COMMUNICATION SCIENCES INSTITUTE

Decoding Error-Correction Codes Utilizing Bit-Error Probability Estimates

by

Gregory Dubney

CSI-06-01-01

USC VITERBI SCHOOL OF ENGINEERING
UNIVERSITY OF SOUTHERN CALIFORNIA
ELECTRICAL ENGINEERING — SYSTEMS
LOS ANGELES, CA 90089-2565

DECODING ERROR-CORRECTION CODES UTILIZING BIT-ERROR PROBABILITY ESTIMATES

by

Gregory Oleg Dubney

A Dissertation Presented to the
FACULTY OF THE GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(ELECTRICAL ENGINEERING)

August 2005

Dedication

First and foremost, this dissertation is dedicated to my amazing wife, Gia, for her love and support, and for the sacrifices she has made that enabled me to earn my Ph.D. degree. She is my soul-mate, my partner, and my best friend, who walks besides me in this journey called life. I love her very deeply with all of my heart and soul.

This dissertation is also dedicated to my sons, Alexander and Zachary, both of whom, I love so very dearly. I am proud to be their father, and they have taught me the true meaning of life.

Finally, this dissertation is dedicated to my grandparents, Deda and Baba, for the love, support, and encouragement they have given me throughout my life. I love them dearly, and I miss them tremendously.

Acknowledgements

I would like to express my most profound gratitude to my advisor, Professor Irving Reed, for his patience and guidance during my Ph.D. program at the University of Southern California (USC). Working with Professor Reed for the last five years has been one of the greatest experiences of my life. Dr. Reed is a brilliant coding theorist, a world renowned scholar, a great mathematician and engineer, a phenomenal mentor, and, above all, a world class gentleman. I have benefited tremendously from his knowledge in scientific research, world history, and life. It has been an honor and a privilege to be Professor Reed's last Ph.D. student, and I will always strive to emulate his excellence.

I would also like to profusely thank Professor Remigijus Mikulevicius for all of his help and guidance with this thesis. He always made time to meet with me and answer all of my questions. As a result, I have greatly benefitted from his vast knowledge of mathematics and his research experience. I also want to thank Professor Charles Weber for his help with the synchronization part of this thesis, and more importantly, for his friendship and sage advice during my graduate studies. I would like to take this opportunity to thank Dr. Trieu-Kein Truong for introducing me to the exciting field of error-control coding. His friendship and suggestions helped me throughout every phase of my Ph.D. program. Finally, I would like to thank Professor Richard Leahy for overseeing my matriculation into the Ph.D. program, and for always listening to my concerns and giving me guidance.

My deepest appreciation and thanks are dedicated to my friends: Dr. Chee-Cheon Chui, Dr. Petros Elia, Dr. Ruhua He, Dr. Yuankai Wang, Dr. Jun Yang, and Dr. Guangcai Zhou for their help and friendship during my Ph.D. program. I would especially like to thank Dr. Jun Yang, Dr. Yuankai Wang, and Dr. Petros Elia for their help in

our course studies and with my research. I have benefited greatly from many hours of stimulating discussions with them, and they have enriched my understanding of wireless communications, error-control coding, and world politics.

Finally, I would like to thank to Milly Montenegro, Mayumi Thrasher, Gerrielyn Ramos, Diane Demetras, and Tim Boston for their administrative help and, more importantly, for the great friendship we shared here at USC. I will miss them all.

Contents

List of Tables

List of Figures

Abstract

The main purpose of this dissertation is to estimate the individual bit-error probabilities of binary symbols or codewords while they are being received. It turns out that the bit-error probabilities are a function of the received-bit amplitudes and the channel noise power, both of which are assumed to be unknown a-priori at the receiver. In this study coherent detection is implemented with a Costas phase-locked loop receiver which facilitates the joint estimation of these two parameters, and as a consequence, the bit-error probabilities. The bit-error probability estimates make it possible to reduce the decoding complexity and improve the performance of various error-correction codes.

One example is reliability-based decoding of cyclic codes. The traditional algebraic techniques to decode these codes up to true minimum distance are difficult and computationally complex. In the reliability-based decoding strategy, the bit-error probability estimates are utilized to cancel one error in the received word, and then a less complex algebraic decoding algorithm is used to correct the remaining errors. Simulation results show that the reliability-search algorithm significantly reduces the decoding time of the (23,12,7) Golay code and the (47,24,11) Quadratic Residue (QR) code by 41.3% and 22.2% respectively, compared with other decoding algorithms.

Another example is erasure decoding of Reed-Solomon (RS) codes. An erasure is a symbol error with its location known to the decoder. However, in current decoding procedures, the locations of the error symbols are unknown and must be estimated. Knowledge of the bit-error probabilities enable the decoder to determine the symbols that have the highest probability of error. Thus, the decoder only needs to calculate the amplitudes of

these symbol erasures to decode the codeword. This erasure decoding technique was implemented on a (255,223) RS code. Simulation results show that correcting combinations of errors and erasures results in a slightly lower symbol-error probability compared with other decoding algorithms that correct errors only.

Chapter 1

Introduction

1.1 Motivation

The term "error-control coding" implies a technique by which redundant symbols are attached "intelligently" to information messages by an error correction encoder in the transmitter. These redundancy symbols are used to correct the erroneous data at the error control decoder in the receiver. In other words error-control coding is achieved by restrictions placed on the characteristics of the encoder of the system. These restrictions make it possible for the decoder to correctly extract the original source signal with high reliability and fidelity from the possibly corrupted received or retrieved signals.

The purpose of the research conducted in this dissertation is to be able to estimate the individual bit-error probabilities of binary symbols or codewords while they are being received. It turns out that the bit or symbol error probability of a codeword is a function of the received-bit amplitudes A and the channel noise power σ^2, both of which are assumed to be unknown a-priori at the receiver. In this study coherent detection is implemented with Costas phase-locked loop receiver which facilitates the joint estimation of these two parameters, and as a consequence, the bit-error probabilities. In Chapter 6 it is shown how the individual bit-error probability estimates reduce the decoding complexity of the (23,12,7) Golay code and the (47,24,11) QR code. It is also shown how the bit-error probability estimates facilitate erasure decoding of Reed-Solomon codes over an additive white Gaussian noise (AWGN) channel.

The (23,12,7) Golay code is a perfect linear error-correcting code that can correct all patterns of three or fewer errors in 23 bit positions. A simple BCH decoding algorithm, given in [1, pg. 19], can decode the (23,12,7) Golay code provided there are no more than two errors. The shift-search algorithm, developed by Reed [2], sequentially inverts the information bits until the third error is canceled. It then utilizes the BCH decoding algorithm to correct the remaining two errors. However, the computational complexity is very high to decode the third error in terms of CPU time. In this dissertation a simplified decoding algorithm, called the reliability-search algorithm, is proposed. This algorithm uses bit-error probability estimates to cancel the third error, and then uses the BCH decoding algorithm to correct the remaining two errors. Simulation results show that this new algorithm significantly reduces the decoding complexity for correcting the third error while maintaining the same BER performance.

Another interesting class of error-control codes are the quadratic residue codes. These codes are near half rate, have very good mathematical structure, and have a high capability to correct random errors in data. Recently, He, Reed, and Truong [3] have developed an algebraic decoding algorithm for the (47,24,11) Quadratic Residue code that can correct up to true minimum distance i.e., five errors. However, the computational complexity is very high to decode the fifth error in terms of CPU time. The reason for this can be explained as follows: The algebraic decoding algorithm needs to calculate the greatest common divisor (gcd) of two polynomials for both the four and five error cases. For the four error case, the decoder must calculate the $\gcd(f_1, f_2)$, where $\deg(f_1) = 3$ and $\deg(f_2) = 33$. For this case, the complexity of $\gcd(f_1, f_2)$ is acceptable. However, for the five error case, the decoder must calculate $\gcd(g_1, g_2)$, where $\deg(g_1) = 34$ and $\deg(g_2) = 258$. Clearly, the complexity to compute $\gcd(g_1, g_2)$ is very high and considerable more memory is needed. In order to circumvent calculating the greatest common divisor of such high degree polynomials, the reliability-search algorithm can be utilized to decode the fifth error. Simulation results show that the decoding complexity is significantly reduced for correcting the fifth error compared to the algebraic decoding algorithm.

Reed-Solomon codes have many communication and memory applications and are very powerful in dealing with burst errors. Reed-Solomon codes also allow for a very efficient means of erasure decoding. An erasure is a symbol error with its location known to the decoder. However, in current or traditional decoding procedures, the locations of the error symbols are not known a-priori and must be estimated. Knowledge of the bit-error probabilities enable the decoder to determine, with considerable confidence, the symbols that have the highest likelihood of being in error. Once the locations of the most likely symbol errors are known, the decoder only needs to calculate the amplitudes of these symbol erasures to decode the codeword. Simulations were conducted on a (255,223) Reed-Solomon code over an additive white Gaussian noise (AWGN) channel to determine the accuracy of the symbol erasure estimate. Error/erasure decoding was performed using a combination of 15 errors and 2 erasures and compared to error only decoding for 16 errors using the Berlekamp-Massey algorithm. The performance of the error/erasure decoder yielded a slightly lower symbol error probability compared to the error only decoder.

1.2 Organization of the Dissertation

This dissertation consists of seven chapters with five appendices and is organized as follows:

Chapter 1 is the introduction and the motivation for conducting the research in this dissertation. Chapter 2 establishes a rigorous mathematical development for the spectral representation of a real wide-sense stationary stochastic noise process, and develops the fundamental results of narrow-band stochastic processes. These concepts are fundamental to understanding the estimation methods used in the study.

The first purpose of Chapter 3 is to analyze the Costas phase-locked loop under the strict requirement of perfect phase lock, that is, no phase error. It is shown that the correlation of the received signal with a coherent reference signal yields the following baseband representation: Signal corrupted with Gaussian noise at the output of the in-phase (I) channel, and Gaussian noise only with no signal at the output of the quadrature (Q) channel. The second purpose of this chapter is to analyze the Costas circuit when

perfect phase-lock is not achieved. In Section 3.4, a new analytical expression for the phase-error variance is derived when there is a mismatch between the modulation bandwidth and the RF filter bandwidth in the receiver. The last section uses the phase-error variance to ascertain the fraction of signal and noise power that leaks from the in-phase channel into the quadrature channel, and the fraction of noise power that leaks from the quadrature channel into the in-phase channel. The main purpose of Chapter 4 is to estimate the individual bit-error probabilities of binary symbols or codewords while they are being received. It turns out that the bit or symbol error probability of a codeword is a function of the received-bit amplitudes A and the channel noise power σ^2, both of which are assumed to be unknown a-priori at the receiver. In this study coherent detection is implemented with Costas phase-locked loop receiver which facilitates the joint estimation of these two parameters, and as a consequence, the bit-error probability.

Chapter 5 is devoted to the mathematical preliminaries and the general properties of error-control coding. The basic concepts of groups, rings, fields, primitive elements, cyclic codes, Reed-Solomon codes, syndrome equations, and the Berlekamp-Massey (BM) algorithm are discussed. In Chapter 6 it is shown how the individual bit-error probability estimates reduce the decoding complexity of the (23,12,7) Golay code and the (47,24,11) Quadratic Residue code. It is also shown how the bit-error probability estimates facilitate erasure decoding of Reed-Solomon codes over an additive white Gaussian noise (AWGN) channel.

Finally, the main results of this dissertation are summarized in Chapter 7, and some future areas of research are proposed. The appendices give the detailed proofs of the facts stated in the dissertation.

Chapter 2

Spectral Representation of Wide-Sense Stationary Processes

The primary purpose of this chapter is to establish a rigorous mathematical development for the spectral representation of a real wide-sense stationary stochastic noise process. Following this development, the original definition of the one-sided power spectral density is reviewed and related to the more modern two-sided definition. In Section 2.3 the spectral representation of a real noise process is extended to a complex-valued noise process. Consequently, this complex process is used in Section 2.4 to develop the fundamental results of narrow-band stochastic processes. Finally, a specialization is made to the Gaussian noise process where the probability density function for two jointly Gaussian random variables is derived.

2.1 The Real Process

Let $x(t)$ be a real wide-sense stationary (WSS) stochastic noise process with zero mean and continuous two-sided power spectral density $G(f)$. From the theories of shot and thermal noise, it can be assumed that such a process, with a suitable choice for $G(f)$, can represent the background noise in a radio receiver. The correlation function of $x(t)$ is defined by

$$R(\tau) \triangleq \mathbf{E}\{x(t+\tau)x(t)\} = \int_{-\infty}^{\infty} G(f)e^{i2\pi f\tau}df, \tag{2.1}$$

where $\mathbf{E}\{\cdot\}$ is the statistical expectation operator. The power spectral density is the Fourier transform of the correlation function, defined by

$$G(f) = \int_{-\infty}^{\infty} R(\tau) e^{-i2\pi f\tau} d\tau, \tag{2.2}$$

and has the interpretation of being the density of average power distribution per unit frequency (Wong and Hajek [4, pg. 92]). The goal of this section is to prove that the spectral representation of any WSS stochastic process $x(t)$ can be represented in the form of the following inverse Fourier-Stieltjes integral,

$$x(t) = \int_{-\infty}^{\infty} e^{i2\pi ft} dX(f), \tag{2.3}$$

where $X(f)$ is a stochastic process of orthogonal increments and where equality holds with probability one (Doob [5, pg. 527], Yaglom [6, pg. 38]).

In many engineering situations, finite energy signals are analyzed in the frequency domain because linear system analysis is often more tractable in the frequency than in the time domain. To analyze stochastic processes in the frequency domain, one would like to compute the Fourier transform of $x(t)$ provided it exists. However, there is no way to rigorously define the integral

$$\int_{-\infty}^{\infty} x(t) e^{-i2\pi ft} dt, \tag{2.4}$$

because $x(t)$ could be a non-integrable function and the integral in (2.4) might not exist. Even though the integral in (2.4) is not well defined, the integrated spectrum of $x(t)$, namely,

$$X(f) = \lim_{T\to\infty} \int_{-T}^{T} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t) dt, \tag{2.5}$$

is well defined and does exist as a limit in the mean-square sense (Yaglom [6, pg. 39]). The proof is given in Appendix A and is similar to the one given in Yaglom with more mathematical details (Yaglom [6, pp. 36-43]).

It is first shown that $X(f)$ is a process of orthogonal increments. Recall that a process $X(f)$ is said to have orthogonal increments if

$$\mathbf{E}\{|X(f_1) - X(f_2)|^2\} < \infty \tag{2.6}$$

and, if whenever the frequencies f_1, f_2, f_3, f_4 satisfy the inequality, $f_1 < f_2 < f_3 < f_4$, the correlation of the two finite increments, $[X(f_1) - X(f_2)]$ and $[X(f_3) - X(f_4)]$, satisfy the relation

$$\mathbf{E}\{[X(f_1) - X(f_2)][X(f_3) - X(f_4)]^*\} = 0, \tag{2.7}$$

where " $*$ " denotes complex conjugation (Doob [5, pg. 99]). Since $X(f)$ exists as a limit in the mean-square sense, it is straightforward to show that equation (2.6) is satisfied. In order to verify equation (2.7), first express the integral in (2.5) more simply by

$$X(f) = \int_{-\infty}^{\infty} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t)dt, \tag{2.8}$$

where it exists in the same sense that equation (2.5) exists - as a limit in the mean-square sense. Consider a finite increment in $X(f)$, given by

$$X(f_1) - X(f_2) = \int_{-\infty}^{\infty} \frac{e^{-i2\pi f_1 t} - e^{-i2\pi f_2 t}}{-2\pi it} x(t)dt. \tag{2.9}$$

Now define the real square pulse function by

$$\Phi_{f_1,f_2}(f) \triangleq \begin{cases} 1, \ f_1 < f < f_2 \ , \\ 0, \text{otherwise.} \end{cases}$$

The inverse Fourier transform of $\Phi_{f_1,f_2}(f)$ is given by

$$\int_{-\infty}^{\infty} \Phi_{f_1,f_2}(f) e^{i2\pi ft} df = \frac{e^{i2\pi f_2 t} - e^{i2\pi f_1 t}}{2\pi it}, \tag{2.10}$$

and thus, the Fourier transform of (2.10) is

$$\begin{aligned}\Phi_{f_1,f_2}(f) &= \int_{-\infty}^{\infty} \frac{e^{i2\pi f_2 t} - e^{i2\pi f_1 t}}{2\pi i t} e^{-i2\pi f t} dt \\ &= \int_{-\infty}^{\infty} \frac{e^{-i2\pi f_2 t} - e^{-i2\pi f_1 t}}{-2\pi i t} e^{i2\pi f t} dt, \end{aligned} \tag{2.11}$$

since $\Phi_{f_1,f_2}(f) = \Phi^*_{f_1,f_2}(f)$. Thus, a substitution of (2.9) into (2.7) yields

$$\begin{aligned} & \mathbf{E}\{[X(f_1) - X(f_2)][X(f_3) - X(f_4)]^*\} \\ = & \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left(\frac{e^{-i2\pi f_1 t} - e^{-i2\pi f_2 t}}{-2\pi i t}\right)\left(\frac{e^{-i2\pi f_3 t'} - e^{-i2\pi f_4 t'}}{-2\pi i t'}\right)^* \mathbf{E}\{x(t)x(t')\} dt dt' \\ = & \int_{-\infty}^{\infty} G(f) \int_{-\infty}^{\infty} \frac{e^{-i2\pi f_1 t} - e^{-i2\pi f_2 t}}{-2\pi i t} e^{i2\pi f t} dt \left(\int_{-\infty}^{\infty} \frac{e^{-i2\pi f_3 t'} - e^{-i2\pi f_4 t'}}{-2\pi i t'} e^{i2\pi f t'} dt'\right)^* df \\ = & \int_{-\infty}^{\infty} G(f)\Phi_{f_1,f_2}(f)\Phi^*_{f_3,f_4}(f) df \\ = & \ 0, \end{aligned} \tag{2.12}$$

where $[(f_1, f_2) \cap (f_3, f_4)] = \varnothing$, the empty set. Therefore, $X(f)$ is a stochastic process of orthogonal increments for $-\infty < f < \infty$.

It is now shown that $x(t)$ can be represented by the inverse Fourier-Stieltjes integral in (2.3) with probability one. To facilitate this goal, it is sufficient to show that for all t

$$\mathbf{E}\left|x(t) - \int_{-\infty}^{\infty} e^{i2\pi f t} dX(f)\right|^2 = 0. \tag{2.13}$$

Squaring and expanding equation (2.13) yields

$$\begin{aligned}\mathbf{E}\left|x(t) - \int_{-\infty}^{\infty} e^{i2\pi f t} dX(f)\right|^2 &= \mathbf{E}\{|x(t)|^2\} + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{i2\pi(f-f')t} \mathbf{E}\{dX(f)dX^*(f')\} \\ &\quad - \int_{-\infty}^{\infty} e^{i2\pi f t} \mathbf{E}\{x^*(t)dX(f)\} - \int_{-\infty}^{\infty} e^{i2\pi f' t} \mathbf{E}\{x(t)dX^*(f')\}. \end{aligned} \tag{2.14}$$

The first term in (2.14) is simply the total power in the process $x(t)$ which is obtained by integrating the power spectral density over all values of f, given by

$$\mathbf{E}\{|x(t)|^2\} = \int_{-\infty}^{\infty} G(f)df. \tag{2.15}$$

In order to evaluate the second term, define a differential increment in the process $X(f)$ as $dX(f) = X(f + df) - X(f)$. It can be shown, using the same techniques to derive equation (2.12), that the correlation of two differential increments is given by

$$\mathbf{E}\{dX(f)d^*X(f^{'})\} = \begin{cases} G(f)df, f = f^{'} \\ 0, f \neq f^{'}. \end{cases} \tag{2.16}$$

A substitution of (2.16) into the the second term in (2.14) yields

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{i2\pi(f-f^{'})t}\mathbf{E}\{dX(f)dX^*(f^{'})\} = \int_{-\infty}^{\infty} G(f)df. \tag{2.17}$$

To facilitate the analysis of the third term, first compute $\mathbf{E}\{x^*(t)dX(f)\}$ as follows:

$$\begin{aligned}
\mathbf{E}\{x^*(t)dX(f)\} &= \mathbf{E}\{x^*(t)[X(f+df) - X(f)]\} \\
&= \int_{-\infty}^{\infty} \frac{e^{-i2\pi(f+df)t^{'}} - e^{-i2\pi ft^{'}}}{-2\pi it^{'}}\mathbf{E}\{x^*(t)x(t^{'})\}dt^{'} \\
&= \int_{-\infty}^{\infty} \frac{e^{-i2\pi(f+df)t^{'}} - e^{-i2\pi ft^{'}}}{-2\pi it^{'}} \int_{-\infty}^{\infty} G(f^{'})e^{i2\pi f^{'}(t^{'}-t)}df^{'}dt^{'} \\
&= \int_{-\infty}^{\infty} G(f^{'})e^{-i2\pi f^{'}t}\left(\int_{-\infty}^{\infty} \frac{e^{-i2\pi(f+df)t^{'}} - e^{-i2\pi ft^{'}}}{-2\pi it^{'}}e^{i2\pi f^{'}t^{'}}dt^{'}\right)df^{'} \\
&= \int_{-\infty}^{\infty} G(f^{'})e^{-i2\pi f^{'}t}\Phi_{f+df,f}(f^{'})df^{'} \\
&= G(f)e^{-i2\pi ft}df.
\end{aligned} \tag{2.18}$$

A substitution of (2.18) into the the third term in (2.14) yields

$$\int_{-\infty}^{\infty} e^{i2\pi ft}\mathbf{E}\{x^*(t)dX(f)\} = \int_{-\infty}^{\infty} G(f)df. \tag{2.19}$$

A similar computation for the fourth term is shown to be identical to the value given in (2.19). Thus, a substitution of (2.15), (2.17), and (2.19) into equation (2.14), yields

$$\mathbf{E}\big|x(t) - \int_{-\infty}^{\infty} e^{i2\pi ft} dX(f)\big|^2 = 2\int_{-\infty}^{\infty} G(f)df - 2\int_{-\infty}^{\infty} G(f)df = 0.$$

Therefore, $x(t)$ is given by

$$x(t) = \int_{-\infty}^{\infty} e^{i2\pi ft} dX(f),$$

where equality holds with probability one.

2.2 Original Definition of Power Spectral Density

In this section the relationship between the original one-sided definition of the power spectral density (PSD) and the more modern two-sided definition is discussed. This relationship is needed in the next section on complex stochastic processes. The power spectral density or power spectrum $G(f)$, given in (2.2), is the modern two-sided definition. The relationship between the original one-sided spectral density, denoted by $G_1(f)$, and the two-sided spectral density is obtained from the one-sided cosine transform as follows: (Larson and Uhlenbeck, [7, pg. 40])

$$\begin{aligned} R(\tau) &\triangleq \int_0^{\infty} G_1(f)\cos(2\pi f\tau)df \\ &= \frac{1}{2}\int_0^{\infty} G_1(f)(e^{i2\pi f\tau} + e^{-i2\pi f\tau})df \\ &= \frac{1}{2}\int_0^{\infty} G_1(f)e^{i2\pi f\tau}df + \frac{1}{2}\int_{-\infty}^{0} G_1(-f)e^{i2\pi f\tau}df \\ &= \int_{-\infty}^{\infty} G(f)e^{i2\pi f\tau}df. \end{aligned} \tag{2.20}$$

If one equates the first and last expressions for $R(\tau)$, given in (2.20), then the two-sided spectral density is related to the one-sided spectral density as follows:

$$G(f) = \begin{cases} \frac{1}{2}G_1(f), & \text{if } f > 0 \\ \frac{1}{2}G_1(-f), & \text{if } f < 0, \end{cases} \tag{2.21}$$

where $G_1(f)$ and $G(f)$ are the original one-sided and modern two-sided definition of power spectral density, respectively. The graphical interpretation of the one and two-sided power spectral density for an arbitrary stochastic process is illustrated in Figure 2.1.

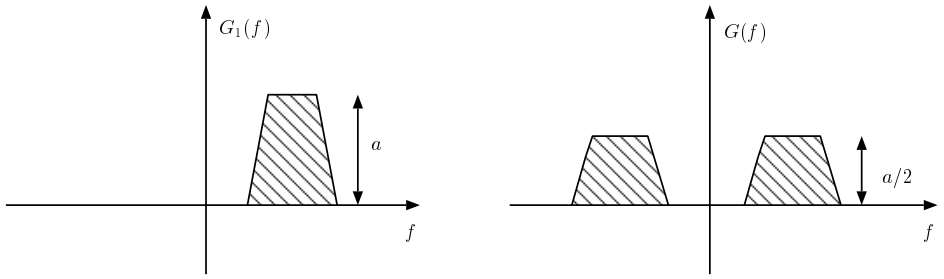

Figure 2.1: One and two-sided power spectral density for an arbitrary process

2.3 The Complex Process $z(t)$, Associated with the Real Process $x(t)$

The complex WSS stochastic process $z(t)$, associated with the real process $x(t)$, is defined by the one-sided spectral representation,

$$z(t) = 2\int_0^\infty e^{i2\pi ft} dX(f). \tag{2.22}$$

For this definition to be consistent with the real process, it is demonstrated next that

$$x(t) = \mathrm{Re}\{z(t)\} = \frac{z(t) + z^*(t)}{2}. \tag{2.23}$$

To proceed with the proof, first note that the complex conjugate of $z(t)$ is given by

$$z^*(t) = 2\int_0^\infty e^{-i2\pi ft} dX^*(f). \tag{2.24}$$

The differential element $dX^*(f)$ is calculated from the conjugate of the $X(f)$ in equation (2.8) as follows:

$$\begin{aligned} X^*(f) &= \left(\int_{-\infty}^{\infty} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t)dt\right)^* \\ &= \int_{-\infty}^{\infty} \frac{e^{i2\pi ft} - 1}{2\pi it} x(t)dt \\ &= \int_{-\infty}^{\infty} \frac{e^{-i2\pi(-f)t} - 1}{2\pi it} x(t)dt \\ &= -X(-f). \end{aligned} \tag{2.25}$$

By (2.25), a substitution of the differential $dX^*(f) = -dX(-f)$ into (2.24) yields

$$\begin{aligned} z^*(t) &= 2\int_0^\infty e^{-i2\pi ft}[-dX(-f)] \\ &= 2\int_{-\infty}^0 e^{i2\pi ft} dX(f). \end{aligned}$$

Finally, a substitution of $z(t)$ and $z^*(t)$ into equation (2.23) demonstrates that

$$x(t) = \int_{-\infty}^{\infty} e^{i2\pi ft} dX(f). \tag{2.26}$$

Now that the real part of $z(t)$ is well understood, a natural extension is to obtain an expression for the imaginary part of $z(t)$. In a similar manner as above, the imaginary part of $z(t)$ is calculated as follows:

$$\begin{aligned} y(t) = \mathrm{Im}\{z(t)\} &= \frac{z(t) - z^*(t)}{2i} \\ &= \frac{1}{i}\left[\int_0^\infty e^{i2\pi ft} dX(f) - \int_{-\infty}^0 e^{i2\pi ft} dX(f)\right] \\ &= -i\int_{-\infty}^\infty e^{i2\pi ft}\mathrm{sgn}(f) dX(f) \\ &= \int_{-\infty}^\infty e^{i2\pi ft} H(f) dX(f), \end{aligned} \tag{2.27}$$

where $H(f)$ is the Hilbert transform filter, defined by

$$H(f) \triangleq -i\,\mathrm{sgn}(f),$$

and $\mathrm{sgn}(\cdot)$ is the signum function, defined by

$$\mathrm{sgn}(f) = \begin{cases} +1, f \geq 0 \\ -1, f < 0. \end{cases}$$

A substitution $x(t)$ and $y(t)$, given in (2.26) and (2.27) respectively, into the expression $x(t) + iy(t)$ yields

$$\begin{aligned} x(t) + iy(t) &= \int_{-\infty}^\infty e^{i2\pi ft} dX(f) + i\int_{-\infty}^\infty e^{i2\pi ft} H(f) dX(f) \\ &= \int_{-\infty}^0 e^{i2\pi ft} dX(f) + \int_0^\infty e^{i2\pi ft} dX(f) \\ &\quad - \int_{-\infty}^0 e^{i2\pi ft} dX(f) + \int_0^\infty e^{i2\pi ft} dX(f) \\ &= 2\int_0^\infty e^{i2\pi ft} dX(f) \\ &= z(t). \end{aligned}$$

Therefore, one can write the complex stochastic process $z(t)$ as

$$z(t) = x(t) + iy(t), \tag{2.28}$$

where $x(t)$ and $y(t)$ are real WSS stochastic processes.

To gain further insight and knowledge about the complex process $z(t)$, the mean and the four second moments of $z(t)$ are computed below:

1. Since $x(t)$ and $y(t)$ are both zero mean processes, clearly $z(t)$ is also a zero mean process.

2. The complex cross-correlation function is calculated using equation (2.16) as follows:

$$\begin{aligned} R_{zz^*}(\tau) &= \mathbf{E}\{z(t+\tau)z^*(t)\} \\ &= 4\int_0^\infty \int_0^\infty e^{i2\pi f\tau} e^{i2\pi(f-f')t} \mathbf{E}\{dX(f)dX^*(f')\} \\ &= 4\int_0^\infty G(f)e^{i2\pi f\tau} df. \end{aligned} \tag{2.29}$$

Define

$$G_z(f) = \begin{cases} 4G(f), f > 0 \\ 0, f < 0, \end{cases} \tag{2.30}$$

so that $R_{zz^*}(\tau)$ can be expressed as

$$R_{zz^*}(\tau) = \int_{-\infty}^{\infty} G_z(f)e^{i2\pi f\tau} df. \tag{2.31}$$

This result reveals the important fact that, for all frequencies, the power spectral density $G_z(f)$ of a complex process $z(t)$ is four times as large as the two-sided power spectral density $G(f)$ of the corresponding real process $x(t)$. The graphical representation of the power spectral densities of $x(t)$ and $z(t)$ are shown in Figure 2.2, respectively.

