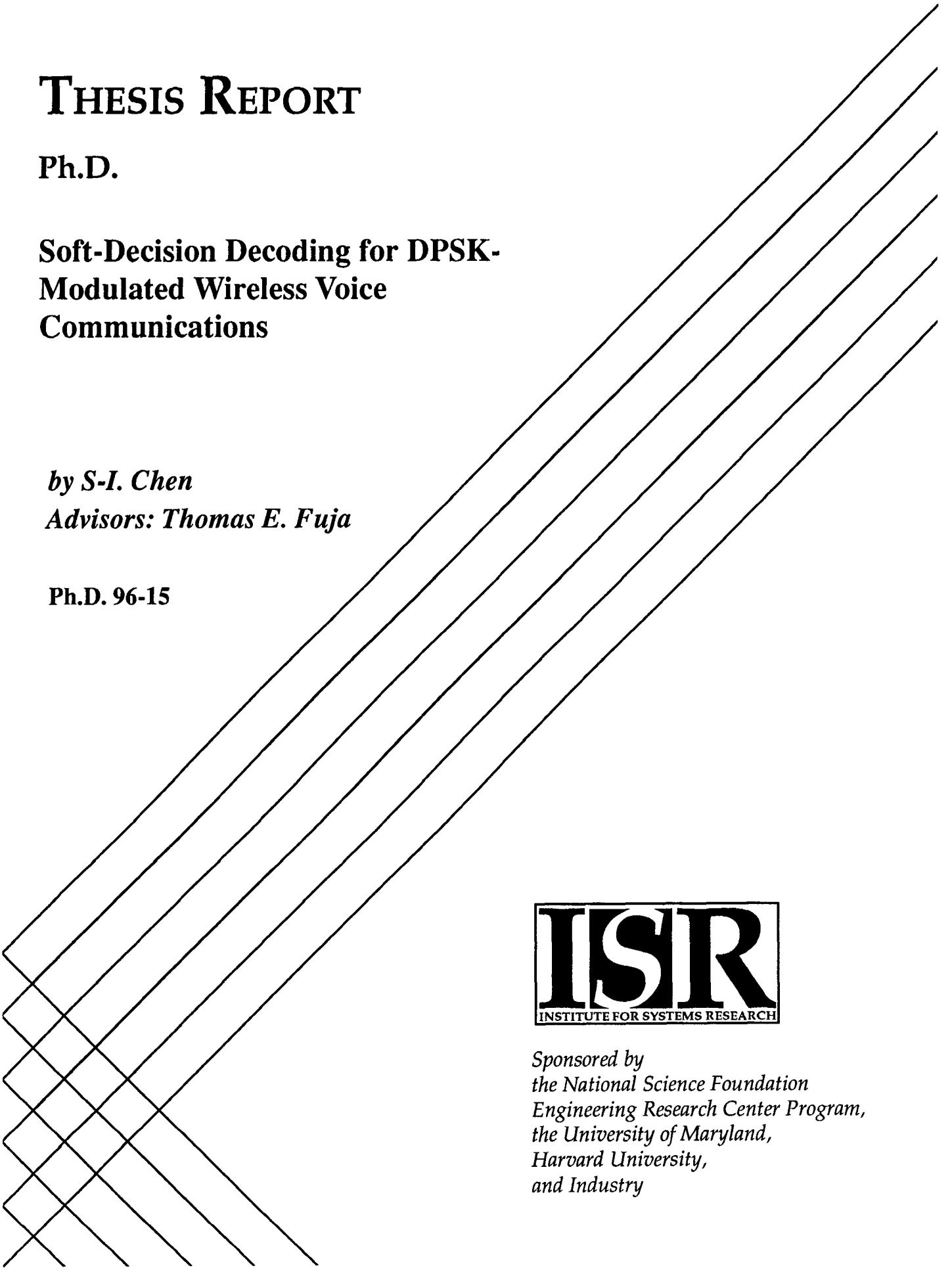
THESIS REPORT

Ph.D.

Soft-Decision Decoding for DPSK-Modulated Wireless Voice Communications

by S-I. Chen
Advisors: Thomas E. Fuja

Ph.D. 96-15



Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry

Abstract

Title of Dissertation: SOFT-DECISION DECODING FOR DPSK-MODULATED WIRELESS VOICE COMMUNICATIONS

Shih-I Chen, Doctor of Philosophy, 1996

Dissertation directed by: Associate Professor Thomas E. Fuja
Department of Electrical Engineering

This thesis addresses some techniques that enhance a receiver's performance in a wireless voice communication system where differential phase shift keying (DPSK) is the adopted modulation scheme and soft-decision decoding is used to improve the effectiveness of the channel coding scheme.

First, several fundamental issues regarding the statistical properties of fading channels are provided. We demonstrate the constraints that must be satisfied so that the channel can be regarded as impaired by "flat" (i.e., non-frequency-selective) fading with a constant fading factor over each symbol duration. Throughout this thesis these constraints are assumed to be satisfied.

We next investigate the channel capacity and cutoff rates for fading channels with DPSK-modulated input signals and perfect symbol interleaving. The impact of the channel state information (CSI) on these information-theoretic limits is also discussed. We introduce several symbol metrics for soft-decision decoding, and their performance is investigated by analytical derivation as well

as by simulation. Furthermore, we define a bit metric for DQPSK modulation, and compare this bit metric to dibit (symbol) metrics.

We then consider the problem of error concealment for mobile radio communications with a maximum-likelihood soft-decision decoder. A normalized codeword reliability is defined as the decision reliability information when CSI is not available. We employ a given interpolation algorithm on a particular land mobile radio system and design a rule for selecting unreliable codewords. Simulation results show that error concealment can decrease the minimum operational signal-noise ratio (SNR) by 3 dB or more.

Finally, we address the problem of exploiting the residual redundancy in the source to enhance the channel decoder's performance -i.e., maximum a posteriori (MAP) decoding. We use two simple source models to demonstrate that MAP decoding can achieve significant gain over maximum-likelihood decoding. We employ a practical CELP-based land mobile radio system to show that significant residual redundancy does exist in the output of some source encoders. Simulation results show that a 2 - 3 dB gain can be achieved by MAP decoding (over ML) at low SNR.

SOFT-DECISION DECODING FOR DPSK-MODULATED WIRELESS VOICE COMMUNICATIONS

by

Shih-I Chen

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland at College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1996

Advisory Committee:

Associate Professor Thomas E. Fuja, Chairman/Advisor
Professor Nariman Farvardin
Associate Professor Adrianos Papamarcou
Associate Professor Steven Tretter
Associate Professor Tzong-Yow Lee

Dedication

To my parents

Acknowledgements

My first gratitude goes to my thesis advisor, Professor Thomas Fuja, for his guidance, continuing support and encouragement, and constructive suggestion. His deep insight and broad knowledge make every discussion with him very valuable and beneficial. I also thank all other members of my thesis advisory committee for their comments and suggestions: Dr. Nariman Farvardin, Dr. Adrianos Papamarcou, Dr. Steve Tretter, and Dr. Tzong-Yow Lee.

I would like to thank the National Security Agency (NSA) in U.S. Department of Defense for sponsoring a two-year project in which I participated. Many topics in this thesis originate from the research for that project.

My gratitude also goes to my colleagues and friends at UMCP for their kindly help. I am especially indebted to Ms. Gerry Mathews for her effort to help me edit this thesis.

Finally come my deepest thanks to my parents and all other members in my family for their constant love and support throughout these years.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

With the rapidly growing demand for cellular mobile radio and other wireless transmission services, there has been an increasing interest in exploiting various technologies to provide reliable, secure, and efficient wireless communications. Unlike wireline transmission, signals transmitted over wireless channels may undergo both long term and short term fading due to shadowing and multipath propagation. As a result, signals received from a wireless channel are subject to high error rates, and so the required operational signal to noise ratio (SNR) range for a wireless system is usually much higher than those of wireline systems. Even at a high average SNR, a wireless system may still encounter intermittent deep fading, causing a significant performance degradation.

In order to mitigate the high-error-rate nature of wireless channels, wireless systems must employ error detection/correcting techniques for better performance. There are many issues associated with channel coding techniques, e.g., code design, encoding functions, decoding algorithms, and coded modulation, etc. Figure 1.1 depicts an overview of various techniques which can be utilized to enhance the overall system performance [1]. The source coder compresses the input signal to allow for more efficient representation of the signal. The channel

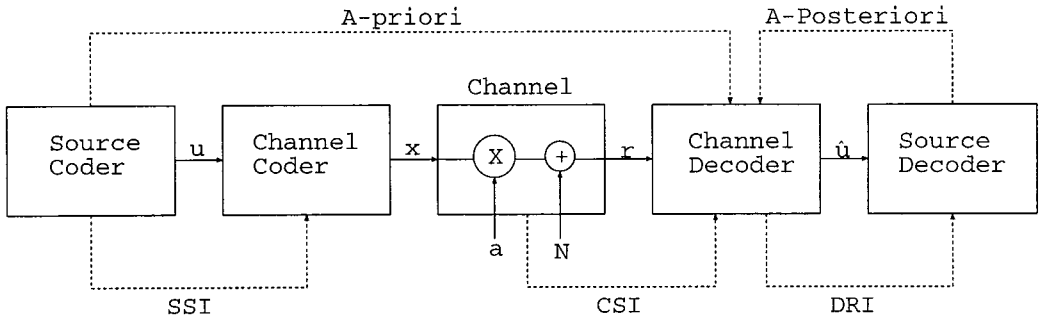


Figure 1.1: An overview of source-channel coding

encoder adds redundancy to protect the compressed signal from transmission errors. The SSI (source significance information) can be passed to the channel coder for (dynamic or static) unequal error protection, and obtain significant gain (e.g., [2]). The channel state information (CSI) is usually utilized in soft-decision decoding algorithms for better results. In some systems (e.g., QAM systems), knowledge of the CSI is very critical for the receiver to function properly. The decision reliability information (DRI) tells the source decoder how reliable the estimated information is. When the decision reliability is low, the source decoder may take appropriate steps to conceal errors. The a priori information of the source gives the channel decoder the opportunity to exploit the residual redundancy left in the source so that the decoding errors may be reduced - e.g., by maximum a posteriori (MAP) decoding. The a posteriori information from the previous and current estimated source symbols may also help to estimate the future source symbols more accurately. This thesis focuses on the decoding side - in particular, CSI and soft-decision decoding, DRI and error concealment, and the use of a priori statistics of the source.

Since soft-decision decoding significantly outperforms hard-decision decod-

ing [3, 4], it is preferable for use in wireless systems. In a soft-decision decoding algorithm, a **metric** must be defined and the metric is closely related to the modulation schemes. This thesis focuses on one of the most popular modulation schemes in wireless systems – differential phase shift keying (DPSK) [5]. For example, the IS-54 standard for North American TDMA cellular mobile radio communications adopts a quaternary DPSK (DQPSK) as the modulation technique. One of the primary reasons for using differential modulation is the difficulty of tracking the received carrier phase accurately in many wireless systems.

In order to perform soft-decision decoding, a decoding metric must be defined and calculated from received signals. For DPSK, a few decoding metrics and their performance have been discussed in the literature [30, 31, 41, 50, 56]. However, they are based largely on applying decision variables in heuristic approaches; this thesis investigate DPSK metrics and their performance in a more rigorous and comprehensive manner.

With regard to information transported over wireless channels, voice may be the most popular media type. Since the decoder may make decoding errors, it is common that the receiver uses a CRC (Cyclic Redundancy Code) or other means to detect the errors and that interpolation is employed to conceal the errors. More sophisticated (and better) error detection can be achieved by using the "soft output" (which is a form of DRI) from the soft-decision decoder. This is a "value-added" approach, and can be applied to systems with or without a "frame quality indicator" (e.g., CRC bits). This thesis investigates an error concealment scheme using a form of soft output from the soft-decision decoder under a quaternary DPSK (DQPSK) modulation for a land mobile radio system.

Since most vocoder output contains significant redundancy, we also investigate the problem of utilizing the source redundancy (or source a priori information) to further enhance the quality of estimated (decoded) signal output.

1.1 Basic System Diagram

The communication system shown in Figure 1.2 provides the basis for the following discussion. On the transmitter side, the vocoder encodes (compresses)

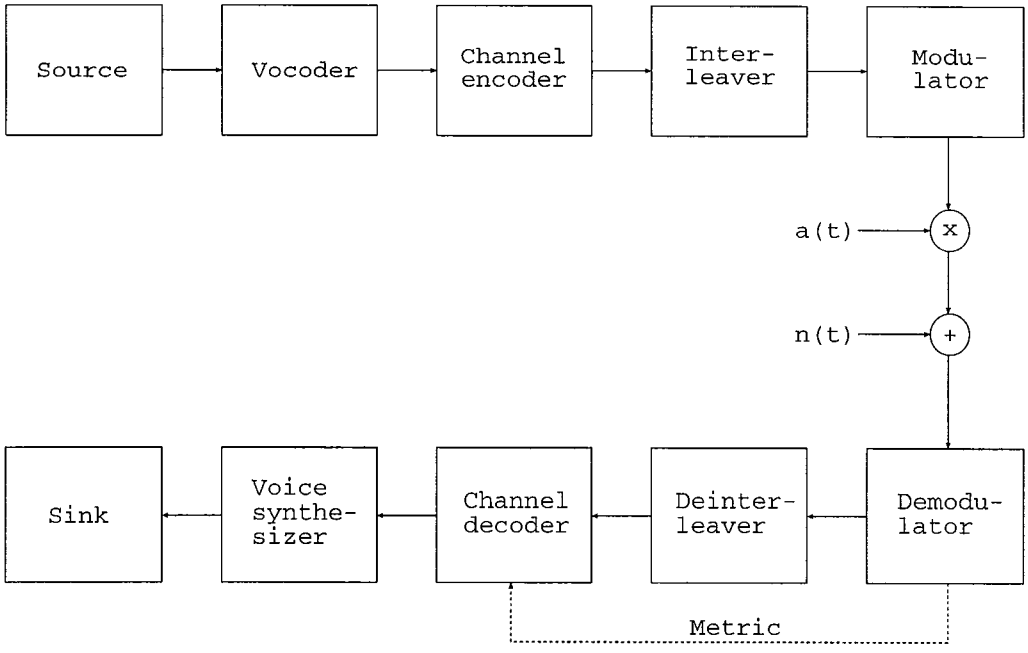


Figure 1.2: Basic system diagram

the voice source into an information bit stream; the channel encoder adds redundancy for error detection and/or correction; the interleaver rearranges the codeword bit streams for transmission in order to avoid bursty errors in a single codeword; and the modulator transforms the resulting bits into analog signals suitable for transmission over the channel. The channel incurs a multiplica-

tive fading factor $a(t)$ and additive white Gaussian noise (AWGN) $n(t)$ on the transmitted signal $s(t)$. The received signal $r(t)$ is therefore represented as

$$r(t) = a(t) \cdot s(t) + n(t).$$

This multiplicative fading model depicts flat fading channels, and will receive further elaboration in the next chapter. On the receiver side, inverse operations are performed. Furthermore, the demodulator extracts metrics for soft-decision decoding.

Besides the basic functional modules in Figure 1.2, there are also some optional modules for overall performance improvement. For example, a module for estimating the fading channel state information can be inserted so that better soft-decision decoding metrics can be extracted. Another possible optional module can be an interpolation module which is responsible for detecting unreliable codeword and performing error concealment. These issues will be described in detail in later chapters.

This thesis considers a block-based vocoder, in particular the Federal Standard 1016 CELP [6], as an example. The selection of a specific vocoder is not important; our discussion can be extended and applied to other vocoders as well. The modulation scheme in this thesis is differential phase shift keying (DPSK) with non-coherent demodulation. The modulation scheme affects the soft-decision decoding metric; if other modulation is selected, the method of extracting the metric must be changed accordingly. Recently, there has been some interest in estimating the received carrier phase and amplitude by using a pilot tone [7] or pilot symbols [8]-[10] so that coherent modulation techniques can be used. However, these results are still preliminary, and further work is still required in this area.

Since the fading factor $a(t)$ — which is a random process — has significant memory, (deep) fades may last for a duration of one or even several codewords, causing bursty errors. As a result, interleaving/deinterleaving are required so that the bursty errors can be spread out over several codewords, thereby lowering the decoding error probability.

In a typical wireless system, inter-symbol and cochannel interference may also impose a strict limit on the overall system performance. The techniques for mitigating interference involve pulse shaping, equalization, and power control. Since we are focusing on the coding aspect of the system, we will ignore the effect of interference.

1.2 Soft-Decision Decoding

In a conventional communication system, the transmitter transmits one of M possible information symbols, and the demodulator uses the received signal to decide which one of the M symbols was transmitted and passes this particular symbol to the decoder. In essence, the demodulator makes an M-level quantization, and this decoding methodology is called **hard-decision decoding**. For instance, a binary demodulator (i.e., $M = 2$) decides whether 0 or 1 was transmitted. In contrast, if the demodulator makes a finer quantization prior to the decoder (or doesn't quantize at all), the demodulator is making a **soft-decision** [4]. Although soft-decision decoding makes the decoder more complex to implement, it offers significant performance improvement over hard-decision decoding. If the M symbols are equally likely to be transmitted, the ultimate soft-decision decoding is maximum-likelihood (ML) decoding, where the demodulator doesn't

quantize at all. In [3], it is shown that ML soft-decision decoding achieves a 2 dB gain over hard-decision decoding on AWGN channels; in a Rayleigh fading channel, the improvement is even more significant - up to 6 dB and beyond.

One of the major challenges for implementing a soft-decision decoding system is designing an efficient soft-decision decoding algorithm - i.e., an algorithm that efficiently uses the information provided by the demodulator. Numerous approaches to soft-decision decoding of linear block codes have been offered over the past three decades (e.g. [11]-[25]), and new algorithms are still being investigated. For codes with good trellis structures, like convolutional codes, trellis codes, and some block codes, the Viterbi decoding algorithm [26] is by far the most popular. Other soft-decision decoding algorithms are still available for convolutional codes, e.g., sequential decoding [27, 28].

Except for Viterbi decoding and decoding algorithms for some small or medium-size block codes, most soft-decision decoding algorithms are not optimal (i.e., they do not achieve ML soft-decision decoding). Practically, a soft-decision decoding algorithm is designed to achieve a near-ML decoding performance with feasible decoding complexity.

Soft-decision decoding algorithms (especially the Viterbi algorithm) can also be jointly designed with the modulation — the result is called coded-modulation — to significantly improve system performance without any loss of bandwidth efficiency [29, 30]. This approach is particular attractive for band-limited systems. By calculating the cutoff rates for QPSK and 8PSK, it is shown that coded modulation can obtain a 5.2 dB gain over an uncoded system at error probability of 10^{-5} on AWGN channels; on Rayleigh fading channels, the improvement significantly increases to 38 dB [31]. (The bandwidth efficiency in each case is 2

bits/symbol.)

Due to the poor performance of hard-decision decoding on a fading channel, and the significant improvement of soft-decision decoding, it is very clear that soft-decision decoding is highly preferable in wireless systems. As such, this thesis focuses on soft-decision decoding.

Assume that a codeword from a block code $\mathcal{C}$ over GF(2^a) is expressed in the binary form $\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$, and is to be conveyed by M-ary modulation through a possibly fading channel. Assuming that $n' = n/(\log_2 M)$ is an integer, codeword $\mathbf{c}$ can be rewritten as a set of n' M-ary symbols, i.e. $\mathbf{c} = (b_0, b_1, \ldots, b_{n'-1})$.

Let $r_{cw}(t)$ denote the received waveform for demodulating/decoding a codeword, where $r_{cw}(t)$ consists of multiple disjointed smaller waveform segments because of interleaving. We further denote $r_j(t)$ as the segment of $r_{cw}(t)$ for demodulating the j^{th} symbol, where $j = 0, 1, ..., n' - 1$. Assuming $r_j(t)$ doesn't overlap with other waveform segments, the ML decoder selects the codeword $\hat{\mathbf{c}}$ as the transmitted codeword so that

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}\in\mathcal{C}} \sum_{j=0}^{j=n'-1} \log p(r_j(t)|b_j) \tag{1.1}$$

Usually, $\log p(r_j(t)|b_j)$ can be written as the sum of two terms as follows:

$$\log p(r_j(t)|b_j) = F(r_j(t)) + \mu_{b_j}(r_j(t)), \tag{1.2}$$

where $F(r_j(t))$ doesn't depend on b_j, and can be left out in the maximization. The optimal symbol metric of b_j is given by $\mu_{b_j}(r_j(t))$ (subject to a scaling). By this definition, every symbol position j will be associated with M metric values corresponding to the M possible different symbols that might lie at that position. Since the metric is derived from the conditional probability of the

received signal, it is strongly linked to the modulation scheme. In later chapters, metrics corresponding to DPSK will be investigated.

In case the a priori codeword probabilities are not uniform (i.e., there is some residual redundancy left in the source encoder's output), ML decoding can be modified to become maximum a posteriori (MAP) decoding simply by incorporating $\log p(\mathbf{c})$ into Eq. (1.1), where $p(\mathbf{c})$ is the priori probability of codeword $\mathbf{c}$. In addition, if the encoding function is systematic and information bits are mutually statistically independent, then the MAP metric $\tilde{\mu}_{b_j}(t)$ can be defined as:

$$\tilde{\mu}_{b_j}(t) = \begin{cases} \log(p(b_j)) + \mu_{b_j}(r_j(t)) & \text{if } b_j \text{ is an information symbol} \\ \mu_{b_j}(r_j(t)) & \text{otherwise.} \end{cases} \tag{1.3}$$

In this thesis, we will exploit the a priori statistics of the vocoder's output parameters and apply MAP decoding.

1.3 Error Concealment for Voice Communications

Although soft-decision decoding yields much better results than hard-decision decoding (especially on fading channels), soft-decision decoding may still fail to decode correctly, especially during deep fades. For some applications, such as mobile radio, it is preferable to have the channel decoder detect the presence of errors - i.e., declare a decoder failure - rather than attempt to correct the error and (in all likelihood) do so incorrectly. The detected errors can be masked by interpolation/extrapolation or by squelching [32]. In voice communications, undetected decoding errors often result in clicks, squeaks, and other perceptually

annoying effects; whereas for detected noise, error masking techniques can be applied to conceal the errors. Often there is sufficient "residual redundancy" in compressed voice to permit a form of error masking.

Since "pure" maximum likelihood soft-decision decoding algorithms always make an estimate of the transmitted codeword, the inherent capability of a source decoder to mask detected errors is not exploited. We will address the issue of augmenting the pure ML decoder with the capability of detecting unreliable codewords for subsequent interpolation.

1.4 Thesis Outline

This thesis is presented in the following order. Chapter 2 will give a brief preliminary description of a multipath propagation channel. Among the channel aspects to be mentioned are long-term and short-term statistical distribution of the received signals, Doppler frequency, delay spread, coherent bandwidth, average fade duration, and fading channel simulation on computers.

In Chapter 3, the theoretical limits - i.e., the channel capacity and cutoff rates - of fading channels under DPSK modulation will be addressed. These theoretical figures are important in that they provide fundamental limits on how well a coding scheme can perform under DPSK modulation. In order to calculate cutoff rates, a decoding metric must first be specified; therefore, we define several metrics for DPSK and investigate the corresponding cutoff rates. By comparing the cutoff rates of various DPSK metrics, we obtain an idea about the quality of these metrics. These metrics will be investigated more closely in Chapter 4 by calculating the corresponding decoding error probabilities. We will

show that the metric performance comparison in terms of decoding error rates is consistent with their cutoff rates. In addition, we also investigate the issue of bit interleaving versus symbol interleaving on fading channels.

Chapter 5 provides an error detection procedure for ML soft-decision decoding. This detection procedure is followed by an interpolation algorithm for concealing detected errors. The details of the error detection and interpolation algorithm are closely related to the application and the underlying vocoder. We chose the Federal Standard 1016 CELP as our exemplary vocoder.

Chapter 6 investigates the problem of exploiting the a priori probabilities and correlation of the information bits in order to enhance a channel decoder's performance. In other words, MAP soft-decision decoding will be addressed. To justify the usefulness of MAP soft-decision decoding, we measure the statistics of some parameters of the F.S. 1016 CELP and demonstrate that significant coding gain can be achieved (especially at low SNR) by MAP decoding.

Finally, conclusions and future work will be presented in Chapter 7.

Chapter 2

Statistical Models for Fading Channels

Wireless channels are usually too complex to characterize precisely due to virtually unlimited factors which may dynamically influence the radio signal. There are basically two models that describe a wireless channel: the propagation model and the statistical model [5]. The propagation model is concerned with those physical phenomena which could affect the propagating signals. Among them are free space signal loss, reflection, surface wave, diffraction, geometrical environment, antenna height, antenna structure, carrier frequency, polarization, atmospheric condition, etc. As one can see, countless factors need to be considered in a deterministic treatment of the channel, leading to a very complicated analysis. As an alternative, one usually treats the channel on a statistical basis and interprets the signal being affected by random events occurring with given distributions. Based on [5] and [35], this chapter will briefly describes the statistical aspects of fading channels, some of which will be useful in later chapters.

2.1 Signal Distributions

There are three basic distributions which closely characterize the envelope of the received signal - lognormal, Rayleigh, and Rician. The lognormal distribution describes long-term fading, whereas Rayleigh and Rician distributions describe short-term fading. Besides these three distributions, there are still other distributions for characterizing fading channels; for example, the Suzuki distribution is formed by mixing a lognormal and a Rayleigh distribution, and corresponds to the long-term overall distribution [5].

Since a channel codeword is usually transmitted over a relatively short-term period, long-term fading is usually not explicitly considered in the design of a coding scheme [34]. In contrast, Rayleigh and Rician distribution describe short-term fading so that they are important factors affecting the performance of a coding scheme. In addition to the envelope distributions, we will also consider the corresponding phase distributions.

2.1.1 Lognormal Distribution

As the received signal is shadowed by obstructions such as hills, woods, buildings, and other large objects, the lognormal distribution is a suitable description of the received signal envelope.

Assume E_0 as the electric field magnitude of a plane wave, and E as the magnitude after the plane wave propagates through a medium with thickness h and an attenuation factor α. By the plane wave propagation formula [33], we have

$$E = E_0 \exp(-\alpha h).$$

If the signal propagates through L objects, the above equation is rewritten as

$$E = E_0 \exp\left(-\sum_{i=1}^{L} \alpha_i h_i\right). \tag{2.1}$$

As L becomes large, the exponent in the above equation approaches a normal (Gaussian) distribution (by the central limit theorem). As a result, the received signal envelope has a lognormal distribution.

2.1.2 Rayleigh Distribution

For a mobile radio channel, it is rare that the received signal contains a direct line-of-sight component; in other words, the received signal is the sum of the signals formed by the transmitted signal scattered by randomly placed obstructions imposing different attenuations and phases on the resultant signal. This is known as multipath propagation. Letting n be the number of the scattered waves, the resultant signal (in complex form) can be written as

$$r\exp(j\theta) = \sum_{i=1}^{n} a_i \exp(j\theta_i) \tag{2.2}$$

where a_i and θ_i is the amplitude and phase of the i^{th} scattered wave. Denoting

$$x = \sum_{i=1}^{n} a_i \cos\theta_i \quad \text{and} \quad y = \sum_{i=1}^{n} a_i \sin\theta_i$$

(2.2) can be rewritten as

$$r\exp(j\theta) = x + jy.$$

It is suitable to assume that θ_i is uniformly distributed on $[0, 2\pi]$, and each scattered wave is independent of other scattered waves; as a result, x and y are identically distributed but mutually independent Gaussian variables (assuming n is large) with zero mean.

Since $x = r\cos(\theta)$ and $y = r\sin(\theta)$, the joint probability density $p(r, \theta)$ can be derived from the distribution of x and y. It can be shown [5, 35] that

$$p(r, \theta) = \frac{r}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) \quad \text{for } r \geq 0,\ 0 \leq \theta \leq 2\pi, \tag{2.3}$$

where σ^2 is the variance of x (and y). Then the distribution of r is

$$\begin{aligned} p(r) &= \int_0^{2\pi} p(r, \theta) d\theta \\ &= \begin{cases} \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) & r \geq 0 \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{2.4}$$

It is clear that the envelope r has a Rayleigh distribution and is independent of the phase θ (which is uniformly distributed on $[0, 2\pi]$).

2.1.3 Rician Distribution

In an environment where a line-of-sight signal exists along with multipath propagation, (2.2) can be modified as

$$r\exp(j\theta) = m\exp(j\eta) + \sum_{i=1}^{n} a_i \exp(j\theta_i) \tag{2.5}$$

where $m\exp(j\eta)$ is the line-of-sight component. We then have

$$r\cos(\theta) = m\cos(\eta) + x \quad \text{and} \quad r\sin(\theta) = m\sin(\eta) + y,$$

where x and y represent the in-phase and quadrature components of the multipath propagation. It can be verified [35] that the conditional probability density function of the envelope given the line-of-sight phase η is

$$p(r|\eta) = \frac{r}{\sigma^2} \exp\left[-\frac{1}{2\sigma^2}(r^2 + m^2)\right] I_0\left(\frac{mr}{\sigma^2}\right), \text{ for } r \geq 0, \tag{2.6}$$

where $I_0(\cdot)$ is the zeroth order modified Bessel function of the first kind. Clearly, r has a Rician distribution, and it is independent of the phase η of the line-of-sight signal. As for the distribution of θ, it can be shown [35] that

$$\begin{aligned} p(\theta|\eta) &= \frac{\exp(-m^2/2\sigma^2)}{2\pi} + \frac{m\cos(\eta-\theta)}{2(2\pi)^{1/2}\sigma}\exp\left[-\frac{m^2}{2\sigma^2}\sin^2(\eta-\theta)\right] \\ &\quad \times\left\{1+\operatorname{erf}\left[\frac{m\cos(\eta-\theta)}{\sqrt{2}\sigma}\right]\right\} \end{aligned} \tag{2.7}$$

where

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-z^2}dz$$

is the error function. From (2.7), we see that the phase of the resultant signal is symmetrically distributed around the line-of-sight phase η.

2.2 Doppler Frequency

In mobile radio communications, a mobile is moving at a speed v. If the carrier arrives at an angle ϕ, the speed of the mobile will cause a new propagation phase velocity v_p' given by

$$v_p' = v_p - v\cos\phi$$

where v_p is the original phase velocity. In turn, this will cause a new frequency f':

$$f' = f_c - \frac{v}{\lambda}\cos\phi$$

where f_c and λ are the carrier frequency and wavelength, respectively. We denote $f_D \triangleq \frac{v}{\lambda}$ as the maximum Doppler shift.

Doppler shift is an important factor affecting the correlation of received signals. Usually, higher speed will cause faster fading and less correlated signals,

while lower speed results in slow fading and increased correlation. This issue will be elaborated on shortly.

2.3 Delay Spread

Due to multipath propagation, different replicas of the transmitted signal arrive at the receiver at different time instants and with different amplitudes. If an impulse $a\delta(t)$ is transmitted at time instant $t = 0$, the received signal $r(t)$ will be

$$r(t) = \sum_{i=1}^{n} a_i \delta(t - T_i)$$

where T_i is the time delay of the ith arrived pulse.

Each time delay T is a random variable, and the delay spread σ_T is defined as the standard deviation of T. A precise characterization of arrival time usually is not available; thus, the probability density of T is described as an exponential function:

$$p(T) = \frac{1}{m_T} \exp\left(-\frac{T}{m_T}\right)$$

where m_T is the mean arrival time.

In practice, the delay spread is short (a fraction of microsecond) in suburban and open areas, whereas it is longer (a few microseconds) in urban areas. Like Doppler shift, delay spread also contributes to the correlation of received signals; furthermore, delay spread could cause intersymbol interference.

2.4 Coherence Bandwidth

Consider a mobile radio signal with a given bandwidth. Due to the multipath effect (e.g., Doppler shift and delay spread), each frequency component of the signal may arrive at the destination with different time delays, causing some signal distortion. Therefore, it is very important to determine the maximum frequency separation for which the signals are still considered to be strongly correlated (i.e., not severely distorted). This maximum frequency separation is called the coherence bandwidth [5].

Systems operating with channels substantially narrower than the coherence bandwidth are known as narrowband systems, whereas those systems operating with a bandwidth larger than the coherence bandwidth are wideband systems. In a narrowband system, all signal components are equally influenced by the multipath propagation, and the radio signal experiences **flat fading**. In a wideband system, the signal components are not equally affected by the channel, characterizing the phenomenon known as **frequency selective fading**.

2.4.1 Joint Probability Density of In-Phase and Quadrature Components

Assuming v as the mobile speed, f_c the carrier frequency, and T_i the propagation delay of the ith path, the received signal can be written as

$$E = E_0 \sum_{i=1}^{n} a_i \exp[j2\pi(f_c t + f_i t - f_c T_i)] \tag{2.8}$$

where $f_i = \frac{v \cos\theta_i}{\lambda}$ is the Doppler shift for the ith path and a_i^2 represents the fraction of incoming power at angle θ_i and time delay T_i. Assuming a uniform

distribution of the incident angle and the number of incidence paths n approaching infinity, we have

$$\lim_{n\to\infty} a_i^2 = \frac{1}{2\pi m_T} \exp\left(-\frac{T}{m_T}\right) d\theta dT. \tag{2.9}$$

Let X and Y be the in-phase and quadrature component of the received signal, i.e.,

$$X = E_0 \sum_{i=1}^{n} a_i \cos[2\pi(f_i t - f_c T_i)] \triangleq r\cos\theta \tag{2.10}$$

$$Y = E_0 \sum_{i=1}^{n} a_i \sin[2\pi(f_i t - f_c T_i)] \triangleq r\sin\theta \tag{2.11}$$

Note that

$$r^2 = X^2 + Y^2 \text{ and } \theta = \tan^{-1}(Y/X). \tag{2.12}$$

Now consider two signals propagating at frequencies f_{c_1} and f_{c_2}. Assume that the first signal arrives at time t and the second signal arrives at $t+\tau$. We have the following four Gaussian random variables:

$$X_1 = E_0 \sum_{i=1}^{n} a_i \cos[2\pi(f_i t - f_{c_1} T_i)] \triangleq r_1 \cos\theta_1$$

$$Y_1 = E_0 \sum_{i=1}^{n} a_i \sin[2\pi(f_i t - f_{c_1} T_i)] \triangleq r_1 \sin\theta_1$$

$$X_2 = E_0 \sum_{i=1}^{n} a_i \cos[2\pi(f_i t + f_i \tau - f_{c_2} T_i)] \triangleq r_2 \cos\theta_2$$

$$Y_2 = E_0 \sum_{i=1}^{n} a_i \sin[2\pi(f_i t + f_i \tau - f_{c_2} T_i)] \triangleq r_2 \sin\theta_2$$

To determine the joint probability density of the four zero-mean Gaussian variables, it suffices to calculate the corresponding covariance matrix. There are 16 entries in the covariance matrix, but many of them are the same. We now compute $\mathrm{Cov}[X_1, X_1]$ and $\mathrm{Cov}[X_1, X_2]$; other covariance entries can be obtained

in the same manner.

$$\begin{aligned}\mathrm{Cov}[X_1, X_1] &= < X_1^2 > \\ &= E_0^2 \sum_{i,j} < a_i a_j \cos(2\pi f_i t - 2\pi f_{c_1} T_i) \cos(2\pi f_j t - 2\pi f_{c_1} T_j) >\end{aligned} \tag{2.13}$$

Clearly, the average is 0 unless $i = j$; therefore,

$$\mathrm{Cov}[X_1, X_1] = \sigma^2 \sum_i a_i^2$$

where $\sigma^2 = E_0^2/2$. Furthermore,

$$\lim_{n\to\infty} \mathrm{Cov}[X_1, X_1] = \sigma^2 \int_0^{2\pi} \int_0^{\infty} \frac{1}{2\pi m_T} \exp\left(-\frac{T}{m_T}\right) d\theta dT = \sigma^2.$$

For $\mathrm{Cov}[X_1, X_2]$, we have

$$\begin{aligned}\mathrm{Cov}[X_1, X_2] &= < X_1 X_2 > \\ &= E_0^2 \sum_{i,j} < a_i a_j \cos(2\pi f_i t - 2\pi f_{c_1} T_i) \cos(2\pi f_j t + 2\pi f_j \tau - 2\pi f_{c_2} T_j) > \\ &= \sigma^2 \sum_i a_i^2 \cos[2\pi(f_i \tau - \triangle f T_i)]\end{aligned} \tag{2.14}$$

where $\triangle f = f_{c_2} - f_{c_1}$. As $n \to \infty$,

$$\begin{aligned}\mathrm{Cov}[X_1, X_2] &= \int_0^{2\pi} \int_0^{\infty} \frac{1}{2\pi m_T} \exp\left(-\frac{T}{m_T}\right) \cos(\frac{2\pi v \tau}{\lambda} \cos\theta - 2\pi \triangle f T) d\theta dT \\ &= \frac{\sigma^2 J_0(2\pi f_D \tau)}{1 + (2\pi \triangle f m_T)^2}\end{aligned} \tag{2.15}$$

where $f_D = \frac{v}{\lambda}$ is the maximum Doppler shift and $J_0(\cdot)$ is the zeroth order Bessel function of the first kind.

