourier Transform (FFT) and the inverse FFT (IFFT).

Figure 2-8 shows a block diagram of the detector for user *s*. After discarding the cyclic prefix, the FFT is applied to of the block of N samples from each antenna, giving, for antenna *a*,

$$R_k^{(a)} = \sum_{n=0}^{N-1} r_n^{(a)} e^{-j2\pi\frac{kn}{N}}, \qquad \forall \ k \in \{0,1,2,...,N-1\} \tag{2-9}$$

where N is the FFT block length, which is the same as the length of the block of transmitted symbols. Because of the presence of the cyclic prefix, the data symbols $\{v_n^{(s)}\}$ can be assumed to be periodic, $v_m^{(s)} = v_{m+N}^{(s)}$, for $m \in \{-L+1,-L+2,...,-2,-1\}$. Then the linear convolution of the channel is equivalent to a circular convolution, which is the essential to the proper functioning of the frequency domain filtering. In the discrete frequency domain, we can get

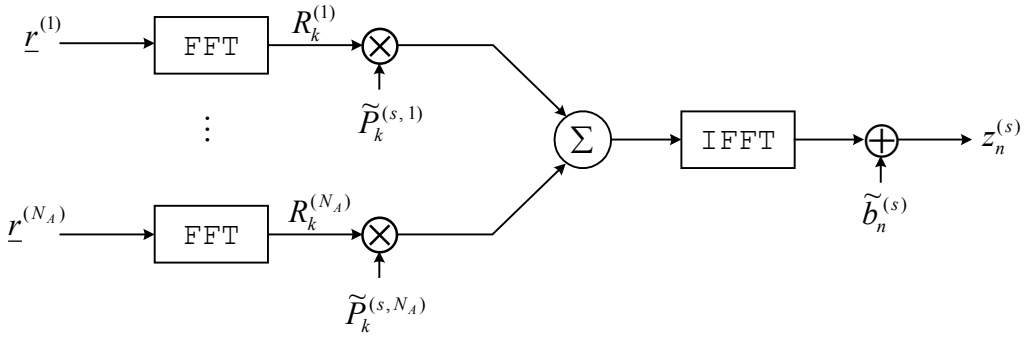


Figure 2-8. Block diagram of the detector.

$$R_k^{(a)} = \sum_{s=1}^{N_S} H_k^{(s,a)} V_k^{(s)} + W_k^{(a)} \tag{2-10}$$

is the DFT of the samples at frequency k received from antenna a, where

$$H_k^{(s,a)} = \sum_{n=1}^{N-1} h_n^{(s,a)} e^{-j2\pi\frac{kn}{N}} \tag{2-11}$$

is the DFT of the discrete-time channel impulse response between the s^{th} user and the a^{th} receiver antenna,

$$V_k^{(s)} = \sum_{n=0}^{N-1} v_n^{(s)} e^{-j2\pi\frac{kn}{N}} \tag{2-12}$$

is the DFT of $\left\{v_n^{(s)}\right\}$ for user s at frequency k, and

$$W_k^{(a)} = \sum_{n=0}^{N-1} w_n^{(a)} e^{-j2\pi\frac{kn}{N}} \tag{2-13}$$

is the DFT of the noise samples $\left\{w_n^{(a)}\right\}$ at frequency k.

Filtering is performed in the frequency domain by multiplying $R_k^{(a)}$ by the filter coefficient $\widetilde{P}_k^{(s,a)}$. The filtered frequency domain components from each antenna are then combined additively, and an inverse FFT is applied to the result. After adding an estimation anti-biasing term, $\widetilde{b}_n^{(s)}$, the detector output for user s at time n is

$$z_n^{(s)} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} \widetilde{P}_k^{(s,a)} R_k^{(a)} e^{j2\pi\frac{kn}{N}} + \widetilde{b}_n^{(s)} . \tag{2-14}$$

The filter coefficients and bias are determined according to the minimum mean square error (MMSE) criterion, so that $z_n^{(s)}$ is the MMSE estimate of $v_n^{(s)}$ based on the received samples. When calculating the coefficients and bias, it is further assumed that all the transmitted symbols, $\left\{ d_n^{(s)} \middle| \forall n, s \right\}$, have a known but arbitrary *a priori* probability

distribution, $\Pr\left\{ d_n^{(s)} = d \right\}$, that potentially differs for all *n* and *s*. The significance of this assumption will be explained in Section 2.5.

A detailed derivation of the MMSE optimal filter coefficients is given in Appendix A, where it is shown that the filter output can also be expressed as

$$z_n^{(s)} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} P_k^{(s,a)} \left(R_k^{(a)} - \overline{R}_k^{(a)} \right) e^{j2\pi \frac{kn}{N}} + q_0^{(s)} \overline{v}_n^{(s)} \tag{2-15}$$

where $\overline{R}_k^{(a)} = \mathrm{E}\left[R_k^{(a)} \right]$ is the expected value of $R_k^{(a)}$, which depends on *a priori* probability distribution of the transmitted symbols, and

$$q_0^{(s)} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} P_k^{(s,a)} H_k^{(s,a)} \,. \tag{2-16}$$

According to Equation (2-15), the detector can be implemented as shown in Figure 2-9. The source for $\overline{R}_k^{(a)}$ will be discussed in Section 2.5.

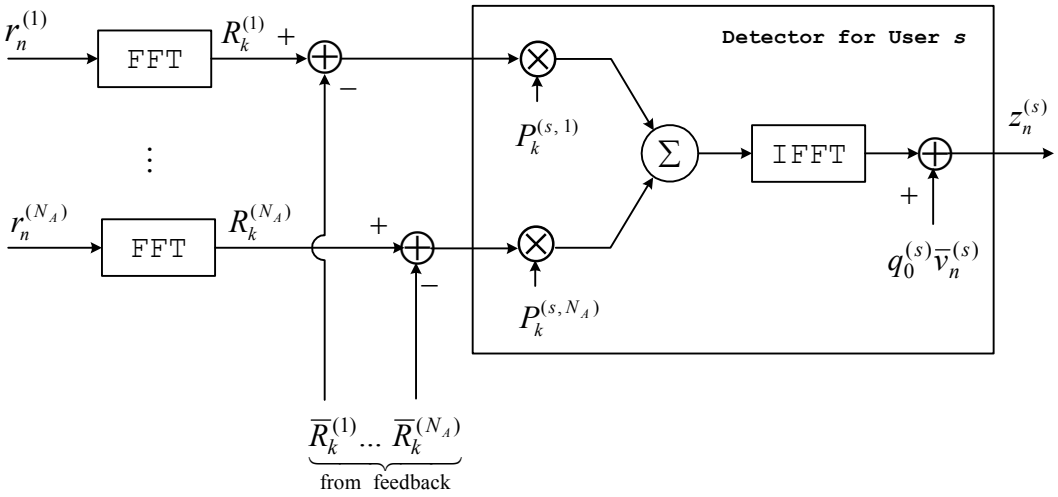


Figure 2-9. Block diagram of the implementation of detector.

The filter coefficients $\left\{P_k^{(s,a)}\right\}$ in Equation (2-15) are found by solving N sets of linear equations, one set for each FFT point. Each set consist of the N_A equations, given by

$$\begin{bmatrix} \Gamma_k^{(1,1)} & \Gamma_k^{(1,2)} & \cdots & \Gamma_k^{(1,N_A)} \\ \Gamma_k^{(2,1)} & \Gamma_k^{(2,2)} & \cdots & \Gamma_k^{(2,N_A)} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_k^{(N_A,1)} & \Gamma_k^{(N_A,2)} & \cdots & \Gamma_k^{(N_A,N_A)} \end{bmatrix} \begin{bmatrix} P_k^{(1,2)} \\ P_k^{(s,2)} \\ \vdots \\ P_k^{(s,N_A)} \end{bmatrix} = \begin{bmatrix} H_k^{(s,1)*} \\ H_k^{(s,2)*} \\ \vdots \\ H_k^{(s,N_A)*} \end{bmatrix}, \tag{2-17}$$

for all $k \in \{0,1,...,N-1\}$, where $*$ denotes complex conjugate, and

$$\Gamma_k^{(b,a)} = \sum_{s=1}^{N_S} H_k^{(s,a)} H_k^{(s,b)*} \overline{\sigma}_{v,s}^2 + N_0 \delta_{a-b} . \tag{2-18}$$

The quantity $\overline{\sigma}_{v,s}^2$ is given by

$$\overline{\sigma}_{v,s}^2 = \frac{1}{N} \sum_{n=0}^{N-1} \sigma_{v,n,s}^2 , \tag{2-19}$$

where

$$\sigma_{v,n,s}^2 = \mathrm{E}\left[\left| v_n^{(s)} - \overline{v}_n^{(s)} \right|^2 \right] = \sum_{d=0}^{M-1} \left| \mathrm{SM}[d] \right|^2 \Pr\left\{ d_n^{(s)} = d \right\} - \left| \overline{v}_n^{(s)} \right|^2 \tag{2-20}$$

is the variance of the n^{th} transmitted symbol from user s, based on the arbitrary a priori probability distribution of $d_n^{(s)}$. $\overline{\sigma}_{v,s}^2$ is then the time-averaged variance of all the symbols transmitted by user s. Although from a theoretical standpoint it would be better to use $\sigma_{v,n,s}^2$ instead of $\overline{\sigma}_{v,s}^2$ in Equation (2-18), doing so would greatly increase the system complexity since the filter coefficients would be different for each n. Frequency domain filtering with time-variant filter coefficients is unduly complicated. However, as the simulation result presented in Chapter 3 indicate, the suboptimality introduced by using $\overline{\sigma}_{v,s}^2$ is not a great concern.

2.4 Decoder

Since we use a convolutional encoder at the transmitter, a MAP decoder is needed at the receiver to decode the convolutional code. A MAP decoder calculates the *a posteriori* probabilities (APP's) of the message symbols, $\Pr\{a_n = a|\underline{z}\}$ for all $n \in \{0,1,...,N_a - 1\}$, which are fed into a decision device, where hard decisions about the message symbols are made. To calculate the APP's, the MAP decoding algorithm needs the branch metrics, $\chi_n(c) = \Pr\{z_n | c_n = c\}$. So a metric calculation unit (MCU) is needed before MAP decoder. Between the MCU and the MAP decoder, a de-interleaver is also needed to restore the proper order of the symbols, corresponding to the changes made by the interleaver at the transmitter. The deinterleaver helps disperse error burst at the output of the detector, making them appear as random errors to the decoder. Figure 2-10 shows the block diagram of the decoder.

Figure 2-10. Block diagram of the decoder.

2.4.1 Metric Calculation Unit (MCU)

The branch metrics used by the MAP decoder are based on the likelihood function of the detector output, $z_n^{(s)}$, conditioned on the transmitted symbols, $d_n^{(s)}$. The metrics calculated by the MCU are

$$\gamma_n^{(s)}(d) = f\left(z_n \middle| d_n^{(s)} = d\right), \tag{2-21}$$

for each $d \in \{0,1,...,M-1\}$ and $n \in \{0,1,...,N-1\}$. That is, a set of M metrics are calculated for each detector output sample.

Under the widely-used assumption that the residual interference in $z_n^{(s)}$ can be modeled with a Gaussian distribution, the branch metrics can be expressed as

$$\gamma_n^{(s)}(d) = \frac{1}{\pi\overline{\sigma}_{z,s}^2}\exp\left\{-\frac{1}{\overline{\sigma}_{z,s}^2}\left| z_n^{(s)} - \mu_n^{(s)}(d) \right|^2\right\}, \tag{2-22}$$

where $\mu_n^{(s)}(d)$ is the mean of $z_n^{(s)}$ under the hypothesis that $d_n^{(s)} = d$, and $\overline{\sigma}_{z,s}^2$ is the time-averaged variance. From Appendix A, the mean is given by

$$\mu_n^{(s)}(d) = \mathrm{E}\left[z_n^{(s)} \middle| d_n^{(s)} = d\right] = q_0^{(s)}\,\mathrm{SM}[d], \tag{2-23}$$

and the variance by

$$\overline{\sigma}_{z,s}^2 = \frac{1}{N}\sum_{n=0}^{N-1}\mathrm{E}\left[\left|z_n^{(s)} - \mu_n^{(s)}(d)\right|^2 \middle| d_n^{(s)} = d\right] = q_0^{(s)*}\left(1 - \overline{\sigma}_{v,s}^2 q_0^{(s)}\right), \tag{2-24}$$

where $q_0^{(s)}$ is given by Equation (2-16). The time-averaged variance is used instead of the instantaneous variance solely for computational expedience.

2.4.2 Metric De-interleaver

Because the order of the code symbols is changed by the interleaver at the transmitter, a de-interleaver is needed at the receiver to restore the proper order. This provides the benefit of breaking any error bursts produced by the detector. The design of the deinterleaver depends on whether symbol or bit interleaving was used at the transmitter.

2.4.2.1 Symbol De-interleaver

When symbol interleaving is used at the transmitter, the operation of the deinterleaver is quite simple. Only the order of the metrics needs to be changed, not the actual values. The output of the de-interleaver is

$$\chi_{\mathrm{IL}[n]}(d) = \gamma_n(d), \tag{2-25}$$

where $\{\gamma_n(d)\}$ are the metrics produced by the MCU, and IL[·] is the interleaver mapping used at the transmitter.

2.4.2.2 Bit De-interleaver

When bit interleaving is used, the de-interleaving is more complicated. The metrics must first be converted from symbol-based to bit-based metrics. These bit metrics are then deinterleaved, and then converted back to symbol-based metrics. Figure 2-11 shows the required operations.

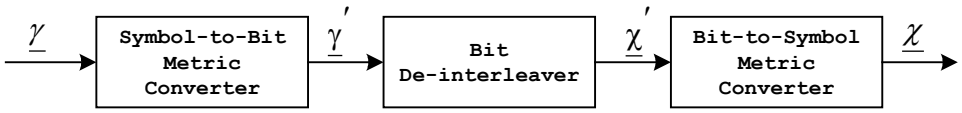


Figure 2-11. Structure of a bit de-interleaver.

1) Symbol-to-bit metric converter

The output of metric calculation unit (MCU) is the branch metrics $\gamma_n(d)$, which are symbol-based and must therefore be converted to bit metrics. The bit metric of the i^{th} bit of the n^{th} transmitted symbol $d_{n,i}$ from any user, is

$$\gamma_{n,i}(k) = f\left(z_n \middle| d_{n,i} = k\right) = \frac{\Pr\left\{ d_{n,i} = k \middle| z_n \right\} f(z_n)}{\Pr\{d_{n,i} = k\}}. \tag{2-26}$$

Note that

$$\Pr\left\{d_{n,i} = k \middle| z_n\right\} = \sum_{d \middle| B_i[d]=k} \Pr\left\{d_n = d \middle| z_n\right\}, \tag{2-27}$$

where $B_i[d]$ is the i^{th} bit of d, so the summation is over all value of $d \in \{0,1,...,M-1\}$ that have a value of $k \in \{0,1\}$ in the i^{th} bit position. Since

$$\Pr\left\{d_n = d \middle| z_n\right\} = \frac{f\left(z_n \middle| d_n = d\right)\Pr\{d_n = d\}}{f(z_n)}, \tag{2-28}$$

we have

$$\begin{aligned}
\gamma_{n,i}(k) &= \sum_{d \middle| B_i[d]=k} f\left(z_n \middle| d_n = d\right) \frac{\Pr\{d_n = d\}}{\Pr\{d_{n,i} = k\}} \\
&= \sum_{d \middle| B_i[d]=k} \gamma_n(d) \frac{\Pr\{d_n = d\}}{\Pr\{d_{n,i} = k\}} \\
&= \sum_{d \middle| B_i[d]=k} \gamma_n(d) \prod_{\substack{j=0 \\ j \neq i}}^{n_d - 1} \Pr\{d_{n,j} = B_j[d]\}.
\end{aligned} \tag{2-29}$$

The output of symbol-to-bit converter is $\gamma'_{nn_d+i}(k) = \gamma_{n,i}(k)$.

For example, when QPSK is used, $n_d = 2$ and $M = 2^{n_d} = 4$, so the symbol-to-bit metric converter calculates

$$\begin{aligned}
\gamma'_{nn_d+0}(0) &= \gamma_{n,0}(0) = \gamma_n(00)\Pr\{d_{n,1} = 0\} + \gamma_n(01)\Pr\{d_{n,1} = 1\} \\
\gamma'_{nn_d+0}(1) &= \gamma_{n,0}(1) = \gamma_n(10)\Pr\{d_{n,1} = 0\} + \gamma_n(11)\Pr\{d_{n,1} = 1\} \\
\gamma'_{nn_d+1}(0) &= \gamma_{n,1}(0) = \gamma_n(00)\Pr\{d_{n,0} = 0\} + \gamma_n(10)\Pr\{d_{n,0} = 1\} \\
\gamma'_{nn_d+1}(1) &= \gamma_{n,1}(1) = \gamma_n(01)\Pr\{d_{n,0} = 0\} + \gamma_n(11)\Pr\{d_{n,0} = 1\}
\end{aligned} \tag{2-30}$$

for every $n \in \{0,1,...,N-1\}$.

2) De-interleaver

The input to the de-interleaver is $\underline{\gamma}'$, which correspond to the interleaved code bits at the transmitter. The effect of interleaving is reversed with

$$\chi'_{\mathrm{IL}[m]}(k) = \gamma'_m(k) \tag{2-31}$$

for $k \in \{0,1\}$ and $m \in \{0,1,...,n_d N_d - 1\}$, where IL[·] is the interleaver mapping used at the transmitter.

