國 立 交 通 大 學

電信工程學系

博 士 論 文

結合錯誤更正與通道估計的系統化編碼設計及其最大概度解碼

Systematic Code Design for Combined Channel Estimation and Error Correction and Its Maximum-Likelihood Decoding

研 究 生： 吳佳龍

指導教授： 陳伯寧博士

中 華 民 國 九 十 八 年 七 月

結合錯誤更正與通道估計的系統化編碼設計及其最大概度解碼

Systematic Code Design for Combined Channel Estimation and Error Correction and Its Maximum-Likelihood Decoding

研究生：吳佳龍　　Student: Chia-Lung Wu

指導教授：陳伯寧　博士　　Advisor: Dr. Po-Ning Chen

國立交通大學

電信工程學系

博士論文

A Dissertation
Submitted to Institute of Communication Engineering
College of Electrical and Computer Engineering
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in
Communication Engineering
Hsinchu, Taiwan

2009 年 7 月

結合錯誤更正與通道估計的系統化編碼設計及其最大概度解碼

研究生: 吳佳龍　　　　　　　　　指導教授: 陳伯寧

國立交通大學電信工程學系

摘要

傳統通訊系統是使用獨立的訓練序列於接收機來估計通道參數，之後再使用該通道參數來對錯誤更正碼進行解碼。在某些應用上，接收機通道估計值可能需要耗費極高的計算複雜度來獲取，即便如此甚至可能還不能達到所需要的精準估計。因此，非同調系統在此時就成為一個解決方案。基於上述的背景，本論文討論當傳送機與接收機兩者皆對通道參數完全一無所知的環境下之通道編碼設計，我們也提出一個相對應的有效率解碼演算法。

事實上，結合通道估計與錯誤更正的編碼方式最近受到相當的關注並且被視為對抗多路徑衰減的重要技術之一。與一般具有個別獨立的通道估計與錯誤更正裝置相較，在相同的碼率下，模擬證明結合考量通道估計的編碼設計可以顯著增進系統效能。然而，這類編碼的實際使用有個主要障礙，由於目前設計都是經由電腦搜尋而來，以至於所獲得的碼是不具結構，故亦無法有效率的解碼，導致複雜度最高的完全搜尋演算法成為唯一的解碼方法，所以解碼複雜度將會隨著碼的增長而巨幅增加。在本論文中，我們提出一個系統化的建碼方法來設計具有明確結構的結合通道估計與錯誤更正的編碼，用於多輸入多輸出通道的延伸設計也將會被討論。模擬顯示我們所提出的編碼與電腦搜尋所獲得最佳碼兩者效能幾乎不分軒輊。再者，基於系統化建碼所具備的結構，我們可以推導

出具有遞迴關係的最大概度解碼量度，進而可以使用以碼樹爲基礎的循序解碼演算法來做最大概度解碼，因此可以避免使用完全搜尋解碼而大幅降低解碼複雜度。

Systematic Code Design for Combined Channel Estimation and Error Correction and Its Maximum-Likelihood Decoding

Student: Chia-Lung Wu   Advisor: Po-Ning Chen

Institute of Communications Engineering

National Chiao-Tung University

Abstract

A traditional communication system uses separate training sequence for the estimation of channel state information (CSI) at the receiver. This channel estimation will then be used as a base for error correction through channel codes. In applications that channel estimation at the receiver is either of infeasibly high complexity or statistically impossible, noncoherent system design apparently becomes the due selection. At this background, we study the coding scheme that can be applied in an environment that the channel coefficients are completely unknown to both the transmitters and receivers. We subsequently investigate efficient decoding algorithms for our proposed codes.

In fact, the coding technique that combines channel estimation and error correction has received attention recently, and has been regarded as a promising approach to counter the effects of multi-path fading. It has been shown by simulation that a proper code design that jointly considers channel estimation can improve the system performance subject to a fixed code rate as compared to a conventional system which performs channel estimation and error correction separately. Nevertheless, the major obstacle that prevents the practice of such coding technique is that the existing codes are mostly searched by computers, and subsequently exhibit no apparent structure for efficient decoding. Hence, the operation-intensive exhaustive search becomes the only decoding option, and the decoding complexity increases dramatically with codeword length. In this dissertation, a systematic construction is derived for a class of structured codes that support joint channel estimation and error correction. The extension designs that take into consideration the varying characteristic of channels and multiple-input multiple-output channels are also discussed. Simulations show that our codes have comparable performance to the best simulated-annealing-based computer-searched codes. Moreover, the systematically constructed codes can now be maximum-likelihoodly decoded with respect to the unknown-channel criterion in terms of a newly derived recursive metric for use by the priority-first search decoding algorithm. Thus, the decoding complexity is significantly reduced as compared with that of an exhaustive decoder.