Given $\mathrm{Cov}[X_1, X_1]$ and $\mathrm{Cov}[X_1, X_2]$, the correlation coefficient - denoted as ξ - between X_1 and X_2 is easily derived as

$$\xi = \frac{J_0(2\pi f_D \tau)}{1 + (2\pi \triangle f m_T)^2}. \tag{2.16}$$

After a tedious procedure to determine the whole covariance matrix, the joint distribution among X_1, Y_1, X_2, and Y_2 is [5]

$$p(X_1, Y_1, X_2, Y_2) = \frac{1}{4\pi^2\sigma^4(1-\rho^2)} \times \exp$$
$$\left\{-\frac{\sigma^2(X_1^2+Y_1^2+X_2^2+Y_2^2) - 2\xi\sigma^2(X_1X_2+Y_1Y_2) + 2\xi\sigma^2\triangle f m_T(X_1Y_2 - X_2Y_1)}{2\sigma^8(1-\rho^2)^2}\right\} \tag{2.17}$$

where

$$\rho^2 = \xi J_0(2\pi f_D \tau).$$

2.4.2 Correlation of Envelope and Phases

The joint distribution among r_1, r_2, θ_1 and θ_2 can be derived from (2.12) and (2.17), and is evaluated to be

$$p(r_1, r_2, \theta_1, \theta_2) = \frac{r_1 r_2}{4\pi^2\sigma^4(1-\rho^2)} \times \exp$$
$$\left\{-\frac{1}{2\sigma^8(1-\rho^2)^2}[\sigma^2(r_1^2+r_2^2) - 2r_1r_2\xi\sigma^2\cos(\theta_2-\theta_1) + 2r_1r_2\xi\sigma^2\triangle f m_T \sin(\theta_2-\theta_1)]\right\}$$

The joint density for the envelope is

$$\begin{aligned} p(r_1, r_2) &= \int_0^{2\pi}\int_0^{2\pi} p(r_1, r_2, \theta_1, \theta_2)\, d\theta_1 d\theta_2 \\ &= \frac{r_1 r_2}{\sigma^4(1-\rho^2)} \exp\left[-\frac{r_1^2+r_2^2}{2\sigma^2(1-\rho^2)}\right] I_0\left(\frac{r_1 r_2 \rho}{\sigma^2(1-\rho^2)}\right) \end{aligned} \tag{2.18}$$

and the joint density for the phase is

$$\begin{aligned} p(\theta_1, \theta_2) &= \int_0^{\infty}\int_0^{\infty} p(r_1, r_2, \theta_1, \theta_2)\, dr_1 dr_2 \\ &= \left(\frac{1-\rho^2}{4\pi^2}\right) \frac{(1-\Upsilon^2)^{1/2} + \Upsilon\cos^{-1}(-\Upsilon)}{(1-\Upsilon^2)^{3/2}} \end{aligned} \tag{2.19}$$

where

$$\Upsilon = \rho\cos[\theta_2 - \theta_1 + \tan^{-1}(2\pi\triangle f m_T)].$$

The correlation coefficients of the received signal envelopes and phases can be derived from the joint density $p(r_1, r_2)$ and $p(\theta_1, \theta_2)$ presented above. After some algebra, the correlation coefficient for the envelope is

$$\rho_r \approx \frac{\pi}{4(4-\pi)}\rho^2 \approx \rho^2 = \frac{J_0^2(2\pi f_D \tau)}{1 + (2\pi \triangle f m_T)^2} \tag{2.20}$$

and that for the phases is

$$\rho_\theta = 3\Gamma(\rho, \phi)[1 + 2\Gamma(\rho, \phi)] - \frac{3}{4\pi^2}\sum_{i=1}^{\infty}\frac{\rho^{2i}}{i^2} \tag{2.21}$$

where

$$\begin{aligned} \Gamma(\rho, \phi) &= \frac{1}{2\pi}\sin^{-1}(\rho\cos\phi) \\ \phi &= -\tan^{-1}(2\pi \triangle f m_T). \end{aligned}$$

2.4.3 The Coherence Bandwidth

The coherence bandwidth (B_c) is defined as the frequency separation for which the correlation coefficient is at least 0.5 under a time delay $\tau = 0$. If the envelope correlation is used, B_c is determined by (2.20) to be $\frac{1}{2\pi m_T}$; if the phase correlation is used, B_c is determined by (2.21) to be $\frac{1}{4\pi m_T}$. Clearly, B_c is decided by the mean delay spread m_T, and the coherence bandwidth for the envelope is larger than the one for the phase. Usually, the coherence bandwidth for the envelope is adopted.

For a mean delay spread $m_T = 2\mu s$, the corresponding B_c is 80 kHz; for $m_T = 5\mu s$, the corresponding B_c is 32 kHz. Practically, a delay spread larger than $10\mu s$ is very rare. In this thesis, we assume the delay spread is short enough so that the channel bandwidth is smaller than the coherence bandwidth. In other words, we consider flat fading.

In M-ary digital modulation systems, each symbol transmitted conveys $\log_2 M$ bits. If the symbol rate is less than 20 thousand symbols per second, approximately 20 kHz bandwidth is required for the channel. Assuming the maximum delay spread is less than $m_T = 5\mu s$, the channel bandwidth is much less than the coherence bandwidth, and the flat fading assumption is appropriate.

2.5 Average Duration of Fades

Another aspect of fading channels which may affect mobile radio system performance is the duration of fades [5]. Fading occurs when the received signal envelope falls below some prescribed threshold r_T; the channel may exit the fading status very soon or remain in fading for a while. We will show that the average duration of fading is inversely proportional to the mobile speed. Long duration of fading causes bursty errors at the receiver and may seriously degrade the communication quality; usually, interleaving is employed to mitigate the effect of bursty errors.

2.5.1 Level Crossing Rate

The level crossing rate is the average number of times a fading signal crosses a given signal level within a certain period. Let $\dot{r} = dr/dt$ and the crossing signal level be R. The corresponding average level crossing rate R_c can be shown to be [36]

$$R_c = \int_0^\infty \dot{r} p(R, \dot{r}) d\dot{r} \tag{2.22}$$

where $p(R, \dot{r})$ is the joint probability density of r and $\dot{r}$ at $r = R$. As such, the joint density $p(r, \dot{r})$ is needed for calculating (2.22).

Taking the derivative of (2.10) and (2.11), we have

$$\dot{X} = E_0 2\pi f_D \sum_{i=1}^{n} -a_i \sin(2\pi f_i t - 2\pi f_c T_i)\cos\theta_i \tag{2.23}$$

$$\dot{Y} = E_0 2\pi f_D \sum_{i=1}^{n} a_i \cos(2\pi f_i t - 2\pi f_c T_i)\cos\theta_i \tag{2.24}$$

Obviously, both $\dot{X}$ and $\dot{Y}$ are still Gaussian variables; furthermore, it can be verified that the four variables $\{X, Y, \dot{X}, \dot{Y}\}$ are mutually independent. Thus, the joint probability density $p(X, Y, \dot{X}, \dot{Y})$ can be written as

$$p(X, Y, \dot{X}, \dot{Y}) = \frac{1}{4\pi^2\sigma^2\sigma_d^2} \exp\left[-\frac{1}{2}\left(\frac{X^2+Y^2}{\sigma^2} + \frac{\dot{X}^2+\dot{Y}^2}{\sigma_d^2}\right)\right] \tag{2.25}$$

where $\sigma^2 = E_0^2/2$ and $\sigma_d^2 = (E_0\pi f_D)^2$. Since $X = r\cos\theta$ and $Y = r\sin\theta$,

$$\dot{X} = \dot{r}\cos\theta - r\dot{\theta}\sin\theta \tag{2.26}$$

$$\dot{Y} = \dot{r}\sin\theta + r\dot{\theta}\cos\theta. \tag{2.27}$$

The joint probability density $p(r, \dot{r}, \theta, \dot{\theta})$ is derived from $p(X, Y, \dot{X}, \dot{Y})$ and can be easily shown to be

$$p(r, \dot{r}, \theta, \dot{\theta}) = \frac{r^2}{4\pi^2\sigma^2\sigma_d^2} \exp\left[-\frac{1}{2}\left(\frac{r^2}{\sigma^2} + \frac{\dot{r}^2 + r^2\dot{\theta}^2}{\sigma_d^2}\right)\right] \tag{2.28}$$

The distribution $p(r, \dot{r})$ is now obtained as

$$p(r, \dot{r}) = \int_{-\infty}^{\infty}\int_{0}^{2\pi} p(r, \dot{r}, \theta, \dot{\theta}) d\theta d\dot{\theta} \tag{2.29}$$

Substituting (2.29) into (2.22), we have

$$R_c = \sqrt{\pi} f_D \frac{R}{\sigma} \exp\left[-\left(\frac{R}{\sqrt{2}\sigma}\right)^2\right]. \tag{2.30}$$

2.5.2 Average Fade Duration

In a time period T, there are (on the average) R_cT occurrences of fading. Let ν be the average fade duration; by definition,

$$\nu = \frac{\sum \nu_i}{R_cT}$$

where ν_i is the duration of ith fade. The ratio $\sum \nu_i/T$ corresponds to the probability that the signal is below R. Hence,

$$\nu = \frac{1}{R_c}\text{Prob}(r \leq R) = \frac{1}{R_c}\int_0^R p(r)dr. \tag{2.31}$$

Assuming $p(r)$ is Rayleigh distributed and substituting (2.30) into (2.31), we obtain

$$\nu = \frac{1}{\sqrt{2\pi}f_D(R/\sqrt{2}\sigma)}\left[\exp\left(\frac{R}{\sqrt{2}\sigma}\right)^2 - 1\right]. \tag{2.32}$$

By using (2.32), it is easy to verify that ν is a monotonically increasing function of R, which is expected. In addition, ν is inversely proportional to f_D - the maximum Doppler shift. This indicates that the higher/lower the mobile speed is, the shorter/longer the average fade duration is. For example, at $R = \sigma$ and $f_D = 60$Hz, the average fade duration is 5.5 ms.

2.6 Fading Channel Simulation

The performance evaluation of various modulation and coding strategies over a fading channel can be performed using a channel simulator [37]-[40]. In the following description, we describe a software channel simulator that will be employed throughout this thesis [41].

Let the modulation symbol duration be T_s. The required bandwidth for such a waveform is roughly equal to $1/T_s$. From our earlier discussion on coherence

bandwidth, it is clear that the channel is a flat fading channel if $T_s >> m_T$. Another important consideration is the coherence time - a quantity describing how fast the fading process is. From (2.20) with $\triangle f = 0$, it is observed that the envelope won't change rapidly as long as $f_D T_s << 1$; in this case, the signal during a symbol period (namely T_s seconds) can be treated as constant (sometimes referred to as slow-fading). In a typical mobile radio system, the double inequality $m_T << T_s << 1/f_D$ usually holds; hence, this thesis assumes flat fading and a constant received signal for each symbol. Under this assumption, the fading factor for each symbol can be described by a discrete time quantity, say $ae^{j\theta}$. Our channel simulator is therefore based on discrete time representation.

Specifically, let k index the modulation symbol; the transmitted signal is represented by u_k and the received signal by r_k, where both u_k and r_k are complex representation of the signals. Let Z_k be the fading factor and n_k be the additive white Gaussian noise, we have

$$r_k = Z_k \cdot u_k + n_k \tag{2.33}$$

For Rayleigh fading, Z_k is a complex Gaussian random variable with zero mean; for Rician fading, Z_k is still Gaussian, but with non-zero mean.

Let $Z_k = X_k + jY_k$; the correlation coefficient between X_k and X_{k-h} (and also between Y_k and Y_{k-h}) is $J_0(2\pi f_D h T_s)$ (see (2.16)). Since the Fourier transform of $J_0(2\pi f_D t)$ (i.e., the power spectrum of X_k and Y_k) is [31]

$$S(f) = \frac{1}{\pi\sqrt{f_D^2 - f^2}}, \quad |f| < f_D \tag{2.34}$$

X_k can be obtained by passing a white Gaussian random process into a FIR filter with a power shaping function of (2.34). Y_k (which is independent of X_k) can be obtained by passing another independent white Gaussian random process into

the same FIR filter. The taps of the FIR only depend on the maximum Doppler shift $f_D = v/\lambda$. As such, given the speed of the mobile as well as the carrier frequency, the FIR filter can be designed.

The channel simulator is shown in Fig. 2.1. This simulator contains two

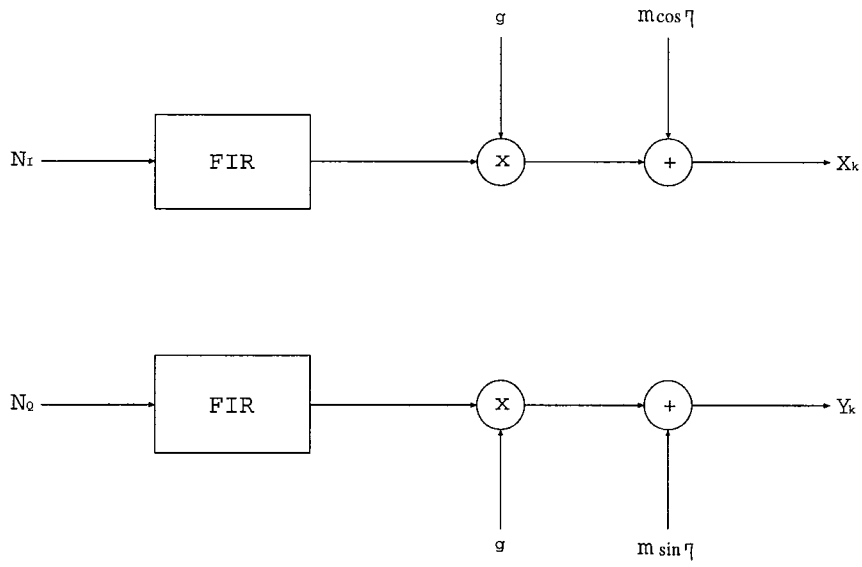


Figure 2.1: The fading channel simulator

branches; the upper one corresponds to the in-phase component and is fed with a white Gaussian random sequence N_I; the lower one corresponds to the quadrature component and is fed with another independent random sequence N_Q. The terms $m\cos\eta$ and $m\sin\eta$ correspond to the line-of-sight component, where η and m are the phase and the amplitude of this line-of-sight component, respectively. If $m = 0$, this channel degenerates into a Rayleigh channel and g represents the gain of multipath propagation.

In 2.1, we let both variances of N_I and N_Q are unity and the taps of the FIR be scaled so that the FIR output Gaussian variables still have unity variance. Without loss of generality, we assume the fading process doesn't change the

average signal energy (i.e., the fading factor is normalized), $E[|Z_k|^2] = 1$. We then have

$$m = \sqrt{\frac{K_r}{1+K_r}} \quad \text{and} \quad g = \frac{1}{\sqrt{2(1+K_r)}} \tag{2.35}$$

where K_r is the ratio of the line-of-sight energy with respect to the multipath energy, and is usually named the Rician factor. When $K_r = 0$, the channel is a normalized Rayleigh fading channel; as $K_r \to \infty$, it corresponds to an AWGN channel.

Chapter 3

Theoretic Limits of Fading Channels Under DPSK

Since DPSK modulation is assumed throughout this thesis, it is worthwhile to investigate the fundamental information-theoretic limits of fading channels under this modulation scheme. This chapter evaluates the channel capacity and cutoff rates of fading channels under two different assumptions – that channel state information (CSI) is available at the receiver and that CSI is **not** available at the receiver. The CSI in this chapter refers to the effects of fading on the received signal. These theoretic figures can serve as a performance limit against which the performance of a specific coding scheme can be compared. In addition, these limits also shed light on how the CSI influences a receiver's performance.

The information theoretic limits of fading channels have become of increasing interest as wireless communications have become more popular. Past work related to fading channel capacity has considered signal constellations of unbounded cardinality, or alternatively, M-ary signaling with coherent detection [42]-[45]. In contrast, this chapter is concerned with the non-coherent detection of differentially-encoded PSK signals. Some work on DPSK cutoff rates has been

reported [30, 48]; however, a heuristic "Gaussian metric" was consistently used in those papers, and was labeled as sub-optimal. We re-examine the cutoff rates by deriving the optimal metric and show that the "Gaussian metric" is in fact optimal for Rayleigh fading channels when the CSI is not known. In addition to the optimal metric (i.e., the metric which yields the maximum-likelihood decoding), some suboptimal metrics that allow for easier implementation are introduced. We calculate the cutoff rates for both the optimal and the suboptimal metrics.

3.1 Channel Model

Before any theoretic limit of a channel may be derived, the channel must be precisely specified. We assume $\log_2 M$ bits are conveyed every T_s seconds by an M-ary DPSK scheme - i.e., by a phase change in adjacent transmitted signals. The statistics of the metrics extracted from the received waveform corresponding to any $\log_2 M$ bits in a long codeword are independent of those of other bits in the same codeword. This can be achieved with a sufficiently large interleaving table. For example, consider such a table consisting of n columns and J rows; each element of the table is a $\log_2 M$-bit symbol, and assume that each row contains a codeword from a block code. If the data from such an interleaver is modulated and transmitted column-by-column, then the phase changes associated with two $\log_2 M$ symbols in the same codeword are separated in time by at least J signal intervals; assuming J is sufficiently large, this arrangement allows us to neglect the channel memory. The method of extracting metrics from the transition between two received signals will be explained later.

As indicated in the previous chapter, it is assumed that (1) the inter-symbol

and cochannel interference can be neglected; and (2) the channel is a flat fading channel and the fading process remains essentially constant over an entire signal interval. However, the fading process may change over two adjacent signals. In this thesis, a channel is called slow fading if the fading process remains unchanged over two adjacent received signals; otherwise, it is called fast fading.

The remaining important characteristics of a system that need to be specified, in order to determine the theoretic limits, are the channel input/output and their interaction with the channel; these are specified below.

3.1.1 The DPSK Signal

Consider a M-ary DPSK scheme in which a phase shift of ϕ_i between two adjacent signals is used to convey symbol i, where $i \in \{0, 1, 2, ..., M-1\}$. Let T_s be the symbol duration and let $s_i(t)$ be the signal used to convey symbol i in the time interval $[0, 2T_s]$, i.e.,

$$s_i(t) = \begin{cases} A \cdot \cos(2\pi f_c t + \eta), & \text{if } 0 < t \leq T_s \\ A \cdot \cos(2\pi f_c t + \eta + \phi_i), & \text{if } T_s < t \leq 2T_s \end{cases} \tag{3.1}$$

where $A = \sqrt{\frac{2E_s}{T_s}}$ is the carrier amplitude and η, uniformly distributed on $[0, 2\pi]$, is the phase uncertainty from the viewpoint of the DPSK receiver. Practically, the carrier amplitude may be subject to a shaping pulse (e.g., raised cosine pulse) in order to conserve bandwidth and avoid intersymbol interference. However, since the matched filter will also be designed to match the shaping pulse, it can be verified that the following discussion will not be affected by the pulse shaping; for simplicity, we assume a constant carrier amplitude A throughout this chapter.

On fading channels, the received carrier experiences both amplitude fade and phase shift. Let a_1 and θ_1 be the carrier amplitude fading factor and the carrier phase respectively during $[0, T_s]$, and a_2 and $\theta_2 + \phi_i$ be the fading factors and the phase respectively during $[T_s, 2T_s]$. On Rayleigh fading channels (in which there is no line-of-sight component received), we thus have (during $[0, T_s]$)

$$Aa_1 \cos(2\pi f_c t + \theta_1) = \sum_k s_{i,k}(t)$$

where $s_{i,k}(t)$ is the k^{th} multipath component received at the receiver. Under the conventional assumption that the number of multipath components is large, and every one of them is independent and has random phase, a_1 becomes Rayleigh distributed and θ_1 is uniformly distributed on $[0, 2\pi]$. Obviously, the same properties hold for a_2 and θ_2 during $[T_s, 2T_s]$. For Rician channels (in which a line-of-sight component is received), we have (during $[0, T_s]$)

$$Aa_1 \cos(2\pi f_c t + \theta_1) = A\cos(2\pi f_c t + \eta) + \sum_k s_{i,k}(t)$$

where a_1 is Rician distributed. The phase θ_1 can be written as $\eta + \psi_1$, where ψ_1 is the phase shift caused by multipath components. Since η is uniformly distributed on $[0, 2\pi]$, θ_1 is therefore also uniformly distributed on $[0, 2\pi]$, as in the Rayleigh fading case. Again, the characteristics of a_1 and θ_1 also hold for a_2 and θ_2 during $[T_s, 2T_s]$.

Without loss of generality, assume the fading factors are normalized, i.e., $E(a_1^2) = E(a_2^2) = 1$. In addition, assume there is also additive white Gaussian noise $n(t)$ with power spectrum $N_0/2$. Thus the received waveform is

$$r(t) = \begin{cases} a_1 A \cdot \cos(2\pi f_c t + \theta_1) + n(t), & 0 < t \leq T_s \\ a_2 A \cdot \cos(2\pi f_c t + \theta_2 + \phi_i) + n(t), & T_s < t \leq 2T_s. \end{cases} \tag{3.2}$$

Letting the matched filters project $r(t)$ onto the two-dimensional space spanned by $\cos(2\pi f_c t)$ and $\cos(2\pi f_c t+\pi/2)$, we have the following matched filter outputs:

$$\begin{aligned} x_1 &= \int_0^{T_s} r(t)\cos(2\pi f_c t)dt \\ y_1 &= \int_0^{T_s} r(t)\cos(2\pi f_c t+\pi/2)dt \\ x_2 &= \int_{T_s}^{2T_s} r(t)\cos(2\pi f_c t)dt \\ y_2 &= \int_{T_s}^{2T_s} r(t)\cos(2\pi f_c t+\pi/2)dt. \end{aligned}$$

Assuming $s_i(t)$ is the transmitted signal, the matched filter outputs can be expressed as

$$x_1 = \frac{T_s A a_1}{2}\cos\theta_1 + n_{c,1} \tag{3.3}$$

$$y_1 = \frac{T_s A a_1}{2}\sin\theta_1 + n_{s,1} \tag{3.4}$$

$$x_2 = \frac{T_s A a_2}{2}\cos(\theta_2+\phi_i) + n_{c,2} \tag{3.5}$$

$$y_2 = \frac{T_s A a_2}{2}\sin(\theta_2+\phi_i) + n_{s,2}, \tag{3.6}$$

where $n_{c,1}$, $n_{s,1}$, $n_{c,2}$ and $n_{s,2}$ are independent zero-mean Gaussian variables with variance $N_0 T_s/4$.

By fixing the channel fading factors, the probability density of the received signals conditioned on s_i being transmitted is

$$\begin{aligned} p(x_1,y_1,x_2,y_2|s_i,a_1,a_2,\theta_1,\theta_2) &= \frac{4}{\pi^2 N_0^2 T_s^2}\cdot \\ & e^{-\frac{(\sqrt{\frac{2}{T_s}}x_1 - a_1 A\sqrt{\frac{T_s}{2}}\cos\theta_1)^2}{N_0}}\cdot e^{-\frac{(\sqrt{\frac{2}{T_s}}y_1 - a_1 A\sqrt{\frac{T_s}{2}}\sin\theta_1)^2}{N_0}}\cdot \\ & e^{-\frac{(\sqrt{\frac{2}{T_s}}x_2 - a_2 A\sqrt{\frac{T_s}{2}}\cos(\theta_2+\phi_i))^2}{N_0}}\cdot e^{-\frac{(\sqrt{\frac{2}{T_s}}y_2 - a_2 A\sqrt{\frac{T_s}{2}}\sin(\theta_2+\phi_i))^2}{N_0}} \end{aligned} \tag{3.7}$$

For convenience in the sequel, the notation $p(\cdot|s_i)$ for conditional probability may be equivalently expressed as $p_i(\cdot)$. Assuming slow fading channels, namely

$a_1 = a_2 \triangleq a$ and $\theta_1 = \theta_2 \triangleq \theta$, Equation (3.7) can be rewritten as

$$p_i(x_1, y_1, x_2, y_2|a, \theta_1) = \frac{4}{\pi^2 N_0^2 T_s^2} \cdot e^{-\frac{\frac{2}{T_s}(x_1^2+y_1^2+x_2^2+y_2^2)+a^2A^2T_s}{N_0}} \cdot e^{\frac{2aA}{N_0}\ell_i \cos(\theta+\varphi)}, \quad (3.8)$$

where

$$\ell_i^2 = (x_1 + x_2\cos\phi_i + y_2\sin\phi_i)^2 + (y_1 - x_2\sin\phi_i + y_2\cos\phi_i)^2, \quad (3.9)$$

and

$$\varphi = \arctan\frac{y_1 - x_2\sin\phi_i + y_2\cos\phi_i}{x_1 + x_2\cos\phi_i + y_2\sin\phi_i}. \quad (3.10)$$

Recognizing θ and a are mutually independent (see Chapter 2) and averaging Eq (3.8) with respect to θ on $[0, 2\pi]$, we have

$$p_i(x_1, y_1, x_2, y_2|a) = \frac{4}{\pi^2 N_0^2 T_s^2} \cdot e^{-\frac{\frac{2}{T_s}(x_1^2+y_1^2+x_2^2+y_2^2)+a^2A^2T_s}{N_0}} \cdot I_0(\frac{2aA}{N_0}\ell_i), \quad (3.11)$$

where $I_0(\cdot)$ is the zeroth order modified Bessel function of the first kind.

For Rayleigh channels, the four random variables x_1, y_1, x_2, y_2 are Gaussian distributed with zero mean; while for Rician channels, these four variables involve $\cos\eta$ or $\sin\eta$, which cause these four variables to be non-Gaussian distributed, and the capacity or the cutoff rates are difficult to evaluate when CSI is not known. Hence, we will focus only on Rayleigh fading channels when CSI is not known; on the other hand, when CSI is available, our discussion will be quite general and can be applied to Rician fading channels.

3.2 Capacity and Cutoff Rates for Slow Fading Channels

In this section, we consider slow fading channels – i.e., channels for which the fading process over two adjacent symbol intervals is unchanged. Using the no-

tation developed above, we have $a_1 = a_2 \triangleq a$ and $\theta_1 = \theta_2 \triangleq \theta$. Later in this chapter, fast fading will be considered and slow fading regarded as a special case.

3.2.1 Channel Capacity

For a memoryless channel with discrete input alphabet $\{s_i, i = 0, 1, ..., M-1\}$ and continuous output taking values in the set Z, the channel capacity C is

$$C = \max_{p(s_i)} \sum_{i=0}^{M-1} p(s_i) \int_Z p(z|s_i) \cdot \log_2 \left\{ \frac{p(z|s_i)}{\sum_{k=0}^{M-1} p(s_k) p(z|s_k)} \right\} dz. \tag{3.12}$$

For DPSK, the signal symmetry indicates that the capacity is achieved when the input distribution is equiprobable. This may be demonstrated by considering a discrete-output version of the channel with quantized outputs; such a quantized DPSK channel is symmetric and thus its capacity is achieved with an equiprobable distribution. As the quantization becomes finer, the channel becomes the DPSK channel and the equiprobable distribution remains optimal. It then follows that

$$C = \log_2(M) - \frac{1}{M} \sum_{i=0}^{M-1} E_{z|s_i} \left\{ \log_2 \frac{\sum_{k=0}^{M-1} p(z|s_k)}{p(z|s_i)} \right\}. \tag{3.13}$$

The symmetry noted above means that the conditional expectation operator $E_{z|s_i}$ is independent of i; thus equation (3.13) may be simplified to

$$C = \log_2(M) - E_{z|s_0} \left\{ \log_2 \frac{\sum_{k=0}^{M-1} p(z|s_k)}{p(z|s_0)} \right\}. \tag{3.14}$$

Note that each channel output z may in fact include more than one variable.

Capacity When CSI is Known

If the CSI is known, the received output is $z = \{x_1, y_1, x_2, y_2, a\}$. By writing

$$p_i(x_1, y_1, x_2, y_2, a) = p_i(x_1, y_1, x_2, y_2|a) \cdot p_i(a), \tag{3.15}$$

we can combine Equation (3.14) and (3.11) to obtain[1]

$$C = \log_2(M) - E_{z|s_0}\left\{\log_2 \frac{\sum_{k=0}^{M-1} I_0(\frac{2aA\ell_k}{N_0})}{I_0(\frac{2aA\ell_0}{N_0})}\right\}. \tag{3.16}$$

The above formula cannot be further simplified; the expectation is therefore obtained by numerical simulation. In following sections where an expectation cannot be simplified, we also resort to simulation to compute the underlying expected value.

Capacity When CSI is Not Known

When the CSI is not available, the output is $z = \{x_1, y_1, x_2, y_2\}$, with conditional distribution

$$p_i(x_1, y_1, x_2, y_2) = \int_0^\infty p_i(x_1, y_1, x_2, y_2|a)p(a)da, \tag{3.17}$$

where $p(a) = 2a \cdot e^{-a^2}$ $(a \geq 0)$ is the Rayleigh density. Using the definite integral

$$\int_0^\infty a \cdot e^{-\alpha a^2} \cdot I_0(\beta a)da = \frac{1}{2\alpha} \cdot e^{\frac{\beta^2}{4\alpha}}, \tag{3.18}$$

Equation (3.17) can be simplified as

$$p_i(x_1, y_1, x_2, y_2) = \frac{4}{\pi^2 N_0^2 T_s^2} \cdot e^{-\frac{2}{N_0 T_s}(x_1^2+y_1^2+x_2^2+y_2^2)} \cdot \frac{N_0}{N_0 + A^2 T_s} \cdot e^{\frac{A^2\ell_i^2}{N_0(N_0+A^2T_s)}}. \tag{3.19}$$

The capacity then becomes

$$C = \log_2(M) - E_{z|s_0}\left\{\log_2 \frac{\sum_{k=0}^{M-1} \exp(\frac{A^2\ell_k^2}{N_0(N_0+2E_s)})}{\exp(\frac{A^2\ell_0^2}{N_0(N_0+2E_s)})}\right\}. \tag{3.20}$$

Recall that the channel capacity is the mutual information between the channel input and output variables, subject to a suitable input distribution. If we

[1]Note: $p_i(a)$ is the same for all $i \in \{0, 1, ..., M-1\}$, so it can be written as $p(a)$.

denote the input as S, the above two capacities can be related by the following chain rule:

$$I(x_1, y_1, x_2, y_2, a; S) = I(x_1, y_1, x_2, y_2; S) + I(a; S|x_1, y_1, x_2, y_2). \tag{3.21}$$

This equation, together with the fact that mutual information is non-negative, indicates that the capacity when CSI is known is no less than the capacity without CSI.

3.2.2 Cutoff Rates

Besides the channel capacity, another quantity of theoretic importance is the channel cutoff rate R_0. R_0 is the largest number such that, for any $R < R_0$, there exists a sequence of codes with rate at least R and increasing blocklength n such that the block error probability $\bar{p_e}$ satisfies [46]

$$\bar{p_e} < 2^{-n(R_0 - R)}.$$

Thus R_0 provides both an achievable rate and performance bound. The cutoff rate is also widely believed to be the rate beyond which it is very expensive to communicate over the channel [47].

Suppose the receiver uses as its decoding metric $m(z, s_i)$ – i.e., it estimates the transmitted signal sequence so as to maximize the cumulative value of this function. If s_i is transmitted, the Chernoff upper bound on the probability that $m(z, s_j) > m(z, s_i)$ is given by [3]

$$C(s_i, s_j, \lambda) = E_{z|s_i}\{\exp(\lambda(m(z, s_j) - m(z, s_i)))\}, \tag{3.22}$$

where λ is to be optimized for the tightest bound. The cutoff rate of a memoryless

channel can be defined in terms of the average Chernoff bound [31], specifically,

$$R_0 = -\log_2 \min_{\lambda} \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} p(s_i)p(s_j)C(s_i, s_j, \lambda). \tag{3.23}$$

For discrete channels with the log-likelihood metric (which achieves maximum likelihood decoding), i.e., $m(z, s_i) = \ln p(z|s_i)$, the optimal λ is $\lambda^* = 1/2$ [49]. Following the approach used for the discrete channel, we can show that the optimal λ is still $\lambda^* = 1/2$ for continuous output channels.

Choosing $\ln p(z|s_i)$ as the metric, we have

$$C(s_i, s_j, \lambda) = \int_Z p(z|s_i)^{1-\lambda} \cdot p(z|s_j)^{\lambda} dz. \tag{3.24}$$

Define the the average Chernoff factor

$$\overline{C(s_i, s_j, \lambda)} = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} p(s_i)p(s_j)C(s_i, s_j, \lambda). \tag{3.25}$$

It is easy to verify that $C(s_i, s_j, \lambda)$ is convex in λ; it follows that $\overline{C(s_i, s_j, \lambda)}$ is still convex in λ. Substituting Equation (3.24) into Equation (3.25), we have

$$\overline{C(s_i, s_j, \lambda)} = \int_Z [\sum_{i=0}^{M-1} p(s_i)p(z|s_i)^{1-\lambda}][\sum_{j=0}^{M-1} p(s_j)p(z|s_j)^{\lambda}]dz. \tag{3.26}$$

Since i and j are just dummy indices, $\overline{C(s_i, s_j, \lambda)}$ is symmetric around $\lambda^* = 1/2$. Combining its convexity and symmetry around $1/2$, we conclude that the optimal value is $\lambda^* = 1/2$ when the log-likelihood metric is used.

In practice, the metric may be achieved from $\ln p(z|s_i)$ by shifting and scaling by constants.[2] For those metrics, the optimal λ is obtained by multiplying the scaling constant by $\lambda^* = 1/2$.

When the metric is not based on the log-likelihood, a close form representation of the optimal λ usually doesn't exist; therefore, numerical methods are required to calculate R_0.

[2] A constant here refers to a value independent of the signals s_i.

As in the previous section, we assume the input distribution is uniform; based on this assumption and the symmetry of input signals, Eq (3.23) is simplified to be

$$R_0 = \log_2(M) - \log_2 \min_{\lambda} \sum_{j=0}^{M-1} C(s_0, s_j, \lambda). \tag{3.27}$$

Cutoff Rate When the CSI is Known

From Eq (3.11) and (3.15), the log-likelihood metric is $m(z, s_i) = \ln I_0(2a\ell_i/N_0)$, which we shall refer to as the **Bessel metric**. By employing this metric, choosing the optimal $\lambda^* = 1/2$ and combining (3.11), (3.15), (3.24), and (3.27), the cutoff rate is expressed as

$$R_0 = \log_2(M) - \log_2 \sum_{j=0}^{M-1} E_{z|s_0} \left\{ \sqrt{\frac{I_0(\frac{2aA\ell_j}{N_0})}{I_0(\frac{2aA\ell_0}{N_0})}} \right\}. \tag{3.28}$$

Cutoff Rate When the CSI is Not Known

When CSI is not available to the receiver, the log-likelihood metric (subject to shifting and scaling) is $m(z, s_i) = \ell_i^2$ (see Eq (3.19)). This metric is referred to as the (unweighted) **square metric**. By multiplying the scaling factor (i.e., $\frac{A^2}{N_0(N_0+A^2T_s)}$) by $\lambda^* = 1/2$, we obtain the optimal value $\lambda^* = A^2/[2N_0(N_0+A^2T_s)]$, and so

$$C(s_i, s_j, \lambda^*) = E_{x1,y1,x2,y2|s_i} e^{\lambda^*(\ell_j^2 - \ell_i^2)} \tag{3.29}$$

Alternatively, we may find the optimal λ by first simplifying the Chernoff factor for arbitrary λ and then finding the optimal λ. In particular, by fixing the fading factor a first, we calculate the Chernoff bound to yield

$$C(s_i, s_j, \lambda) = E_a \left\{ \frac{1}{1 - \frac{N_0^2 T_s^2 \lambda^2}{2}[1 - \cos(\triangle\phi_{j,i})]} \cdot \exp\left(\frac{\lambda T_s^2 a^2 A^2 [1 - \cos(\triangle\phi_{j,i})](T_s N_0 \lambda - 1)}{2(1 - \frac{N_0^2 T_s^2 \lambda^2}{2}[1 - \cos(\triangle\phi_{j,i})])}\right) \right\},$$

(3.30)

where $\triangle\phi_{j,i} = \phi_j - \phi_i$. Note that the above expression doesn't contain θ. Further averaging with respect to a, this expression is simplified to

$$C(s_i, s_j, \lambda) = \frac{1}{1 - \frac{N_0^2 T_s^2 \lambda^2}{2}[1 - \cos(\triangle\phi_{j,i})] + \frac{1}{2}\lambda T_s^2 A^2 (1 - T_s N_0 \lambda)[1 - \cos(\triangle\phi_{j,i})]}. \tag{3.31}$$

By differentiating the denominator of the right side of the above equation, we obtain the optimal $\lambda^* = \frac{A^2}{2N_0(N_0 + A^2 T_s)}$. Substituting the optimal λ into Eq (3.31), the cutoff rate is simplified to become

$$R_0 = \log_2(M) - \log_2 \sum_{j=0}^{M-1} \frac{1}{1 + \frac{E_s^2[1-\cos(\triangle\phi_{j,0})]}{2N_0(N_0+2E_s)}}. \tag{3.32}$$

In the past, (e.g., [30, 48]), a heuristic "Gaussian" metric was used for DPSK and was believed to be sub-optimal. We can show that it is in fact optimal for Rayleigh fading channels when CSI is not known. Define $w_1 = \frac{2}{T_s A}(x_1 + jy_1)$ and $w_2 = \frac{2}{T_s A}(x_2 + jy_2)$; then the Gaussian metric is defined as $m_G(z, s_i) = -|w_1^* w_2 - e^{j\phi_i}|^2$. By combining Eq. (3.9), the Gaussian metric can be written as

$$m_G(z, s_i) = -|w_1^* w_2|^2 - 1 + \frac{4}{T_s^2 A^2} \cdot (\ell_i^2 - U), \tag{3.33}$$

where $U = x_1^2 + y_1^2 + x_2^2 + y_2^2$. Eq. (3.33) implies that the Gaussian metric is equivalent to ℓ_i^2, since one can be obtained from the other by shifting and scaling by constants. Since ℓ_i^2 is the optimal metric for a Rayleigh fading channel when the CSI is not available, it follows that the Gaussian metric is optimal as well. Indeed, the cutoff rates derived from the Gaussian metric (e.g. Eq. (8.15) of [31]) can be shown to be equivalent to Eq. (3.32).

Sub-optimal Metrics When CSI is Known

Since the optimal metric when CSI is known involves the Bessel function and knowledge of N_0, it may not be practical to implement. We therefore introduce two sub-optimal metrics and investigate the resultant cutoff rates.