3) Bit-to-symbol metric converter

Because the MAP decoder requires symbol-based metrics, a bit-to-symbol metric converter is needed to convert the de-interleaved bit metrics to code symbol branch metrics,

$$\chi_n(c) = f\left(z_n \middle| c_n = c\right) = \frac{\Pr\left\{c_n = c \middle| z_n\right\} f(z_n)}{\Pr\{c_n = c\}}. \tag{2-32}$$

If we assume the code bits are transmitted independently, which is approximated, if the interleaver is sufficiently large, we have

$$\Pr\left\{c_n = c \middle| z_n\right\} = \prod_{i=0}^{n_c-1} \Pr\left\{c_{n,i} = B_i[c] \middle| z_n\right\}. \tag{2-33}$$

Since

$$\Pr\left\{c_{n,i} = B_i[c] \middle| z_n\right\} = \frac{f\left(z_n \middle| c_{n,i} = B_i[c]\right) \Pr\left\{c_{n,i} = B_i[c]\right\}}{f(z_n)}, \tag{2-34}$$

we can get

$$\chi_n(c) = \frac{f(z_n)}{\Pr\{c_n = c\}} \left[\prod_{i=0}^{n_c-1} f\left(z_n \middle| c_{n,i} = B_i[c]\right) \right] \prod_{i=0}^{n_c-1} \frac{\Pr\left\{c_{n,i} = B_i[c]\right\}}{f(z_n)}$$

$$= \prod_{i=0}^{n_c-1} f\left(z_n \middle| c_{n,i} = B_i[c]\right). \tag{2-35}$$

The symbol metric can therefore be expressed as

$$\chi_n(c) = \prod_{i=0}^{n_c-1} \chi_{n,i}(B_i[c]), \tag{2-36}$$

where $\chi_{n,i}(k) = \chi'_{nn_c+i}(k)$.

2.4.3 MAP decoder

Maximum *a posteriori* (MAP) probability decoding provides an approach to decoding convolutional codes which is based on minimizing the probability of a message symbol error. To make decisions, the decoder needs to first calculate the *a posteriori* probabilities (APP's) of the message symbols. The APP that message symbol a_n is equal to some $a \in \{0,1,\ldots,2^{n_a}-1\}$ is $\Pr\left\{ a_n = a \middle| \underline{z} \right\}$. These can be calculated efficiently using the algorithm proposed by Bahl *et al.* [2].

Once the APP's for a_n have been calculated, a decision can be made by choosing $\hat{a}_n = \arg\max_a \Pr\left\{a_n = a \middle| \underline{z}\right\}$, which results in the most probable transmitted symbol being selected.

2.5 Iterative Detection

The system as described in Section 2.4 is capable of partially suppressing the interference, but because there is still considerable residual interference at the output of the detectors the BER performance is quite poor. Much better result can be obtained if iterative

processing is enabled. This is accomplished by using the receiver structure shown in Figure 2-12. Output from the decoders is after some feedback processing, fed back to the detectors where it is used to better cancel the interference in the second and subsequent iterations of the receiver.

The feedback processing is discussed in this section. It is composed of a feedback interleaver and a soft symbol generator. In addition to calculating the a posteriori probabilities of the message symbols, $\Pr\left\{a_n^{(s)} = a \middle| \underline{z}^{(s)}\right\}$, the MAP decoder is configured to also calculate the a posteriori probabilities of the code symbols, $\psi_n^{(s)}(c) = \Pr\left\{ c_n^{(s)} = c \middle| \underline{z}^{(s)} \right\}$ for all $c \in \{0,1,\ldots,2^{n_c}-1\}$ and $n \in \{0,1,\ldots,N_c-1\}$. This output of the MAP decoder is passed through the feedback interleaver to the soft-symbol generator, which calculates the expected value of the transmitted symbols, $\bar{v}_n^{(s)} = \mathrm{E}\left[v_n^{(s)}\right]$. These soft symbols are

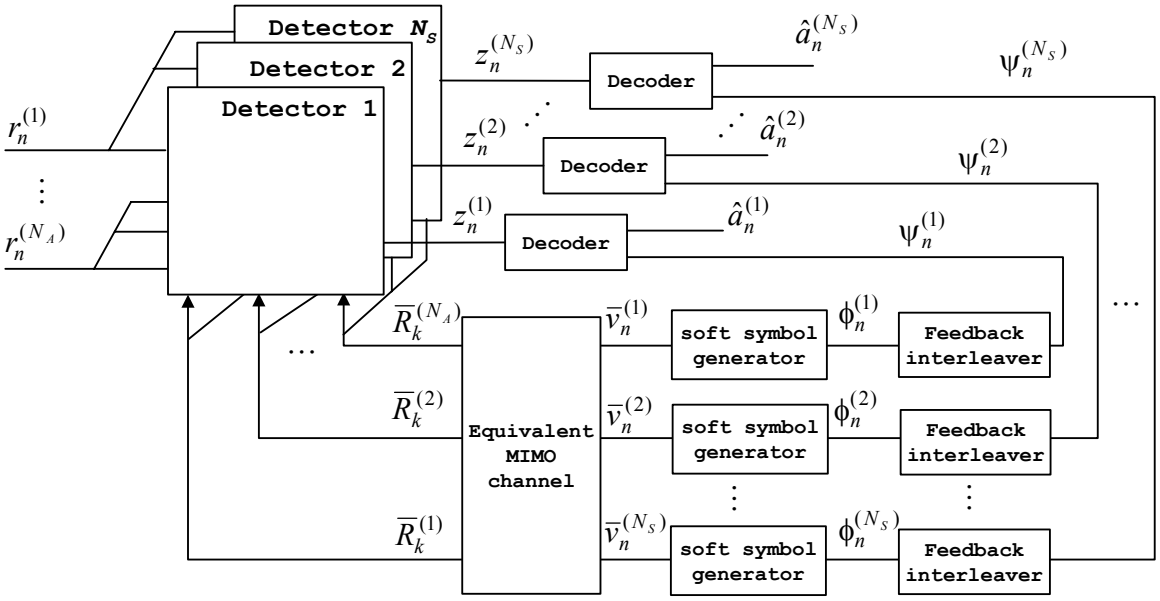


Figure 2-12. Block diagram of turbo detector/equalizer.

converted to the frequency domain using an FFT. The frequency domain soft symbols from all the users are combined by an equivalent frequency domain MIMO channel to generate a replica of the interference. All these operations are described in greater detail in the following.

2.5.1 Feedback Interleaver

Like the de-interleaver discussed in Section 2.4.2, the feedback interleaver works on probabilities, and can be categorized as either symbol-based or bit-based, depending on what was used at the transmitter.

2.5.1.1 Symbol Feedback Interleaver

The input of the interleaver is $\psi_n(\bullet)$, which is reordered to produce

$$\phi_n(d) = \psi_{\mathrm{IL}[n]}(d) \tag{2-38}$$

for all $n \in \{0,1,\ldots,N_d - 1\}$, where $\mathrm{IL}[\cdot]$ is the interleaver mapping used at the transmitter.

2.5.1.2 Bit Feedback Interleaver

The structure of the feedback bit interleaver is similar to the bit de-interleaver discussed in Section 2.4.2.2. It still has three parts, and Figure 2-13 shows the structure.

Figure 2-13. Structure of a feedback bit interleaver.

The output of the MAP decoder is the soft information, $\psi_n(c) = \Pr\{c_n = c|\underline{z}\}$, which is symbol-based and must be converted to bit-based.

The output of the i^{th} bit of the n^{th} code, $c_{n,i}$ is

$$\Pr\{ c_{n,i} = k|\underline{z}\} = \sum_{c|B_i[c]=k} \Pr\{c_n = c|\underline{z}\} = \sum_{c|B_i[c]=k} \psi_n(c). \tag{2-39}$$

where $B_i[c]$ is the i^{th} bit of c, so the summation is over all values of $c \in \{0,1,\ldots,2^{n_c}-1\}$ that have a value of $k \in \{0,1\}$ in the i^{th} bit position.

Note that

$$\Pr\{ c_{n,i} = k|\underline{z}\} = \frac{f\{\underline{z}|c_{n,i} = k\}\Pr\{c_{n,i} = k\}}{f(\underline{z})}, \tag{2-40}$$

we have

$$\begin{aligned}\Pr\{ c_{n,i} = k\} &= f(\underline{z}) \frac{\Pr\{ c_{n,i} = k|\underline{z}\}}{f\{\underline{z}|c_{n,i} = k\}} \\ &= K \frac{\sum_{c|B_i[c]} \psi_n(c)}{f\{ \underline{z}|c_{n,i} = k\}} \\ &= K \frac{\sum_{c|B_i[c]} \psi_n(c)}{\chi_{n,i}(k)}.\end{aligned} \tag{2-41}$$

Then

$$\psi_{n,i}(k) = \sum_{c|B_i[c]=k} \frac{\psi_n(c)}{\chi_n(c)} \prod_{\substack{j=0 \\ j\neq i}}^{n_c-1} \chi_{n,j}(B_j[c]). \tag{2-42}$$

The output of symbol-to-bit converter is $\psi'_{nn_c+i}(k) = \psi_{n,i}(k)$.

The input of feedback interleaver is $\underline{\psi}'$ which is reordered by

$$\phi'_n(k) = \psi'_{\mathrm{IL}[n]}(k) \tag{2-43}$$

for $n \in \{0,1,...,n_c N_c - 1\}$, $k \in \{0,1\}$ and $\mathrm{IL}[n] \in \{0,1,...,n_d N_d - 1\}$, IL[·] is the same interleaver mapping used in the transmitter.

The Bit-to-symbol converter converts soft information from bit-based to symbol-based. Let

$$\phi_{n,i}(k) = \phi'_{nn_c+i}(k), \tag{2-44}$$

then the symbol information can be expressed as:

$$\phi_n(d) = \prod_{i=0}^{n_d-1} \phi_{n,i}(B_i[d]) \qquad \text{for } n \in \{0,1,... N_d - 1\}. \tag{2-45}$$

2.5.2 Soft-Symbol generator

The estimated transmit symbols can be obtained by using a soft-symbol generator. The output is:

$$\bar{v}_n = \sum_{d=0}^{M-1} \mathrm{SM}[d] \Pr\{d_n = d\} = \sum_{d=0}^{M-1} \mathrm{SM}[d] \phi_n(d), \tag{2-46}$$

where SM[·] is the symbol mapping described in Section 2.1.4, and $\phi_n(d) = \Pr\{d_n = d\}$ is the *a priori* probability of each possible symbol, as determined by the posteriori probabilities calculated by the MAP decoder.

The $\bar{v}_n$ can be considered as an averaged value of all possible transmitted symbols at time *n*. If $\Pr\{d_n = d\}$ is close to 1 for one value of *d*, then $\bar{v}_n$ will be close to one of the points in the signal constellation. On the other hand, if $\Pr\{d_n = d\}$ is roughly the same for all *d*, the $\bar{v}_n$ will be close to zero.

2.5.3 Equivalent MIMO channel

Figure 2-14 shows the diagram of equivalent MIMO channel. The expected value of the FFT of the transmitted symbols for user s, $\overline{V}_k^{(s)}$, multiplies $H_k^{(s,a)}$ to compose the feedback signal $\overline{R}_k^{(a)}$, s can change from 1 to N_S. The number of feed back signals is N_A. $H_k^{(s,a)}$ is the FFT of the discrete-time channel impulse response between s^{th} user and a^{th} receiver antenna. Using equation (A-24), we can calculate $\overline{R}_k^{(a)}$ as

$$\overline{R}_k^{(a)} = \sum_{s=1}^{N_S} H_k^{(s,a)} \mathrm{E}\left[V_k^{(s)}\right] + \mathrm{E}\left[W_k^{(a)}\right], \tag{2-47}$$

Since $\mathrm{E}\left[W_k^{(a)}\right] = 0$ (the AWGN have zero mean),

we can get

$$\overline{R}_k^{(a)} = \sum_{s=1}^{N_S} H_k^{(s,a)} \overline{V}_k^{(s)} . \tag{2-48}$$

We define

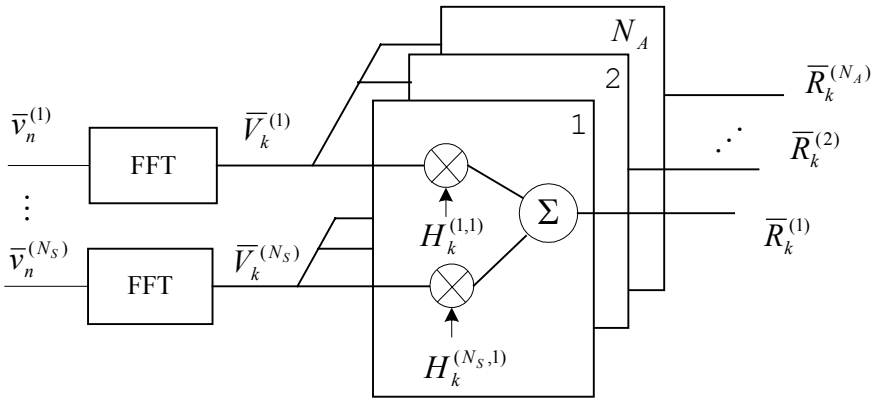


Figure 2-14. Block diagram of equivalent MIMO channel.

$$\overline{V}_k^{(s)} = \mathrm{E}\left[V_k^{(s)}\right] = \sum_{n=0}^{N-1} \mathrm{E}\left[v_n^{(s)}\right] e^{-j2\pi\frac{kn}{N}} = \sum_{n=0}^{N-1} \overline{v}_n^{(s)} e^{-j2\pi\frac{kn}{N}} \tag{2-49}$$

as the expected value of the FFT of the transmitted symbols for user *s*, with

$$\overline{v}_n^{(s)} = \mathrm{E}\left[v_n^{(s)}\right] = \sum_{d=0}^{M-1} \mathrm{SM}[d] \Pr\{d_n^{(s)} = d\} \tag{2-50}$$

being the expected value of $v_n^{(s)}$, with the expectation performed over the arbitrary probability distribution.

For each detector the input signals are same, the number is N_A and equal $R_k^{(a)} - \overline{R}_k^{(a)}$, $a \in \{0,1,...,N_A\}$. Different input signals multiply different coefficients and combine the outputs of the detector.

2.6 Complexity comparison

In this section we compare the complexity of the time and frequency domain turbo detector/equalizers for MIMO channels. The complexity involved in performing both the filtering operations and calculating the filter coefficients are considered.

2.6.1 Complexity comparison of time-domain and frequency-domain MIMO detector

In Shufen Sun's thesis, she analyses the complexity of a single-user/single-antenna system with time domain and frequency domain filtering [15]. By extending her analysis, we can determine the complexity of the MIMO detector. Table 2-1 shows the complexity required by the *p*-filter (feed forward) for a filter with 3*L* taps, where *L* is the length of the

channel impulse response. This table shows the number of real multiplications (RM) and real additions (RA) required to produce one output symbol from one of the detectors (i.e. for one user), for one iteration. It is based on the number of complex multiplications (CM) and complex additions (CA). The complexity for the *q*-filter (feed back) is shown in Table 2-2, for one filter output for one antenna. The total complexity is shown in Table 2-3 for all the filters and for N_i iterations. Tables 2-4, 2-5, 2-6, contain the corresponding results for the frequency domain system.

		RM	**RA**
p-filter	$(3L+1)\,N_A \times$ **(CM)**	$4\,(3L+1)\,N_A$	$2\,(3L+1)\,N_A$
	$3L\,N_A \times$ **(CA)**		$6L\,N_A$
Calculate $z_n^{(s)}$	$(N_A-1) \times$ **(CA)**		$2\,(N_A-1)$
Total		$4\,(3L+1)\,N_A$	$12L\,N_A + 4N_A - 2$

Table 2-1. Per-symbol complexity of the *p*-filter with $3L + 1$ taps of the time- domain MIMO detector.

		RM	**RA**
q-filter	$(L+1)N_S \times$ **(CM)**	$4\,(L+1)N_S$	$2\,(L+1)N_S$
	$LN_S \times$ **(CA)**		$2\,LN_S$
Calculate $\bar{r}_n^{(a)}$	$(N_S\text{-}1) \times$ **(CA)**		$2\,(N_S - 1)$
Subtract $r_n^{(a)} - \bar{r}_n^{(a)}$	$1 \times$ **(CA)**		2
Total		$4\,(L+1)N_S$	$4LN_S + 4N_S$

Table 2-2. Per-symbol complexity of the *q*-filter with $L+1$ taps of the time-domain MIMO detector.

	RM	**RA**
p-filter	$4\,(3L+1)\,N_A N_S\,N_i$	$(12LN_A+4N_A-2)\,N_S\,N_i$
q-filter	$4\,(L+1)\,N_S N_A\,(N_i-1)$	$(4LN_S+\ 4N_S)\,N_A(N_i-1)$
Total	$4\,(4LN_{\mathrm{i}}+2N_{\mathrm{i}}-L-1)\,N_A N_S$	$4\,N_A N_S(4LN_i+\ 2N_i-L-1)-\ 2N_S N_i$

Table 2-3. Complexity of the time domain MIMO detector per block.

		RM	**RA**
p-filter	$N_A \times$ **(CM)**	$4N_A$	$2N_A$
combine	$(N_A-1) \times$ **(CA)**		$2(N_A-1)$
IFFT	$1 \times$ IFFT	$2\log_2 N$	$3\log_2 N$
Calculate $z_n^{(s)}$	$2 \times$ **(RM)**	2	
	$1 \times$ **(CA)**		2
Total		$4N_A + 2\log_2 N+2$	$4N_A+3\log_2 N$

Table 2-4. Per-symbol complexity of the *p*-filter of the frequency-domain MIMO detector.