Acknowledgements

I have been extremely fortunate to have had Prof. Po-Ning Chen as my supervisor during my graduate studies. His enthusiasm for research and dedication to both my professional and personal developments are truly inspiring. It has been a great honor to be advised by Prof. Yunghsiang S. Han during the joint meeting between our lab and his lab. I have benefited enormously from his comments and suggestions on most of the chapters in this thesis. I sincerely thank them for their great guidance and continuous support throughout my graduate studies at National Chiao-Tung University.

I also thank my groupmates for their support and assistance throughout my time in Network Technology Laboratory. Special thanks go to Prof. Stefan Moser and Mr. Daniel Greenhoe for helping me to revise my paper submitted to IEEE Transaction Information Theory.

I would like to thank Prof. M. Skoglund, Dr. J. Giese and Prof. S. Parkvall of the Royal Institute of Technology (KTH), Stockholm, Sweden, for kindly providing us their codes for further study in the preparation of this disseration. My special thanks go to Prof. Skoglund for the opportunity to visit his research group in 2008. Financial support of National Science Council of Taiwan is gratefully acknowledged.

Last but not least, I would like to dedicate this thesis to my parents and family for their love, support and encouragement.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Currently, a typical receiver in a wireless communication system performs channel estimation and data estimation separately. The former task estimates channel characteristics based on a known training sequence or pilot, while the latter uses these characteristics to estimate the transmitted coded data.

Recent research results [7,12,29,30] have confirmed that better system performance can be obtained by jointly performing channel and data estimation, as compared to a typical system that performs these tasks separately. In 1994, Seshadri [29] proposed a blind maximum-likelihood sequence estimator (MLSE) that performs the two tasks simultaneously. Skoglund *et al.* [30] later provided a milestone evidence that a code design that jointly considers channel estimation and error correction is able to counter multi-path block fading more efficiently than the approach with a separate error-correcting code and channel estimation scheme. They applied the same idea to a multiple-input multiple-output (MIMO) system as described in a subsequent publication [13]. Related techniques developed for MIMO systems [1,2,16,24] also have substantiated that a joint design that combines channel estimation, channel coding and space-time transmission can improve the system performance over that of a separate design.

Specifically, by computer search, Skoglund *et al.* identified nonlinear codes that support joint channel estimation and error correction in a multi-path block fading channel. Through simulations, they found that a communication system using these nonlinear codes can outperform a typical communication system with perfect channel estimation by 2 dB. Their results hint that a single, perhaps nonlinear, code may improve the transmission rate in a highly mobile environment in which traditional channel estimation becomes technically infeasible. A similar idea was also proposed by [7], and the authors actually named such codes *training codes*.

One of the drawbacks of these joint estimation codes found by computer search is that they lack a systematic structure, and can therefore be decoded only by an operation-intensive exhaustive search. This naturally leads to the research query of how to construct an efficiently decodable code that supports joint channel estimation and error correction.

In this dissertation, this query was resolved firstly by discovering that regardless of the fading statistics, the codeword that maximizes the system signal-to-noise ratio (SNR) must be orthogonal to the delayed version of itself. We termed this property *self-orthogonality*. We secondly found that the code that consists of properly chosen self-orthogonal codewords has a performance comparable to that of the simulated-annealing-based computer-searched code. Because the maximum-likelihood metrics for self-orthogonal codewords can be equivalently transformed into a recursively formulated metric, it is finally shown that these structured codes can be maximum-likelihoodly decoded by the priority-first search algorithm [6, 17, 20, 27], resulting in a decoding complexity significantly smaller than that required by exhaustive decoding.

It is worth mentioning that although the codes selected by computer search in [13] and [30] target unknown channels, for which the channel coefficients are assumed constant within a given coding block, the evaluation of the pair-wise error probability (PEP) criterion does presume knowledge of the channel statistics. Even if the dependence of the code design on channel statistics is relaxed in [7], the pairwise distance criterion proposed therein is still for computer search, and no systematic code design is resulted. The code constructed based on the algorithm we propose, however, is guaranteed to achieve an acceptable system SNR regardless of the statistics of the channel. This suggests that our systematically constructed codes are also suitable in cases where channel blindness becomes a stringent system restriction.