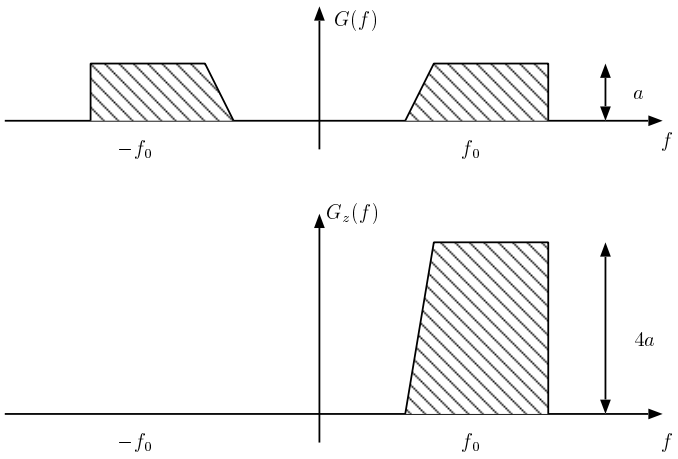

Figure 2.2: Power spectral densities of the real and complex process

3. The unconjugated autocorrelation function is given by

$$\begin{aligned} R_{zz}(t,t^{'}) &= \mathbf{E}\{z(t)z(t^{'})\} \\ &= 4\mathbf{E}\int_0^\infty\int_0^\infty e^{i2\pi(ft+f^{'}t^{'})}dX(f)dX(f^{'}) \\ &= 4\int_0^\infty\int_0^\infty e^{i2\pi(ft+f^{'}t^{'})}\mathbf{E}\{dX(f)dX(f^{'})\}, \end{aligned}$$

where $\mathbf{E}\{dX(f)dX(f^{'})\}$ is calculated as follows:

$$\begin{aligned} \mathbf{E}\{dX(f)dX(f^{'})\} &= \mathbf{E}\{[X(f_1)-X(f_2)][X(f_3)-X(f_4)]\} \\ &= \int_{-\infty}^\infty G(f)\left(\int_{-\infty}^\infty \frac{e^{-i2\pi f_1 t}-e^{-i2\pi f_2 t}}{-2\pi i t}e^{i2\pi ft}dt\right) \\ &\times \left(\int_{-\infty}^\infty \frac{e^{-i2\pi(-f_3)t^{'}}-e^{-i2\pi(-f_4)t^{'}}}{2\pi i t^{'}}e^{i2\pi ft^{'}}dt^{'}\right)^* df \\ &= -\int_{-\infty}^\infty G(f)\Phi_{f_1,f_2}(f)\Phi^*_{-f_3,-f_4}(f)df \\ &= 0, \end{aligned}$$

for all $f,\ f^{'} > 0$ or for all $f, f^{'} < 0$. Hence, the value of the unconjugated autocorrelation is given by

$$R_{zz}(t, t^{'}) = 0. \tag{2.32}$$

4. From equation (2.29), it is easily seen that

$$R_{z^*z}(\tau) = R^*_{zz^*}(\tau).$$

5. Finally, following the same steps in the proof of equation (2.32), it is not difficult to show that the conjugated autocorrelation function is given by

$$R_{z^*z^*}(t, t^{'}) = \mathbf{E}\{z^*(t)z^*(t^{'})\} = 0.$$

Now that it has been established that $R_{zz}(t, t^{'}) = 0$, it is straightforward to compute the second moment and the cross correlation of $x(t)$ and $y(t)$. To accomplish this aim, note by equation (2.32) that

$$\begin{aligned} 0 = R_{zz}(t, t^{'}) &= \mathbf{E}\{z(t)z(t^{'})\} \\ &= \mathbf{E}\{[x(t) + iy(t)][x(t^{'}) + iy(t^{'})]\} \\ &= \mathbf{E}\{x(t)x(t^{'})\} - \mathbf{E}\{y(t)y(t^{'})\} + i[\mathbf{E}\{x(t)y(t^{'})\} + \mathbf{E}\{x(t^{'})y(t)\}]. \end{aligned}$$

This relation is true if and only if

$$\mathbf{E}\{x(t)x(t^{'})\} = \mathbf{E}\{y(t)y(t^{'})\} = R(t - t^{'}) \tag{2.33a}$$

$$\mathbf{E}\{x(t)y(t^{'})\} = -\mathbf{E}\{x(t^{'})y(t)\} = R_{xy}(t, t^{'}). \tag{2.33b}$$

If one lets $t = t^{'}$ in (2.33a), then the average power in each real WSS stochastic is equal and given by

$$R(0) = \mathbf{E}\{x^2(t)\} = \mathbf{E}\{y^2(t)\} = \sigma^2. \tag{2.34}$$

Hence, the total average power in the complex stochastic process $z(t)$ is equal to twice the power of the real process, given by

$$\begin{aligned}\mathbf{E}\{|z(t)|^2\} &= \mathbf{E}\{x^2(t)\} + \mathbf{E}\{y^2(t)\} \\ &= 2\sigma^2. \end{aligned} \tag{2.35}$$

Similarly, if $t = t^{'}$ in (2.33b), then the value of the cross correlation of $x(t)$ and $y(t)$ is given by

$$R_{xy}(t,t) = \mathbf{E}\{x(t)y(t)\} = 0, \tag{2.36}$$

which proves that $x(t)$ and $y(t)$ are uncorrelated for all time t.

2.4 Narrow-Band Stochastic Processes

A narrow-band stochastic process is described in the frequency domain by the property that its modulation bandwidth is smaller than the radio carrier frequency. Suppose the spectral density of $x(t)$ is concentrated in a small neighborhood of frequency f_0. That is, if the noise is narrow-band about the center frequency f_0, then it is natural to express $z(t)$ explicitly in terms of the frequency f_0 as follows:

$$\begin{aligned} z(t) &= [z(t)e^{-i2\pi f_0 t}]e^{i2\pi f_0 t} \\ &= z_0(t)e^{i2\pi f_0 t}, \end{aligned} \tag{2.37}$$

where $z_0(t)$ is called the complex video at baseband (Kelly, Reed, and Root [8] pg. 317). Thus, the complex process of the modulation is given by

$$z_0(t) = z(t)e^{-i2\pi f_0 t}, \tag{2.38}$$

where $z(t)$ is the original complex process on the carrier at frequency f_0. Alternatively, $z_0(t)$ can be written as

$$z_0(t) = r(t)e^{i\phi(t)}, \tag{2.39}$$

where $r(t)$ is the envelope of the base-band modulation and $\phi(t)$ is the phase modulation of $z_0(t)$. The real process of $z(t)$ can be written in terms of the instantaneous envelope $r(t)$ and the instantaneous phase modulation $\phi(t)$ with respect to the carrier signal $\cos(2\pi f_0 t)$ as follows:

$$x(t) = \mathrm{Re}\{z(t)\} = r(t)\cos(2\pi f_0 t + \phi(t)). \tag{2.40}$$

Similarly, the imaginary part of $z(t)$ is given by

$$y(t) = \mathrm{Im}\{z(t)\} = r(t)\sin(2\pi f_0 t + \phi(t)). \tag{2.41}$$

Thus, a substitution of (2.40) and (2.41) into $z(t) = x(t) + iy(t)$ enables one to write the complex WSS stochastic process as

$$z(t) = r(t)[\cos(2\pi f_0 t + \phi(t)) + i\sin(2\pi f_0 t + \phi(t))].$$

In the narrow-band case it can be deduced from the stochastic integral representation of $z_0(t)$, utilizing (2.37) and (2.39), that $r(t)$ and $\phi(t)$ are slowly varying functions of t compared with the carrier signal $\cos(2\pi f_0 t)$ (Kelly, Reed, and Root [8] pg. 317). However, the above representation is exact regardless of the shape of the spectrum. The practical reason for considering $z_0(t)$ is as follows: if $x(t)$ is mixed with the signal $2\cos(2\pi f_0 t)$, then one obtains

$$\begin{aligned} x(t) \cdot 2\cos(2\pi f_0 t) &= r(t)\cos(2\pi f_0 t + \phi(t)) \cdot 2\cos(2\pi f_0 t) \\ &= r(t)\cos(\phi(t)) + r(t)\cos(4\pi f_0 t + \phi(t)). \end{aligned} \tag{2.42}$$

When equation (2.42) is filtered through a low-pass filter it leaves, in the narrow-band case, the real part of the signal modulation, namely,

$$x_0(t) = r(t)\cos(\phi(t)), \tag{2.43}$$

where $x_0(t)$ is called the in-phase component of the modulating signal. Similarly, the imaginary part of $z_0(t)$, or $y_0(t) = \mathrm{Im}\{z_0(t)\}$, can be physically realized by mixing $x(t)$ with the signal $2\sin(2\pi f_0 t)$ to yield

$$\begin{aligned} x(t) \cdot 2\sin(2\pi f_0 t) &= r(t)\cos(2\pi f_0 t + \phi(t)) \cdot 2\sin(2\pi f_0 t) \\ &= -r(t)\sin(\phi(t)) + r(t)\sin(4\pi f_0 t + \phi(t)). \end{aligned} \tag{2.44}$$

When (2.44) is filtered through a low-pass filter it leaves the imaginary part of the signal modulation, namely,

$$y_0(t) = r(t)\sin(\phi(t)), \tag{2.45}$$

where $y_0(t)$ is called the quadrature component of the modulated signal. The base-band process described above can be implemented by the use of a quadrature demodulator circuit as shown in Figure 2.3.

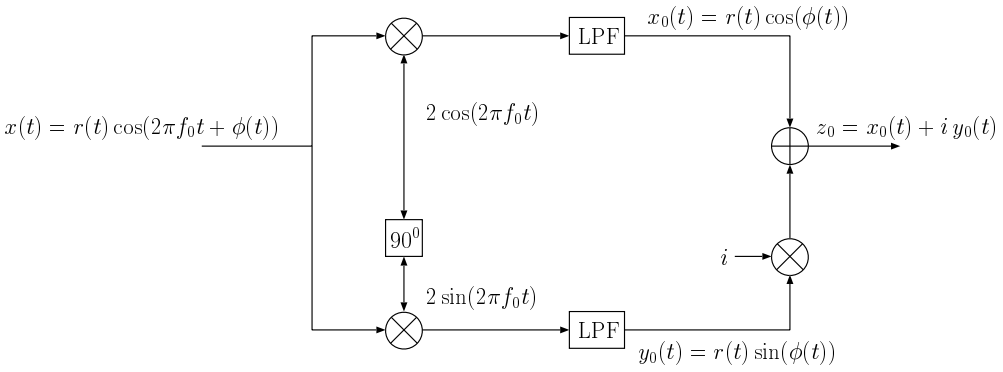

Figure 2.3: Quadrature demodulator

Consider now two new Gaussian baseband processes $u(t)$ and $v(t)$. Further, assume that $u(t)$ and $v(t)$ are uncorrelated processes. Hence, by Gaussianity they are also independent. Next suppose the complex envelope $w(t)$ at baseband is put on a carrier of frequency f_0 by the multiplication of $e^{i2\pi f_0 t}$. Then a multiplication of $w(t)$ with $e^{i2\pi f_0 t}$ produces the Rice representation [9] for a narrow-band Gaussian noise process on a carrier with frequency f_0, given by

$$\begin{aligned} n(t) &= \mathrm{Re}\{w(t)e^{i2\pi f_0 t}\} \\ &= \mathrm{Re}\{[u(t) + iv(t)]e^{i2\pi f_0 t}\} \\ &= u(t)\cos(2\pi f_0 t) - v(t)\sin(2\pi f_0 t), \end{aligned} \tag{2.46}$$

where $w(t) = u(t) + iv(t)$.

2.5 Gaussian Stochastic Processes

Additive noise is characteristic of most physical layer communication systems. Additive noise arises from thermal noise generated by the random motion of electrons in the conductors and resistors comprising the receiver. In a communication system, the thermal noise is generated at or near the first stage of amplification. This is the point in a communication system where the desired signal has the lowest power level, and as a consequence, the thermal noise has the greatest impact on performance. It is a well established fact that thermal noise is characterized accurately as a Gaussian stochastic process (Chapter 6, Davenport and Root [10] and Chapter 11, Papoulis [11]).

A stochastic process is said to be a Gaussian stochastic process if for every set of sampling times t_i, $i = 1, 2, \ldots, n$, the random variables $x(t_i)$ have a joint Gaussian probability density. Let $\underline{x}$ be a Gaussian random vector which is comprised of the Gaussian random variables described above and written as the column vector, $\underline{x} = [x_1, x_2, \ldots, x_n]^T = [x(t_1), x(t_2), \ldots, x(t_n)]^T$, where "T" denotes transpose. The distribution of this Gaussian vector is determined from its mean vector $\underline{m}$ and its correlation matrix R. Since each of

the Gaussian random variables is zero mean, it follows that the mean of $\underline{x}$ is zero and that the correlation matrix is defined by the following outer product

$$R = \mathbf{E}[\underline{x}\underline{x}^T]. \tag{2.47}$$

Hence, the joint probability density function of the components of $\underline{x} = [x(t_1), x(t_2), \ldots, x(t_n)]^T$ is given by

$$\begin{aligned} f(\underline{x}) &= f(x(t_1), x(t_2), \ldots, x(t_n) \\ &= \frac{1}{(2\pi)^{n/2}\sqrt{\det(R)}} \exp\left(\frac{-(\underline{x}-\underline{m})^T R^{-1}(\underline{x}-\underline{m})}{2}\right), \end{aligned} \tag{2.48}$$

provided that the correlation matrix is invertible.

Let $x(t)$ and $y(t)$ be the real and imaginary part of a zero mean complex random variable, $z(t) = x(t) + iy(t)$, such that $x(t)$ and $y(t)$ are jointly Gaussian. One would like to obtain the probability density function of $z(t)$ in order to have a full statistical understanding of the complex random variable. To do this, start by writing $z(t)$ as a random vector and denote it by $\underline{z} = [x, y]^T$, where the index t has been suppressed for notational convenience. Since the mean of $\underline{z}$ is zero, one only needs to calculate the correlation matrix to find the density function of $\underline{z}$. The correlation matrix is given by

$$R = \mathbf{E}[\underline{z}\underline{z}^T] = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}, \tag{2.49}$$

where $\sigma^2 = \mathbf{E}\{x^2(t)\} = \mathbf{E}\{y^2(t)\}$ is the average power in the process, established in (2.34). A substitution of $\underline{m} = 0$ together with the inverse of the correlation matrix R^{-1} into (2.48) yields the probability density function for a complex random vector, given by

$$\begin{aligned} f(z) &= f(x, y) \\ &= \frac{1}{2\pi\sigma^2}\exp\left[-\frac{(x^2 + y^2)}{2\sigma^2}\right] \\ &= \frac{1}{2\pi\sigma^2}\exp\left[-\frac{|z|^2}{2\sigma^2}\right]. \end{aligned} \tag{2.50}$$

Chapter 3

I and Q Analysis of a Costas Phase-Locked Loop

A Costas phase-locked loop (PLL) is a circuit that generates a coherent reference signal which in turn locks onto the carrier phase of a received suppressed-carrier signal. The first purpose of this chapter is to show that under the strict requirement of perfect phase lock, i.e., no phase error, the correlation of the received signal with a coherent reference signal yields the following baseband representation: Signal corrupted with Gaussian noise at the output of the in-phase (I) channel, and Gaussian noise only with no signal at the output of the quadrature (Q) channel. This is the result needed in Chapter 4 to jointly estimate the channel noise power and the magnitudes and signs of the received pulses. These are the unknown parameters needed to estimate the individual bit-error probabilities of a codeword, also derived in Chapter 4.

The second purpose of this chapter is to derive the phase-error variance for the Costas PLL when there is a mismatch between the modulation bandwidth and the RF filter bandwidth in the receiver. While the phase-error variance is already well known for the Costas PLL with matched filters (Holmes [12], Lindsey and Simon [13]), it is not a trivial extension to the case when there is a filter mismatch in the system. The final purpose of this chapter is to analyze the Costas circuit when perfect phase-lock is not achieved. That is, to ascertain the fraction of signal and noise power that leaks from the in-phase channel into the quadrature channel, and the fraction of noise power that leaks from the quadrature channel into the in-phase channel. In this section, it is shown that a relatively large input

signal-to-noise ratio (SNR), or equivalently a small phase-error variance, limits the amount of signal leakage from the in-phase channel to the quadrature channel and visa-versa.

3.1 Analysis of the Costas Phase-Locked Loop

Successful transmission of information through a phase-coherent communication system requires a receiver that can determine the phase and frequency of the received signal with as little error as possible. Quite often the information-bearing signal directly modulates an RF carrier in such a manner that a residual carrier component exists in the overall signal power spectrum. This component can be tracked with a narrow-band phase-locked loop (PLL) and used to provide the desired reference signal. However, the power contained in this residual carrier component does not convey any information other than the phase and frequency of the carrier. Thus, it represents power not available for the transmission of information, and techniques that conserve power are important to an efficient communication system. The greatest power saving is achieved in a suppressed carrier modulation system that has no residual carrier component that a narrow-band PLL can track. In suppressed carrier modulation all of the power is put into the modulation of the data, and as a consequence no energy is wasted on the carrier. To accomplish a carrier lock-on, a suppressed carrier-tracking loop is required to establish a coherent reference carrier. In this study a Costas loop is used to establish a coherent reference signal in order to coherently detect the received signals.

The Costas PLL generates a phantom carrier by a feedback loop that endeavors to lock onto the incoming phase of the received signal. To see how this is accomplished, consider the Costas loop illustrated in Figure 3.1. An estimate of the carrier phase is obtained as follows: First multiply, using two phase detectors, the received input signal $r(t)$ by the output of the voltage-controlled oscillator (VCO) and also by a 90° phase shift of it. Next low-pass filter the results of these two multiplications to remove second and higher order harmonics, and use the product of the two filtered signals to control by feedback the phase and frequency of the VCO output that is part of the loop.

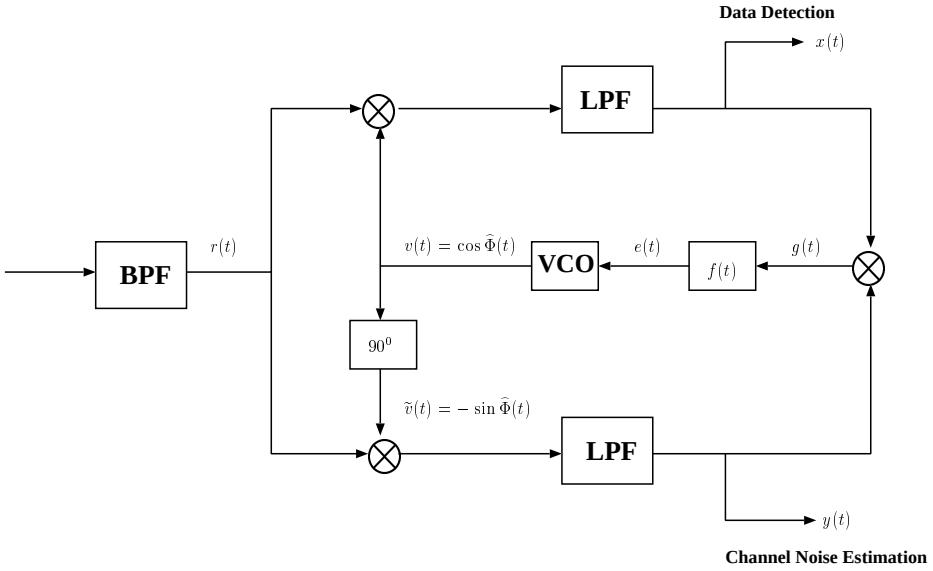

Figure 3.1: Costas phase-locked loop

Binary phase-shift keying (BPSK) is a digital modulation technique that conveys the data by changing, or modulating, the carrier wave using two phases separated by 180^0. Let $s(t)$ BPSK modulated signal of the form

$$s(t) = \sqrt{2}Am(t)\cos\Phi(t), \tag{3.1}$$

where A is the signal amplitude, $m(t) = \pm 1$ is the digital data pulse modulation, and $\Phi(t)$ is the carrier phase. The phase of the signal is given by

$$\Phi(t) = 2\pi f_c t + \theta(t), \tag{3.2}$$

where f_c is the carrier frequency and $\theta(t)$ is the relative phase to the carrier to be estimated. The data sequence $m(t)$ can be modeled by the following WSS process:

$$m(t) = \sum_{n=-\infty}^{\infty} a_n p(t - nT_b), \tag{3.3}$$

where T_b is the pulse width in time, $p(t)$ has a unit amplitude rectangular pulse shape defined by

$$p(t) = \begin{cases} 1, & |t| \leq \frac{T_b}{2} \\ 0, & \text{otherwise,} \end{cases}$$

and $\{a_n\}$ is an independent and identically distributed (i.i.d.) sequence taking on the assumed equiprobable values of ± 1. These rectangular data pulses are assumed to be equal width, non-overlapping, and have no separation between them with a modulation bandwidth, given by

$$B_m = \frac{1}{T_b} \text{ Hz.} \tag{3.4}$$

The front-end of a radio receiver usually includes some provision for preprocessing the received signal. In this study the preprocessing takes the form of a band-pass filter with a finite bandwidth, given by

$$B = \frac{1}{T} \text{ Hz,} \tag{3.5}$$

where T is the RF filter response time. The RF filter bandwidth B is sufficient to pass the modulation $m(t)$ with little or no distortion but not so large as to admit excessive noise through the receiver, that is,

$$B_m < B \quad \text{or} \quad T < T_b. \tag{3.6}$$

It is well known that the additive noise in an RF channel is predominately front-end receiver noise with a flat power spectral density of $N_0/2$ W/Hz, (Watts per Hertz). A multiplication of the power spectral density with the filter bandwidth yields the received noise power, given by

$$P = \frac{N_0}{2} 2B = N_0 B = K T_0 B \text{ Watts,}$$

where K is Boltzmann's constant, T_0 is the absolute temperature in degrees Kelvin, and $N_0 = KT_0$. Mathematically, this receiver noise can be modeled as a narrow-band Gaussian

noise process about the carrier frequency of the observed input data. It is given by the Rice narrow-band representation [9] or by equation (2.46) in Chapter 2 as

$$n(t) = \sqrt{2}n_x(t)\cos\Phi(t) - \sqrt{2}n_y(t)\sin\Phi(t), \tag{3.7}$$

where $n_x(t)$ and $n_y(t)$ are statistically independent, stationary, white Gaussian noise processes. Since the front-end receiver noise is additive, the received signal $r(t)$ can be modeled as follows:

$$\begin{aligned} r(t) &= s(t) + n(t) \\ &= \sqrt{2}[A\,m(t) + n_x(t)]\cos\Phi(t) - \sqrt{2}n_y(t)\sin\Phi(t), \end{aligned} \tag{3.8}$$

where $\Phi(t)$ and $m(t)$ are defined in (3.2) and (3.3), respectively. Both $n_x(t)$ and $n_y(t)$ have identical power spectral density, given by

$$S_{n_x}(f) = S_{n_y}(f) = \begin{cases} N_0, & \frac{-1}{2T} \le f \le \frac{1}{2T} \\ 0, & |f| > \frac{1}{2T} \end{cases} \tag{3.9}$$

with a filter bandwidth B, defined in (3.5), where $B < f_c$ (Haykin [14, pp. 289-290]). The autocorrelation function is the inverse Fourier transform of the power spectral density, given by

$$R_{n_x}(\tau) = R_{n_y}(\tau) = \frac{N_0}{T}\,\mathrm{sinc}\left(\frac{\tau}{T}\right). \tag{3.10}$$

To proceed with the analysis, consider the effects on $r(t)$ in (3.8) as it propagates through the in-phase branch of the Costas loop. In this study the phase detectors are modeled as ideal multipliers that have unit gain with dimensions V^{-1}. The received signal $r(t)$ is first multiplied by a reference signal that is generated by the VCO, given by $v(t) = \sqrt{2}\cos\widehat{\Phi}(t)$. Here $\widehat{\Phi}(t)$ is the estimate of the received phase, given by

$$\widehat{\Phi}(t) = 2\pi f_c t + \widehat{\theta}(t), \tag{3.11}$$

where $\widehat{\theta}(t)$ is the estimated phase, referenced to the carrier. The phase error between $\Phi(t)$ and $\widehat{\Phi}(t)$ is defined by

$$\phi(t) \triangleq \Phi(t) - \widehat{\Phi}(t) = \theta(t) - \widehat{\theta}(t). \tag{3.12}$$

Without loss of generality, it can be assumed that the frequency of the reference signal is equal to the frequency of the received signal because any difference in the instantaneous frequency can be included in the time-varying function $\widehat{\theta}(t)$ (Meyer and Ascheid [15, pg. 21]). Thus, a multiplication of $r(t)$ with $v(t)$ yields the output of the in-phase detector as follows:

$$\begin{aligned}
x(t) &= r(t) \cdot v(t) \\
&= \sqrt{2}\big[A\,m(t)\cos\Phi(t) + n_x(t)\cos\Phi(t) - n_y(t)\sin\Phi(t)\big] \cdot \sqrt{2}\cos\Phi(t) \\
&= 2\big[A\,m(t)\cos\Phi(t)\cos\Phi(t) + n_x(t)\cos\Phi(t)\cos\Phi(t) - n_y(t)\sin\Phi(t)\cos\Phi(t)\big] \\
&= A\,m(t)\cos[\Phi(t) - \widehat{\Phi}(t)] + A\,m(t)\cos[\Phi(t) + \widehat{\Phi}(t)] + n_x(t)\cos[\Phi(t) - \widehat{\Phi}(t)] \\
&+ n_x(t)\cos[\Phi(t) + \widehat{\Phi}(t)] - n_y(t)\sin[\Phi(t) + \widehat{\Phi}(t)] - n_y(t)\sin[\Phi(t) - \widehat{\Phi}(t)].
\end{aligned}$$

Next the output of the in-phase detector is low-pass filtered to remove the second-order harmonic terms. The bandwidth B of the low-pass filters is designed to be the same as the bandwidth of the band-pass filter, defined in (3.5). It is assumed that the system operates at a moderately high input SNR and that the bandwidth of the low-pass filters is wide enough so that the information $m(t)$ is negligibly distorted. The second-order harmonic terms, $\sin[\Phi(t) + \widehat{\Phi}(t)]$ and $\cos[\Phi(t) + \widehat{\Phi}(t)]$, are removed approximately by the low-pass filter to yield the expression for $x(t)$ as

$$\begin{aligned}
x(t) &= A\,m(t)\cos[\theta(t) - \widehat{\theta}(t)] + n_x(t)\cos[\theta(t) - \widehat{\theta}(t)] - n_y(t)\sin[\theta(t) - \widehat{\theta}(t)] \\
&= [A\,m(t) + n_x(t)]\cos\phi(t) - n_y(t)\sin\phi(t),
\end{aligned} \tag{3.13}$$

where $\phi(t)$ is the phase error defined in (3.12). If the loop has perfect phase lock, i.e., $\widehat{\theta}(t) = \theta(t)$, then $\phi(t) = 0$, and the signal in (3.13) reduces to

$$x(t) = A\, m(t) + n_x(t). \tag{3.14}$$

Hence, the output of the in-phase channel in the Costas receiver perfectly reproduces the sum of the information $m(t)$ and the in-phase receiver noise process $n_x(t)$.

In a similar fashion $r(t)$ propagates into the quadrature branch and is multiplied by $\widetilde{v}(t) = -\sqrt{2}\sin\widehat{\Phi}(t)$ which is generated by means of a 90° degree phase shift out of the VCO output. The output $y(t)$ of the quadrature phase detector is given by

$$\begin{aligned}
y(t) &= r(t) \cdot \widetilde{v}(t) \\
&= \sqrt{2}\big[A\, m(t)\cos\Phi(t) + n_x(t)\cos\Phi(t) - n_y(t)\sin\Phi(t)\big] \cdot \big[-\sqrt{2}\sin\Phi(t)\big] \\
&= 2\big[-A\, m(t)\cos\Phi(t)\sin\Phi(t) - n_x(t)\cos\Phi(t)\sin\Phi(t) + n_y(t)\sin\Phi(t)\sin\Phi(t)\big] \\
&= -A\, m(t)\sin[\Phi(t) + \widehat{\Phi}(t)] + A\, m(t)\sin[\Phi(t) - \widehat{\Phi}(t)] - n_x(t)\sin[\Phi(t) + \widehat{\Phi}(t)] \\
&+ n_x(t)\sin[\Phi(t) - \widehat{\Phi}(t)] + n_y(t)\cos[\Phi(t) - \widehat{\Phi}(t)] - n_y(t)\cos[\Phi(t) + \widehat{\Phi}(t)].
\end{aligned}$$

Again the second-order harmonic terms, $\sin[\widehat{\Phi}(t) + \Phi(t)]$ and $\cos[\widehat{\Phi}(t) + \Phi(t)]$, are removed approximately by the low-pass filter to yield the expression for $y(t)$ as

$$\begin{aligned}
y(t) &= A\, m(t)\sin[\theta(t) - \widehat{\theta}(t)] + n_x(t)\sin[\theta(t) - \widehat{\theta}(t)] + n_y(t)\cos[\theta(t) - \widehat{\theta}(t)] \\
&= [A\, m(t) + n_x(t)]\sin\phi(t) + n_y(t)\cos\phi(t).
\end{aligned} \tag{3.15}$$

If the loop has perfect phase lock, i.e., $\theta(t) = \widehat{\theta}(t)$, then $\phi(t) = 0$, and the quadrature signal in (3.15) becomes

$$y(t) = n_y(t). \tag{3.16}$$

Hence, for perfect lock-on, the output of the quadrature channel of the Costas loop reproduces only the quadrature receiver noise process $n_y(t)$ without the information signal $m(t)$ and the in-phase noise process $n_x(t)$.

3.2 Phase Error Stochastic Integro-Differential Equation

In order to generate a signal that contains the carrier harmonic that controls the lock-on of the loop, the signals in the in-phase and quadrature phase channels of the Costas circuit are multiplied together using a phase detector with gain K_m with dimension V^{-1} where V is volts. This signal, denoted by $g(t)$, is called the dynamic-error signal and is a voltage that depends on the differences between the phase and frequencies of the signals $r(t)$ and $v(t)$, and on the additive noise $n(t)$. A multiplication of $x(t)$ and $y(t)$, given in (3.13) and (3.15), respectively, yields

$$
\begin{aligned}
g(t) &= K_m[x(t)\cdot y(t)] \\
&= K_m\Big[[Am(t)+n_x(t)]\cos\phi(t) - n_y(t)\sin\phi(t)\Big]\Big[[A\,m(t)+n_x(t)]\sin\phi(t) + n_y(t)\cos\phi(t)\Big] \\
&= K_m\Big[[Am(t)+n_x(t)]^2\cos\phi(t)\sin\phi(t) + [A\,m(t)+n_x(t)]n_y(t)\cos\phi(t)\cos\phi(t) \\
&\quad - [A\,m(t)+n_x(t)]n_y(t)\sin\phi(t)\sin\phi(t) - n_y^2(t)\sin\phi(t)\cos\phi(t)\Big] \\
&= K_m\Big[A^2m^2(t)\cos\phi(t)\sin\phi(t) + 2Am(t)n_x(t)\cos\phi(t)\sin\phi(t) + x^2(t)\cos\phi(t)\sin\phi(t) \\
&\quad + Am(t)n_y(t)\cos\phi(t)\cos\phi(t) + n_x(t)n_y(t)\cos\phi(t)\cos\phi(t) - Am(t)n_y(t)\sin\phi(t)\sin\phi(t) \\
&\quad - n_x(t)n_y(t)\sin\phi(t)\sin\phi(t) - n_y^2(t)\sin\phi(t)\cos\phi(t)\Big].
\end{aligned}
\tag{3.17}
$$

A use of standard trigonometric identities enables one to simplify the expression by expanding and collecting like terms, to yield

$$g(t) = \frac{K_m}{2}\Bigg[A^2m^2(t)\sin(2\phi(t)) + 2Am(t)n_x(t)\sin(2\phi(t)) \\ + [x^2(t) - n_y^2(t)]\sin(2\phi(t)) + 2[Am(t) + n_x(t)]n_y(t)\cos(2\phi(t))\Bigg]. \tag{3.18}$$

The expression, given in (3.18), can be linearized on the assumption of a sufficiently small phase error, i.e., $|\phi(t)| \leq 30^0 = \pi/6$ (Lindsey [16, pg. 131]). Thus, if the phase-locked loop is capable of reducing the phase error to a small value, then one can use the first term in the Taylor series expansion for $\sin(2\phi(t))$ and $\cos(2\phi(t))$ which are given by

$$\sin(2\phi(t)) \approx 2\phi(t) \tag{3.19a}$$

$$\cos(2\phi(t)) \approx 1. \tag{3.19b}$$

A substitution of (3.19) into (3.18) yields the linear approximation for $g(t)$, given by

$$\begin{aligned} g(t) &= \frac{K_m}{2}\Bigg[2A^2m^2(t)\phi(t) + 4Am(t)n_x(t)\phi(t) + 2[n_x^2(t) - n_y^2(t)]\phi(t) \\ &\quad + 2[Am(t) + n_x(t)]n_y(t)\Bigg] \\ &= K_m\Bigg[A^2m^2(t)\phi(t) + N(t,\phi(t))\Bigg], \end{aligned} \tag{3.20}$$

where $N(t,\phi(t))$ is defined by

$$N(t,\phi(t)) = 2Am(t)n_x(t)\phi(t) + [n_x^2(t) - n_y^2(t)]\phi(t) + [Am(t) + n_x(t)]n_y(t). \tag{3.21}$$

In this study it is assumed that the data pulses are unit amplitude, equal width, and are non-overlapping. Furthermore, the input signal-to-noise ratio is relatively high and the low-pass filters are wide enough so that the data modulation $m(t)$ incurs relatively

little distortion due to filtering. Thus, the square of the data modulation can be computed as follows: For any time t, one has that

$$\begin{aligned} m^2(t) &= \Big(\sum_{n=-\infty}^{\infty} a_n p(t-nT_b)\Big)^2 \\ &= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} a_n a_m p(t-nT_b)p(t-mT_b) \\ &= \sum_{n=-\infty}^{\infty} a_n^2 p^2(t-nT_b) + \sum_{n=-\infty}^{\infty}\sum_{\substack{m=-\infty \\ m\neq n}}^{\infty} a_n a_m p(t-nT_b)p(t-mT_b) \\ &= \sum_{n=-\infty}^{\infty} p^2(t-nT_b) = 1, \end{aligned}$$

where the value of the double sum for $m \neq n$ is zero because for any $t \in \mathbb{R}$, the product of two non-overlapping unit pulse functions is zero. A substitution of $m^2(t) = 1$ into (3.20) enables the dynamic error signal to be written as follows:

$$\begin{aligned} g(t) &= K_m\Big[A^2\phi(t) + N(t,\phi(t))\Big] \\ &= K_m A^2\Big[\phi(t) + \frac{N(t,\phi(t))}{A^2}\Big] \\ &= K_D[\phi(t) + N^{'}(t,\phi(t))], \end{aligned} \tag{3.22}$$

where K_D is referred to as the phase detector gain measured in volts, defined by

$$K_D = K_m A^2, \tag{3.23}$$

and $N^{'}(t,\phi(t))$ is the normalized noise process, given by

$$N^{'}(t,\phi(t)) = \frac{N(t,\phi(t))}{A^2}. \tag{3.24}$$

The $K_D\phi(t)$ term in (3.22) represents a useful control signal which can be considered as the dc-term of the phase detector output assuming a small phase error. The other part,

$N^{'}(t, \phi(t))$, is a zero-mean, rapidly fluctuating disturbance which needs to be filtered out by the PLL. One of the major attractions of a PLL is that it can cope with a significant amount of noise. Note that $N^{'}(t, \phi(t))$ is a dimensionless quantity that can be viewed as an angular phase disturbance which replaces the additive bandpass noise $n(t)$ in the equivalent base-band model (Meyer and Ascheid [15, pp. 106-107]).