		RM	**RA**
FFT	$Ns \times$ FFT	$2N_S \log_2 N$	$3N_S \log_2 N$
q-filter	$N_A\,N_S \times$ **(CM)**	$4N_A\,N_S$	$2N_A\,N_S$
Calculate $\overline{R}_k^{(a)}$	$(N_S-1)N_A \times$ **(CA)**		$2(N_S-1)N_A$
Subtract $R_k^{(a)}-\overline{R}_k^{(a)}$	$N_A \times$ **(CA)**		$2N_A$
Total		$2N_S \log_2 N+4N_A\,N_S$	$(3\log_2 N+4N_A)N_S$

Table 2-5. Per-symbol complexity of the *q*-filter of the frequency -domain MIMO detector.

	RM	RA
$r_n^{(a)} \rightarrow R_k^{(a)}$	$2N_A \log_2 N$	$3\ N_A \log_2 N$
p-filter	$(4N_A+2\log_2 N+2)N_S N_i$	$(4N_A+3\log_2 N)N_S N_i$
q-filter	$(2N_S \log_2 N+4N_A N_S)\ (N_i-1)$	$(3\log_2 N+4N_A)N_S\,(N_i-1)$
Total	$2N_A \log_2 N -2N_S \log_2 N- 4N_A N_S$ $+ 8N_A N_S N_i +4N_S N_i \log_2 N+2N_S N_i$	$6\ N_S N_i \log_2 N + 8N_A N_S N_i + 3N_A \log_2 N- 4N_S N_A$ $- 3\ N_S \log_2 N$

Table 2-6. Complexity of the frequency domain MIMO detector.

From Table 2-3, we find the item $16LN_iN_AN_S$ dominates the number of real multiplies for the time-domain detector and from Table 2-6 we find that $8N_AN_SN_i$ + $4N_SN_i\log_2 N$ dominates for the frequency-domain detector. Because $16LN_A$ will be much bigger than $8N_A+4\log_2 N$ when channel delay spread is large, the complexity of time-domain MIMO detector is much higher than the frequency-domain MIMO detector.

Let us also consider a numerical example. According to the Table 2-3 and 2-6, when $L = 4$, $N = 1024$, $N_A = 3$, $N_S = 3$ and $N_i = 5$, the complexity of the time-domain MIMO detector is 3060 RM and 3030 RA per symbol. Compare this with the complexity of frequency domain MIMO detector, which is 954 RM and 1224 RA per symbol. We can see the complexity of time-domain MIMO detector is much higher than frequency-domain MIMO detector. If L becomes larger, the complexity will be ever worse.

2.6.2 Complexity of calculating the filter coefficients

For the frequency domain MIMO detector, we need to solve N sets of linear equation, one set for each FFT point, and each set consists of N_A equations. The size of the

matrix is $N_A \times N_A$, so the complexity of calculating the filter coefficients is roughly $N(N_A)^3$. For the time domain equalizer, the size of the matrix is $3LN_A \times 3LN_A$, and then the complexity of calculating filter coefficients is about $(3LN_A)^3$. When L is big, the time-domain MIMO detector is much more complex than frequency domain.

Chapter 3 Simulation Results

This c**h**apter presents computer simulation BER performance results for the proposed MIMO detector. The purpose of these simulations is to show that operating in the frequency domain is as effective at eliminating ISI and MAI as operating in the time domain, to show that frequency domain turbo detection/equalization is an effective tool for channels with long delay spreads where time-domain processing is not possible, and to highlight some of the remarkable advantages of using MIMO systems.

To provide a benchmark for system comparison we consider the matched filter bound (MFB) with maximum ratio combining (MRC). If the data rate is reduced by inserting a long guard interval between consecutively transmitted symbols, intersymbol interference can be avoided but the benefits of multipath diversity still enjoyed. The performance of a receiver with a filter matched to the received pulse shape is known as the matched filter bound. Systems with only a single user, using matched filter for each of the receive antennas, and using MRC on the filter output, is known as a MFB/MRC system. The performance of this system is optimal in the sense that it does not suffer from any ISI or MAI, but benefits from the multipath and multi-antenna diversity. It is hoped that our system can come close to the MFB/MRC performance.

Unless otherwise stated, all simulations are based on the following system parameters. Because the FFT operation we used requires that the number of symbols in an FFT block be a power of two, we choose a block size of $N = 1024$ for all simulations. A rate ½ convolutional encoder with generator $(5,7)_8$ and a constraint length of three ($L_c = 3$) is used, so the message block size is $N_a = 1022$ message symbols and each message symbol consists of one bit ($n_a = 1$). Pseudo-random bit interleaving is used, with an

interleaving size of 2048 bits. The BER results shown are after five iterations of the receiver. To reduce statistical uncertainty, the BER results are averaged over 20000 transmitted blocks for each user. Simulation is based on the discrete-time baseband ISI channel model, with symbol-rate sampling.

Section 3.1 discusses performance over a 5-tap fading channel. Then Section 3.2 discusses performance over the SUI-5 channel. Section 3.3 discusses the use of 16-QAM, and Section 3.4 discusses mixed modulation where one user uses QPSK and the other uses 16-QAM. Section 3.5 discusses symbol interleaving and bit interleaving.

3.1 System Performance over 5-tap channel

In this section, the performance of the MIMO system is discussed for QPSK over a 5-tap fading channel. The channel model has 5 path components and all taps have equal average power attenuation, so the variances of the taps are $(\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5},\frac{1}{5})$, with each path

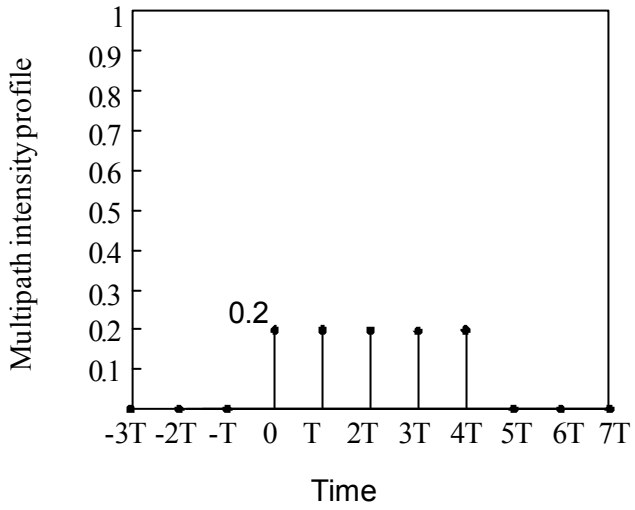


Figure 3-1. Channel delay profile of the 5-tap channel, where T is the symbol period.

experiencing independent fading. Figure 3-1 shows the multipath intensity profile of this channel model. All transmitted-to-receive antenna links use the same multipath intensity profile, but with independent fading. Furthermore, all links yield the same average received signal power.

Figure 3-2 shows the performance of the frequency domain MIMO detector over the 5-tap channel after each of the first five iterations. In this case, there are two users (N_S = 2) and two receive antennas (N_A = 2). As expected, the BER performance improves with each iteration of the MIMO equalizer. The system suffers a 7 dB interference penalty at a BER of 10^{-5} after the first iteration compared with the MFB/MRC performance. After the second iteration, the penalty is reduced to about 1 dB. Further iterative processing gives an overall improvement of 7 dB at a BER of 10^{-5}, and almost

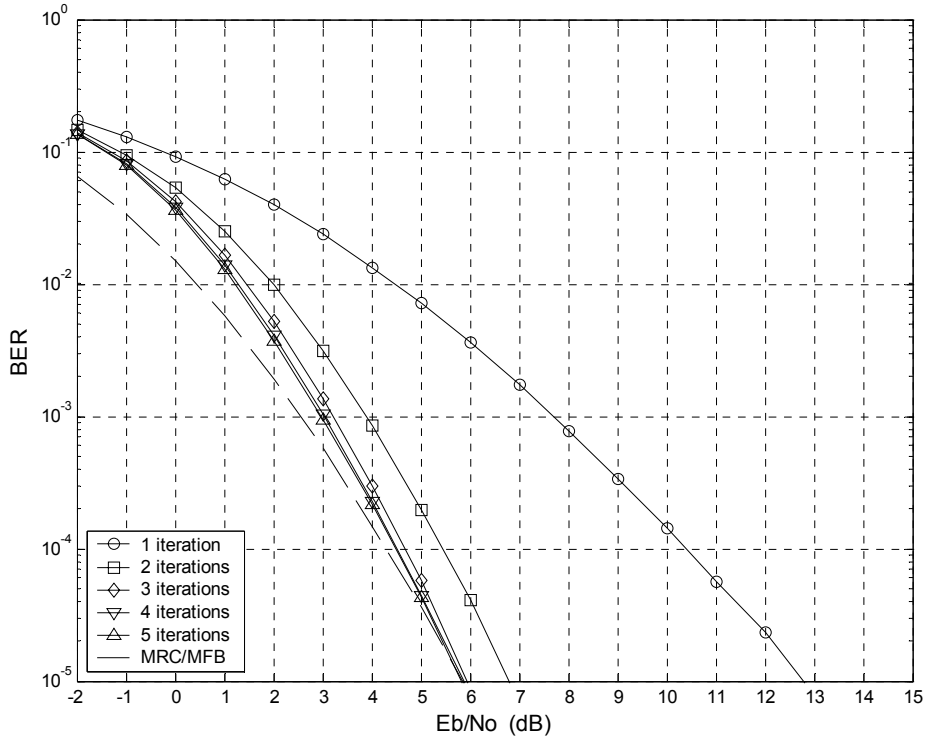


Figure 3-2. BER performance with N_S = 2 and N_A=2, for the 5-tap channel with QPSK

completely eliminates the interference from the received signal. The more iterations taken, the smaller the performance gain relative to the previous iteration. For example, the performance gain between the first and second iterations is about 6 dB at a BER of 10^{-5}, because there is no *a priori* feedback information in the first iteration. The performance gain between the third and fourth iterations is less than 0.1 dB, and further gain is not possible, because as the feedback becomes more and more accurate, the potential gain is smaller. When the feedback is perfect, there can be no additional performance gain, because the MFB has been reached.

Figure 3-3 provides a system performance comparison after the fifth iteration for two users and different numbers of receive antennas. It is found that increasing the

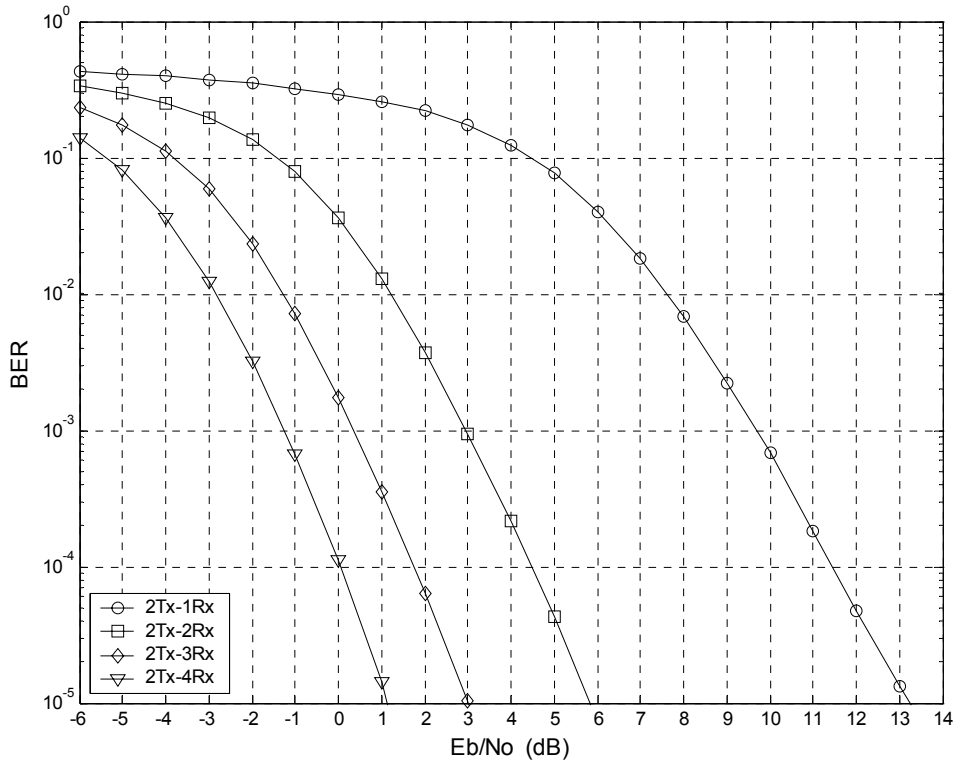


Figure 3-3. BER Performance with N_S = 2 and N_A = 1, 2, 3, 4, for 5-tap channel with QPSK.

number of receive antennas improves the BER performance as expected. If one radio path undergoes a deep fade, another independent path may have a strong signal, so by combining the signals from more than one receive antenna, the average SNR at the receiver may be improved. This is space diversity, also known as antenna diversity. We can find from Figure 3-3 that the performance gain between one and two antennas is 7.4 dB at a BER of 10^{-5}, and when the number of antennas is increased to three or four, the performance becomes even better. There is around 2.8 dB and 4.6 dB additional gain by using three or four antennas instead. Each additional antennas used provides smaller relative performance gain.

Figure 3-4 shows the system performance when time-domain processing is used

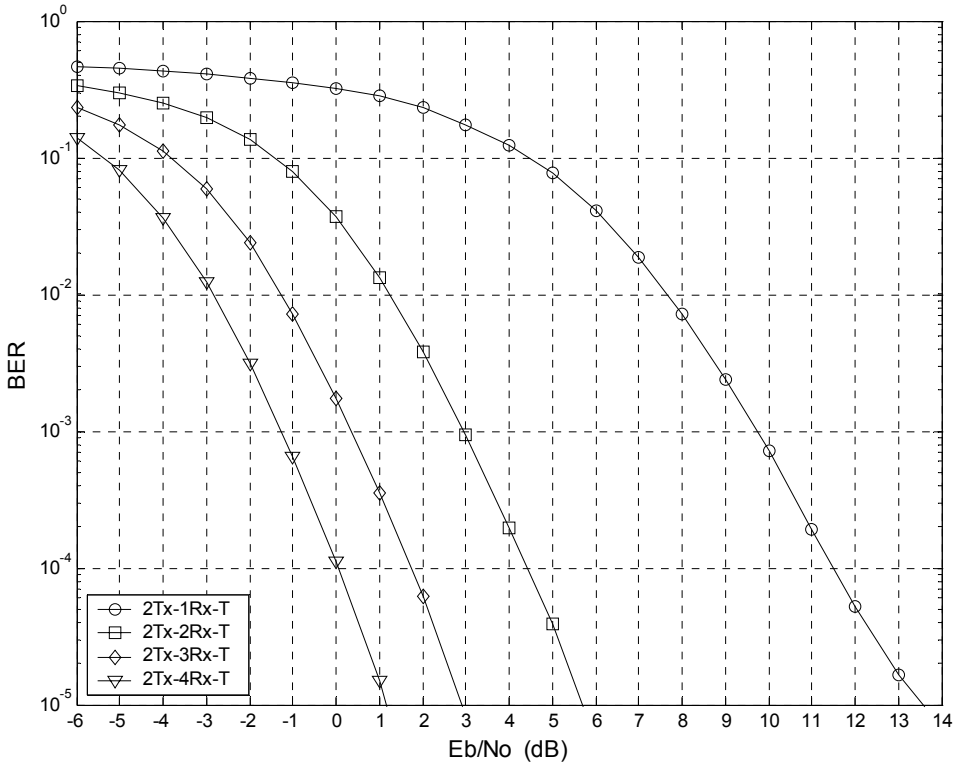


Figure 3-4. BER performance with $N_S = 2$ and $N_A = 1, 2, 3, 4$, for the 5-tap channel with QPSK in time domain, with 15 equalizer taps.

instead. Abe and Matsumoto's receiver is used [21], with 15 equalizer taps. Compared with Figure 3-3, we can find the system performance is almost the same. So we can get the result that operating in the frequency domain is as effective at eliminating the interference as in the time domain. However, the complexity of the frequency domain system is much lower.

Figure 3-5 shows the BER performance of the MIMO equalizer when the number of receive antennas is fixed at $N_A = 1$ and the number of users varies from $N_S = 1$ to $N_S = 4$. When $N_S = 1$ the system exhibits very good performance, coming within 0.6 dB of the MFB/MRC performance at a BER at 10^{-5}. Increasing the number of users to two leads to a doubling of the system throughput, but at the expense of a 1.1 dB degradation in performance at a BER of 10^{-5}. Further increases in the number of users leads to a system

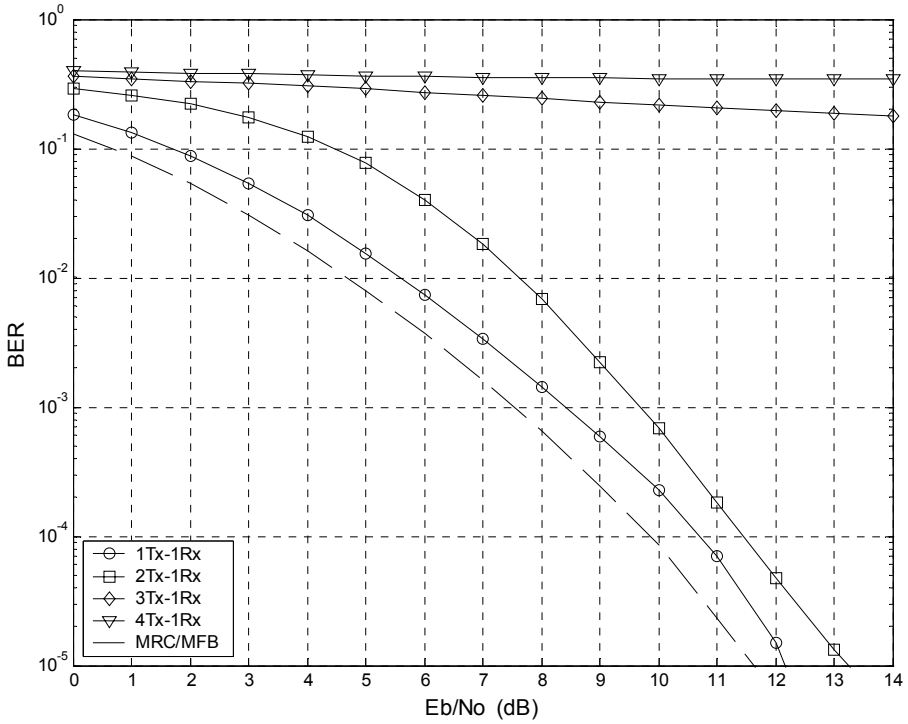


Figure 3-5. BER Performance with $N_A = 1$ and $N_S = 1, 2, 3, 4$, for the 5-tap channel with QPSK.

overload, where the detector is unable to distinguish the signal from the different users, even at high SNRs.