By applying the two approximations: (1) $\ln I_0(x) \approx x$ for large x, and (2) $\ln I_0(x) \approx x^2/4$ for small x, we propose two sub-optimal metrics: $m_1(z, s_i) = a\ell_i$ and $m_2(z, s_i) = a^2\ell_i^2$. They are referred to as the **weighted linear metric** and the **weighted square metric**, respectively. For the weighted linear metric (i.e., $a\ell_i$), the cutoff rate is

$$R_0 = \log_2(M) - \log_2 \min_{\lambda} \sum_{j=0}^{M-1} E_{x1,y1,x2,y2,a|s_0} e^{\lambda(a\ell_j - a\ell_0)} \tag{3.34}$$

For the weighted square metric (i.e., $a^2\ell_i^2$), the Chernoff bound can be simplified by first averaging with respect to x_1, y_1, x_2, and y_2 under a fixed fading factor a to yield (cf. Equation (3.30))

$$C(s_i, s_j, \lambda) \;=\; E_a \left\{ \frac{1}{1 - \frac{N_0^2 T_s^2 a^4 \lambda^2}{2}[1 - \cos(\triangle\phi_{j,i})]} \cdot \exp\left(\frac{\lambda a^4 T_s^2 A^2 [1 - \cos(\triangle\phi_{j,i})](T_s N_0 a^2 \lambda - 1)}{2(1 - \frac{N_0^2 T_s^2 a^4 \lambda^2}{2}[1 - \cos(\triangle\phi_{j,i})])}\right) \right\} \tag{3.35}$$

The cutoff rate is then determined by substituting the Chernoff bound into Equation (3.27).

3.3 Capacity and Cutoff Rates for Fast Fading Channels

We now assume fast fading – i.e., the fading amplitude and phase vary over adjacent symbols. Let $\theta_\triangle = \theta_2 - \theta_1$ be the phase distortion (also called the random FM) imposed by the channel. In this section, we regard $\theta_\triangle$ as one part of the CSI, along with the amplitude fading factors a_1 and a_2. Practically, the value of $\theta_\triangle$ may be more difficult to estimate or measure than a_1 and a_2; therefore, we will later investigate the performance of systems in which only a_1 and a_2 are known to the receiver.

3.3.1 Channel Capacity

As we did for slow fading channels, we will separate our discussions corresponding to the availability of CSI to the receiver. When CSI is known, our approach here will be similar to that for slow fading channels; we will adopt a different approach to consider the case where CSI is not known.

Capacity When CSI is Known

Defining $\gamma_a = a_2/a_1$ and $\phi_i^* = \phi_i + \theta_\triangle$, Eq. (3.8) can be modified to become

$$p_i(x_1, y_1, x_2, y_2 | a_1, a_2, \theta_1, \theta_\triangle) = \frac{4}{\pi^2 N_0^2 T_s^2} \cdot e^{-\frac{\frac{2}{T_s}(x_1^2+y_1^2+x_2^2+y_2^2)+\frac{(a_1^2+a_2^2)}{2}A^2 T_s}{N_0}} \cdot e^{\frac{2a_1 A}{N_0}\ell_i^* \cos(\theta_1+\varphi^*)}, \quad (3.36)$$

where

$$\ell_i^{*2} = (x_1 + \gamma_a x_2 \cos\phi_i^* + \gamma_a y_2 \sin\phi_i^*)^2 + (y_1 - \gamma_a x_2 \sin\phi_i^* + \gamma_a y_2 \cos\phi_i^*)^2, \quad (3.37)$$

and

$$\varphi^* = \arctan \frac{y_1 - \gamma_a x_2 \sin\phi_i^* + \gamma_a y_2 \cos\phi_i^*}{x_1 + \gamma_a x_2 \cos\phi_i^* + \gamma_a y_2 \sin\phi_i^*}. \tag{3.38}$$

Since θ_1 is uniformly distributed on $[0, 2\pi]$, it follows that

$$p_i(x_1, y_1, x_2, y_2 | a_1, a_2, \theta_\triangle) = \frac{4}{\pi^2 N_0^2 T_s^2} \cdot e^{-\frac{\frac{2}{T_s}(x_1^2+y_1^2+x_2^2+y_2^2)+\frac{(a_1^2+a_2^2)}{2}A^2 T_s}{N_0}} \cdot I_0(\frac{2a_1 A}{N_0}\ell_i^*). \tag{3.39}$$

The channel capacity then becomes

$$C = \log_2(M) - E_{z|s_0}\left\{\log_2 \frac{\sum_{k=0}^{M-1} I_0(\frac{2a_1 A \ell_k^*}{N_0})}{I_0(\frac{2a_1 A \ell_0^*}{N_0})}\right\}, \tag{3.40}$$

where $z = \{x_1, y_1, x_2, y_2, a_1, a_2, \theta_\triangle\}$.

Assuming s_0 is transmitted and substituting Eq. (3.3) and (3.6) into Eq. (3.37), it can be verified that

$$\begin{aligned} \ell_i^{*2} &= [\frac{T_s a_1 A}{2}(\cos\theta_1 + \gamma_a^2 \cos(\theta_1 + \triangle\phi_{0,i})) + n_1]^2 + \\ &\quad [\frac{T_s a_1 A}{2}(\sin\theta_1 + \gamma_a^2 \sin(\theta_1 + \triangle\phi_{0,i})) + n_2]^2, \end{aligned} \tag{3.41}$$

where

$$n_1 = n_{c,1} + \gamma_a(n_{c2}\cos\phi_i^* + n_{s_2}\sin\phi_i^*) \tag{3.42}$$

$$n_2 = n_{s,1} + \gamma_a(n_{s2}\cos\phi_i^* - n_{c_2}\sin\phi_i^*) \tag{3.43}$$

Recognizing that n_1 and n_2 are mutually independent Gaussian variables with mean zero and variance $\frac{N_0 T_s}{4}(1+\gamma_a^2)$, ℓ_i^* is Rician distributed with a parameter $\frac{T_s^2 a_1^2 A^2}{4}(1+\gamma_a^4+2\gamma_a^2\cos(\triangle\phi_{0,i}))$. The statistics of ℓ_i^* are independent of $\theta_\triangle$; in other words, the phase distortion won't affect the receiver's performance as long as it is known.

Capacity When CSI is Not Known

When CSI is not available to the receiver, $p_i(x_1, y_1, x_2, y_2)$ could be obtained by integrating $p_i(x_1, y_1, x_2, y_2|a_1, a_2, \theta_\triangle) \cdot p_i(a_1, a_2, \theta_\triangle)$ as in the slow fading case. However, this approach is not feasible since there is no simple representation of $p_i(a_1, a_2, \theta_\triangle)$. This difficulty can be avoided by considering the amplitude fading factors (a_1 and a_2) and the phase distortion (θ_1 and θ_2) simultaneously. Specifically, let $G_1 = a_1 \cos\theta_1$, $H_1 = a_1 \sin\theta_1$, $G_2 = a_2 \cos\theta_2$, and $H_2 = a_2 \sin\theta_2$. Then G_1 and G_2 are the in-phase components of the fading, while H_1 and H_2 are the quadrature components. For Rayleigh fading channels, these four variables are all Gaussian distributed with mean zero and variance one-half. The in-phase factors are independent of the quadrature factors; however, G_1 and G_2 are correlated with correlation coefficient ξ, as are H_1 and H_2. Note that the slow fading case in the previous section corresponds to $\xi = 1$. With s_i being the transmitted signal, Eq. (3.3) to (3.6) can be written as

$$x_1 = \frac{T_s A}{2} G_1 + n_{c,1} \tag{3.44}$$

$$y_1 = \frac{T_s A}{2} H_1 + n_{s,1} \tag{3.45}$$

$$x_2 = \frac{T_s A}{2} (G_2 \cos\phi_i - H_2 \sin\phi_i) + n_{c,2} \tag{3.46}$$

$$y_2 = \frac{T_s A}{2} (H_2 \cos\phi_i + G_2 \sin\phi_i) + n_{s,2} \tag{3.47}$$

where x_1, y_1, x_2, and y_2 are all Gaussian distributed with zero mean and variance $T_s(E_s + N_0)/4$. The correlation coefficient ξ incurs the following pair-wise correlation among the four output variables $\{x_1, y_1, x_2, y_2\}$:

$$E(x_1 x_2) = \frac{E_s T_s}{4} \xi \cos\phi_i \tag{3.48}$$

$$E(x_1 y_2) = \frac{E_s T_s}{4} \xi \sin\phi_i \tag{3.49}$$

$$E(y_1 x_2) = -\frac{E_s T_s}{4}\xi \sin\phi_i \tag{3.50}$$

$$E(y_1 y_2) = \frac{E_s T_s}{4}\xi \cos\phi_i. \tag{3.51}$$

Denoting $\sigma^2 = \frac{T_s(E_s+N_0)}{4}$ and $\psi = \frac{E_s T_s}{4}\xi$, the correlation matrix of $\{x_1, y_1, x_2, y_2\}$ is

$$\begin{bmatrix} \sigma^2 & 0 & \psi\cos\phi_i & \psi\sin\phi_i \\ 0 & \sigma^2 & -\psi\sin\phi_i & \psi\cos\phi_i \\ \psi\cos\phi_i & -\psi\sin\phi_i & \sigma^2 & 0 \\ \psi\sin\phi_i & \psi\cos\phi_i & 0 & \sigma^2 \end{bmatrix} \tag{3.52}$$

The conditional probability $p_i(x_1, y_1, x_2, y_2)$ can then be shown to be

$$p_i(x_1, y_1, x_2, y_2) = \frac{1}{(4\pi^2)(\sigma^4 - \psi^2)} \cdot e^{-\frac{\sigma^2+\psi}{2(\sigma^4-\psi^2)}(x_1^2+y_1^2+x_2^2+y_2^2)} \cdot e^{\frac{\xi A^2 \ell_i^2}{\frac{1}{4}A^4 T_s^2(1-\xi^2)+N_0(N_0+A^2T_s)}}, \tag{3.53}$$

where ℓ_i has been defined in Eq. (3.9). Unlike ℓ_i^*, the statistics of ℓ_i are affected by the phase distortion $\theta_\triangle$. Note that Eq. (3.53) degenerates into Eq. (3.19) when the correlation coefficient $\xi = 1$.

The capacity becomes

$$C = \log_2(M) - E_{x_1,y_1,x_2,y_2|s_0}\left\{\log_2 \frac{\sum_{k=0}^{M-1} \exp\left(\frac{\xi A^2 \ell_k^2}{\frac{1}{4}A^4 T_s^2(1-\xi^2)+N_0(N_0+A^2T_s)}\right)}{\exp\left(\frac{\xi A^2 \ell_0^2}{\frac{1}{4}A^4 T_s^2(1-\xi^2)+N_0(N_0+A^2T_s)}\right)}\right\}. \tag{3.54}$$

3.3.2 Cutoff Rates

As in the slow fading case, we will investigate the cutoff rates corresponding to the availability of CSI. In addition, we will also consider the case in which the CSI is only partially known; i.e., a_1 and a_2 are known to the receiver but $\theta_\triangle$ is not.

Cutoff Rates When the CSI is Known

The cutoff rate when the CSI is known is similar to the slow fading case, specifically,

$$R_0 = \log_2 M - \log_2 \sum_{j=0}^{M-1} E_{z|s_0} \left\{ \sqrt{\frac{I_0(\frac{2a_1 A\ell_j^*}{N_0})}{I_0(\frac{2a_1 A\ell_0^*}{N_0})}} \right\}. \tag{3.55}$$

Cutoff Rates When The CSI is Not Known

By Equation (3.53), we see that when the CSI is not available, the log-likelihood metric is still (subject to scaling and shifting) ℓ_i^2 regardless of the correlation coefficient ξ. It follows that the "Gaussian metric" is still optimal for fast Rayleigh fading channels.

Using ℓ_i^2 as the metric, the Chernoff factor is

$$C(s_i, s_j, \lambda) = E_{x1,y1,x2,y2|s_i} e^{\lambda(\ell_j^2 - \ell_i^2)}. \tag{3.56}$$

Equation (3.56) can be calculated via the characteristic function of a quadratic form of Gaussian variables. Specifically, defining $\mathbf{r} = [x_1 + jy_1 \;\; x_2 + jy_2]^T$ and

$$\mathbf{s}_i = \begin{bmatrix} 0 & e^{-j\phi_i} \\ e^{j\phi_i} & 0 \end{bmatrix} \tag{3.57}$$

it can be verified that

$$\ell_j^2 - \ell_i^2 = \mathbf{r}^{*^T}(\mathbf{s}_j - \mathbf{s}_i)\mathbf{r}. \tag{3.58}$$

Using the characteristic function of the quadratic form of Gaussian variables derived in [9], the Chernoff factor of Eq. (3.56) is evaluated to be

$$C(s_i, s_j, \lambda) = \frac{1}{1 + \{\frac{\lambda A^2 T_s^2}{2}(\xi - T_s N_0 \lambda) - \frac{\lambda^2 T_s^2 N_0^2}{2}[1 + (1-\xi^2)\frac{A^4 T_s^2}{4N_0^2}]\}[1 - \cos(\triangle\phi_{j,i})]} \tag{3.59}$$

By differentiating the denominator of the above equation with respect to λ, the optimal λ is found to be $\lambda^* = \frac{\xi A^2}{\frac{1}{2}A^4T_s^2(1-\xi^2)+2N_0(N_0+A^2T_s)}$. The cutoff rate is then simplified to

$$R_0 = \log_2 M - \log_2 \sum_{j=0}^{M-1} \frac{1}{1+\frac{\xi^2 A^4 T_s^2}{8N_0(N_0+A^2T_s)+2(1-\xi^2)A^4T_s^2}[1-\cos(\triangle\phi_{j,0})]}. \quad (3.60)$$

Cutoff Rates With Sub-optimal Metrics

We now consider three sub-optimal metrics as we did in the slow fading case. When CSI is known, we choose the weighted linear metrics $a_1\ell_i^*$ and the weighted square metric $a_1^2\ell_i^{*^2}$. The corresponding cutoff rates are

$$R_0 = \log_2(M) - \log_2 \min_{\lambda} \sum_{j=0}^{M-1} E_{z|s_0} e^{\lambda(a_1\ell_j^* - a_1\ell_0^*)} \quad (3.61)$$

and

$$R_0 = \log_2(M) - \log_2 \min_{\lambda} \sum_{j=0}^{M-1} E_{z|s_0} e^{\lambda(a_1^2\ell_j^{*^2} - a_1^2\ell_0^{*^2})}. \quad (3.62)$$

When CSI is not known, we consider the linear metric ℓ_i; the resulting cutoff rate is

$$R_0 = \log_2(M) - \log_2 \min_{\lambda} \sum_{j=0}^{M-1} E_{x1,y1,x2,y2|s_0} e^{\lambda(\ell_j - \ell_0)}. \quad (3.63)$$

Cutoff Rates When CSI is Partially Known

Practically, the amplitude fading factors a_1 and a_2 are easier to estimate than the phase distortion $\theta_\triangle$. We therefore investigate the cutoff rates when a_1 and a_2 are known to the receiver but $\theta_\triangle$ is not. Since $\theta_\triangle$ is not known, it will be treated as zero by the receiver.

If the receiver uses the weighted linear metric, the metric will be $a_1\ell_i^\dagger$, where $a_1\ell_i^\dagger$ is obtained from $a_1\ell_i^*$ by setting $\theta_\triangle = 0$, specifically (see Eq. (3.37)),

$$\ell_i^{\dagger^2} = (x_1 + \gamma_a x_2\cos\phi_i + \gamma_a y_2 \sin\phi_i)^2 + (y_1 - \gamma_a x_2 \sin\phi_i + \gamma_a y_2\cos\phi_i)^2. \quad (3.64)$$

The cutoff rate becomes (for the linear weighted metric)

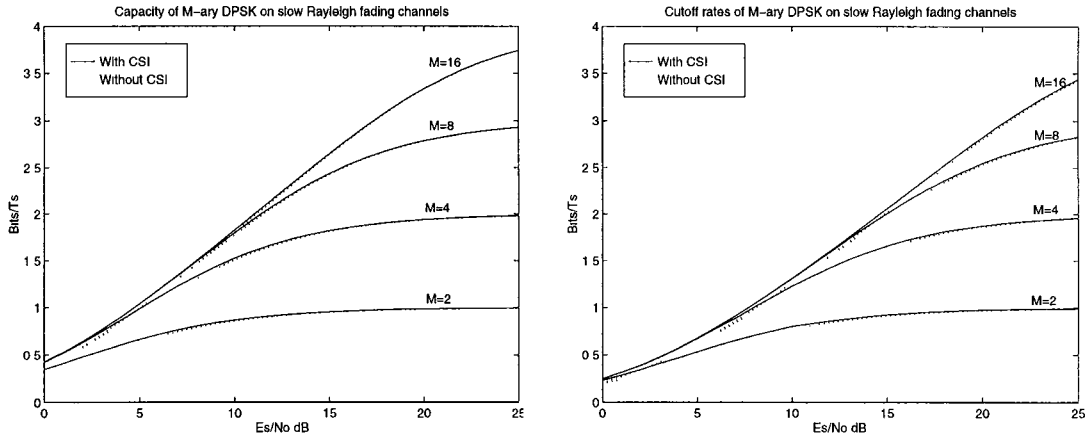
$$R_0 = \log_2 M - \log_2 \min_{\lambda} \sum_{j=0}^{M-1} E_{x1,y1,x2,y2,a_1,a_2|s_0} e^{\lambda(a_1 \ell_j^{\dagger} - a_1 \ell_0^{\dagger})}. \tag{3.65}$$

If the weighted square metric (i.e. $a_1^2 \ell_i^{\dagger 2}$) is used, the cutoff rate expression is obtained by replacing the linear metric of (3.65) by the square metric.

3.4 Numerical Results

In this section, we present numerical results for Rayleigh fading channels. Assume one of the communication terminals is a mobile moving with speed v; the maximum Doppler shift f_D is then obtained as $f_D = v f_c / c$, where c is the speed of light. For a land mobile radio system using an omnidirectional antenna, if the received plane waves arrive at uniform angles, the correlation coefficient ξ can be modeled as $J_0(2\pi f_D T_s)$ (see Chapter 2), where $J_0(\cdot)$ is the zeroth order Bessel function of the first kind. The product $f_D T_s$ is called the normalized fading bandwidth, and will be denoted as b in this chapter. When $b = 0$, $\xi = 1$; this corresponds to slow fading channels. For example, if $v = 60$ mph and $f_c = 900$ MHz, the maximum Doppler shift $f_D = 80$ Hz. For a system with a 5k baud rate, the normalized fading bandwidth b equals 0.016. As will be demonstrated shortly, a channel with $b = 0.016$ still can be treated as a slow fading channel; however, if b increases to 0.05 or beyond, the channel becomes fast fading.

Figure 3.1 depicts the capacities and cutoff rates (with the log-likelihood metric) of slow Rayleigh fading channels for M-ary DPSK. It is observed that knowledge of CSI only minimally affects the capacity and cutoff rates. This is not surprising since the CSI is the carrier amplitude for slow fading cases, while

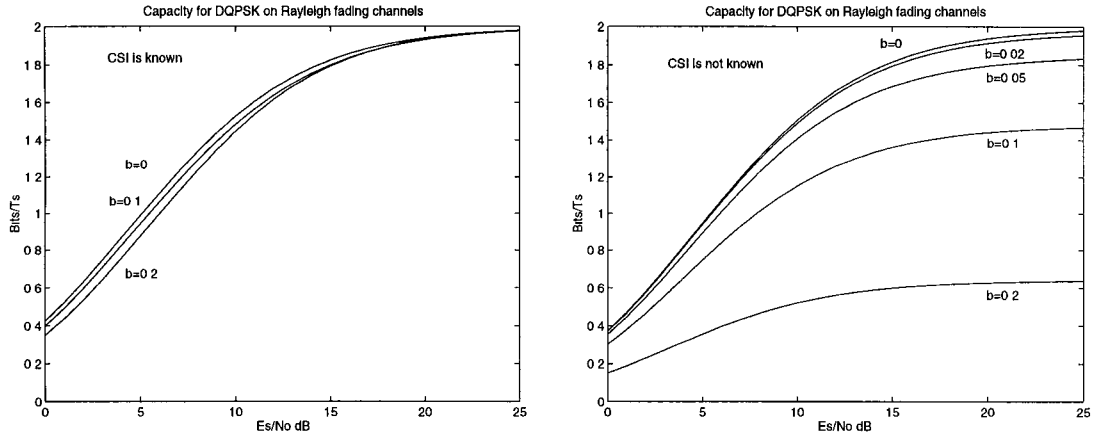


(a) (b)

Figure 3.1: Capacity and cutoff rates of M-ary DPSK on slow Rayleigh fading channels: (a) Capacity; (b) Cutoff rates.

the information is conveyed by a phase change between adjacent symbol signals in DPSK modulations.

We now fix $M = 4$ (i.e., DQPSK) in the following discussion concerning the effects of fast fading and various sub-optimal metrics. Figure 3.2 shows the capacities of DQPSK with or without CSI. We observe that varying b does not dramatically affect the capacity when CSI is known; furthermore, as the SNR increases, the capacity converges to $\log_2(M) = 2$. By contrast, the capacity when CSI is not known degrades significantly as b increases and may converge to a value less than $\log_2(M)$. This phenomenon can be explained by Eq. (3.40) and Eq. (3.54); when CSI is known and N_0 approaches zero (or SNR becomes very large), ℓ_k^* converges to $\sqrt{\frac{T_s^2 a_1^2 A^2}{4}(1 + \gamma_a^4 + 2\gamma_a^2 \cos(\triangle\phi_{0,k}))}$, where $k = 0, 1, 2..., M-1$. If s_0 is transmitted, $\ell_0^* > \ell_k^*$ with probability approaching 1 for any $k \neq 0$. Since SNR is very small, we may approximate $I_0(\frac{2a_1 A\ell_k^*}{N_0})$ in Eq. (3.40) by $\exp(\frac{2a_1 A\ell_k^*}{N_0})$. Using this approximation, and the fact that $\ell_0^* > \ell_k^*$



(a) (b)

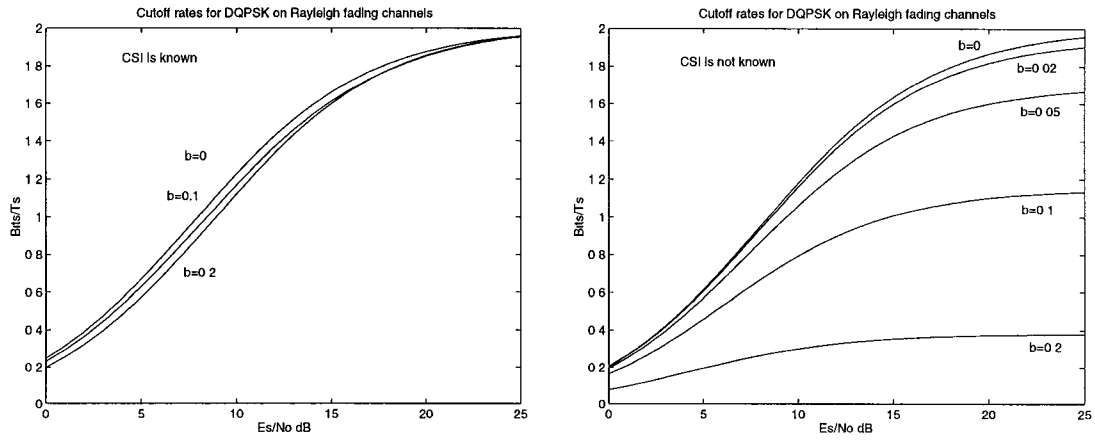
Figure 3.2: Capacity of DQPSK on Rayleigh fading channels: (a) with CSI; (b) without CSI.

with probability approaching 1, the term $\frac{\sum_{k=0}^{M-1} I_0(\frac{2a_1 A\ell_k^*}{N_0})}{I_0(\frac{2a_1 A\ell_0^*}{N_0})}$ in Eq. (3.40) approaches one. Therefore, the capacity approaches $\log_2(M)$ as SNR grows. In contrast, when the CSI is not known, the statistics of ℓ_k in (3.54) are affected by θ_Δ. As a result, the probability that $\ell_0 < \ell_k$ for some $k \neq 0$ does **not** converge to zero, and so the capacity saturates at a value less than $\log_2(M)$.

Figure 3.3 depicts the cutoff rates when the log-likelihood metric is used. Again, the cutoff rates for known CSI do not differ by much for different b and will approach $\log_2(M)$ at very large SNR; for unknown CSI, the cutoff rates saturate at a value less than $\log_2 M$. Specifically, as N_0 goes to 0, Eq. (3.60) degenerates to

$$R_0 = \log_2(M) - \log_2 \sum_{j=0}^{M-1} \frac{1}{1 + \frac{\xi^2}{2(1-\xi^2)}[1 - \cos(\triangle\phi_{j,0})]} \tag{3.66}$$

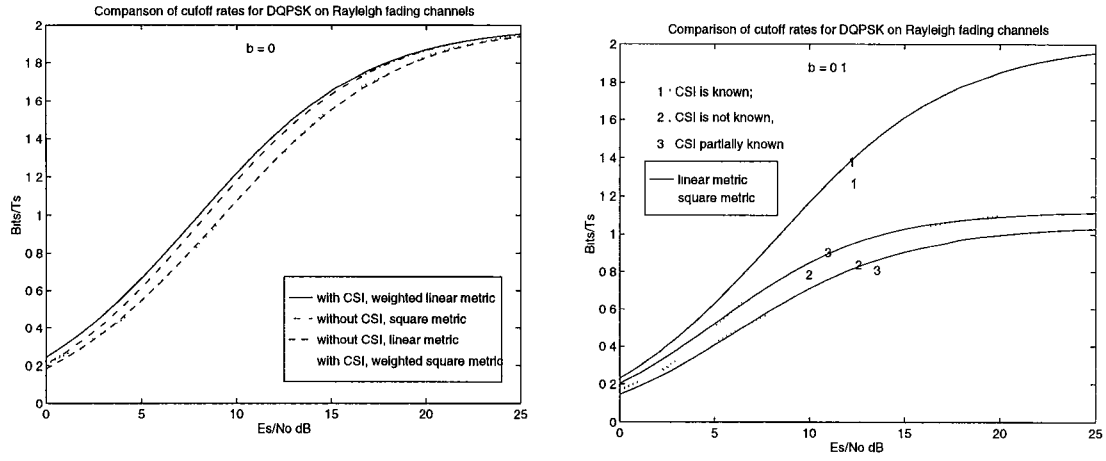
This is the cutoff rate limit at very large SNR for unknown CSI. Apparently, this limit is decided by ξ.



(a) (b)

Figure 3.3: Cutoff rates of DQPSK with log-likelihood metrics on Rayleigh fading channels: (a) with CSI; (b) without CSI.

Figure 3.4 depicts the cutoff rates for different sub-optimal metrics in slow fading (Figure a) and fast fading (Figure b). For convenience, we use linear/square metrics to refer to both weighted and unweighted metrics in (b); this shouldn't cause confusion since the metrics are weighted when the carrier amplitudes are known and unweighted when they're not known. When CSI is known, we find that the cutoff rates of the weighted linear metric are almost the same as that of the optimal Bessel metric; if we were to plot the cutoff rate corresponding to the Bessel metric, it would lie on top of the curve corresponding to the weighted linear metric. However, if other sub-optimal metrics are employed, the cutoff rates may degrade significantly. For example, if the weighted square metric is used when CSI is known, the cutoff rates may be degraded by two dB for moderate SNR. In short, the weighted linear metric is near-optimal when CSI is known, while the square metric is best (for Rayleigh fading channels) when CSI is not known. The same conclusion is also observed in part (b) of



(a) (b)

Figure 3.4: Cutoff rates for various metrics: (a) b = 0; (b) b = 0.1

Figure (3.4) where $b = 0.1$. The cutoff rates for partial knowledge of CSI are also shown in this figure; it is observed that the weighted linear metric is better than the weighted square metric when only a_1 and a_2 are known to the receiver. Furthermore, the performance of the weighted linear metric for partial CSI is comparable (or slightly better at lower to moderate SNR range) to the optimal (i.e., square) metric when CSI is not known. In other words, it pays to estimate the carrier amplitude even when $\theta_\triangle$ is not available.

Figure 3.5 presents a comparison of the Rayleigh fading channels against the AWGN channels in terms of cutoff rates. It can be seen that channel fading significantly degrades the channel cutoff rates (or capacity). When $\theta_\triangle$ is 0 (i.e., for AWGN channels or slow fading channels), cutoff rates are not significantly affected by the availability of CSI as long as good metrics are used. For fast fading channels, if the CSI is known, the cutoff rates are very close to the one of slow fading channel; however, if the CSI is not known, the cutoff rates will degrade significantly even the best metric is used. For this reason, CSI is very

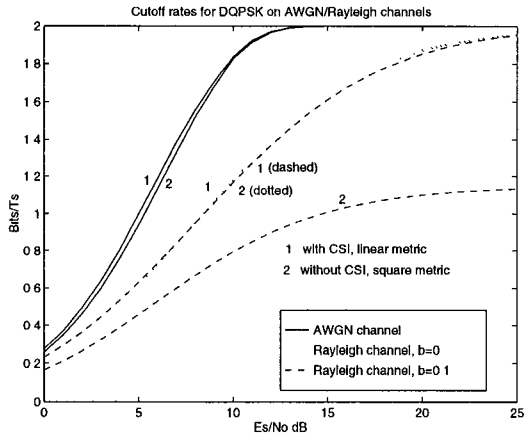


Figure 3.5: Cutoff rate comparison of Rayleigh fading channels with AWGN channels

important for DPSK systems when the channels suffer fast fading. As the carrier frequency becomes higher and higher (e.g., in the GHz range), the normalized fading bandwidth will not become negligible even if the mobile does not move at high speeds; in this circumstance, the CSI becomes a key element for the success of DPSK systems. (Of course, as the fading becomes faster, it becomes more difficult to estimate CSI reliably.)

3.5 Summary

We have derived and calculated the capacity and cutoff rates of fading channels for DPSK modulation. When CSI is known to the receiver, the capacity or the cutoff rate won't be affected much by the correlation coefficient of the fading process; these two figures will approach $\log_2 M$ at large SNR. On the other hand, lack of CSI may degrade the capacity and the cutoff rate significantly when the channel suffers from fast fading; these two figures will saturate at a

value less than $\log_2(M)$ at large SNR. This phenomenon reflects the "error floor" effects of DPSK when the demodulation/decoding error rates are calculated. The error floor is the limit of the decoding error probability as SNR become infinite. Clearly, knowledge of CSI can be employed to lower the error floor.

In the literature, a "Gaussian metric" is consistently employed as a sub-optimal metric; we show that this metric is in fact optimal (i.e., equivalent to the log-likelihood metric) for slow/fast Rayleigh fading channels when CSI is not known.

The method of determining the cutoff rates not only yields the cutoff rates, but it also provides insight into a metric's performance. For example, the weighted linear metric yields almost the same cutoff rates as those determined by the log-likelihood Bessel metric. Therefore, the weighted linear metric can be regarded as the best practical sub-optimal metric. Other sub-optimal metrics can be evaluated the same way without resorting to any specific coding scheme.

Chapter 4

Performance of Various Soft-Decision Decoding Metrics for DPSK on AWGN and Rayleigh Fading Channels

In the previous chapter, we introduced optimal and suboptimal metrics for DPSK based on whether the channel state information (CSI) is available or not. This chapter studies the properties and performance of those metrics more closely. Since quaternary DPSK (DQPSK) is currently the most popular DPSK scheme, a great portion of this chapter is devoted to this particular modulation.

All the metrics in the previous chapter are based on **symbol interleaving**, i.e., a modulation symbol ($\log_2 M$ bits) of a codeword is the basic unit for interleaving and $\log_2 M$ bits from the same codeword are conveyed each time a modulated signal is transmitted. In contrast, in a **bit interleaving** scheme, a bit is the basic interleaving unit, and $\log_2 M$ bits – each bit from a different codeword – are conveyed by a modulated signal. Such an interleaving scheme may be preferable to a symbol interleaving scheme in a fading environment because it avoids the burstiness of errors that occurs when $\log_2 M$ bits of the same

codeword are conveyed with a single phase change.

In this chapter, we develop bit metrics suitable for systems employing bit interleaving. As we did for symbol metrics, we will define weighted/unweighted linear and square bit metrics[1] and compare them with those used in the literature [41, 50]. For DQPSK, we show that (1) with dibit interleaving, the weighted/unweighted square dibit metric and the weighted/unweighted square bit metric are in fact equivalent, respectively; (2) dibit interleaving is better than bit interleaving on AWGN channels, whereas the opposite is true on Rayleigh fading channels.

4.1 Soft-Decision Decoding Metrics and Algorithms

In Chapter 1, we saw that the optimal metric (i.e., the metric achieving ML decoding) for a symbol b_j is derived from $\log(r_j|b_j)$, where r_j is the received signal. This metric is appropriate when symbol interleaving is used. However, there are some situations in which either the above metric cannot be calculated or it cannot be used directly. This difficulty typically occurs when a modulation symbol does not correspond to a codeword element. For example, if a binary codeword is **bit interleaved** and transmitted by an M-ary DPSK scheme (where $M > 2$), the optimal symbol metric is difficult to define and may be inappropriate to use. If symbol interleaving is employed for this binary code and $\log(r_j|b_j)$ calculated, the metric still may be inappropriate if the underlying soft-decision decoding

[1] As in the previous chapter, a metric is a weighted metric when CSI is known; otherwise, it is an unweighted metric.

algorithm requires bit metrics. For all these difficulties, there are basically two classes of solutions:

- If the optimal metric cannot be calculated, define a suboptimal (but calculable) metric. In this chapter, the (binary) extended Golay code will be used; when the codeword is bit interleaved and transmitted by DQPSK, we will define suboptimal bit metrics. Once bit metrics are defined, they can also be calculated under dibit interleaving.

- If an existing soft-decision decoding algorithm uses bit metrics whereas the demodulator provides symbol metrics, the decoding algorithm can be modified in order to utilize symbol metrics. In this chapter, when the Golay code is dibit interleaved and transmitted via DQPSK, we will modify the soft-decision decoding algorithm of [21], which requires bit metrics, so that dibit metrics can be used.

4.2 Symbol Metrics for DPSK

In this section, we briefly review and examine the various optimal and suboptimal DPSK symbol metrics developed in the previous chapter. These metrics are useful only when symbol interleaving is employed.

For convenience, Eq.(3.44)-(3.47) are re-stated here for the matched filter outputs x_1, y_1, x_2, and y_2 given that $i \in \{0, 1, ..., \log_2 M - 1\}$ is the transmitted symbol:

$$x_1 = \frac{T_s A}{2} G_1 + n_{c,1} \tag{4.1}$$

$$y_1 = \frac{T_s A}{2} H_1 + n_{s,1} \tag{4.2}$$

$$x_2 = \frac{T_s A}{2}(G_2 \cos\phi_i - H_2 \sin\phi_i) + n_{c,2} \tag{4.3}$$

$$y_2 = \frac{T_s A}{2}(H_2 \cos\phi_i + G_2 \sin\phi_i) + n_{s,2}. \tag{4.4}$$

where $G_1 = a_1 \cos\theta_1$, $H_1 = a_1 \sin\theta_1$, $G_2 = a_2 \cos\theta_2$, and $H_2 = a_2 \sin\theta_2$. Suppose the received waveform includes line-of-sight as well as multipath components, the four variables $\{G_1, H_1, G_2, H_2\}$ can be decomposed as

$$G_1 = \alpha\cos(\eta) + g_1 \tag{4.5}$$

$$H_1 = \alpha\sin(\eta) + h_1 \tag{4.6}$$

$$G_2 = \alpha\cos(\eta) + g_2 \tag{4.7}$$

$$H_2 = \alpha\sin(\eta) + h_2, \tag{4.8}$$

where α represents the line-of-sight component, and $\{g_1, g_2, h_1, h_2\}$ (which are four zero-mean Gaussian variables) represent the multipath components. Letting K_r be the Rician factor (i.e., the ratio of line-of-sight power to multipath signal power), the normalization assumption (i.e., $E[a_1^2] = E[a_2^2] = 1$) on the fading factors requires that $\alpha = \sqrt{\frac{K_r}{1+K_r}}$ and each of the four variables $\{g_1, g_2, h_1, h_2\}$ has variance $\frac{1}{2(1+K_r)}$. We assume the in-phase pair (g_1, g_2) are independent of the quadrature pair (h_1, h_2); however, g_1 and g_2 are correlated with a correlation coefficient ξ, and so are h_1 and h_2. As a special case, $K_r = 0$ corresponds to Rayleigh fading channels, whereas $K_r = \infty$ corresponds to pure AWGN channels.

The four variables $\{x_1, y_1, x_2, y_2\}$ are the received signals. When CSI is not known, the decoding metric will be computed solely from these variables, and the optimal symbol metric for symbol i on Rayleigh fading channels is ℓ_i^2, where

$$\ell_i^2 = (x_1 + x_2 \cos\phi_i + y_2 \sin\phi_i)^2 + (y_1 - x_2 \sin\phi_i + y_2 \cos\phi_i)^2. \tag{4.9}$$

In terms of continuous waveforms, ℓ_i^2 can be equivalently written as

$$\begin{aligned}\ell_i^2 = \; & [\int_0^{T_s} r(t)\cos(2\pi f_c t)dt + \int_{T_s}^{2T_s} r(t)\cos(2\pi f_c t + \phi_i)dt]^2 + \\ & [\int_0^{T_s} r(t)\sin(2\pi f_c t)dt + \int_{T_s}^{2T_s} r(t)\sin(2\pi f_c t + \phi_i)dt]^2. \end{aligned} \tag{4.10}$$

Clearly, ℓ_i^2 is the squared envelope output from an envelope detector, where one filter is matched to $\cos(2\pi f_c t)$ in $[0, T_s]$ and $\cos(2\pi f_c t + \phi_i)$ in $[T_s, 2T_s]$; and the other filter is matched to $\sin(2\pi f_c t)$ in $[0, T_s]$ and $\sin(2\pi f_c t + \phi_i)$ in $[T_s, 2T_s]$.

The metric ℓ_i^2 is called the (unweighted) **square metric**. For $M = 2$ (i.e., DBPSK), the square metric is equivalent to the traditional decision variable metric in [51]. In Chapter 3, it was also shown that the square metric ℓ_i^2 is in fact equivalent to the "Gaussian metric" employed in the literature. Since the square operation is monotonic, we also investigate the (unweighted) **linear metric** ℓ_i.