Now consider the effects of filtering the signal $g(t)$ with the loop filter response function $f(t)$, shown in Figure 3.1. When $g(t)$ is passed through the time response function of the loop filter $f(t)$, the output is given by the convolution,

$$f(t) * g(t) = \int_0^t f(t-\tau)g(\tau)d\tau,$$

where $f(t) = 0$ on the interval $(-\infty < t < 0)$. Since convolution integrals can be complicated, the effects of filtering $g(t)$ are more easily analyzed in the Laplace-frequency domain. The analysis of a linear system is performed conveniently by the use of Laplace transform techniques because it transforms linear ordinary differential equations into algebraic equations. The unilateral Laplace transform of the loop filter $f(t)$ is the filter-transfer function $F(s)$, defined by

$$F(s) = \int_0^\infty e^{-st} f(t)dt,$$

where $s = \alpha + i\omega$ is the Laplace transform variable. Since the system is linear, the output of the filter is a product in the Laplace domain and one has the following Laplace transform pair:

$$f(t) * g(t) \longleftrightarrow F(s)G(s),$$

where $G(s)$ is the Laplace transform of the signal $g(t)$, provided it exists.

The purpose of the loop filter $F(s)$ is to smooth the dynamic-error signal $g(t)$ and to reduce the amount of noise that enters the VCO [16]. Furthermore, $F(s)$ is a filter element of the closed-loop transfer function $H(s)$ which controls what is called the "loop-noise" bandwidth B_L. Both $H(s)$ and B_L are defined later in Section 3.5 of this Chapter. The most common type of loop filters used in applications are first and second-order filters. A

first-order loop filter corresponds to a transfer function of 1, i.e., $F(s) = 1$ or $f(t) = \delta(t)$, the Dirac delta function in the time domain. A first-order loop filter accurately tracks a constant phase offset with zero steady-state phase error but will not track a frequency offset (Meyer and Ascheid [15, pp. 28-30]). In practical communication systems there exists almost always a difference between the incoming signal frequency and the free-running VCO frequency (corresponding to the zero control voltage of the VCO). To track a phase and frequency offset, an integrator in the loop filter $F(s)$ is needed to compensate for this frequency difference in such a way that no steady-state phase error remains. An integrator in the loop filter $F(s)$ corresponds to a pole at the origin. There are two types of second-order loop filters: A perfect integrator that corresponds to an active filter, and an imperfect integrator that corresponds to a passive filter. The passive filters along with their circuit implementations are discussed in detail in Meyer and Ascheid [15, pp. 35-42]. Note that by the use of higher order loop filters, one can make the loop-noise bandwidth B_L quite narrow-band about the zero frequency. This narrow-band restriction has the property that it reduces the noise that enters into the voltage-controlled oscillator (VCO), thereby enabling the loop to track the phase more accurately. However, the price paid for better tracking performance is an increased time to accurately acquire the signal phase. The process of how to bring the loop into lock from its initial state to the tracking mode is called acquisition. The subject of acquisition is large and is not discussed further in this chapter. The interested reader should consult Meyer and Ascheid [15, Chapters 4, 5, 6].

The stochastic integro-differential equation that governs the behavior of the phase error $\phi(t)$ is derived next. The control voltage $e(t)$ is the result of $g(t)$ being filtered by the loop filter $f(t)$. It is the control voltage that changes the output phase and frequency of the VCO, causing them to coincide with the phase and frequency of the received signal. Thus, the control voltage is given by the convolution $e(t) = f(t) * g(t)$. Clearly from Figure 3.1, the angular frequency of the VCO is directly proportional to the control voltage $e(t)$. When the control voltage is removed, the VCO generates a signal with angular frequency $2\pi f_c$, called the quiescent frequency of the VCO. When the control signal $e(t)$ is applied,

the VCO angular frequency becomes $2\pi f_c + K_0 e(t)$, where K_0 is the VCO gain factor with dimension $s^{-1}V^{-1}$, where V denotes volts and s, seconds. The constant K_0 multiplies $e(t)$ so that the dimension of the phase estimate $\widehat{\Phi}(t)$ is radians. It is a well known fact that frequency is the time derivative of phase. Thus, the control law of the VCO is given by

$$\frac{d\widehat{\Phi}(t)}{dt} = 2\pi f_c + K_0 \int_0^t f(t-\tau)g(\tau)d\tau. \tag{3.25}$$

A differentiation of $\widehat{\Phi}(t)$, given in (3.11) and a comparison with (3.25), shows that the frequency deviation of the VCO from the quiescent frequency is governed by

$$\frac{d\widehat{\theta}(t)}{dt} = K_0 \int_0^t f(t-\tau)g(\tau)d\tau. \tag{3.26}$$

A substitution of $g(t)$, given in (3.22), into (3.26) shows that the frequency deviation of the VCO from the quiescent frequency is given by

$$\frac{d\widehat{\theta}(t)}{dt} = K_0 K_D \left[\int_0^t f(t-\tau)\phi(\tau)d\tau + \int_0^t f(t-\tau)N^{'}(\tau,\phi(\tau))d\tau \right]. \tag{3.27}$$

Equation (3.27) can be rewritten in terms of the phase error, $\phi(t) = \theta(t) - \widehat{\theta}(t)$, as follows:

$$\frac{d}{dt}[\theta(t) - \phi(t)] = K_0 K_D \left[\int_0^t f(t-\tau)\phi(\tau)d\tau + \int_0^t f(t-\tau)N^{'}(\tau,\phi(\tau))d\tau \right].$$

A slight rearrangement of this relation yields the linear first-order stochastic integro-differential equation for the phase error, given by

$$\frac{d\phi(t)}{dt} = \frac{d\theta(t)}{dt} - K_0 K_D \left[\int_0^t f(t-\tau)\phi(\tau)d\tau + \int_0^t f(t-\tau)N^{'}(\tau,\phi(\tau))d\tau \right], \tag{3.28}$$

where $N^{'}(t,\phi(t))$ is the phase-noise process, or disturbance given in (3.24).

3.3 The Dominant Noise Term in $N(t, \phi(t))$

The noise process $N(t, \phi(t))$, given in (3.21), can be written as the sum of three terms as follows:

$$N(t, \phi(t)) = n_1(t) + n_2(t) + n_3(t), \tag{3.29}$$

where

$$n_1(t) = 2Am(t)n_x(t)\phi(t), \tag{3.30a}$$

$$n_2(t) = [n_x^2(t) - n_y^2(t)]\phi(t), \tag{3.30b}$$

$$n_3(t) = [Am(t) + n_x(t)]n_y(t). \tag{3.30c}$$

Clearly the noise terms $n_1(t)$ and $n_2(t)$ are dependent on the phase error $\phi(t)$, whereas $n_3(t)$ is independent of $\phi(t)$. The goal of this section is to prove that $n_3(t)$, is the dominant noise term of the three terms, $n_1(t)$, $n_2(t)$ and $n_3(t)$. In fact, it is proved below that the noise terms $n_1(t)$ and $n_2(t)$ are negligible compared with $n_3(t)$ for $|\phi(t)|$ sufficiently small. This simplification allows one to obtain a closed-form expression for the phase-error variance by means of an equivalent linear control-system circuit. Otherwise, these two noise terms make the analytic solution for phase-error variance quite complex, (Simon and Lindsey [13] and Holmes [12]).

To prove that $n_3(t)$ is the dominate noise term, it is shown next that the variance of $[n_1(t) + n_2(t)]$, conditioned on the phase error $\phi(t)$, is much less that the variance of the noise term $n_3(t)$ for $|\phi(t)|$ sufficiently small, i.e.,

$$\mathbf{Var}\{[n_1(t) + n_2(t)]\big|\phi(t)\} \ll \mathbf{Var}\{n_3(t)\}. \tag{3.31}$$

Conditioning on the phase error enables $\phi(t)$ to be treated as a constant when computing conditional variances. Physically, this is equivalent to the assumption that the correlation times of the processes $n_x(t)$ and $n_y(t)$ are short compared with the correlation time of $\phi(t)$ (Lindsey [16, pg. 131]). Consequently, one can evaluate the variance of the process

$[n_1(t)+n_2(t)]$ by holding $\phi(t)$ fixed and averaging over the more rapidly varying quantities $n_x(t)$ and $n_y(t)$. Equivalently, according to Viterbi [17, pg. 78]), any physical process described by first order differential equation with a white noise driving function will generally be a Markov process. Thus, the phase error $\phi(t)$ described by the first order differential equation, given in (3.28), can be modeled as a Markov process. Recall that a process is said to be Markov if the transition-probability distribution (or density function) is a function only of the present value of the process (or position on the real line) and not at all of its past values [17].

The conditional variance, $\mathbf{Var}\{[n_1(t)+n_2(t)]\big|\phi(t)\}$, appears to be formidable to compute at first. However, if $n_1(t)$ and $n_2(t)$ are uncorrelated processes, when conditioned on $\phi(t)$, then it simplifies to

$$\mathbf{Var}\{[n_1(t)+n_2(t)]\big|\phi(t)\} = \mathbf{Var}\{n_1(t)\big|\phi(t)\} + \mathbf{Var}\{n_2(t)\big|\phi(t)\}. \tag{3.32}$$

To show that $n_1(t)$ and $n_2(t)$ are uncorrelated, first note that the modulation $m(t)$ is a zero mean random process that is independent of the receiver noises $n_x(t)$ and $n_y(t)$ which themselves are uncorrelated processes. Thus, the conditional correlation of $n_1(t)$ and $n_2(t)$ is calculated as follows:

$$\begin{aligned}
\mathbf{E}\{n_1(t)n_2(t)\big|\phi(t)\} &= \mathbf{E}\{[2Am(t)n_x(t)\phi(t)][(n_x^2(t)-n_y^2(t))\phi(t)]\big|\phi(t)\} \\
&= 2\mathbf{E}\{[Am(t)n_x^3(t)\phi^2(t) - Am(t)n_x(t)n_y^2(t)\phi^2(t)]\big|\phi(t)\} \\
&= 2[\mathbf{E}\{Am(t)n_x^3(t)\phi^2(t)\big|\phi(t)\} - \mathbf{E}\{Am(t)n_x(t)n_y^2(t)\phi^2(t)\big|\phi(t)\}] \\
&= 2[A\phi^2(t)\mathbf{E}\{m(t)\}\mathbf{E}\{n_x^3(t)\} - A\phi^2(t)\mathbf{E}\{m(t)\}\mathbf{E}\{n_x(t)\}\mathbf{E}\{n_y^2(t)\}] \\
&= 0.
\end{aligned}$$

Hence, $n_1(t)$ and $n_2(t)$ are uncorrelated processes, holding $\phi(t)$ fixed.

The conditional variances $\mathbf{Var}\{n_1(t)\big|\phi(t)\}$ and $\mathbf{Var}\{n_2(t)\big|\phi(t)\}$ are evaluated next. First note that $\mathbf{E}\{n_1(t)\big|\phi(t)\} = \mathbf{E}\{n_2(t)\big|\phi(t)\} = 0$. With these facts, the conditional variance of $n_1(t)$ is given by

$$\begin{aligned}\mathbf{Var}\{n_1(t)\big|\phi(t)\} &= \mathbf{E}\{n_1^2(t)\big|\phi(t)\} \\ &= 4A^2\mathbf{E}\{m^2(t)n_x^2(t)\phi^2(t)\big|\phi(t)\} \\ &= 4A^2\phi^2(t)\mathbf{E}\{n_x^2(t)\} \\ &= 4A^2\phi^2(t)\sigma^2. \end{aligned} \tag{3.33}$$

Similarly, the conditional variance of $n_2(t)$ is calculated below and given by

$$\begin{aligned}\mathbf{Var}\{n_2(t)\big|\phi(t)\} = \mathbf{E}\{n_2^2(t)\big|\phi(t)\} &= \mathbf{E}\{(n_x^2(t) - n_y^2(t))^2\phi^2(t)\big|\phi(t)\} \\ &= \phi^2(t)\mathbf{E}\{x^4(t) - 2n_x^2(t)n_y^2(t) + y^4(t)\} \\ &= \phi^2(t)\big[\mathbf{E}\{x^4(t)\} - 2\mathbf{E}\{n_x^2(t)n_y^2(t)\} + \mathbf{E}\{y^4(t)\}\big].\end{aligned}$$

Next use the n$^{\text{th}}$ moment theorem for jointly Gaussian processes to evaluate the higher order moments of $n_x(t)$ and $n_y(t)$. Thus, the conditional variance of $n_2(t)$ is given by

$$\begin{aligned}\mathbf{Var}\{n_2(t)\big|\phi(t)\} &= \phi^2(t)[3\sigma^4 - 2\sigma^4 + 3\sigma^4] \\ &= 4\phi^2(t)\sigma^4.\end{aligned} \tag{3.34}$$

Since $n_3(t)$ is a zero mean process, the variance of $n_3(t)$ is calculated as follows:

$$\begin{aligned}\mathbf{Var}\{n_3(t)\} &= \mathbf{E}\{n_3^2(t)\} \\ &= \mathbf{E}\{[Am(t) + n_x(t)]^2 n_y^2(t)\}. \\ &= \mathbf{E}\{[A^2m^2(t) + 2Am(t)n_x(t) + n_x^2(t)]n_y^2(t)\} \\ &= (A^2\mathbf{E}\{n_y^2(t)\} + 2A\mathbf{E}\{m(t)n_x(t)n_y^2(t)\} + \mathbf{E}\{n_x^2(t)n_y^2(t)\}) \\ &= (A^2\sigma^2 + \sigma^4).\end{aligned} \tag{3.35}$$

A substitution of the values for the variances given in (3.33), (3.34), and (3.35) into equation (3.31) shows that for a sufficiently small phase error $\phi(t)$, one has that

$$4(A^2\sigma^2 + \sigma^4)\phi^2(t) << (A^2\sigma^2 + \sigma^4). \tag{3.36}$$

Next cancel the common term $(A^2\sigma^2+\sigma^4)$ so that the inequality in (3.36) and hence (3.31) are valid if the square of the phase error satisfies the inequality,

$$\phi^2(t) << 1/4. \tag{3.37}$$

Equivalently, taking the square root yields the following upper bound on the phase error,

$$|\phi(t)| << 1/2. \tag{3.38}$$

Therefore, it has been established that for the inequality in (3.31) to be satisfied, the phase-error magnitude must be much less than 1/2 radian. Equivalently, one can say that $|\phi(t)| << 1/2$ is a sufficient condition for the inequality in (3.31) to be valid. Thus, for sufficiently small $|\phi(t)|$, the noise terms $n_1(t)$ and $n_2(t)$ in $N(t, \phi(t))$ are negligible and the dominate noise term is $n_3(t)$. Thus, the simplified noise term is given by

$$N(t, \phi(t)) \approx n_3(t) = [Am(t) + n_x(t)]n_y(t), \tag{3.39}$$

for $|\phi(t)| << 1/2$. Equivalently, the simplified angular phase-noise disturbance $N^{'}(t, \phi(t))$, defined in (3.24), is given by

$$N^{'}(t, \phi(t)) \approx n_3^{'}(t) = \frac{1}{A^2}[Am(t) + n_x(t)]n_y(t), \tag{3.40}$$

for $|\phi(t)| << 1/2$.

3.4 The Phase-Error Variance

The goal of this section is to derive the phase-error variance for the Costas loop circuit depicted in Figure 3.1. For simplicity of notation, rename the angular phase disturbance in (3.40), to

$$n^{'}(t) = \frac{1}{A^2}[Am(t) + n_x(t)]n_y(t). \tag{3.41}$$

Here $n^{'}(t)$ is a dimensionless quantity that can be viewed as an angular phase disturbance that replaces the dominant additive bandpass noise $n_3(t)$ in the baseband model (Meyer and Ascheid [15, pg. 107]). Thus, the stochastic integro-differential equation, given in (3.28), can be represented by the equivalent baseband linear control circuit shown in Figure 3.2.

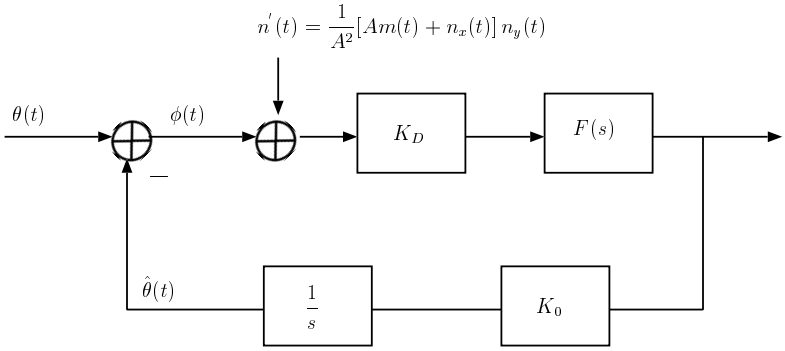

Figure 3.2: Linearized baseband model of the Costas loop

First the closed-loop transfer function $H(s)$ is derived for the baseband model depicted in Figure 3.2. The closed-loop transfer function relative to the input signal $\theta(t)$ is defined as the ratio of the system output $\widehat{\Theta}(s)$ to the system input $\Theta(s)$ (Lindsey and Simon [18, pg. 29]). On the assumption that the Laplace transforms $\widehat{\Theta}(s)$ and $\Theta(s)$ exist, the closed-loop transfer function is given by

$$H(s) \triangleq \frac{\widehat{\Theta}(s)}{\Theta(s)}. \tag{3.42}$$

To facilitate the analysis of $H(s)$, assume that the only input to the control system is $\theta(t)$. Then on the assumption that $n^{'}(t) = 0$, or equivalently $N^{'}(t, \phi(t)) = 0$, the Laplace transform of the stochastic integro-differential equation in (3.28) is given by

$$s\Phi(s) = s\Theta(s) - K_0 K_D F(s)\Phi(s). \tag{3.43}$$

A substitution of $\Phi(s) = \Theta(s) - \widehat{\Theta}(s)$ into (3.43) together with a division by s yields

$$\Theta(s) - \widehat{\Theta}(s) = \Theta(s) - K_0 K_D \frac{F(s)}{s}\left[\Theta(s) - \widehat{\Theta}(s)\right]. \tag{3.44}$$

After simplifying and collecting like terms, equation (3.44) reduces to

$$\widehat{\Theta}(s)\left[1 + K_0 K_D \frac{F(s)}{s}\right] = K_0 K_D \frac{F(s)}{s}\Theta(s).$$

Thus, the closed-loop transfer function is given by

$$H(s) = \frac{K_0 K_D F(s)}{s + K_0 K_D F(s)}, \tag{3.45}$$

where $F(s)$ is the loop filter. Equivalently, the closed-loop transfer function can also be written in the frequency domain by evaluating $H(s)$ on the imaginary axis by setting $\alpha = 0$ to yield

$$H(s)|_{\alpha=0} = H(i2\pi f) \triangleq H_1(f), \tag{3.46}$$

where $s = \alpha + i2\pi f$ is the complex Laplace variable.

Since the system is linear, the principle of superposition holds and the effects of the noise and the useful signal can be determined independently. Thus, one can let $\theta(t) = 0$ and use only $n^{'}(t)$ instead of $\theta(t) + n^{'}(t)$ as the input to the system (Meyer and Ascheid [15, pp. 123-124]). It is well known that when an input process is passed through a linear filter, the output power spectral density is equal to the product of the magnitude squared

of the closed-loop transfer function and the input power spectral density (Haykin [14, pg. 259]). Thus, the power spectral density of the VCO output phase estimate is given by

$$S_{\hat{\theta}}(f) = |H_1(f)|^2 S_{n'}(f),$$

where $H_1(f)$ is the system closed-loop transfer function, and $S_{n'}(f)$ is the power spectral density of $n^{'}(t)$. Since the phase error $\phi(t)$ is equal to $-\widehat{\theta}(t)$ when $\theta(t) = 0$, the phase-error output power spectral density is given by

$$S_{\phi}(f) = |H_1(f)|^2 S_{n'}(f).$$

Hence, the phase-error variance is the integral of the phase-error power spectral density over all possible frequencies, given by

$$\sigma_{\phi}^2 = \int_{-\infty}^{\infty} |H_1(f)|^2 S_{n'}(f) df. \tag{3.47}$$

In order to obtain a closed form expression for the phase-error variance, the value of noise power spectral density $S_{n'}(f)$ must be determined . To facilitate this goal, first calculate the autocorrelation function of $n^{'}(t)$ as follows:

$$\begin{aligned} R_{n'}(\tau) &= \mathbf{E}\{n^{'}(t)n^{'}(t+\tau)\} \\ &= \frac{1}{A^4}\mathbf{E}\{[Am(t)n_y(t) + n_x(t)n_y(t)]\,[Am(t+\tau)n_y(t+\tau) + n_x(t+\tau)n_y(t+\tau)]\} \\ &= \frac{1}{A^4}\Big[A^2\mathbf{E}\{m(t)m(t+\tau)n_y(t)n_y(t+\tau)\} + \mathbf{E}\{Am(t)n_y(t)n_x(t+\tau)n_y(t+\tau)\} \\ &+ \mathbf{E}\{n_x(t)n_y(t)[Am(t+\tau)n_y(t+\tau)]\} + \mathbf{E}\{n_x(t)n_x(t+\tau)n_y(t)n_y(t+\tau)\}\Big]. \end{aligned} \tag{3.48}$$

Since $m(t)$ is a zero mean process independent of $n_x(t)$ and $n_y(t)$, the autocorrelation function simplifies to

$$\begin{aligned} R_{n'}(\tau) &= \frac{1}{A^4}\Big[A^2\mathbf{E}\{m(\tau)m(t+\tau)\}\mathbf{E}\{n_y(\tau)n_y(t+\tau)\} \\ &\quad + \mathbf{E}\{n_x(\tau)n_x(t+\tau)\}\mathbf{E}\{n_y(\tau)n_y(t+\tau)\}\Big] \\ &= \frac{1}{A^4}\Big[A^2R_m(\tau)R_{n_y}(\tau) + R_{n_x}(\tau)R_{n_y}(\tau)\Big], \end{aligned}$$

where $R_m(\tau)$ is the correlation function of the modulation and $R_{n_x}(\tau)$, $R_{n_y}(\tau)$ are the noise correlation functions. By definition, the power spectral density $S_{n'}(f)$ is the Fourier transform of the autocorrelation function $R_{n'}(\tau)$, given by

$$S_{n'}(f) = \frac{1}{A^2}\mathcal{F}\{R_m(\tau)R_{n_y}(\tau)\} + \frac{1}{A^4}\mathcal{F}\{R_{n_x}(\tau)R_{n_y}(\tau)\}, \tag{3.49}$$

where $\mathcal{F}\{\cdot\}$ denotes the Fourier transform. Now recall that the low-pass filters pass the data modulation with very small distortion, that is $B_m < B$. This assumption together with a substitution of $R_{n_y}(\tau)$, given in (3.10), enables the first Fourier transform in (3.49) to be approximated by

$$\mathcal{F}\{R_m(\tau)R_{n_y}(\tau)\} = \mathcal{F}\left\{\frac{N_0}{T}\operatorname{sinc}\left(\frac{\tau}{T}\right)R_m(\tau)\right\} \approx \frac{N_0}{T}\mathcal{F}\{R_m(\tau)\} = \frac{N_0}{T}S_m(f), \tag{3.50}$$

where $S_m(f)$ is the modulation power spectral density. Now it is a well known fact from transform theory that the Fourier transform of a product in the time domain is equal to the convolution of the Fourier transforms in the frequency domain [19]. A use of this fact enables the second Fourier transform in (3.49) to be written as

$$\begin{aligned} \mathcal{F}\{R_{n_y}(\tau)R_{n_y}(\tau)\} &= \mathcal{F}\{R_{n_x}(\tau)\} * \mathcal{F}\{R_{n_y}(\tau)\} \\ &= S_{n_x}(f) * S_{n_y}(f), \end{aligned} \tag{3.51}$$

where $S_{n_x}(f)$ and $S_{n_y}(f)$ are the noise power spectral densities, and $*$ denotes frequency-domain convolution. A substitution of (3.50) and (3.51) into (3.49) yields the power spectral density of $S_{n'}(f)$ as

$$S_{n'}(f) \approx \frac{1}{A^2}\frac{N_0}{T}S_m(f) + \frac{1}{A^4}S_{n_x}(f) * S_{n_y}(f). \tag{3.52}$$

The power spectral density of the modulation $S_m(f)$ is derived in Appendix B following the techniques of Couch [20, pg. 395] and is given by

$$S_m(f) = T_b \operatorname{sinc}^2(T_b f). \tag{3.53}$$

The convolution of the two power spectra, $S_{n_x}(f)$ and $S_{n_y}(f)$, is also computed in Appendix B and given by

$$S_{n_x}(f) * S_{n_y}(f) = \frac{N_0^2}{T}\Big[1 - T|f|\Big]. \tag{3.54}$$

A substitution of (3.53) and (3.54) into (3.52) yields the noise power spectral density, given by

$$S_{n'}(f) \approx \frac{N_0 T_b}{A^2 T}\operatorname{sinc}^2(T_b f) + \frac{1}{A^4}\frac{N_0^2}{T}\Big[1 - T|f|\Big]. \tag{3.55}$$

Define the equivalent loop-noise bandwidth by

$$B_L = \frac{2\int_0^\infty |H_1(f)|^2 df}{|H_1(0)|^2}, \tag{3.56}$$

where B_L is measured in Hertz (Meyer and Ascheid [15, pg. 124]). The equivalent loop-noise bandwidth is the width of a fictitious rectangular spectrum with height $H_1(0)$ such that the power in that rectangular band is equal to the power associated with the actual spectrum over all positive frequencies (Couch [20, pg. 103]). If the loop bandwidth is very narrow-band, that is, $B_L \ll B_m$, then the performance of the loop depends essentially only on the spectrum at the origin (Holmes [12, pg. 143]). A typical bandwidth relationship

is given by $B_m = 10^4 B_L$ (Meyer and Ascheid [15, pp. 124-125]). Thus, the noise power spectral density evaluated at the origin is approximated by

$$S_{n'}(0) \approx \frac{N_0 T_b}{TA^2} + \frac{N_0^2}{TA^4}. \tag{3.57}$$

Finally, a substitution of (3.57) into (3.47) yields the phase-error variance, given by

$$\begin{aligned} \sigma_\phi^2 &= \int_{-\infty}^{\infty} |H_1(f)|^2 S_{n'}(0) df \\ &= \left[\frac{N_0 T_b}{TA^2} + \frac{N_0^2}{TA^4} \right] \int_{-\infty}^{\infty} |H_1(f)|^2 df \\ &= \frac{N_0 B_L}{A^2} \left[\frac{T_b}{T} + \frac{N_0}{TA^2} \right]. \end{aligned} \tag{3.58}$$

Two sanity checks are given next to ensure that the expression for the phase-error variance is correct.

1. If one lets $T_b = T$ in equation (3.58), then the expression for the phase-error variance reduces to the one given in Holmes for the matched filter case (Holmes [12, pg. 143]).

2. Recall that the smallest value for the phase-error variance is for a narrow-band phase-locked loop that tracks only a sine wave, given by

   $$\sigma_\phi^2 = \frac{N_0 B_L}{A^2}.$$

   (Meyer and Ascheid [15, pg. 124]). Since the bit time T_b is greater than the RF filter response time T, the phase-error variance in (3.58) is greater than the phase-error variance for a narrow-band phase-locked loop.

A plot of the phase-error variance versus T_b/T is given in Figure 3.3 for a fixed A^2/N_0 with $B_L = 1$. As the ratio of T_b/T increases, more noise enters the loop resulting in a higher phase-error variance. Recall, that a high data rate system has a smaller bit time T_b compared to a low data rate system and in this study, it is assumed that $T < T_b$. Thus, for a fixed T_b/T, the term N_0/TA^2, given in (3.58), suggests that a high data rate system

yields a higher phase-error variance than a low data rate system. Hence, for a high data rate system, it is crucial to have the data bandwidth matching the signal bandwidth in order to yield the smallest possible phase-error variance.

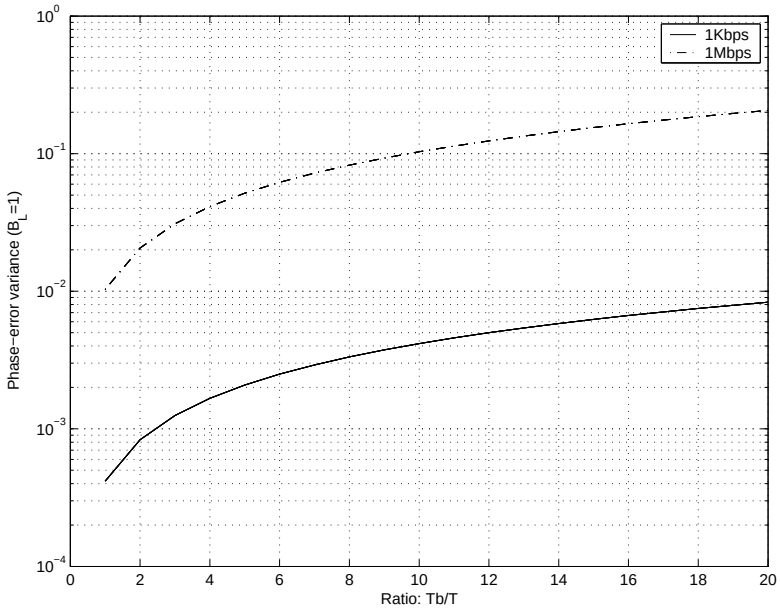

Figure 3.3: Phase-error variance with filter mismatch

It is also instructive to relate the phase-error variance σ_ϕ^2 to the input signal-to-noise ratio (SNR) which is defined by

$$\text{SNR} = \frac{A^2}{N_0 B} = \frac{A^2}{\sigma^2}. \tag{3.59}$$

If the phase-error variance, given in (3.58), is multiplied and divided by the RF filter bandwidth B, then it can be written as

$$\sigma_\phi^2 = \frac{B_L}{B}\frac{N_0 B}{A^2}\left[\frac{T_b}{T} + \frac{N_0}{TA^2}\right] = \frac{N_0 B}{A^2}\left[\frac{B_L}{B_m} + \frac{B_L N_0}{A^2}\right] = \frac{1}{\text{SNR}}\left[\frac{B_L}{B_m} + \frac{1}{\text{SNR}_\text{L}}\right], \tag{3.60}$$

where $\mathrm{SNR_L}$ is the signal-to-noise ratio inside the loop, or equivalently the "loop" SNR, defined by

$$\mathrm{SNR_L} = \frac{A^2}{B_L N_0}. \tag{3.61}$$

Hence, the phase-error variance σ_ϕ^2 is inversely proportional to the product of the input and loop signal-to-noise ratios. This fact agrees with the intuition that reduced amounts of noise produces less phase error. Thus, for a sufficiently high input signal-to-noise ratio, the phase-error variance becomes very small.

3.5 Signal and Noise Leakage Analysis

The realistic situation of imperfect carrier-phase synchronization is analyzed in this section. Assume that there is a relatively large input signal-to-noise ratio, or equivalently a small phase-error. The goal is to quantify the fraction of signal and noise power that leaks into the quadrature channel, and the fraction of noise power that leaks into the in-phase channel. Recently, Dubney and Reed [21], have developed a new method that reduces the decoding time for (23,12,7) Golay code. This method uses the real channel data to estimate the individual bit-error probabilities in a codeword of length n bits. The accuracy of the bit-error probability estimate depends directly on the lock-on of the Costas PLL. Small amounts of leakage imply that the loop has locked onto the phase of the incoming signal which in turn yields higher accuracy of the bit-error probability estimates. A lack of lock-on results in a degradation to the bit-error probability estimate and increased decoding time.