Figure 3-6 shows the BER performance when the number of receive antennas is fixed at $N_A = 2$. Compared with the case of one receive antenna as showed in Figure 3-5, the overall system performance is much better, due to the additional space diversity. With one user the performance can reach the MFB/MRC performance for SNRs above 5 dB. Furthermore, a second user can be added with no detrimental effects on the performance. Even with four users the degradation is still very small, about 0.8 dB. Therefore a four-fold increase in system throughput is possible with a penalty of less than 1 dB at a BER of 10^{-5}.

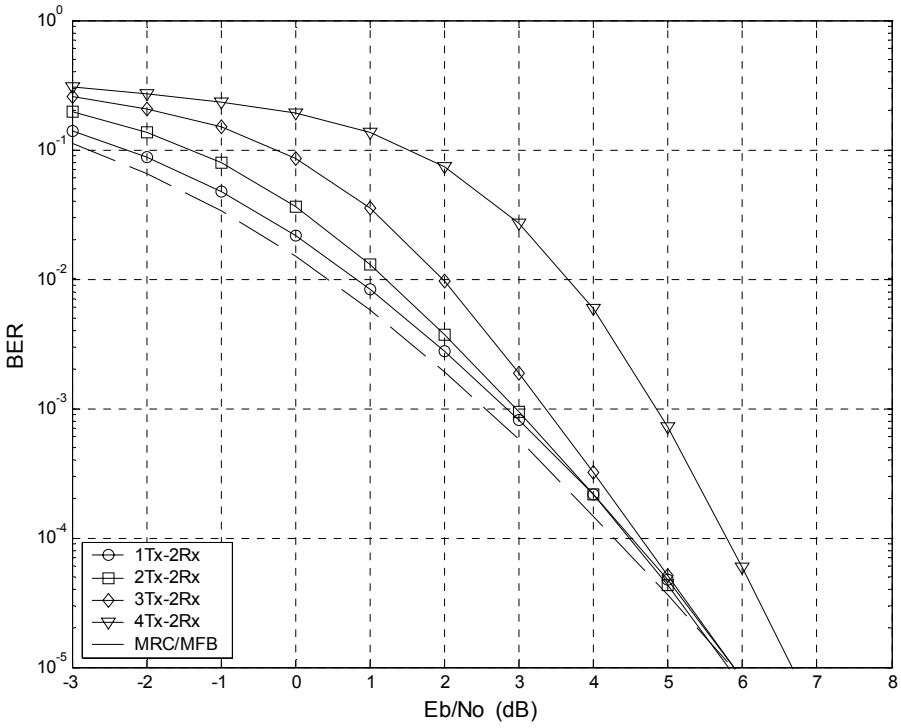


Figure 3-6. BER Performance with $N_A = 2$ and $N_s = 1, 2, 3, 4$, for the 5-tap channel with QPSK.

Figures 3-7 and 3-8 show the BER performance with three receive antennas (N_A = 3) and four receive antennas (N_A = 4), respectively. For the three antennas case we see that the system can support up to 5 simultaneous users, with no degradation in performance at SNRs over 2 dB. The four-antennas system can support up to 7 users.

From Figures 3-5 to 3-8, we find that if the number of users is less than twice the number of receive antennas, then the equalizer can completely eliminate the effects of the interference. So we can get the result that increasing the number of antennas not only leads to better performance because of the space diversity gain, but also allows the system to handle more users without any performance loss. In general, the system can

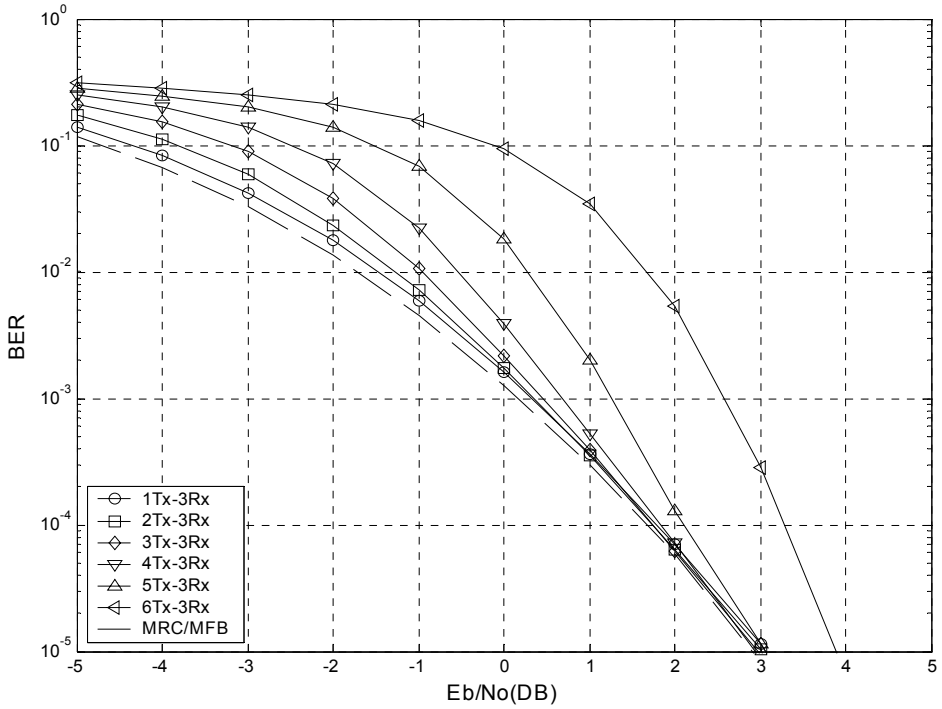


Figure 3-7. BER Performance with N_A = 3 and N_S =1, 2, 3, 4, for the 5-tap channel with QPSK .

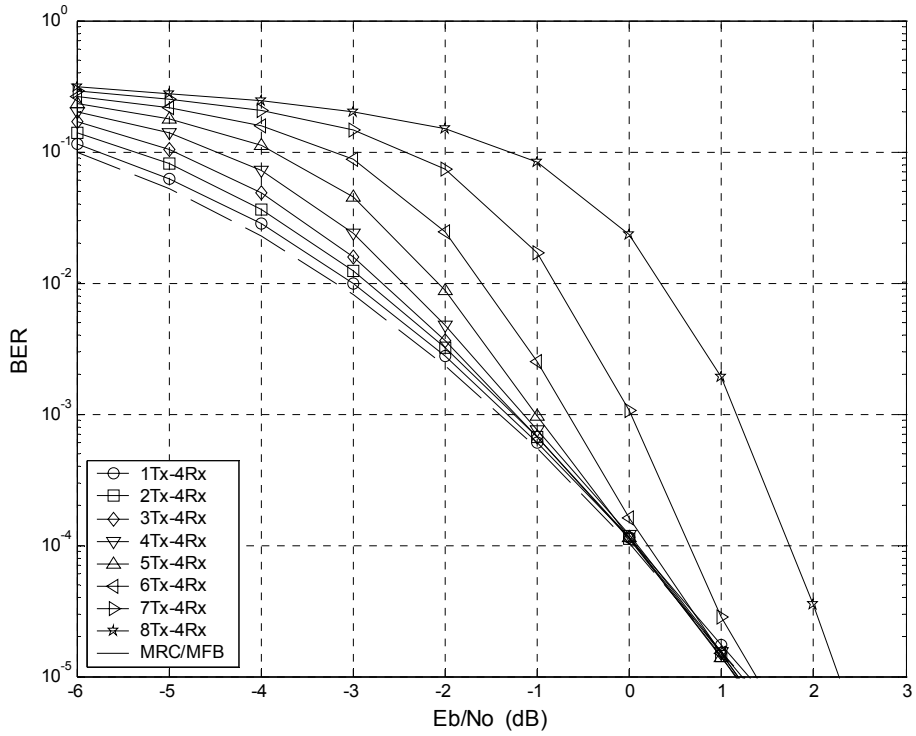


Figure 3-8. BER Performance with N_A = 4 and N_S= 1, 2, 3, 4, for the 5-tap channel with QPSK.

handle twice as many users as the number of receive antennas with a penalty of less than about 1 dB at a BER of 10^{-5}.

3.2 System Performance over the SUI-5 Channel

The 5-tap channel model considered in the previous section exhibits fairly mild fading, because it is unlikely that all five taps will simultaneously suffer from a deep fade. In this section we consider a more difficult channel, the SUI-5 channel model [16], which has a delay spread profile as shown in Figure 3-9. Because the three taps have unequal variances (specifically, 0.706, 0.223 and 0.0706), severe fading is much more probable.

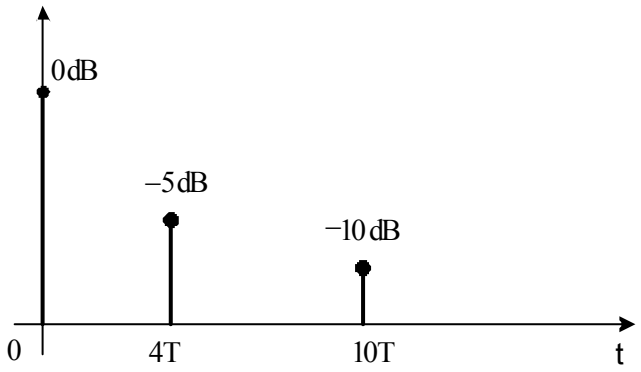


Figure 3-9. Delay spread profile for SUI-5 channel.

Further, because the delay spread is fairly long (covering 11 symbols), time domain equalization is very complicated.

Figure 3-10 shows the performance of the frequency domain MIMO detector over

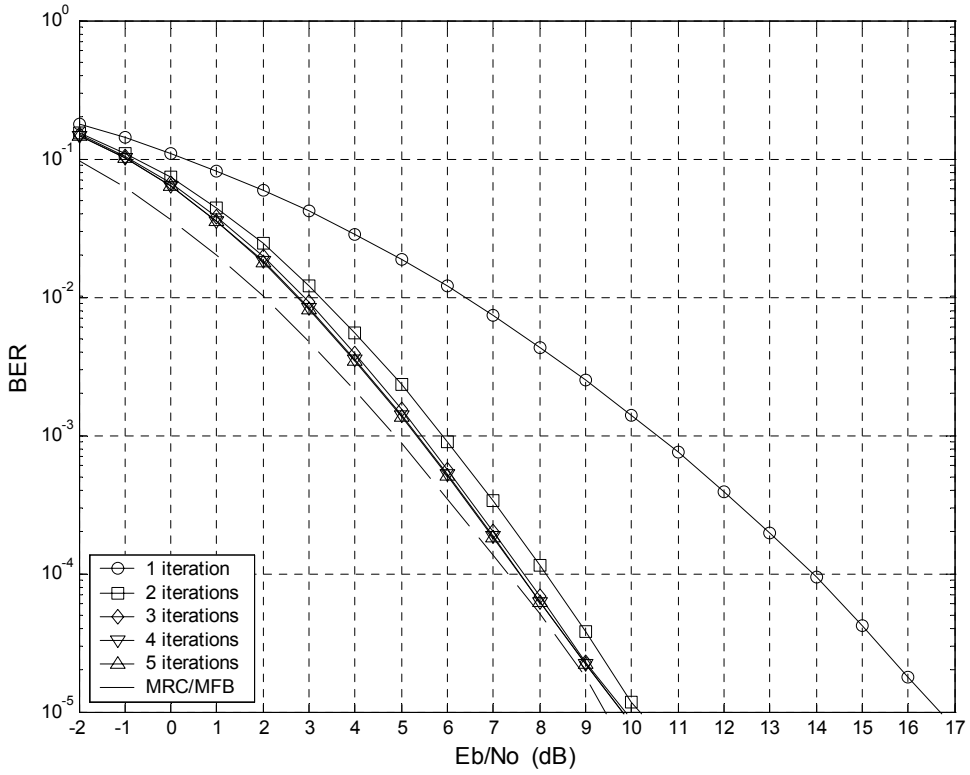


Figure 3-10. BER Performance with N_S = 2 and N_A = 2, for the SUI-5 channel with QPSK.

the SUI-5 channel after each of the first five iterations. In this case, the number of users is two and the number of receive antennas is also two. The system suffers a 7 dB interference penalty at a BER of 10^{-5} after the first iteration compared with the MFB/MRC performance. After the second iteration, the interference penalty is reduced to about 1 dB. Further iterative processing gives an overall improvement of 6.6 dB, and the interference penalty is reduced to about 0.4 dB.

Compared with the 5-tap channel, as shown in Figure 3-2, we get the same result that the more iterations, the better the performance, but the system performance is 4 dB worse than for the 5-tap channel at a BER of 10^{-5}.

Figures 3-11 to 3-14 provides a system performance comparison when the number

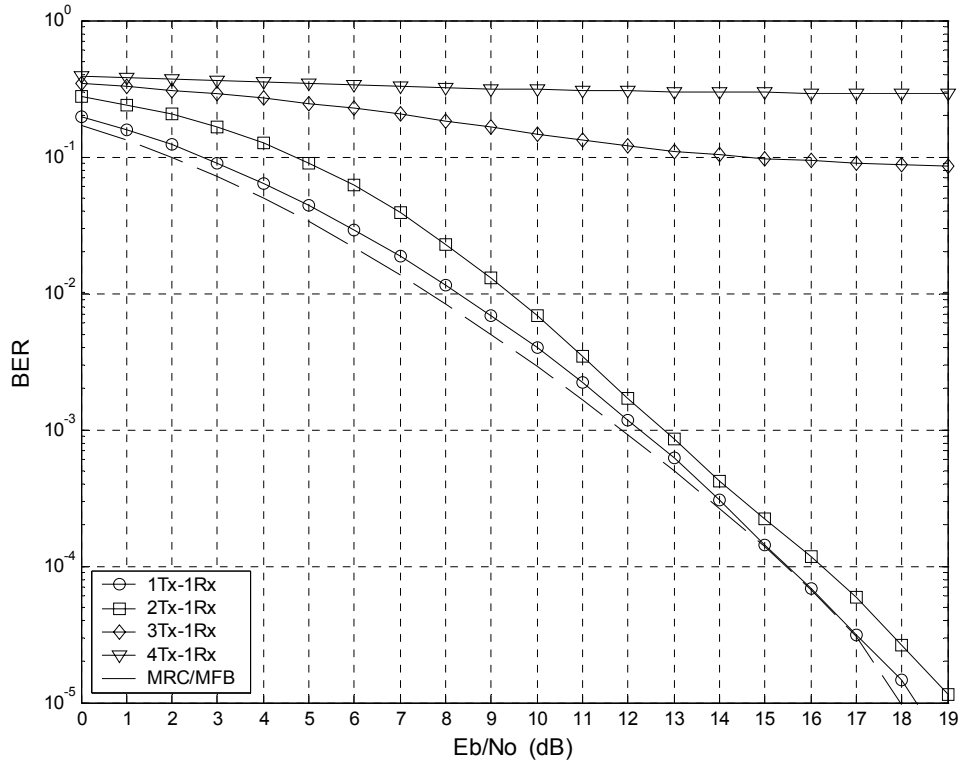


Figure 3-11. BER Performance with $N_A = 1$ and $N_S = 1, 2, 3, 4$, for the SUI-5 channel with QPSK.

of antennas is fixed and the number of users is increased from one to four. From these figures we observe that the receiver is capable of eliminating the interference, just as it is when 5-tap channel is used, provided the number of users is not too large relative to the number of receive antennas. It is worth noting that although the overall system performance is worse with the SUI-5 channel than the 5-tap channel, the advantage of using multiple receive antennas is greater. For example, with only two receive antennas the SUI-5 channel is 4 dB worse than the 5-tap channel, but with four receive antennas the difference is only 2 dB. Therefore multiple receive antennas can compensate for a lack of multipath diversity in the channel.