1.2 Contributions

The main part of this dissertation is placed in Chapter 3. Extensions of the key idea in Chapter 3 will be presented in subsequent chapters. For clarity, the contributions of Chapter 3 are briefed as follows.

1. A code of comparable performance to the computer-searched code is constructed according to certain rules so that its code tree can be efficiently and systematically generated (Section 3.2).

2. Efficient recursive computation of the maximum-likelihood evaluation function f from the predecessor path to the successor paths is established (Section 3.3).

This part will appear in *IEEE Transactions on Information Theory* [36] and was presented in parts at the *2008 International Symposium on Information Theory and its Application* [35].

In Chapter 4, with the availability of the above Items 1 and 2, the construction and maximum-likelihood decoding of codes with longer codeword length becomes possible, and hence, makes the assumption that the unknown channel coefficients are fixed during a long decoding block somewhat impractical especially for mobile transceivers. Extension of Items 1 and 2 to the unknown channels whose channel coefficients may change several times during one decoding block is thus proposed in this chapter.

All the previous results concern a frequency selective environment. When a frequency nonselective fading channel is considered, our simulations found that the original code designs in Chapters 3 and 4 do not perform as close as we have expected to the computer-searched best codes. Further investigation indicates that adjustment of interval of the original uniform codeword pick method can improve the performance. This result is summarized in Chapter 5, and has been accepted by *the 20th Personal, Indoor and Mobile Radio Communications Symposium* [37].

In Chapter 6, we extend our code design approach to an MIMO system, and propose a systematic space-time code construction for joint channel estimation and error correction subject to two transmit antennas and 1/2 code rate. Similar to Chapter 3, its *maximum-likelihood* decoder that follows a priority-first search principle is also established. Our systematic code construction, together with a fairly low complexity optimal decoder, then allows one to work with longer codes with no sacrifice in performance. The results of this chapter has already been published in *2009 IEEE International Symposium on Information Theory* [38].

In the end, we conclude our results, and propose some future work in Chapter 7.

1.3 Acronyms

The acronyms used in this dissertation are listed in the following.

AWGN additive white Gaussian noise

BER bit error rate

CQI channel quality indicator

CSI channel state information

GLRT generalized likelihood-ratio test

GSM global system for mobile communication

MIMO multiple-input multiple-output

ML maximum-likelihood

OFDM orthogonal frequency division multiplexing

PEP pairwise error probabilities

PFSD priority-first search decoding

PSK phase-shift keying

SNR signal-to-noise ratio

TDD time-division duplex

UMTS universal mobile telecommunication system

WER word error rate

1.4 Notations

Throughout the dissertation, the following notations are used.

Symbol	Meaning
$\boldsymbol{v}$	a vector *(The following 13 notations are simple representative examples. Similar notations applies to other alphabets.)*
v_k	the k-th component of a vector
$\\|\boldsymbol{v}\\|^2$	the norm of a vector
$\mathbb{X}$	a matrix
$x_{k,\ell}$	the element of a matrix in row k and column ℓ
$\mathbb{X}^T$	the matrix operations of Hermitian transpose
$\mathbb{X}^H$	the matrix operations of transpose
$\mathrm{tr}(\mathbb{X})$	the trace of a square matrix
$\det\|\mathbb{X}\|$	the determinant of a matrix
$\mathbb{X} \oplus \mathbb{Y}$	the direct sum of $\mathbb{X}$ and $\mathbb{Y}$, i.e., $\mathbb{X} \oplus \mathbb{Y} = \begin{bmatrix} \mathbb{X} & \mathbf{0} \\ \mathbf{0} & \mathbb{Y} \end{bmatrix}$
$\mathbb{X} \otimes \mathbb{Y}$	the Kronecker product of matrices $\mathbb{X}$ and $\mathbb{Y}$
$\mathbb{X} \odot \mathbb{Y}$	the Hadamard product of matrices $\mathbb{X}$ and $\mathbb{Y}$
$\mathrm{vec}(\mathbb{X})$	the operation to transform a matrix into a vector. For an $M \times N$ matrix $\mathbb{X}$, $\mathrm{vec}(\mathbb{X})$ is defined as $\mathrm{vec}(\mathbb{X}) = [x_{1,1} \ \ldots \ x_{M,1} \ x_{1,2} \ \ldots \ x_{M,2} \ x_{1,N} \ \ldots \ x_{M,N}]^T$.
$\mathbb{I}_L$	the $L \times L$ identity matrix
$\binom{m}{n}$	binomial coefficient, $\binom{m}{n} \triangleq \frac{m!}{(m-n)!n!}$
$\|c\|$	the absolute value of a number c
$\|\mathcal{S}\|$	the cardinality of a set $\mathcal{S}$, i.e., the number of members in $\mathcal{S}$

Below are some common identifiers, if not state otherwise.