Recall that Section 3.2 established the two following facts about the Costas circuit when perfect phase lock is achieved: (1) The in-phase channel contains the signal information together with the in-phase noise without any quadrature noise. (2) The quadrature channel contains quadrature noise without any signal information. The purpose of this section is to analyze the Costas circuit when perfect phase-lock is not achieved. That is, to ascertain the fraction of signal and noise power leakage from the in-phase channel into the quadrature

channel, and the fraction of noise power leakage from the quadrature channel into the in-phase channel.

Consider the expression for $x(t)$, given in (3.13), which is reproduced here for convenience as

$$x(t) = [A\,m(t) + n_x(t)]\cos\phi(t) - n_y(t)\sin\phi(t).$$

When $\phi(t) \neq 0$, the quadrature noise leakage into the in-phase channel is characterized by

$$w(t) = -n_y(t)\sin\phi(t).$$

Since the system operates at a high SNR, it is reasonable to assume that the phase error $\phi(t)$ is also relatively small. Thus, using the small angle approximation for $\sin\phi(t)$, $w(t)$ reduces to

$$w(t) = -n_y(t)\phi(t).$$

The noise power of $w(t)$ is calculated by conditioning on the phase error $\phi(t)$ as follows:

$$\begin{aligned} P_w = \mathbf{E}\{\mathbf{E}\{w^2(t)\big|\phi(t)\}\} &= \mathbf{E}\{\mathbf{E}\{n_y^2(t)\phi^2(t)\big|\phi(t)\}\} \\ &= \mathbf{E}\{\phi^2(t)\}\mathbf{E}\{n_y^2(t)\} \\ &= \sigma_\phi^2\sigma^2. \end{aligned}$$

Next the total signal power in the in-phase channel is computed. Under the small angle assumption, the expression for $x(t)$ reduces to

$$x(t) = [A\,m(t) + n_x(t)] - n_y(t)\phi(t).$$

With this simplification, the total signal power in the in-phase channel is computed by conditioning on the phase error $\phi(t)$ as follows:

$$\begin{aligned} P_x = \mathbf{E}\{\mathbf{E}\{x^2(t)\big|\phi(t)\}\} &= \mathbf{E}\{\mathbf{E}\{[\big(Am(t)+n_x(t)\big)-n_y(t)\phi(t)]^2\big|\phi(t)\}\} \\ &= \mathbf{E}\{\mathbf{E}\{(Am(t)+n_x(t))^2-2(Am(t)+n_x(t))n_y(t)\phi(t) \\ &+ n_y^2(t)\phi^2(t)\}\big|\phi(t)\} \\ &= A^2+\sigma^2(1+\sigma_\phi^2). \end{aligned}$$

The percentage of noise power leakage L_I into the in-phase channel is defined as the ratio of the quadrature noise power P_w to the total in-phase signal power P_x and is given by

$$L_I \triangleq \frac{P_w}{P_x} = \frac{\sigma_\phi^2\sigma^2}{[A^2+\sigma^2(1+\sigma_\phi^2)]} = \frac{\sigma_\phi^2}{\mathrm{SNR}+1+\sigma_\phi^2}. \tag{3.62}$$

A substitution of the phase-error variance, given in (3.60), into (3.62) enables the leakage L_I to be written in terms of input and loop SNRs as follows:

$$L_I = \frac{(B_L/B_m+1/\mathrm{SNR_L})}{\mathrm{SNR}^2+\mathrm{SNR}+(B_L/B_m+1/\mathrm{SNR_L})}. \tag{3.63}$$

Thus, for sufficiently high input SNR, the phase-error variance becomes negligible and the leakage L_I approaches zero.

In a similar manner the fractional amount of signal and in-phase noise interference that leaks into the quadrature channel can be determined. First consider the expression for $y(t)$, given in (3.15), which is reproduced here for convenience as

$$y(t) = \big[Am(t)+n_x(t)\big]\sin\phi(t) - n_y(t)\cos\phi(t).$$

When $\phi(t) \neq 0$, the signal and in-phase noise leakage into the quadrature channel is given by

$$\tilde{w}(t) = [Am(t)+n_x(t)]\sin\phi(t).$$

The power of $\tilde{w}(t)$, assuming the small angle approximation, is calculated using conditional expectation as follows:

$$\begin{aligned} P_{\tilde{w}} = \mathbf{E}\{\mathbf{E}\{\tilde{w}^2(t)\big|\phi(t)\}\} &= \mathbf{E}\{\mathbf{E}\{[Am(t)+n_x(t)]^2\phi^2(t)\big|\phi(t)\}\} \\ &= \mathbf{E}\{\phi^2(t)\}\mathbf{E}\{A^2m^2(t)+2Am(t)n_x(t)+n_x^2(t)\} \\ &= \sigma_\phi^2(A^2+\sigma^2). \end{aligned}$$

The total signal power in the quadrature channel is calculated in a similar manner to the total signal power in the in-phase channel and is given by

$$\begin{aligned} P_y = \mathbf{E}\{\mathbf{E}\{y^2(t)\}\big|\phi(t)\} &= \mathbf{E}\{\mathbf{E}\{[\big(Am(t)+n_x(t)\big)\phi(t)+n_y(t)]^2\}\big|\phi(t)\} \\ &= [A^2+\sigma^2]\sigma_\phi^2+\sigma^2. \end{aligned}$$

The percentage of signal and in-phase receiver noise leakage into the quadrature channel, denoted by L_Q, is defined as the ratio of the power $P_{\tilde{w}}$ to the total in-phase signal power P_y and is given by

$$L_Q \triangleq \frac{P_{\tilde{w}}}{P_y} = \frac{\sigma_\phi^2(A^2+\sigma^2)}{[\sigma_\phi^2(A^2+\sigma^2)+\sigma^2]} = \frac{\sigma_\phi^2}{\sigma_\phi^2+\frac{1}{\mathrm{SNR}+1}}. \tag{3.64}$$

Finally, a substitution of the phase-error variance, given in (3.60), into (3.64) yields the leakage L_Q as

$$L_Q = \frac{1/\mathrm{SNR}(B_L/B_m+1/\mathrm{SNR_L})}{1/\mathrm{SNR}(B_L/B_m+1/\mathrm{SNR_L})+\frac{1}{\mathrm{SNR}+1}} = \frac{(B_L/B_m+1/\mathrm{SNR_L})}{(B_L/B_m+1/\mathrm{SNR_L})+1} \tag{3.65}$$

for a sufficiently high input SNR.

In order to ascertain the amount of leakage L_Q into the quadrature channel, an example of a second-order PLL with a passive RC filter is taken from Meyer and Ascheid [15, pg. 125]. The second-order loop filter $F(s)$ is given by

$$F(s) = \frac{1+sT_2}{1+sT_1},$$

where $T_1 = (R_1 + R_2)C = 25.7$ ms, $T_2 = R_2C = 1.09$ ms, $K_0 = 66,881\ s^{-1}V^{-1}$ and $K_D = 0.587\ V$. The closed loop transfer function $H(s)$ is calculated by substituting $F(s)$ into (3.45) and is given by

$$H(s) = \frac{1 + (2\zeta - \beta)(s/\omega_n)}{(s/\omega_n)^2 + 2\zeta(s/\omega_n) + 1},$$

where ω_n and ζ are defined in Table 2.4.1 as

$$\omega_n = \left(\frac{K_D K_0}{T_1}\right)^{1/2}, \qquad \zeta = \frac{T_2}{2}\left(\frac{K_D K_0}{T_1}\right)^{1/2}\left(1 + \frac{1}{K_D K_0 T_2}\right)$$

(Meyer and Ascheid [15, pg. 37]), and β is defined as $\beta = 1/(T_1\omega_n)$ (Meyer and Ascheid [15, pg. 125]). A routine calculation shows that $\omega_n = 1,236\ s^{-1}$, $\zeta = 0.69$ and $\beta = 0.031$. The loop bandwidth B_L is calculated by substituting $H(s)$ into (3.56) and integrating to yield

$$B_L = \frac{\omega_n \zeta}{2}\left(1 - \frac{\beta}{\zeta} + \frac{1 + \beta^2}{4\zeta^2}\right). \tag{3.66}$$

A substitution of the values for ω_n, ζ and β given above into (3.66) yields the value of the loop bandwidth as $B_L = 631.4$ Hz. The final expression needed to evaluate the leakage L_Q in (3.65) is $\mathrm{SNR_L}$, the signal-to-noise ratio in the loop. To facilitate this analysis, $\mathrm{SNR_L}$ can be written in terms of the input SNR as follows:

$$\mathrm{SNR_L} = \frac{A^2}{B_L N_0} = \left(\frac{B}{B_L}\right)\frac{A^2}{BN_0} = \left(\frac{B}{B_L}\right)\mathrm{SNR}. \tag{3.67}$$

Following Meyer and Ascheid [15, pp. 125-126], let $B = 15,000$ Hz with an input SNR = 10 dB. Then a simple calculation shows that $\mathrm{SNR_L} = 23.75$ dB which demonstrates the ability of a PLL to cope with a significant amount of noise. Now it is well known that the loop bandwidth is much smaller than the modulation bandwidth i.e., $B_L \ll B_m$ (Holmes [12, pg. 143]). Let the modulation bandwidth have a value of $B_m = 12,000$ Hz. Then a substitution of B_L, B_m, and $\mathrm{SNR_L}$ into (3.65) yields $L_Q = 0.04426$ or equivalently,

$L_Q = -13.5\,\mathrm{dB}$. Therefore, one can conclude that a large $\mathrm{SNR_L}$ together with a large modulation bandwidth B_m drives the leakage in L_Q to be very small.

Chapter 4

Estimation of Individual Bit-Error Probabilities

The main purpose of this chapter is to estimate the individual bit-error probabilities of binary symbols or codewords while they are being received. It turns out that the bit or symbol error probability of a codeword is a function of the received-bit amplitudes A and the channel noise power σ^2, both of which are assumed to be unknown a-priori at the receiver. In this study, coherent detection is implemented with Costas phase-locked loop receiver which facilitates the joint estimation of these two parameters, and as a consequence, the bit-error probabilities. In Chapter 6 it is shown how the individual bit-error probability estimates reduce the decoding complexity of the (23,12,7) Golay code and the (47,24,11) Quadratic Residue code. It is also shown how the bit-error probability estimates facilitate erasure decoding of Reed-Solomon codes over an additive white Gaussian noise (AWGN) channel.

4.1 The Matched Filter

A fundamental problem that arises in a communication system is detecting a pulse transmitted over a channel that is corrupted by additive noise at the front-end of the receiver. Let $h(t)$ be the time impulse response of a linear time-invariant (LTI) filter. The filter input $x(t)$ consists of a stream of signal pulses each of the form $s(t)$ corrupted by additive noise $n(t)$, given by

$$x(t) = s(t) + n(t). \tag{4.1}$$

The signal $s(t)$ represents a binary 0 or 1 bit waveform for a digital communication system. The noise $n(t)$ is a zero-mean white noise process with the two-sided power spectral density $N_0/2$ W/Hz. The function of the receiver is to detect the signal pulses in an optimum manner and decide which bit was transmitted. To satisfy this requirement, one must design a filter that makes the instantaneous signal-to-noise ratio at time $t = t_0$, as large as possible. Consequently, this is equivalent to maximizing the signal-to-noise ratio, defined by

$$\text{SNR}_{\text{out}} = \frac{\text{Signal-waveform power}}{\text{Average noise power}}. \tag{4.2}$$

The problem is to specify the impulse response $h(t)$ of the filter such that the output signal-to-noise ratio, defined in (4.2), is maximized. Dr. Dwight O. North was first to formalize this concept, which he first published in a 1943 classified report at RCA Laboratory at Princeton, New Jersey. Dr. North did not use the name, "matched filter". This term was coined later by David Middleton and J. H. Van Vleck, who independently published the result in 1944 a year after Dr. North in a classified Harvard Radio Research Laboratory report. It was originally called the North filter, and Dr. North's report was later reprinted in the *Proceedings of the IEEE*, in July 1963. The usual derivation of the matched filter is in the frequency domain. However, in this section the matched filter is derived from the time-domain point of view, because the proof demonstrates the importance of sampling at time $t = t_0$ when the maximum output SNR is first attained.

The output process of an LTI filter $h(t)$ with an input signal $x(t)$ is given by the convolution integral

$$y(t) = \int_{-\infty}^{t} h(t-u)x(u)du. \tag{4.3}$$

A substitution of (4.1) into (4.3) yields the mean of the output process, given by

$$\begin{aligned} \mathbf{E}\{y(t)\} &= \int_{-\infty}^{t} h(t-u)\mathbf{E}\{s(u)+n(u)\}du \\ &= \int_{-\infty}^{t} h(t-u)s(u)du, \end{aligned} \tag{4.4}$$

where $\mathbf{E}\{n(u)\} = 0$. The zero-mean output noise process is given by

$$\begin{aligned} w(t) &= y(t) - \mathbf{E}\{y(t)\} \\ &= \int_{-\infty}^{t} h(t-u)(x(u) - s(u))du \\ &= \int_{-\infty}^{t} h(t-u)n(u)du. \end{aligned} \tag{4.5}$$

A substitution of (4.4) and (4.5) into (4.2) enables the signal-to-noise ratio to be calculated as follows:

$$\begin{aligned} \text{SNR}_{\text{out}} = \frac{[\mathbf{E}\{y(t)\}]^2}{\mathbf{E}\{w^2(t)\}} &= \frac{\left(\int_{-\infty}^{t} h(t-u)s(u)du\right)^2}{\mathbf{E}\int_{-\infty}^{t}\int_{-\infty}^{t} h(t-u)h(t-u')n(u)n(u')dudu'} \\ &= \frac{\left(\int_{-\infty}^{t} h(t-u)s(u)du\right)^2}{\int_{-\infty}^{t}\int_{-\infty}^{t} h(t-u)h(t-u')\mathbf{E}\{n(u)n(u')\}dudu'} \\ &= \frac{\left(\int_{-\infty}^{t} h(t-u)s(u)du\right)^2}{\int_{-\infty}^{t}\int_{-\infty}^{t} h(t-u)h(t-u')R_n(u-u')dudu'}, \end{aligned}$$

where $R_n(u-u^{'})$ is the autocorrelation of white noise. It is well known that the autocorrelation function of white noise is the delta-function correlated noise process, given by

$$R_n(u-u^{'}) = \frac{N_0}{2}\delta(u-u^{'}), \tag{4.6}$$

where $N_0/2$ is the two-sided power spectral density in Watts per Hertz (Haykin [14, pg. 420]). A substitution of the autocorrelation function into the output signal-to-noise ratio yields

$$\begin{aligned} \text{SNR}_{\text{out}} &= \frac{\left(\int_{-\infty}^{t} h(t-u)s(u)du\right)^2}{\frac{N_0}{2}\int_{-\infty}^{t}\int_{-\infty}^{t} h(t-u)h(t-u')\delta(u-u')dudu'} \\ &= \frac{\left(\int_{-\infty}^{t} h(t-u)s(u)du\right)^2}{\frac{N_0}{2}\int_{-\infty}^{t} h^2(t-u)du}. \end{aligned} \tag{4.7}$$

The Cauchy-Schwartz inequality [12], states that

$$\left(\int_{-\infty}^{\infty} g_1(t)g_2(t)dt\right)^2 \leq \int_{-\infty}^{\infty} g_1^2(t)dt \int_{-\infty}^{\infty} g_2^2(t)dt$$

with equality if and only if $g_2(t) = cg_1(t)$, with c being any constant. Thus, a use of the Cauchy-Schwartz inequality enables the output signal-to-noise ratio in (4.7) to be upper bounded as follows:

$$\begin{aligned}
\mathrm{SNR}_{\mathrm{out}} &\leq \frac{\int_{-\infty}^{t} h^2(t-u)du \int_{-\infty}^{t} s^2(u)du}{\frac{N_0}{2}\int_{-\infty}^{t} h^2(t-u)du} \\
&= \frac{2}{N_0}\int_{-\infty}^{t} s^2(u)du \\
&\longrightarrow \frac{2E}{N_0},
\end{aligned}$$

as $t \to t_0$, and where the total signal energy is given by

$$E = \int_{-\infty}^{t_0} s^2(u)du,$$

with t_0 being the extent of $s(t)$ in time, or the time at which the signal energy is maximized.

To obtain equality for the output SNR, it is clear from equation (4.7) that at time $t = t_0$ one should choose the impulse response filter to be

$$h(t_0 - u) = s(u), \tag{4.8}$$

where c is chosen to be 1. If one makes the change of variables $v = t_0 - u$, then the impulse response filter is given by

$$h(v) = s(t_0 - v).$$

The upper bound on the output SNR is attained at time $t = t_0$ by a substitution of the impulse response $h(v) = s(t_0 - v)$ into the output signal-to-noise ratio, given in (4.7), as follows:

$$\begin{aligned}
\mathrm{SNR}_{\mathrm{out}}\big|_{t=t_0} &= \frac{\left(\int_{-\infty}^{t_0} s(t_0 - t_0 + u)s(u)du\right)^2}{\frac{N_0}{2}\int_{-\infty}^{t_0} s^2(t_0 - t_0 + u)du} \\
&= \frac{\left(\int_{-\infty}^{t_0} s^2(u)du\right)^2}{\frac{N_0}{2}\int_{-\infty}^{t_0} s^2(u)du} \\
&= \frac{2}{N_0}\int_{-\infty}^{t_0} s^2(u)du \\
&= \frac{2E}{N_0}.
\end{aligned}$$

Hence, the optimum impulse response filter is a time-reversed and delayed version of the input signal that attains its maximum for the first time at $t = t_0$. Equivalently, one can say that the optimum filter that maximizes the output signal-to-noise ratio at time $t = t_0$ is the signal waveform run backwards in time from $t = t_0$.

4.2 Periodic Signed-Pulse Matched Filter Detection

The output of a matched filter for the k-th pulse, sampled at time $t_0 = kT_b$ is found next. Here kT_b is the trailing edge of the k-th pulse, where the pulse is sampled. Let $x(t)$ be the k-th pulse signal embedded in receiver noise. The output of a matched filter is given by the convolution integral in equation (4.3) and reproduced here as

$$y(t) = \int_{-\infty}^{t} h(t - u)x(u)du. \tag{4.9}$$

In this study the information signals are represented by a sequence of rectangular pulses, each of width T_b. Let $p(u)$ be the k-th rectangular unit pulse function, given by

$$p(u) = \begin{cases} 1, \ (k-1)T_b \leq u \leq kT_b \\ 0, \text{ otherwise.} \end{cases}$$

In order to attain the upper bound of the output signal-to-noise ratio, it is clear from equation (4.8) that one should choose the impulse response filter to be

$$h(t_0 - u) = p(kT_b - u) = \begin{cases} 1, \ (k-1)T_b \leq u \leq kT_b \\ 0, \text{ otherwise.} \end{cases} \tag{4.10}$$

A substitution of the time reversed and shifted pulse function $p(kT_b - u)$ in (4.10) into equation (4.9) yields

$$y_k = y(kT_b) = \int_{(k-1)T_b}^{kT_b} x(u)du,$$

for all integers k.

4.3 The Costas PLL Receiver with Matched Filters.

Consider the Costas PLL receiver illustrated in Figure 4.1. Recall that Section 3.2 of Chapter 3 established the following fact under the requirement of perfect phase synchronization: the output of the in-phase and quadrature phase channels are Gaussian random processes, given respectively, by

$$x(t) = A\, m(t) + n_x(t)$$
$$y(t) = n_y(t),$$

where $n_x(t)$ and $n_y(t)$ represent, respectively, the in-phase and quadrature phase noise processes of the receiver input.

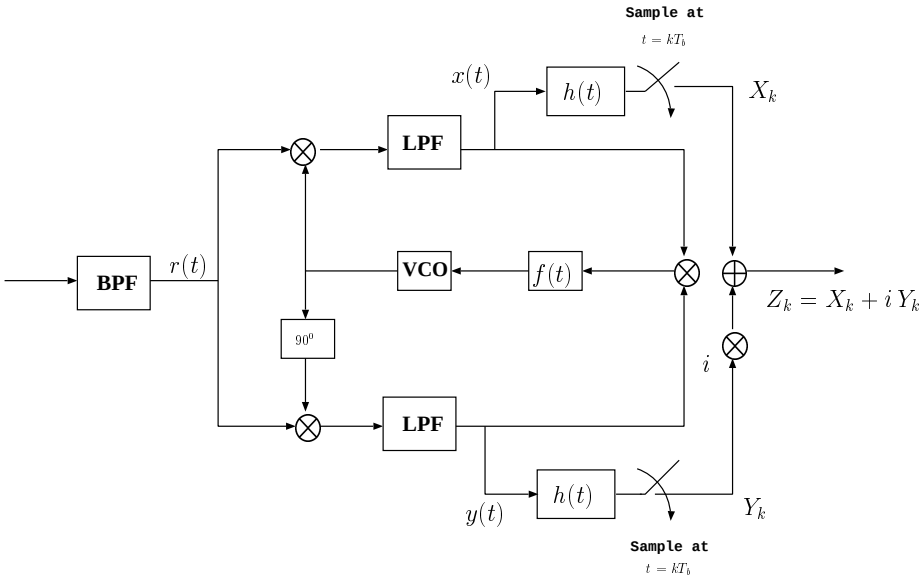

Figure 4.1: Costas phase-locked loop receiver functional diagram

The data sequence $m(t)$ is modeled by the following WSS process:

$$m(t) = \sum_{n=-\infty}^{\infty} a_n p(t - nT_b),$$

where T_b is the pulse width in time, $p(t)$ has a unit amplitude rectangular pulse shape defined by

$$p(t) = \begin{cases} 1, & |t| \leq \frac{T_b}{2} \\ 0, & \text{otherwise,} \end{cases}$$

and $\{a_n\}$ is an independent and identically distributed (i.i.d.) sequence taking on the assumed equiprobable values of ± 1. These rectangular data pulses are assumed to be equal width, non-overlapping, and have zero separation between them with a modulation bandwidth $B_m = 1/T_b$ Hz.

As discussed in the previous section, in order to detect the k-th received bit, $x(t)$ is sent through an optimum matched filter for a pulse, where the integration time is performed over the duration of the k-th pulse or bit interval, $((k-1)T_b, kT_b)$. The pulse is then sampled optimally and periodically at the Nyquist rate at every $t = kT_b$ seconds at the end of the k-th bit interval. Since integration is a linear operation, it is well known that the integral of a Gaussian process is a Gaussian random variable (Doob [5, pg. 426]). Thus, sampling the output at times $t = kT_b$ yields a Gaussian random sequence, given by

$$\begin{aligned} X_k = \frac{1}{T_b} \int_{(k-1)T_b}^{kT_b} x(t)dt &= \frac{1}{T_b} \int_{(k-1)T_b}^{kT_b} [A\, m(t) + n_x(t)]dt \\ &= \frac{1}{T_b} \int_{(k-1)T_b}^{kT_b} A\, m(t)dt + \frac{1}{T_b} \int_{(k-1)T_b}^{kT_b} n_x(t)dt. \end{aligned} \tag{4.12}$$

Define the amplitude A_k by:

$$A_k = \frac{1}{T_b} \int_{(k-1)T_b}^{kT_b} Am(t)dt = \begin{cases} +A \text{ with } \Pr = 1/2, \\ -A \text{ with } \Pr = 1/2. \end{cases} \tag{4.13}$$

Clearly, the amplitudes A_k, $\{k = 1, 2, \cdots\}$ constitute a sequence of independent Bernoulli random variables that take on the values $a_k = \pm A$ with equal probability. Now define an increment of the integrated noise process by

$$N_{x,k} = \frac{1}{T_b} \int_{(k-1)T_b}^{kT_b} n_x(t)dt. \tag{4.14}$$

It is well known that $N_{x,k}$ is a Gaussian sequence or more classically a one-dimensional increment of Brownian Motion, (Doob [5, pp. 97-98]). Hence, for each integer k, X_k is a sum of Bernoulli and Gaussian random variables. Thus, X_k is called a discrete-time conditional Gaussian random sequence, given by

$$X_k = A_k + N_{x,k}. \tag{4.15}$$

Since X_k is conditional Gaussian, only the conditional mean and variance are needed to specify the probability density of X_k. The conditional mean is given by

$$\mathbf{E}\{X_k|A_k\} = \mathbf{E}\{A_k + N_{x,k}|A_k\} = A_k = \pm A. \tag{4.16}$$

The conditional variance of X_k is calculated, using the integrated process along with the established fact that white noise is a delta-function correlated process as defined in (4.6), as follows:

$$\begin{aligned}
\sigma^2 = \mathbf{Var}\{X_k|A_k\} &= \mathbf{E}\big\{[X_k - \mathbf{E}\{X_k|A_k\}]^2\big|A_k\big\} \\
&= \mathbf{E}\left\{\left[\frac{1}{T_b}\int_{(k-1)T_b}^{kT_b} n_x(t)dt\right]^2\right\} \\
&= \frac{1}{T_b^2}\int_{(k-1)T_b}^{kT_b}\int_{(k-1)T_b}^{kT_b} \mathbf{E}\{n_x(t)n_x(t')\}dtdt' \\
&= \frac{1}{T_b^2}\frac{N_0}{2}\int_{(k-1)T_b}^{kT_b}\int_{(k-1)T_b}^{kT_b} \delta(t-t')\}dtdt' \\
&= \frac{N_0}{2T_b}.
\end{aligned} \tag{4.17}$$

Define $\sigma^2 \triangleq N_0/2T_b$ to be the noise power at the input to the receiver. Thus, the probability density of X_k is calculated by taking the expectation of the conditional Gaussian probability density as follows:

$$\begin{aligned} f(x_k) &= \mathbf{E}\{f(x_k|a_k)\} \\ &= \frac{1}{2}f(x_k - a_k) + \frac{1}{2}f(x_k + a_k) \\ &= \frac{1}{2}\frac{1}{\sqrt{2\pi\sigma^2}}\left[\exp\left(\frac{-(x_k - a_k)^2}{2\sigma^2}\right) + \exp\left(\frac{-(x_k + a_k)^2}{2\sigma^2}\right)\right], \end{aligned} \tag{4.18}$$

where x_k and a_k represent particular realizations of the random variables X_k and A_k, respectively. Note that it is customary to represent random variables with capital letters, and to write probability density functions with the corresponding lower case letters.

Similarly, in the quadrature channel, $y(t)$ is also sent through the same pulse matched filter, where the integration time is performed over the duration of the k-th pulse or bit, i.e., the time T_b. It is assumed that the two matched filters of the in-phase and quadrature phase channels of the PLL have the same bit synchronization. The pulse is then sampled at the Nyquist rate at every $t = kT_b$ seconds at the end of the k-th bit interval, to yield the Gaussian sequence

$$Y_k = \int_{(k-1)T_b}^{kT_b} y(t)dt = \int_{(k-1)T_b}^{kT_b} n_y(t)dt. \tag{4.19}$$

Thus, for each integer k, the output is a Gaussian random noise sequence or a one-dimensional increment of Brownian motion, defined by

$$N_{y,k} = \int_{(k-1)T_b}^{kT_b} n_y(t)dt. \tag{4.20}$$

Since Y_k is Gaussian, the probability density function is given by

$$f(y_k) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{-y_k^2}{2\sigma^2}\right), \tag{4.21}$$

where the variance σ^2 is given in equation (4.17).

As illustrated in Figure 4.1, the samples X_k and Y_k are combined together to form a two dimensional, complex Gaussian random sequence,

$$\begin{aligned} Z_k &= X_k + iY_k \\ &= (A_k + N_{x,k}) + iN_{y,k}, \end{aligned} \tag{4.22}$$

which represents the observation of the k-th received bit in the complex z-plane. Since X_k is a conditional Gaussian distribution, and Y_k is Gaussian, both of which are independent identically distributed random sequences, the joint density function $f(x_k, y_k)$ is obtained by multiplying the probability densities of X_k and Y_k as follows:

$$\begin{aligned} f(x_k, y_k) &= f(x_k)f(y_k) \\ &= \frac{1}{4\pi\sigma^2}\left[\exp\left(\frac{-(x_k - a_k)^2}{2\sigma^2}\right) + \exp\left(\frac{-(x_k + a_k)^2}{2\sigma^2}\right)\right]\exp\left(\frac{-y_k^2}{2\sigma^2}\right) \\ &= \frac{1}{4\pi\sigma^2}\left[\exp\left(\frac{-[(x_k - a_k)^2 - y_k^2]}{2\sigma^2}\right) + \exp\left(\frac{-[(x_k + a_k)^2 - y_k^2]}{2\sigma^2}\right)\right]. \end{aligned} \tag{4.23}$$

Alternatively, since a_k is real, the joint probability density in (4.23) can be written more compactly in terms of the density function of Z_k as follows:

$$f(z_k) = \frac{1}{4\pi\sigma^2}\left[\exp\left(\frac{-|z_k - a_k|^2}{2\sigma^2}\right) + \exp\left(\frac{-|z_k + a_k|^2}{2\sigma^2}\right)\right], \tag{4.24}$$

where $z_k = x_k + iy_k$.

4.4 Estimation of Bit-Error Probability

The likelihood ratio test (LRT) is a criterion that is used to decide what was the most likely transmitted signal. This test compares the ratio of the joint conditional probability

densities, given either pulse signal hypothesis, to the ratio of the a-priori probabilities of the transmitted signals. The likelihood ratio test is given by,

$$L(z_k) = \frac{f(z_k|A_k = +A)}{f(z_k|A_k = -A)} \underset{-A}{\overset{+A}{\gtrless}} \frac{\Pr(A_k = -A)}{\Pr(A_k = +A)} = \frac{1/2}{1/2} = 1, \tag{4.25}$$

where the quantity on the right-hand-side of (4.25) is the threshold of the test, (Van Trees [22, pp. 23-27]). It is reasonable in a communication system to assume that the two transmitted signals are equally likely, so that the threshold equals 1. Hence, one can interpret the likelihood ratio test as follows: The receiver decides that $+A$ was transmitted if $L(z_k) > 1$, or decides $-A$ was transmitted if $L(z_k) < 1$. By fixing $A_k = a_k$, Z_k is a two-dimensional complex random vector with jointly Gaussian random variable components $N_{x,k}$ and $N_{y,k}$. The conditional probability density function $f(z_k|A_k = a_k)$ can be determined from equation (2.50) in Chapter 2 and is given by

$$f(z_k|A_k = a_k) = \frac{1}{2\pi\sigma^2}\exp\left[\frac{-|z_k - a_k|^2}{2\sigma^2}\right]. \tag{4.26}$$

A substitution of (4.26) into (4.25) together with canceling common terms yields

$$L(z_k) = \frac{\exp[-|z_k - A|^2/2\sigma^2]}{\exp[-|z_k + A|^2/2\sigma^2]} = \exp\left[-\frac{1}{2\sigma^2}\left(|z_k - A|^2 - |z_k + A|^2\right)\right] \underset{-A}{\overset{+A}{\gtrless}} 1. \tag{4.27}$$

A substitution of $z_k = x_k + iy_k$ into (4.27) and a simplification yields the likelihood-ratio test only as a function of x_k, given by

$$L(x_k) = \exp\left[\frac{2Ax_k}{\sigma^2}\right] \underset{-A}{\overset{+A}{\gtrless}} 1. \tag{4.28}$$

Since the natural logarithm is a monotonically increasing function, and both sides of (4.28) are positive, an equivalent test is given by

$$L^{'}(x_k) = \ln L(x_k) = \frac{2Ax_k}{\sigma^2} \underset{-A}{\overset{+A}{\gtrless}} 0,$$

known as the log-likelihood ratio test (LLRT). Equivalently, the LLRT reduces to

$$\operatorname{sgn}(x_k) \underset{-A}{\overset{+A}{\gtrless}} 0.$$

Thus, if $\operatorname{sgn}(x_k)$ is positive, then the receiver decides the transmitted pulse was $+A$, or if $\operatorname{sgn}(x_k)$ is negative, then it decides the transmitted pulse was $-A$.

The bit-error probability of a particular bit in a codeword or symbol was first developed by Reed in [23] at MIT Lincoln Laboratory in 1959. This bit-error probability is shown to be a function of only the noise power and the received bit-amplitude sampled data in the channel. However, in an actual communications system both the channel noise power σ^2 and the received bit amplitudes $\pm A$ are not known a-priori at the front-end of the receiver. In the next section it is shown how a Costas-loop receiver enables one to jointly estimate these two parameters and as a consequence the bit-error probability. First, however, the bit-error probability is derived in a manner similar to that in [23].