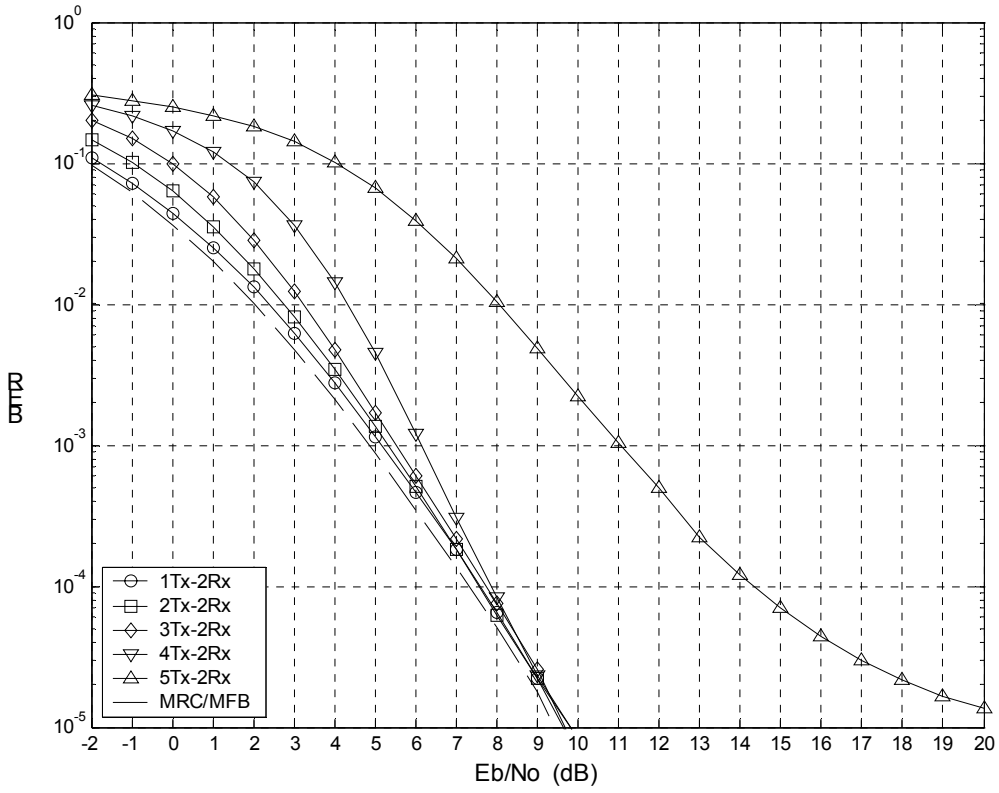


Figure 3-12. BER Performance with $N_A = 2$ and $N_s = 1, 2, 3, 4$, for the SUI-5 channel with QPSK.

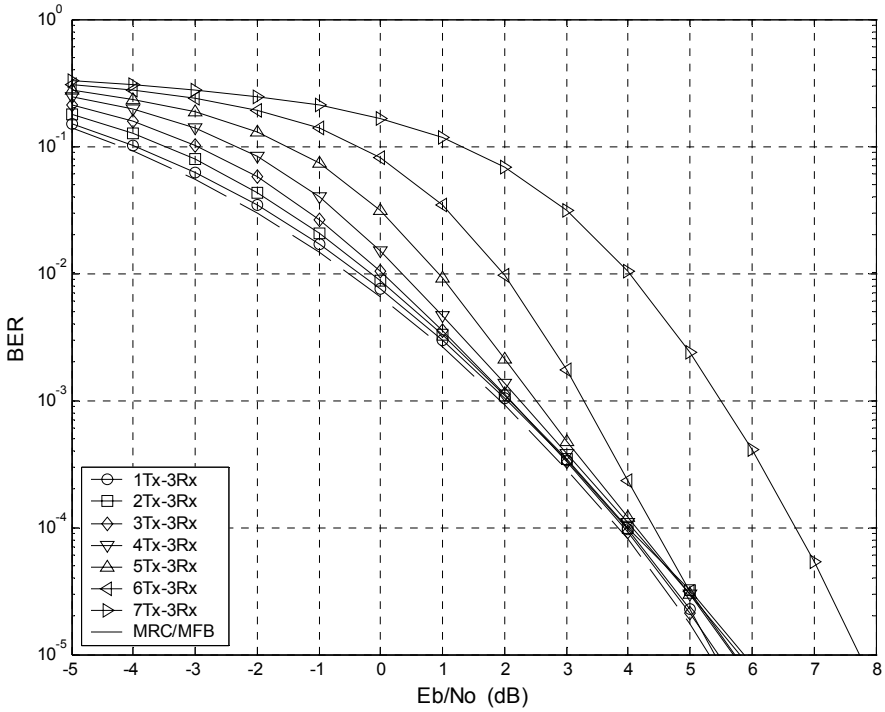


Figure 3-13. BER Performance with $N_A = 3$ and $N_S = 1, 2, 3, 4$, for the SUI-5 channel with QPSK.

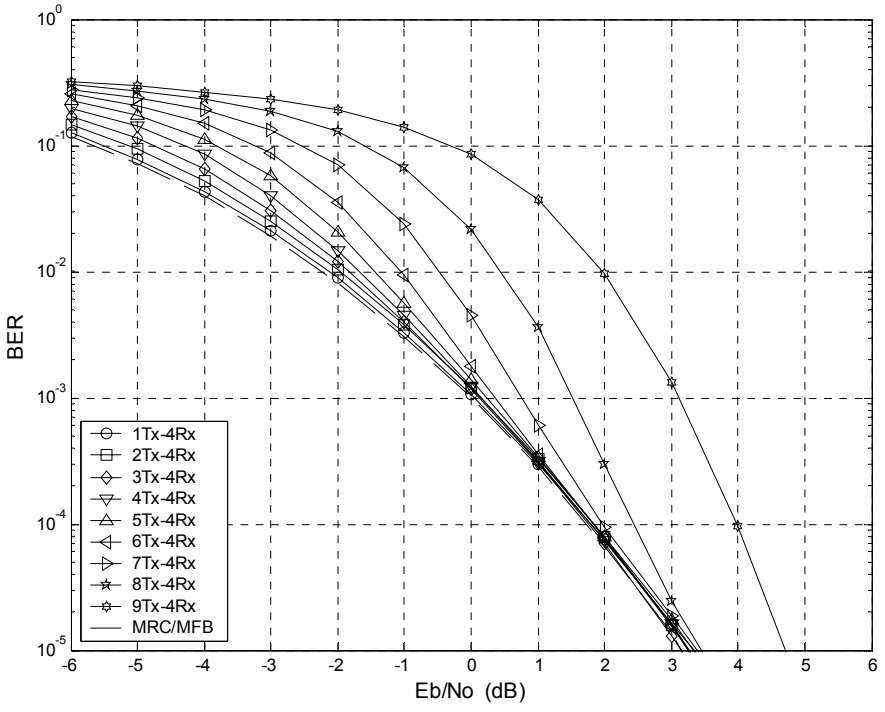


Figure 3-14. BER Performance with $N_A = 4$ and $N_S = 1, 2, 3, 4$, for the SUI-5 channel with QPSK.

3.3 Effects of Spectrally Efficient Modulation

The results in the previous section are all based on QPSK with rate ½ codes, which delivers one information bit per transmitted symbol. To improve the spectral efficiency, we consider trellis coded 16-QAM modulation, with the rate ¾ convolutional encoder as shown in Figure 3-15 [17]. Since the constraint length is $L_c = 3$, the number of message symbols for each block is 1022 symbols ($N_a = 1022$) and each message symbol consists of three bits ($n_a = 3$). The number of code symbols is 1024 symbols ($N_c = 1024$) and each code symbol has four code bits. The throughput per user is three information bits per transmitted symbol. For this case, symbol interleaving is used.

The BER results in Figure 3-16 to 3-19 show a similar trend as for the QPSK system (as shown in Figure 3-5 to 3-8), in that increasing the number of receive antennas gives better results. But overall the results for 16-QAM are much worse than for QPSK. For example, when there are two receive antennas, the QPSK system can support four users within 1 dB of the MFB/MRC performance, but the 16-QAM system can only

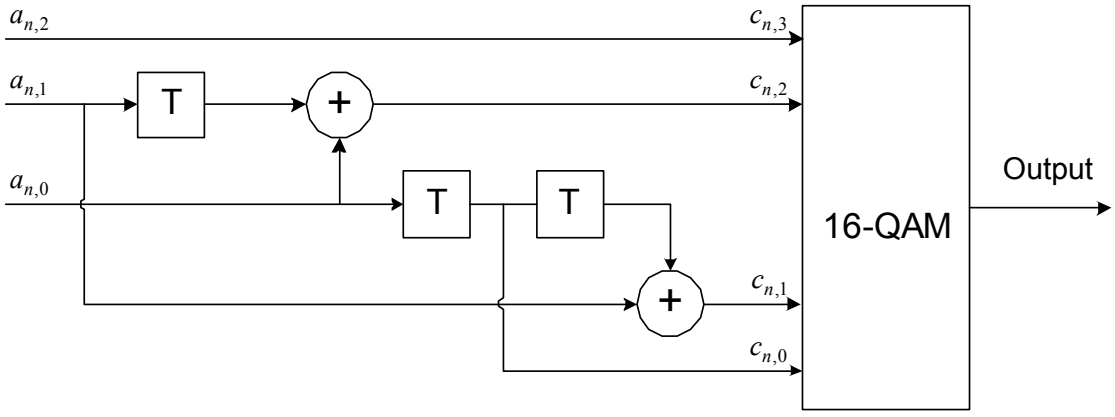


Figure 3-15. The rate ¾ convolutional encoder.

support two users, and ever then the interference penalty is about 4 dB at a BER of 10^{-5}. Further with two users the 16-QAM system is 7.5 dB worse than the QPSK system.

In general, the 16-QAM systems appears to be capable of supporting only half as many users as the QPSK system, with a much larger interference penalty and higher overall SNR requirements. However, even though a lower number of users can be supported, the aggregate system throughput can be higher than for the QPSK system. For example, as shown in Figure 3-19, the 16-QAM system with four users and four antennas can achieve a BER of 10^{-5} at a SNR of 6.8 dB. Since each user sends three information bits per transmitted symbol, this corresponds to an aggregate throughput of 12 bits per

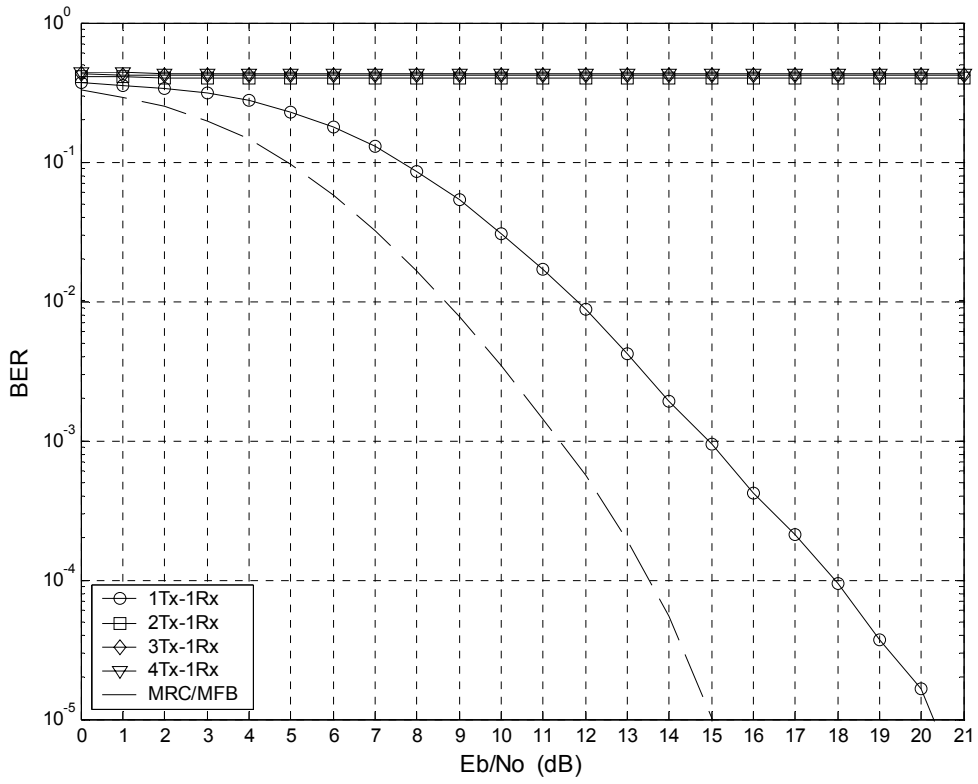


Figure 3-16. BER Performance with N_A=1 and N_S= 1, 2, 3, 4, for the 5-tap channel with 16QAM.

channel use. With four receive antennas the QPSK system can only support 8 users, for an aggregate throughput of 8 bits per channel use. To match the throughput of the 16-QAM system, the number of QPSK users would have to be increased to 12. However, with four antennas the receiver is unable to separate the signals from that many users, regardless of how large the SNR is. Therefore the 16-QAM system can provide higher spectral efficiency.

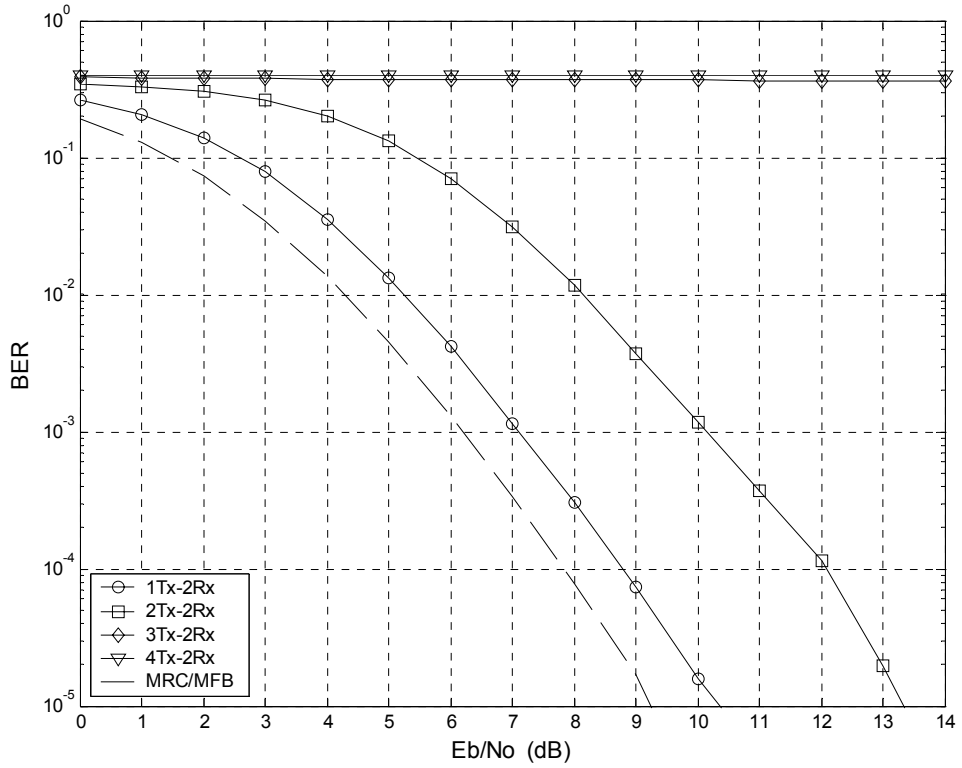


Figure 3-17. BER Performance with N_A =2 and N_S = 1, 2, 3, 4, for the 5-tap channel with 16QAM.

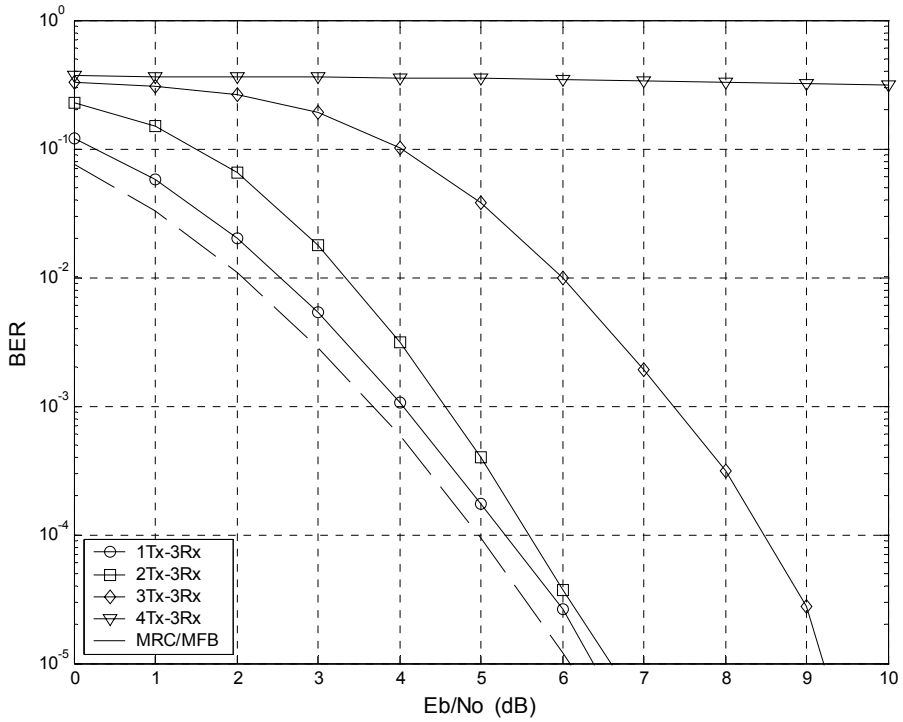


Figure 3-18. BER Performance with $N_A = 3$ and $N_S = 1, 2, 3, 4$, for the 5-tap channel with 16QAM.