Symbol	Meaning
A_T	number of transmitter antennas
A_R	number of receiver antennas
N	block length of a code, i.e., number of symbols in a codeword
K	number of bits in an information sequence to be encoded
P	number of channel paths or taps
Q	number of symbol period within a coherent time of block fading channels
L	$= N + P - 1$ is the sum of a code word length plus one minimum guard period that can prevent the inter-symbol interference.

Chapter 2

Technical Background

For a better understanding, some background knowledge about our coding scheme is provided in this chapter. In short, joint maximum-likelihood (JML) sequence and channel estimation is introduced in Section 2.1. What follows is Section 2.2 that gives a short description of the code tree for the (N, K) code $\mathcal{C}$ over which the decoding search in Chapter 3 is performed.

2.1 Joint Maximum-Likelihood Sequence and Channel Estimation

In this section, the basic knowledge about joint maximum-likelihood (JML) sequence and channel estimation, or so-called generalized likelihood ratio test (GLRT), is presented. More detail can be found in [3, 11, 13].

The single-input-single-output (SISO) time-discrete system model we consider in this dissertation can be described as

$$\boldsymbol{y} = \mathbb{B}\boldsymbol{h} + \boldsymbol{n}$$

where

$$\mathbb{B} \triangleq \begin{bmatrix} b_1 & 0 & \cdots & 0 \\ \vdots & b_1 & \ddots & \vdots \\ b_N & \vdots & \ddots & 0 \\ 0 & b_N & \ddots & b_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & b_N \end{bmatrix}_{L\times P},$$

is a $L \times P$ signaling matrix that emulates the convolution operation with channel coefficient $\boldsymbol{h}$, and every non-zero entity of $\mathbb{B}$ is a bipolar symbol. Here, $\boldsymbol{h}$ is the $P \times 1$ channel coefficient vector and $\boldsymbol{n}$ is a $L \times 1$ noise vector.

Under the condition that transmitted signal $\mathbb{B}$ is equally likely, the maximum-likelihood (ML) detector is

$$\hat{\mathbb{B}} = \arg\max_{\mathbb{B}} \Pr(\boldsymbol{y}|\mathbb{B})$$

Obviously, the ML decision requires the knowledge of probability of $\boldsymbol{y}$ given $\mathbb{B}$.

For the case that $\boldsymbol{n}$ is zero-mean Gaussian distributed with covariance matrix $\sigma^2\mathbb{I}_L$, and that the receiver has perfect knowledge about $\boldsymbol{h}$, we know that

$$\Pr(\boldsymbol{y}|\mathbb{B}, \boldsymbol{h}) = \frac{1}{\det|\pi\sigma^2\mathbb{I}_L|} e^{-\|\boldsymbol{y}-\mathbb{B}\boldsymbol{h}\|^2/\sigma^2}$$

Yet, since $\boldsymbol{h}$ is completely unknown to the receiver (and also the transmitter), the optimal decision maker at the receiver end can only rely on the joint maximum-likelihood (JML) detection:

$$(\hat{\mathbb{B}}, \hat{\boldsymbol{h}}) = \arg\max_{\mathbb{B}} \max_{\boldsymbol{h}} \Pr(\boldsymbol{y}|\mathbb{B}, \boldsymbol{h}).$$

For a given transmitted signal $\mathbb{B}$, channel coefficient $\boldsymbol{h}$ that maximizes $\Pr(\boldsymbol{y}|\mathbb{B}, \boldsymbol{h})$ can be obtained as

$$\hat{\boldsymbol{h}} = \arg\max_{\boldsymbol{h}} \Pr(\boldsymbol{y}|\mathbb{B}, \boldsymbol{h}) = (\mathbb{B}^H\mathbb{B})^{-1}\mathbb{B}^H\boldsymbol{y},$$