When CSI is known to the receiver, the metrics will be defined from the four variables $\{x_1, y_1, x_2, y_2\}$ combined with the CSI. We assume the CSI includes the two amplitude fading factors $\{a_1, a_2\}$ as well as the phase distortion $\theta_\triangle$ (called random FM) defined as $\theta_\triangle = \theta_2 - \theta_1$. Introducing the two quantities $\gamma_a = a_2/a_1$ and $\phi_i^* = \phi_i + \theta_\triangle$, the optimal metric is found to be $\ln I_0(2a_1 A\ell_i^*/N_0)$, where $I_0(\cdot)$ is the 0^{th} order modified Bessel function of the first kind and ℓ_i^* is defined by

$$\ell_i^{*^2} = (x_1 + \gamma_a x_2 \cos\phi_i^* + \gamma_a y_2 \sin\phi_i^*)^2 + (y_1 - \gamma_a x_2 \sin\phi_i^* + \gamma_a y_2 \cos\phi_i^*)^2. \tag{4.11}$$

This optimal metric is called the **Bessel metric**. Note that this metric is optimal not only for Rayleigh fading channels but also for Rician and unfaded AWGN channels when CSI is known. Eq. (4.11) indicates that the amplitude fading factors $\{a_1, a_2\}$ and $\theta_\triangle$ are incorporated into the metric. When $a_1 = a_2$ and

$\theta_\triangle = 0$, the channel is called slow fading; otherwise, it is fast fading. Clearly, ℓ_i^* degenerates into ℓ_i for slow fading channels.

The Bessel metric is somewhat inconvenient to implement because it requires the knowledge of N_0; hence, two approximations are investigated for easier implementation. Specifically, based on the value of the Bessel function argument $2a_1 A\ell_i^*/N_0$, two approximations are made to obtain two sub-optimal metrics: (1) the weighted linear metric $a_1\ell_i^*$ from the large argument approximation, and (2) the weighted square metric $a_1^2\ell_i^{*^2}$ from the small argument approximation. For the Neyman-Pearson multiple pulse detection problem, the linear approximation has been shown to be slightly better than the square approximation on AWGN channels unless the diversity order (i.e., the number of pulses) is very large [52]; in our simulation to be shown later, the weighted linear metric is also better. The value of the argument $\frac{2Aa_1\ell_i^*}{N_0}$ can be roughly investigated by the following example. Assuming $a_1 = a_2 = 1$ and $\theta_\triangle = 0$, then $\ell_i^* = \ell_i$ and $A^2\ell_i^2$ is a noncentral χ^2 random variable with mean $[2(Es/N_0) + 4(Es/N_0)^2]N_0^2$. Using $\mathrm{E}[\ell_i^2]$ to approximate $(\mathrm{E}[\ell_i])^2$, and assuming $Es/N_0 = 5$ (which corresponds to 7 dB), $2\mathrm{E}[A\ell_i]/N_0 = 21$, which can be regarded as a large number and justify the large-argument approximation.

4.3 Bit Metrics for DQPSK

Bit metrics are of great interest in many situations, e.g., when the underlying soft-decision decoding algorithm requires bit metrics, or when bit interleaving is performed so that only bit metrics can be defined.

We now define bit metrics for DQPSK in a way similar to that in which

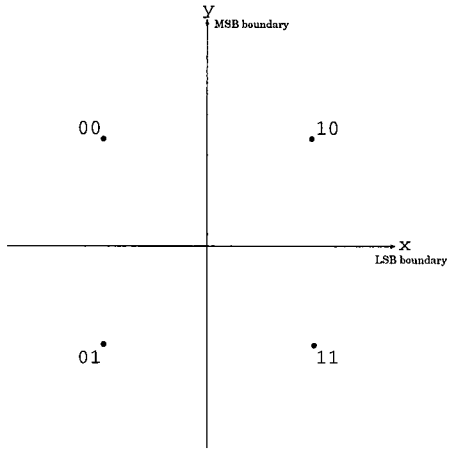


Figure 4.1: The DQPSK signals

symbol metrics are defined. We also discuss how the widely used decision variable bit metric corresponds to our metrics.

Let's consider a DQPSK scheme in which ϕ_0, ϕ_1, ϕ_2, and ϕ_3 are $\frac{3\pi}{4}$, $-\frac{3\pi}{4}$, $\frac{\pi}{4}$, and $-\frac{\pi}{4}$ respectively[2]. The signal set is plotted in Figure 4.1, where the x-axis is the decision boundary for the LSB (least significant bit), and the y-axis is the decision boundary for the MSB (most significant bit). We consider the LSB only, and the MSB can be treated similarly.

If the phase change $(\tan^{-1}\frac{y_2}{x_2} - \tan^{-1}\frac{y_1}{x_1})$ is between $[0, \pi]$, the demodulator will decide LSB=0; otherwise, LSB=1. This phase change formulation can be simplified to yield the simpler decision variable $x_1y_2 - x_2y_1$ [53]. If this decision variable is positive, LSB=0; otherwise, LSB=1. Therefore, the absolute value of this decision variable has usually been defined as the LSB metric in the literature, e.g., in [50]. To take the fading factors into consideration, the decision variable $x_1y_2 - x_2y_1$ was heuristically weighted by $\sqrt{a_1a_2}$ in [41].

[2]This scheme is usually called $\pi/4$-shift DQPSK.

Interestingly, bit metrics can be defined more systematically in a manner similar to the manner in which symbol (or dibit in this case) metrics were defined in the previous section. If we imagine LSB = 0 is conveyed by a phase change $\phi_{0,lsb} = \pi/2$, and LSB = 1 is conveyed by a phase change $\phi_{1,lsb} = -\pi/2$, Eq. (4.11) becomes

$$\begin{aligned} \ell^{*^2}_{0,lsb} &= (x_1 + \gamma_a x_2 \cos(\pi/2 + \theta_\triangle) + \gamma_a y_2 \sin(\pi/2 + \theta_\triangle))^2 + \\ &\quad (y_1 - \gamma_a x_2 \sin(\pi/2 + \theta_\triangle) + \gamma_a y_2 \cos(\pi/2 + \theta_\triangle))^2, \end{aligned} \tag{4.12}$$

and

$$\begin{aligned} \ell^{*^2}_{1,lsb} &= (x_1 + \gamma_a x_2 \cos(-\pi/2 + \theta_\triangle) + \gamma_a y_2 \sin(-\pi/2 + \theta_\triangle))^2 + \\ &\quad (y_1 - \gamma_a x_2 \sin(-\pi/2 + \theta_\triangle) + \gamma_a y_2 \cos(-\pi/2 + \theta_\triangle))^2. \end{aligned} \tag{4.13}$$

If we consider slow fading - i.e., $\gamma_a = 1$ and $\theta_\triangle = 0$ - then $\ell^{*^2}_{0,lsb}$ and $\ell^{*^2}_{1,lsb}$ degenerate into $\ell^2_{0,lsb}$ and $\ell^2_{1,lsb}$, respectively. Like symbol metrics, bit metrics can be defined as weighted/unweighted linear metrics or weighted/unweighted square metrics; for example, $a_1 \ell^*_{0,lsb}$ is the weighted linear bit metric, while $\ell^2_{0,lsb}$ is the (unweighted) square bit metric.

If the square bit metric is used, the demodulator decides LSB=0 when $\ell^2_{0,lsb} > \ell^2_{1,lsb}$; otherwise LSB=1. By appropriate constant offset and scaling, the above inequality (i.e., $\ell^2_{0,lsb} > \ell^2_{1,lsb}$) can be rewritten as $x_1 y_2 - x_2 y_1 > 0$, which matches the decision variable formula of [50]. If the weighted square metric is used and $\theta_\triangle = 0$, the demodulator decides LSB=0 when $a_1 a_2 (x_1 y_2 - x_2 y_1) > 0$. Thus, if $\theta_\triangle = 0$, the weighted bit metric is equivalent to the weighted decision variable metric, where the weighting factor is $a_1 a_2$ instead of $\sqrt{a_1 a_2}$ of [41].

Since our bit metrics are derived from **fictitious** phase change, one may think that the resulting bit metrics should be inferior to dibit metrics, which

are derived from **true** phase change. Surprisingly, it can be shown that when dibit interleaving is employed, the weighted/unweighted square dibit metrics are in fact equivalent to the weighted/unweighted square bit metrics, respectively. We will return to this issue shortly. However, this equivalence doesn't hold for linear metrics. Nonetheless, linear bit metrics and linear dibit metrics yield very close performance under dibit interleaving. In short, under dibit interleaving, dibit metrics and bit metrics yield the same or similar performance. On the other hand, only bit metrics are defined under bit interleaving. We will show that dibit interleaving is better than bit interleaving for AWGN channels, and the opposite is true for Rayleigh fading channels.

We now show that the weighted (unweighted) square bit metrics and the weighted (unweighted) square dibit metrics are equivalent, respectively, under dibit interleaving. Since unweighted metrics are just special cases of weighted metrics by setting $a_1 = 1$, $\gamma_a = 1$ and $\theta_\triangle = 0$, we need only consider weighted metrics.

Assume dibit 00 is transmitted by the DQPSK scheme specified above, and the receiver wants to decide whether 00 or 01 was transmitted. Under dibit metrics, a decision error happens when $\ell_{00}^{*2} < \ell_{01}^{*2}$; under bit metrics, a decision error happens when $\ell_{0,lsb}^{*2} < \ell_{1,lsb}^{*2}$. Similarly, if the receiver wants to decide whether 00 or 10 was transmitted, a decision error happens when $\ell_{00}^{*2} < \ell_{10}^{*2}$ under dibit metrics, and a decision error happens when $\ell_{0,msb}^{*2} < \ell_{1,msb}^{*2}$ under bit metrics. If the receiver wants to decide whether 00 or 11 was transmitted, a decision error happen when $\ell_{00}^{*2} < \ell_{11}^{*2}$ under dibit metrics, and a decision error happens when when $\ell_{0,lsb}^{*2} + \ell_{0,msb}^{*2} < \ell_{1,lsb}^{*2} + \ell_{1,msb}^{*2}$ under bit metrics. By algebraic manipulation of Eq. (4.11), (4.12) and (4.13), we can show the following results.

Note that only addition and subtraction are used to obtain these results; no multiplication is involved.

- $\ell_{0,lsb}^{*^2} < \ell_{1,lsb}^{*^2}$ can be rewritten as $\sqrt{2}\ell_{00}^{*^2} < \sqrt{2}\ell_{01}^{*^2}$.
- $\ell_{0,msb}^{*^2} < \ell_{1,msb}^{*^2}$ can be rewritten as $\sqrt{2}\ell_{00}^{*^2} < \sqrt{2}\ell_{10}^{*^2}$.
- $\ell_{0,lsb}^{*^2} + \ell_{0,msb}^{*^2} < \ell_{1,lsb}^{*^2} + \ell_{1,msb}^{*^2}$ can be rewritten as $\sqrt{2}\ell_{00}^{*^2} < \sqrt{2}\ell_{11}^{*^2}$.

Since the same scaling factor $\sqrt{2}$ happens in each of the above three cases, we conclude that the weighted square dibit metric is equivalent to the weighted square bit metric when the receiver is to distinguish between n-tuple 0 and any other n-tuple. Although we have assumed n-tuple 0 is transmitted, this assumption is only for convenience and can be dropped.

We notice that none of the values of a_1, γ_a, and $\theta_\triangle$ affect the equivalence property mentioned above; hence, the equivalence property also holds even when the receiver uses incorrect (or estimated) CSI parameters in the metrics.

4.4 Analytical Error Performance of the (Unweighted) Metrics for DBPSK and DQPSK

In this section, we analyze the error performance of the unweighted metrics for DBPSK and DQPSK. On AWGN channels, where the weighted metrics degenerate into unweighted metrics, we will analyze the performance of both the linear and square metrics; on Rayleigh fading channels, where the unweighted square metric is optimal provided CSI is not available, we only analyze the unweighted square metric. The performance of other metrics not analyzed in this section will

be investigated by simulation results in the next section. The pair-wise decoding error probabilities between two codewords will be derived first; then the union bound is obtained as the summation of all pair-wise error probabilities.

Without loss of generality, assume the all-zero codeword $\mathbf{c}_0$ is transmitted. If the decoder estimates that the codeword $\hat{\mathbf{c}}$ was transmitted, the decoding codeword error probability is $P_{err} = \text{Prob}(\hat{\mathbf{c}} \neq \mathbf{c}_0)$. Suppose $\mathbf{c}_h$ is any codeword, i.e., $\mathbf{c}_h = [b_0, b_1, \ldots, b_{n'-1}]$, and denote m_{b_j} as the metric for b_j, i.e., $m_{b_j} = \ell_{b_j}$ under linear metrics and $m_{b_j} = \ell_{b_j}^2$ under square metrics. Defining $U_h = \sum_{j=0}^{n'-1} m_{b_j}$, the pair-wise error probability for $\mathbf{c}_0$ and $\mathbf{c}_h$ $(h \neq 0)$ is $P_e = \text{Prob}(U_0 < U_h)$ and the union bound is expressed as

$$P_{err} \leq \sum_{h \neq 0} \text{Prob}(U_0 < U_h). \tag{4.14}$$

With the zero codeword transmitted, we replace ϕ_i in Eq. (3.44) through (3.47) with ϕ_0 (i.e., ϕ_0 being the **true** phase change), and substitute (3.44) through (3.47) into (3.9) to obtain:

$$\ell_i^2 = R_i^2 + I_i^2, \tag{4.15}$$

where

$$\begin{aligned} R_i &= \frac{T_s A}{2}\sqrt{\frac{K_r}{1+K_r}}\cos(\eta) + \frac{T_s A}{2} g_1 + n_{c,1} + \frac{T_s A}{2}\sqrt{\frac{K_r}{1+K_r}}\cos(\eta + \triangle\phi_i) + \\ &\quad \frac{T_s A}{2} g_2 \cos(\triangle\phi_i) - \frac{T_s A}{2} h_2 \sin(\triangle\phi_i) + n_{c,2}\cos(\phi_i) + n_{s,2}\sin(\phi_i), \end{aligned} \tag{4.16}$$

$$\begin{aligned} I_i &= \frac{T_s A}{2}\sqrt{\frac{K_r}{1+K_r}}\sin(\eta) + \frac{T_s A}{2} h_1 + n_{s,1} + \frac{T_s A}{2}\sqrt{\frac{K_r}{1+K_r}}\sin(\eta + \triangle\phi_i) + \\ &\quad \frac{T_s A}{2} h_2 \cos(\triangle\phi_i) + \frac{T_s A}{2} g_2 \sin(\triangle\phi_i) - n_{c,2}\sin(\phi_i) + n_{s,2}\cos(\phi_i), \end{aligned} \tag{4.17}$$

and $\triangle\phi_i = \phi_0 - \phi_i$. If bit metrics for DQPSK are considered, symbol i in the three equations above is replaced by a bit; e.g., ℓ_i^2 is replaced by $\ell_{0,lsb}^2$ and ϕ_i is replaced by $\phi_{0,lsb}$.

4.4.1 Performance of Binary DPSK

For DBPSK, we assume $\phi_0 = 0$ and $\phi_1 = \pi$. For convenience, define two complex Gaussian random variables:

$$X = R_0 + iI_0 \tag{4.18}$$

$$Y = R_1 + iI_1. \tag{4.19}$$

So far, we have been concentrating on the time interval $[0, 2T_s]$; in order to consider other time intervals, we append the index j with X and Y to form the notation X_j and Y_j. In particular,

$$\ell^2_{b_j=0} = |X_j|^2, \tag{4.20}$$

and

$$\ell^2_{b_j=1} = |Y_j|^2, \tag{4.21}$$

are the square metrics associated with "0" and "1", respectively, at the j^{th} bit position.

Linear metric on AWGN Channels

Letting k index the non-zero bits of $\mathbf{c}_h$, we have

$$U_0 - U_h = \sum_k (|X_k| - |Y_k|). \tag{4.22}$$

The pair-wise error probability is given by $P_e = \text{Prob}(\sum_k (|Y_k| - |X_k|) > 0)$. Substituting $K_r = \infty$ into Eq. (4.16) and (4.17), it can be verified that $|X_k|$ is a Rician variable and $|Y_k|$ a Rayleigh variable. Furthermore, $|X_k|$ and $|Y_k|$ are mutually independent. As such, the pair-wise error probability is the complementary probability distribution function of a sum (or difference) of independent

Rayleigh and Rician random variables. The complementary probability of the sum of Rayleigh random variables was addressed in [57]; we next employ the same approach (i.e., the infinite series method) in [57] to address the Rician random variables.

Strictly, the indices k's may not be consecutive integers; however, for convenience, we assume $k \in \{0, 1, 2, ..., L-1\}$. This assumption clearly doesn't cause any loss of generality. Define $2L$ variables $Z_0, Z_1, ..., Z_{2L-1}$ such that, for $k \in \{0, 1, ..., L-1\}$, $Z_k = |Y_k|$ and $Z_{k+L} = -|X_k|$. The pair-wise error probability becomes $P_e = \text{Prob}(\sum_{i=0}^{2L-1} Z_i > 0)$. Let $W = \sum_{i=0}^{2L-1} Z_i$ and T be a positive number large enough such that $\text{Prob}(W < -T/2)$ can be neglected, P_e can be written as [57]

$$P_e = \frac{1}{2} + \frac{2}{\pi} \sum_{\substack{n=1 \\ n:\, odd}} \frac{A_n \sin\theta_n}{n}, \tag{4.23}$$

where

$$A_n = \prod_{i=0}^{2L-1} A_{i,n}$$

$$\theta_n = \sum_{i=0}^{2L-1} \theta_{i,n}$$

$$w = \frac{2\pi}{T}$$

$$A_{i,n} = \sqrt{\{E[\cos(nwZ_i)]\}^2 + \{E[\sin(nwZ_i)]\}^2}$$

$$\theta_{i,n} = \tan^{-1}\left\{\frac{E[\sin(nwZ_i)]}{E[\cos(nwZ_i)]}\right\}.$$

From Eq. (4.23), P_e is easily obtained if both $E[\cos(nwZ_i)]$ and $E[\sin(nwZ_i)]$ are known, where Z_i may be a Rayleigh (for $i \in \{0, 1, ..., L-1\}$) or a **negative** Rician variable (for $i \in \{L, L+1, ..., 2L-1\}$).

For a Rayleigh variable Z with the probability density $\frac{z}{\sigma^2}e^{-z^2/2\sigma^2}$, it was shown [57] that

$$E[\cos(nwZ)] = {}_1F_1(1, \frac{1}{2}, \frac{-n^2w^2\sigma^2}{2}), \tag{4.24}$$

$$E[\sin(nwZ)] = \sqrt{\frac{\pi}{2}}nw\sigma e^{-n^2w^2\sigma^2/2}, \tag{4.25}$$

where ${}_1F_1(\cdot,\cdot,\cdot)$ is the confluent hypergeometric function [54] defined as

$${}_1F_1(a,b,x) = 1 + \frac{ax}{b} + \frac{a(a+1)x^2}{b(b+1)2!} + \frac{a(a+1)(a+2)x^3}{b(b+1)(b+2)3!} + ... \tag{4.26}$$

If Z is a Rician variable with probability density $\frac{z}{\sigma^2}e^{-(z^2+\beta^2)/2\sigma^2}I_0(\frac{\beta z}{\sigma^2})$, we have (see Appendix A):

$$E[\cos(nwZ)] = 1 + \sum_{\substack{m=2 \\ m:\ even}} (-1)^{m/2}\frac{(nw\sqrt{2}\sigma)^m}{m\cdot(m-1)...(m/2+1)}{}_1F_1(-m/2;1;-\alpha^2/2) \tag{4.27}$$

and

$$E[\sin(nwZ)] = \frac{nw\sqrt{2}\sigma}{2}\Gamma(\frac{1}{2}){}_1F_1(-m/2;1;-\alpha^2/2) + \sum_{\substack{m=3 \\ m:\ odd}} (-1)^{(m-1)/2}\frac{(nw\sqrt{2}\sigma)^{(m-1)/2}}{2\cdot 4\cdot 6...(m-1)}(nw\sqrt{2}\sigma)^{(m+1)/2}\Gamma(\frac{1}{2}){}_1F_1(-m/2;1;-\alpha^2/2) \tag{4.28}$$

where

$$\alpha^2 = \frac{\beta^2}{\sigma^2}.$$

For DBPSK on AWGN channels, it is easily verified that $\sigma^2 = T_sN_0/2$ and $\beta^2 = T_s^2A^2$. Substituting (4.24) through (4.28) into (4.23), P_e is obtained. Note that the negative Rician variable won't cause further difficulty, since, for $Z' = -Z$, we have $E[\cos(nwZ')] = E[\cos(nwZ)]$ and $E[\sin(nwZ')] = -E[\sin(nwZ)]$.

Square Metrics on AWGN and Rayleigh Channels

For square metrics, the pair-wise decoding error probability is

$$p_e = Prob(\sum_k (|X_k|^2 - |Y_k|^2) < 0$$

This is a special case of Appendix 4B of [3]. For convenience, Appendix B of this thesis defines a set of parameters to be used later based on the notation of [3]. Besides the notation definition, Appendix B also lists three equations from [3] which will be referred to in this chapter.

Based on Eq. (4.16) and (4.17), we obtain the following parameters defined in Appendix B:

$|\bar{X}_k|^2 = (T_s A)^2 \cdot \frac{K_r}{1+K_r}$

$|\bar{Y}_k|^2 = 0$

$\mu_{XX} = \frac{T_s^2 A^2}{2(1+K_r)} + T_s N_0 + \xi \frac{T_s^2 A^2}{2(1+K_r)}$

$\mu_{YY} = \frac{T_s^2 A^2}{2(1+K_r)} + T_s N_0 - \xi \frac{T_s^2 A^2}{2(1+K_r)}$

$\mu_{XY} = 0$

$w = \frac{\xi \frac{T_s^2 A^2}{2(1+K_r)}}{(\frac{T_s^2 A^2}{2(1+K_r)} + T_s N_0)^2 - \xi^2 (\frac{T_s^2 A^2}{2(1+K_r)})^2}$

$\alpha_{1k} = (T_s A)^2 \cdot \frac{K_r}{1+K_r} \cdot (\frac{T_s^2 A^2}{2(1+K_r)} + T_s N_0 - \xi \frac{T_s^2 A^2}{2(1+K_r)})$

$\alpha_{2k} = (T_s A)^2 \cdot \frac{K_r}{1+K_r}$

$v_1 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY}}} - w$

$v_2 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY}}} + w$

These parameters will in turn yield the parameters α_1, α_2, a, and b, which will appear in the error probability formula.

A. Square Metrics on AWGN Channels

Setting $K_r = \infty$ corresponds to AWGN channels. Denoting L as the Hamming weight of $\mathbf{c}_h$, we obtain $a = 0$ and $b = \sqrt{\frac{LA^2 T_s}{N_0}}$. Usually, Eq. (B.3) in

Appendix B can be used to describe the pair-wise error probability we need; however, with $a = 0$, that equation cannot be readily applied because the term $I_n(ab) \cdot (\frac{b}{a})^n$ in that equation is not defined for $a = 0$, where $I_n(\cdot)$ is the n^{th} order modified Bessel function of the first kind. To circumvent the difficulty, we employ this equality

$$I_n(ab) \cdot (\frac{b}{a})^n = \frac{1}{2\pi i} \int_\Gamma \frac{1}{p^{n+1}} \exp(\frac{a^2}{2}\frac{1}{p} + \frac{b^2}{2}p)dp; \tag{4.29}$$

and, making a approach 0, obtain

$$\lim_{a \to 0} I_n(ab) \cdot (\frac{b}{a})^n = \frac{1}{2\pi i} \int_\Gamma \frac{1}{p^{n+1}} \exp(\frac{b^2}{2}p)dp. \tag{4.30}$$

Note Γ is a circular contour enclosing the original point in the complex plane. The circular integral can be performed by applying the Cauchy integral formula to obtain:

$$\lim_{a \to 0} I_n(ab) \cdot (\frac{b}{a})^n = \frac{b^{2n}}{2^n \cdot n!} \tag{4.31}$$

Also noting that $Q(0, b) = \exp(-b^2/2)$, $I_0(0) = 1$, and $\sum_{k=0}^{L-1} \binom{2L-1}{k} = 2^{2L-2}$, we simplify Eq. (B.3) to obtain the following pairwise error probability:

$$P_e = e^{-b^2/2} \left\{ \frac{1}{2} + \frac{1}{2^{2L-1}} \sum_{n=1}^{L-1} \left[\frac{b^{2n}}{2^n \cdot n!} \cdot \sum_{k=0}^{L-1-n} \binom{2L-1}{k} \right] \right\}. \tag{4.32}$$

As a special case with $L = 1$, (4.32) is simplified as $P_e = \frac{1}{2}\exp(-E_b/N_0)$, which is identical to Eq. (7.3.9) of [3]. Based on the probability $\text{Prob}(U_0 < U_h)$, the union bound is obtained.

B. Square Metrics on Rayleigh Fading Channels

Setting $K_r = 0$ corresponds to Rayleigh fading channels. In this case, $\alpha_1 = \alpha_2 = a = b = 0$. In this case, we exploit Eq. (B.2), i.e.,

$$P_e = \frac{1}{(1 + v_2/v_1)^{2L-1}} \cdot \frac{1}{2\pi i} \int_\Gamma \frac{[1 + (v_2/v_1)p]^{2L-1}}{p^L(1-p)} dp. \tag{4.33}$$

Since Γ is a circular contour of radius less than one, Eq. (4.33) only involves one pole — $p = 0$. By residue theory, $P_e = \frac{\text{Res}_0}{(1+v_2/v_1)^{2L-1}}$, where Res_0 is the residue of $\frac{[1+(v_2/v_1)p]^{2L-1}}{p^L(1-p)}$ at $p = 0$, and it is the coefficient of the p^{-1} term when $\frac{[1+(v_2/v_1)p]^{2L-1}}{p^L(1-p)}$ is expanded as a Laurent series. By using this equality

$$\frac{1}{1-p} = \sum_{k=0}^{\infty} p^k, \tag{4.34}$$

Res_0 equals the coefficient of the p^{L-1} term in the Taylor series expansion of $\frac{[1+(v_2/v_1)p]^{2L-1}}{1-p}$ with respect to p, and is evaluated to be $\sum_{k=0}^{L-1} \binom{2L-1}{k} (\frac{v_2}{v_1})^k$. Therefore,

$$P_e = \frac{1}{(1+v_2/v_1)^{2L-1}} \cdot \sum_{k=o}^{L-1} \binom{2L-1}{k} (\frac{v_2}{v_1})^k. \tag{4.35}$$

When $L = 1$ and $\xi = 1$, we have

$$v_1 = \frac{1}{T_s(N_0 + A^2T_s)}, \quad v_2 = \frac{A^2T_s + N_0}{N_0T_s(N_0 + A^2T_s)}$$

and $P_e = \frac{1}{2(E_s/N_0+1)}$, which is identical to Eq. (7.3.10) of [3]. This probability is also identical to the result of [56]. The metric for DPSK in [56] was obtained assuming perfect estimation of ξ and SNR; however, that metric can be verified to be equivalent to the heuristic "Gaussian metric", which in turn is equivalent to our unweighted square metric and does not require the knowledge of ξ and SNR.

4.4.2 Performance of Quaternary DPSK With Dibit Interleaving

We adopt the particular DQPSK scheme specified in the previous section, and consider the performance under dibit interleaving[3]. We are interested in the

[3]We have already shown that the square bit metric and square dibit metric are equivalent under dibit interleaving

performance of dibit metrics. However, as we will see shortly, it is difficult to analyze the performance of dibit metrics (for both linear and square metrics) on AWGN channels; to circumvent this difficulty, we will use a mixture of dibit and bit metrics in a codeword to analyze the performance of DQPSK on AWGN channels. By doing so, the obtained pair-wise error probability could be larger than the one when only dibit metrics are used. Nevertheless, the union bound so obtained is still quite tight (approximately 0.2 dB away from the simulated error probability at high SNR), as will be shown later. For Rayleigh fading channels, though, the error probabilities of dibit metrics can be analyzed without using a mixture of dibit and bit metrics.

With dibit metrics, the indices k's now refer to nonzero dibits. Define X_k and Y_k as follows:

$$X_k = R_{00} + iI_{00} \tag{4.36}$$

and

$$Y_k = R_{b_k} + iI_{b_k} \tag{4.37}$$

where b_k may be 01, 10 or 11.

Linear Metrics on AWGN Channels

With linear metrics, the pair-wise error probability is $P_e = \text{Prob}(\sum_k |Y_k| - |X_k| > 0)$. Unless $b_k = 11$, X_k and Y_k won't be mutually independent, making an analytical solution of P_e under dibit metrics very difficult. (Note that the infinite series method in [57] is valid only for independent X_k and Y_k.)

The reason that $\mu_{X_k Y_k} = 0$ for $b_k = 11$ is the assumption that $|\phi_{00} - \phi_{11}| = \pi$. When $b_k = 01$ or 10, however, X_k and Y_k will be correlated. To avoid this correlation, we choose bit metrics for $b_k = 01$ or 10. By employing bit metrics

for 01 and 10, the bit metric value for the "0" in 01 and 10 is the same as the corresponding bit metric value in 00, so only the non-zero bit metric needs to be considered. Since $\phi_{0,lsb} - \phi_{1,lsb} = \pi$ (for $b_k = 01$), the bit metrics for the LSB will be statistically independent. By doing so, we can again employ the infinite series method to calculate the pair-wise decoding error probability.

Let k_2 be the index of the "1" in dibit 01 or 10 of $\mathbf{c}_h$, and X_{k_2} and Y_{k_2} be defined in terms of bit metrics. On the other hand, let k_1 be the index of dibit 11, and X_{k_1} and Y_{k_1} be defined in terms of dibit metrics. The task now becomes to calculate

$$Prob(\sum_{k_1}(|Y_{k_1}| - |X_{k_1}|) + \sum_{k_2}(|Y_{k_2}| - |X_{k_2}|) > 0). \tag{4.38}$$

Here, $|Y_{k_1}|$ is Rayleigh distributed, while $|X_{k_1}|$, $|Y_{k_2}|$ and $|X_{k_2}|$ are Rician distributed. Eq. (4.38) can be evaluated by the infinite series technique described earlier.

Strictly speaking, using dibit and bit metrics at different positions of the same codeword may degrade the decoding performance, and may yield a looser upper bound. Nevertheless, the resulting upper bound (to be shown later) is still tight enough to justify our approach.

Square Metrics on AWGN and Rayleigh Channels

For $b_k = 11$, the set of parameters (like μ_{XX}, μ_{YY}, μ_{XY}, etc) derived from X_k and Y_k are of the same form as those in the DBPSK case, and have been shown in Section 4.1; for $b_k = 01$, we have the following parameters: (Note that the following parameters are for $b_k = 01$ only. In order to distinguish it from the $b_k = 11$ case, we have changed the notation v_1 and v_2 into u_1 and u_2 for $b_k = 01$.)
$|\bar{X}_k|^2 = (T_s A)^2 \cdot \frac{K_r}{1+K_r}$

$$|\bar{Y}_k|^2 = \frac{(T_s A)^2}{2} \cdot \frac{K_r}{1+K_r}$$

$$\mu_{XX} = \frac{(T_s A)^2}{2(1+K_r)} + T_s N_0 + \frac{\xi T_s^2 A^2}{2(1+K_r)}$$

$$\mu_{YY} = \frac{(T_s A)^2}{2(1+K_r)} + T_s N_0$$

$$\mu_{XY} = (1+\xi)\frac{T_s^2 A^2}{2} \cdot \frac{1}{2(1+K_r)}(1+i)$$

$$w = \frac{\mu_{XX} - \mu_{YY}}{2(\mu_{XX}\mu_{YY} - |\mu_{XY}|^2)}$$

$$\alpha_{1k} = |\bar{X}_k|^2 \mu_{YY} + |\bar{Y}_k|^2 \mu_{XX} - \bar{X}_k{}^* \bar{Y}_k \mu_{XY} - \bar{X}_k \bar{Y}_k{}^* \mu_{XY}^*$$

$$\alpha_{2k} = |\bar{X}_k|^2 - |\bar{Y}_k|^2$$

$$u_1 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY} - |\mu_{XY}|^2}} - w$$

$$u_2 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY} - |\mu_{XY}|^2}} + w$$

The parameters for $b_k = 10$ are almost the same, except μ_{XY} is the conjugate of that for $b_k = 01$. As such, the resulting α_{1k}, α_{2k}, u_1, and u_2 are the same.

A. Square Metrics on AWGN Channels

Setting $K_r = \infty$, it is clear that $\mu_{X_k Y_k}$ may not be the same for all k; hence, Eq. (B.3) are not applicable. Specifically, $\mu_{X_k Y_k} = 0$ for $b_k = 11$, but $\mu_{X_k Y_k} \neq 0$ for $b_k = 01$ or $b_k = 10$. To circumvent this difficulty, we follow the earlier approach of using dibit metric for $b_k = 11$ and bit metric for $b_k = 01$ or 10. We then have

$$P_e = Prob(\sum_{k_1}(|X_{k_1}|^2 - |Y_{k_1}|^2) + \sum_{k_2}(|X_{k_2}|^2 - |Y_{k_2}|^2) < 0). \tag{4.39}$$

Eq. (4.39) is calculated by formulations of [3] by first determining those parameters such as v_1, v_2, α_1, α_2, a, and b. The parameter a is not 0 as long as there exist dibit 01 or 10 in $\mathbf{c}_h$. If this is the case, Eq. (B.3) can be simplified to

$$P_e = Q(a,b) - \frac{I_0(ab)}{2} e^{-\frac{a^2+b^2}{2}} + \frac{e^{-\frac{a^2+b^2}{2}}}{2^{2L-1}} \sum_{n=1}^{L-1} I_n(ab) \left\{ \sum_{k=0}^{L-1-n} \binom{2L-1}{k} \left[(\frac{b}{a})^n - (\frac{a}{b})^n \right] \right\} \tag{4.40}$$

For those codewords that do not contain dibit 01 or 10, only dibit metrics are used and the pair-wise probability is expressed by Eq. (4.32).

B. Dibit Square Metrics on Rayleigh Fading Channels

For Rayleigh fading channels, not only $\mu_{X_kY_k}$ may not be the same for all k, but $\mu_{Y_kY_k}$ may not be the same either. Specifically, $\mu_{Y_kY_k} = (1-\xi)T_s^2A^2/2+T_sN_0$ for $b_k = 11$, but $\mu_{Y_kY_k} = T_s^2A^2/2 + T_sN_0$ for $b_k = 01$ or $b_k = 10$. With a modification to Eq. (B.1), the pair-wise decoding probability can be expressed as

$$P_e = -\frac{(v_1v_2)^{L_1}(u_1u_2)^{L_2}}{2\pi i}\int_{-\infty+i\varepsilon}^{\infty+i\varepsilon}\frac{dz}{z(z+iv_1)^{L_1}(z-iv_2)^{L_1}(z+iu_1)^{L_2}(z-iu_2)^{L_2}}, \tag{4.41}$$

where L_1 and L_2 are the numbers of dibit 11 and 01 (or 10), respectively, in the codeword $\mathbf{c}_h$, and ε is a small positive number. Defining

$$f(z) = \frac{1}{z(z+iv_1)^{L_1}(z-iv_2)^{L_1}(z+iu_1)^{L_2}(z-iu_2)^{L_2}}, \tag{4.42}$$

the integral contour of $\int_{-\infty+i\varepsilon}^{\infty+i\varepsilon} f(z)dz$ can be extended to an upper semicircle connecting the two ends of the line $[-\infty+i\varepsilon, \infty+i\varepsilon]$, since the integral over the infinitely large semicircle is 0. Therefore,

$$P_e = -(v_1v_2)^{L_1}(u_1u_2)^{L_2}\cdot(\mathrm{Res}_{v_2}+\mathrm{Res}_{u_2}), \tag{4.43}$$

where Res_{v_2} and Res_{u_2} are the residues of $f(z)$ for the poles iv_2 and iu_2, respectively. A similar procedure using the two-sided Laplace transform was presented in [56] for calculating the pair-wise error probability of trellis-coded modulation on Rayleigh fading channels, where the fading factors are estimated by the receiver and the estimated fading factors are assumed to be jointly Gaussian distributed with the actual fading factors.

4.4.3 Performance of Quaternary DPSK With Bit interleaving

We now investigate the performance of bit metrics of DQPSK under bit interleaving. For simplicity, we only consider square metrics.

With bit interleaving, k indexes the nonzero bits of $\mathbf{c}_h$. Without loss of generality, we assume all the nonzero bits are the LSB bit in a modulation symbol (dibit). As mentioned earlier, $\phi_{0,lsb} = \pi/2$ and $\phi_{1,lsb} = -\pi/2$. After some algebra, we have the following parameters:

$$|\bar{X}_k|^2 = \frac{(T_s A)^2}{2} \cdot \frac{K_r}{1+K_r}(1 + \cos\tfrac{\pi}{4})$$

$$|\bar{Y}_k|^2 = \frac{(T_s A)^2}{2} \cdot \frac{K_r}{1+K_r}(1 - \cos\tfrac{\pi}{4})$$

$$\mu_{XX} = (1 + \xi\cos\tfrac{\pi}{4})\frac{(T_s A)^2}{2(1+K_r)} + T_s N_0$$

$$\mu_{YY} = (1 - \xi\cos\tfrac{\pi}{4})\frac{(T_s A)^2}{2(1+K_r)} + T_s N_0$$

$$\mu_{XY} = iT_s^2 A^2 \cdot \xi\frac{1}{2(1+K_r)}\sin\tfrac{\pi}{4}$$

$$w = \frac{\xi\frac{(T_s A)^2}{2(1+K_r)} \cdot \frac{\sqrt{2}}{2}}{(\frac{(T_s A)^2}{2(1+K_r)} + T_s N_0)^2 - \frac{\xi^2}{2}\frac{(T_s A)^2}{2(1+K_r)})^2 - \frac{1}{2}(T_s^2 A^2 \xi\frac{1}{2(1+K_r)})^2}$$

$$\alpha_{1k} = |\bar{X}_k|^2\mu_{YY} + |\bar{Y}_k|^2\mu_{XX} - \bar{X}_k{}^*\bar{Y}_k\mu_{XY} - \bar{X}_k\bar{Y}_k{}^*\mu_{XY}^*$$

$$\alpha_{2k} = (T_s A)^2\frac{K_r}{1+K_r} \cdot \cos\tfrac{\pi}{4}$$

$$v_1 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY} - |\mu_{XY}|^2}} - w$$

$$v_2 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY} - |\mu_{XY}|^2}} + w$$

A. Square Bit Metrics on AWGN Channels

With $K_r = \infty$, $v_1 = v_2$ and $a \neq 0$; the pair-wise error probability is expressed by Eq. (4.40).