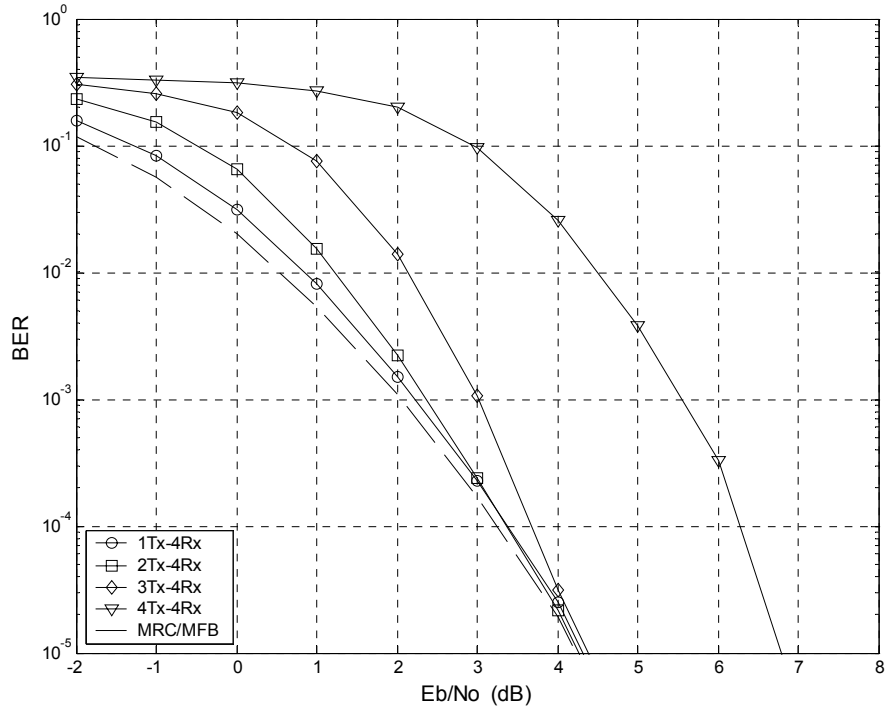


Figure 3-19. BER Performance with $N_A = 4$ and $N_S = 1, 2, 3, 4$, for the 5-tap channel with 16QAM.

3.4 Mixed Modulation

In this section we investigate the system performance when different users employ different modulation schemes. In particular, we consider a system with two users, one using QPSK and the other using 16-QAM. For the 5-tap channel with two receive antennas the individual BER performance of the two users is shown in Figure 3-20. The performance curve for the QPSK user is labeled as "QPSK (w/16QAM)", and the curve for the 16-QAM user is labeled as "16QAM(w/QPSK)". As expected, the BER performance of the QPSK user is much better than the 16-QAM user, by about 2.5 dB at a BER of 10^{-5}.

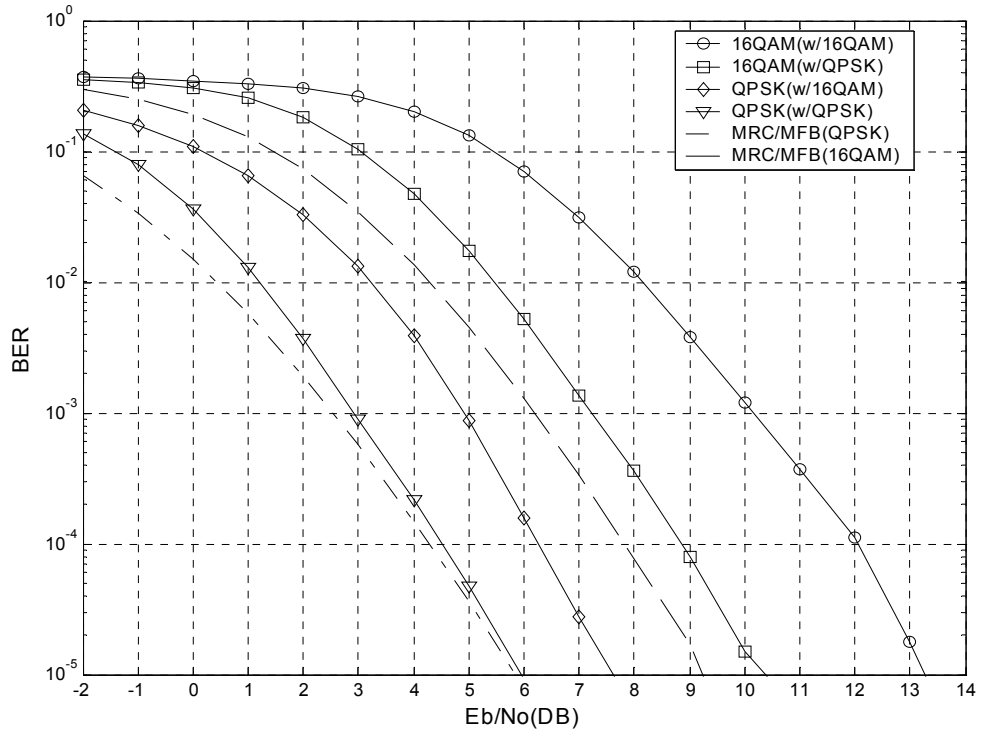


Figure 3-20. BER Performance with $N_S = 2$ and $N_A = 2$, for the 5-tap channel, QPSK and 16-QAM.

For comparison, we also show the BER performance of a QPSK user when paired with another QPSK user instead of a 16-QAM user labeled as "QPSK(w/QPSK)", and a 16-QAM user paired with another 16-QAM user labeled as "16QAM(w/16QAM)". For the QPSK user there is a 1.7 dB penalty when paired with a 16-QAM user instead of another QPSK user. This is because, at a given SNR, the feedback from the decoder from a 16-QAM user is much less reliable than from a QPSK user. As a result, the interference suppression is less effective, leading to worse performance for the QPSK user. For the same reason there is a 3 dB advantage for the 16-QAM user when paired with a QPSK user than with another 16-QAM user. The more reliable feedback from the other user leads to better interference suppression.

3.5 Symbol Interleaving vs. Bit Interleaving

In Section 3.1 we discussed the performance using bit-interleaving. We know symbol interleaving has the advantages of lower implementation complexity at the receiver, so in this section we will discuss the use of symbol interleaving in fading channels.

Figure 3-21 provides a system performance comparison when the number of users is two and the number of antennas is increased from one to four. Clearly the performance of symbol interleaving is almost same as that of bit interleaving for the 5-tap fading channel when the modulation scheme is QPSK agreeing with the observations in [8]. Since symbol interleaving is easy to implement, it is therefore a good choice. Note, however, that this conclusion may not be valid when other modulation schemes, such as 16-QAM, are used.

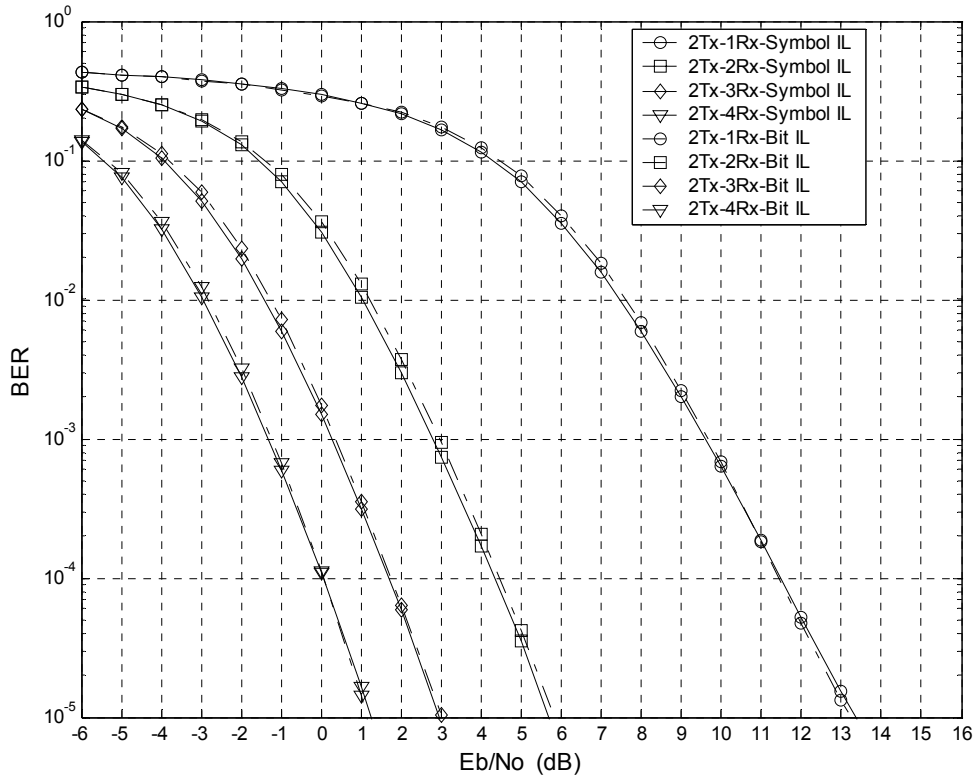


Figure 3-21. BER Performance with $N_S = 2$ and $N_A = 1, 2, 3, 4$, for the 5-tap channel, symbol interleaving and bit interleaving.

Chapter 4 Conclusions and Future work

4.1 Conclusions

In this thesis, we derived an iterative MIMO detector that uses frequency domain filtering. The receiver consists of a bank of MIMO detectors and soft output decoders. The soft decoders' outputs are used as feedback information for improving the detector's estimates in the next iteration. The work in this thesis has investigated the performance of the MIMO detector for different MIMO channels and modulation schemes. In particular, the following results have been observed:

- The proposed MIMO detector has excellent performance and can completely eliminate ISI and MAI given enough receiving antennas by iterative processing and is a suitable candidate for broadband wireless system.
- Operating in the frequency domain is as effective at eliminating the interference as in the time domain. However, the complexity of the frequency domain system is much lower.
- For QPSK, the system can handle twice as many users as the number of receive antennas for both channel models considered. For 16-QAM, the system can only handle as many users as the number of receive antennas. However, the aggregate throughput for 16-QAM may be higher than for QPSK.
- Using multiple antennas in the receiver not only gives diversity gain, but also allows the system to handle more users without any performance loss.

- Symbol interleaving is less complex than bit interleaving, but when QPSK is used, the performance of symbol interleaving is almost the same as that of bit interleaving.
- When mixed modulation methods are used, each user's BER is strongly affected by the other user's BER.

4.2 Future Work

The results presented in this thesis raise some interesting topics for future research:

- For simplicity, we assume that the users transmit signal at same time, I.e. cyclic prefix are synchronized. A more realistic model would support asynchronous transmitters.
- In this thesis, we assume the detector knows all of the parameters related to the channels. In practice, the parameters have to be estimated. Iterative frequency-domain channel estimation should be explored in the future work.
- Investigate the use of bit-interleaved coded modulation for spectral efficiency.
- More powerful error control coding, such as turbo codes or LDPC codes.
- Explore the use of space-time coding, where users have multiple transmit antennas.
- Combination with spectrum- spreading (CDMA).

References

[1] C. Douillard, A. Picat, P. Didier, M. Jezequel, C. Berrou and A. Glavieux, "Iterative correction of intersymbol interference: turbo equalization", *European Transaction Telecommunications*, vol. 6, pp. 507-511, Sept./Oct. 1995.

[2] L.R. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Transactions on Information Theory*, vol. 20, pp. 284-287, March 1974.

[3] A. Glavieux, C. Laot and J. Labat, "Turbo-equalization over a frequency selective channel," in *International Symposium On Turbo Codes*, Brest, France, Sept. 1997, pp. 96-102.

[4] M. Tuchler, A.C. Singer and R. Koetter, "Minimum mean square error equalization using a priori information", *IEEE Transactions on Signal Processing*, vol. 50, pp. 673-683, March 2002.

[5] M. Tuchler, R. Kotter and A. Singer, "Turbo equalization: principles and new results", *IEEE Transaction on Communications*, vol. 50, pp. 754-767, May 2002.

[6] P.A. Dmochowski, P.J. McLane, "Frequency domain equalization for high data rate multipath channels", in *Proc. IEEE Pacific Rim Conference on Communications, Computers and signal Processing*, Victoria, B.C., Canada, vol. 2, Aug. 2001, pp. 534–537.

[7] M.V. Clark, "Adaptive frequency-domain equalization and diversity combining for broadband wireless communications", *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1385 – 1395, Oct. 1998.

[8] D. Falconer, S.L. Ariyavisitakul, A. Benyamin-Seeyar and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless system", *IEEE Communications Magazine*, vol. 40, pp. 58-66, April 2002.

[9] M. Tuchler and J. Hagenauer, "Turbo equalization using frequency domain equalizers," in *Proc. Allerton Conf.,* Monticello, IL, USA, Oct. 2000.

[10] M. Tuchler and J. Hagenauer, "Linear time and frequency domain turbo equalization", in *Proc. IEEE Vehicular Technology Conference*, Rhodes, Greece, vol. 2, May 2001, pp. 1449 – 1453.

[11] G.J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multiple antennas", *Bell Labs Technical Journal*, vol. 1, pp. 41-59, Autumn 1996.

[12] G.J. Foschini and M.J. Gans, “On limits of wireless communications in a fading environment when using multiple antennas”, *Wireless Personal Communications*, vol. 6, pp. 311-335, 1998.

[13] K.K. Wong, R.D. Murch, K.B. Letaief, “Optimizing time and space MIMO antenna system for frequency selective fading channels”, *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 1395-1407, Jul. 2001.

[14] T. Abe and T. Matsumoto, “Space-time turbo equalization in frequency-selective MIMO channels”, *IEEE Transactions on Vehicular Technology*, vol. 52, pp. 469 – 475, May 2003.

[15] S. Sun, “Reduced-complexity frequency domain turbo equalization” , Master’s thesis, Carleton University, Jan. 2004.

[16] V. Erceg, K.K.S. Hari, M.S. Smith, K.P. Sheikh, C. Tappenden, J.M. Costa, D.S. Baum and C. Bushue, “Channel models for fixed wireless applications”, IEEE 802.16. Proposal 802.16.3c-01/29r4.

[17] G. Ungerboeck, “Channel coding with multilever/phase signal”, *IEEE Transaction on Information Theory*, vol. 28, pp. 55-66, Jan. 1982.

Appendix MIMO detector

A.1 Introduction

This appendix contains a detailed description of the proposed MIMO detector. The filter coefficients $\widetilde{P}_k^{(s,a)}$ and $\widetilde{b}_n^{(s)}$ are determined according to the MMSE criterion under the assumption that the *a priori* probability of the transmitted symbols, $\{v_n^{(s)}\}$, are arbitrary. Since the filter coefficients under this assumption depend on *n*, to simplify the problem we find the filter coefficients that minimize the time-averaged MSE, instead of the instantaneous MSE.

This appendix is composed of four sections, including this introduction. Section 2 contains the derivation of the detector coefficients, Section 3 contains the calculation of mean and variance of $z_n^{(s)}$, and the last section describes a modification to the detector which avoids a numerical instability problem.

A.2 Finding the Detector Coefficients

The output of the filter for user s at time *n* is

$$z_n^{(s)} = \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}R_k^{(a)}e^{j2\pi\frac{kn}{N}} + \widetilde{b}_n^{(s)}, \tag{A-1}$$

which is just a linear combination of all the frequency-domain received samples from all the antennas. According to the minimum mean square error (MMSE) criterion, we must find $\widetilde{P}_k^{(s,a)}$ and $\widetilde{b}_n^{(s)}$ which minimize the mean square error between $z_n^{(s)}$ and $v_n^{(s)}$,

$$J^{(s)} = \frac{1}{N}\sum_{n=0}^{N-1}\mathrm{E}\left[\left|z_n^{(s)} - v_n^{(s)}\right|^2\right]. \tag{A-2}$$

The time-averaged MSE is used to ensure that the filter coefficients $\widetilde{P}_k^{(s,a)}$ do not depend on time.

To find the filter coefficients for user s, we find the partial derivatives of $J^{(s)}$ with respect to $\{\widetilde{b}_n^{(s)}\}$ and $\{\widetilde{P}_l^{(s,b)}\}$ for all n, l and b. Setting these derivates to zero gives a set of linear equations that can be solved to find the filter coefficients.

Setting the partial derivative of $J^{(s)}$ with respect to $\widetilde{b}_n^{(s)}$ to zero is equivalent to setting

$$\frac{1}{N}\sum_{m=0}^{N-1}\mathrm{E}\left[\left(z_m^{(s)}-v_m^{(s)}\right)\frac{\partial}{\partial\widetilde{b}_n^{(s)}}\left(z_m^{(s)}-v_m^{(s)}\right)^*\right]=0\,. \tag{A-3}$$

Since

$$\frac{\partial}{\partial\widetilde{b}_n^{(s)}}\left(z_m^{(s)}-v_m^{(s)}\right)=\frac{\partial}{\partial\widetilde{b}_n^{(s)}}z_m^{(s)}=\delta_{n-m}\,, \tag{A-4}$$

Equation (A-3) becomes

$$\frac{1}{N}\sum_{m=0}^{N-1}\mathrm{E}\left[\left(z_m^{(s)}-v_m^{(s)}\right)\delta_{n-m}\right]=0 \tag{A-5}$$

or

$$\mathrm{E}\left[z_n^{(s)}\right]=\mathrm{E}\left[v_n^{(s)}\right]. \tag{A-6}$$

Defining $\overline{v}_n^{(s)}=E[v_n^{(s)}]$ and substituting Equation (A-1) for $z_n^{(s)}$ gives

$$\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\,\mathrm{E}\left[R_k^{(a)}\right]e^{j2\pi\frac{kn}{N}}+\widetilde{b}_n^{(s)}=\overline{v}_n^{(s)}\,, \tag{A-7}$$

so

$$\widetilde{b}_n^{(s)}=\overline{v}_n^{(s)}-\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\overline{R}_k^{(a)}e^{j2\pi\frac{kn}{N}} \tag{A-8}$$

is a solution for $\widetilde{b}_n^{(s)}$, where $\overline{R}_k^{(a)} = \mathrm{E}\left[R_k^{(a)}\right]$. Substituting this back into Equation (A-1) gives

$$z_n^{(s)} = \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left(R_k^{(a)} - \overline{R}_k^{(a)}\right)e^{j2\pi\frac{kn}{N}} + \overline{v}_n^{(s)} \tag{A-9}$$

for the filter output.