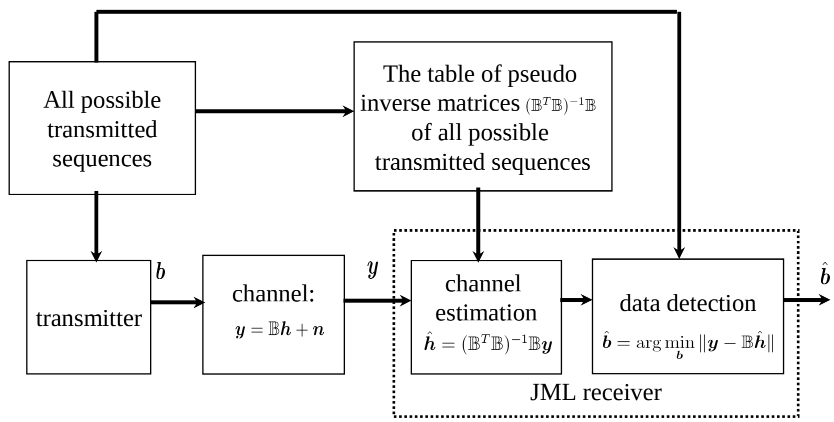

Figure 2.1: An illustration of a JML receiver.

Then, we can use the maximizer to detect the transmitted code word via

$$\begin{aligned}
\hat{\boldsymbol{b}} &= \arg\max_{\mathbb{B}} \Pr(\boldsymbol{y}|\mathbb{B}, \hat{\boldsymbol{h}}) \\
&= \arg\max_{\mathbb{B}} \frac{1}{\det|\pi\sigma^2\mathbb{I}|} e^{-\|\boldsymbol{y}-\mathbb{B}\hat{\boldsymbol{h}}\|^2/\sigma^2} \\
&= \arg\min_{\mathbb{B}} \|\boldsymbol{y} - \mathbb{B}(\mathbb{B}^H\mathbb{B})^{-1}\mathbb{B}^H\boldsymbol{y}\|^2 \\
&= \arg\min_{\mathbb{B}} \|\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y}\|^2, \qquad (2.1)
\end{aligned}$$

where $\mathbb{P}_B \triangleq \mathbb{B}(\mathbb{B}^H\mathbb{B})^{-1}\mathbb{B}^H$.

It is obvious from the above equation that we do not need to calculate $\hat{\boldsymbol{h}}$ corresponding to each code word $\boldsymbol{b}$ when making the final decision. What is required in the decision rule is actually the pre-preparation of $\mathbb{P}_B$ table corresponding to each code word $\boldsymbol{b}$ (or equivalently, $\mathbb{B}$). Nevertheless, an JML decision in (2.1) can be regarded as performing an implicit channel estimate $\hat{\boldsymbol{h}}$ upon the reception of $\boldsymbol{y}$ as shown in Fig. 2.1.

It is worth mentioning that the performance of the ML detector with perfectly known $\boldsymbol{h}$ and the JML detector with no knowledge on $\boldsymbol{h}$ have been analyzed in [3,11]. Specifically, the authors in [3] derived approximate formulas of PEP for both ML detection with known $\boldsymbol{h}$ and JML detection with unknown $\boldsymbol{h}$ at high SNR. Their results are cited here for reference. The approximate PEP of the ML detection with known $\boldsymbol{h}$ is

$$\Pr\left(\hat{\mathbb{B}} = \mathbb{B}(j) \middle| \mathbb{B}(i) \text{ transmitted}\right) \approx \frac{K_{i,j}}{\det|\mathbb{C}_h| \, \det|\mathbb{B}(i)^H(\mathbb{I} - \mathbb{P}_{B(j)})\mathbb{B}(i)|}$$

where

$$K_{i,j} = \begin{cases} \dfrac{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}\displaystyle\sum_{k=0}^{P-1}\binom{2P-1-k}{P-1}\frac{1}{k!}\left(\ln\frac{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}\right)^k, & \text{if } \dfrac{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}\leq 1 \\ \displaystyle\sum_{k=0}^{P}\binom{2P-1-k}{P-1}\frac{1}{k!}\left(\ln\frac{\det|\mathbb{B}(i)^H\mathbb{B}(i)|}{\det|\mathbb{B}(j)^H\mathbb{B}(j)|}\right)^k, & \text{otherwise.} \end{cases}$$