B. Square Bit Metrics on Rayleigh Fading Channels

With $K_r = 0$, we have $a = b = 0$; the pair-wise error probability is expressed by Eq. (4.35).

4.5 Simulation Results

In this section, we present simulation results to compare various DPSK metrics mentioned in the previous sections. The upper bounds that were developed in the previous section will also be included to evaluate their tightness. The error probabilities will be plotted as a function of the **channel** signal-to-noise ratio. Specifically, the energy required to transmit each **codeword bit** is given by $E_c = E_s/\log_2 M$. The signal-to-noise ratio (SNR) used in this section (as well as in later chapters) is defined as E_c/N_0. Another commonly used notation for the energy to transmit an **information bit** is E_b. For a binary (n, k) code, we have $nE_c = kE_b$.

The (24,12,8) extended Golay code will be used throughout our simulation. When dibit metrics are used, we modify the ML decoding algorithm in [21] so that dibit metrics can be applied. The algorithm in [21] uses the "Hexacode" structure of the (24,12) Golay code - i.e., a $(6, 3)$ code over GF(4). In particular, this algorithm first maps every four bits of a Golay codeword into a Hexacode symbol over $GF(4)$, decodes the Hexacode, and finally interprets the output Hexacode codeword back to a Golay codeword. Since four bits are two dibits, we can modify the algorithm in order to use dibit confidence without much difficulty.

Since the simulation results are presented in terms of bit error rates, the upper bound for codeword error rates derived in the previous section must be converted to bit error rates for fair comparison. For our Golay encoding operation, the average Hamming weight of the 12 information bits which are encoded to a Golay codeword of Hamming weight 8 (which is the minimal Hamming weight of the extended Golay code) is 4. Since these minimum-weight codewords dominate

the error events, the bit error rate is estimated as 1/3 of the codeword error rate. (Our simulation results show that this estimation is extremely accurate.) Accordingly, the BER upper bound is derived as 1/3 of the upper bound for codeword error probabilities.

4.5.1 On AWGN Channels

We first present our simulation results for binary signaling - i.e., DBPSK - on AWGN channels. Figure 4.2 compares the bit-error-rate (BER) performance of the (optimal) Bessel metric, the linear metric, and the square metric. (For AWGN channels, weighted and unweighted metrics are the same.) This figure

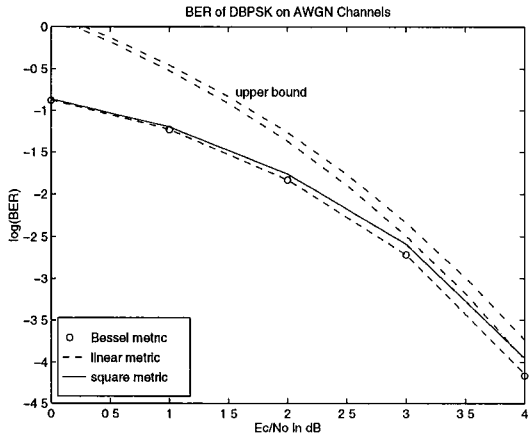


Figure 4.2: BER performance of DBPSK on AWGN channels

indicates that the Bessel metric and the linear metric yields virtually the same performance, and the square metric is slightly inferior; hence, the linear metric can be regarded as the practical optimal choice over practical SNR ranges. The two upper bound curves correspond to the linear metric (the lower curve) and the square metric (the upper curve). We notice that the difference between the

two upper bounds is very close to the difference between the simulated results.

We now begin to show the simulation results for DQPSK. Figure 4.3 compares the performance of the bit metric and the dibit metric under dibit interleaving for DQPSK on AWGN channels. For square dibit and square bit metrics, they are equivalent; for the linear dibit and bit metrics, these two metrics are not the same but yield very close performance. As in DBPSK, the two linear metrics are better than the square metric. Of the two upper bound curves in this figure,

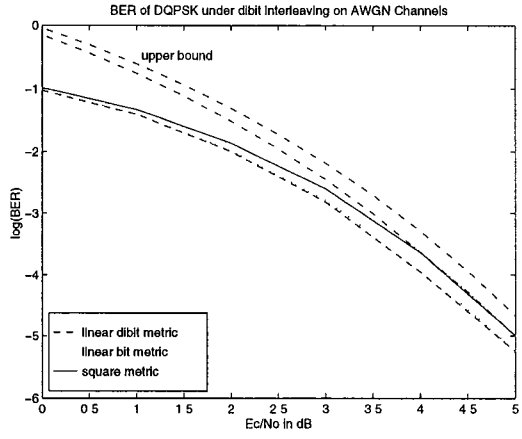
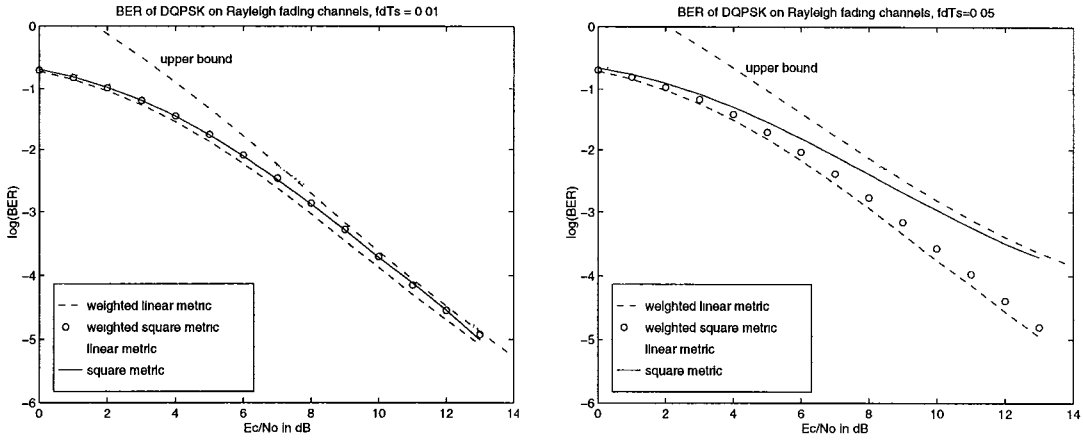


Figure 4.3: BER performance of DQPSK over AWGN channels

the upper one corresponds to square metrics and the lower one corresponds to linear metrics. These two curve are tight (approximately 0.2 dB away from the simulation results at moderate to high SNR), which indicates that the simultaneous use of dibit and bit metrics in the same codeword only minimally degrade the decoding performance.

4.5.2 On Rayleigh Fading Channels

Figure 4.4 depicts the error probabilities on Rayleigh fading channels with DQPSK and dibit interleaving. The upper bounds shown in this figure correspond to bounds for the (unweighted) square metric. As in Chapter 3, we assume the



(a) (b)

Figure 4.4: BER performance of DQPSK over Rayleigh fading channels.

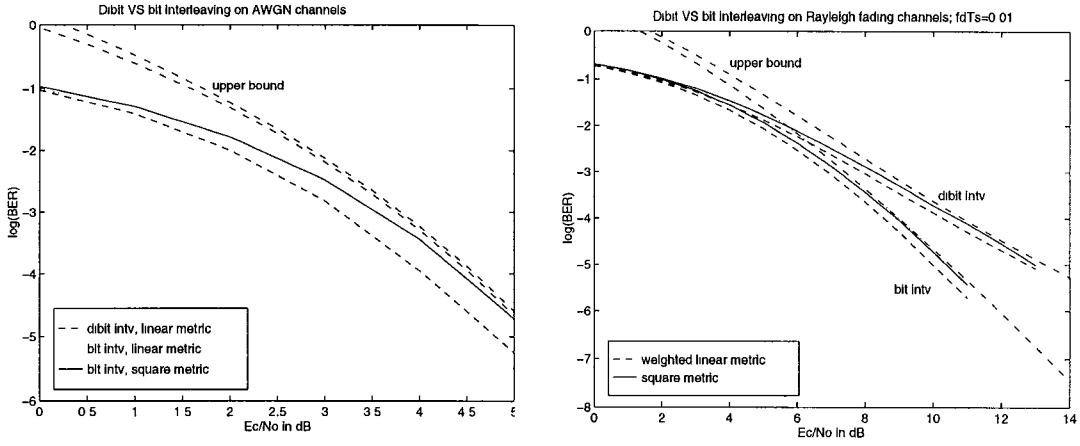
correlation coefficient $\xi = J_0(2\pi f_D T_s)$. Part (a) of Figure 4.4 corresponds to the normalized fading factor $f_D T_s = 0.01$, while part (b) corresponds to $f_D T_s = 0.05$. (If the carrier frequency $f_c = 900$ MHz and the transmission rate is 5000 symbols per second, a normalized fading factor 0.01 (0.05) correspond to a 37.5 (187.5) mph vehicle speed, respectively. If the carrier frequency increases to 2.25 GHz, a normalized fading factor 0.05 correspond to a 75 mph vehicle speed.) From (a), we observe that the weighted linear metric (when CSI is known) yields the best performance, whereas the (unweighted) linear metric yields the worst. This figure also shows that the (unweighted) square metric (that is optimal when CSI is unknown) yields a performance similar to the weighted linear metric. We also

observe that the upper bound for the square metric is very tight from moderate to high SNR. In calculating the upper bounds, we only consider the 859 codewords with the minimal Hamming weight for simpler computation of the residue values; this will have negligible effect on the bounds at higher SNR. In part (b), where the fading becomes faster, we observe that the weighted metrics are much better than the unweighted metrics; as such, the CSI is important on fast fading channels. This observation is consistent with the conclusions of Chapter 3 in terms of the channel capacity and cutoff rates.

4.5.3 Bit vs Symbol Interleaving

Figure 4.5 presents the comparison between dibit and bit interleaving for DQPSK on AWGN as well as on Rayleigh fading channels. All metrics used in this figure are bit metrics. (Note that bit metrics can be calculated under both bit and dibit interleaving.) For AWGN channels, part (a) of Figure. 4.5 shows that dibit interleaving is (slightly) better than bit interleaving. Of the two upper bound curves in part (a) of this figure, the upper one corresponds to square metrics with bit interleaving, and the lower one corresponds to square metrics with dibit interleaving. On the other hand, we observe from part (b) that bit interleaving is much better than dibit interleaving on Rayleigh fading channels. Since dibit interleaving treats two bits as a single unit, channel fading will cause a burst error of two bits; on the contrary, bit interleaving can randomize all errors. Of the two upper bound curves in part (b), the upper one corresponds to square metrics with dibit interleaving, and the lower one corresponds to square metrics with bit interleaving.

It is worthwhile to emphasize that this phenomenon (i.e., bit interleaving



(a) (b)

Figure 4.5: Dibit vs bit interleaving: (a) on AWGN channels; (b) on Rayleigh fading channels ($f_D T_s = .01$).

is better than dibit interleaving on Rayleigh fading channels) holds for other channel codes. To see it, let's consider the pair-wise error probability between 00 and b_k. If b_k is 01 or 10, it can be shown (by almost the same procedure used in Section 3) that dibit interleaving yields the same performance as bit interleaving. But if $b_k = 11$, this is not true. For dibit interleaving, the pair-wise error (between 00 and 11) probability is expressed by Eq. (4.41) with $L_1 = 1$ and $L_2 = 0$. Specifically, the error probability is[4]

$$-\frac{v_1 v_2}{2\pi i}\int_{-\infty+i\varepsilon}^{\infty+i\varepsilon}\frac{dz}{z(z+iv_1)(z-iv_2)} = \frac{v_1}{v_1+v_2}. \tag{4.44}$$

At $E_c/N_0 = 10$ dB, we have $v_1 = 0.121951$ and $v_2 = 5.0$; then the dibit pair-wise error probability for dibit interleaving is 0.02381. For bit interleaving with $b_k = 11$, the pair-wise error probability is expressed by Eq. (4.35) with $L = 2$,

[4]Since $L_2 = 0$, u_1 and u_2 vanish in the error probability formula.

and is simplified to be

$$\left(\frac{v_1}{v_1+v_2}\right)^2 \cdot \frac{v_1+3v_2}{v_1+v_2} \tag{4.45}$$

At $E_c/N_0 = 10$ dB, $v_1 = 0.168541$ and $v_2 = 3.617843$; the dibit pair-wise error probability for bit interleaving is 0.00577. Since $v_1 << v_2$, Eq. (4.45) can be approximated as $3 \cdot (v_1/v_2)^2$. The squaring operation reflects the two independent bit metrics induced by bit interleaving, and makes the error probability very small. In contrast, there is no such squaring operation for dibit interleaving. Since every codeword is made of many dibits, it is clear that bit interleaving yields smaller codeword error probabilities on Rayleigh fading channels.

4.6 Summary

We have analyzed and presented the performance of various metrics for DPSK modulation on possibly fading channels when CSI is known and/or unknown. The optimal symbol metric (under symbol interleaving) when CSI is known involves the Bessel function $I(\cdot)$ and the noise level N_0, so approximated metrics were introduced for easier implementation. We adopt two possible approximations: weighted linear metrics and weighted square metrics. For both AWGN and fading channels, the weighted linear metric is better at all practical SNR ranges; furthermore, our simulation results show that the weighted linear metric is virtually as effective as the optimal Bessel metric. When CSI is not available, we show that the (unweighted) square metric is optimal on Rayleigh fading channels.

We also define bit metrics for DQPSK by using fictitious phase changes. Bit metrics can be defined for both dibit and bit interleaving. Under dibit

interleaving, the symbol (dibit) metrics and the bit metrics yield similar or the same performance; in fact, the weighted/unweighted square bit metric and the weighted/unweighted square dibit metric are in fact equivalent, respectively, under dibit interleaving. With the aid of bit metrics, we are able to derive an upper bound of codeword error rates for DQPSK on AWGN channels. As to the issue about interleaving methods, we show that dibit interleaving is better than bit interleaving on AWGN channels, but the opposite is true on Rayleigh fading channels.

In the literature, decision variables are almost always used as bit metric values. We have shown that the (unweighted) square bit metric is equivalent to a decision variable metric for binary and quaternary DPSK. Since the weighted linear metric is better than the unweighted linear or square metrics, metrics based on the decision variables can be replaced by weighted linear metrics for better performance when CSI is known (or accurately estimated).

Chapter 5

Voice Error Concealment Under ML Soft-Decision Decoding

5.1 Introduction

The performance of soft-decision decoding metrics for DPSK has been investigated in the previous chapter; it was shown that (near) ML decoding can be achieved by a suitably chosen implementable metric. Although ML soft-decision decoding yields the smallest decoding error probability, it may still fail to decode correctly when channels suffer from severe fading. This is due to the low capacity of the fading channel; Chapter 3 has shown that the capacity of fading channels is worse than that of AWGN channels.

For some applications, e.g., mobile radio, it may be preferable to have the channel decoder detect the presence of errors, for example, declare a decoder **failure**, rather than (in all likelihood) make an error. The errors detected can be later masked by interpolation/extrapolation. In voice communications, undetected decoding errors often result in clicks, squeaks, and other perceptually annoying effects; while for detected noise, error masking (or interpolation) tech-

niques can be applied to conceal the errors. These annoying effects are especially severe during long fades, indicating that interpolation is particularly needed during such fades. Our interpolation results, shown later in this chapter, will demonstrate that the longer the fade lasts, the larger the interpolation gain will be. Since the interleaving table used for speech communications usually is not very large (so that the interleaving delay can be tolerated), a long fade may last for many interleaving tables. For this reason, this chapter will treat the channel as a concatenation of AWGN channels with different SNR. Since the decoding metric should be derived from the short-term characteristic of the channel, the employed metric in this chapter will be based on AWGN channels.

Speech interpolation is a popular technique in speech coding even when no transmission errors are involved. In the adaptive predictive coding of speech invented more than three decades ago [58], the long-term periodicity (or pitch) of speech signals has been used as a basis for prediction of speech parameters; the pitch prediction is in fact a form of interpolation. Since then, long-term prediction has found more use in the linear-prediction based analysis-by-synthesis speech coders, such as code-excited linear prediction (CELP) [59]. In a typical CELP scheme, a speech waveform is divided into small segments called frames; the vocoder extracts a set of parameters from each speech waveform segment to represent the frame so that compression is achieved. In order to achieve a significant compression ratio, correlation characteristics of speech signal are usually incorporated into the design of a codec [32, 60]. For example, the parameters for long-term periodicity of speech may not be needed to be fully specified for every frame, since these parameters can be interpolated from neighboring frames with negligible distortion.

Although the correlation characteristics of speech signals have been utilized by the vocoder for compression, it is not uncommon that a significant residual redundancy is still left in the vocoder's output. The redundancy allows for the interpolation of those speech parameters which are damaged due to storage or transmission errors. In this thesis, we will use the Federal Standard 1016 CELP [6] as our exemplary vocoder. The parameters to be interpolated are not confined to those corresponding to long-term periodicity but also for other parameters which still contain redundancy in them. A description of this vocoder will be given later.

Since "pure" maximum likelihood soft-decision decoding algorithms always make an estimate of the transmitted codeword, the inherent capability of a source decoder to mask detected errors is not exploited. We therefore augment the "pure" ML decoder with the capability of detecting unreliable (or damaged) codewords and declaring decoding failure for subsequent interpolation. In particular, the "enhanced" soft-decision decoder not only performs decoding as traditional decoders do, but also uses a decision rule δ to attach every decoded codeword with a flag indicating whether a decoding failure has occured or not. The interpolator then carries out its operation based on these flags. Because interpolation still causes speech distortion and the decoding failure rate is a tradeoff of the undetected error rate, the decision rule δ must be designed to balance these two rates.

Before we proceed, let's imagine what quantities the receiver needs to know in order to detect unreliable codewords efficiently. Clearly, based on the received signal, if the a posteriori probability of every codeword being transmitted and the decoding error probability are available to the receiver, the optimal δ and

interpolator can somehow be designed, at least theoretically. In order to calculate the a posteriori probability, the CSI as well as the variance of the white noise must be known. For example, the (short-term) AWGN assumption implies that the a posteriori probability for symbol i is proportional to $I_0(\frac{2A\ell_i}{N_0})$ (see Eq.(3.11)). Usually, the amplitude A is unavailable to the receiver, so the error probability may not be known. Even if A and N_0 are known, it is still very difficult to design the optimal error detector and interpolator for many reasons, e.g.,:

- The average distortion caused by interpolation, denoted as d_1, is different from the average distortion caused by decoding errors, denoted as d_2. Furthermore, these two distortions may be difficult to measure or quantify for source-encoded signals. For this reason, it may be impossible to design the receiver solely by analytical derivation.

- Both d_1 and d_2 are not fixed but depend on many factors. These factors include: (1) the number of decoding failures and decoding errors in a neighboring time frame, and (2) the details of the interpolation procedure. The first factor, in turn, is affected by the channel noise, channel fading, and the adopted decision rule δ. These factors are all interwoven so that the optimal receiver is very difficult to design.

Because of all these difficulties, a designer may need to try different combinations of δ and interpolation algorithms before the best combination is selected. This procedure is very tedious, and the combination chosen heavily depends on the underlying application. To avoid elaborating too much on any specific application, this thesis will assume a fixed interpolation algorithm and focus discussion on how to decide the detection rule δ based on the interpolation algorithm.

If both the carrier amplitude A and N_0 are known to the receiver, the error probability of each estimated codeword can be determined and the detection rule δ could be designed based on the decoding error probability. In this thesis, however, we will investigate a somewhat more complex problem; namely, that the carrier amplitude is unknown. In particular, we will define the **normalized codeword reliability** (NCR) for a DPSK system and use it to decide whether the estimated codeword is reliable or not.

In the next section, NCR will be defined and its statistics be investigated. The knowledge of the noise variance N_0 is needed in order to calculate NCR; we will assume it is known to the receiver in that section. The effects caused by estimated N_0 will be briefly addressed at the end of Section 5.4. In Section 5.3, we will present the coding and interleaving scheme for an exemplary land mobile radio (LMR) system. The chosen interpolation algorithm also will be described in that section. In Section 5.4, the detection rule δ will be designed based on calculated NCR; actually, two rules will be discussed and their performance be compared in terms of segmented SNR (SEGSNR). In addition to an objective segmented SNR measurement, a subjective listening test will also be employed in assessing the performance.

5.2 The Normalized Codeword Reliability for DPSK Systems

Since we are concerned with the problem of detecting unreliable codewords for error concealment under ML soft-decision decoding, a measurement of the reliability of a soft-decision decoded codeword must be defined. The issue with regard

to the reliability of a demodulated symbol (or codeword element) has been addressed in literature [11, 16]. In [11], every decoded codeword element is assigned a **reliability class**, and a **reliability weight** is associated with the class. The **generalized distance** between a received word and a codeword was defined based on the class to which a codeword element belongs. An improved version of [11] was presented in [16], where the weight in [11] is generalized to permit every one of the q codeword elements[1] to have a weight. If the weights are defined in terms of a log-likelihood metric, ML decoding can be achieved. (Clearly, the weight is another name for the decoding metric.) Denoting $\alpha_{k,j}$ as the weight of the codeword element b_j at position k, and defining[2] $\alpha_{k,j} = \ln(Pr(r_k|b_j))$, the ML decoded codeword is the one which maximizes the cumulative weight. In [16], another parameter β_k is defined as the difference between the two largest weights at position k. If the largest weight, say α_{k,j_1}, is much larger than any other weight at position k, the codeword element b_{j_1} apparently has high reliability. In other words, the parameter β_k is an indicator of the reliability of the codeword element at position k.

We now extend the concept of β_k in [16] from the codeword element level to the codeword level. Denoting r as the received signal used to estimate a codeword and $\mathbf{c}_j$ as a codeword, we define weight $\alpha_j = \ln(Pr(r|\mathbf{c}_j))$. Apparently, the codeword estimated by the decoder is the one which corresponds to the largest weight among M weights, where M is the size of the code. We define β as the difference between the two largest weights; then β can serve as an indicator

[1] We assume the code is over $GF(q)$.

[2] Our definition excludes the negation used in [16]; therefore, minimum of $\alpha_{k,j}$ in [16] is the maximum in this paper.

of the reliability of the estimated codeword. Since β is the difference of two log-likelihood values, we give it the acronym DLL in this thesis.

Without loss of generality, assume $\mathbf{c}_0$ and $\mathbf{c}_1$ correspond to the largest and the second largest weights, and define

$$\beta \triangleq \alpha_0 - \alpha_1 = \ln(\frac{Pr(r|\mathbf{c}_0)}{Pr(r|\mathbf{c}_1)})$$

If we assume the transmitted codeword was (with probability one) either $\mathbf{c}_0$ or $\mathbf{c}_1$, the decoding error probability by choosing $\mathbf{c}_0$ is

$$\frac{Pr(r|\mathbf{c}_1)}{Pr(r|\mathbf{c}_0) + Pr(r|\mathbf{c}_1)},$$

which can be written as

$$\frac{1}{e^{\beta}+1}.$$

This illustrates that β is a very good measurement of the reliability of the decoded codeword, because it almost describes the error probability — the ultimate measurement of reliability. Unfortunately, since the carrier amplitude A is unknown, α_0 and α_1 (therefore β) cannot be determined; therefore, some other reliability measurement must be defined without the knowledge of A.

Let's first investigate the problem of devising the reliability measurement for a modulation symbol, and then extend it to the codeword level. For DQPSK, the log-likelihood for dibit $i \in \{00, 01, 10, 11\}$ is (by Eq.(3.11))

$$F + \ln I_0(\frac{2A\ell_i}{N_0}), i = 0, 1, 2, 3. \tag{5.1}$$

where F doesn't depend on i and can be omitted. In Chapters 3 and 4, we have shown that the logarithm of $I_0(\cdot)$ can be approximated to yield the linear metric ℓ_i with almost no decoding performance degradation; this chapter also makes this assumption. For a given dibit position, if ℓ_{i_1} and ℓ_{i_2} are the largest and the

second largest, respectively, among the four dibit metrics, it is very tempting to define $\ell_{i_1} - \ell_{i_2}$ as the reliability for this dibit position. Is this an appropriate reliability measurement? If the transmitted dibit is 00, then Eq. (3.41) can be rewritten in the following form:

$$\ell_i^2 = (G_i \cos(\zeta_i) + n_{c_i})^2 + (G_i \sin(\zeta_i) + n_{s_i})^2 \tag{5.2}$$

where $G_0 = AT_s$, $G_1 = G_2 = AT_s/\sqrt{2}$, $G_3 = 0$; ζ_i is a phase depending on i and the phase uncertainty η; n_{c_i} and n_{s_i} are two independent Gaussian random variables with zero mean and variance $\sigma^2 = \frac{N_0 T_s}{2}$. Thus, ℓ_0, ℓ_1, and ℓ_2 are Rician distributed, and ℓ_3 is Rayleigh distributed. These distributions are not normalized; therefore, $\ell_{i_1} - \ell_{i_2}$ (which doesn't reflect the underlying SNR) cannot indicate how reliable a decoded dibit is. For this reason, we normalize every ℓ_i by dividing it by $\sigma = \sqrt{\frac{N_0 T_s}{2}}$; the distributions of normalized metrics, denoted as $\tilde{\ell}_i$, can be expressed as

$$f_{\tilde{\ell}_i}(x) = x \exp[-\frac{x^2 + \gamma_i^2}{2}] I_0(\gamma_i x), \tag{5.3}$$

where $\gamma_i^2 = G_i^2/\sigma^2$. For $i = 3$, $G_3 = 0$ and the above distribution degenerates into the normalized Rayleigh distribution. It is clear that the distributions of normalized metrics depend only on the SNR (e.g., E_c/N_0). We define the symbol reliability as $\tilde{\ell_{i_1}} - \tilde{\ell_{i_2}}$, and call it the **normalized symbol reliability** (NSR). Clearly, NSR does reflect the underlying SNR; the NSR has a high/low mean when SNR is high/low. Hence, NSR is an indication as to the short-term quality of the channel.

Without discarding the scaling factor $2A/N_0$, the DLL corresponding to a symbol (which is still denoted as β) is $\frac{2A}{N_0}(\ell_{i_1} - \ell_{i_2})$. This quantity can be written

as

$$\beta = \frac{2A}{\sqrt{N_0}} \frac{\ell_{i_1} - \ell_{i_2}}{\sqrt{N_0}} = \sqrt{\frac{2A^2 T_s}{N_0}} \cdot (\text{NSR}). \tag{5.4}$$

Eq. (5.4) indicates that β equals the product of NSR and the square-root of $4E_s/N_0$. Although NSR may not be as good as β, the calculation of NSR only requires the knowledge of N_0 but not A. Furthermore, NSR does reflect the current SNR so it could be the most efficient reliability indicator when CSI is not available. In practice, e.g., for fading channels, the amplitude A may vary too fast to be accurately estimated by the receiver; on the other hand, N_0 stays relatively stable and can be estimated by the receiver quite accurately. In this section, N_0 is assumed to be known by the receiver; the effect of estimated N_0 will be addressed later.

By summing up all the normalized symbol metrics of a codeword, the result is called the normalized codeword metric; the **normalized codeword reliability** (NCR) is defined as the difference between the two largest normalized codeword metrics. We will use the NCR as the basic quantity to detect decoding failures in the rest of the paper. To employ NCR wisely, it is necessary to understand its statistics. We first investigate some statistics of NSR. The properties of NCR are then drawn from those of NSR.

To calculate NSR, the receiver first selects the two largest from the four figures — $\tilde{\ell}_0$, $\tilde{\ell}_1$, $\tilde{\ell}_2$, and $\tilde{\ell}_3$ — and then takes their difference. The order statistics involved are quite complex because these four figures are not mutually independent. Specifically, grouping $(\tilde{\ell}_0, \tilde{\ell}_3)$ as one pair, and $(\tilde{\ell}_1, \tilde{\ell}_2)$ as the other pair, the two variables within the same pair are mutually independent, but each variable in a pair is correlated with both variables in the other pair. Because of the correlation, the distribution of the NSR is difficult to obtain analytically. In order

to investigate its statistics, we take two approaches: (1) we approximate the pdf of NSR by simulation; (2) we neglect the correlation among the four normalized dibit metrics and derive the pdf of the NSR analytically. Results from these two approaches will be compared later.

Assuming dibit 00 is transmitted, $f_{\tilde{\ell_1}} = f_{\tilde{\ell_2}}$. For notational convenience, let

$$f_0 = f_{\tilde{\ell_0}};\ \ f_2 = f_{\tilde{\ell_1}} = f_{\tilde{\ell_2}};\ \ f_3 = f_{\tilde{\ell_3}}.$$

We also let y and x denote the largest and the second largest values, respectively, from among the four normalized dibit metrics. Assuming independence, the joint density $w(x,y)$ can be written as $w(x,y) = w_1(x,y) + w_2(x,y)$, where

$$w_1(x,y) = 2f_0(y)f_2(x)\int_{-\infty}^{x} f_2(z)dz \int_{-\infty}^{x} f_3(z)dz + f_0(y)f_3(x)[\int_{-\infty}^{x} f_2(z)dz]^2, x \le y$$

and

$$\begin{aligned} w_2(x,y) \ = \ & 2f_0(x)f_2(y)\int_{-\infty}^{x} f_2(z)dz \int_{-\infty}^{x} f_3(z)dz + f_3(y)f_0(x)[\int_{-\infty}^{x} f_2(z)dz]^2 + \\ & 2f_2(x)f_2(y)\int_{-\infty}^{x} f_0(z)dz \int_{-\infty}^{x} f_3(z)dz + 2f_3(x)f_2(y)\int_{-\infty}^{x} f_0(z)dz \int_{-\infty}^{x} f_2(z)dz + \\ & 2f_2(x)f_3(y)\int_{-\infty}^{x} f_0(z)dz \int_{-\infty}^{x} f_2(z)dz, x \le y \end{aligned}$$

The term $w_1(x,y)$ corresponds to correct symbol estimation, while the term $w_2(x,y)$ corresponds to erroneous estimation. Let $\triangle = y - x$, the pdf of NSR (or $\triangle$) can be written as $w(\triangle) = w_1(\triangle) + w_2(\triangle)$, where

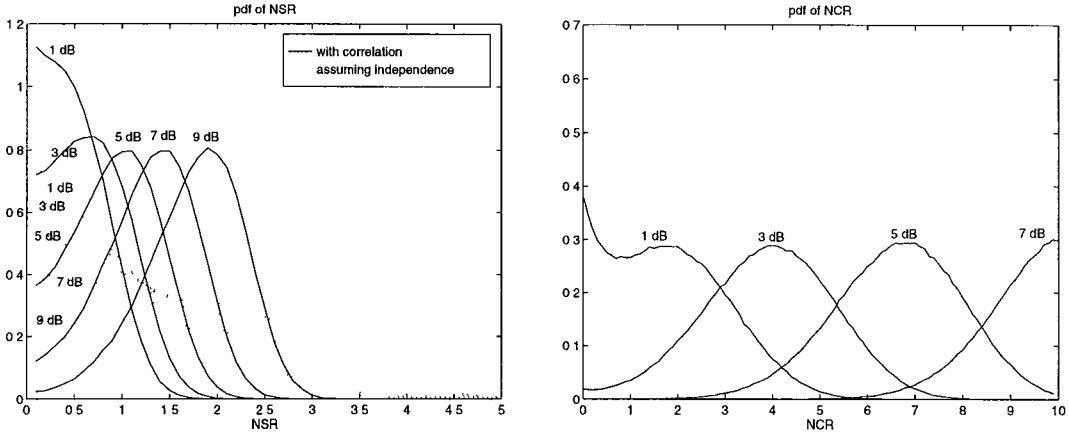
$$\begin{aligned} w_1(\triangle) \ = \ & \int_{-\infty}^{\infty} \{2f_0(y)f_2(y-\triangle)\int_{-\infty}^{y-\triangle} f_2(z)dz \int_{-\infty}^{y-\triangle} f_3(z)dz \\ & + f_0(y)f_3(y-\triangle)[\int_{-\infty}^{y-\triangle} f_2(z)dz]^2\}dy \end{aligned}$$

and

$$w_2(\triangle) \ = \ \int_{-\infty}^{\infty} \{2f_0(y-\triangle)f_2(y)\int_{-\infty}^{y-\triangle} f_2(z)dz \int_{-\infty}^{y-\triangle} f_3(z)dz$$

$$+f_3(y)f_0(y-\triangle)[\int_{-\infty}^{y-\triangle} f_2(z)dz]^2$$
$$+2f_2(y-\triangle)f_2(y)\int_{-\infty}^{y-\triangle} f_0(z)dz\int_{-\infty}^{y-\triangle} f_3(z)dz$$
$$+2f_3(y-\triangle)f_2(y)\int_{-\infty}^{y-\triangle} f_0(z)dz\int_{-\infty}^{y-\triangle} f_2(z)dz$$
$$+2f_2(y-\triangle)f_3(y)\int_{-\infty}^{y-\triangle} f_0(z)dz\int_{-\infty}^{y-\triangle} f_2(z)dz\}dy$$

Figure 5.1.(a) plots the pdf $w(\triangle)$ of the NSR assuming independence as well as the true pdf approximated by simulation without assuming independence. (The curves in this figure, including both part (a) and (b), are parameterized by (E_c/N_0).) We can see that the pdf curves with independence assumption do not

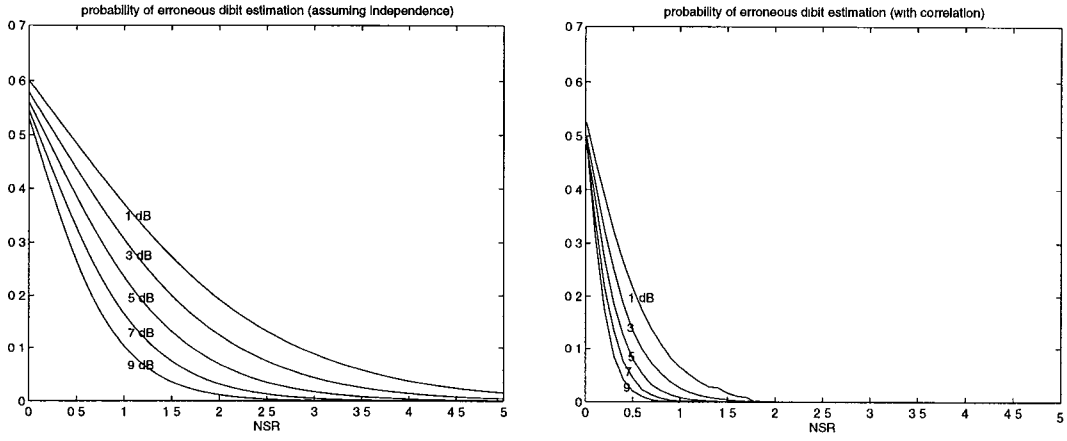


(a) (b)

Figure 5.1: Probability density functions: (a) NSR; (b) NCR for Golay code

exactly match the real pdf curves; in fact, the independence assumption tends to cause the pdf curves to spread out. Nevertheless, the mean of a NSR variable with independence assumption is quite close to the real mean.

Figure 5.2 plots the error probability with respect to the observed NSR, where Figure 5.2.(a) is obtained by plotting $w_2(\triangle)/(w_1(\triangle)+w_2(\triangle))$, and Figure 5.2.(b) is obtained by simulation. All the curves in Figure 5.2 are parameter-



(a) (b)

Figure 5.2: Probability of erroneous dibit estimation with respect to observed NSR: (a) Assuming independence; (b) No independence assumption

ized by E_c/N_0. By this figure, it is clear that a fixed NSR will give a lower error probability when E_c/N_0 is higher, and vice versa. This phenomenon can been explained by Equation 5.4, where a high SNR leads to a high β (equivalently, a low error rate) with a fixed NSR, and vice versa. When NSR is zero (or very close to zero), the error probability is greater than 0.5, instead of being equal to 0.5; this is because the transmitted dibit may not always correspond to one of the two largest metric values.

Although the knowledge of the NSR alone cannot describe the error probability perfectly, it still provides excellent information about the reliability of the received dibit, especially for higher NSR; for example, when NSR is greater than 2, the error probability is virtually 0 (see part (b) of Figure 5.2). If the SNR value can be (roughly) estimated, the estimation may be coupled with the observed NSR to render a more accurate error probability estimate (or more accurate reliability measurement). It is easy to obtain a rough estimate of the

underlying SNR; for example, Figure 5.1 indicates that the underlying SNR in a period can be reflected by the mean of NSR during that period.

The extension from NSR to NCR is quite obvious, but the statistics of the NCR are far more complicated, because it involves many mutually correlated variables (i.e., normalized codeword metrics). Even with the independence assumption, an analytical derivation of the pdf of NCR is still prohibitive. For this reason, we resort to simulation to ascertain some of the properties of NCR. Figure 5.1.(b) plots the measured pdf of NCR for the extended Golay code. Apparently, the curve shape of NCR is quite similar to that of NSR, except that the means of NCR are larger, and the curves are more spread out because of the large number of symbols per codeword. The error probability curves, with respect to an observed NCR, still take a shape similar to that of NSR, and a fixed NCR will yield a lower error probability when SNR is higher and vice versa, as in the NSR case.

The usage of NCR in determining whether to erase a decoded codeword may vary for different applications. Since the unreliable codewords need to be interpolated, the interpolation algorithm unavoidly affects the rule δ. In order to avoid elaboration of a specific application in this paper, our primary focus is not on designing the interpolation algorithm for a particular application; instead, we will simply choose a given interpolation algorithm for a particular system, and illustrate how to use the NCR to design δ.