Suppose on reception of a binary symbol or codeword that the receiver records the x_k and y_k data of the k-th bit. The conditional probability of either signal hypothesis, $a_k = +A$ or $a_k = -A$, given the observation $z_k = (x_k, y_k)$, is now calculated. For this purpose note the identity

$$f(A_k = a_k, z_k) = \Pr(A_k = a_k) f(z_k | A_k = a_k) = f(z_k) \Pr(A_k = a_k | z_k), \tag{4.29}$$

where the joint density is

$$f(z_k) = \sum_{a_k} f(A_k = a_k, z_k) = f(A_k = +A, z_k) + f(A_k = -A, z_k).$$

From (4.29), the conditional probability of either hypothesis $a_k = +A$ or $a_k = -A$, given the point observation $z_k = (x_k, y_k)$, is given by

$$\Pr(A_k = a_k | z_k) = \frac{\Pr(A_k = a_k) f(z_k | A_k = a_k)}{f(z_k)}. \tag{4.30}$$

A substitution of (4.26) and (4.24) into (4.30) yields

$$\Pr(A_k = a_k|z_k) = \frac{\frac{1}{2}[\frac{1}{2\pi\sigma^2}\exp(-|z_k - a_k|^2/2\sigma^2)]}{\frac{1}{4\pi\sigma^2}[\exp(-|z_k - a_k|^2/2\sigma^2) + \exp(-|z_k + a_k|^2/2\sigma^2)]}. \tag{4.31}$$

A simple division by the numerator enables (4.31) to be re-expressed as

$$\Pr(A_k = a_k|z_k) = \frac{1}{1 + \exp(-|z_k + a_k|^2/2\sigma^2)\exp(|z_k - a_k|^2/2\sigma^2)}. \tag{4.32}$$

A substitution of $z_k = x_k + iy_k$ into (4.32) and a simplification yields the conditional probability of either hypothesis, given the point observation x_k, as

$$\Pr(A_k = a_k|x_k) = \frac{1}{1 + \exp(-2a_k x_k/\sigma^2)}.$$

Evidently, the probability of making the correct decision is given by

$$\mathrm{P_C} = \Pr(A_k = \mathrm{sgn}(x_k)\, A|x_k),$$

and the probability of a wrong decision is given by

$$\mathrm{P_E} = \Pr(A_k = -\mathrm{sgn}(x_k)\, A|x_k).$$

Hence, the probability that the receiver makes an error is given by the formula

$$\mathrm{P_E} = \Pr(A_k = -\mathrm{sgn}(x_k)\, A|x_k) = \frac{1}{1 + \exp(2A|x_k|/\sigma^2)}. \tag{4.33}$$

Note that the bit-error probability, given in (4.33), is a function only of the channel noise power σ^2, the received bit amplitude A, and the absolute value of the observation x_k. Since both the channel noise power and the received-bit amplitude are not known a-priori in the front-end of the receiver, a joint estimation of these two parameters is derived in the next section.

4.5 Joint Estimation of Channel Noise Power and Received Bit Amplitude

The goal of this section is to derive a joint estimate of the channel-noise power and the received bit amplitude. The estimate of the channel noise power is derived first and used in the subsequent derivation of the received bit amplitude. Recall that the quadrature channel matched filter output of the Costas PLL receiver is a zero-mean discrete-time Gaussian noise sequence, denoted by Y_k. It is well known in the statistical literature that the sample variance is an unbiased maximum-likelihood estimator (Casella and Berger [24, pg. 221]). Thus, the estimate of the noise power in the quadrature channel is given by

$$\sigma_{n_y}^2 \approx S_{n_y} = \frac{1}{N-1}\sum_{k=1}^{N} Y_k^2,$$

where Y_k is defined in (4.19) and N is the number of received-bit samples. The received bit amplitude can be estimated from equation (4.15) by calculating the second moment of X_k as follows:

$$\begin{aligned}\mathbf{E}\{X_k^2\} &= \mathbf{E}\{(A_k + N_{x,k})^2\} \\ &= \mathbf{E}\{A_k^2\} + 2\mathbf{E}\{A_k\}\mathbf{E}\{N_{x,k}\} + \mathbf{E}\{N_{x,k}^2\} \\ &= A^2 + \sigma_{n_x}^2.\end{aligned}$$

Solving for A yields

$$A = \pm\sqrt{\mathbf{E}\{X_k^2\} - \sigma_{n_x}^2}. \tag{4.34}$$

In order to obtain numerical estimates of the received bit amplitude, expressions for $\mathbf{E}\{X_k^2\}$ and $\sigma_{n_x}^2$ are found next. Recall that equation (2.34) of Chapter 2 proved that the noise power in the in-phase channel is identical to the noise power in the quadrature phase channel, i.e., $\sigma_{n_x}^2 = \sigma_{n_y}^2$. For notational convenience, denote this identical noise power by

σ^2. Therefore, it is reasonable to estimate the noise power in the in-phase channel with the sample variance estimator S_{n_y} of the quadrature channel, that is,

$$\sigma^2 \approx S_{n_y} = \frac{1}{N-1}\sum_{k=1}^{N} Y_k^2. \tag{4.35}$$

The Strong Law of Large Numbers enables one to estimate $\mathbf{E}\{X_k^2\}$ as follows: Let $Z_k = X_k^2$. Clearly, $Z_1, Z_2, \cdots, Z_N$ constitute an i.i.d. random sequence with mean $\mu_z = \mathbf{E}\{X_k^2\}$ and finite variance $\sigma_z^2 = \mathbf{E}\{X_k^4\} - \mathbf{E}^2\{X_k^2\} < \infty$. The mean of Z_k is given by

$$\bar{Z}_N = \frac{1}{N}\sum_{k=1}^{N} Z_k = \frac{1}{N}\sum_{k=1}^{N} X_k^2.$$

Then, for every $\epsilon > 0$,

$$\Pr(\lim_{N\to\infty} |\bar{Z}_N - \mu_z| < \epsilon) = 1;$$

that is, $\bar{Z}_N$ converges almost surely (a.s.) to μ_z, i.e.,

$$\bar{Z}_N \xrightarrow{a.s.} \mu_z.$$

A substitution of $\bar{Z}_N = \frac{1}{N}\sum_{k=1}^{N} X_k^2$ and $\mu_z = \mathbf{E}\{X_k^2\}$ yields the desired estimate, namely,

$$\frac{1}{N}\sum_{k=1}^{N} X_k^2 \xrightarrow{a.s.} \mathbf{E}\{X_k^2\}.$$

Thus, by the Strong Law of Large Numbers, the second moment of X_k is approximated by the average of the received squared values, for N sufficiently large,

$$\mathbf{E}\{X_k^2\} \approx \frac{1}{N}\sum_{k=1}^{N} X_k^2. \tag{4.36}$$

Thus, a substitution of (4.36) and (4.35) into (4.34) yields an estimate of the received signal amplitude, given by

$$\begin{aligned}\widehat{A} &= \pm\sqrt{\frac{1}{N}\sum_{k=1}^{N} X_k^2 - \frac{1}{N-1}\sum_{k=1}^{N} Y_k^2} \\ &\approx \pm\sqrt{\frac{1}{N}\sum_{k=1}^{N}(X_k^2 - Y_k^2)},\end{aligned} \tag{4.37}$$

for N sufficiently large.

4.6 Simulation Results

A computer simulation was performed using MATLAB to verify the analytical expressions obtained for the joint estimation of the received bit amplitude and the front-end receiver noise power. The signal amplitude A was fixed to be 1, and the SNR values were varied from 0 to 15 dB. For each SNR value, 200 bit samples were collected and used to estimate the signal amplitude and the noise power. The results are summarized below in Table 4.1.

Table 4.1: Estimation of received bit amplitude and noise power

SNR (dB)	A = 1	Estimated A	Error	Actual σ^2	Estimated S^2	Error
0	1.000	0.9920	0.0080	1.0000	0.9577	0.0423
3	1.000	0.9927	0.0073	0.5012	0.4947	0.0065
6	1.000	0.9943	0.0057	0.2512	0.2636	0.0124
9	1.000	0.9991	0.0009	0.1259	0.1255	0.0004
12	1.000	1.0005	0.0005	0.0631	0.0629	0.0002
15	1.000	1.0000	0.0000	0.0316	0.0325	0.0009

In Figure 4.2 the estimate of A versus the number of bit samples is shown. The simulation was performed for 200 bit samples at a signal-to-noise ratio of 6 dB. It can be inferred from the graph, that it requires approximately 160 bit samples for the estimate of A to converge to the actual value of $A = 1$.

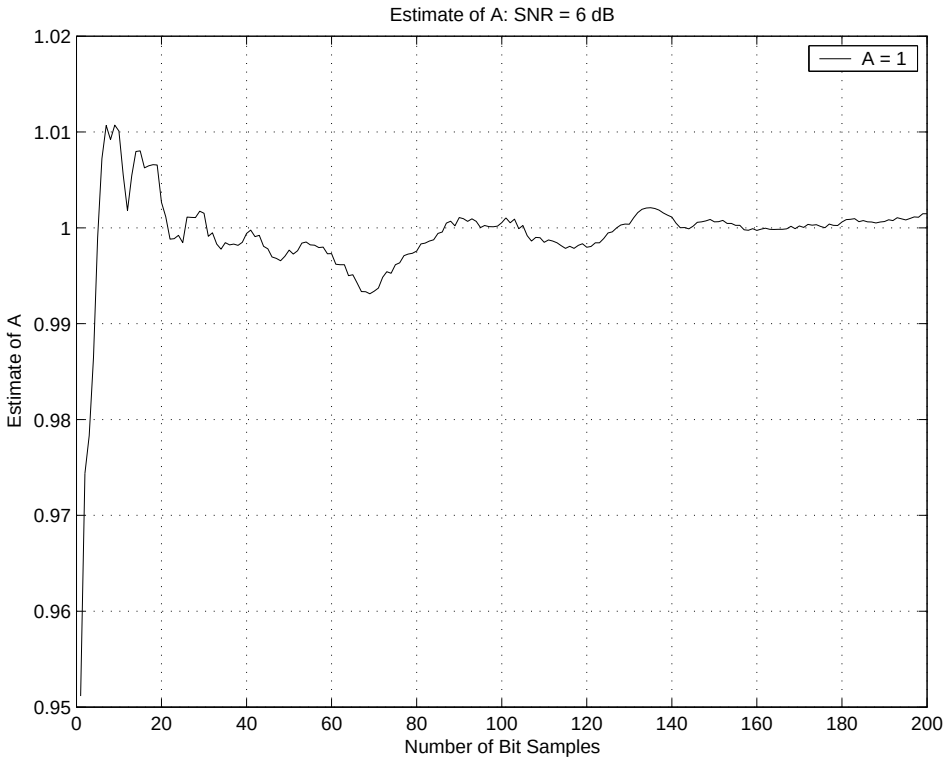

Figure 4.2: Estimated value of the received amplitude A

4.7 Accuracy of Bit-Error Probability Estimates

In this section the accuracy of the bit-error probability estimate is analyzed. First note that the bit-error probability estimate, given in equation (4.33), a function of the received information which is discrete-time conditional Gaussian random sequence. Hence, this bit-error probability estimate is actually a random variable. Denote this random variable by

$$p_k = \frac{1}{1 + \exp(2A|x_k|/\sigma^2)}, \qquad -\infty < x_k < \infty.$$

Since the maximum value of p_k is $1/2$, the probability that p_k exceeds the threshold $1/2-\varepsilon$ is calculated from (4.18) as follows:

$$\begin{aligned}\Pr(p_k \geq 1/2-\varepsilon) &= \int_I f(x_k)dx \\ &= \frac{1}{2}\frac{1}{\sqrt{2\pi\sigma^2}}\int_I \exp\left(\frac{-(x-A)^2}{2\sigma^2}\right)dx + \frac{1}{2}\frac{1}{\sqrt{2\pi\sigma^2}}\int_I \exp\left(\frac{-(x+A)^2}{2\sigma^2}\right)dx,\end{aligned}$$

where $I = \{x|p_k \geq 1/2-\varepsilon\}$ and $0 \leq \varepsilon \leq 1/2$. The limits of integration are easily found by solving $p_k \geq 1/2-\varepsilon$ for x and are given by

$$|x| \leq \frac{\sigma^2 K}{2A},$$

where $K = \ln(\frac{1+2\varepsilon}{1-2\varepsilon})$. A simple change of variables enables these integrals to be evaluated quite easily. Thus, the probability that the bit-error estimate exceeds the threshold $1/2-\varepsilon$ is given by

$$\Pr(p_k \geq 1/2-\varepsilon) = 1 - \mathrm{Q}\left(\frac{\sigma K}{2A} + \frac{A}{\sigma}\right) - \mathrm{Q}\left(\frac{\sigma K}{2A} - \frac{A}{\sigma}\right), \tag{4.38}$$

where $\mathrm{Q}(\cdot)$ is the Gaussian Q function. Note that the estimated values for σ^2 and A, given in (4.35) and (4.37), respectively, need to be used when calculating this probability for real applications. A quick sanity check shows that for $\varepsilon = 0$, the probability that p_k exceeds $1/2$ is zero.

A plot of equation (4.38) is illustrated in Figure 4.3 as a function of ε for different signal-to-noise ratios (SNR) A^2/σ^2. Figure 4.3 illustrates that for increasing SNR, the curve becomes very steep. Thus, for medium to high SNR and for ε close to $1/2$, the probability that the error estimate is inside a small interval is very large. For the 8dB case, 95% of those p_k's will be in the interval $[0, 0.01]$. Thus, if one of the estimates of p_k is large, then it is highly likely to be an error.

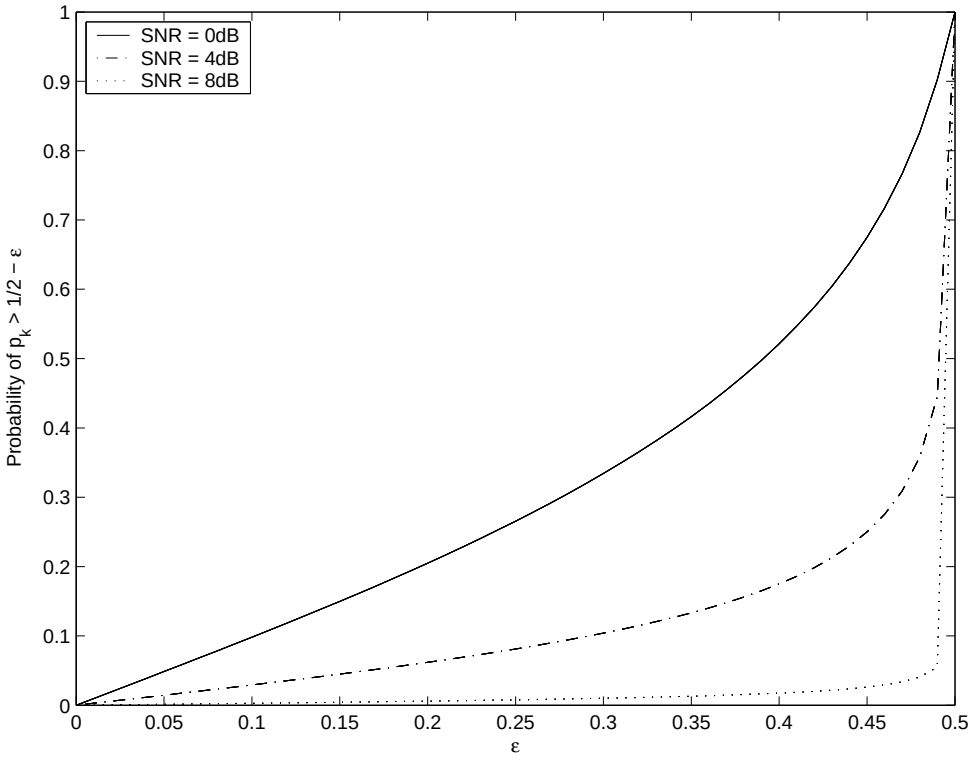

Figure 4.3: Probability of $p_k \geq 1/2 - \varepsilon$

Chapter 5

Error-Correction Codes: Mathematics and General Properties

The purpose of this chapter is to develop the mathematical preliminaries and the general properties of error-correction codes. The first two sections review the definitions of basic algebraic structures. The rest of the chapter is dedicated to the development of error-correcting codes and various decoding algorithms. More properties can be found in Hungerford [25], MacWillaims and Sloane [26], Reed and Chen [27], and Wicker [28].

5.1 Group, Ring, and Field

Definition 5.1 (Group) A group is an algebraic system $(G, \cdot)$ which consists of a set of elements on which a binary operation "$\cdot$" has been defined. The set is denoted by G, and $a \cdot b$ is the result of the binary operation on the two elements a and b in G. The binary operation satisfies the four following axioms:

1. Algebraic Closure: $a \cdot b \in G$ for all $a, b \in G$.
2. Associativity: $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in G$.
3. Identity: There exists an element $e \in G$ such that $a \cdot e = e \cdot a = a$ for all $a \in G$.
4. Inverse: For any element $a \in G$, there exists an element $b \in G$ such that $b \cdot a = e = a \cdot b$, where e is the identity element of the group. □

The following properties of a group can be deduced from Definition 5.1.

1. The identity element e of a group G is unique.

2. The inverse element of $a \in G$, denoted by a^{-1}, is unique in the group G.

3. $(a^{-1})^{-1} = a$ for all $a \in G$.

4. One can always solve $ax = b$ for arbitrary a and b in a group. The solution is $x = a^{-1} \cdot b \in G$.

5. The integer power of an element can be defined inductively on n by $a^{n+1} = a^n \cdot a$ for all positive integers n. Also one has $a^0 = e$ and $a^{-n} = (a^n)^{-1}$ for integer powers of a.

Note that $a \cdot b$ and $b \cdot a$ are not guaranteed to be equal in a group. A group such that $a \cdot b = b \cdot a$ for all a and b in the group is called a commutative or Abelian group. A group with a finite number of elements is called a finite group. A cyclic group is a group such that all elements are of the form g^n for some integer n, i.e., some integer power of an element g in the group. This element g is called a generator of the group. The next algebraic structure to be introduced is a ring.

Definition 5.2 (Ring) A ring $(R, +, \cdot)$ is a collection of elements with two binary operations, the ring addition $a + b$ and the ring multiplication $a \cdot b$ that satisfy the following four axioms:

1. $(R, +)$ is an Abelian group with zero as the additive identity.

2. Closure for the ring multiplication: $a \cdot b \in R$ for all $a, b \in R$.

3. Associativity: $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in R$.

4. Distributive Law: $a \cdot (b + c) = a \cdot b + a \cdot c$ and $(b + c) \cdot a = b \cdot a + c \cdot a$ for all $a, b, c, \in R$. □

The identity element of ring addition is called the zero of the ring, and is denoted by 0. The additive inverse of $a \in R$ is denoted by $-a$. Subtraction in a ring is given by $a - b = a + (-b)$ for all $a, b \in R$. If a ring has a multiplicative identity element e, satisfying $a \cdot e = e \cdot a = a$ for all $a \in R$, then the ring is called a ring with identity. The identity in the ring R is often denoted by 1. Also if the ring satisfies $a \cdot b = b \cdot a$ for all $a, b \in R$, the ring is called a commutative ring.

A typical example of a ring structure is the set of integers with integer addition and integer multiplication, denoted by $(\mathbb{Z}, +, \cdot)$. More precisely speaking, $(\mathbb{Z}, +, \cdot)$ is a commutative ring with identity. Another example is the ring $\mathbb{Z}_n$ consisting of the integers modulo some integer n, i.e., $\mathbb{Z}_n = \{0, 1, 2, \cdots, n-1\}$. In a ring one can add, subtract, and multiply arbitrary elements in the ring, but the existence of a multiplicative inverse of a ring element is not guaranteed. There are two basic substructures in a ring: subrings and ideals.

Definition 5.3 (Subring) Let $(R, +, \cdot)$ be a ring and S be a non-empty subset of R. If $(S, +, \cdot)$ is a ring, then S is called a subring of R. □

Definition 5.4 (Ideal) Let R be a commutative ring and I be a non-empty subset of R. Then I is called an ideal of R if

1. $(I, +, \cdot)$ is a subring of $(R, +, \cdot)$.

2. $a \cdot r = r \cdot a \in I$ for all $a \in I$ and $r \in R$. □

Example 5.1 Let R be a commutative ring and $a \in R$. The set of multiples of the element a is an ideal, denoted by $< a >$ or (a). Such an ideal is called a principal ideal and a is called a generator of this ideal. More generally, let $P = \{a_1, a_2, \cdots, a_n\} \subset R$. Then the set that consists of all the linear combination of the elements in P is an ideal of R, given by

$$< P >=< a_1, a_2, \cdots, a_n >= \left\{ \sum_{i=1}^{n} r_i a_i \middle| r_i \in R, 1 \leq i \leq n \right\}.$$

Let R be a ring and I be an ideal of R. Then for any $a \in R$, the set $\{a + n | n \in I\}$ is called a coset of ideal I with respect to element a of R. This coset is denoted as $a + I$ or sometimes by $\bar{a}$. Every element in $a + I$ is called a representative if it is in a coset. Two cosets $\bar{a}$, $\bar{b}$ are either identical, if $a - b \in I$, or disjoint if $a - b \notin I$. Let S denote the set that consists of all distinct cosets. Two operations $(\oplus, \odot)$ on S can be induced from the two operations $(+, \cdot)$ on R as follows: $\bar{a} \oplus \bar{b} = \overline{a + b}$, $\bar{a} \odot \bar{b} = \overline{a \cdot b}$. One can verify that the two operations $(\oplus, \odot)$ are well-defined on S as follows: Let $a^{'}$ and $b^{'}$ be different representatives of $\bar{a}$ and $\bar{b}$, respectively, i.e., $\bar{a} = \bar{a}^{'}$ and $\bar{b} = \bar{b}^{'}$. Then there exist $n_1, n_2 \in I$ such that $a = a^{'} + n_1$ and $b = b^{'} + n_2$. Hence, $a + b = a^{'} + n_1 + b^{'} + n_2 = a^{'} + b^{'} + n_1 + n_2$, and $a \cdot b = (a^{'} + n_1) \cdot (b^{'} + n_2) = a^{'} \cdot b^{'} + a^{'} \cdot n_2 + n_1 \cdot b^{'} + n_1 \cdot n_2$. Since $n_1 + n_2 \in I$ and $a^{'} \cdot n_2 + n_1 \cdot b^{'} + n_1 \cdot n_2 \in I$, one has that $\overline{a + b} = \overline{a^{'} + b^{'}}$ and $\overline{a \cdot b} = \overline{a^{'} \cdot b^{'}}$. Thus, $(\oplus, \odot)$ is well-defined on the set S. It is also easy to verify that the set S forms a ring under the two operations $(\oplus, \odot)$. This ring is called the quotient ring modulo I, denoted by R/I.

Example 5.2 Let $(\mathbb{Z}, +, \times)$ be the ring of integers. Clearly, $nZ = \{na | a \in \mathbb{Z}\} =< n >$ is an ideal that consists of multiples of n. Thus, the quotient ring $\mathbb{Z}/n\mathbb{Z}$ is the ring $\mathbb{Z}_n$ that consists of the integers modulo the integer n, denoted by $\mathbb{Z}/n\mathbb{Z} = \{0, 1, 2, \cdots, n - 1\}$. □

A more powerful algebraic structure, known as a field, can be defined upon a ring structure. A field is an algebraic system in which one can add, subtract, and multiply arbitrary elements and, in addition, one can divide any element by any non-zero element.

Definition 5.5 (Field) A field $(F, +, \cdot)$ is a commutative ring in which the non-zero elements form an Abelian group under multiplication. □

Definition 5.6 (Extension Field) A non-empty subset $F^{'}$ of a field F is a subfield if and only if $F^{'}$ is a field with the same operations of F. The field F is called an extension field of $F^{'}$. □

An example of a field is the real number system $(\mathbb{R}, +, \times)$. Note that $(\mathbb{R}, +, \times)$ is a field with an infinite number of elements. The set $\mathbb{Q}$ of all rational numbers is a subfield of $\mathbb{R}$.

A field with a finite number of elements is called a finite field or a Galois field. When a field is finite, the number of elements in F is called the order of F. A finite field of order q is denoted by $GF(q)$ or F_q.

Example 5.3 Let p be a prime integer. Then the integers, modulo p form a field of order p, denoted by $GF(p)$. The elements of $GF(p)$ are $\{0, 1, 2, \cdots, p-1\}$, and the additive and multiplicative operations are carried out $(\mathrm{mod}\, p)$, e.g., $GF(3)$ is the ternary field $\{0, 1, 2\}$ with $1 + 2 = 3 \equiv 0 \,(\mathrm{mod}\, 3), 2 \cdot 2 = 4 \equiv 1 \,(\mathrm{mod}\, 2), 1 - 2 = -1 \equiv 2 \,(\mathrm{mod}\, 3)$, etc. □

Let F be an arbitrary finite field. It is easy to see that there must exist an integer n such that $\sum_{i=1}^{n} 1 = 0$. Let p be the smallest integer such that $\sum_{i=1}^{p} 1 = 0$. The integer p is called the characteristic of the field F. The following propositions are listed without proof. The interested reader can consult the excellent algebra text by Hungerford [25].

Proposition 5.1 The characteristic p of a finite field is a prime number. □

Proposition 5.2 Let $GF(q)$ be a finite field of characteristic p. Then there exists an integer n such that $q = p^n$, and each element $a \in GF(q)$, $a \neq 0$, satisfies $a^q = a^{p^n} = a$. □

Clearly, the finite field $GF(q)$ is an extension of the field $GF(p)$, where p is the characteristic of the finite field $GF(q)$. The field $GF(p)$ is also called the ground field of $GF(q)$.

Proposition 5.3 The non-zero elements of the field $GF(q)$ form a cyclic group under multiplication of the field. This group is denoted by $GF(q)^*$ or F_q^*. □

Proposition 5.4 Let α be a nonzero element of a finite field $GF(q)$. Then $\alpha^{q-1} = 1$. □

Proposition 5.5 Let α be a nonzero element of a finite field $GF(q)$. Also, let n be the order of α. Then n divides $q - 1$. □

In a finite field $GF(q)$, a nonzero element α is said to be primitive if the order of α is $q - 1$. Therefore, the powers of a primitive element generate all of the nonzero elements

of $GF(q)$. If α is a primitive element, then α^k is also a primitive element provided that the $\gcd(k, q-1) = 1$, where $\gcd(\cdot, \cdot)$ denotes the greatest common divisor.

Proposition 5.6 Any finite field contains a primitive element. □

5.2 Univariate Polynomial Rings, Ideals, and Euclid's Division Algorithm

The univariate polynomial ring $F[x]$ over the field F consists of all univariate polynomials whose coefficients come from a field F. A typical element $f(x)$ in $F[x]$ is of the following form,

$$f(x) = f_n x^n + f_{n-1} x^{n-1} + \cdots + f_1 x^1 + f_0,$$

where x is the indeterminate of $F[x]$ and $f_k \in F$ for $k = 0, 1, \cdots, n$ are the coefficients of the polynomial $f(x)$. The degree of the polynomial $f(x)$, denoted by $\deg(f(x))$, is defined as the largest integer i such that $f_i \neq 0$. Let $\deg(f(x)) = n$. Then the coefficient f_n is called the leading coefficient. A polynomial is called a monic polynomial if its leading coefficient equals 1. A polynomial is said to be linear if its degree is one.

Proposition 5.7 (Euclid's Division Algorithm for Polynomials) Let $f(x), g(x) \in F[x]$. Then there exists unique polynomials $q(x), r(x) \in F[x]$ such that $f(x) = q(x)g(x) + r(x)$ with $\deg(r(x)) < \deg(g(x))$. The polynomial $q(x)$ is called the quotient polynomial, denoted by

$$q(x) = \left[\frac{f(x)}{g(x)}\right],$$

and $r(x)$ is called the remainder polynomial. □

Let $f(x), g(x) \in F[x]$. If there exists a polynomial $h(x) \in F[x]$ such that $f(x) = h(x)g(x)$, then $g(x)$ is said to be a factor or divisor of $f(x)$ and $f(x)$ is divisible by $g(x)$, denoted by $g(x)|f(x)$. A polynomial is said to be an irreducible polynomial on $F[x]$ if it has no divisor in $F[x]$ which has degree larger than 0, or in other words, it does not have a non-constant divisor. Let $f(x), g(x) \in F[x]$. The greatest common divisor of $f(x)$ and $g(x)$ is a

polynomial $h(x)$ with the greatest degree such that $h(x)|f(x)$ and $h(x)|g(x)$. The greatest common divisor of $f(x)$ and $g(x)$ is denoted by $\gcd(f(x), g(x))$. If $\gcd(f(x), g(x)) = 1$, then $f(x)$ and $g(x)$ are said to be relatively prime.

The greatest common divisor of two polynomials $f(x), g(x) \in F[x]$ can be computed by Euclid's algorithm as follows: Initially let $f_{-1} = g(x), f_0(x) = f(x)$. Then for $i = 1, 2, \cdots$, one recursively defines $f_i(x)$ as follows:

$$q_i(x) = \left[\frac{f_{i-2}(x)}{f_{i-1}(x)}\right], \quad f_i(x) = f_{i-2}(x) - q_i(x)f_{i-1}(x).$$

The recursion stops once $f_i(x) = 0$ for the first time at some $i = n$. It is easy to see that

$$\gcd(f_i(x), f_{i-1}(x)) = \gcd(f_{i-1}(x), f_{i-2}(x)),$$

and hence, one has $\gcd(f(x), g(x)) = \gcd(f_{n-1}(x), f_n(x)) = f_{n-1}(x)$.

The ideals constructed in $F[x]$ are called polynomial ideals. The following proposition is well known.

Proposition 5.8 Every polynomial ideal $I[x]$ of $F[x]$ is principal. □

Let $I[x] = (g(x))$, where $g(x) \in F[x]$. Then, by the Euclid's division algorithm for polynomials, any $f(x) \in F[x]$ can be expressed as follows:

$$f(x) = q(x)g(x) + r(x) \equiv r(x) \bmod g(x),$$

where $\deg(r(x)) < \deg(g(x))$. The set of all remainder polynomials, modulo a polynomial $g(x)$, forms a quotient ring under the operations multiplication and addition induced from $F[x]$ and is denoted by $F[x]/I[x] = F[x]/(g(x))$. If $g(x)$ is an irreducible polynomial over the field F, then the quotient ring $F[x]/(g(x))$ is actually a field. This field $F[x]/(g(x))$ is said to be an extension field of F by adding a root of $g(x)$ into F, and $g(x)$ is called the generator polynomial of the field $F[x]/(g(x))$. In this case, one has $g(\bar{x}) = 0$ where $\bar{x} = x + (g(x)) \in F[x]/(g(x))$. If $\bar{x}$ is a primitive element of the field $F[x]/(g(x))$, then

$g(x)$ is called a primitive polynomial. Let $f(x) \in F[x]$ be any polynomial over F. Then the splitting field of $f(x) \in F[x]$ is defined as the smallest extension field of F over which $f(x)$ can be factored as the product of linear polynomials.

Let $F \subset F^{'}$ be a field extension. An element $\alpha \in F^{'}$ is said to be an algebraic element over F if and only if there exists a polynomial $f(x) \in F[x]$ such that $f(\alpha) = 0$. The minimal polynomial of an algebraic element α over F is the monic polynomial with the least degree, denoted by $m_\alpha(x) \in F[x]$, such that $m_\alpha(\alpha) = 0$.

Proposition 5.9 Let $m_\alpha(x) \in F[x]$ be the minimal polynomial of α and $f(x) \in F[x]$ be any polynomial such that $f(\alpha) = 0$, then $m_\alpha(x)|f(x)$. □

5.3 Cyclic Codes

Let $F_q = GF(q)$ be a finite field of q elements. An encoded codeword of an (n, k) code over $GF(q)$ consists of n symbols with k message symbols and $n-k$ parity symbols. These parity symbols are derived from the k message symbols in the packet. Each symbol is an element of the same finite field $GF(q)$. The encoding operation maps a k-tuple message into an n-tuple codeword. The Hamming weight of a codeword is the number of non-zero symbols in a symbol sequence. Note that for binary signaling, the Hamming weight is the number of "1" bits in the binary sequence. The Hamming distance of two codewords is the number of distinct symbols in the two codewords. The minimum distance d of an (n, k) code is the minimum Hamming distance between any two distinct codewords.