Setting the partial derivative of $J^{(s)}$ with respect to $\{\widetilde{P}_l^{(s,b)}\}$ to zero is equivalent to setting

$$\frac{1}{N}\sum_{n=0}^{N-1}\mathrm{E}\left[\left(z_n^{(s)} - v_n^{(s)}\right)\frac{\partial}{\partial\widetilde{P}_l^{(s,b)}}\left(z_n^{(s)} - v_n^{(s)}\right)^*\right] = 0\,. \tag{A-10}$$

Since

$$\frac{\partial}{\partial\widetilde{P}_l^{(s,b)}}\left(z_n^{(s)} - v_n^{(s)}\right) = \frac{1}{N}\left(R_l^{(b)} - \overline{R}_l^{(b)}\right)e^{j2\pi\frac{ln}{N}}, \tag{A-11}$$

Equation (A-10) becomes

$$\frac{1}{N}\sum_{n=0}^{N-1}\mathrm{E}\left[\left(z_n^{(s)} - v_n^{(s)}\right)\frac{1}{N}\left(R_l^{(b)} - \overline{R}_l^{(b)}\right)^* e^{-j2\pi\frac{ln}{N}}\right] = 0\,. \tag{A-12}$$

Substituting Equation (A-9) for $z_n^{(s)}$ gives

$$\frac{1}{N}\sum_{n=0}^{N-1}\mathrm{E}\left[\left(\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left(R_k^{(a)} - \overline{R}_k^{(a)}\right)e^{j2\pi\frac{kn}{N}} + \overline{v}_n^{(s)} - v_n^{(s)}\right)\right.$$

$$\left.\cdot\left(R_l^{(b)} - \overline{R}_l^{(b)}\right)^*\right]e^{-j2\pi\frac{ln}{N}} = 0\,, \tag{A-13}$$

or

$$\frac{1}{N^2}\sum_{n=0}^{N-1}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\,\mathrm{E}\left[\left(R_k^{(a)} - \overline{R}_k^{(a)}\right)\left(R_l^{(b)} - \overline{R}_l^{(b)}\right)^*\right]e^{j2\pi\frac{(k-l)n}{N}}$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)}\right)\left(R_l^{(b)} - \overline{R}_l^{(b)}\right)^*\right] e^{-j2\pi\frac{ln}{N}}, \tag{A-14}$$

or

$$\begin{aligned}\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A} \widetilde{P}_k^{(s,a)} \mathrm{Cov}\left[R_k^{(a)}, R_l^{(b)}\right] \frac{1}{N}\sum_{n=0}^{N-1} e^{j2\pi\frac{(k-l)n}{N}} \\ = \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{Cov}\left[v_n^{(s)}, R_l^{(b)}\right] e^{-j2\pi\frac{ln}{N}},\end{aligned} \tag{A-15}$$

where

$$\mathrm{Cov}[X, Y] = \mathrm{E}\left[\left(X - \mathrm{E}[X]\right)\left(Y - \mathrm{E}[Y]\right)^*\right] \tag{A-16}$$

is the covariance of X and Y. Since

$$\frac{1}{N}\sum_{n=0}^{N-1} e^{j2\pi\frac{kn}{N}} = \delta_k \ , \tag{A-17}$$

Equation (A-15) becomes

$$\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A} \widetilde{P}_k^{(s,a)} \mathrm{Cov}\left[R_k^{(a)}, R_l^{(b)}\right] \delta_{k-l} = \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{Cov}\left[v_n^{(s)}, R_l^{(b)}\right] e^{-j2\pi\frac{ln}{N}}, \tag{A-18}$$

or

$$\frac{1}{N}\sum_{a=1}^{N_A} \widetilde{P}_l^{(s,a)} \mathrm{Cov}\left[R_l^{(a)}, R_l^{(b)}\right] = \frac{1}{N}\sum_{n=0}^{N-1} \mathrm{Cov}\left[v_n^{(s)}, R_l^{(b)}\right] e^{-j2\pi\frac{ln}{N}} \ . \tag{A-19}$$

To evaluate Equation (A-19), it is necessary to first find expressions for the covariances.

The covariance between two transmitted symbols, $v_n^{(s)}$ and $v_m^{(u)}$, is

$$\begin{aligned}\mathrm{Cov}\left[v_n^{(s)}, v_m^{(u)}\right] &= \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)}\right)\left(v_m^{(u)} - \overline{v}_m^{(u)}\right)^*\right] \\ &= \begin{cases} \mathrm{E}\left[v_n^{(s)} - \overline{v}_n^{(s)}\right]\mathrm{E}\left[v_m^{(u)} - \overline{v}_m^{(u)}\right]^* & \text{if} \quad m \neq n \ \text{ or } \ s \neq u \\ \mathrm{E}\left[\left|v_n^{(s)} - \overline{v}_n^{(s)}\right|^2\right] & \text{if} \quad m = n \ \text{ and } s = u, \end{cases}\end{aligned}$$

(A-20)

since symbols transmitted at different times or by different users are independent. So $\mathrm{E}\left[v_n^{(s)} - \overline{v}_n^{(s)}\right] = \mathrm{E}\left[v_n^{(s)}\right] - \mathrm{E}\left[v_n^{(s)}\right] = 0$. By defining $\sigma_{v,n,s}^2 = \mathrm{E}\left[\left|v_n^{(s)} - \overline{v}_n^{(s)}\right|^2\right]$ as the variance of $v_n^{(s)}$, Equation (A-20) can be written as

$$\mathrm{Cov}\left[v_n^{(s)}, v_m^{(u)}\right] = \sigma_{v,n,s}^2 \delta_{m-n} \delta_{s-u} \quad . \tag{A-21}$$

To find $\mathrm{Cov}\left[v_n^{(s)}, R_l^{(b)}\right]$, note that since

$$V_k^{(s)} = \sum_{n=0}^{N-1} v_n^{(s)} e^{-j2\pi \frac{kn}{N}} \tag{A-22}$$

is the DFT of the symbols transmitted by user s ,

$$\begin{aligned}
\mathrm{Cov}\left[v_n^{(s)}, V_l^{(u)}\right] &= \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)}\right)\left(V_l^{(u)} - \overline{V}_l^{(u)}\right)^*\right] \\
&= \sum_{m=0}^{N-1} \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)}\right)\left(v_m^{(u)} - \overline{v}_m^{(u)}\right)^*\right] e^{j2\pi \frac{lm}{N}} \\
&= \sum_{m=0}^{N-1} \mathrm{Cov}\left[v_n^{(s)}, v_m^{(u)}\right] e^{j2\pi \frac{lm}{N}} \\
&= \sum_{m=0}^{N-1} \left[\sigma_{v,n,s}^2 \delta_{m-n} \delta_{s-u}\right] e^{j2\pi \frac{lm}{N}} \\
&= \sigma_{v,n,s}^2 \delta_{s-u} e^{j2\pi \frac{ln}{N}} .
\end{aligned} \tag{A-23}$$

Since

$$R_k^{(a)} = \sum_{u=1}^{N_S} H_k^{(u,a)} V_k^{(u)} + W_k^{(a)} \tag{A-24}$$

is the DFT of the samples received from antenna a ,

$$\mathrm{Cov}\left[v_n^{(s)}, R_l^{(b)}\right] = \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)}\right)\left(R_l^{(b)} - \overline{R}_l^{(b)}\right)^*\right]$$

$$= \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)} \right) \left(\sum_{u=1}^{N_S} H_l^{(u,b)} V_l^{(u)} + W_l^{(b)} - \sum_{u=1}^{N_S} H_l^{(u,b)} \overline{V}_l^{(u)} \right)^* \right]$$

$$= \sum_{u=1}^{N_S} H_l^{(u,b)*} \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)} \right) \left(V_l^{(u)} - V_l^{(u)} \right)^* \right] + \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)} \right) W_l^{(b)*} \right]$$

$$= \sum_{u=1}^{N_S} H_l^{(u,b)*} \mathrm{Cov}\left[v_n^{(s)}, V_l^{(u)} \right] + \mathrm{E}\left[\left(v_n^{(s)} - \overline{v}_n^{(s)} \right) W_l^{(b)*} \right]$$

$$= \sum_{u=1}^{N_S} H_l^{(u,b)*} \left[\sigma_{v,n,s}^2 \delta_{s-u} e^{j2\pi \frac{ln}{N}} \right] + \mathrm{E}\left[v_n^{(s)} - \overline{v}_n^{(s)} \right] \mathrm{E}\left[W_l^{(b)} \right]^*$$

$$= H_l^{(s,b)*} \sigma_{v,n,s}^2 e^{j2\pi \frac{ln}{N}} \quad . \tag{A-25}$$

Furthermore,

$$\mathrm{Cov}\left[V_k^{(s)}, R_l^{(b)} \right] = \mathrm{E}\left[\left(V_k^{(s)} - \overline{V}_k^{(s)} \right) \left(R_l^{(b)} - \overline{R}_l^{(b)} \right)^* \right]$$

$$= \sum_{m=0}^{N-1} \mathrm{E}\left[\left(v_m^{(s)} - \overline{v}_m^{(s)} \right) \left(R_l^{(b)} - \overline{R}_l^{(b)} \right)^* \right] e^{-j2\pi \frac{km}{N}}$$

$$= \sum_{m=0}^{N-1} \mathrm{Cov}\left[v_m^{(s)}, R_l^{(b)} \right] e^{-j2\pi \frac{km}{N}}$$

$$= \sum_{m=0}^{N-1} \left[H_l^{(s,b)*} \sigma_{v,m,s}^2 e^{j2\pi \frac{lm}{N}} \right] e^{-j2\pi \frac{km}{N}}$$

$$= H_l^{(s,b)*} \sum_{m=0}^{N-1} \sigma_{v,m,s}^2 e^{-j2\pi \frac{(k-l)m}{N}} \tag{A-26}$$

and

$$\mathrm{Cov}\left[R_k^{(a)}, R_l^{(b)} \right] = \mathrm{E}\left[\left(R_k^{(a)} - \overline{R}_k^{(a)} \right) \left(R_l^{(b)} - \overline{R}_l^{(b)} \right)^* \right]$$

$$= \mathrm{E}\left[\left(\sum_{s=1}^{N_S} H_k^{(s,a)} V_k^{(s)} + W_k^{(a)} - \sum_{s=1}^{N_S} H_k^{(s,a)} \overline{V}_k^{(s)} \right) \left(R_l^{(b)} - \overline{R}_l^{(b)} \right)^* \right]$$

$$= \sum_{s=1}^{N_S} H_k^{(s,a)} \mathrm{Cov}\left[V_k^{(s)}, R_l^{(b)} \right] + \mathrm{E}\left[W_k^{(a)} \left(R_l^{(b)} - \overline{R}_l^{(b)} \right)^* \right]$$

$$= \sum_{s=1}^{N_S} H_k^{(s,a)} \left(H_l^{(s,b)*} \sum_{m=0}^{N-1} \sigma_{v,m,s}^2 e^{-j2\pi\frac{(k-l)m}{N}} \right)$$

$$+ \sum_{u=1}^{N_s} H_l^{(u,b)*} \mathrm{E}\left[W_k^{(a)} \left(V_l^{(b)} - \overline{V}_l^{(b)} \right)^* \right] + \mathrm{E}\left[W_k^{(a)} W_l^{(b)*} \right]$$

$$= \sum_{s=1}^{N_S} H_k^{(s,a)} H_l^{(s,b)*} \sum_{m=0}^{N-1} \sigma_{v,m,s}^2 e^{-j2\pi\frac{(k-l)m}{N}} + NN_0\delta_{k-l}\delta_{a-b}. \quad \text{(A-27)}$$

Substituting Equation (A-25) and (A-27) into Equation (A-19) gives

$$\frac{1}{N} \sum_{a=1}^{N_A} \widetilde{P}_l^{(s,a)} \left[\sum_{u=1}^{N_S} H_l^{(u,a)} H_l^{(u,b)*} \sum_{m=0}^{N-1} \sigma_{v,m,u}^2 + NN_0\delta_{a-b} \right]$$

$$= \frac{1}{N} \sum_{n=0}^{N-1} \left[H_l^{(s,b)*} \sigma_{v,n,s}^2 e^{j2\pi\frac{ln}{N}} \right] e^{-j2\pi\frac{ln}{N}}, \quad \text{(A-28)}$$

or

$$\sum_{a=1}^{N_A} \widetilde{P}_l^{(s,a)} \left[\sum_{u=1}^{N_S} H_l^{(u,a)} H_l^{(u,b)*} \left(\frac{1}{N} \sum_{m=0}^{N-1} \sigma_{v,m,u}^2 \right) + N_0\delta_{a-b} \right]$$

$$= H_l^{(s,b)*} \frac{1}{N} \sum_{n=0}^{N-1} \sigma_{v,n,s}^2 . \quad \text{(A-29)}$$

Defining $\overline{\sigma}_{v,s}^2 = \frac{1}{N} \sum_{n=0}^{N-1} \sigma_{v,n,s}^2$ as the time-averaged variance of the transmitted symbols gives

$$\sum_{a=1}^{N_A} \widetilde{P}_l^{(s,a)} \left[\sum_{u=1}^{N_S} H_l^{(u,a)} H_l^{(u,b)*} \overline{\sigma}_{v,u}^2 + N_0\delta_{a-b} \right] = H_l^{(s,\,b)*} \overline{\sigma}_{v,s}^2 \quad \text{(A-30)}$$

for all $l \in \{0, 1, ..., N-1\}$ and $b \in \{1, 2, ..., N_A\}$. It is convenient to define

$$\Gamma_l^{(b,a)} = \sum_{u=1}^{N_S} H_l^{(u,a)} H_l^{(u,b)*} \overline{\sigma}_{v,u}^2 + N_0\delta_{a-b}, \quad \text{(A-31)}$$

so for each frequency l, the coefficients $\widetilde{P}_l^{(s,b)}$ are found by solving the N_A equations

$$\begin{bmatrix} \Gamma_l^{(1,1)} & \Gamma_l^{(1,2)} & \cdots & \Gamma_l^{(1,N_A)} \\ \Gamma_l^{(2,1)} & \Gamma_l^{(2,2)} & \cdots & \Gamma_l^{(2,N_A)} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_l^{(N_A,1)} & \Gamma_l^{(N_A,2)} & \cdots & \Gamma_l^{(N_A,N_A)} \end{bmatrix} \begin{bmatrix} \widetilde{P}_l^{(s,1)} \\ \widetilde{P}_l^{(s,2)} \\ \vdots \\ \widetilde{P}_l^{(s,N_A)} \end{bmatrix} = \overline{\sigma}_{v,s}^2 \begin{bmatrix} H_l^{(s,1)*} \\ H_l^{(s,2)*} \\ \vdots \\ H_l^{(s,N_A)*} \end{bmatrix}. \tag{A-32}$$

A.3 The Mean and Variance of $z_n^{(s)}$

For all n and s the MCU calculates the branch metrics

$$\gamma_n^{(s)}(d) = f\left(z_n^{(s)} \middle| d_n^{(s)} = d\right) \tag{A-33}$$

for each hypothetical value $d \in \{0, 1, ..., M-1\}$. Under the widely-used assumption that the residual interference in $z_n^{(s)}$ can be modeled with a Gaussian distribution, the branch metrics can be expressed as

$$\gamma_n^{(s)}(d) = \frac{1}{\pi\widetilde{\sigma}_{z,n,s}^2} \exp\left\{ -\frac{1}{\widetilde{\sigma}_{z,n,s}^2} \left| z_n^{(s)} - \widetilde{\mu}_n^{(s)}(d) \right|^2 \right\}, \tag{A-34}$$

where

$$\widetilde{\mu}_n^{(s)}(d) = \mathrm{E}\left[z_n^{(s)} \middle| d_n^{(s)} = d \right] \tag{A-35}$$

and

$$\widetilde{\sigma}_{z,n,s}^2 = \mathrm{E}\left[\left| z_n^{(s)} - \mu_n^{(s)}(d) \right|^2 \middle| d_n^{(s)} = d \right] \tag{A-36}$$

are respectively the mean and variance of $z_n^{(s)}$ under the hypothesis that $d_n^{(s)} = d$. Calculation of the branch metrics therefore requires expressions for $\widetilde{\mu}_n^{(s)}(d)$ and $\widetilde{\sigma}_{z,n,s}^2$. Finding the mean is relatively straightforward. Using equation (A-9) for $z_n^{(s)}$ gives

$$\widetilde{\mu}_n^{(s)}(d) = \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left(\mathrm{E}\left[R_k^{(a)}\middle|d_n^{(s)}=d\right]-\overline{R}_k^{(a)}\right)e^{j2\pi\frac{kn}{N}}+\overline{v}_n^{(s)}. \qquad \text{(A-37)}$$

To evaluate $\mathrm{E}\left[R_k^{(a)}\middle|d_n^{(s)}=d\right]$, note that

$$\mathrm{E}\left[v_m^{(u)}\middle|d_n^{(s)}=d\right]=\begin{cases}\mathrm{E}\left[v_m^{(u)}\right] & \text{if} \quad m\neq n \quad \text{or} \quad s\neq u\\ \mathrm{E}\left[v_n^{(s)}\middle|d_n^{(s)}=d\right] & \text{if} \quad m=n \ \text{and} \ s=u\end{cases}$$

$$=\begin{cases}\overline{v}_m^{(u)} & \text{if} \quad m\neq n \quad \text{or} \quad s\neq u\\ \mathrm{SM}[d] & \text{if} \quad m=n \ \text{and} \ s=u\end{cases}$$

$$=\overline{v}_m^{(u)}\left(1-\delta_{m-n}\delta_{s-u}\right)+\mathrm{SM}[d]\delta_{m-n}\delta_{s-u}$$

$$=\overline{v}_m^{(u)}+\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)\delta_{m-n}\delta_{s-u} \qquad \text{(A-38)}$$