In the next section, we will explain the specific system configuration and the interpolation algorithm.

5.3 An Error Protection Configuration and An Interpolation Algorithm

In this section, we describe channel coding aspects of a land mobile radio application. The speech codec is the FS 1016 CELP standard. An associated interpolation algorithm is also described.

5.3.1 The Federal Standard 1016 Codec

The FS 1016 Codec standard [6] is a CELP-based voice coding standard, which breaks a sampled speech signal into blocks of samples that are processed as one unit. This coding technique is based on analysis-by-synthesis search procedures, perceptually weighted vector quantization, and linear prediction. The input speech is sampled at 8 kHz and the resulting output bit stream is produced at 4800 bps.

In FS 1016, every block (or frame) is 30 msec long, which consists of four 7.5 msec subframes. The CELP analysis consists of three basic functions:

- Short-term spectrum prediction over the entire 30 msec frame using a tenth order linear prediction filter
- Modeling long-term speech periodicity by an adaptive codebook VQ (called the pitch VQ)
- Vector quantization of the residual from the spectrum prediction and the pitch VQ using a fixed stochastic codebook.

The selected codewords from the pitch VQ and the stochastic VQ are further scaled before transmission. The optimal scaled excitation vectors (i.e., code-

words) from these two codebooks are selected by minimizing a time-varying, perceptually weighted distortion measure. The perceptual weighting operation improves subjective speech quality by exploiting masking properties of human hearing. CELP synthesis consists of the corresponding inverse operations of the three analysis functions performed in the reverse order, with the optional addition of a postfilter to enhance the output speech.

Each CELP frame consists of 144 bits, which includes bits for 1) the adaptive codebook indices (pitch delay and delta delay) and the associated gains; 2) the stochastic codebook indices and the associated gains; 3) the 10 line spectral parameters (LSP), and 4) three control parameters (which contain a frame synchronization bit, the Hamming code parity bits for correction of some parameters, and a future expansion bit).

5.3.2 An Error Protection Configuration

We assume a digital land mobile radio system whose total channel bit rate is 8000 bps, consisting of the Federal Standard 1016 codec (4800 bps), 2400 bps voice coder error protection, and 800 bps framing, system control, and error protection bits. This was in fact a scheme proposed for the FS 1024 standard of the U.S. Government for the future digital land mobile radio applications [61][3]. We select the extended Golay code for error protection; in particular, we use six Golay codewords in each CELP frame to protect 72 important CELP bits. A CELP bit is called important based on the effect an error in that bit causes in the synthesized speech.

[3]This was an earlier version of the proposal; later versions used bit rates that are slightly different.

Based on the results from [62], we pick 72 CELP bits as important and encode them into six Golay codewords. These 72 bits are organized as follows:

- Codeword 1: LSP bit 5-7, LSP bit 25, pitch delay bit 35-42
- Codeword 2: LSP bit 9-11, LSP bit 28, pitch delay bit 87-94
- Codeword 3: LSP bit 31, delta delay bit 65-67, pitch gain bit 45-47, (stochastic) codebook gain bit 57-61
- Codeword 4: delta delay bit 116-119, delta gain bit 70-72, codebook gain bit 82-86
- Codeword 5: LSP bit 13-15, LSP bit 34, pitch gain bit 97-99, codebook gain bit 109-113
- Codeword 6: LSP bit 18-19, LSB bit 21-22, delta gain bit 122-124, codebook gain bit 134-138.

After 24 more overhead bits are included, a system frame contains 240 bits. These 240 bits are then dibit-interleaved and modulated for transmission. At the receiver, the decoder performs the ML soft-decision decoding algorithm and calculates the NCR. For the extended Golay code, we modified the ML soft-decision decoding algorithm in [21] to obtain the two codewords corresponding to the two largest codeword metrics so that the NCR can be calculated. Given the NCR of every decoded codeword, the decision rule δ then attaches each codeword with a flag; specifically, if the flag is 0, the codeword is regarded good (or reliable); if the flag is 1, the codeword will be regarded as unreliable and interpolation will be attempted for this codeword. We will discuss the design of the decision rule δ in the next section.

5.3.3 An Interpolation Algorithm

An interpolation algorithm tries to recover the erased CELP bits in those codewords which are flagged as unreliable by the receiver. Designing an interpolation algorithm involves many issues, for example:

- What are the correlation statistics between CELP parameters in different subframes or frames? Interpolation cannot be well done unless the parameter correlation properties are used wisely.

- What is the occurrence pattern of decoding failures? Is it a completely random pattern? In land mobile radio, deep fades may span over many system frames, so decoding failures may not occur totally randomly but tend to concentrate during fading periods. In our simulation, we find that a deep fade may last as long as seven frames or more when the vehicle speed is 20 mph. When a long fade occurs, a good interpolator should be able to detect it and take appropriate steps to obtain better results.

- What is the best result an "optimal" interpolator may achieve? Because the ultimate judgement of the quality of the synthesized speech is subjective, it is impossible to define an "optimal" interpolator; an interpolator can always be modified to appeal to another subjective taste. Nevertheless, some objective measurements can still be used for evaluation of an interpolation algorithm although caution should be taken because objective measurements do not correlate well with subjective speech quality [63]. Later, we will use the segmented SNR as the objective measurement in addition to subjective listening tests.

Since the interpolation and the error detection rule δ are related, joint design of these two operations may yield a better result. However, since there are too many quantities (including some that cannot even be objectively quantified) involved in the two operations, the joint design cannot be accomplished purely by analytical derivation; usually, a designer may need to try different combination of δ and interpolation algorithms before a very good combination is selected. For simplicity, this thesis will not address this tedious procedure; instead, a simple and reasonable interpolation algorithm will be adopted later in this section[4], and emphasis will focus on the design of the detection rule δ in the next section.

Before the interpolation is described, let us first investigate a segment of the protected CELP parameters. Those parameters include the stochastic codebook gain, pitch delay, pitch gain, and some LSPs. Since the CELP bits are just encoded versions of physical quantities, the CELP bits should be converted to the corresponding physical quantities for the purpose of interpolation. For example, "LSP1=0" means that the physical value of LSP1 is 100 hertz. For the rest of this section, we will discuss interpolation in the context of the physical quantity. For the stochastic codebook gains and the pitch gains, negative values may happen (see Table 5.1 and 5.2 for the physical encoding/decoding levels of stochastical codebook gains and pitch gains). Negative gains represent "sign inversion" of the codewords in the two codebooks, whereas their amplitudes (or absolute values) represent the energy of the signals. The redundancy of the codebook gains and pitch gains is not well exhibited by the "signed" gains; instead, the redundancy resides in their amplitudes. Thus, the absolute values of the two

[4]This interpolation algorithm was obtained and tuned from experiment, and could appear *ad hoc*.

-1330	-870	-660	-520	-418	-340	-278	-224
-178	-136	-98	-64	-35	-13	-3	-1
1	3	13	35	64	98	136	178
224	278	340	418	520	660	870	1330

Table 5.1: Stochastic codebook gain encoding/decoding levels

-0.993	-0.831	-0.693	-0.555	-0.414	-0.229	0.000	0.139
0.255	0.368	0.457	0.531	0.601	0.653	0.702	0.745
0.780	0.816	0.850	0.881	0.915	0.948	0.983	1.020
1.062	1.117	1.193	1.289	1.394	1.540	1.765	1.991

Table 5.2: Pitch gain encoding/decoding levels

gains are used in interpolation. For the pitch delays of the second and the fourth CELP subframes, the CELP bits represent the offset of the corresponding pitch delay from the previous pitch delays; for these two parameters, true pitch delays will be calculated by considering the offset and the previous pitch delays.

Figure 5.3 shows a segment of the protected CELP parameters. Except for the LSPs, every CELP frame contains four samples of each type of parameter; for the LSPs, only one sample appears in each CELP frame. This figure illustrates that the CELP parameters are indeed correlated to their neighboring parameters of the same types, making interpolation possible; in the next chapter, we will present an entropy rate measurement for the LSP parameters as a more quantitative description of the correlation. Although correlation is apparent within CELP parameters, they may oscillate and make interpolation difficult. Pitch delays especially suffer from this oscillation phenomenon. In fact, new CELP-type vocoders are being proposed (e.g., [32]) to avoid the pitch delay oscillation. Due

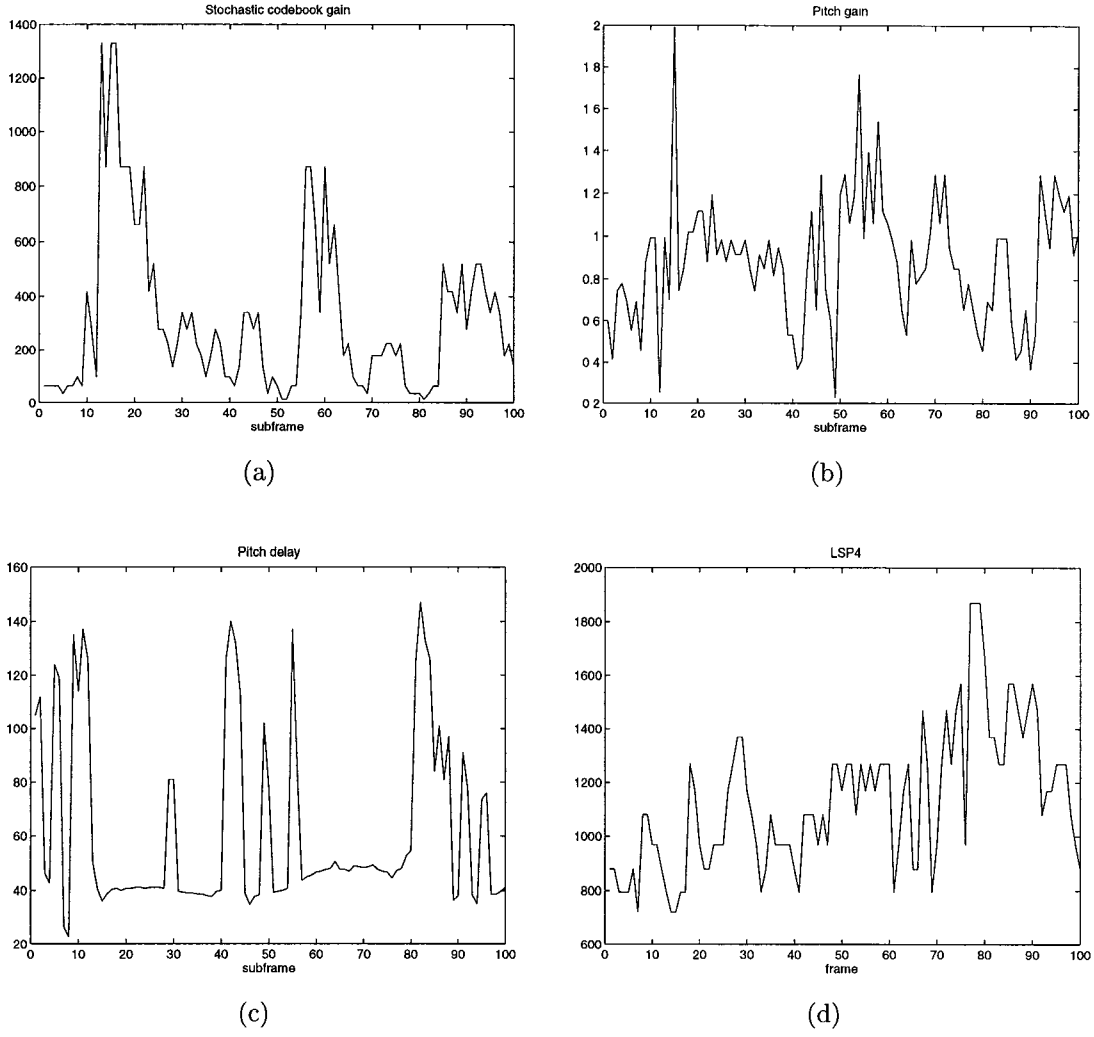


Figure 5.3: CELP parameters from a segment of speech: (a) Stochastic codebook gain; (b) Pitch gain; (c) Pitch delay; (d) LSP4

to parameter oscillation, our interpolator will attempt to detect the existence of oscillation; when oscillation is detected, appropriate steps will be taken to avoid bad interpolation.

If a parameter is to be interpolated (i.e., with a flag 1), the interpolator will examine a small window of neighboring parameters of the same type and calculate the average of those parameters with flags equal to 0, as a substitute for the parameter. However, if there are not sufficient reliable neighboring parameters or if parameter oscillation is detected, reliable interpolation may not be achieved; in these instances, appropriate steps will be taken to avoid bad interpolation. For different types of parameters, the details of interpolation will slightly differ.

When the receiver delay is concerned, we don't want any significant delay to be introduced by interpolation; hence, only the past and the present frames will be used to interpolate the present frame. As mentioned previously, an interpolation window is examined for each to-be-interpolated parameter; the window size cannot be too large, otherwise uncorrelated parameters could be involved in interpolation. We therefore pick the appropriate window size as four (not including the underlying parameter) for every type of parameter. A window ideally will extend equally at both sides of the corresponding parameter; unfortunately, it is not possible for some parameters because future frames are not used. For this reason, a window may be aligned to extend unequally on each side of the corresponding parameter when it is impossible to be aligned equally.

We now describe the interpolation details for each type of parameters:

- Stochastic codebook gains: If more than two out of the four parameters in a window have reliability flags being 0, the sample average and the sample variance of the reliable parameters are calculated; otherwise, interpolation

cannot be reliably performed and the underlying codebook gain will be substituted with a small number, which we pick as 35, in order to prevent blast. If the variance is larger than a pre-selected threshold, which is selected as 30000, the average is still regarded as unreliable; hence, the small gain 35 is used to substitute the underlying gain. If the sample variance is smaller than the threshold, the sample average is regarded as good and is used to substitute the codebook gain. Finally, in order to prevent the scenario in which the average is erroneously too large, the average is divided by two if it is greater than 400. (This is a conservative approach to avoid blasts.)

- Pitch gains: The procedure is very similar to the case of stochastic codebook gains except the sample variance threshold is 0.3 and the small gain for substitution is 0.531 instead of 35. Moreover, no division by two is ever performed.

- Pitch delay: Again, the procedure is similar to those used for the previous two types of parameters; but the variance threshold is 15 and again there is no division by two. Another difference is that when no reliable interpolation is done, the to-be-interpolated pitch delay is left unchanged.

- LSPs: The procedure is almost the same as the pitch delay case except the variance thresholds are different. Different LSPs may have different variance thresholds. For LSP2 through LSP10, the variance thresholds are chosen as 20000, 30000, 60000, 60000, 60000, 60000, 60000, 60000, and 20000, respectively. LSP1 is absent here because no bit of LSP1 is protected.

Those specific figures in this interpolation algorithm are chosen based on empirical simulation. Our experience shows that erroneous stochastic codebook gains and pitch gains are the major sources of annoying blasts, so it is very important to substitute "un-interpolable" gains with small values to avoid blasts.

5.4 Designing the Decision Rule δ

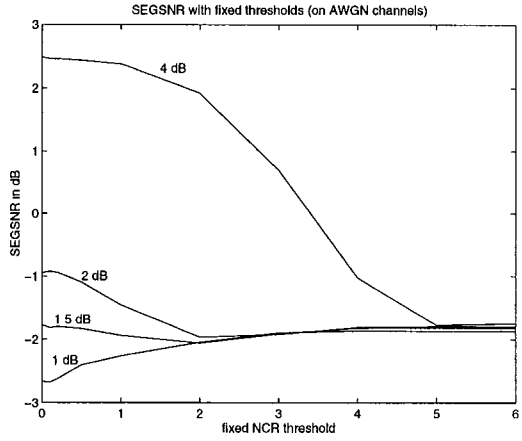
We now describe how to use the NCR to set the reliability flag of a decoded codeword for the system described in the previous section. Strictly, this task may be a big challenge because we don't have an analytical method for calculating the output speech quality based on the reliability of the decoded codewords, even with known CSI. For this reason, we rely heavily on simulation results in designing δ. All the data in this section was obtained by using a one-minute long speech in which two male and two female speakers speak the same set of sentences. The speech is originally PCM-coded with 8000 samples per second and each sample is quantized by 16 bits.

We select the segmented SNR (SEGSNR) [64] measurement to describe the objective speech quality in this thesis. The SEGSNR is defined as the average of the SNR of all short segments of synthesized speech from a long speech; in this thesis, a segment is chosen to be 7.5 ms (which corresponds to a CELP subframe).

Although our major concerns are on fading channels, fading channels may be too complex to deal with directly for designing the rule δ. We therefore first focus AWGN channels, and then extend to fading channels by modeling a fading channel as a changing AWGN channel with a different SNR at different time

periods.

Basically, the NCR of a codeword is compared with a threshold T to decide which flag value to take on. In particular, if the NCR is greater than T, the flag is chosen as 0; otherwise, the flag is 1. Obviously, a fixed threshold constitutes the simplest δ. Figure 5.4 shows the SEGSNR of the output speech with respect to different fixed thresholds on AWGN channels with different values of E_c/N_0. Apparently, when E_c/N_0 is greater than 1.5 dB, it is better not to declare any



(a)

Figure 5.4: SEGSNR by fixed thresholds on AWGN channels

unreliable codewords; on the contrary, when E_c/N_0 drops below 1.5 dB, it may be better to declare many unreliable codewords. Although declaring all codewords unreliable may be best for low SNR cases from the SEGSNR's point of view, it may not be practical because it will result in virtually quiet speech. Note that the curve for $E_c/N_0 = 1.5$ dB is essentially flat, indicating that whether the codeword is reliable is not a matter for concern. The NCR mean at $E_c/N_0 = 1.5$ dB is approximately 2.2, which was obtained by simulation.

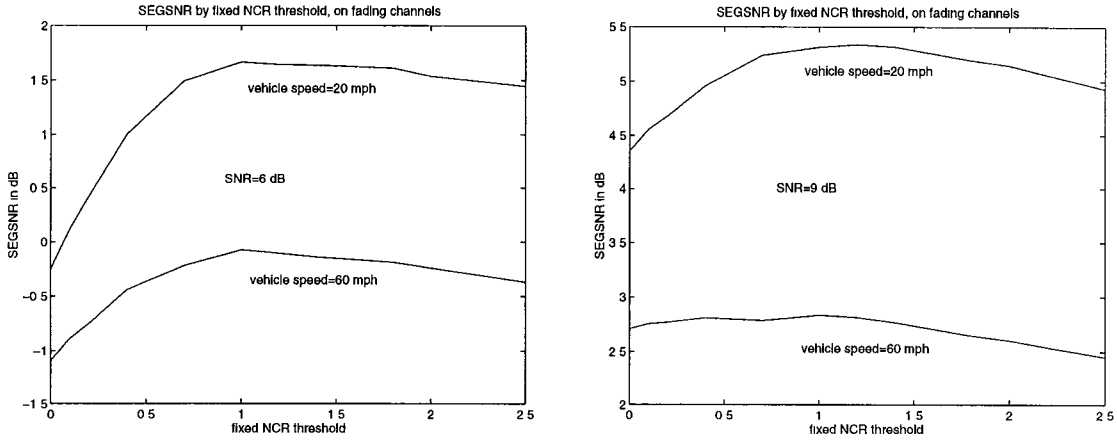
If we model a fading channel simply as a changing AWGN channel with

varying SNR at the receiver, the optimal fixed threshold, denoted as T_0, will be the result of a tradeoff among various time segments with possibly different SNR values so that the resulting average SEGSNR of the output speech is maximized. Apparently, the optimal threshold will depend on the distribution of SNR values at the receiver. Due to channel fading, SNR values at the receiver may vary from more than 10 dB down to negative dB values; from Figure 5.4, a fixed threshold value around 1.0 seems to be a good tradeoff. Figure 5.5 shows the performance of fixed thresholds on Rayleigh fading channels. In the employed channel simulator, the carrier frequency $f_c = 900$ MHz and the symbol rate is 8000 symbol/sec; hence, the normalized fading factors are 0.01 and 0.0033 for mobile speed 60 mph and 20 mph, respectively. From Figure 5.5, the optimal thresholds are indeed around 1.0 for all the cases. In this figure, we pick three E_c/N_0 values — 6 dB, 9 dB, and 12 dB. Note that these three E_c/N_0 values represent the signal-noise ratio for each bit at the transmitter, and do not necessarily reflect the SNR values at the receiver for a given time segment in a fading environment. When $E_c/N_0 = 6$ dB, the BER of a system without using channel codes is slightly more than 10%[5]; when $E_c/N_0 = 12$ dB, the BER of an uncoded system is approximately 3.5%.

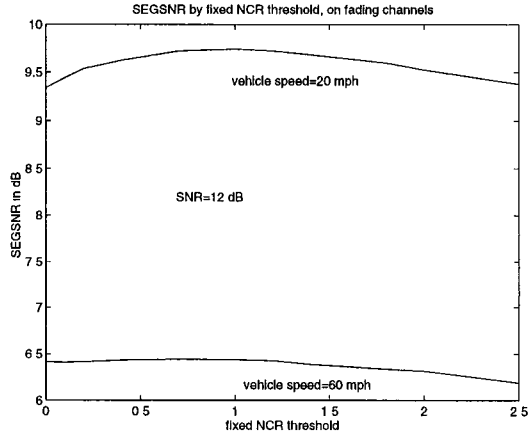
Because the fixed optimal threshold T_0 is a compromise during the whole communication period, this threshold is not optimal on a short-term basis. For example, during a non-fading period, the best threshold should be simply zero. For this reason, performance can be improved if the threshold can be adapted to current channel conditions.

According to the previous section, the mean of the NCR reflects the under-

[5]The vehicle speed does not noticeably affect the BER of uncoded systems.



(a) (b)



(c)

Figure 5.5: SEGSNR performance of fixed threshold on fading channels. (a) $E_c/N_0 = 6$ dB; (b) $E_c/N_0 = 9$ dB; (c) $E_c/N_0 = 12$ dB

lying SNR during a time segment; therefore, we calculate the average of the six NCRs in a frame as an indicator of the current SNR. If the average is high (i.e., SNR may be high), the threshold should be small; and vice versa. Strictly speaking, the SNRs associated with the six codewords in a frame are not the same, so the NCR average only reflects the average SNR in the corresponding frame. From Figure 5.4, the curve for $E_c/N_0 = 1.5$ dB (when the average NCR is about 2.2) is essentially flat; this implies that an NCR average larger/smaller than 2.2 should cause the optimal threshold to be lower/higher than the optimal fixed threshold, say 1.0. It could be extremely difficult to increase or decrease the threshold optimally, so we chose this simple adaptation rule:

$$m_{NCR} \cdot T_a = 2.2 \cdot T_0,$$

where m_{NCR} is the NCR average in a frame, T_a is the adaptive threshold, and T_0 is the optimal fixed threshold. We call the quantity $2.2 \cdot T_0$ the **target product**.

Figure 5.6 shows the performance of adaptive δ for different target products. This figure also includes the performance by the fixed threshold 1.0 as well as the performance of pure ML soft-decision decoding (i.e., no interpolation). It is clear that interpolation can improve the pure soft-decision receiver by as much as 1.5 dB on low E_c/N_0 channels. When E_c/N_0 is high enough (e.g., 12 dB), fewer decoding errors will occur so that fewer improvement can be obtained by interpolation. We also notice that, by using the adaptive threshold, the SEGSNR does not improve much over the fixed threshold; however, this small amount of improvement may still be noticeable subjectively. If one thinks that a smarter adaptation rule may further significantly improve the performance, this is, in fact, not the case. Figure 5.4 explains this line of thought. If the SNR at the receiver is consistently lower than 1.5 dB, choosing 1.0 as the fixed threshold

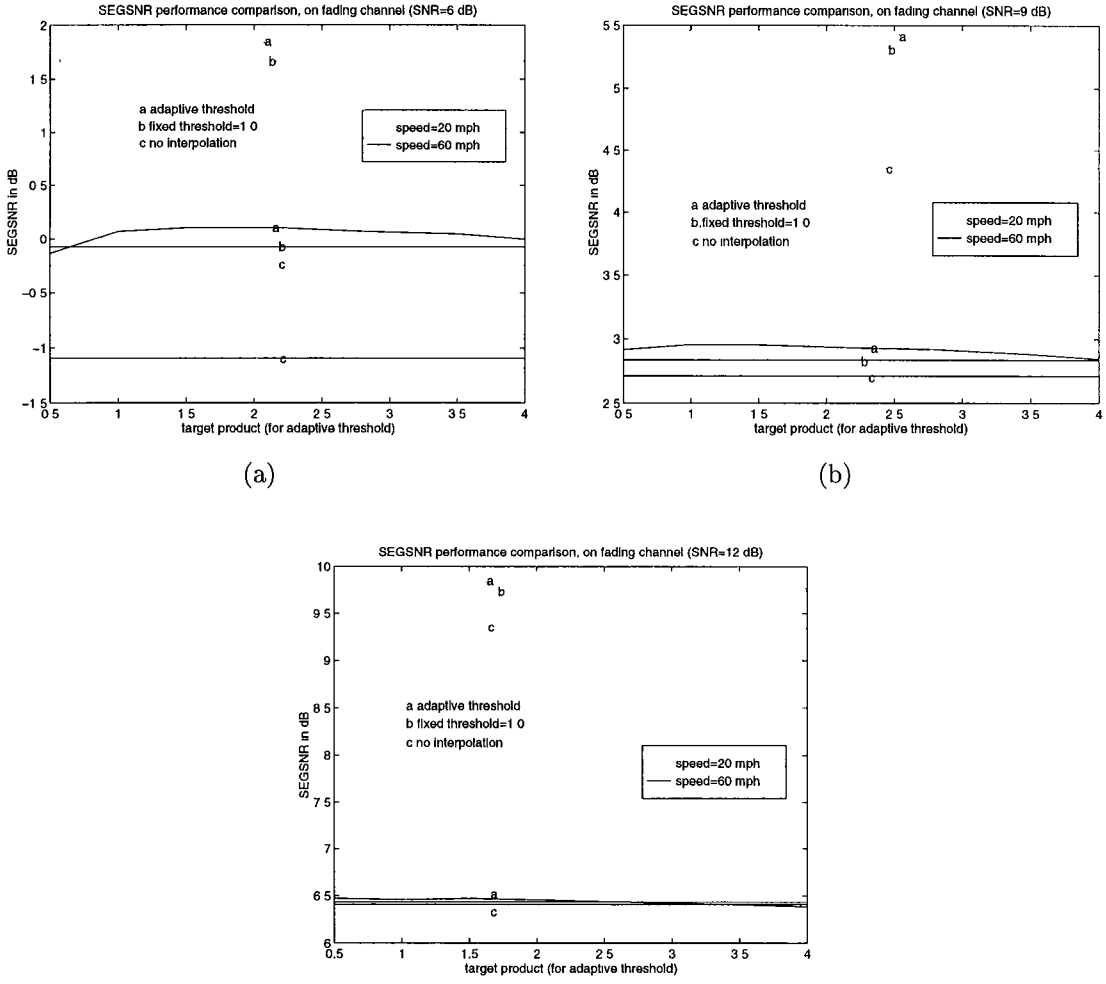


Figure 5.6: SEGSNR performance by adaptive thresholds, fixed threshold, and no interpolation at all: (a) $E_c/N_0 = 6$ dB; (b) $E_c/N_0 = 9$ dB; (c) $E_c/N_0 = 12$ dB

will only degrade the output SEGSNR by a small fraction of 1 dB against a very large (or the best) threshold; if the SNR at the receiver is consistently higher than 1.5 dB, choosing 1.0 as the fixed threshold will cause even smaller degradation. Therefore, only a small fraction of 1 dB improvement can be obtained by adapting the threshold around the best fixed threshold; apparently, our simple adaptation rule has achieved that amount of improvement.

We also perform a set of listening tests using the same speech and simulation channels just mentioned. We find that our interpolator yields satisfactory speech with virtually all the perceptually annoying effects being eliminated when E_c/N_0 lies between 9 dB through 12 dB. When E_c/N_0 drops to 6 dB, errors will happen so frequently that it is virtually impossible for an interpolator to generate a very good output speech; however, our interpolation still can eliminate most of the annoying effects and maintain the intelligibility as much as possible. Specifically, at low E_c/N_0, the interpolator will detect many unreliable and questionable codewords so that it is impossible to perform a reliable interpolation by using reliable neighboring codewords; therefore, the interpolator simply squelches those speech segments by choosing a small stochastic codebook gain and a small pitch gain, as mentioned earlier.

Up to now, we have always assumed N_0 is known to the receiver while calculating NCR; in reality, N_0 may need to be estimated. At the end of this section, we briefly evaluate the effect of estimated N_0 on the output SEGSNR. Table 5.3 shows the output SEGSNR by our adaptive δ rule with erroneous N_0 estimates when $E_c/N_0 = 6$ dB and the vehicle speed is 60 mph. As a reference, the resultant SEGSNR with accurate N_0 is 0.102 dB. From this table, we can conclude that δ is quite robust and can sustain -30% — $+20\%$ N_0 inaccuracy without

N_0 error	-60%	-30%	-10%	-5%	5%	10%	30%
SEGSNR	0.043	0.112	0.101	0.111	0.093	0.093	0.057

Table 5.3: Effects of N_0 errors

noticeable performance degradation. Even when the error of N_0 goes beyond this range, the performance degradation is still limited.

5.5 Summary and Remarks

We used normalized codeword reliability to augment the pure ML decoder with the capability of distinguishing unreliable codewords from reliable ones. Without the knowledge of CSI, NCR may be the most efficient parameter for describing the "goodness" of the decoded codeword. The NCR of a decoded codeword is compared to a threshold to decide which reliability flag (0 or 1) should be attached to the codeword. The threshold may be fixed or be adapted to the current channel conditions.

We used an exemplary system and interpolation algorithm to illustrate the use of NCR. Although the details of using NCR may vary for different applications and systems, our example has illustrated the most important issues.

Throughout this chapter, we inherently assumed that the traditional ML soft-decision decoder has been modified to search for the two codewords corresponding to the two largest codeword metrics. Is this type of modification feasible? Furthermore, many block codes do not have efficient maximum likelihood soft-decision decoding algorithms, thus sub-optimal soft-decision algorithms are used; for those codes, how can the codeword reliability be defined? The latter question may be answered by defining a sub-optimal codeword reliability which

is obtained from the two best codewords that the underlying sub-optimal soft-decision decoding algorithm can find. As to the question of modifying existing soft-decision decoding algorithms (optimal or suboptimal) to attain the second codeword, the answer may vary for different codes and different decoding algorithms. Fortunately, the modification is quite feasible at least for two popular classes of soft-decision decoding algorithms:

- If the underlying decoding algorithm is based on the code trellis (or graph or tree), the number of the survivor paths to each state can be doubled to accommodate the second best codeword. In this sense, the concept of codeword reliability can be applied to convolutional codes and trellis codes.

- If the underlying decoding algorithm first exhibits a set of good codeword candidates, then the best two (instead of only the best one) are selected.

The majority of the soft-decision decoding algorithms are found to fit into one (or both) of the above two classes; for other decoding algorithms, separate consideration may be needed.

Occasionally, problems may still arise with some decoding algorithms, even after modification. The problem occurs when the decoder finds only one or no codeword. In this case, the codeword reliability cannot be calculated. For example, the Chase decoding algorithm may encounter this difficulty. Usually, this situation happens when noise is high. To circumvent this problem, the receiver should choose a decoding algorithm that guarantees obtaining two codeword candidates. Recently, an order-statistic-based soft-decision decoding algorithm was proposed in [25]. This algorithm may be applied to any binary code and achieves excellent performance at relatively low computation complexity for short codes,

medium length codes, and high-rate long codes. More important for us, this algorithm can guarantee a set of good codeword candidates (at least two). Although this algorithm is originally for binary codes, it is not difficult to extend it to (some) non-binary codes, e.g., the Reed-Solomon codes. Specifically, in any (n, k) Reed-Solomon code, any k codeword elements constitute a set of **most reliable independent (MRI) symbols**, so the proposed decoding algorithm of [25] may be extended to Reed-Solomon codes.

Since most of the soft-decision decoding algorithms can be modified to obtain the second best codeword at very little complexity increase (usually, less than two folds of the original complexity), the proposed concept of normalized codeword reliability can be applied to many other systems.

Chapter 6

Source-Dependent Sequence Decoding

6.1 Introduction

The previous chapter utilized the "residual redundancy" of the CELP vocoder's output to conceal the detected errors from the ML soft-decision decoder. This chapter will investigate the problem of utilizing the residual redundancy to directly enhance the decoder's performance (not via interpolation), especially at low SNR. In short, we will investigate maximum a posteriori (MAP) decoding. The possibility of enhancing the decoder's performance by using the redundancy inherent in the information source was mentioned in 1948 in Shannon's celebrated paper [65]: "However, any redundancy in the source will usually help if it is utilized at the receiving point. In particular, if the source already has redundancy and no attempt is made to eliminate it in matching to the channel, this redundancy will help combat noise.". Forney [26] also mentioned the possibility of using the a priori distribution of the source bits in the Viterbi algorithm, but did not provide any use or applications. Hagenauer [1] derived a procedure for using the a priori information in the Viterbi algorithm when the source bits are independent, and called the modified algorithm the APRI-VA.

Although the fundamental theory of MAP decoding is nothing new, there is no general MAP decoding algorithm that can be realistically applied to every source and encoding scheme. Unless the source statistics are very simple (e.g., independent but non-uniform distribution), the MAP decoding algorithm may be very complex and may cause a long delay; therefore, simplified decoding algorithms are almost always used for feasible realization. For example, if bits in a codeword are correlated to bits in subsequent codewords (however, bits in the same codewords are mutually independent), [1] proposes and calculates the a posteriori information for each information bit from the received signals to be used as a part of the a priori information for bits to arrive later; thus, a decoder with soft input and soft output. The a posteriori information may be regarded as the information provided by the previously received signals for handy use to help decode bits that may arrive in the next codeword or a few codeword later. This approach substantially simplifies the MAP decoding algorithm; otherwise the MAP decoding may be very complex. A version of this decoding algorithm with soft output is the soft-output Viterbi algorithm (SOVA) proposed in [66]. SOVA is combined with the redundancy of the source (called APRI-SOVA) in [1], and is applied to PCM coded sources and the GSM codec to yield significant decoding gain. The soft-in and soft-out decoding procedure also forms the foundation of iterative decoding algorithms [67]. The very promising success of iterative decoding is its application to the recently invented "turbo codes" [68], which have shown good performance at rates close to capacity.

The extraction of the a posteriori information from the received signal dates back to the pioneering work on symbol-by-symbol maximum a posteriori decoding of linear codes [69]. The underlying decoding procedure uses the received

signal sequence to decide the maximal a posteriori probability for each symbol. (There is no a priori probability of the source involved here.) In contrast, the conventional Viterbi algorithm determines the most likely trellis path, and the conventional block decoding algorithm determines the most likely block (codeword). One drawback of the symbol-by-symbol maximum a posteriori decoding algorithms is their higher decoding complexity, and thus for block codes, can only be applied effectively to special codes (e.g., codes with simpler trellis structures). This may constitute one of the main reasons why the symbol-by-symbol maximum a posteriori decoding algorithms are not widely employed. Furthermore, in most applications of interest, the performance of the conventional Viterbi or block decoding algorithms yield performance identical to that of symbol-by-symbol maximum a posteriori decoding algorithms [69]. The main virtue of the symbol-by-symbol decoding algorithms may be their capability to provide soft output for interative decoding.

In order to focus on the usage of the redundancy inherent in the source, this thesis will primarily consider the problem of estimating the most likely sequence but not symbol (i.e., codeword for block codes and trellis path for convolutional codes). As mentioned above, the maximum likelihood sequence decoding will not degrade the performance compared to symbol-by-symbol decoding algorithms. The resulting decoding algorithm is simpler and often can be obtained directly from the original soft-decision decoding algorithm.

The derivation of MAP decoding algorithms in the literature usually assume PSK modulation, e.g., [1]. In this chapter, the discussion will be based on DQPSK. We will focus on block codes that may or may not have a simple trellis structure. The discussion on trellis-based decoding for block codes can be

directly applied to the Viterbi decoding of convolutional codes.

The source a priori statistics and how the information bits are grouped into a codeword could significantly influence the employed MAP decoding algorithms and their complexity. In order to avoid complex decoding algorithms, the next two sections will focus on binary sources with two simplifying assumptions:

- The information bits are statistically independent but may be unevenly distributed.
- The information bits form a Markov chain.

Since the a priori statistics of source bits in a practical system do not usually follow one of the two simple statistics, the primary purpose of the next two sections is only to demonstrate the advantage of exploiting the source a priori statistics and to introduce some basic MAP decoding techniques. Later, we will employ a practical land mobile radio example to demonstrate a significant performance gain by exploiting the a priori source redundancy.

6.2 MAP Decoding for Independent Sources

This section considers an information source, where every bit occurs independently of every other bit (with "0" occurring with probability p).