Here, the authors assume $0^0 = 1$ for convenience. For JML detection with unknown $\boldsymbol{h}$, its approximate PEP is

$$\Pr\left(\hat{\mathbb{B}} = \mathbb{B}(j)\middle|\mathbb{B}(i) \text{ transmitted}\right) \approx \frac{\binom{2P-1}{P-1}}{\det|\mathbb{C}_h|\ \det|\mathbb{B}(i)^H(\mathbb{I}-\mathbb{P}_{B(j)})\mathbb{B}(i)|}$$

It can be seen that when $\det|\mathbb{B}(i)^H\mathbb{B}(i)| = \det|\mathbb{B}(j)^H\mathbb{B}(j)|$, the approximate PEP of ML-with-known-$\boldsymbol{h}$ and JML detections are the same. Accordingly, both [3] and [11] concluded that the performances of the JML detection is asymptotically equivalent to the ML detection with known $\boldsymbol{h}$.

We close this section by emphasizing that when $P = 1$, i.e., the channel is reduced to a flat fading channel, the JML is simplified to

$$\hat{\boldsymbol{b}} = \arg\max_{\boldsymbol{b}} \|\boldsymbol{y}^H\boldsymbol{b}\|^2. \tag{2.2}$$

This criterion will be investigated further in Chapter 5.

2.2 The Maximum-Likelihood Priority-First Search Decoding Algorithm

A code tree of an (N, K) binary code represents every codeword as a path on a binary tree as shown in Fig. 2.2. The code tree consists of $(N + 1)$ levels. The single leftmost node at level zero is usually called the *origin node*. There are at most two branches leaving each node at each level. The 2^K rightmost nodes at level N are called the *terminal nodes*.

Each branch on the code tree is labeled with the appropriate code bit b_i. As a convention, the path from the single origin node to one of the 2^K terminal nodes is termed the *code path* corresponding to the codeword. Since there is a one-to-one correspondence between the codeword and the code path of $\mathcal{C}$, a codeword can be interchangeably referred to by its respective code path or the branch labels that the code path traverses. Similarly, for any node in the code tree, there exists a unique path traversing from the single original node to it; hence, a node can also be interchangeably indicated by the path (or the path labels) ending at it. We can then denote the path ending at a node at level ℓ by the sequence of branch labels $[b_1, b_2, \ldots, b_\ell]$ it traverses. For convenience, we abbreviate $[b_1, b_2, \ldots, b_\ell]^T$ as $\boldsymbol{b}_{(\ell)}$, and will drop the subscript when $\ell = N$. The successor paths of a path $\boldsymbol{b}_{(\ell)}$ are those whose first ℓ labels are exactly the same as $\boldsymbol{b}_{(\ell)}$.

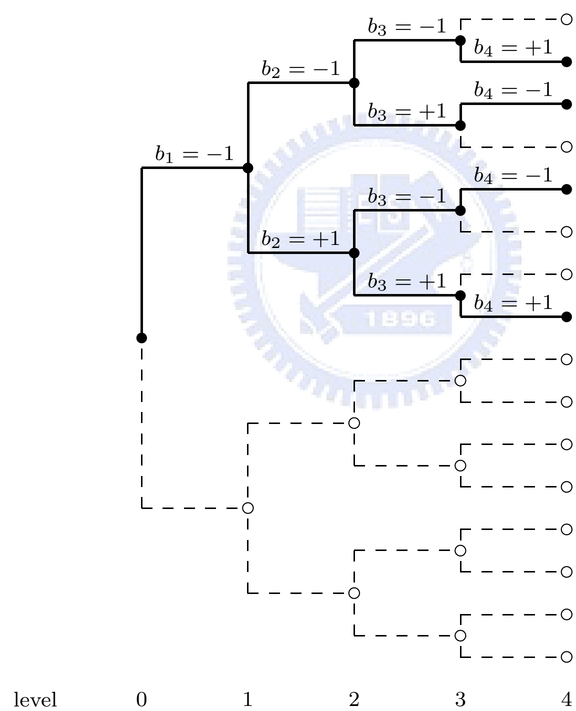

Figure 2.2: The code tree for a computer-searched PEP-minimum $(4, 2)$ code with b_1 fixed as -1.