The k-symbol message in a packet can be represented by a polynomial $m(x) \in GF(q)[x]$ as

$$m(x) = \sum_{i=0}^{k-1} m_i x^i, \tag{5.1}$$

where $m_i \in GF(q)$ and $(m_0, m_1, \cdots, m_{k-1})$ is the k-tuple message in the packet. The corresponding n-tuple codeword can be represented by the polynomial $c(x)$ over $GF(q)$ as

$$c(x) = \sum_{i=0}^{n-1} c_i x^i. \tag{5.2}$$

If a codeword $c(x)$ is corrupted by an error polynomial $e(x)$, then the received polynomial is modeled as

$$r(x) = c(x) + e(x) = \sum_{i=0}^{n-1} r_i x^i, \tag{5.3}$$

where the error polynomial is given by

$$e(x) = \sum_{i=0}^{n-1} e_i x^i. \tag{5.4}$$

Each term $e_i x^i$ with $e_i \neq 0$ represents a symbol error of the corrupted data. If the location of the error is known at the beginning of the decoding process, then it is called an erasure. An erasure is a symbol error with a known location in the corrupted codeword. That is, the erasure is some $e_v \neq 0$ for a known location $v \in [0, n-1]$. Errata are either symbol errors or erasures. Let v denote the number of erasures and t denote the number of symbol errors with unknown locations. The following proposition is well known.

Proposition 5.10 A code with a minimum Hamming distance d is capable of correcting t errors and v erasures into a unique codeword, provided that $2t + v \leq d - 1$. □

A code is called linear if all of the codewords form a vector space over $GF(q)$. That is, for any two codewords $c_1(x)$ and $c_2(x)$ and any two elements $\alpha, \beta \in GF(q)$, one has that $\alpha c_1(x) + \beta c_2(x)$ is also a codeword. Recall that the Hamming weight of a codeword is defined as the number of non-zero symbols in the codeword. For a linear code, the minimum distance of a code is the same as the minimum Hamming weight of all codewords of a code. A code is called cyclic if it is linear, and if a cyclic shift of a codeword is still a codeword. That is, if $c = (c_0, c_1, \cdots, c_{n-1})$ is a codeword, then $c' = (c_{n-1}, c_0, c_1, \cdots, c_{n-2})$ is also a codeword. It is well known that a cyclic code of length n is equivalent to an ideal $I(x)$ of the quotient ring $F_q[x]/(x^n - 1)$, where $F_q = GF(q)$. A multiplication by x in $F_q[x]/(x^n - 1)$ corresponds to a cyclic shift. Let $g(x)$ be the monic polynomial in $I(x)$ whose degree is the smallest. Then one has the following proposition.

Proposition 5.11 The ideal $I(x)$ is generated by $g(x)$ where $g(x)|(x^n-1)$. The polynomial $g(x)$ is called the generator of the cyclic code. □

Thus, any codeword in a cyclic code can be expressed by $c(x) = m(x)g(x)$, for some polynomial $m(x)$ with $\deg(m(x)) < n - \deg(g(x))$. Since one is free to choose the coefficients of $m(x)$, it is easy to see that the dimension of the code, denoted by k, is given by $k = n - \deg(g(x))$.

Let p be the characteristic of the field $GF(q)$ and n be the code-length of a cyclic code. Usually one also assumes that $\gcd(n, p) = 1$. For this case, there exists a smallest integer m such that $q^m \equiv 1 \bmod n$. Then the field $GF(q^m)$ is the splitting field of $x^n - 1$ over the field $GF(q)$. It is the smallest field that contains the field $GF(q)$, and over which the polynomial $x^n - 1$ can be factored into the product of linear factors. Thus, the n distinct n-th roots of unity, which are in $GF(q^m)$, form a cyclic multiplicative subgroup of $(GF(q^m))^*$. Here $(GF(q^m))^*$ denotes the multiplication group that consists of all non-zero elements of $GF(q^m)$.

Let γ be a primitive element of $GF(q^m)$, then a primitive n-th root of unity exists, denoted by α, given by

$$\alpha = \gamma^{(q^m-1)/n}.$$

As a consequence one has

$$x^n - 1 = \prod_{i=0}^{n-1}(x - \alpha^i).$$

Example 5.4 Consider $n = 47$ and $q = p = 2$, then one has $m = 23$ so that the splitting field of $x^{47} - 1$ is $GF(2^{23})$. The primitive 47-th root is given by $\alpha = \gamma^{(2^{23}-1)/47}$, where γ is a primitive element of $GF(2^{23})$. □

Definition 5.7 (Cyclotomic Coset, modulo n with respect to q) Let $\mathbb{Z}_n$ denote the integer ring modulo n. Then $\mathbb{Z}_n$ can be represented as a disjoint union of cyclotomic cosets C_i modulo n, which are given by $C_i = i, iq, iq^2, \cdots, iq^{m_i-1} \bmod n$, with m_i being the smallest integer such that $iq^{m_i} \equiv i \bmod n$. □

Note that if n is a prime number, then $m_i = m$ and $|C_i| = m$ for all $i \neq 0$, where m is the smallest integer such that $q^m \equiv 1 \bmod n$. In this case the number of disjoint non-zero cyclotomic cosets is equal to $(n-1)/m$.

Example 5.5 In Example 5.4 one has $n = 47, q = p = 2$ and $m = 23$. Thus, there are three different cyclotomic set C_0, C_1, C_5, given by

$$\begin{aligned}
C_0 &= 0 \\
C_1 &= 1, 2, 2^2, 2^3, \cdots, 2^{22} \bmod 47 \\
&= 1, 2, 3, 4, 6, 7, 8, 9, 12, 14, 16, 17, 18, 21, 24, 25, 27, 28, 32, 34, 36, 37, 42 \\
C_5 &= 55 \cdot 2, 5 \cdot 2^2, \cdots, 5 \cdot 2^{22} \bmod 47 \\
&= 5, 10, 11, 13, 15, 19, 20, 22, 23, 26, 29, 30, 31, 33, 35, 38, 39, 40, 41, 43, 44, 45, 46
\end{aligned}$$

and

$$|C_1| = |C_5| = 23,$$

where $|C_i|$ denotes the cardinality of the set C_i. □

Definition 5.8 (Fundamental Set) The set B_n that consists of all minimal representative indices of the cyclotomic cosets mod n in $\mathbb{Z}_n$, is called the fundamental set. Hence,

$$\mathbb{Z}_n = \bigcup_{r \in B_n} C_r \quad \text{and} \quad x^n - 1 = \prod_{r \in B_n} \prod_{i \in C_r} (x - \alpha^i).$$

Proposition 5.12 Let $GF(q) \subset GF(q^m)$ be a field extension and let $\alpha \in GF(q^m)$ with $\alpha^n = 1$. A polynomial

$$f(x) = \prod_{i \in K} (x - \alpha^i)$$

has all of its coefficients in the field $GF(q)$ if and only if, for all $i \in K$, $iq \,(\bmod n) \in K$, where K is an index set. □

Let $M_r(x)$ be given by

$$M_r(x) = \prod_{i \in C_r} (x - \alpha^i).$$

Then $M_r(x)$ is the minimal polynomial of α^r and

$$x^n - 1 = \prod_{r \in B_n} M_r(x).$$

Since the generator polynomial $g(x)$ of a linear cyclic code has all of its coefficients in the field $GF(q)$, it has the form

$$g(x) = M_{r_1}(x) \cdots M_{r_k}(x), \quad r_1, \cdots r_k \in B_n,$$

for some integer k, $1 \leq k \leq |B_n|$. Let $S = r_1, r_2, \cdots, r_k$. Then S is called the base set of the cyclic code. The set,

$$Q = \bigcup_{r \in S} C_r$$

is called the "complete defining set" of the code. As a consequence, a cyclic code can be defined by

$$C = \{c(x) | c(\alpha^i) = 0, \text{ for all } i \in Q\},$$

where α is the primitive n-th root of unity and $c(x)$ is a polynomial in the quotient ring, $GF(q)[x]/(x^n - 1)$.

5.4 Syndrome Equations and Decoding Binary BCH Codes

Let $g(x)$ be the generator polynomial of an (n, k, d) cyclic code. Let α be the n-th primitive root of unity. Clearly, any root of unity β can be expressed in terms of α as $\beta = \alpha^i$ for some integer i. Suppose the codeword $c(x)$ is corrupted by an error polynomial $e(x)$, and $r(x) = c(x) + e(x)$ is the received data that is corrupted. The syndromes are defined by

$$S_i = e(\alpha^i), \tag{5.5}$$

and the following is true for all integers i

$$S_{n+i} = S_i. \tag{5.6}$$

If α^i is a root of $g(x)$, then $g(\alpha) = 0$. Thus, one must have that $c(\alpha^i) = 0$ and hence, the syndrome S_i can be found by the relation,

$$S_i = r(\alpha^i) = e(\alpha^i). \tag{5.7}$$

Thus, the syndromes can be calculated from the corrupted data $r(x)$ and are called the known syndromes. The remaining syndromes are called the unknown syndromes and are calculated below. Let the error polynomial be expressed as

$$e(x) = \sum_{j=1}^{v} e_{n_j} x^{n_j},$$

where $e_{n_j} \neq 0$. Then the syndromes can be written as follows:

$$S_i = e(\alpha^i) = \sum_{j=1}^{v} e_{n_j} (\alpha^i)^{n_j} = \sum_{j=1}^{v} e_{n_j} ((\alpha)^{n_j})^i = \sum_{j=1}^{v} Y_j X_j^i, \tag{5.8}$$

where $Y_j = e_{n_j}$ and $X_j = (\alpha)^{n_j}$. The X_j's are called the error locators and the Y_j's are called the error magnitudes.

Proposition 5.13 The mapping between the unknown syndromes S_i, $i \in Q$, of a cyclic code and the error patterns $e(x)$ of weight $\leq t$ is one-to-one, where $t = \lfloor (d-1)/2 \rfloor$. $\square$

Proof: Suppose $e_1(x)$ and $e_2(x)$ are the error polynomials of two distinct error patterns of weight $\leq t$ corresponding to the same known syndromes. That is, for all $i \in Q$

$$S_i = e_1(\alpha^i) = e_2(\alpha^i).$$

Then $e_1(x) - e_2(x)$ must be a codeword of weight $\leq 2t \leq d$. This relation holds, if and only if, $e_1(x) - e_2(x) = 0$ or $e_1(x) = e_2(x)$. $\square$

A t-error correcting BCH code is constructed by first identifying the smallest field $GF(2^m)$ that contains a primitive n-th root of unity α. A binary generating polynomial $g(x)$ is then selected so that it has as zeros some $2t$ consecutive powers of α, i.e.,

$$g(\alpha) = g(\alpha^2) = \cdots = g(\alpha^{2t}).$$

A binary vector $\mathbf{c} = (c_0, c_1, \cdots, c_{n-1})$ is a code word if and only if its associated polynomial $c(x) = c_0 + c_1 x + c_2 x^2 + \cdots + c_{n-1} x^{n-1}$ has as zeros these same $2t$ consecutive powers of α. Now consider a received polynomial $r(x)$ which can be expressed as a sum of the transmitted code polynomial $c(x)$ and an error polynomial $e(x) = e_0 + e_1 x + \cdots + e_{n-1} x^{n-1}$. A series of syndromes is obtained by evaluating the received polynomial at the $2t$ zeros, given by

$$S_j = r(\alpha^j) = c(\alpha^j) + e(\alpha^j) = e(\alpha^j) = \sum_{k=0}^{n-1} e_k (\alpha^j)^k, \tag{5.9}$$

where $j = 1, 2, \ldots 2t$. The computations in equation (5.9) are performed in $GF(2^m)$, the field containing the primitive n-th root of unity. Now assume that the received word r has v errors in positions $i_1, i_2, \cdots, i_v$. Since the code is binary, the errors in these positions

have value $e_{i_l} = 1$. The syndrome sequence can be re-expressed in terms of these error locations as follows:

$$S_j = \sum_{l=1}^{v} e_{i_l}(\alpha^j)^{i_l} = \sum_{l=1}^{v} e_{i_l}(\alpha_l^i)^j = \sum_{l=1}^{v} X_l^j, \tag{5.10}$$

where $j = 1, 2, \ldots, 2t$. The $\{X_l\}$ are the error locators, for their values indicate the positions of the errors in the received word. If equation (5.8) is expanded, then a sequence of $2t$ algebraic syndrome equations in the v unknown error locations is obtained, given by

$$\begin{aligned} S_1 &= X_1 + X_2 + \cdots + X_v \\ S_2 &= X_1^2 + X_2^2 + \cdots + X_v^2 \\ S_3 &= X_1^3 + X_2^3 + \cdots + X_v^3 \\ &\vdots \\ S_{2t} &= X_1^{2t} + X_2^{2t} + \cdots + X_v^{2t}. \end{aligned} \tag{5.11}$$

Equations of this form are called power-sum symmetric functions. Since they form a system of nonlinear algebraic equations in multiple variables, they are difficult to solve in a direct manner. Peterson showed, however, that the BCH syndrome equations can be translated into a series of linear equations that are much easier to work with [29].

Let $\Lambda(x)$ be the error locator polynomial that has as its roots the inverses of the v error locators $\{X_l\}$.

$$\Lambda(x) = \prod_{l=1}^{v}(1 - X_l x) = \Lambda_v x^v + \Lambda_{v-1} x^{v-1} + \cdots \Lambda_{v1} x^1 + \Lambda_0 \tag{5.12}$$

Equation (5.12) can be used to express the coefficients of $\Lambda(x)$ directly in terms of the $\{X_l\}$'s as follows:

$$
\begin{aligned}
\Lambda_0 &= 1 \\
\Lambda_1 &= \sum_{i=1}^{v} X_i = X_1 + X_2 + \cdots + X_{v-1} + X_v \\
\Lambda_2 &= \sum_{i<j} X_i X_j = X_1X_2 + X_1X_3 + \cdots + X_{v-2}X_v + X_{v-1}X_v \\
\Lambda_3 &= \sum_{i<j<k} X_i X_j X_k = X_1X_2X_3 + X_1X_2X_4 + \cdots + X_{v-3}X_{v-1}X_v + X_{v-2}X_{v-1}X_v \\
&\vdots \\
\Lambda_v &= \prod_{i=1}^{v} X_i = X_1X_2\cdots X_{v-1}X_v.
\end{aligned}
\tag{5.13}
$$

The expressions in equation (5.13) are the elementary symmetric functions of the error locators. Power-sum symmetric functions and elementary symmetric functions are related by Newton's identities which are generally expressed as follows for polynomials over arbitrary fields

$$
\begin{aligned}
S_1 + \Lambda_1 &= 0 \\
S_2 + \Lambda_1 S_1 + 2\Lambda_2 &= 0 \\
S_3 + \Lambda_1 S_2 + \Lambda_2 S_1 + 3\Lambda_3 &= 0 \\
&\vdots \\
S_v + \Lambda_1 S_{v-1} + \Lambda_2 S_{v-2} + \cdots + \Lambda_{v-1} S_1 + v\Lambda_v &= 0 \\
S_{v+1} + \Lambda_1 S_v + \Lambda_2 S_{v-1} + \cdots + \Lambda_v S_1 &= 0 \\
&\vdots \\
S_{2t} + \Lambda_1 S_{2t-1} + \Lambda_2 S_{2t-2} + \cdots + \Lambda_v S_{2t-v} &= 0.
\end{aligned}
\tag{5.14}
$$

Newton's identities are linear in the v unknown coefficients of the error locator polynomial. Since the characteristic of the field is 2, equation (5.14) can be simplified by noting that

$j\Lambda_j = \Lambda_j$ if j is odd, and $j\Lambda_j = 0$ if j is even. The syndromes for the binary case have some additional useful structure, given below.

For a binary cyclic code of length n, let C_r be a cyclotomic coset of n. Then for any $i \in C_r$, one has that $i \equiv r \times 2^k \bmod n$ for some integer k. As a consequence, one has

$$s_i = S_{r \cdot 2^k} = S_r^{2^k}. \tag{5.15}$$

Hence, all the syndromes are dependent provided that their indices belong to the same cyclotomic coset of n. In other words, only the syndromes S_j for $j \in B_n$ are independent, where B_n is the fundamental set of n. The syndrome sequence for binary codes is thus highly constrained, with even-indexed syndromes being the squares of earlier-indexed syndromes. Given this constraint, it is not necessary to make use of all the Newton identities in order to obtain the coefficients of the error locator polynomial. Assume that $v = t$ errors have occurred, where t is the error correcting capability of the code. Newton's identities can be reduced to a system of t equations in t unknowns, as shown below.

$$\begin{aligned}
S_1 + \Lambda_1 &= 0 \\
S_3 + \Lambda_1 S_2 + \Lambda_2 S_1 + \Lambda_3 &= 0 \\
S_5 + \Lambda_1 S_4 + \Lambda_2 S_3 + \Lambda_3 S_2 + \Lambda_4 S_1 + \Lambda_5 &= 0 \\
&\vdots \\
S_{2t-1} + \Lambda_1 S_{2t-2} + \Lambda_2 S_{2t-3} + \cdots + \Lambda_t S_{t-1} &= 0.
\end{aligned} \tag{5.16}$$

5.5 Reed-Solomon Codes

Reed-Solomon (RS) codes are constructed and decoded by means of finite-field arithmetic. Recall that a finite field of q elements is denoted by $GF(q)$. The number of elements in a finite field is an integer of the form, $q = p^m$, where p is a prime number and m is an integer. The finite field $GF(q)$ is completely determined by its size q, however, in this study all of the codes are binary so that $p = 2$. Next let α be a primitive element in $GF(q)$, where α is a root of some primitive irreducible polynomial $p(x)$. Then the $(q-1)$ consecutive powers of α, namely $\{1, \alpha, \alpha^2, \ldots, \alpha^{q-2}\}$, must be distinct.

The number of information symbols k is called the dimension of the the RS code since the RS codewords form a vector space of dimension k over $GF(q)$. Also, the code length of n symbols equals q since each codeword has q coordinates. When RS codes are discussed, they are usually denoted by their length n and dimension k in symbols as (n, k) codes. All symbols are equal length binary sub-words of m bits. For example, the finite field $GF(256)$ is used in many applications because each of the $2^8 = 256$ field elements can be represented as an $m = 8$ bit sequence or a byte.

A Reed-Solomon code is a cyclic code over a field $GF(q)$ with code length $n = q - 1$. The dimension of the RS code can be any integer k satisfying $1 \leq k \leq q-2$. The generator polynomial is of the form

$$g(x) = \prod_{i=h}^{h+2t-1} (x - \alpha^i),$$

where h is an integer constant to be chosen, and α is a primitive element of the finite field $GF(q)$. Note that if $h = 1$, then the RS code is narrow-sense. For any message polynomial $m(x)$, with $\deg(m(x)) \leq k$, the encoded codeword is

$$c(x) = x^{n-k} m(x) + p(x),$$

where $p(x)$ is the parity polynomial of degree $\leq n - k$, given by

$$p(x) = -x^{n-k} m(x) \bmod g(x).$$

Proposition 5.14 The minimum distance of a (n, k) RS code is $d = n - k - 1$. □

It is well known that the minimum distance of a linear (n, k) code satisfies $d \leq n - k - 1$. A linear (n, k) code with minimum distance $d = n - k - 1$ is also called a maximum distance separable (MDS) code. Hence, RS codes are all MDS codes. The maximum burst-error correction capability for an (n, k) linear code satisfies $b \leq (n - k)/2$, where b is the length of the burst error in bits or symbols for non-binary codes. Evidently, for RS codes, the maximum random error-correction capability equals or coincides with the maximum burst-error-correction capability since $t = (d - 1)/2 = (n - k)/2 = b$ symbols. Therefore, RS codes are the most powerful block codes for both random and burst-error correction.

5.6 The Berlekamp-Massey Algorithm

The goal of this section is to give a detailed analysis of the Berlekamp-Massey (BM) algorithm. By using the syndrome equations and linear recursion, the BM algorithm is able to locate the symbol errors in a received word. The discussion closely follows the one given in Wicker [28, pp. 214-224].

Assume that some v errors have corrupted the received word. The syndromes are given in equation (5.9) and reproduced here for convenience as

$$S_j = e(\alpha^j) = \sum_{k=0}^{n-1} e_k(\alpha^j)^k = \sum_{l=1}^{v} e_{i_l} X_l^j. \tag{5.17}$$

Equation (5.17) defines a series of $2t$ algebraic equations in $2t$ unknowns, given by

$$\begin{aligned} S_1 &= e_{i_1} X_1 + e_{i_2} X_2 + \cdots + e_{i_v} X_v \\ S_2 &= e_{i_1} X_1^2 + e_{i_2} X_2^2 + \cdots + e_{i_v} X_v^2 \\ &\vdots \\ S_{2t} &= e_{i_1} X_1^{2t} + e_{i_2} X_2^{2t} + \cdots + e_{i_v} X_v^{2t}. \end{aligned} \tag{5.18}$$

Unlike the binary case, the syndrome equations are not power-sum symmetric functions. It is still possible, however, to reduce the system in (5.18) to a set of linear functions in the unknown quantities. To facilitate this goal, make use of an error locator polynomial $\Lambda(x)$ whose zeros are the inverses of the error locators $\{X_i\}$

$$\Lambda(x) = \prod_{l=1}^{v}(1 - X_l x) = \Lambda_v x^v + \Lambda_{v-1}x^{v-1} + \cdots + \Lambda_1 x + \Lambda_0. \tag{5.19}$$

It follows immediately that for some error locator X_l,

$$\Lambda(X_l^{-1}) = \Lambda_v X_l^{-v} + \Lambda_{v-1}X_l^{-v+1} + \cdots + \Lambda_1 X_l^{-1} + \Lambda_0 = 0. \tag{5.20}$$

Multiply both sides of (5.20) by $e_{i_l}X_l^j$ to yield

$$\begin{aligned} &e_{i_l}X_l^j(\Lambda_v X_l^{-v} + \Lambda_{v-1}X_l^{-v+1} + \cdots + \Lambda_1 X_l^{-1} + \Lambda_0) \\ &\qquad = e_{i_l}(\Lambda_v X_l^{-v+j} + \Lambda_{v-1}X_l^{-v+j+1} + \cdots + \Lambda_1 X_l^{j-1} + \lambda_0 X_l^j) = 0. \end{aligned} \tag{5.21}$$

Sum equation (5.21) over all indices l, which yields an expression from which Newton's identities can be constructed

$$\begin{aligned} &\sum_{l=1}^{v} e_{i_l}(\Lambda_v X_l^{j-v} + \Lambda_{v-1}X_l^{j-v+1} + \cdots + \Lambda_1 X_l^{j-1} + \lambda_0 X_l^j) \\ &= \Lambda_v \sum_{l=1}^{v} e_{i_l}X_l^{j-v} + \Lambda_{v-1}\sum_{l=1}^{v} e_{i_l}X_l^{j-v+1} + \cdots + \Lambda_1 \sum_{l=1}^{v} e_{i_l}X_l^{j-1} + \Lambda_0 \sum_{l=1}^{v} e_{i_l}X_l^j \\ &= \Lambda_v S_{j-v} + \Lambda_{v-1}S_{j-v+1} + \cdots + \Lambda_1 S_{j-1} + \Lambda_0 S_j = 0. \end{aligned} \tag{5.22}$$

From equation (5.19) it is clear that Λ_0 is always one. Thus, syndrome S_j can be expressed in recursive form as a function of the coefficients of the error locator polynomial $\Lambda(x)$ and the earlier syndromes $S_{j-1}, \ldots, S_{j-v}$ as follows:

$$\Lambda_v S_{j-v} + \Lambda_{v-1}S_{j-v+1} + \cdots + \Lambda_1 S_{j-1} = -S_j. \tag{5.23}$$

Expressions of this form can be given the physical interpretation through the use of a linear feedback shift register (LFSR) (Wicker [28, pp. 218]). Hence, the problem of decoding BCH and Reed-Solomon codes can thus be reexpressed as follows: find a LFSR of minimal length such that the first $2t$ elements in the LFSR output sequence are the syndromes $S_1, S_2, \ldots, S_{2t}$. The taps of this shift register provide the desired error locator polynomial $\Lambda(x)$.

To illustrate the algorithm, let $\Lambda^{(k)}(x) = \Lambda_k x^k + \Lambda_{k-1} x^{k-1} + \cdots + \Lambda_1 x + 1$ be the connection polynomial of length k whose coefficients specify the taps of the length k LFSR. The Berlekamp-Massey algorithm starts by finding $\Lambda^{(1)}$ such that the first element output by the corresponding LFSR is the first syndrome S_1. The second output of the LFSR is then compared to the second syndrome. If the two do not have the same value, then the discrepancy between the two is used to construct a modified connection polynomial. If there is no discrepancy, then the same connection polynomial is used to generate a third sequence element, which is compared to the third syndrome. The process continues until a connection polynomial is obtained that specifies an LFSR capable of generating all $2t$ elements of the syndrome sequence. Dr. Massey showed that, given an error pattern of weight $v \leq t$, the connection polynomial resulting from the Berlekamp-Massey algorithm uniquely specifies the correct error locator polynomial. For detailed proofs of this assertion and the overall validity of the algorithm, the diligent reader is referred to Massey [30] and Berlekamp [1].

The algorithm has five parameters: the connection polynomial $\Lambda^{(k)}(x)$, the correction polynomial $T(x)$, the discrepancy $\Delta^{(k)}$, the length L of the shift register, and the indexing variable k. The algorithm proceeds as follows:

The Berlekamp-Massey shift register synthesis decoding algorithm

1. Compute he syndrome sequence $S_1, \ldots, S_{2t}$ for the received word.
2. Initialize the algorithm variables as follows: $k = 0$, $\Lambda^{(0)}(x) = 1$, $L = 0$, and $T(x) = x$.
3. Set $k = k + 1$. Compute the discrepancy $\Delta^{(k)}$ by subtracting the k-th output of the LFSR defined by $\Lambda^{(k-1)}(x)$ from the k-th syndrome

$$\Delta^{(k)} = S_k - \sum_{i=1}^{L} \Lambda_i^{(k-1)} S_{k-i}.$$

4. If $\Delta^{(k)} = 0$, then go to step 8.
5. Modify the connection polynomial: $\Lambda^{(k)}(x) = \Lambda^{(k-1)}(x) - \Delta^{(k)} T(x)$.
6. If $2L \geq k$, then go to step 8.
7. Set $L = k - L$ and $T(x) = \Lambda^{(k-1)}(x)/\Delta^{(k)}$.
8. Set $T(x) = x \cdot T(x)$.
9. If $k < 2t$, then go to step 3.
10. Determine the roots of $\Lambda(x) = \Lambda^{(2t)}(x)$. If the roots are distinct and lie in the right field, then determine the error magnitudes, correct the corresponding locations in the received word, and stop.
11. Declare a decoding failure and stop.

Example 5.6 Find the error locator polynomial for a (7,3) Reed-Solomon Code using the Berlekamp-Massey algorithm.

This code can correct $t = (n - k)/2 = 2$ errors. Let the received polynomial be $r(x) = \alpha^2 x^6 + \alpha^2 x^4 + x^3 + \alpha^5 x^2$. The first four syndromes are calculated from the received polynomial, given by $S_1 = \alpha^6$, $S_2 = \alpha^3$, $S_3 = \alpha^4$, $S_4 = \alpha^3$. The algorithm generates the following set of connection polynomials, discrepancies, and correction polynomials.

Table 5.1: Finding the error locator polynomial

k	S_k	$\Lambda^{(k)}(x)$	$\Delta^{(k)}$	L	$T(x)$
0	—	1	—	0	x
1	α^6	$1+\alpha^6 x$	$S_1 - 0 = \alpha^6$	1	αx
2	α^3	$1+\alpha^4 x$	$S_2 - \alpha^5 = \alpha^2$	1	αx^2
3	α^4	$1+\alpha^4 x+\alpha^6 x^2$	$S_3 - 1 = \alpha^5$	2	$\alpha^2 x+\alpha^6 x^2$
4	α^3	$1+\alpha^2 x+\alpha x^2$	$S_4 - \alpha^4 = \alpha^6$	—	—

The algorithm terminates when $k \not< 2t$. In this example, the error locator polynomial is given by $\Lambda(x) = \alpha x^2 + \alpha^2 x + 1$. The roots of $\Lambda(x)$ are found using the Chien Search, given by α^2 and α^4. Finally, the location of the errors are found by computing the inverse of the roots and are given by α^3 and α^5.

Chapter 6

Decoding Algorithms

The purpose of this chapter is to illustrate how the individual bit-error probabilities of a codeword can be used to improve the decoding algorithms of various error-correction codes. New algorithms are developed that reduce the decoding complexity, in terms of CPU time, of the (23,12,7) Golay code and the (47,24,11) Quadratic Residue code. The last section gives a detailed analysis of erasure decoding of Reed-Solomon codes. A new method is developed to estimate the codeword symbols with the greatest probability of error using the information about the bit-error probabilities. The ability to determine the codeword symbols with the greatest probability of error at the beginning of the decoding process is a necessary requirement for erasure decoding of Reed-Solomon codes.

6.1 Reliability-Based Decoding of the (23,12,7) Golay Code

The Golay code C_{23} is the (23,12,7) Quadratic Residue code over GF(2^{11}). The defining set is the set of quadratic residues mod 23, given by Q = $\{1, 2, 3, 4, 6, 8, 9, 12, 13, 16, 18\}$. Let α be a primitive 23-rd root of unity in $GF(2^{11})$. The distinct powers of α form two cyclotomic cosets modulo 23 with respect to $GF(2)$, given by: $C_1 = \{1, 2, 3, 4, 6, 8, 9, 12, 13, 16, 18\}$ and $C_2 = \{5, 7, 10, 11, 14, 15, 17, 19, 20, 21, 22\}$. Thus, $x^{23} + 1$ factors into three binary irreducible polynomials as follows:

$$x^{23} + 1 = (x + 1) \cdot (x^{11} + x^{10} + x^6 + x^5 + x^4 + x^2 + 1) \cdot (x^{11} + x^9 + x^7 + x^6 + x^5 + x + 1).$$

Depending on the selection of α, there are two possible generator polynomials for C_{23}

$$g_1(x) = x^{11} + x^{10} + x^6 + x^5 + x^4 + x^2 + 1$$
$$g_2(x) = x^{11} + x^9 + x^7 + x^6 + x^5 + x + 1.$$

By using either one of these generating polynomials, the resulting code can be shown to have the following properties [26]:

1. Every codeword in C_{23} whose weight is even has weight divisible by 4.

2. The minimum distance of the C_{23} is given by $d = 7$.

The (23,12,7) Golay code is a perfect code in the sense that the codewords and their three-error correction spheres exhaust the vector space of 23-bit binary vectors. Let t be the error correcting capability of the code. Then from the inequality, $2t+1 \leq d$, the Golay code can correct three errors.

The codewords of Golay code over GF(2) are first expressed as the coefficients of a polynomial, given by

$$c(x) = \sum_{i=0}^{22} c_i x^i,$$

where $c_i \in \mathrm{GF}(2)$ and x is an indeterminate. Now let the polynomials

$$i(x) = c_{22}x^{22} + c_{21}x^{21} + \cdots + c_{11}x^{11}$$

and

$$p(x) = c_{10}x^{10} + c_9 x^9 + \cdots + c_1 x + c_0$$

be the information and parity check polynomials of the codeword $c(x)$, respectively. Thus, the codeword can be represented as the sum of the information and parity check polynomials, given by

$$c(x) = i(x) + p(x).$$

To be a (23,12,7) cyclic BCH codeword, $c(x)$ must also be a multiple of the generating polynomial $g(x)$, that is, $c(x) = q(x)g(x)$. The polynomial $p(x)$ is obtained by dividing $i(x)$ by $g(x)$ as follows

$$i(x) = q(x)g(x) + r(x),$$

where $r(x)$ is the remainder polynomial of degree less than 11. Then one sets $p(x) = r(x)$. Thus, the following identities are true

$$q(x)g(x) = i(x) + r(x) = i(x) + p(x) = c(x).$$

A code generated in this manner is a cyclic BCH code with parity-check polynomial $r(x)$.