Let $\mathbf{c} = (c_0, c_1, ..., c_{n-1})$ be a codeword from a (n, k) block code $\mathcal{C}$, which is transmitted via a M-ary DPSK modulation. Denote $n' = n/\log_2 M$ and $k' = k/\log_2 M$; codeword $\mathbf{c}$ can be written in terms of the modulation symbols, namely $\mathbf{c} = (b_0, b_1, ..., b_{n'-1})$. Letting r be the received signal corresponding to a transmitted codeword, and $p(\mathbf{c})$ be the a priori probability of codeword $\mathbf{c}$, the

MAP decoder will estimate $\hat{\mathbf{c}}$ as the transmitted codeword where

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}\in\mathcal{C}}\{\ln[p(\mathbf{c})\cdot p(r|\mathbf{c})]\} \tag{6.1}$$

Assuming systematic encoding (namely, the first k bits on the first $k' = k/\log_2 M$ symbols of a codeword are information) and r_j as the segment of r corresponding to b_j, Eq. (6.1) can be decomposed as

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}\in\mathcal{C}}\left\{\sum_{j=0}^{k'-1}[\ln p_{b_j} + \ln p(r_j|b_j)] + \sum_{j=k'}^{n'-1}\ln p(r_j|b_j)\right\}, \tag{6.2}$$

where p_{b_j} is the a priori probability of symbol b_j. As mentioned in earlier chapters,

$$\ln p(r_j|b_j) = F(r_j) + \mu_{b_j}(r_j),$$

where $F(r_j)$ does not depend on b_j and can be left out. The MAP decoding metric can be defined by

$$\hat{\mu}_{b_j}(r_j) = \begin{cases} \ln p_{b_j} + \mu_{b_j}(r_j) & \text{if } j \le k'-1 \\ \mu_{b_j}(r_j) & \text{otherwise} \end{cases} \tag{6.3}$$

Given this MAP metric, Eq. (6.2) can be rewritten as

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}\in\mathcal{C}}\sum_{j=0}^{n'-1}\hat{\mu}_{b_j} \tag{6.4}$$

From Eq. (3.11), we have for AWGN channels

$$\mu_{b_j}(r_j) = \ln(I_0(\frac{2A}{N_0}\ell_{b_j})) \approx \frac{2A}{N_0}\ell_{b_j};$$

from Eq. (3.53), we have for Rayleigh fading channels (without CSI)

$$\mu_{b_j}(r_j) = \frac{\xi A^2}{\frac{1}{4}A^4T_s^2(1-\xi^2) + N_0(N_0 + A^2T_s)}\cdot\ell_{b_j}^2.$$

Assume the all zero codeword $\mathbf{c}_0$ is transmitted, the decoder will make a decoding error by estimating another codeword $\mathbf{c}_h$ as the transmitted codeword if

$$\sum_{b_j \in \mathbf{c}_0} \hat{\mu}_{b_j}(r_j) < \sum_{b_j \in \mathbf{c}_h} \hat{\mu}_{b_j}(r_j). \tag{6.5}$$

On AWGN channels, we modify Eq. (4.38) to obtain the pair-wise error probability :

$$P_e = Prob(\sum_{k_1}(\mid Y_{k_1} \mid - \mid X_{k_1} \mid) + \sum_{k_2}(\mid Y_{k_2} \mid - \mid X_{k_2} \mid) > \delta_1) \tag{6.6}$$

where k_1 and k_2 refer to dibit 11 and 01 (or 10) in $\mathbf{c}_h$, respectively, and $\delta_1 = \frac{N_0}{2A}(\ln p_{\mathbf{c}_0} - \ln p_{\mathbf{c}_h})$. On Rayleigh fading channels, the pair-wise error probability is:

$$P_e = Prob(\sum_{k}(\mid X_k \mid^2 - \mid Y_k \mid^2) < \delta_2) \tag{6.7}$$

where k refers to nonzero dibits, and $\delta_2 = \frac{\frac{1}{4}A^4T_s^2(1-\xi^2)+N_0(N_0+A^2T_s)}{\xi A^2} \cdot (\ln p_{\mathbf{c}_h} - \ln p_{\mathbf{c}_0})$.

Since every codeword may be transmitted with a different probability, we cannot use the pair-wise error probabilities to determine the union bound based on the assumption that the all-zero codeword is transmitted. In fact, the decoding error probability given a particular codeword transmitted is usually different than that given another codeword transmitted. As such, except for small codes, the performance of MAP decoding is usually difficult to determine by analytical derivation, and is usually investigated based on simulation. We now use pair-wise error probabilities to calculate MAP decoding error probabilities for a simple binary repetition code; after that, we use simulation to investigate the performance of MAP decoding for the extended Golay code.

For a binary $(n, 1)$ repetition code, each information bit "0"/"1" is encoded into a sequence of 0/1 of length n, respectively. Since "0" occurs with probability

p, the a priori codeword probability of $\mathbf{c}_0$ is p and that of the other codeword $\mathbf{c}_1$ is $1-p$. Let n be even and $L = n/2$; thus, a repetition codeword contains L DQPSK modulation symbols.

For AWGN channels, the error probability in (6.6) for a repetition code can be determined by the infinite series method of Chapter 4. In particular,

$$P_e = \frac{1}{2} + \frac{2}{\pi} \sum_{\substack{n=1 \\ n:\ odd}} \frac{A_n \sin\theta_n}{n}, \tag{6.8}$$

where A_n are θ_n are given in (4.23), i.e.,

$$A_n = \prod_{i=0}^{2L-1} A_{i,n}$$

$$\theta_n = \sum_{i=0}^{2L-1} \theta_{i,n}$$

$$A_{i,n} = \sqrt{\{E[\cos(nwZ_i)]\}^2 + \{E[\sin(nwZ_i)]\}^2},$$

but $\theta_{i,n}$ is modified as

$$\theta_{in} = \tan^{-1}\left\{\frac{E[\sin(nwZ_i)]\cos(nw\epsilon) - E[\cos(nwZ_i)]\sin(nw\epsilon)}{E[\cos(nwZ_i)]\cos(nw\epsilon) + E[\sin(nwZ_i)]\sin(nw\epsilon)}\right\}$$

where $\epsilon = \frac{1}{2L} \cdot \frac{N_0}{2A} \ln\frac{p}{1-p}$. When $p = 0.5$, the source is uniformly distributed and the MAP decoding degenerates into ML decoding.

For Rayleigh fading channels with the square metric, the pair-wise error probability in (6.7) can be determined by a modification of (4.41). In particular, the error probability of (6.7) can be expressed as

$$P_e = -\frac{(v_1 v_2)^L}{2\pi i} \int_{-\infty+i\varepsilon}^{\infty+i\varepsilon} \frac{e^{-i\delta_2 z} dz}{z(z+iv_1)^L (z-iv_2)^L}. \tag{6.9}$$

As in (4.41), P_e of Eq. (6.9) can be calculated by the residue method. When $\delta_2 \leq 0$, the integral contour of (6.9) can be extended to the infinite upper

semicircle (as was done in (4.41)). The pole inside the resultant closed integral contour is iv_2. However, when $\delta_2 > 0$, the upper semicircle may diverge. In this case, the integral contour of (6.9) can be extended to the infinite lower semicircle, and both $z = 0$ and $z = -iv_1$ are poles inside the resultant closed integral contour[1]. The residues are easy to calculate by hand only for small L. For $L = 1$, (6.9) is simplified to become (for negative δ_2)

$$P_e = \frac{v_1 e^{v_2 \delta_2}}{v_1 + v_2}$$

or (for positive δ_2)

$$P_e = 1 - \frac{v_2 e^{-v_1 \delta_2}}{v_1 + v_2}$$

For $L = 2$, (6.9) is simplified to become (for negative δ_2)

$$P_e = \left[\frac{v_1^2(3v_2^2 + 4v_1 v_2 + v_1^2)}{(v_1 + v_2)^4} - \frac{v_1^2 v_2 \delta_2}{(v_1 + v_2)^2} \right] e^{v_2 \delta_2}$$

or (for positive δ_2)

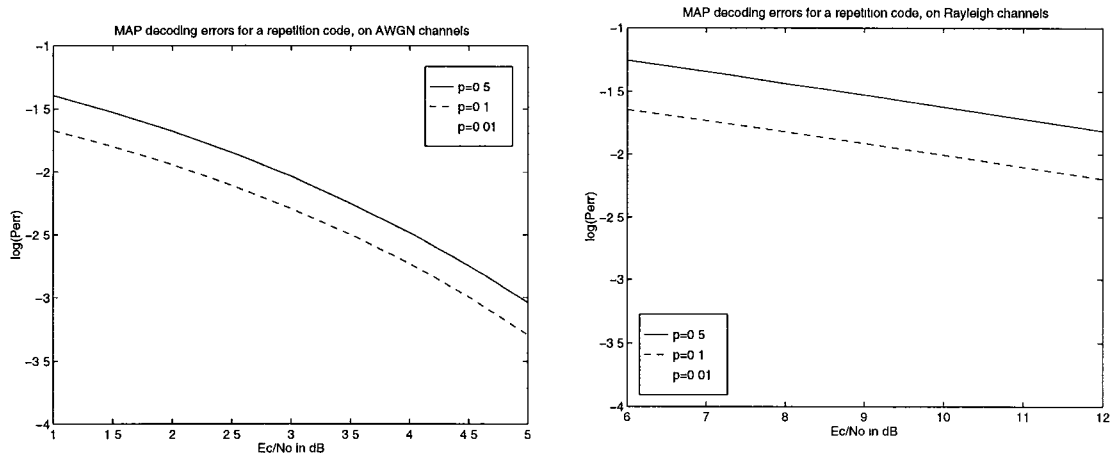
$$P_e = 1 - \left[\frac{v_1 v_2^2 \delta_2}{(v_1 + v_2)^2} + \frac{v_2^2(3v_1^2 + 4v_1 v_2 + v_2^2)}{(v_1 + v_2)^4} \right] e^{-v_1 \delta_2}$$

All of the pair-wise error probabilities mentioned above are based on the assumption that $\mathbf{c}_0$ is transmitted, and therefore can be denoted as $P_e(\mathbf{c}_0 \to \mathbf{c}_1)$. The error probability $P_e(\mathbf{c}_1 \to \mathbf{c}_0)$ can be defined and calculated in the same manner. The MAP decoding error probability for the repetition code is given by

$$P_{err} = p \cdot P_e(\mathbf{c}_0 \to \mathbf{c}_1) + (1 - p) \cdot P_e(\mathbf{c}_1 \to \mathbf{c}_0). \tag{6.10}$$

P_{err} is depicted in Figure 6.1 for both AWGN and slow Rayleigh fading channels. The curve for $p = 0.5$ represents the performance of ML decoding for both evenly

[1]The contour integral direction is now clockwise.



(a) (b)

Figure 6.1: MAP decoding error probabilities for a repetition code ($L = 1$): (a) on AWGN channels; (b) on slow Rayleigh fading channels.

and unevenly distributed source. This figure indicates that MAP decoding can obtain a significant coding gain over ML decoding, especially on fading channels.

Figure 6.2 shows the simulated bit error rates of MAP decoding where the extended Golay code is employed. This figure was obtained on AWGN channels, where dibit interleaving was performed and the linear dibit metric was used. It is observed that MAP decoding may achieve significant coding gain over ML decoding (which corresponds to the curve of $p = 0.5$) when the source bits are not uniformly distributed (e.g., $p = 0.1$).

6.3 MAP Decoding for Markov Sources

In the previous section where information bits are mutually independent, the a priori source probability of an information bit was directly incorporated into the symbol metrics. In this section, where information bits form a Markov

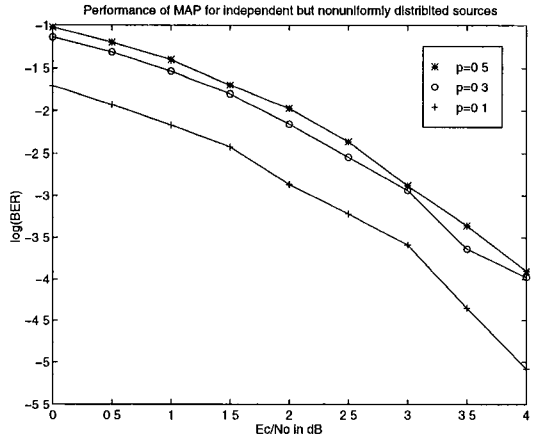


Figure 6.2: Bit error rates of the Golay code by MAP with independent but nonuniformly distributed sources.

chain, the a priori source statistics are better described in terms of a trellis rather than single bits, because the source transition probability can be easily incorporated into the trellis branch metric as long as the encoding is systematic. As such, systems using trellis-based decoding (e.g., Viterbi decoding) can be easily modified to take advantage of the source correlation statistics to help decode.

J. K. Wolf proposed a soft-decision decoding algorithm for block codes based on trellis diagrams of block codes [13]. Since then, numerous trellis-based soft-decision decoding algorithms for block codes have been proposed. Unfortunately, the trellis structures of block codes are usually complex except for very small codes, high-rate codes, and some special codes. Therefore, this section will also investigate a sub-optimal MAP decoding algorithm which does not use the code trellis.

6.3.1 Decoding by Trellis Structure

If the encoding is systematic and the used code has a simple trellis, the transition probability from one state to another state is incorporated into the traditional ML decoding branch metric. If p_{ij} is the transition probability from state i to state j in the trellis and μ_{ij} is the log-likelihood of the output corresponding to that transition, then the MAP metric for the branch can be defined as $\ln p_{ij} + \mu_{ij}$. For convolutional codes, this procedure is straightforward; for block codes, however, this procedure will depend on how the trellis is formed, and may become somewhat complex. As an example, we investigate the trellis-based MAP decoding by using the extended Golay code.

In [70], Forney constructed a three-section trellis for the extended Golay code and presented an efficient ML soft-decision decoding algorithm for this code. We will adopt this trellis to decode Markov sources. The three-section trellis is shown in Figure 6.3. This trellis contains four stages, where both the first and the fourth stages consist of only one state, whereas the second and the third stages contain 64 states which are further divided into eight 8-state groups and only the first and the eighth groups are shown in this figure. Moreover, every branch shown in this figure (which we hereafter call a simplified branch, to avoid confusion) actually contains two branches which represent two mutually complementary binary 8-tuples. Because each branch consists of a binary 8-tuple, every branch contains seven one-bit transitions. There is also a transition from the last bit of a branch to the first bit of every following branch (which is called a branch transition). Note that there is a simplified branch from a state of group i at stage 2, to every state of group i at stage 3 but not to any other state at stage 3. Any path from section 1 through section 3 is a codeword; therefore, MAP

soft-decision decoding needs to find the "best" 3-section path as the estimated codeword.

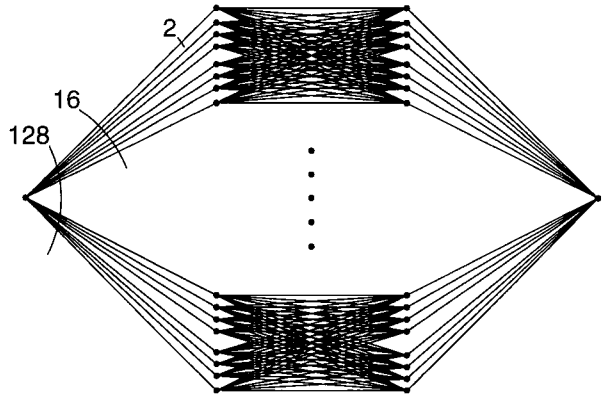


Figure 6.3: A trellis structure of the extended Golay code

In a trellis-based ML algorithm, each bit is associated with a metric derived from the received signal as the weight of the transition. The decoding algorithm finds as the estimated codeword the path with the maximal cumulative weight. If we regard transition probabilities as the analogy of the a priori probabilities of the previous section, the MAP metric for Markov sources can be achieved by taking the logarithm of the transition probabilities and including them in the ML metric.

A meaningful transition probability occurs only between information bits. Therefore, the MAP decoding algorithm must recognize transitions among information bits. Conventionally, under systematic encoding, we may assume the information bits constitute the first part of a codeword. Unfortunately, the first 12 bits of the Golay code in Figure 6.3 do not correspond to the 12 information bits. To find out the information bits in this figure, we systematically encoded the Golay code and then permuted the codeword bits such that the resulting

Golay code could be represented as in Figure 6.3. To determine a way to permute the Golay code into the version in Figure 6.3 is not a trivial task. If it was done by exhaustive search, the computation complexity could be formidable 24!. Fortunately, the extended Golay code assumes very nice properties so that the permutation can be determined by the properties of octads and the MOG [71, 72].

We first systematically encoded the (24, 12) Golay code so that the first half of a codeword are information bits. The generator matrix is shown in Table 6.1. If the 24 bits of each systematically encoded codeword are labeled from left to

100000000000110001110101
010000000000011000111011
001000000000111101101000
000100000000011110110100
000010000000001111011010
000001000000110110011001
000000100000011011001101
000000010000001101100111
000000001000110111000110
000000000100101010010111
000000000010100100111110
000000000001100011101011

Table 6.1: A generator matrix for the (24, 12) Golay code

right as bit 23 down to bit 0 as follows,

23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

and then they are permuted into this form:

14 11 2 0 23 22 21 20 15 17 10 18 3 19 6 8 13 16 5 12 7 9 4 1,

the permuted codeword can be represented by Figure 6.3. We can see, after permutation, that the first section contains five information bits (i.e., the five

bits with labels 14, 20, 21, 22, and 23), the second section four bits and the third section three bits. Based on this permutation format, the arriving information stream is first sequentially labeled as 14, 23, 22, 21, 20, 15, 17, 18, 19, 13, 16, and 12. These 12 bits are then rearranged into decreasing order in terms of their labels, i.e., 23,22,...,12, for systematic encoding. The systematically encoded codewords are then permuted to the form which can be represented by Figure 6.3. By doing so, the information bits in Figure 6.3 appear in the same order (i.e., from left to right) in which they actually arrive at the encoder. Because information bits are not necessarily linked together, non-information bits between two information bits will be neglected while computing transition probabilities.

Given the code trellis and the information bit positions, the MAP decoding algorithm for Markov sources is similar to the Viterbi algorithm, with some necessary modification. In conventional Viterbi algorithms, only one survivor path enters each state; in our MAP algorithm, however, we must retain two survivor paths into each state (the ending point of each 8-tuple simplified branch in Figure 6.3 is called a state), one survivor ending in "0" and the other ending in "1", except at the final stage. The reason for this is because the decoder cannot decide which one of the two survivor paths is better at that stage; further transition in the next branch will incorporate different transition probabilities to these two survivor paths. Here is the algorithm:

(1) Compute the "conventional" metric for each bit. The branch metric is simply the sum of the metrics of its component bits.

(2) Within each branch, incorporate into the branch metric the logarithms of the underlying transition probabilities within information bits.

(3) Based on the modified branch metric from step (2), perform the Viterbi-like algorithm but retain two survivor paths into each state. As the trellis proceeds from section 1 into section 2, and from section 2 into section 3, the **branch transition** probability must be incorporated into the total weights of paths. The branch transition probability is the probability of transition from the last information bit of a branch to the first information bit of the succeeding branch. (4) At the end of section 3, choose as the output codeword the path which corresponds to the largest cumulative metric.

The MAP decoding algorithm mentioned above does not consider the correlation between two consecutive codewords. If correlation is to be considered, the algorithm can be easily modified for this purpose by simply incorporating the **codeword transition** probability into the weight. However, this may increase the total algorithm complexity and the decoding delay; therefore, we neglect the correlation between two consecutive codewords.

Assuming binary symmetric Markov sources with p being the transition probability from bit 0/1 to bit 1/0 and DQPSK modulation, Figure 6.4 depicts the bit error rates of MAP for the Golay code with p equal to 0.1, 0.3, and 0.5 respectively. As in Figure 6.2, this figure was obtained on AWGN channels, where dibit interleaving was performed and the linear dibit metric was used.

As mentioned above, the main difference between the MAP algorithm and the Viterbi algorithm lies at step (3), where two survivor paths, instead of one, are retained at each state. By doing so, the algorithm is more complex than the Viterbi algorithm. To reduce the complexity, we may disregard the correlation between sections by retaining only one survivor path into a state, as the Viterbi algorithm does. We call the approach near-MAP decoding algorithm 1. Note

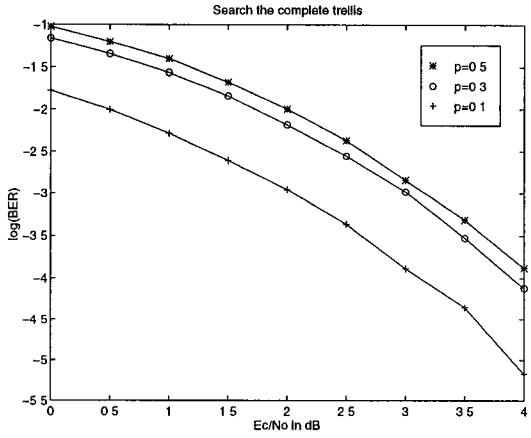


Figure 6.4: Performance of MAP decoding for Markov sources.

that this simplified algorithm is equivalent to the MAP algorithm if $p = 0.5$, because $p = 0.5$ represents independent sources and no transition probabilities need to be considered.

6.3.2 Decoding Without Using Code Trellis

Many block codes do not have simple trellis structures, so the trellis-based MAP decoding methods cannot be applied. A brute-force solution to this difficulty is calculating the a priori codeword probability $p(\mathbf{c})$ from the transition probability, and determining the MAP output by using Eq. (6.1). However, this approach requires that the decoder examines every codeword, a computationally complex procedure except for very small codes. To mitigate the heavy computation, the receiver may only search for a portion of all codewords and pick the best estimate from among those codewords. In the next section where an application is introduced, one version of this type of simplied decoding method will be applied to a Reed-Solomon code.

There are many possible ways to choose a set of good codeword candidates and reach a final decision of the best one among them. Usually, the tradeoff between performance and complexity should be considered when a decoding algorithm is designed. We now propose an algorithm based on the Chase algorithm [12] and the Markov transition trellis.

We can encode binary (n, k) block codes systematically so that the first k bits of a codeword correspond to source bits. Because these k bits form a Markov chain, we may represent them in terms of a Markov chain transition state diagram. We regard the remaining $n-k$ parity check bits as independent. Figure 6.5 shows a codeword in terms of its Markov state transition diagram (or trellis). In each of the first k bit positions (or stages), there are two states (state 0

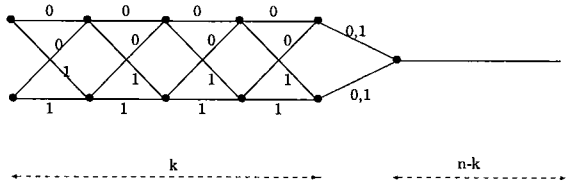


Figure 6.5: Codewords in terms of Markov state transition.

and state 1); for the remaining stages, there is only one state. Assuming a bit metric is available, the branch weights in the first k stages are modified metrics (namely metrics including the transition probabilities), while the remaining branch weights are just conventional metrics. Every codeword corresponds to a path from stage 1 to stage n, but any binary n-tuple (not necessarily a valid codeword) also corresponds to a path. Therefore, Viterbi-like algorithms applied to this diagram may not necessarily output a valid codeword. Nevertheless, we will employ Viterbi-like algorithms to select "good" paths (or n-tuples) and apply a hard-decision decoding algorithm to each of these good n-tuples to achieve a set

of "good" candidate codewords. We then choose from this set of candidates the one that corresponds to the largest cumulative weight. This approach is called near-MAP decoding algorithm 2. The approach of first achieving a set of good candidate codewords and then choosing the best one is similar to the Chase algorithm; however, we have incorporated the source transition statistics and chosen good paths instead of choosing unreliable bits as in the Chase algorithm.

More specifically, this algorithm could be described as follows: We use a Viterbi-like algorithm based on the trellis in figure 6.5 to choose m_1 best paths (k-tuples) up to stage k, and choose m_2 least reliable bits from the $n-k$ parity bits. By toggling these m_2 bits (as in the Chase algorithm) and attaching the resulting 2^{m_2} binary $(n-k)$-tuples to the ends of the m_1 binary k-tuples, we achieve $M = m_1 \cdot 2^{m_2}$ test n-tuples. By applying a hard-decision decoding algorithm to each of these M test n-tuples, we achieve a set of candidate codewords and the one corresponding to the largest cumulative metric is the best output codeword. To achieve the best m_1 best k-tuples, we use a Viterbi-like algorithm which retains m_1 survivor paths into both state 0 and state 1 at every stage. At the end of the k^{th} stage, the m_1 survivor paths (or k-tuples) into state 0/1 are the best m_1 binary k-tuples from all possible k-tuples ending at state 0/1. From among these remaining $2 \cdot m_1$ binary k-tuples, we choose the best m_1 binary k-tuples.

We take M as the complexity measure of this algorithm, but by doing so, we face the problem of choosing the best m_1 and m_2 if M is fixed. Since the Golay code has a rate of 1/2, it is reasonable to choose m_1 and m_2 so that $m_1 = 2^{m_2}$. On the other hand, because our purpose is to minimize the information bit error rate, we may try to increase m_1 to correct more information bits; however, large

m_1 requires small m_2 and some good candidate codewords resulting from chasing more bits in the $n-k$ parity bits may be missing. We had tried $m_1 = 16, 8, 4, 2$ and 1 with M fixed at 16 in our simulation and found that $m_1 = 8$ is best while $m_1 = 1$ worst.[2] Figure 6.6 depicts the bit error rates of best and worst m_1 with $p = 0.5$ as well as $p = 0.1$. In the following discussion, we will choose $m_1 = 8$ and $M = 16$ (and so $m_2 = 1$) for our algorithm.

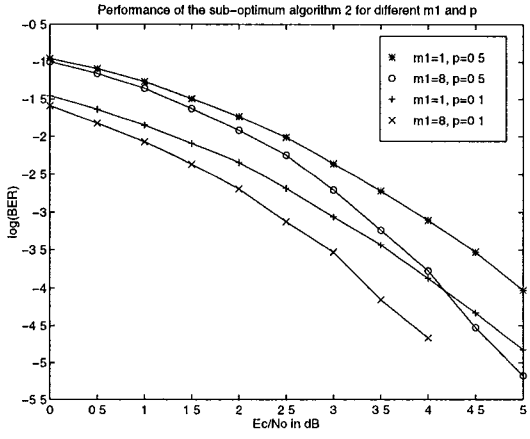


Figure 6.6: Performance of near-MAP decoding algorithm 2 for different m_1

Figure 6.7 compares the bit error rates of the Golay code by the MAP algorithm, near-MAP algorithm 1 and near-MAP algorithm 2 with Markov transition probability p equal to 0.5 (the dash-dotted lines or the upper two curves) and 0.1 (the dashed lines) on AWGN channels with dibit interleaving. The dotted line will be mentioned later. When $p = 0.5$, these three algorithms yield very close performance; in fact, the near-MAP algorithm 1 and the MAP algorithm yield the same results when $p = 0.5$. When p becomes small, the MAP algorithm out-performs the other two simplified ones. This is reasonable for near-MAP

[2]For $m_1 = 16$ or $m_1 = 4$, the performance is very close to that of $m_1 = 8$.

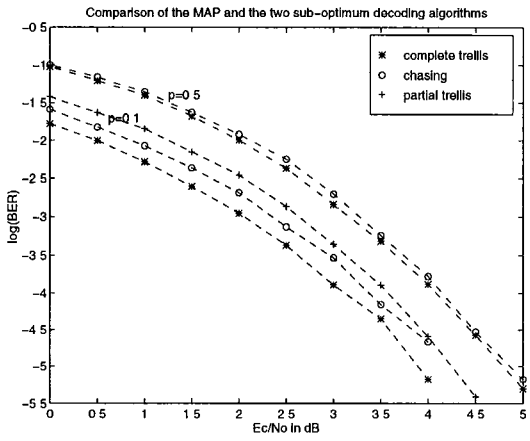


Figure 6.7: Performance comparison of the MAP and the two near-MAP decoding algorithms.

algorithm 1 because this algorithm ignores the correlation between two consecutive sections in the Golay code trellis and performance degradation from this situation becomes noticeable as p becomes small. For near-MAP algorithm 2, the performance is very close to that of near-MAP algorithm 1.

Since dibit is the basic unit in DQPSK modulation, we can redraw Figure 6.5 as Figure 6.8 in terms of dibits. In Figure 6.8, there are two states and two branches connecting two consecutive states before the the first $k/2$ stages (i.e., the information part); for the remaining stages, there is only one state. The metric for this dibit-based trellis is apparently the dibit metric, which is the natural metric since the modulation is dibit based.

Based on this trellis as well as dibit metrics, near-MAP algorithm 2 was performed and its performance for $p = 0.1$ was plotted as the dotted line in Figure 6.7. In this figure, we observe that the performance of near-MAP algorithm 2 has improved and become very close to the MAP performance. Therefore, near-MAP decoding algorithms for Markov sources can achieve a very good per-

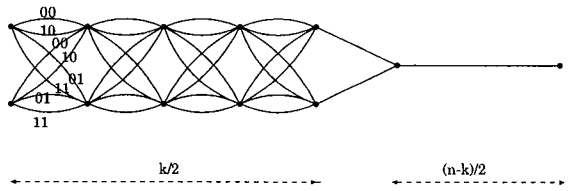


Figure 6.8: Morkov transition trellis in terms of dibits

formance, and can be efficient without using the code trellis.

6.4 A Practical Application

In this section, we apply the concept of MAP decoding to a mobile radio scheme. The discussion in this section is based on [74, 75]. The land mobile radio scheme is almost identical to the one used in Chapter 5, except a Reed-Solomon code is used instead of the extended Golay code. Furthermore, we only focus on the protection of LSPs, and assume all other CELP parameters are received without errors. The main purpose of this section is to demonstrate that there indeed is enormous source redundancy in the CELP encoded output, and the receiver can utilize the redundancy to significantly improve the decoding performance.

We first quantify the "residual redundancy" inherent in the LSP's of the CELP-encoded speech. The redundancy will be quantified by adopting two statistical models for the LSPs. The first model incorporates only the non-uniformity of the LSPs and their correlation within a CELP frame; the second model allows correlation between frames as well. When these models were "trained" using an actual CELP bit stream, it was shown that as many as 12.5 of the 30 high-order LSP bits in each frame may be redundant.

Once the residual redundancy in the LSPs is quantified, we present decoding

algorithms that attempt to exploit that redundancy assuming Reed-Solomon encoding. We will present four soft-decision decoding (SDD) algorithms, each of which exploits a different level of the residual redundancy. Simulation results based on the four algorithms will show that significant performance improvement – both objectively and subjectively – can be achieved over systems without exploiting the residual redundancy within sources.

6.4.1 LSP Residual Redundancy

This subsection quantifies the residual redundancy in the encoded LSPs of FS 1016 CELP. In FS 1016 CELP, each LSP is quantized by either three bits or four bits; the second through fifth LSPs are quantized by four bits whereas the rest are quantized by three bits. For simplicity, we only consider the redundancy with respect to the three most significant bits of each LSP, thus ignoring the least significant bit in the second through fifth LSPs.

Let $\{U_{i,j} : 1 \leq i \leq 10, j = 1, 2, \ldots\}$ denote a sequence of encoded LSPs, where $U_{i,j}$ is the i^{th} (three-bit) quantized LSP in frame j. Let

$$\mathbf{U}_j = [U_{1,j}, U_{2,j}, \ldots, U_{10,j}]$$

denote the vector consisting of the 10 quantized LSP's in frame j. Assuming $\{\mathbf{U}_j\}$ is a stationary random process, the entropy rate **per frame** is given by

$$H_F = \lim_{n \to \infty} \frac{1}{n} H(\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_n),$$

where

$$H(\mathbf{U}_1, \ldots, \mathbf{U}_n) = - \sum_{\mathbf{u}_1, \ldots, \mathbf{u}_n} P(\mathbf{U}_1 = \mathbf{u}_1, \ldots, \mathbf{U}_n = \mathbf{u}_n) \log_2 P(\mathbf{U}_1 = \mathbf{u}_1, \ldots, \mathbf{U}_n = \mathbf{u}_n).$$

H_F represents the minimum number of bits required to describe all LSPs per frame; hence, the total redundancy (per frame) in the LSPs is provided by

$$\rho_T = 30 - H_F(\text{bits/frame}).$$

We measure H_F by observing a long training sequence and matching the observation to a particular model of a random process. We use the following two model for $U_{i,j}$:

- Model A assumes that the LSPs in two different frames are independent, and the LSPs within a frame form a first-order Markov chain. More specifically, it assumes that

$$\Pr(\mathbf{U}_j = \mathbf{u}_j | \mathbf{U}_1 = \mathbf{u}_1, \mathbf{U}_2 = \mathbf{u}_2, \ldots, \mathbf{U}_{j-1} = \mathbf{u}_{j-1}) = \Pr(\mathbf{U}_j = \mathbf{u}_j),$$

and

$$\begin{aligned}\Pr(U_{i,j} = u_{i,j} | U_{1,j} = u_{1,j}, \ldots, U_{i-1,j} = u_{i-1,j}) \\ &= \Pr(U_{i,j} = u_{i,j} | U_{i-1,j} = u_{i-1,j}), \\ &= P_A^{(i)}(u_{i,j} | u_{i-1,j}), \end{aligned} \tag{6.11}$$

$i = 2, 3, \ldots, 10$, $j = 1, 2, \ldots$. For $i = 1$, the above becomes $P_A^{(1)}(u_{1,j})$.

- Model B is more complex; it assumes a second-order Markov structure – that $U_{i,j}$ is independent of all the LSPs that came before it conditioned on knowing $U_{i-1,j}$ and $U_{i,j-1}$ – the LSP that immediately precedes it in the same frame and the corresponding LSP in the frame immediately preceding. More specifically, it assumes

$$\begin{aligned}\Pr(U_{i,j} = u_{i,j} | \mathbf{U}_1 = \mathbf{u}_1, \ldots, \mathbf{U}_{j-1} = \mathbf{u}_{j-1}, U_{1,j} = u_{1,j} \ldots, U_{i-1,j} = u_{i-1,j}) \\ &= \Pr(U_{i,j} = u_{i,j} | U_{i,j-1} = u_{i,j-1}, U_{i-1,j} = u_{i-1,j}), \\ &= P_B^{(i)}(u_{i,j} | u_{i,j-1}, u_{i-1,j}), \end{aligned} \tag{6.12}$$

Redundancy	ρ_D	ρ_M	ρ_T
LSP 1	0.6816	0.0000	0.6816
LSP 2	0.4804	0.4415	0.9219
LSP 3	0.7566	0.4425	1.1991
LSP 4	0.7093	0.4303	1.1396
LSP 5	0.3495	0.7184	1.0679
LSP 6	0.3585	0.6287	0.9872
LSP 7	0.6764	0.7575	1.4339
LSP 8	0.4511	0.3521	0.8032
LSP 9	0.2953	0.3840	0.6793
LSP 10	0.5160	0.4377	0.9537
Frame Redundancy	5.2747	4.5927	9.8674

Table 6.2: Redundancy (in bits/frame) results for Model A using 83826 frames of the TIMIT speech database.

$i = 2, 3, \ldots, 10$, $j = 2, 3, \ldots$. For $i = 1$, the above becomes $P_B^{(1)}(u_{1,j}|u_{1,j-1})$.

Since the source is assumed stationary, the transition probabilities in (6.11) and (6.12) are independent of j.

Since every frame is either treated as independent (model A) or Markov (model B), the corresponding entropy rate can be easily estimated from the training sequence. The results are shown in Table 6.2 and Table 6.3, where ρ_D is the redundancy due to the non-uniform distribution of the LSPs, and ρ_M is the redundancy due to the memory (or correlation).

Clearly, substantial redundancy exists within the LSPs. Using Model B as

Redundancy	ρ_D	ρ_M	ρ_T
LSP 1	0.6816	0.2765	0.9581
LSP 2	0.4804	0.8469	1.3273
LSP 3	0.7566	0.7624	1.5190
LSP 4	0.7093	0.7529	1.4622
LSP 5	0.3495	0.8986	1.2481
LSP 6	0.3585	0.9367	1.2952
LSP 7	0.6764	0.8144	1.4908
LSP 8	0.4511	0.7990	1.2501
LSP 9	0.2953	0.6224	0.9177
LSP 10	0.5160	0.5007	1.0167
Frame Redundancy	5.2747	7.2105	12.4852

Table 6.3: Redundancy (in bits/frame) results for Model B using 83826 frames of the TIMIT speech database.

our guide, there are only 17.515 bits of "real" information buried in the 30 high-order LSP bits.

6.4.2 Four Soft-Decision Decoding Algorithms

We use a $(15, 10)$ code $\mathbf{C}$ over $GF(8)$ to encode the 30 LSPs of each frame. This code is a concatenation of a $(9, 6)$ extended Reed-Solomon code and a $(6, 4)$ shortened Reed-Solomon code. It has a generator matrix

$$G = \begin{pmatrix} G_1 & 0 \\ 0 & G_2 \end{pmatrix},$$

where

$$G_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & \alpha^4 & \alpha^5 & \alpha^2 \\ 0 & 1 & 0 & 0 & 0 & 0 & \alpha & \alpha^3 & \alpha^4 \\ 0 & 0 & 1 & 0 & 0 & 0 & \alpha^4 & 1 & \alpha^6 \\ 0 & 0 & 0 & 1 & 0 & 0 & \alpha^2 & \alpha^6 & \alpha \\ 0 & 0 & 0 & 0 & 1 & 0 & \alpha^2 & 1 & \alpha^3 \\ 0 & 0 & 0 & 0 & 0 & 1 & \alpha & 1 & \alpha^5 \end{pmatrix}$$

and

$$G_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & \alpha^3 & \alpha^4 \\ 0 & 1 & 0 & 0 & \alpha^3 & \alpha^5 \\ 0 & 0 & 1 & 0 & \alpha & \alpha^5 \\ 0 & 0 & 0 & 1 & \alpha & \alpha^4 \end{pmatrix}.$$

(Here, α is a primitive root of $x^4 + x + 1 = 0$.)

As in the scheme in Chapter 5, we assume a frame (after all overhead is included) consists of 240 bits. We use bit interleaving, and interleaving is performed within each frame. Every interleaved frame is transmitted through a

Rayleigh fading channel via DQPSK modulation. We also assume the CSI and N_0 are known to the receiver. Let r_j be the received signal (consisting of two adjacent modulation signals) corresponding to bit c_j of a codeword $\mathbf{c} = [c_0, c_1, ..., c_{44}]$. From Section 4.3, the bit metric is derived by hypothetical transmitted signals, and the conditional probability $p(r_j|c_j)$ is regarded as (subject to a constant scaling) $I_0(2Aa_1\ell^*_{c_j}/N_0)$, where c_j may be a LSB (see Eq. (4.12)-(4.13)) or a MSB. In Chapter 4, the weighted bit metric (for ML decoding) of bit c_j is defined as $a_1\ell^*_{c_j}$; alternatively, we can also define the bit metric for position j as $\mu_j = Aa_1(\ell^*_{c_j=1} - \ell^*_{c_j=0})/N_0$.

We now describe the four soft-decision decoding algorithms that make increasing use of the residual redundancy present in the LSPs.