The *priority-first search algorithm* (also known as the *best-first search algorithm*) is a common graph search algorithm that explores a graph by expanding the most promising path that is selected according to some criterion. Examples are *Algorithm A** [27], *Dijkstra's Algorithm* [6], or *Stack Algorithm* [20]. In implementation, the most promising path is usually drawn from a list of candidates in a stack or a priority queue. One of the main distinctions among the family of priority-first search algorithms is the *metric* associated with paths on the search graph.[1] By adopting different metrics, some algorithms guarantee optimal search results, while some can only yield suboptimal ones. A typical priority-first search algorithm is exemplified by the following sequence of operations:

Step 1. Load the stack with the path that ends at the original node.

Step 2. Evaluate the metric values of the successor paths of the current top path in the stack. Then delete this top path from the stack.

Step 3. Insert the successor paths obtained in Step 2 into the stack such that the paths in the stack are ordered according to their ascending metric values.

Step 4. If the top path in the stack ends at a terminal node in the code tree, output the labels corresponding to the top path, and the algorithm stops; otherwise, go to Step 2.

Next, we give a sufficient condition under which the above priority-first search algorithm is guaranteed to locate the path with the smallest metric among all paths.

Lemma 1. *If the metric f is nondecreasing along every path $\boldsymbol{b}_{(\ell)}$ in the code tree, i.e.,*

$$f\left(\boldsymbol{b}_{(\ell)}\right) \leq \min_{\left\{\tilde{\boldsymbol{b}}\in\mathcal{C}:\, \tilde{\boldsymbol{b}}_{(\ell)}=\boldsymbol{b}_{(\ell)}\right\}} f(\tilde{\boldsymbol{b}}), \tag{2.3}$$

then the priority-first search algorithm always yields the code path with the smallest metric value among all code paths of $\mathcal{C}$.

Proof. Let $\boldsymbol{b}^*$ be the first top path that reaches a terminal node (and hence, is the output code path of the priority-first search algorithm.) Then, *Step 3* of the algorithm ensures that

[1] In the optimization literature, this metric is sometimes called *evaluation function*. Since we apply the algorithm in decoding, we adopt the term *metric* in this work.

$f(\boldsymbol{b}^*)$ is no larger than the metric value of any path currently in the stack. Since condition (2.3) guarantees that the metric value of any other code path, which should be the offspring of some path $\boldsymbol{b}_{(\ell)}$ currently existing in the stack, is no less than $f(\boldsymbol{b}_{(\ell)})$, we have

$$f(\boldsymbol{b}^*) \le f(\boldsymbol{b}_{(\ell)}) \le \min_{\{\tilde{\boldsymbol{b}} \in \mathcal{C}:\ \tilde{\boldsymbol{b}}_{(\ell)} = \boldsymbol{b}_{(\ell)}\}} f(\tilde{\boldsymbol{b}}).$$

Consequently, the lemma follows. $\square$

When defining a metric f, it is convenient to represent it as the sum of two components:

$$f(\boldsymbol{b}_{(\ell)}) \triangleq g(\boldsymbol{b}_{(\ell)}) + \varphi(\boldsymbol{b}_{(\ell)}).$$

The first component g is directly defined based on the maximum-likelihood metric such that

$$\arg\min_{\boldsymbol{b} \in \mathcal{C}} g(\boldsymbol{b}) = \arg\min_{\boldsymbol{b} \in \mathcal{C}} \|\boldsymbol{y} - \mathbb{P}_B \boldsymbol{y}\|^2.$$

After g is defined, the second component φ is designed to validate (2.3) with $\varphi(\boldsymbol{b}) = 0$ for any $\boldsymbol{b} \in \mathcal{C}$. Then from $f(\boldsymbol{b}) = g(\boldsymbol{b}) + \varphi(\boldsymbol{b}) = g(\boldsymbol{b})$ for all $\boldsymbol{b} \in \mathcal{C}$, the desired maximum-likelihood priority-first search decoding algorithm is established. A typical interpretation of the so-called *heuristic function* φ is that it helps predict a future route from the end node of the current path to a terminal node [17]. Notably, the design of the heuristic function φ that validates condition (2.3) is not unique. Different designs may result in variations in computational complexity.

Chapter 3

Code Designs for Frequency-Selective Block Fading Channels

In the literature, no systematic code constructions have been proposed for joint channel and data estimation in quasi-static fading channels. Efforts have mostly been invested in computer searches for codes that counter channel fading [7,12,25,26,30,32,40]. The decoding of such structureless computer-searched codes thus becomes an engineering challenge.