6.1.1 BCH Decoding Algorithm

The BCH decoding algorithm, when applied to the (23,12,7) binary Golay Code, can correct all patterns of two or fewer errors. To illustrate this method, define the error polynomial as

$$e(x) = e_{22}x^{22} + e_{21}x^{21} + \cdots + e_1x + e_0.$$

Then the received polynomial has the form

$$r(x) = c(x) + e(x).$$

Suppose that v errors occur in $r(x)$, and further assume that $2t \leq d-1$. The decoder begins by dividing the received polynomial $r(x)$ by the generator polynomial $g(x)$ as follows

$$r(x) = v(x)g(x) + s(x),$$

where $\deg[s(x)] < \deg[g(x)]$. Since α and α^3 are both roots of the generating polynomial $g(x)$, one has

$$s_1 \triangleq r(\alpha) = s(\alpha) \quad \text{and} \quad s_3 \triangleq r(\alpha^3) = s(\alpha^3),$$

where s_1 and s_3 are called the syndromes of the code. The error-locator polynomial is defined by [1, pg. 19]

$$L(z) = \prod_{i=1}^{v}(z - Z_i) = z^v + \sum_{j=1}^{v} \sigma_j z^{v-j}.$$

Here, Z_j for $1 \leq j \leq v < t$ are the locations of the v errors. That is, $Z_j = \alpha^{r_j}$ indicates that an error has occurred in the r_j position. After reception of the received polynomial $r(x)$, the two-error correcting decoder computes the syndromes s_1 and s_3 as well as the error-locator polynomial. Depending on the number of errors, the BCH decoding algorithm satisfies the following scheme [1, pg. 19]

$$L(z) = \begin{cases} 1, & \text{no errors, } s_1 = s_3 = 0 \\ 1 + s_1 z, & \text{one error, } s_1 \neq 0; s_3 = s_1^3 \\ 1 + s_1 z + (s_1^2 + \frac{s_3}{s_1})z^2, & \text{two errors, } s_1 \neq 0; s_3 \neq s_1^3. \end{cases}$$

The error locations are given by the inverse of the roots of $L(z)$ provided there are no more than two errors. If there are three errors, then the two roots of $L(z)$ are not among the twenty-third roots of unity and a decoding failure is declared.

6.1.2 Shift-Search Decoding Algorithm

The shift-search algorithm, developed by Reed [2], sequentially inverts the information bits until the third error is canceled. It then utilizes the BCH decoding algorithm to correct the remaining two errors. The following theorem, given in [2], is essential for the shift-search algorithm to successfully decode the third error for the (23,12,7) Golay code.

Theorem 6.1 Let $\mathbf{e}_4$ be any error vector of weight 4 and $\mathbf{c}$ be any code vector of the (23,12,7) Golay code. Then

$$\mathbf{c} + \mathbf{e}_4 = \mathbf{c}_1 + \mathbf{e}_3,$$

where $\mathbf{c}_1$ is some other code vector, and $\mathbf{e}_3$ is some error vector of weight 3. □

In other words, adding an error vector of weight 4 to a codeword of the Golay code produces a 23 bit vector that is equal to some other codeword plus and error vector of weight 3.

From the theorem it can be deduced that shift-search is a complete decoding algorithm able to correct up to three errors. The overall decoding of the (23,12,7) Golay Code using the shift-search algorithm is summarized by the following steps:

1. Apply the BCH decoding algorithm to the received 23-bit word. If there are no more than two errors, then the error pattern can be corrected. However, if both roots of $L(z)$ are not among the twenty-third roots of unity, then three errors are detected.

2. Invert the first information bit and apply the BCH decoding algorithm. If this inversion correctly cancels the third error, then the error pattern can be corrected. However, if the first information bit was originally correct, there are now 4 errors in the received word. By theorem 1, the BCH decoding algorithm detects that three errors still remain. Thus, the decoder flips back the inverted information bit.

3. Repeat step 2 for all the information bits. If 3 errors are detected at the end of the 12th information bit, then conclude that all errors are confined to the parity check bits. Hence, all the information bits are correct.

6.1.3 Reliability-Search Algorithm

The reliability-search algorithm utilizes the "soft" information contained in the matched filter outputs x_k to determine the probability of error for each bit. The first step in the reliability-search algorithm is to calculate the bit-error probabilities corresponding to the 12 information bits. Next, invert the information bit with the highest probability of error. If the bit inversion correctly cancels the third error, then the BCH decoding algorithm corrects the remaining two errors. However, if the inverted information bit was indeed correct, then this inversion yields 4 errors in the received word. By theorem 1, the BCH decoding algorithm detects that three errors still remain. Thus, flip back the

inverted information bit and invert the information bit with the second highest probability of error. The procedure above is repeated recursively for at most 12 steps. If 3 errors are detected at the end of the 12th iteration, then it can be deduced that all errors are confined in the parity check bits.

The reliability-search algorithm is summarized as follows:

1. Given the received data, calculate the bit error probabilities for the information bits.

2. Apply the BCH decoding algorithm. If no more than two errors occur, then the error pattern can be corrected. However, if the roots of $L(z)$ are not among the twenty-third roots of unity, then three errors are detected.

3. Invert the information bit with the highest probability of error and apply the BCH decoding algorithm. If the inversion correctly cancels the third error, then decoding stops. If not, flip back that information bit and select the information bit with the second highest probability of error.

4. Repeat step 3 for all the information bits. If 3 errors detected at the end, then all errors are confined to the parity check bits.

Simulations were conducted using the C programming language to determine the decoding complexity of both algorithms and the frequency distributions. 10,000 Monte-Carlo simulations were conducted on a 1.4 GHz Intel Pentium Processor at a signal-to-noise ratio of 6dB. The reliability-search algorithm decoded the third error in 1.73 ms whereas the shift-search algorithm took 2.95 ms. This results in a reduction of decoding complexity by 41.3 %.

The frequency distributions for correctly estimating the third error for the reliability-search and shift-search algorithms are shown in Figures 6.1 and 6.2, respectively. Figure 6.1 illustrates that the reliability-search algorithm correctly identifies the 3rd error within two attempts with a high probability. In both figures, 13 attempts means that all the errors are confined to the parity check bits and the message is correct. Note that the frequency

of 13 attempts is the same in both algorithms as expected. The frequency distribution of the shift-search algorithm requires many iterations to cancel the third error. It is seen from Figure 6.2, that the shift-search algorithm follows a geometric distribution up to the 12th iteration.

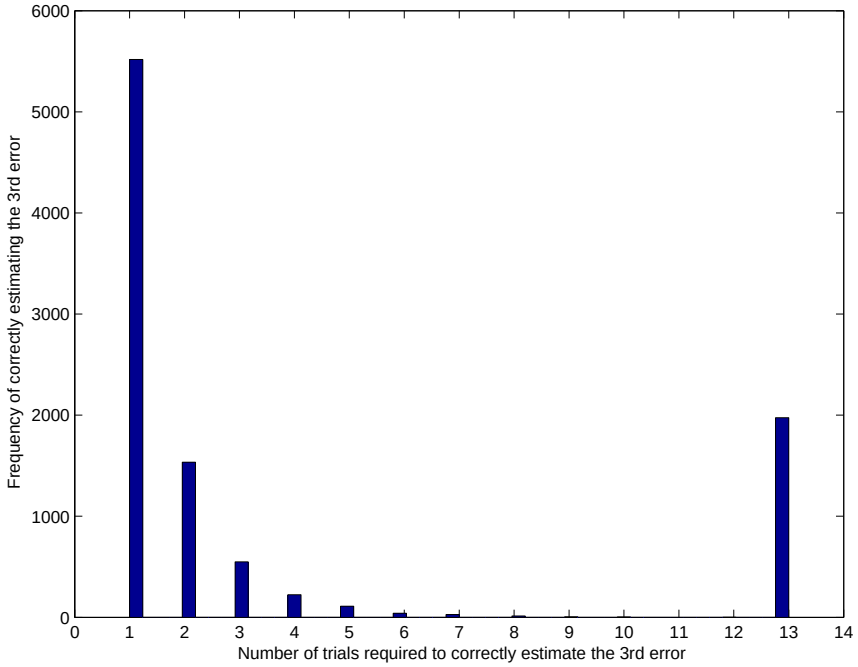

Figure 6.1: Reliability-search frequency distribution

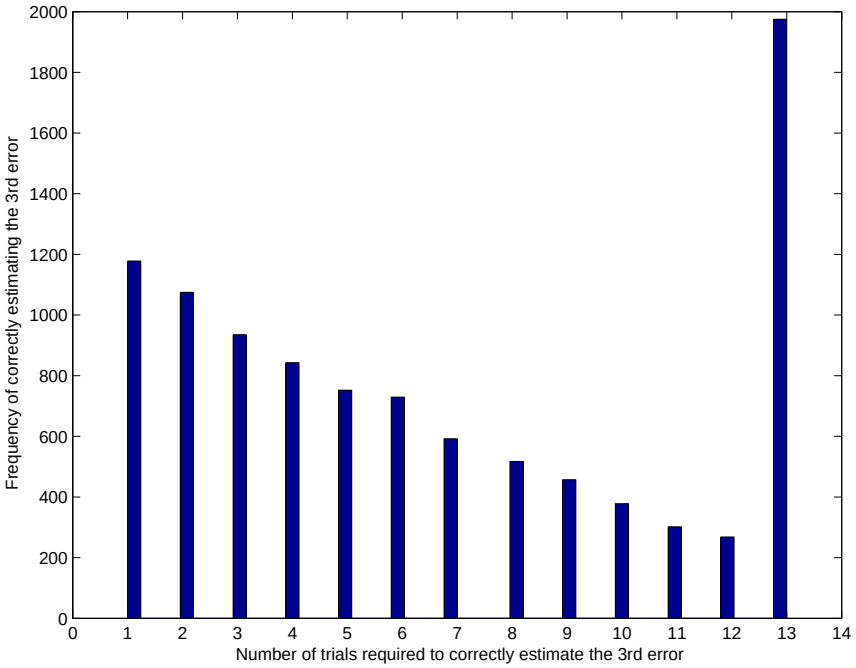

Figure 6.2: Shift-search frequency distribution

6.2 Fast Decoding of the (47,24,11) QR Code

Definition 6.1 (The Quadratic Residues) Let n be a prime integer. The nonzero squares modulo n, $1^2, 2^2, 3^2, \cdots, \bmod n$, are called the quadratic residues modulo n. □

Let Q denote the set of quadratic residues, denoted by

$$Q = \{1^2, 2^2, 3^2, \cdots, \bmod n\},$$

and let N be the set of non-residues. Define the polynomials

$$q(x) = \prod_{i \in Q} (x - \alpha^i), \qquad n(x) = \prod_{i \in N} (x - \alpha^i),$$

where α is a primitive n-th root of unity. In order that Q forms a complete defining set of some cyclic code, one must have $q \in Q$. That is, q is a quadratic residue mod n.

Definition 6.2 (Quadratic Residue Code) Quadratic residue codes Q, $Q^{'}$, N, $N^{'}$ are defined to be cyclic codes with the following generating polynomials: $q(x)$, $(x-1)q(x)$, $n(x)$, $(x-1)n(x)$, respectively. □

Unless otherwise stated, all QR codes in this study are generated by $q(x)$. If $q = 2$, then the QR code becomes a binary QR code. It is shown in MacWillaims and Sloane [26], that the codeword length of a binary QR code must be a prime number of the form, $n = 8m \pm 1$. In this case one has $2 \in Q$, and 2 is a quadratic residue mod $n = 8m \pm 1$. Since $|Q| = (n-1)/2$, the binary QR code is a $(n, (n-1)/2, d)$ code for some minimum distance d.

Example 6.1 ((47,24,11) QR code) For n = 47, the set of quadratic residues are given by $Q = C_1 = \{1, 2, 3, 4, 6, 7, 8, 9, 12, 14, 16, 17, 18, 21, 24, 25, 27, 28, 32, 34, 36, 37, 42\}$. The base set is $S = \{1\}$ and $GF(2^{23})$ is the splitting field of $x^{47} - 1$. □

6.2.1 Reliability-Search Algorithm

Recently, He and Reed in [3] have developed an algebraic decoding algorithm for the QR (47,24,11) code that can correct up to five errors. However, the computational complexity is very high to decode the fifth error in terms of CPU time. The reason for this can be explained as follows: The algebraic decoding algorithm needs to calculate the greatest common divisor (gcd) of two polynomials for both the four and five error cases. For the four error case, the decoder must calculate the $\gcd(f_1, f_2)$, where $\deg(f_1) = 3$ and $\deg(f_2) = 33$. For this case, the complexity of $\gcd(f_1, f_2)$ is acceptable. However, for the five error case, the decoder must calculate $\gcd(g_1, g_2)$, where $\deg(g_1) = 34$ and $\deg(g_2) = 258$. Clearly, the complexity to compute $\gcd(g_1, g_2)$ is very high and considerable more memory is needed.

In order to circumvent calculating the greatest common divisor of such high degree polynomials, the reliability-search algorithm can be utilized to decode the fifth error. Suppose that more than four errors have occurred and are detected. The reliability-search algorithm is now developed to decode the code for the five-error case. First the calculate the bit-error probabilities for all 47 bits of the received word. Next apply the

four-error decoding algorithm to the received word. If more than four errors are detected, then invert the bit with the highest probability of error and apply the four-error decoding algorithm. If the bit inversion correctly cancels the fifth error, then the four-error decoding algorithm corrects the remaining four errors. However, if the inverted information bit was indeed correct, then this inversion produces six errors and the four-error decoding algorithm detects that more than four errors have occurred. Hence, flip back the inverted bit and invert the bit with the second highest probability of error. The procedure above is repeated recursively until the fifth error is canceled. A pseudo code for the reliability-search decoding algorithm is given next as follows:

1. Estimate the bit-error probabilities for the 47 bits.
2. Apply four-error decoding algorithm.
3. If more than four errors detected, then invert the bit with highest probability of error.
4. Apply the four-error decoder again.
5. If the inverted information bit is actually correct, then this inversion introduces 6 errors and the four-error decoding algorithm still detects more than four errors have occurred.
6. Flip back the inverted bit and invert the bit with the second highest probability of error.
7. Repeat.

10,000 Monte-Carlo simulations were conducted on a 1.4 GHz Intel Pentium Processor at SNR of 6dB. The algebraic decoding algorithm took 0.76 ms to decode the fifth error, the shift-search algorithm took 1.35 ms to decode the fifth error, whereas the reliability-search algorithm decoded the fifth error in 0.59 ms. This results in a reduction of decoding complexity in terms of CPU time by 22% and 56%, respectively.

The frequency distributions for correctly estimating the fifth error for the reliability-search and shift-search algorithms are shown in Figure 6.3. The reliability-search algorithm correctly identifies the fifth error within two attempts with a high probability whereas the shift-search algorithm requires many iterations to cancel the third error.

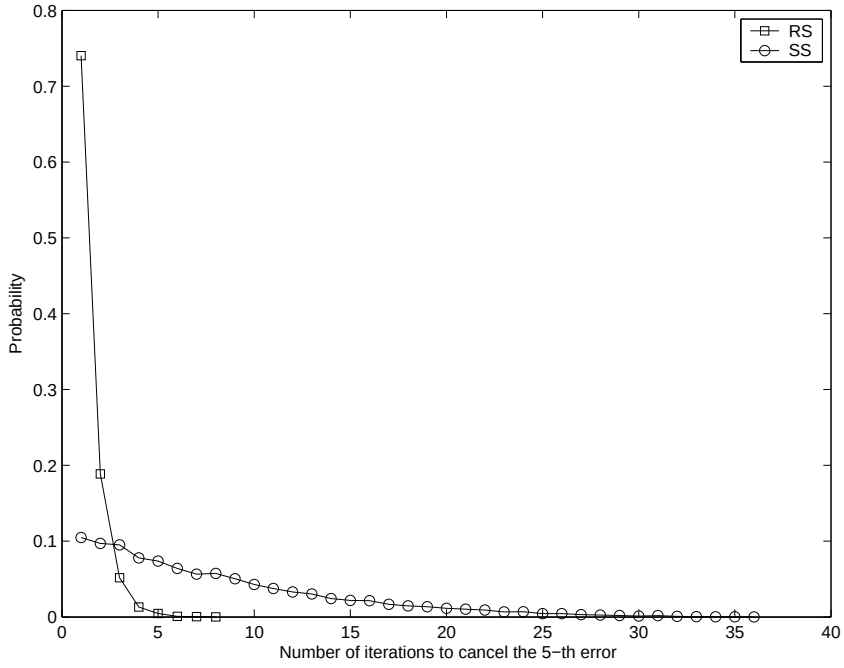

Figure 6.3: Probability mass functions

6.3 Erasure Decoding of Reed-Solomon Codes

An erasure is a symbol error with its location known to the decoder. However, in current or traditional decoding procedures, the locations of the error symbols are not known a-priori and must be estimated. Knowledge of the bit-error probabilities enable the decoder to determine, with considerable confidence, the symbols that have the highest likelihood of being in error. Once the locations of the most likely symbol errors are known, the decoder only needs to calculate the amplitudes of these symbol erasures to decode the codeword.

To illustrate this method, recall that the bit-error probability estimate for the i-th bit in codeword is given by

$$p_i = \frac{1}{1 + \exp(2A|x_i|/\sigma^2)}, \tag{6.1}$$

where A is the received bit amplitude, x_i is the observation, and σ^2 is the channel noise power. Suppose a codeword consisting of n symbols is transmitted over a noisy channel. Because of channel anomalies and front-end receiver noise, the received word is a corrupted version of the original transmitted codeword. Since p_i is the probability that the i-th received bit is an error, clearly $(1 - p_i)$ is the probability that the i-th bit is correct. Suppose that there are b bits in a symbol and all the bits are independent. Then the probability that the j-th symbol is correct is given by

$$P_c^j = \prod_{i=bj}^{bj+b-1} (1 - p_i) = (1 - p_{bj})(1 - p_{bj+1}) \cdots (1 - p_{bj+b-1}),$$

where j takes on the values in the index set $J = \{0, 1, 2, \ldots, n-1\}$. Thus, the probability of the j-th symbol error is given by

$$P_e^j = 1 - \prod_{i=bj}^{bj+b-1} (1 - p_i). \tag{6.2}$$

Hence, all of the symbol errors in the received vector can be calculated by successively setting j to take on the values in the index set $J = \{0, 1, 2, \ldots, n-1\}$.

A method is now developed that illustrates erasure decoding of Reed-Solomon codes. The discussion closely follows the one given in Wicker [28, pp. 227-232]. Suppose that a received word contains v errors and f erasures. The errors occur in coordinates $i_1, i_2, \cdots, i_v$, while the erasures occur in coordinates $j_1, j_2, \cdots, j_f$. In the analysis that follows, the coordinates for the errors and erasures are designated using the error locators $X_1 = \alpha^{i_1}, X_2 = \alpha^{i_2}, \cdots, X_v = \alpha^{i_v}$ and the erasures locators $Y_1 = \alpha^{j_1}, Y_2 = \alpha^{j_2}, \cdots, Y_f = \alpha^{j_f}$. Remember that the primary difference between the error locators and the erasure locators is that the values for the latter is obtained using equation (6.2) and is known at the beginning of the

decoding operation. The first of the two tasks for our decoding operation is to determine the values of the error locators. The second task is to find the values $\{e_{i_k}\}$ associated with the error locators and the values $\{f_{j_k}\}$ associated with the erasure locators.

The erasure locator polynomial is computed using the erasure locators as follows:

$$\Gamma(x) = \prod_{l=1}^{f}(1 - Y_l s). \tag{6.3}$$

In order to compute the syndrome for the received word, values must be inserted at every coordinate where an erasure has been indicated. Naturally the computations that follow are much simpler if the value zero is selected for this substitution. Assume that the code is narrow-sense, having as zeros $\alpha, \alpha^2, \cdots, \alpha^{2t}$. Since the syndrome is only a function of the error/erasure polynomial, the syndrome computations have the following form

$$S_l = r(\alpha^l) = \sum_{k=1}^{v} e_{i_k} X_k^l + \sum_{k=1}^{f} f_{j_k} Y_k^l. \tag{6.4}$$

Construct a syndrome polynomial

$$S(x) = \sum_{l=1}^{2t} S_l x^l, \tag{6.5}$$

with which the key equation for errors and erasure decoding is obtained

$$\Lambda(x)\Gamma(x)[1 + S(x)] = \Omega(x) \bmod x^{2t+1}, \tag{6.6}$$

where $\Lambda(x)$ is the error locator polynomial. Equation (6.6) can be slightly simplified by combining all the information that is know at the beginning of the decoding operation into a single modified syndrome polynomial $\Xi(x)$ [31]

$$1 + \Xi(x) \equiv \Gamma(x)[1 + S(x)] \bmod x^{2t+1}. \tag{6.7}$$

The key equation now takes the form

$$\Lambda(x)[1 + \Xi(x)] \equiv \Omega(x) \bmod x^{2t+1} \tag{6.8}$$

which can be solved using the Berlekamp-Massey (BM) algorithm. Once the error locator polynomial is known, combine it with the erasure locator polynomial to obtain a single error/erasure locator polynomial $\Psi(x)$, given by

$$\Psi(x) = \Lambda(x)\Gamma(x). \tag{6.9}$$

A modified version of the Forney algorithm (Theorem 9-2, Wicker [28, pg. 222]) can then be used to compute the error and erasure values, given respectively by

$$e_{i_k} = \frac{-X_k\Omega(X_k^{-1})}{\Psi'(X_k^{-1})}, \quad f_{i_k} = \frac{-Y_k\Omega(Y_k^{-1})}{\Psi'(Y_k^{-1})}, \tag{6.10}$$

where $\Psi^{'}(x)$ is the formal derivative defined below. Finally, an error/erasure polynomial is then constructed and subtracted from the received polynomial to obtain the desired code polynomial.

Definition 6.3 Let $f(x) = f_0 + f_1x + f_2x^2 + \cdots + f_nx^n + \cdots$ be a polynomial with coefficients in $GF(q)$. The formal derivative $f^{'}(x)$ is defined as follows:
$f^{'}(x) = f_1 + 2f_2x + \cdots + nf_nx^{n-1} + \cdots$ □

The Berlekamp-Massy algorithm for erasure decoding is summarized as follows:

1. Compute $\Gamma(x)$ using the erasure information provided in equation (6.2).
2. Replace the erased coordinates with zeros and compute the syndrome polynomial $S(x)$.
3. Compute the modified syndrome polynomial $\Xi(x) \equiv (\Gamma(x)[1 + S(x)] - 1) \bmod x^{2t+1}$.
4. Apply the Berlekamp-Massey algorithm to find the connection polynomial $\Lambda(x)$ for the LFSR that generates the modified syndrome coefficients $\Xi_{f+1}, \Xi_{f+2}, \cdots, \Xi_{2t}$.

5. Find the roots of $\Lambda(x)$ using the Chien Search which determines the error locations.

6. Determine the magnitude of the errors and erasures.

Example 6.2 (Error/Erasure Correction) In the following example, the procedure for error/erasure correction of a (7,3) Reed-Solomon code using the BM Algorithm is demonstrated. A double error correcting narrow sense Reed-Solomon code of length 7 over $GF(8)$ has the generating polynomial

$$\begin{aligned} g(x) &= (x-\alpha)(x-\alpha^2)(x-\alpha^3)(x-\alpha^4) \\ &= x^4 + \alpha^3 x^3 + x^2 + \alpha x + \alpha^3. \end{aligned}$$

Suppose the received polynomial is given by

$$r(x) = \alpha^3 x^2 + f x^3 + x^4 + x^6,$$

where the "f" value indicates an erasure. This erasure occurs at $Y_1 = \alpha^3$, giving the erasure polynomial $\Gamma(x) = 1 + \alpha^3 x$. Now place a zero in the erasure location and compute the syndromes, which are given by

$$\begin{aligned} S_l &= \alpha^3(\alpha^l)^2 + (\alpha^l)^4 + (\alpha^l)^6 \\ \Rightarrow S(x) &= \alpha^2 x + \alpha^2 x^2 + \alpha^6 x^3 + x^4. \end{aligned}$$

The modified syndrome polynomial is easily calculated as follows:

$$\begin{aligned} 1 + \Xi(x) &\equiv \Gamma(x)[1 + S(x)] \bmod x^{2t+1} \\ &\equiv (1 + \alpha^3 x)(1 + \alpha^2 x + \alpha^2 x^2 + \alpha^6 x^3 + x^4) \bmod x^5 \\ &\equiv (1 + \alpha^5 x + \alpha^3 x^2 + \alpha x^3 + \alpha^6 x^4) \bmod x^5, \end{aligned}$$

which reduces to

$$\Xi(x) = \alpha^5 x + \alpha^3 x^2 + \alpha x^3 + \alpha^6 x^4.$$

The error locator polynomial is calculatee using the BM algorithm, given by $\Lambda(x) = 1 + \alpha^5 x$, indicating a single error at $X_1 = \alpha^5$. The error magnitude polynomial and the error/erasure polynomial are obtained from (6.8) and (6.9) respectively, given by

$$\Omega(x) \equiv (1 + \alpha^5 x)(1 + \alpha^5 x + \alpha^3 x^2 + \alpha x^3 + \alpha^6 x^4) \equiv 1 \bmod x^5$$

$$\Psi(x) \equiv (1 + \alpha^5 x)(1 + \alpha^3 x) \equiv 1 + \alpha^2 x + \alpha x^2.$$

The error and erasure magnitudes follow readily, and are given by

$$e_{i_k} = \frac{-X_k \Omega(X_k^{-1})}{\Psi'(X_k^{-1})} = \alpha^3, \quad f_{i_k} = \frac{-Y_k \Omega(Y_k^{-1})}{\Psi'(Y_k^{-1})} = \alpha.$$

Thus, the corrected word is

$$\begin{aligned} c(x) &= r(x) + e(x) + f(x) \\ &= (\alpha^3 x^2 + x^4 + x^6) + \alpha^3 x^5 + \alpha x^3 \\ &= \alpha^3 x^2 + \alpha x^3 + x^4 + \alpha^3 x^5 + x^6 \\ &= x^2 g(x). \end{aligned}$$

Finally, a quick sanity check shows that since $c(x)$ is a factor of $g(x)$, it must be truly be a codeword.

Simulations were conducted on a (255,223) Reed-Solomon code over an additive white Gaussian noise (AWGN) channel to determine the accuracy of the symbol erasure information provided by equation (6.2). Error/erasure decoding was performed using a combination of 15 errors and 2 erasures and compared to error only decoding for 16 errors using the Berlekamp-Massey algorithm. Figure 6.4 illustrates that the performance of the erasure decoder is slightly better than the error only decoder in terms of symbol-error probability. Clearly, erasure decoding does not provide significant additional gain for additive white Gaussian noise channels.

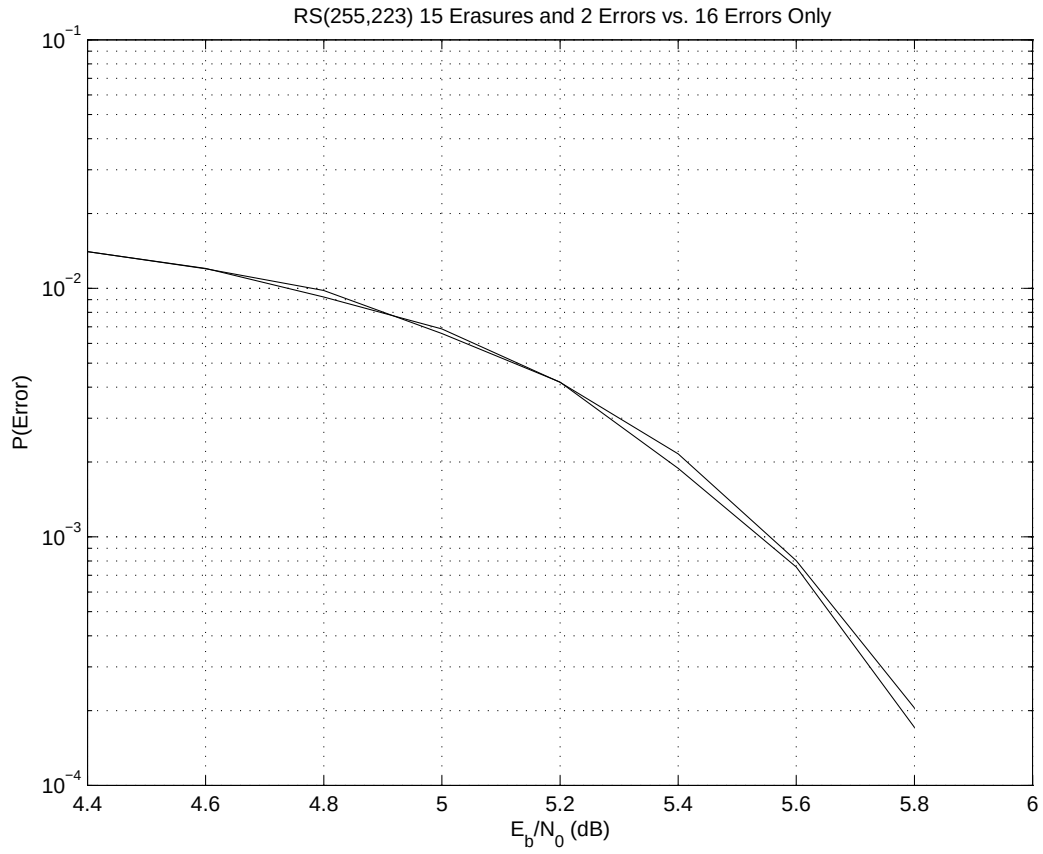

Figure 6.4: Error/Erasure decoding of the RS(255,223) code

Chapter 7

Conclusion

7.1 Synopsis of the Main Results

The important results achieved in this dissertation are summarized as follows:

1. The spectral representation of any WSS stochastic process $x(t)$ can be represented in the form of the following inverse Fourier-Stieltjes integral,

$$x(t) = \int_{-\infty}^{\infty} e^{i2\pi ft} dX(f),$$

where $X(f)$ is a stochastic process of orthogonal increments and where equality holds with probability one.

2. The integrated spectrum of $x(t)$, namely,

$$X(f) = \lim_{T \to \infty} \int_{-T}^{T} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t) dt,$$

is well defined and exists as a limit in the mean-square sense.

3. A perfectly synchronized Costas phase-locked loop has following baseband representation: Signal corrupted with Gaussian noise at the output of the in-phase (I) channel, and Gaussian noise only with no signal at the output of the quadrature (Q) channel.

4. The phase-error variance is derived for the Costas loop when there is a mismatch between the modulation bandwidth and the RF filter bandwidth in the receiver.

5. The Costas circuit is analyzed when perfect phase-lock is not achieved. Analytic expressions are derived that determine the fraction of signal and noise power that leaks from the in-phase channel into the quadrature channel, and the fraction of noise power that leaks from the quadrature channel into the in-phase channel.

6. The individual bit-error probabilities of binary symbols or codewords is derived. It turns out that the bit or symbol error probability of a codeword is a function of the received-bit amplitudes A and the channel noise power σ^2, both of which are assumed to be unknown a-priori at the receiver.

7. A joint estimation is derived for the received-bit amplitudes A and the channel noise power σ^2 using a Costas phase-locked loop receiver.

8. The accuracy of the bit-error probability estimate is analyzed.

9. The reliability-search algorithm is developed to decode the (23,12,7) Golay code. Simulation results show that the reliability search algorithm reduces the decoding time of the Golay code by 41.2% compared to the shift-search algorithm.

10. The reliability-search algorithm is developed to decode the (47,24,11) QR code. Simulation results show that the reliability-search algorithm reduces the decoding time of this code by 22.2% compared to the algebraic decoding algorithm.

11. An estimate is derived for the symbol error probability for nonbinary BCH and Reed-Solomon codes.

12. A method of correcting combinations of errors and erasures is demonstrated. It results in a slightly lower symbol-error probability compared with decoding errors only over an additive white Gaussian noise (AWGN) channel.

7.2 Future Work

There are several areas of future research that can be investigated because of work in this dissertation.

Several quadratic residue codes have not been decoded up to true minimum distance because the decoding complexity is too high. The reliability-search algorithm can be used to cancel one or more errors in the received word, and then a less complex algebraic decoding algorithm can be utilized to decode these codes up to true minimum distance.

The improvement of erasure decoding for Reed-Solomon codes was marginal over and additive white Gaussian noise (AWGN) channel. The reasoning is that the errors do not occur in bursts over an AWGN channel so that the ability of the Reed-Solomon code to correct multiple burst errors is not fully utilized. It will be a good area of future research to try this decoding algorithm over a fading channel where the errors tend to occur in bunches as opposed to random patterns associated with a Bernoulli distributed process which occur on an AWGN channel.

Another interesting research topic would be to concatenate the Reed-Solomon code with the (47,24,11) Quadratic Residue code over various channels. The RS code would server as the outer code correcting the burst errors while the inner code would be the (47,24,11) QR code using the soft information from the receiver to cancel some of the errors.

A final research topic would be to see how the estimate of the individual bit-error probabilities would help decode convolutional codes, and even reduce the complexity of maximum likelihood decoding.

Reference List

[1] E. R. Berlekamp, *Algebraic Coding Theory.* New York: McGraw-Hill, 1968.

[2] I. S. Reed, X. Yin, T. K. Truong, and J. K. Holmes, "Decoding the (24,12,8) Golay code," *IEE Proceedings*, vol. 137, pp. 202–206, May 1990.