- **SDD 1:** This decoding algorithm tries to perform ML decoding (i.e., not to exploit any residual redundancy). Since no efficient ML soft-decision decoding algorithms for Reed-Solomon codes have been found so far, we adopt a near-maximum likelihood decoding algorithm which makes combined use of the Chase algorithm and syndrome decoding. The decoding procedure can be divided into two stages:

  - Stage 1 consists of generating a list of candidates that will "compete" to be the decoder's estimate of the transmitted codeword. We flip the b least confident bits to form $P = 2^b$ different 15-tuple on $GF(8)$; for each 15-tuple, we arrive at L different estimates of the noise, corresponding to the L lowest-weight vectors in the same coset as the 15-tuple. The net result is at most $N = PL$ codeword candidates. (It is possible that not all of the N codeword candidates are different.)

– Stage 2 consists of selecting the best codeword candidate; i.e., find the codeword $\mathbf{c}^*$ so that

$$\mathbf{c}^* = \arg\max \left\{ \sum_{j=0}^{45} (-1)^{c_j+1} \mu_j : \mathbf{c} = [c_0, c_1, \ldots, c_{44}] \text{ is on list.} \right\}$$

- **SDD 2:** This algorithm is identical to SDD 1 with one exception: The ordering property of LSPs is taken into account. More specifically, during Stage 1 we now generate a list of at most N codewords corresponding to **ordered** LSPs; this is done by choosing up to N such codewords or until P_{max} has been reached, whichever occurs first. Stage 2 consists, once again, of choosing the codeword from the list that maximizes the cumulative metric. If the list is empty, we simply repeat the LSPs from the previous frame.

- **SDD 3:** This algorithm is near-maximum a posteriori (near-MAP) decoding. It is identical to SDD 2 except that, during Stage 2, we select the codeword $\mathbf{c}^*$ so that

$$\mathbf{c}^* = \arg\max \left\{ \sum_{j=0}^{45} (-1)^{c_j+1} \mu_j + \ln(P(\mathbf{c})) : \mathbf{c} = [c_0, c_1, \ldots, c_{44}] \text{ is on list.} \right\}.$$

- **SDD 4:** This algorithm seeks to exploit both the intra-frame and the inter-frame redundancy (i.e., exploiting the source Model B). Let $\underline{\mathbf{R}} = [\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_j]$ be a sequence of random variables representing the received CELP frames, and $\underline{\mathbf{r}} = [\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_j]$ be the observation. The MAP estimate of the j^{th} codeword given the observation $\underline{\mathbf{r}}$ is

$$\begin{aligned} \mathbf{c}^* &= \arg\max_{\mathbf{c}\in\mathcal{C}} P(\mathbf{C}_j = \mathbf{c} | \mathbf{R}_1 = \mathbf{r}_1, \mathbf{R}_2 = \mathbf{r}_2, ..., \mathbf{R}_j = \mathbf{r}_j) \\ &= \arg\max_{\mathbf{c}\in\mathcal{C}} P_{\underline{\mathbf{R}}|\mathbf{C}_j}(\underline{\mathbf{r}}|\mathbf{c}) P(\mathbf{C}_j = \mathbf{c}), \end{aligned}$$

where $\mathbf{C}_j$ denotes the j^{th} transmitted codeword. If we define the objective function to be maximized by

$$g^{(j)}(\mathbf{c}) = P_{\underline{\mathbf{R}}|\mathbf{C}_j}(\underline{\mathbf{r}}|\mathbf{c})P(\mathbf{C}_j = \mathbf{c}),$$

then by conditioning on the value of $\mathbf{C}_{j-1}$, it can be shown that

$$g^{(j)}(\mathbf{c}) = \sum_{\mathbf{c}' \in \mathcal{C}} P_{\mathbf{R}_j|\mathbf{C}_j}(\mathbf{r}_j|\mathbf{c})P(\mathbf{C}_j = \mathbf{c}|\mathbf{C}_{j-1} = \mathbf{c}')g^{(j-1)}(\mathbf{c}').$$

The above equation implies that the maximum objective function can be computed recursively. Furthermore, define $P^*_{\mathbf{R}_j|\mathbf{C}_j}(\mathbf{r}_j|\mathbf{c})$ as an approximation to $P_{\mathbf{R}_j|\mathbf{C}_j}(\mathbf{r}_j|\mathbf{c})$:

$$P^*_{\mathbf{R}_j|\mathbf{C}_j}(\mathbf{r}_j|\mathbf{c}) = \begin{cases} P_{\mathbf{R}_j|\mathbf{C}_j}(\mathbf{r}_j|\mathbf{c}), & \text{if } \mathbf{c} \in A(\mathbf{r}_j) \\ 0 & \text{otherwise;} \end{cases} \tag{6.13}$$

where $A(\mathbf{r}_j)$ is the list of N modulated codeword candidates obtained in Stage 1 of the SDD 2 algorithm, given the j^{th} received vector $\mathbf{r}_j$. We then get the simplified objective function:

$$g^{(j)}(\mathbf{c}) = \begin{cases} \sum_{\mathbf{c}' \in A(\mathbf{r}_{j-1})} P^*_{\mathbf{R}_j|\mathbf{C}_j}(\mathbf{r}_j|\mathbf{c})P(\mathbf{C}_j = \mathbf{c}|\mathbf{C}_{j-1} = \mathbf{c}')g^{(j-1)}(\mathbf{c}'), & \text{if } \mathbf{c} \in A(\mathbf{r}_j) \\ 0 & \text{otherwise.} \end{cases}$$

This, coupled with the fact that the conditional probabilities $P(\mathbf{C}_j = \mathbf{c}|\mathbf{C}_{j-1} = \mathbf{c}')$, gives us an iterative decoder that exploits both the inter-frame and the intra-frame correlation present in the LSPs.

6.4.3 Simulation Results

Simulation results for the four algorithms as well as hard-decision decoding (HDML) are shown in this subsection. The underlying channel is a Rayleigh fading channel with a correlation coefficient $\xi = 0.01$.

In evaluating the performance of the various decoders we use two criteria. The first is the average spectral distortion (SD), the most commonly used distortion measure for the LSP parameters [73]. More specifically,

$$SD = \frac{1}{T}\sum_{t=1}^{T}\left[\int_{-\pi}^{\pi}\left(10\log_{10} S_t(w) - 10\log_{10}\hat{S}_t(w)\right)^2 \frac{dw}{2\pi}\right]^{1/2} \quad \text{dB},$$

where $S_t(w)$ and $\hat{S}_t(w)$ are, respectively, the original and reconstructed spectra associated with frame t, and T is the total number of frames. It should be noted that the spectral distortion introduced by the CELP's scalar quantizer alone (when the channel is noiseless) is approximately 1.50 dB. The second measure of the decoder's performance is the symbol error rate[3] – i.e., the fraction of LSPs the decoder decoded incorrectly.

Figure 6.9 depicts the average spectral distortion (SD) and symbol error rates.

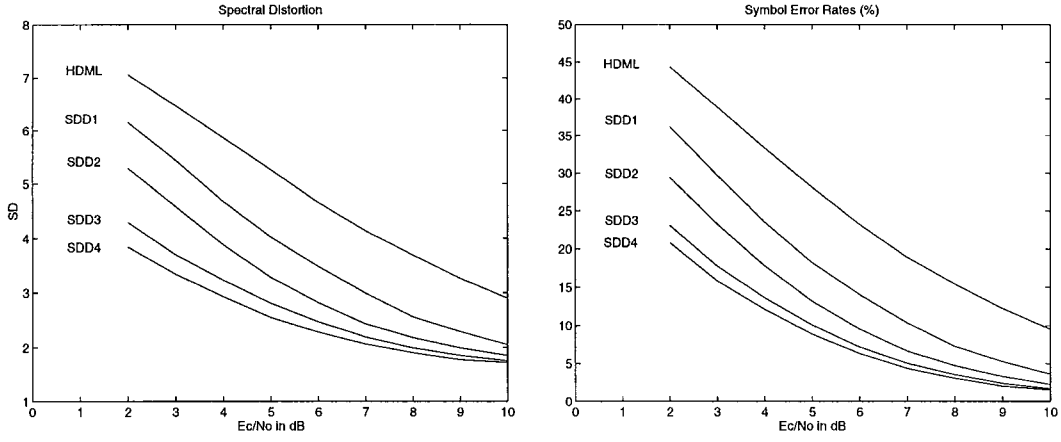


Figure 6.9: Average spectral distortion (SD) and symbol error rate; $N = 64$, $L = 64$ and $P_{max} = 64$; (a) Spectral distortion; (b) Symbol error rate

This figure clearly shows that SDD 3 and SDD 4 substantially outperform

[3]Here, a symbol means a Reed-Solomon codeword element from GF(8).

SDD 1 (ML) and SDD 2 especially when the channel is very noisy. This is because SDD 3 exploits the entire intra-frame redundancy described by Model A, and SDD 4 exploits both intra-frame and inter-frame redundancy described by Model B ($\approx$ 12.5 bits per frame).

In addition to the objective performance measurement presented in Figure 6.9, a set of subjective listening tests are also performed. From the listening tests, we also conclude that there is significant decoding performance improvement by exploiting the source residual redundancy.

6.5 Summary

We have investigated the problem of reliably transmitting sources that contain some residual redundancy so that better decoding results by MAP decoding can be obtained. It is difficult to devise a general MAP decoding algorithm which can be applicable to every source statistic and every coding scheme; therefore, we first use two simple source models (i.e., independent and Markov models) to illustrate the use of the residual redundancy and demonstrate the performance improvement by MAP decoding over ML decoding. If the information bits are independent but not uniformly distributed, the traditional ML metrics can be easily modified into MAP metrics when the encoding function is systematic, and the MAP decoding algorithm is almost identical to the ML decoding. If the information bits can be modeled as a Markov chain, a trellis-based ML decoding algorithm (e.g., the Viterbi algorithm) can be modified into an MAP decoding algorithm. For block codes without simple trellis structures, we propose a simplified decoding procedure which makes combined use of the Markov transition

trellis and the chasing concept.

The performance of MAP decoding based on pair-wise error probabilities is derived for DQPSK on both AWGN channels and Rayleigh fading channels. For the performance of a real coding scheme, we use the extended Golay code in simulation and observe that MAP decoding can indeed achieve substantial performance gain over the traditional ML decoding when the source redundancy is large.

We apply the concept of MAP decoding in a practical mobile radio scheme which uses the FS 1016 CELP vocoder. We quantify the redundancy left in the LSPs, and find that there is significant redundancy left in the LSPs. Four soft-decision decoding algorithms are applied, each of which makes increasing use of the residual redundancy. Simulation results show that the decoder's performance can be improved significantly (approximately 3 dB at low SNR) when the source redundancy is exploited by the receiver.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

This thesis presents some techniques for enhancing the receiver's performance under DPSK modulation for wireless voice communications.

First, the information theoretic limits of fading channels under DPSK modulation are developed. In particular, the channel capacity and cutoff rates are calculated. The theoretic limits are calculated based on the availability of the CSI; generally, knowledge of CSI helps yield a better capacity and cutoff rate. For DPSK, we find that the knowledge of the signal amplitude is not critical to the receiver, but the knowledge of phase distortion (or random FM) may play an important role in the receiver's performance. If the random FM is very small (which is true for many systems), it can be regarded as zero without noticeable performance degradation; on the other hand, when it cannot be ignored (i.e., under fast fading), the DPSK receiver cannot perform well without knowledge of it. In deriving the cutoff rates, we adopt various decoding metrics and determine the resultant cutoff rates. The results can be used to compare the performance of various metrics without employing any specific code.

The performance of the various symbol metrics of Chapter 3 are further investigated in Chapter 4, where their performance on both AWGN and fading channels is analyzed. In addition, we also introduce a bit metric for DQPSK and compare it to some traditional metrics. Bit metrics are important because only they can be calculated under bit interleaving. For DQPSK, we show that dibit interleaving is (slightly) better than bit interleaving on AWGN channels; however, bit interleaving is better than dibit interleaving on Rayleigh fading channels (especially at medium to high SNR).

Chapter 5 investigates the problem of error concealment under ML soft-decision decoding. We assume the CSI is not known and define a normalized codeword reliability (NCR) as a measurement for detecting unreliable codeword estimates. The NCR is coupled with a given interpolation algorithm to design the rule that determines which codeword is reliable. We use a practical land mobile radio system to demonstrate that error concealment can lower an operational SNR range by 3 dB or even more.

Chapter 6 considers the problem of exploiting the residual redundancy left in the source to enhance the decoder's performance. We first analyze the performance of MAP decoding for a small code (which consists of only two codewords), and show that a significant gain can be obtained by MAP decoding on both AWGN and fading channels. The gain is particularly apparent at lower SNR range. For performance of real codes, we use simulation to investigate the performance of MAP decoding. By using two simple statistics models for the source, we discuss some basic MAP decoding algorithms for codes with and without simple trellis structures. Simulation results based on the extended Golay code are presented. Of course, different source statistics and different encoding

procedures may change the MAP decoding algorithm. We also use a practical CELP-based land mobile radio system as a practical example to demonstrate the advantage of exploiting the source residual redundancy. We first quantify the residual redundancy of the LSPs from the vocoder output by assuming two probability models for the LSPs. A long training speech is used to measure the residual redundancy, and it is found that approximately one-third of the bits representing LSPs are redundant; hence it is possible to utilize the redundancy to help decode. We use a Reed-Solomon code to protect the LSPs and devise four decoding algorithms, each of which make increasing use of the redundancy. The simulation results show that a 3 dB gain can be achieved over traditional ML decoding, especially at lower SNR.

7.2 Future Work

Based on the results presented in this thesis, the following is a list of possible future work:

(1) The bit metric introduced in Chapter 4 is only for DQPSK. It would be interesting to investigate how the bit metrics for M-ary DPSK ($M \geq 8$) may be defined by extending the DQPSK bit metric.

(2) The problem of tailoring a channel code for a DPSK modulation (namely coded DPSK-modulation) on a fading channel should be investigated. For coherent modulation schemes with CSI, the most important parameter for designing a good code on Rayleigh fading channels is the **minimum time diversity** – i.e., the minimum symbol-wise Hamming distance [31]. If the modulated version of a transmitted code is $\mathbf{s} = \{s_1, s_2, ..., s_n\}$, the Chernoff bound for estimating

code $\hat{\mathbf{s}}$ is

$$P(\mathbf{s} \to \hat{\mathbf{s}}) \leq \prod_{j=1}^{n} \frac{1}{1 + \frac{|s_j - \hat{s}_j|^2}{4N_0}}$$

This equation implies that the minimum time diversity of the code dominates the performance. Interestingly, there is also an analogy in DPSK modulated systems. From Eq.(3.59), the Chernoff bound for estimating codeword $\hat{\mathbf{c}}$ given $\mathbf{c}$ transmitted is

$$P(\mathbf{c} \to \hat{\mathbf{c}}) \leq \prod_{j} \frac{1}{1 + \frac{\xi^2 A^4 T_s^2}{8N_0(N_0 + A^2 T_s) + 2(1-\xi^2)A^4 T_s^2}[1 - \cos(\phi_{c_j} - \phi_{\hat{c}_j})]}$$

This equation also indicates that the minimum time diversity is an important parameter for designing good codes used with DPSK modulation.

(3) DPSK modulation is not a highly spectral efficient modulation scheme; for better spectral efficiency in a channel where fast and accurate carrier tracking is difficult, differential amplitude and phase shift keying (DAPSK) can be a good candidate [76]. Recently, DAPSK has been attracting much attention for use in mobile systems requiring high spectral efficiency such as digital terrestrial video broadcasting (DTVB) [77]-[79]. Since DAPSK can be regarded as an extension of DPSK, it may be possible to extend our discussion in Chapters 3 and 4 to determine the theoretical limits and soft-decision decoding metrics.

(4) Throughout our discussion, we assume only one receiving antenna. In reality (especially at base stations), space diversity is usually exploited to combat fading on the receiver side by using rake receivers or some kind of diversity combining scheme. It would be interesting to extend (modify) the metrics discussed in this thesis for diversity combining.

(5) In a conventional "concatenated coding" scheme, a small block code or convolutional code is used as the inner code, and a large code (e.g., Reed-Solomon

code) as the outer code. Usually, soft-decision decoding is applied to the inner code; the estimated codewords are then passed to the outer code for hard-decision decoding. The performance of the concatenated scheme could be improved by calculating the "soft output" (i.e., codeword reliability) from the inner decoder and passing the soft output to the outer code for soft-decision decoding. This problem has been investigated on AWGN channels in literature, e.g., in [80], where the soft-output from the inner code was an estimated symbol error probability. On fading channels, we have shown that the error probability is difficult to obtain when CSI is not known; it would therefore be interesting to apply the concept of NSR to concatenated coding schemes.

(6) In Chapter 6, the speech signal is treated as a stationary process, based on which the LSP transition probabilities are measured. In fact, most practical signals (including speech, image, and video) are not stationary; therefore, different source statistics should be used to describe different segments of a sequence of signals. This leads to adaptive MAP decoding algorithms and may yield even better results, although the system could become more complex. As newer source coders manage to compress more redundancy, less redundancy will be available for MAP decoding. However, it is quite unlikely that a source coder can compress all the redundancy for every segment of the signals; it is possible some significant redundancy may still exist at times, so the concept of adaptive MAP decoding can be applied.

Appendix A

Calculating $E[\cos(nwZ)]$ and $E[\sin(nwZ)]$ for Rician Variables

This appendix gives methods for calculating $E[\cos(nwZ)]$ and $E[\sin(nwZ)]$, where Z is a Rician random variable with probability density

$$f(z) = \frac{z}{\sigma^2} e^{-(z^2+\beta^2)/2\sigma^2} I_0(\frac{\beta z}{\sigma^2}).$$

For convenience, we first provide the m^{th} moment of Z, i.e., $E[Z^m]$. By definition,

$$E[Z^m] = \int_0^\infty z^m \cdot \frac{z}{\sigma^2} e^{-(z^2+\beta^2)/2\sigma^2} I_0(\frac{\beta z}{\sigma^2}) dz.$$

By expanding $I(\cdot)$ into an infinite series and taking term-wise integration, it can be verified [55] that

$$E[Z^m] = (2\sigma^2)^{m/2} \Gamma(\frac{m}{2} + 1) {}_1F_1(-m/2; 1; -\alpha^2/2) \tag{A.1}$$

where $\alpha^2 = \beta^2/\sigma^2$, $\Gamma(\cdot)$ is the gamma function, and ${}_1F_1(\cdot;\cdot;\cdot)$ is the confluent hypergeometric function.

To calculate $E[\cos(nwz)]$, we first write

$$\cos(nwz) = 1 - \frac{(nwz)^2}{2!} + \frac{(nwz)^4}{4!} - \frac{(nwz)^6}{6!} \cdots$$

and take expectation on both sides. Using the moment of Rician variables and the fact $\Gamma(k) = (k-1)!$ for positive integer k, it is straightforward to verify that

$$E[\cos(nwZ)] = 1 + \sum_{\substack{m=2 \\ m\ even}} (-1)^{m/2} \frac{(nw\sqrt{2}\sigma)^m}{m \cdot (m-1)...(m/2+1)} {}_1F_1(-m/2; 1; -\alpha^2/2).$$

Similarly,

$$\sin(nwz) = nwz - \frac{(nwz)^3}{3!} + \frac{(nwz)^5}{5!} - \frac{(nwz)^7}{7!}...$$

Taking expectation on both sides, and using the fact that

$$\Gamma(k + \frac{1}{2}) = \frac{1 \cdot 3 \cdot 5...(2k-1)}{2^k} \Gamma(\frac{1}{2}),$$

we have

$$\begin{aligned} E[\sin(nwZ)] &= \frac{nw\sqrt{2}\sigma}{2} \Gamma(\frac{1}{2}) {}_1F_1(-m/2; 1; -\alpha^2/2) + \\ &\sum_{\substack{m=3 \\ m\ odd}} (-1)^{(m-1)/2} \frac{(nw\sqrt{2}\sigma)^{(m-1)/2}}{2 \cdot 4 \cdot 6...(m-1)} (nw\sqrt{2}\sigma)^{(m+1)/2} \Gamma(\frac{1}{2}) {}_1F_1(-m/2; 1; -\alpha^2/2). \end{aligned}$$

Appendix B

A Few Equations Useful for Calculating Pair-Wise Error Probabilities Using Square Metrics

This appendix presents the parameter definitions adopted from Appendix 4B of [3]. A couple of equations from [3] which are referred to in this thesis are also listed here.

Let $D = \sum_{k=1}^{L}(|X_k|^2 - |Y_k|^2)$, where X_k and Y_k are two complex Gaussian random variables. It is assumed that the autocorrelation of X_k (denoted as μ_{XX}) and Y_k (denoted as μ_{YY}), and the cross-correlation (denoted as μ_{XY}) do not depend on k. (For clarity, $\mu_{XY} = E[(X_k - \bar{X_k})(Y_k - \bar{Y_k})^*]$.)

The probability $P_e = Prob(D < 0)$ is to be determined. Define the following parameters:

$$w = \frac{\mu_{XX} - \mu_{YY}}{2(\mu_{XX}\mu_{YY} - |\mu_{XY}|^2)}$$

$$v_1 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY} - |\mu_{XY}|^2}} - w$$

$$v_2 = \sqrt{w^2 + \frac{1}{\mu_{XX}\mu_{YY} - |\mu_{XY}|^2}} + w$$

$$\alpha_{1k} = |\mu_{XX}|^2\mu_{YY} + |\mu_{YY}|^2\mu_{XX} - \bar{X_k}^*\bar{Y}_k\mu_{XY} - \bar{X}_k\bar{Y_k}^*\mu^*_{XY}$$

$$\alpha_{2k} = |\bar{X}_k|^2 - |\bar{Y}_k|^2$$

$$\alpha_1 = \sum_{k=1}^{L}\alpha_{1k} \quad \alpha_2 = \sum_{k=1}^{L}\alpha_{2k}$$

$$A_2 = \frac{v_1^2 v_2}{v_1+v_2}(\alpha_1 v_1 + \alpha_2)$$

$$A_3 = \frac{v_1 v_2^2}{v_1+v_2}(\alpha_1 v_2 - \alpha_2)$$

$$a = \left[\frac{2v_1^2 v_2(\alpha_1 v_2 - \alpha_2)}{(v_1+v_2)^2}\right]^{1/2}$$

$$b = \left[\frac{2v_1 v_2^2(\alpha_1 v_1 + \alpha_2)}{(v_1+v_2)^2}\right]^{1/2}$$

The probability P_e can be written in any of the following three forms:

1)

$$P_e = -\frac{(v_1v_2)^L}{2\pi i}\int_{-\infty+\epsilon}^{\infty+\epsilon}\frac{dz}{z(z+iv_1)^L(z-iv_2)^L}\exp\left[\frac{v_1v_2(iz\alpha_2 - z^2\alpha_1)}{(z+iv_1)(z-iv_2)}\right] \qquad \text{(B.1)}$$

where ϵ is a small positive number.

2)

$$P_e = \frac{\exp\left[v_1v_2(-2\alpha_1v_1v_2 + \alpha_2v_1 - \alpha_2v_2)/(v_1+v_2)^2\right]}{(1+v_2/v_1)^{2L-1}}\frac{1}{2\pi i}\int_\Gamma f(z)dz \qquad \text{(B.2)}$$

where Γ is a circular contour of radius less than unity that encloses the origin and

$$f(z) = \frac{[1+(v_2/v_1)z]^{2L-1}}{z^L(1-z)}\exp\left(\frac{A_2(v_2/v_1)}{v_1+v_2}z + \frac{A_3(v_1/v_2)}{v_1+v_2}\frac{1}{z}\right).$$

3) (for $L > 1$)

$$\begin{aligned} P_e &= Q(a,b) - I_0(ab)\exp\left(-\frac{a^2+b^2}{2}\right) \\ &\quad + \frac{I_0(ab)\exp[-(a^2+b^2)/2]}{(1+v_2/v_1)^{2L-1}}\sum_{k=0}^{L-1}\binom{2L-1}{k}\left(\frac{v_2}{v_1}\right)^k \end{aligned}$$

$$+\frac{\exp[-(a^2+b^2)/2]}{(1+v_2/v_1)^{2L-1}}\sum_{n=1}^{L-1} I_n(ab)$$
$$\times\left\{\sum_{k=0}^{L-1-n}\binom{2L-1}{k}\left[\left(\frac{b}{a}\right)^n\left(\frac{v_2}{v_1}\right)^k-\left(\frac{a}{b}\right)^n\left(\frac{v_2}{v_1}\right)^{2L-1-k}\right]\right\} \quad \text{(B.3)}$$

where $Q(\cdot,\cdot)$ is the Marcum Q function.

Bibliography

[1] J. Hagenauer, "Source-Controlled Channel Decoding," *IEEE Transactions on Communications*, vol. 43, September 1995.

[2] V. Cox, J. Hagenauer, N. Seshadri, and C. E. W. Sundberg, "Subband Speech Coding and Matched Convolutional Channel Coding for Mobile Radio Channels," *IEEE Trans. Signal Processing*, vol. 39, August 1991.

[3] J. G. Proakis, *Digital Communications*, McGraw-Hill 1989.

[4] S. Lin and D. J. Costello, *Error Control Coding: fundamentals and applications*, Prentice-Hall, 1983.

[5] M. D. Yacoub, *Foundations of Mobile Radio Engineering*, CRC press, Inc. 1993.

[6] National Communications System (NCS), "Details to Assist in Implementation of Federal Standard 1016 CELP," *Technical Information Bulletin 92-1*, Office of the Manager, NCS, Arlington, VA 22204

[7] A. Bateman, "Feedforward Transparent Tone-in-band, its Implementation and Applications," *IEEE Transactions on Vehicular Technology*, August 1990.

[8] M. L. Moher and J. H. Lodge, "TCMP–A Modulation and Coding Strategy for Rician Fading Channels," *IEEE Journal on Selected Areas in Communications,* vol. 7, no. 9, Dec. 1989.

[9] J. Cavers, "An Analysis of Pilot Symbol Assisted Modulation for Rayleigh Fading Channels," *IEEE Transactions on Vehicular Technology*, November 1991.

[10] S. Sampei and T. Sunaga, "Rayleigh Fading Compensation for QAM in Land Mobile Radio Communication," *IEEE Transactions on Vehicular Technology*, May 1993.

[11] G. D. Forney, "Generalized Minimum Distance Decoding," *IEEE Transactions on Communications,* vol. IT-13, pp. 125-131, 1966.

[12] D. Chase, "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information," *IEEE Transactions on Information Theory,* vol. IT-18, pp. 170-182, Jan. 1972.

[13] J. K. Wolf, "Efficient Maximum Likelihood Decoding of Linear Block Codes," *IEEE Transactions on Information Theory,* vol. IT-24, pp. 76-80, 1978.

[14] H. Tanaka and K. Kakigahara, "Simplified Correlation Decoding by Selecting Possible Codewords Using Erasure Information," *IEEE Transactions on Information Theory*, Vol. 29, September 1983.

[15] J. H. Conway and N. J. A. Sloane, "Soft Decoding Techniques for Codes and Lattices, Including the Golay code and the Leech Lattice," *IEEE Transactions on Information Theory,* vol. 32, Jan 1986.

[16] D. J. Taipale and M. B. Pursley, "An Improvement to Generalized-Minimum-Distance Decoding," *IEEE Transactions on Information Theory,* vol. IT-37, no. 1, Jan. 1991.

[17] N. J. C. Lous, P. A. H. Bours and H. C. A. van Tilborg, " On Maximum Likelihood Soft-Decision Decoding of Binary Linear Block Codes," *IEEE Transactions on Information Theory*, Vol. 39, January 1993.

[18] Y. S. Han, C. R. P. Hartman and C. C. Chen, "Efficient Priority-First Search Maximum Likelihood Soft-Decision Decoding of Linear Block Codes," *IEEE Transactions on Information Theory*, Vol. 39, September 1993.

[19] J. Snyders and Y. Be'ery, "Maximum Likelihood Soft Decoding of Binary Block Codes and Decoders for the Golay Code," *IEEE Transactions on Information Theory*, Vol 35, September 1989.

[20] A. Vardy and Y. Be'ery, "Bit Level Soft-Decision Decoding of Reed-Solomon Codes," *IEEE Transactions on Communications,* vol. 39, no. 3, Mar. 1991.

[21] A. Vardy and Y. Be'ery, "More Efficient Soft Decoding of The Golay Codes,"*IEEE Transactions on Information Theory,* vol. 37, no. 3, May 1991.

[22] A. Vardy and Y. Be'ery, "Maximum Likelihood Soft-Decision Decoding of BCH Codes," *IEEE Transactions on Information Theory,* vol. 40, Mar. 1994.

[23] J. T. Wu, S. Lin, T. Kasami, T. Fujiwara and T. takata, "An Upper Bound on the Effective Error Coefficient of Two-Stage Decoding and Good Two-Level Decompositions of Some Reed-Muller Codes," *IEEE Transactions on Communications*, Vol. 42, March 1994.

[24] T. Kasami, T. Takata, T. Yamashita and S. Lin, "Suboptimum Decoding of Decomposable Block Codes," *IEEE Transactions on Information Theory*, Vol. 40, September 1994.

[25] M. P. C. Fossorier and S. Lin, "Soft-Decision Decoding of Linear Block Codes Based on Ordered Statistics," *IEEE Transactions on Information Theory,* vol. 41, no. 5, Sep. 1995.

[26] G. D. Forney, Jr., "The Viterbi Algorithm," *Proc. IEEE*, vol. 61, Mar. 1973.

[27] R. M. Fano, "A Heuristic Discussion of Probabilistic Decoding," *IEEE Transactions on Information theory*, IT-9, April 1963.

[28] F. Jelinek, "A Fast Sequential Decoding Algorithm Using a Stack," *IBM J. Research and Development*, November 1969.

[29] G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals," *IEEE Transactions on Information Theory*, IT-28, January 1982.

[30] E. Biglieri, D. Divsalar, P. J. McLane and M. K. Simon, *Introduction to Trellis-coded Modulation with Applications*, Maxwell Macmillan, 1991.

[31] S. H. Jamali and T. Le-Ngoc, *Coded-Modulation Techniques for Fading Channels*, Kluwer Academic Publishers, 1994.

[32] W. B. Kleijn, R. P. Ramachandran, and P. Kroon, "Interpolation of the Pitch-Predictor Parameters in Analysis-by-Synthesis Speech Coders", *IEEE Transactions on Speech and Audio Processing,* vol. 2, no. 1, part I, Jan. 1994.

[33] R. E. Collins, *Antennas and Radiowave Propagation*, McGraw-Hill, New York, 1985.

[34] S. Stein, "Fading Channel Issues in System Engineering," *IEEE J. on Selected Areas Communications*, SAC-5, February 1987.

[35] A. D. Whalen, *Detection of Signals in Noise*, Academic Press, 1971.

[36] W. C. Y. Lee, *Mobile Communication Engineering*, McGraw-Hill, New York, 1982.

[37] W. C. Jakes, *Microwave Mobile Communications*, John Wiley & Sons, New York, 1974.

[38] G. A. Arredondo, W. H. Chriss and E. H. Walker, "A Multipath Fading Simulator for Mobile Radio," *IEEE Transactions on Vehicular Technology*, VT-22, November 1973.

[39] J. R. Ball, "A Real-Time Fading Simulator for Mobile Radio," *The Radio and Electronic Engineer*, Vol. 52, October 1982.

[40] L. Ehrman, L. B. Bates, J. F. Eschle and J. M. Kates, "Real-Time Software Simulation of the HF Radio Channels," *IEEE Transactions on Communications*, COM-30, August 1982.

[41] J. Hagenauer, N. Seshadri and C. W. Sundberg, "The Performance of Rate-Compatible Punctured Convolutional Codes for Digital Mobile Radio," *IEEE Transactions on Communications,* vol. 38, no. 7, Jul. 1990.

[42] R. S. Kennedy, *Fading Dispersive Communication Channels,* New York: Wiley, 1969.

[43] T. Ericson, "A Gaussian Channel with Slow Fading," *IEEE Transactions on Information Theory*, vol. 16, May 1970.

[44] W. C. Y. Lee, "Estimate of Channel Capacity in Rayleigh Fading Environment," *IEEE Transactions on Vehicular Technology*, vol. 39, Aug. 1990.

[45] R. Buz, "Information Theoretic Limits on Communication Over Multipath Fading Channels with Ideal and Non-Ideal Channel State Information," *preprint.*

[46] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*, New York: Wiley, 1965.

[47] R. E. Blahut, *Principles and Practice of Information Theory*, Addison-Wesley Publishing Company, 1987.

[48] T. Matsumoto and F. Adachi, "Performance Limits of Coded Multilevel DPSK in Cellular Mobil Radio", *IEEE Transactions on Vehicular Technology,* vol. 41, no. 4, Nov 1992.

[49] R. G. Gallager, *Information Theory and Reliable Communication*, 1968.

[50] D. Chase, "A Combined Coding and Modulation Approach for Communication over Dispersive Channels," *IEEE Transactions on Communications,* vol. COM-21, no.3, Mar. 1973.

[51] D. Chase, "Digital Signal Design Concepts for a Time-Varying Rician Channel," *IEEE Transactions on Communications,* vol. COM-24, no.2, Feb. 1976.

[52] G. H. Robertson, "Performance Degradation by Postdetector Nonlinearities, *Bell Syst. Tech. J.* 47, No. 3, 407-414, 1968.

[53] M. K. Simon, "Comment on 'On Binary DPSK Detection'," *IEEE Transactions on Communications,* vol. COM-26, no. 10, Oct. 1978.

[54] M. Abramowitz and I. E. Stegun, *Handbook of Mathematical Functions.* National Bureau of Standard, Washington DC, 1972.

[55] S. O. Rice, "Mathematical Analysis of Random Noise." *Selected Papers on Noise and Stochastic Processes*, Dover, New York, 1954.

[56] J. K. Cavers and P. Ho, "Analysis of the Error Performance of Trellis-Coded Modulation in Rayleigh-Fading Channels," *IEEE Transactions on Communicatins,* vol. 40, no. 1, Jan 1992.

[57] N. C. Beaulieu, "An Infinite Series for Computation of the Complememtary Probability Distribution Function of a Sum of Independent Random Variables and Its Application to the Sum of Rayleigh Random Variables," *IEEE Transactions on Communications,* vol. 38, no. 9, Sep. 1990.

[58] B. S. Atal and M. R. Schroeder, "Adaptive Predictive Coding of Speech Signals," *Bell Syst. Tech. J.*, vol. 49, 1970.

[59] P. Kroon and B. S. Atal, "Predictive Coding of Speech Using Analysis-by-Synthesis Techniques," *Review of Advances in Speech Signal Processing*, New York: Marcel Dekker, 1991.

[60] W. B. Kleijn and W. Granzow, "Methods for Waveform Interpolation in Speech Coding," *Digital Signal Processing*, vol. 1, no. 4, 1991.

[61] D. J. Rahikka, *et al*, "CELP Coding for Land Mobile Radio Applications," *ICASSP'90*, Albuquerque, New Mexico, Apr. 1990.

[62] E. Bracha, N. Farvardin, and Y. Yesha, "Channel Coding for CELP Encoded Speech in Mobile Radio Applications," *Technical Reports,* Digital Signal Processing Group, Advanced Systems Division, Computer Systems Laboratory, NIST, MD, Aug. 1994.

[63] T. P. Barnwell, III, "Correlation Analysis of Subjective and Objective Measures for Speech Quality," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Apr. 1980.

[64] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Englewood Cliffs, NJ: Prentice-Hall, 1984.

[65] C. E. Shannon, "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, vol. 27, 1948.

[66] J. Hagenauer and P. Hoeher, "A Viterbi Algorithm with Soft-Decision Output and Its Applications," in *Proc. GLOBECOM'89*, Dallas, TX, November 1989.

[67] J. Hagenauer, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Transactions on Information Theory*, vol. 42, March 1996.

[68] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," in *Proc. ICC'93*, Geneva, Switzerland, May 1993.

[69] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Transactions on Information Theory*, vol. IT-20, March 1974.

[70] G. D. Forney, "Coset Codes–Part II: Binary Lattices and Related Codes," *IEEE Transactions on Information Theory,* vol. 34, no. 5, Sep. 1988.

[71] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices, and Groups*, New York: Springer-Verlag, 1987.

[72] R. T. Curtis,"A New Combinatorial Approach to M_{24}", *Math. Proc. Cambridge Phil. Soc.*, vol 79, pp 25-42, 1976.

[73] R. Laroia, N. Phamdo and N. Farvardin, "Robust and Efficient Quantization of Speech LSP Parameters Using Structured Vector Quantization," *Proceedings ICASSP-91*, pp. 661-664, 1991.

[74] F. Alajaji, N. Phamdo and T. Fuja, "Channel Codes that Exploit the Residual Redundancy in CELP-Encoded Speech," *IEEE Transactions on Speech and Audio Processing,* Vol. 4, no. 5, September 1996.

[75] N. Phamdo, F. Alajaji, S.-I Chen and T. Fuja, "Source-Dependent Channel Coding of CELP Speech over Land Mobile Radio Channels," *MILCOM*, San Diego, Nov. 1995.

[76] W. J. Weber, "Differential Encoding for Multiple Amplitude and Phase Shift Keying Systems," *IEEE Transactions on Communications*, vol. COM-26, March 1978.

[77] R. Monnier, J. B. Rault, T. de Couasnon, "Digital Television Broadcasting with High Spectral Efficiency," *International Broadcasting Convention*, Amsterdam, 1992.

[78] V. Engels, H. Rohling, "Multilevel Differential Modulation Techniques (64-DAPSK) for Multicarrier Transmission Systems," *European Transactions on Telecommunication*, vol. 6, no. 6, Nov.-Dec. 1995.

[79] H. Rohling, V. Engels, "Differential Amplitude Phase Shift Keying (DAPSK) - A New Modulation Methods for DTVB," *International Broadcasting Convention*, Amsterdam, 1995.

[80] X. Wang and S. B. Wicker, "A Soft-Output Decoding Algorithm for Concatenated Systems," *IEEE Transactions on Information Theory*, Vol. 42, no. 2, March 1996.