Using Equation (A-38), we find

$$\mathrm{E}\left[R_k^{(a)}\middle|d_n^{(s)}=d\right]=\sum_{u=1}^{N_S}H_k^{(u,a)}\,\mathrm{E}\left[V_k^{(u)}\middle|d_n^{(s)}=d\right]+\mathrm{E}\left[W_k^{(a)}\middle|d_n^{(s)}=d\right]$$

$$=\sum_{u=1}^{N_S}H_k^{(u,a)}\sum_{m=0}^{N-1}\mathrm{E}\left[v_m^{(u)}\middle|d_n^{(s)}=d\right]e^{-j2\pi\frac{km}{N}}+\mathrm{E}\left[W_k^{(a)}\right]$$

$$=\sum_{u=1}^{N_S}H_k^{(u,a)}\sum_{m=0}^{N-1}\left[\overline{v}_m^{(u)}+\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)\delta_{m-n}\delta_{s-u}\right]e^{-j2\pi\frac{km}{N}}$$

$$=\sum_{u=1}^{N_S}H_k^{(s,a)}\sum_{m=0}^{N-1}\overline{v}_m^{(u)}e^{-j2\pi\frac{km}{N}}$$

$$+\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)\sum_{u=1}^{N_S}H_k^{(u,a)}\sum_{m=0}^{N-1}\delta_{m-n}\delta_{s-u}e^{-j2\pi\frac{km}{N}}$$

$$=\sum_{u=1}^{N_S}H_k^{(s,a)}\overline{V}_k^{(u)}+\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)H_k^{(s,a)}e^{-j2\pi\frac{kn}{N}}$$

$$=\overline{R}_k^{(a)}+\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)H_k^{(s,a)}e^{-j2\pi\frac{kn}{N}}. \qquad \text{(A-39)}$$

Substituting Equation (A-39) into Equation (A-37) gives

$$\begin{aligned}\widetilde{\mu}_n^{(s)}(d) &= \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left[\overline{R}_k^{(a)} + \left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)H_k^{(s,a)}e^{-j2\pi\frac{kn}{N}} - \overline{R}_k^{(a)}\right]e^{j2\pi\frac{kn}{N}} + \overline{v}_n^{(s)} \\ &= \left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}H_k^{(s,a)} + \overline{v}_n^{(s)} \\ &= \left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)\widetilde{q}_0^{(s)} + \overline{v}_n^{(s)} \qquad \text{(A-40)}\end{aligned}$$

where

$$\widetilde{q}_0^{(s)} = \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}H_k^{(s,a)} \quad . \qquad \text{(A-41)}$$

To find the variance, note that

$$\begin{aligned}z_n^{(s)} - \widetilde{\mu}_n^{(s)}(d) &= \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left[R_k^{(a)} - \overline{R}_k^{(a)}\right]e^{j2\pi\frac{kn}{N}} + \overline{v}_n^{(s)} \\ &\qquad - \left[\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)\widetilde{q}_0^{(s)} + \overline{v}_n^{(s)}\right] \\ &= \frac{1}{N}\sum_{k=1}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left[R_k^{(a)} - \overline{R}_k^{(a)}\right]e^{j2\pi\frac{kn}{N}} \\ &\qquad - \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)H_k^{(s,a)} \\ &= \frac{1}{N}\sum_{k=1}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left[R_k^{(a)} - \left(\overline{R}_k^{(a)} + \left(\mathrm{SM}[d]-\overline{v}_n^{(s)}\right)H_k^{(s,a)}e^{-j2\pi\frac{kn}{N}}\right)\right] \\ &\qquad \cdot e^{j2\pi\frac{kn}{N}} \\ &= \frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\widetilde{P}_k^{(s,a)}\left(R_k^{(a)} - \mathrm{E}\left[R_k^{(a)}\middle| d_n^{(s)} = d\right]\right)e^{j2\pi\frac{kn}{N}}, \qquad \text{(A-42)}\end{aligned}$$

where Equation (A-39) was used to give the substitution in the last line. The variance is then

$$\widetilde{\sigma}_{z,n,s}^2 = \mathrm{E}\left[\left|z_n^{(s)} - \widetilde{\mu}_n^{(s)}(d)\right|^2\middle| d_n^{(s)} = d\right]$$

$$= \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} \sum_{l=0}^{N-1} \sum_{b=1}^{N_A} \widetilde{P}_k^{(s,a)} \widetilde{P}_l^{(s,b)*} \operatorname{Cov}\left[R_k^{(a)}, R_l^{(b)} \middle| d_n^{(s)} = d\right] e^{j2\pi \frac{(k-l)n}{N}} \quad \text{(A-43)}$$

where

$$\operatorname{Cov}\left[R_k^{(a)}, R_l^{(b)} \middle| d_n^{(s)} = d\right] = \mathrm{E}\left[\left(R_k^{(a)} - \mathrm{E}\left[R_k^{(a)} \middle| d_n^{(s)} = d\right]\right)\left(R_l^{(b)} - \mathrm{E}\left[R_l^{(b)} \middle| d_n^{(s)} = d\right]\right)^* \middle| d_n^{(s)} = d\right] \quad \text{(A-44)}$$

is the covariance of $R_k^{(a)}$ and $R_l^{(b)}$ conditioned on the hypothesis $d_n^{(s)} = d$. This covariance is given by

$$\begin{aligned} \operatorname{Cov}\left[R_k^{(a)}, R_l^{(b)} \middle| d_n^{(s)} = d\right] &= \sum_{u=1}^{N_S} \sum_{u'=1}^{N_S} H_k^{(u,a)} H_l^{(u',b)*} \operatorname{Cov}\left[V_k^{(u)}, V_l^{(u')} \middle| d_n^{(s)} = d\right] \\ &\quad + \mathrm{E}\left[W_k^{(a)} W_l^{(b)*}\right] \\ &= \sum_{u=1}^{N_S} \sum_{u'=1}^{N_S} H_k^{(u,a)} H_l^{(u',b)*} \sum_{m=0}^{N-1} \sum_{m'=0}^{N-1} \operatorname{Cov}\left[v_m^{(u)} v_{m'}^{(u')} \middle| d_n^{(s)} = d\right] \\ &\quad \cdot e^{-j2\pi \frac{(km-lm')}{N}} + NN_0 \delta_{l-k} \delta_{a-b} . \end{aligned} \quad \text{(A-45)}$$

Due to the independence of different symbol transmitted at different times or by different users,

$$\operatorname{Cov}\left[v_m^{(u)}, v_{m'}^{(u')} \middle| d_n^{(s)} = d\right] = \operatorname{Cov}\left[v_m^{(u)}, v_m^{(u)} \middle| d_n^{(s)} = d\right] \delta_{m-m'} \delta_{u-u'}, \quad \text{(A-46)}$$

so

$$\begin{aligned} \operatorname{Cov}\left[R_k^{(a)}, R_l^{(b)} \middle| d_n^{(s)} = d\right] &= \sum_{u=1}^{N_S} \sum_{u'=1}^{N_S} H_k^{(u,a)} H_l^{(u',b)*} \sum_{m=0}^{N-1} \sum_{m'=0}^{N-1} \operatorname{Cov}\left[v_m^{(u)}, v_{m'}^{(u')} \middle| d_n^{(s)} = d\right] \delta_{m-m'} \delta_{u-u'} e^{-j2\pi \frac{(km-lm')}{N}} \\ &\quad + NN_0 \delta_{l-k} \delta_{a-b} \end{aligned}$$

$$= \sum_{u=1}^{N_S} H_k^{(u,a)} H_l^{(u,b)*} \sum_{m=0}^{N-1} \mathrm{Cov}\left[v_m^{(u)}, v_m^{(u)} \middle| d_n^{(s)} = d \right] e^{-j2\pi \frac{(k-l)m}{N}} + NN_0 \delta_{l-k} \delta_{a-b} .$$

(A-47)

The conditional variance of $v_m^{(u)}$ is

$$\mathrm{Cov}\left[v_m^{(u)}, v_m^{(u)} \middle| d_n^{(s)} = d \right] = \mathrm{E}\left[\left| v_m^{(u)} - \mathrm{E}[v_m^{(u)} \middle| d_n^{(s)} = d] \right|^2 \middle| d_n^{(s)} = d \right]$$

$$= \begin{cases} \mathrm{E}\left[\left| v_m^{(u)} - \bar{v}_m^{(u)} \right|^2 \right] & \text{if } m \neq n \text{ or } s \neq u \\ \mathrm{E}\left[\left| v_n^{(s)} - \mathrm{SM}[d] \right|^2 \middle| d_n^{(s)} = d \right] & \text{if } m = n \text{ and } s = u \end{cases}$$

$$= \begin{cases} \sigma_{v,m,u}^2 & \text{if } m \neq n \text{ or } s \neq u \\ \mathrm{E}\left[\left| \mathrm{SM}[d] - \mathrm{SM}[d] \right|^2 \right] & \text{if } m = n \text{ and } s = u \end{cases}$$

$$= \sigma_{v,m,u}^2 \left(1 - \delta_{m-n} \delta_{s-u} \right) \qquad \text{(A-48)}$$

so

$$\mathrm{Cov}\left[R_k^{(a)}, R_l^{(b)} \middle| d_n^{(s)} = d \right]$$

$$= \sum_{u=1}^{N_S} H_k^{(u,a)} H_l^{(u,b)*} \sum_{m=0}^{N-1} \sigma_{v,m,u}^2 \left(1 - \delta_{m-n} \delta_{s-u} \right) e^{-j2\pi \frac{(k-l)m}{N}} + NN_0 \delta_{l-k} \delta_{a-b} .$$

(A-49)

Substituting Equation (A-49) into Equation (A-43) gives

$$\tilde{\sigma}_{z,n,s}^2 = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} \sum_{l=0}^{N-1} \sum_{b=0}^{N_A} \tilde{P}_k^{(s,a)} \tilde{P}_l^{(s,b)*} \left[\sum_{u=1}^{N_S} H_k^{(u,a)} H_l^{(u,b)*} \sum_{m=0}^{N-1} \sigma_{v,m,u}^2 e^{-j2\pi \frac{(k-l)m}{N}} \right.$$

$$\left. + NN_0 \delta_{l-k} \delta_{a-b} - H_k^{(s,a)} H_l^{(s,b)*} \sigma_{v,n,s}^2 e^{-j2\pi \frac{(k-l)n}{N}} \right] e^{j2\pi \frac{(k-l)n}{N}} .$$

(A-50)

Although it is preferable to have the variance depend on n, it does not seem possible to reduce Equation (A-50) into a form that is computationally feasible. It is therefore useful to use a time-averaged variance,

$$\tilde{\bar{\sigma}}_{z,s}^2 = \frac{1}{N}\sum_{n=0}^{N-1}\tilde{\sigma}_{z,n,s}^2 \tag{A-51}$$

instead. Note that

$$\begin{aligned}
\tilde{\bar{\sigma}}_{z,s}^2 = {} & \frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_k^{(s,a)}\tilde{P}_l^{(s,b)*}\sum_{u=1}^{N_S}H_k^{(u,a)}H_l^{(u,b)*}\sum_{m=0}^{N-1}\sigma_{v,m,u}^2 e^{-j2\pi\frac{(k-l)m}{N}} \\
& \cdot\frac{1}{N}\sum_{n=0}^{N-1}e^{j2\pi\frac{(k-l)n}{N}} \\
& -\frac{1}{N}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\tilde{P}_k^{(s,a)}H_k^{(s,a)}\,\frac{1}{N}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_l^{(s,b)*}H_l^{(s,b)*}\,\frac{1}{N}\sum_{n=0}^{N-1}\sigma_{v,n,s}^2 \\
& +\frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_k^{(s,a)}\tilde{P}_l^{(s,b)*}N_0N\delta_{k-l}\delta_{a-b}\frac{1}{N}\sum_{n=0}^{N-1}e^{j2\pi\frac{(k-l)n}{N}} \\
= {} & \frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_k^{(s,a)}\tilde{P}_l^{(s,b)*}\sum_{u=1}^{N_S}H_k^{(u,a)}H_l^{(u,b)*}\sum_{m=0}^{N-1}\sigma_{v,m,u}^2 e^{-j2\pi\frac{(k-l)m}{N}}\delta_{l-k} \\
& -\left|\tilde{q}_0^{(s)}\right|^2\frac{1}{N}\sum_{n=0}^{N-1}\sigma_{v,n,s}^2 \\
& +\frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{a=1}^{N_A}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_k^{(s,a)}\tilde{P}_l^{(s,b)*}N_0N\delta_{k-l}\delta_{a-b}\delta_{k-l} \\
= {} & \frac{1}{N}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_l^{(s,b)*}\sum_{a=1}^{N_A}\tilde{P}_l^{(s,a)}\sum_{u=1}^{N_S}H_l^{(u,a)}H_l^{(u,b)*}\,\frac{1}{N}\sum_{m=0}^{N-1}\sigma_{v,m,u}^2-\left|\tilde{q}_0^{(s)}\right|^2\frac{1}{N}\sum_{n=0}^{N-1}\sigma_{v,n,s}^2 \\
& +\frac{1}{N}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\tilde{P}_l^{(s,b)*}\sum_{a=1}^{N_A}\tilde{P}_l^{(s,a)}N_0\delta_{a-b}\ .
\end{aligned} \tag{A-52}$$

Substituting

$$\overline{\sigma}_{v,s}^2 = \frac{1}{N}\sum_{n=0}^{N-1}\sigma_{v,n,s}^2 \tag{A-53}$$

gives

$$\widetilde{\overline{\sigma}}_{z,s}^2 = \frac{1}{N}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\widetilde{P}_l^{(s,b)*}\sum_{a=1}^{N_A}\widetilde{P}_l^{(s,a)}\left[\sum_{u=1}^{N_S}H_l^{(u,a)}H_l^{(u,b)*}\overline{\sigma}_{v,u}^2 + N_0\delta_{a-b}\right] - \left|\widetilde{q}_0^{(s)}\right|^2\overline{\sigma}_{v,s}^2 \ . \tag{A-54}$$

Applying Equation (A-30) gives

$$\begin{aligned}\widetilde{\overline{\sigma}}_{z,s}^2 &= \frac{1}{N}\sum_{l=0}^{N-1}\sum_{b=1}^{N_A}\widetilde{P}_l^{(s,b)*}\left[H_l^{(s,b)*}\overline{\sigma}_{v,s}^2\right] - \left|\widetilde{q}_0^{(s)}\right|^2\overline{\sigma}_{v,s}^2 \\ &= \widetilde{q}_0^{(s)*}\overline{\sigma}_{v,s}^2 - \left|\widetilde{q}_0^{(s)}\right|^2\overline{\sigma}_{v,s}^2 \\ &= \overline{\sigma}_{v,s}^2\widetilde{q}_0^{(s)*}\left(1-\widetilde{q}_0^{(s)}\right)\end{aligned} \tag{A-55}$$

for the time-average variance of $z_n^{(s)}$.

A.4 Numerically Stable Version

According to Equation (A-32), the filter coefficients are directly proportioned to $\overline{\sigma}_{v,s}^2$. If $\overline{\sigma}_{v,s}^2$ is very small, the filter coefficients are also small, as is the filter output. Since, when calculating the branch metrics we divide the filter output by $\widetilde{\overline{\sigma}}_{z,s}^2$, which is also proportional to $\overline{\sigma}_{v,s}^2$, we end up with a numerical instability problem due to multiplication followed by division of a small number, leading to large rounding errors. To avoid thus problem we instead calculate the filter coefficients according to

$$\begin{bmatrix} \Gamma_l^{(1,1)} & \Gamma_l^{(1,2)} & \cdots & \Gamma_l^{(1,N_A)} \\ \Gamma_l^{(2,1)} & \Gamma_l^{(2,2)} & \cdots & \Gamma_l^{(2,N_A)} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_l^{(N_A,1)} & \Gamma_l^{(N_A,2)} & \cdots & \Gamma_l^{(N_A,N_A)} \end{bmatrix} \begin{bmatrix} P_l^{(s,1)} \\ P_l^{(s,2)} \\ \vdots \\ P_l^{(s,N_A)} \end{bmatrix} = \begin{bmatrix} H_l^{(s,1)*} \\ H_l^{(s,2)*} \\ \vdots \\ H_l^{(s,N_A)*} \end{bmatrix} \tag{A-56}$$

which is the same as Equation (A-32) except that the factor of $\overline{\sigma}_{v,s}^2$ has been removed from the right hand side. The original filter coefficients could then be calculated by $\widetilde{P}_l^{(s,a)} = \overline{\sigma}_{v,s}^2 P_l^{(s,a)}$, but doing so is not necessary.

Instead, we define the filter output as

$$z_n^{(s)} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} P_k^{(s,a)} \left(R_k^{(a)} - \overline{R}_k^{(a)} \right) e^{j2\pi \frac{kn}{N}} + q_0^{(s)} \overline{v}_n^{(s)} \tag{A-57}$$

where

$$q_0^{(s)} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{a=1}^{N_A} P_k^{(s,a)} H_k^{(s,a)} \,. \tag{A-58}$$

For calculating the branch metrics, the conditional mean is

$$\mu_n^{(s)}(d) = q_0^{(s)} \, \mathrm{SM}[d] \tag{A-59}$$

and variance is

$$\overline{\sigma}_{z,s}^2 = q_0^{(s)*} \left(1 - \overline{\sigma}_{v,s}^2 q_0^{(s)} \right). \tag{A-60}$$

The resulting branch metrics are identical to the ones calculated according to the formula given in Appendix A.3, but without the risk of numerical instability.