[3] R. He, I. S. Reed, T. K. Truong, and X. Chen, "Decoding the (47,24,11) quadratic residue code," *IEEE Transactions on Information Theory*, vol. 47, pp. 1181–1186, March 2001.

[4] E. Wong and B. Hajek, *Stochastic Processes in Engineering Systems.* New York: Springer, 1985.

[5] J. L. Doob, *Stochastic Processes.* New York: Wiley, 1953.

[6] A. M. Yaglom, *An Introduction to the Theory of Stationary Random Functions.* New Jersey: Prentice-Hall, 1962.

[7] J. L. Lawson and G. E. Uhlenbeck, *Threshold Signals.* New York: McGraw-Hill, 1950.

[8] E. J. Kelly, I. S. Reed, and W. L. Root, "The detection of radar echoes in noise," *Journal of the Society for Industrial and Applied Mathematics*, vol. 8, pp. 309–341, June 1960.

[9] S. O. Rice, "Mathematical analysis of random noise," *Bell System Technical Journal*, vol. 23, pp. 282–332, June 1944.

[10] W. B. Davenport and W. L. Root, *An Introduction to the Theory of Random Signals and Noise.* New York: McGraw-Hill, 1958.

[11] A. Papoulis, *Probability, Random Variables, and Stochastic Processes.* New York: McGraw-Hill, 1965.

[12] J. K. Holmes, *Coherent Spread Spectrum Systems.* New York: Wiley, 1982.

[13] M. K. Simon and W. C. Lindsey, "Optimum performance of suppressed carrier receivers with Costas loop tracking." *IEEE Transactions on Communications*, vol. 25, pp. 215–227, February 1977.

[14] S. Haykin, *Communication Systems, Third Edition.* New York: Wiley, 1994.

[15] H. Meyr and G. Ascheid, *Synchronization in Digital Communications, Volume I.* New York: Wiley, 1990.

[16] W. C. Lindsey, *Synchronization Systems in Communication and Control.* New Jersey: Prentice-Hall, 1972.

[17] A. J. Vierbi, *Principles of Coherent Communications.* New York: Wiley, 1966.

[18] W. C. Lindsey and M. K. Simon, *Telecommunication Systems Engineering.* New Jersey: Prentice-Hall, 1973.

[19] A. D. Poularikas and S. Seely, *Signals and Systems, Second Edition.* Florida: Krieger, 1994.

[20] L. W. Couch, *Digital and Analog Communication Systems, Fifth Edition.* New Jersey: Prentice-Hall, 1997.

[21] G. Dubney and I. S. Reed, "Decoding the (23,12,7) Golay code using bit-error probability estimates," in *Proceedings of IEEE Global Telecommunications Conference*, November 2005.

[22] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part 1.* New York: Wiley, 1968.

[23] I. S. Reed, "Statistical error control of a realizable binary symmetric channel." *MIT Press*, vol. 47, pp. 1–12, November 1959.

[24] G. Casella and R. L. Berger, *Statistical Inference, Second Edition.* California: Duxbury, 2002.

[25] T. W. Hugerford, *Algebra.* New York: Springer, 2003.

[26] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes.* Netherlands: Elsevier Science B.V., 1977.

[27] I. S. Reed and X. Chen, *Error-Control Coding for Data Networks.* California: Kluwer Academic Publishers, 1999.

[28] S. Wicker, *Error Control Systems for Digital Communication and Storage.* New Jersey: Prentice Hall, 1995.

[29] W. W. Peterson, "Encoding and error-correction procedures for the bose-chaudhuri codes," *Transactions on Information Theory*, September 1960.

[30] J. L. Massey, "Shift register synthesis and bch decoding," *Transactions on Information Theory*, January 1969.

[31] G. D. Forney, Jr., *Concatenated Codes.* MA: MIT, 1966.

[32] D. J. Sakrison, *Communication Theory: Transmission of Waveforms and Digital Information.* New York: Wiley, 1968.

[33] G. Hedin, J. K. Holmes, W. C. Lindsey, and K. T. Woo, "Theory of false lock in Costas loops," *IEEE Transactions on Communications*, vol. 26, pp. 1–12, January 1978.

[34] C. M. Chie and W. C. Lindsey, "Phase-locked loops: Applications, performance, measures, and summary of analytical results." *IEEE Press*, vol. 26, pp. 1–12, January 1986.

[35] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communication Techniques - Signal Design and Detection*. New Jersey: Prentice-Hall, 1994.

[36] E. J. Kelly and I. S. Reed, "Some properties of stationary Gaussian processes," *MIT Press*, vol. 157, pp. 1–20, June 1957.

[37] M. Elia, "Algebraic decoding the (23,12,7) Golay code," *IEEE Transactions on Information Theory*, vol. 33, pp. 150–151, May 1987.

[38] S. Lin and D. Constello, *Error Control Coding: Fundamental and Applications*. New Jersey: Prentice Hall, 1983.

[39] R. McEliece, *Theory of Information and Coding*. MA: Addison Wesley, 1977.

[40] R. H. Zaragoza-Moreles, *The Art of Error Correcting Coding*. England: Wiley, 2002.

[41] G. Dubney, J. Yang, I. S. Reed, and T. K. Truong, "Fast decoding of the $(47, 24, 12)$ quadratic residue code," *Submitted to IEEE Transactions on Communications*, July 2006.

[42] J. Loyall, J. Ye, R. Shapiro, and et.al, "A case study in applying *QoS* adaptation and model-band design in the design-time optimization of signal analyzer applications," in *Proceedings of IEEE Military Communications Conference*, November 2004.

[43] G. Dubney, C. C. Chui, and I. S. Reed, "I and Q analysis of a Costas phase-locked loop with mismatched filters," in *Proceedings of IEEE Military Communications Conference*, November 2005.

[44] E. Prange, "Some cyclic error-correcting codes with simple decoding algorithms," *Air Force Cambridge Research Center, TN-58-156*, February 1958.

Appendix A

Existence Proof

The purpose of this appendix is to prove that the integrated spectrum of $x(t)$, namely,

$$X(f) = \lim_{T \to \infty} \int_{-T}^{T} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t) dt, \tag{A.1}$$

is well defined and exists as a limit in the mean-square sense (Yaglom [6], pp. 39-40). The proof parallels the one given in Yaglom and provides a thorough analysis of all the mathematical details.

Let X_T, $T > 0$, be a random function defined on $[0, \infty)$. A necessary and sufficient condition for the integrated spectrum in (A.1) to exist in the mean-square sense is the Cauchy convergence criterion (Wong and Hajek [4], pg. 21). It states that a family of random variables $\{X_T\}$ converges to a limiting random variable X in the mean-square sense if and only if

$$\lim_{T, T' \to \infty} \mathbf{E}\{|X_T - X_{T'}|^2\} = 0. \tag{A.2}$$

To proceed with the proof, consider the following family of random variables

$$X_T(f) = \int_{-T}^{T} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t) dt. \tag{A.3}$$

Without loss of generality, assume that $T > T^{'}$. Then a substitution of (A.3) into (A.2) yields the following chain of results:

$$\begin{aligned}
\mathbf{E}\big|X_T(f) - X_{T^{'}}(f)\big|^2 &= \mathbf{E}\left|\int_{-T}^{T} \frac{e^{-i2\pi ft}-1}{-2\pi it}x(t)dt \; - \int_{-T^{'}}^{T^{'}} \frac{e^{-i2\pi ft}-1}{-2\pi it}x(t)dt\right|^2 \\
&= \mathbf{E}\left|\int\limits_{T^{'}<|t|<T} \frac{e^{-i2\pi ft}-1}{-2\pi it}x(t)dt\right|^2 \\
&= \int\limits_{T^{'}<|t|<T}\int\limits_{T^{'}<|s|<T} \frac{e^{-i2\pi ft}-1}{-2\pi it}\frac{e^{i2\pi fs}-1}{2\pi is}\,\mathbf{E}\{x(t)x^*(s)\}dsdt \\
&= \int\limits_{T^{'}<|t|<T}\int\limits_{T^{'}<|s|<T} \frac{e^{-i2\pi ft}-1}{-2\pi it}\frac{e^{i2\pi fs}-1}{2\pi is}R(t-s)dsdt \\
&= \int\limits_{T^{'}<|t|<T}\int\limits_{T^{'}<|s|<T}\int_{-\infty}^{\infty} df^{'}\frac{e^{-i2\pi ft}-1}{-2\pi it}\frac{e^{i2\pi fs}-1}{2\pi is}e^{i2\pi f^{'}(t-s)}G(f^{'})dsdt \\
&= \int_{-\infty}^{\infty}\left|\int\limits_{T^{'}<|t|<T} \frac{e^{-i2\pi ft}-1}{-2\pi it}e^{if^{'}t}dt\right|^2 G(f^{'})df^{'}. \qquad \text{(A.4)}
\end{aligned}$$

A use of the inequality $|x \pm y|^2 \leq 2|x|^2 + 2|y|^2$ enables the inner integral in (A.4) to be upper bounded as follows:

$$\begin{aligned}
\left|\int\limits_{T^{'}<|t|<T} \frac{e^{-i2\pi ft}-1}{-2\pi it}e^{if^{'}t}dt\right|^2 &= \left|\int_{-T}^{T}\frac{e^{-i2\pi ft}-1}{-2\pi it}e^{if^{'}t}dt - \int_{-T^{'}}^{T^{'}}\frac{e^{-i2\pi ft}-1}{-2\pi it}e^{if^{'}t}dt\right|^2 \\
&= \big|[\Psi_T(f^{'},f) - \Psi(f^{'},f)] + [\Psi(f^{'},f) - \Psi_{T^{'}}(f^{'},f)]\big|^2 \\
&\leq 2\big|\Psi_T(f^{'},f) - \Psi(f^{'},f)\big|^2 + 2\big|\Psi(f^{'},f) - \Psi_{T^{'}}(f^{'},f)\big|^2, \qquad \text{(A.5)}
\end{aligned}$$

where $\Psi_T(f^{'},f)$ and $\Psi(f^{'},f)$ are defined, respectively, by

$$\Psi_T(f^{'},f) = \int_{-T}^{T}\frac{e^{-i2\pi ft}-1}{-2\pi it}e^{i\,f^{'}t}dt \qquad \text{(A.6a)}$$

$$\Psi(f^{'},f) = \lim_{T\to\infty}\int_{-T}^{T}\frac{e^{-i2\pi ft}-1}{-2\pi it}\,e^{if^{'}t}dt. \qquad \text{(A.6b)}$$

Then, according to (A.4), the proof of existence of the limit in (A.1) is a consequence of the existence of the limit

$$\begin{aligned}
\Psi(f^{'}, f) &= \lim_{T\to\infty} \int_{-T}^{T} \frac{e^{-i2\pi ft} - 1}{-2\pi it} e^{if^{'}t} dt \\
&= \lim_{T\to\infty} \frac{1}{2\pi} \int_{-T}^{T} \frac{e^{i2\pi f^{'}t} - e^{i2\pi(f^{'}-f)t}}{it} dt \\
&= \lim_{T\to\infty} \frac{1}{2\pi} \int_{-T}^{T} \frac{\cos(2\pi f^{'}t)}{t} dt + \lim_{n\to\infty} \frac{1}{2\pi} \int_{-T}^{T} \frac{\sin(2\pi f^{'}t)}{t} dt \\
&- \lim_{T\to\infty} \frac{1}{2\pi} \int_{-T}^{T} \frac{\cos[2\pi(f^{'} - f)t]}{t} dt - \lim_{n\to\infty} \frac{1}{2\pi} \int_{-T}^{T} \frac{\sin[2\pi(f^{'} - f)t]}{t} dt \\
&= \frac{1}{\pi} \int_{0}^{\infty} \frac{\sin(2\pi f^{'}t)}{t} dt - \frac{1}{\pi} \int_{0}^{\infty} \frac{\sin[2\pi(f^{'} - f)t]}{t} dt.
\end{aligned} \tag{A.7}$$

However, it is well known that

$$\frac{1}{\pi} \int_{0}^{\infty} \frac{\sin(ft)}{t} dt = \begin{cases} +1/2, & f > 0 \\ 0, & f = 0 \\ -1/2, & f < 0 \ , \end{cases}$$

from which it follows that $\Psi(f^{'}, f)$ not only exists but also has the finite value, given by

$$\Psi(f^{'}, f) = \begin{cases} 1, \text{ for } 0 < f^{'} < f \\ 1/2, \text{ for } f^{'} = f > 0, \text{ or } f^{'} = 0, \ f > 0 \\ 0, \text{ for } f^{'} > 0, \ f^{'} > f \text{ or } f^{'} = f = 0 \text{ or } f^{'} < 0, f^{'} < f \\ -1/2, \text{ for } f^{'} = f < 0, \text{ or } f^{'} = 0, \ f < 0 \\ -1, \text{ for } f < f^{'} < 0. \end{cases} \tag{A.8}$$

Thus, a substitution of (A.5) into (A.4) yields the following upper bound

$$0 \leq \mathbf{E}\big|X_T(f) - X_{T'}(f)\big|^2 \leq \int_{-\infty}^{\infty} 2\big|\Psi_T(f^{'}, f) - \Psi(f^{'}, f)\big|^2 G(f^{'})df^{'} + \int_{-\infty}^{\infty} 2\big|\Psi_{T'}(f^{'}, f) - \Psi(f^{'}, f)\big|^2 G(f^{'})df^{'}. \tag{A.9}$$

Consider the first integral in (A.9), namely,

$$\int_{-\infty}^{\infty} 2\big|\Psi_T(f^{'}, f) - \Psi(f^{'}, f)\big|^2 G(f^{'})df^{'}. \tag{A.10}$$

Following the same techniques used to obtain (A.7), $\Psi_T(f^{'}, f)$ can be expressed as follows:

$$\Psi_T(f^{'}, f) = \frac{1}{\pi}\int_0^T \frac{\sin(2\pi f^{'}t)}{t}dt - \frac{1}{\pi}\int_0^T \frac{\sin[2\pi(f^{'} - f)t]}{t}dt. \tag{A.11}$$

Next a substitution of $\Psi(f^{'}, f)$ and $\Psi_T(f^{'}, f)$, given in (A.7) and (A.11), respectively, into $2\big|\Psi_T(f^{'}, f) - \Psi(f^{'}, f)\big|^2$ yields the following upper bound:

$$\begin{aligned} 2\big|\Psi_T(f^{'}, f) - \Psi(f^{'}, f)\big|^2 &= \frac{2}{\pi}\left|\int_T^{\infty} \frac{\sin(2\pi f^{'}t)}{t}dt - \int_T^{\infty} \frac{\sin[2\pi(f^{'} - f)t]}{t}dt\right|^2 \\ &\leq \frac{4}{\pi}\left|\int_T^{\infty} \frac{\sin(2\pi f^{'}t)}{t}dt\right|^2 + \frac{4}{\pi}\left|\int_T^{\infty} \frac{\sin[2\pi(f^{'} - f)t]}{t}dt\right|^2 \\ &= \frac{4}{\pi}\left|\int_{2\pi f^{'}T}^{\infty} \frac{\sin(s)}{s}ds\right|^2 + \frac{4}{\pi}\left|\int_{2\pi(f^{'}-f)T}^{\infty} \frac{\sin(w)}{w}dw\right|^2 \\ &= L(f^{'}, T) + L(f^{'} - f, T), \end{aligned} \tag{A.12}$$

where $L(r, T)$ is defined by

$$L(r, T) = \frac{4}{\pi}\left|\int_{2\pi rT}^{\infty} \frac{\sin(s)}{s}ds\right|^2. \tag{A.13}$$

Hence, a substitution of (A.12) into (A.10) yields

$$\begin{aligned} 0 &\leq \int_{-\infty}^{\infty} 2\big|\Psi_T(f^{'},f) - \Psi(f^{'},f)\big|^2 G(f^{'})df^{'} \\ &\leq \int_{-\infty}^{\infty} [L(f^{'},T) - L(f^{'}-f,T)]G(f^{'})df^{'} \\ &= \int_{-\infty}^{\infty} L(f^{'},T)G(f^{'})df^{'} - \int_{-\infty}^{\infty} L(f^{'}-f,T)G(f^{'})df^{'}. \end{aligned} \tag{A.14}$$

In order to finish the proof, it still remains to show that the integrals in (A.14) approach zero as $T \to \infty$. Let $\delta > 0$. The first integral can be upper bounded as follows:

$$\begin{aligned} \int_{-\infty}^{\infty} L(f^{'},T)G(f^{'})df^{'} &\leq \int_{-\delta}^{\delta} L(f^{'},T)G(f^{'})df^{'} + \int\limits_{|f^{'}|\geq\delta} L(|f^{'}|,T)G(f^{'})df^{'} \\ &\leq C \cdot 2\delta + M \cdot \sup_{|f^{'}|\geq\delta} L(|f^{'}|,T), \end{aligned} \tag{A.15}$$

where C and M are defined, respectively, by

$$C = \sup_{f^{'}} [L(|f^{'}|,T)G(f^{'})] \tag{A.16a}$$

$$M = \int_{-\infty}^{\infty} G(f^{'})df^{'} < \infty. \tag{A.16b}$$

Since for each $\delta > 0$,

$$\overline{\lim_{T\to\infty}} \sup_{|r|\geq\delta} L(|r|,T) \to 0, \tag{A.17}$$

(see Appendix B), taking the upper limit of (A.15) as $T \to \infty$ yields

$$\overline{\lim_{T\to\infty}} \int_{-\infty}^{\infty} L(f^{'},T)G(f^{'})df^{'} \leq C \cdot 2\delta. \tag{A.18}$$

Since $\delta > 0$ is arbitrary, one has that

$$\overline{\lim_{T\to\infty}} \int_{-\infty}^{\infty} L(f^{'},T)G(f^{'})df^{'} = 0. \tag{A.19}$$

The second integral in (A.14) can be upper bounded as follows:

$$\begin{aligned}\int_{-\infty}^{\infty} L(f^{'}-f,T)G(f^{'})df^{'} &\leq \int\limits_{|f^{'}-f|\leq\delta} L(|f^{'}-f|,T)G(f^{'})df^{'} + \int\limits_{|f^{'}-f|\geq\delta} L(|f^{'}-f|,T)G(f^{'})df^{'} \\ &\leq C_1 \cdot 2\delta + \int\limits_{|f^{'}-f|\geq\delta} L(|f^{'}-f|,T)G(f^{'})df^{'},\end{aligned} \tag{A.20}$$

where C_1 is defined by

$$C_1 = \sup_{f^{'}} [L(|f^{'}-f|,T)G(f^{'})]. \tag{A.21}$$

Next take the supremum of (A.20) over f to yield

$$\begin{aligned}\sup_{f}\int_{-\infty}^{\infty} L(f^{'}-f,T)G(f^{'})df^{'} &\leq C_1 \cdot 2\delta + \sup_{f}\int\limits_{|f^{'}-f|\geq\delta_1} L(|f^{'}-f|,T)G(f^{'})df^{'} \\ &\leq C_1 \cdot 2\delta + \sup_{f}\int\limits_{|f^{'}-f|\geq\delta}\sup_{|f^{'}-f|\geq\delta} L(|f^{'}-f|,T)G(f^{'})df^{'} \\ &\leq C_1 \cdot 2\delta + M \cdot \sup_{|f^{'}-f|\geq\delta} L(|f^{'}-f|,T)G(f^{'})df^{'},\end{aligned} \tag{A.22}$$

where M is defined in (A.16b). Since $\delta > 0$ is arbitrary, taking the upper limit of (A.22) as $T \to \infty$ and utilizing equation (A.17) yields

$$\begin{aligned}\overline{\lim_{T\to\infty}}\sup_{f}\int_{-\infty}^{\infty} L(f^{'}-f,T)G(f^{'})df^{'} &\leq C_1 \cdot 2\delta + M \cdot \overline{\lim_{T\to\infty}}\sup_{|f^{'}-f|\geq\delta} L(f^{'}-f,T) \\ &= C \cdot 2\delta \\ &\to 0.\end{aligned} \tag{A.23}$$

Thus, (A.18) and (A.23) together imply that

$$\lim_{T\to\infty}\int_{-\infty}^{\infty} 2\left|\Psi_T(f^{'},f) - \Psi(f^{'},f)\right|^2 G(f^{'})df^{'} = 0. \tag{A.24}$$

Similarly, it can be shown that the second integral in (A.9) is also zero, i.e.,

$$\lim_{T' \to \infty} \int_{-\infty}^{\infty} 2\left|\Psi_{T'}(f', f) - \Psi(f', f)\right|^2 G(f')df' = 0. \tag{A.25}$$

Therefore, equations (A.24) and (A.25) together prove that

$$\lim_{T, T' \to \infty} \mathbf{E}\{|X_T - X_{T'}|^2\} = 0,$$

which shows that the integrated spectrum of $x(t)$, namely,

$$X(f) = \lim_{T \to \infty} \int_{-T}^{T} \frac{e^{-i2\pi ft} - 1}{-2\pi it} x(t)dt,$$

is well defined and exists as a uniform limit of $X_T(f)$ in the mean-square sense.

Appendix B

Proof of Uniform Convergence

The goal of this appendix is to prove that for any $\delta > 0$

$$\overline{\lim_{T\to\infty}} \sup_{|r|\geq\delta} L(|r|, T) \to 0, \tag{B.1}$$

where

$$L(r, T) = \frac{4}{\pi}\left|\int_{2\pi rT}^{\infty} \frac{\sin(s)}{s} ds\right|^2. \tag{B.2}$$

Thus, it must be shown that for every $\varepsilon > 0$ there exists M_ε such that for all $|r| \geq \delta$,

$$T > M_\varepsilon \Rightarrow \left| \int_{2\pi rT}^{\infty} \frac{\sin(s)}{s} ds \right| < \varepsilon. \tag{B.3}$$

To proceed with the proof, it is well known that

$$\lim_{T\to\infty} \int_0^T \frac{\sin(t)}{t} dt = \pi/2. \tag{B.4}$$

This implies that the tail of the integral converges to zero, i.e.,

$$\lim_{T\to\infty} \int_T^{\infty} \frac{\sin(t)}{t} dt \to 0. \tag{B.5}$$

Hence, by definition of convergence, one has that for every $\varepsilon > 0$ there exists N_ε such that

$$T > N_\varepsilon \Rightarrow \left| \int_T^\infty \frac{\sin(t)}{t} dt \right| < \varepsilon. \tag{B.6}$$

Let $|r| \geq \delta$ and choose $M_\varepsilon = N_\varepsilon / 2\pi\delta$. If $T > M_\varepsilon$, then $2\pi r M_\varepsilon \delta > 2\pi M_\varepsilon \delta = N_\varepsilon$. Thus, from (B.6) one has that

$$\left| \int_{2\pi r T}^\infty \frac{\sin(s)}{s} ds \right| < \varepsilon. \tag{B.7}$$

Appendix C

Convolution of $S_{n_x}(f)$ and $S_{n_y}(f)$

The purpose of this Appendix is to derive in detail the convolution of the noise power spectral densities $S_{n_x}(f)$ and $S_{n_y}(f)$ given in equation (3.54). To proceed with the analysis, recall that the power spectral density of the in-phase and quadrature noise processes are given in equation (3.9) as

$$S_{n_x}(f) = S_{n_y}(f) = \begin{cases} N_0, & \frac{-1}{2T} < f < \frac{1}{2T} \\ 0, & |f| > \frac{1}{2T}. \end{cases} \tag{C.1}$$

The convolution of $S_{n_x}(f)$ and $S_{n_y}(f)$ in the frequency domain is computed as follows:

$$S_{n_x}(f) * S_{n_y}(f) = \begin{cases} 0, & f + \frac{1}{2T} < -\frac{1}{2T} \\ \int_{-\frac{1}{2T}}^{f+\frac{1}{2T}} N_0^2 df, & -\frac{1}{2T} < f + \frac{1}{2T} < \frac{1}{2T} \\ \int_{f-\frac{1}{2T}}^{\frac{1}{2T}} N_0^2 df, & -\frac{1}{2T} < f - \frac{1}{2T} < \frac{1}{2T}. \end{cases} \tag{C.2}$$

After performing the integration, (C.2) reduces to

$$S_{n_x}(f) * S_{n_y}(f) = \begin{cases} 0, & f < -\frac{1}{T} \\ N_0^2(f + \frac{1}{T}), & -\frac{1}{T} < f < 0 \\ N_0^2(\frac{1}{T} - f), & 0 < f < \frac{1}{T}. \end{cases} \tag{C.3}$$

Now (C.3) can be written more compactly in terms of the triangular function to yield the final result as

$$S_{n_x}(f) * S_{n_y}(f) = \begin{cases} \frac{N_0^2}{T}\left(1 - \frac{|f|}{\frac{1}{T}}\right), & |f| < \frac{1}{T} \\ 0, & \text{otherwise.} \end{cases} \tag{C.4}$$

Appendix D

Power Spectral Density $S_m(f)$

The purpose of this appendix is to derive in detail the power spectral density of the modulation $S_m(f)$ given in equation (3.53).

By definition, the power spectral density (PSD) of a WSS random process $m(t)$ is given by

$$S_m(f) = \lim_{T\to\infty}\left(\frac{\mathbf{E}\{|M_T(f)|^2\}}{T}\right), \tag{D.1}$$

where

$$M_T(f) = \int_{-T/2}^{T/2} m(t)e^{-i2\pi ft}dt.$$

The random data sequence $m(t)$ can be modeled by the following WSS process

$$m(t) = \sum_{n=-\infty}^{\infty} a_n p(t - nT_b),$$

where T_b is the pulse width in time, $p(t)$ has a unit amplitude rectangular pulse shape defined by

$$p(t) = \begin{cases} 1, & |t| \le \frac{T_b}{2} \\ 0, & \text{otherwise,} \end{cases}$$

and $\{a_n\}$ is an independent and identically distributed (i.i.d.) sequence taking on the assumed equiprobable values of ± 1. These rectangular data pulses are assumed to be

equal width, non-overlapping, and have zero separation between them with a modulation bandwidth, given by $B_m = 1/T_b$ Hz.

Now consider the following truncated version of $m(t)$, given by

$$m_T(t) = \sum_{n=-N}^{N} a_n p(t - nT_b), \tag{D.2}$$

where the duration of $m_T(t)$ is

$$T = (2N + 1)T_b.$$

The Fourier transform of the modulation sequence, given in (D.2), is calculated as follows:

$$\begin{aligned} M_T(f) = \mathcal{F}\{m_T(t)\} &= \sum_{n=-N}^{N} a_n \mathcal{F}\{p(t - nT_b)\} \\ &= \sum_{n=-N}^{N} a_n P(f) e^{-i2\pi f n T_b} \\ &= P(f) \sum_{n=-N}^{N} a_n e^{-i2\pi f n T_b}, \end{aligned} \tag{D.3}$$

where $P(f)$ is the Fourier transform of rectangular pulse $p(t)$. A substitution of (D.3) into (D.1) yields the PSD as

$$\begin{aligned} S_m(f) &= \lim_{T\to\infty} \left(\frac{1}{T} |P(f)|^2 \mathbf{E}\left\{ \left| \sum_{n=-N}^{N} a_n e^{-i2\pi f n T_b} \right|^2 \right\} \right) \\ &= |P(f)|^2 \lim_{T\to\infty} \left(\frac{1}{T} \sum_{n=-N}^{N} \sum_{m=-N}^{N} \mathbf{E}\{a_n a_m\} e^{-i2\pi f (n-m) T_b} \right). \end{aligned}$$

The PSD can be further simplified by evaluating $\mathbf{E}\{a_n a_m\}$ for the special case of binary antipodal signals ($a_n = \pm 1$). Clearly, $\mathbf{E}\{a_n a_m\}$ is given by

$$\mathbf{E}\{a_n a_m\} = \begin{cases} \mathbf{E}\{a_n^2\}, & n = m \\ \mathbf{E}\{a_n a_m\} & n \neq m, \end{cases} \tag{D.4}$$

where $\mathbf{E}\{a_n a_m\} = \mathbf{E}\{a_n\}\mathbf{E}\{a_m\}$ for $n \neq m$ since a_n and a_m are independent. Since a_n follows a binomial distribution, one has that

$$\mathbf{E}\{a_n\} = \mathbf{E}\{a_m\} = (+1)\frac{1}{2} + (-1)\frac{1}{2} = 0, \tag{D.5}$$

and

$$\mathbf{E}\{a_n^2\} = (+1)^2\frac{1}{2} + (-1)^2\frac{1}{2} = 1. \tag{D.6}$$

A substitution of (D.5) and (D.6) into (D.4) yields

$$\mathbf{E}\{a_n a_m\} = \begin{cases} 1, & n = m \\ 0 & n \neq m. \end{cases}$$

With this result, the power spectral density becomes

$$S_m(f) = |P(f)|^2 \lim_{T\to\infty}\left(\frac{1}{T}\sum_{n=-N}^{N} 1\right). \tag{D.7}$$

Finally, a substitution of $T = (2N+1)T_b$ into (D.7) yields the power spectral density as

$$\begin{aligned} S_m(f) &= |P(f)|^2 \lim_{N\to\infty}\left[\frac{(2N+1)}{(2N+1)T_b}\right] \\ &= \frac{1}{T_b}|P(f)|^2. \end{aligned} \tag{D.8}$$

Now it is well known that the Fourier transform of a rectangular pulse shape is given by

$$P(f) = T_b \,\mathrm{sinc}(T_b f). \tag{D.9}$$

Thus, a substitution of (D.9) into (D.8) yields the power spectral density for binary antipodal signaling with rectangular pulse shapes as

$$S_m(f) = T_b \,\mathrm{sinc}^2(T_b f).$$

Appendix E

Speech By Gia Dubney

This speech was given by my wife, Gia, at my graduation party on May 14, 2005 held at the Daily Grill Restaurant in El Segundo, California.

When you hit a bump, drive over it. That is the saying that I have been living by for many years. So, when my husband told me he wanted to firm up his decision to go back to school at USC to pursue his Ph.D. degree in electrical engineering, I thought of it as a small hurdle. Boy, I didn't even know the meaning of hurdle! Ever since I met Greg, he has talked about his Ph.D. We met, once again, at our ten year high school reunion. After a year and a half of courtship, I made a bold move and left my comfortable home, my supporting family, my many friends, and my job to go to Los Angeles. Things worked out. We married, had our first born son, Alek, settled down, and started to build roots here in Los Angeles. I won't kid you or myself by telling you that it was easy, smooth sailing, or even only a bit rocky. Greg had always made it very clear that his Ph.D. program was the most important thing in his life. I knew that I played second fiddle to his Ph.D. But something changed when Alek was born. Though his Ph.D. was still very important, I could see the metamorphosis happening. Greg had become a truly loving husband and father. Greg and I went through so many turbulent times, both financially and emotionally, and we sailed the rough waters in life with everyday hurdles. The really amazing thing is that no matter what, Greg and Gia always tried their best and succeeded.

When I first moved to Los Angeles, I knew nobody. I had left a city where I had so much loving family, a million friends, and a comfortable lifestyle, to pursue a man who I believed was a God send. Through the years, I have met such wonderful friends here. Friends I will have for a whole lifetime. It is these friends who have helped and supported me through the lonely nights, the lonely weekends, and the lonely days. The play dates, the girl's lunches, the birthday parties, and the weekend outings have kept me sane in the insane academic world that my husband lived in. To say that I have survived five years of a Ph.D. program is nothing in comparison to the five years that Greg has survived. Greg's knowledge is so dumbfounding to me. I can still remember when Greg moved our family back to the "hood" when he needed to study for his Ph.D. Screening Exam. Even through the drug deals and late night gun shots, I still believed in Greg and supported his decision to get his Ph.D. I never even imagined that a person could study as much as he did. For his Ph.D. Screening Exam, he literally studied twelve hours a day, six days a week for six months. His determination and perseverance was second to none. He successfully passed his screening exam and overcame that hurdle. However, with Dr. Reed as his advisor, there was no time to rest. Greg then started on his research. And continued on his research, and continued, and continued. In fact, after our second son, Zachary, was born, I told him "You better finish your research and get a job." He did finish and just published three papers with two more papers on the way. So, here we stand today after the screening exam, the qualifying exam, and the final defense. My husband has earned his Ph.D. in electrical engineering from Professor Irving Reed, a world renowned scientist and engineer who changed the world we live in with his work. So, please join me and raise your glass to congratulate Dr. Gregory Dubney. To say that I am proud of him doesn't fully capture my feelings. I guess the only way to really express my feelings is to say this, "Greg, I love you. I support you. I will always be by your side." Congratulations Greg!