In 2003, Skoglund *et al.* [30] relied on simulated annealing to search for nonlinear binary block codes suitable for joint channel and data estimation in quasi-static fading channels. As optimization criterion, they used the sum of all pairwise error probabilities (PEP) under equal prior probabilities. Although the operating signal-to-noise ratio (SNR) for the code search was set at 10 dB, their simulation results demonstrated that their codes perform well under a wide range of SNRs. In addition, the mismatch in the relative powers of different channel coefficients, as well as in the channel Rice factors [33], has no big effect on the resulting performance either. Their results indicate that the nonlinear estimation codes can outperform a typical linear error correcting code operated with a perfect channel estimator.

Later in 2005, Coskun and Chugg [7] replaced the PEP sum by a properly defined pairwise distance measure between two codewords, and proposed a suboptimal greedy algorithm to speed up the code search process. In 2007, Giese and Skoglund [13] re-applied their original idea to single and multiple-antenna systems, and used the asymptotic PEP and the generic gradient-search algorithm, respectively, in place of the PEP and the simulated-annealing algorithm in [30] to reduce system complexity.

In [30], the authors point out that “an important topic for further research is to study how the decoding complexity of the proposed scheme can be decreased.” Moreover, they state that “one main issue is to investigate what kind of structure should be enforced on the code to allow for simplified decoding.” Motivated by these remarks, we take here a different approach for code design. Specifically, we establish a systematic code design constraint for joint channel and data estimation in quasi-static fading channels, and show that the codes constructed based on this constraint can maximize the system SNR regardless of the fading statistics. As it so happens that the computer-searched codes in [30] also satisfy this constraint, their insensitivity to SNR and channel mismatch now find a theoretical support.

Although a recursive metric had been derived in [4] from joint maximum-likelihood decoding metric, however, there is no efficient decoding algorithm that can exploit it due to structureless code design. Taking advantage of the systematic structure of our codes, we can then derive a recursive maximum-likelihood decoding metric that can be used in the priority-first search decoding algorithm. The decoding complexity is therefore significantly decreased in contrast to that of the exhaustive decoder required by the structureless computer-searched codes.

This chapter is organized as follows. Section 3.1 describes the system model. Section 3.2 establishes the self-orthogonal codeword-selection condition that optimizes the system SNR regardless of the fading statistics, and then uses it to construct codes for joint channel and data estimation. The recursive maximum-likelihood decoding metrics for the constructed codes are derived in Section 3.3. Simulations are summarized and discussed in Section 3.4. Section 3.5 summarizes the chapter.

3.1 System Model and Maximum-Likelihood Decoding Criterion

Suppose a codeword $\boldsymbol{b} = [b_1, \ldots, b_N]^T$ of an (N, K) code $\mathcal{C}$ is transmitted over a block fading (specifically, quasi-static fading) channel of memory order $P-1$, where each $b_j \in \{-1, +1\}$, and $N \geq P$. Denote the channel coefficients by $\boldsymbol{h} = [h_1, \ldots, h_P]^T$, and assume that they are *constant* within a coding block of length $L = N + P - 1$. By letting the codeword matrix be

$$\mathbb{B} \triangleq \begin{bmatrix} b_1 & 0 & \cdots & 0 \\ \vdots & b_1 & \ddots & \vdots \\ b_N & \vdots & \ddots & 0 \\ 0 & b_N & \ddots & b_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & b_N \end{bmatrix}_{L \times P},$$

the complex-valued received vector $\boldsymbol{y}$ is given by

$$\boldsymbol{y} = \mathbb{B}\boldsymbol{h} + \boldsymbol{n}, \tag{3.1}$$

where $\boldsymbol{n}$ is zero-mean complex-Gaussian distributed with $E[\boldsymbol{n}\boldsymbol{n}^H] = \sigma_n^2 \mathbb{I}_L$, and $\mathbb{I}_L$ is the $L \times L$ identity matrix. We then make the following assumptions: both transmitter and receiver know nothing about the channel coefficients $\boldsymbol{h}$, but have knowledge of the multi-path parameter P. Also, there are adequate guard periods between consecutive encoding blocks such that zero interblock interference is guaranteed. Based on the system model in (3.1) and the above two assumptions, the least square estimate of the channel coefficients $\boldsymbol{h}$ for a given $\boldsymbol{b}$ (alternatively, $\mathbb{B}$) equals $\hat{\boldsymbol{h}} = (\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T\boldsymbol{y}$, and the joint maximum-likelihood (ML) decision for the transmitted codeword becomes [4]

$$\hat{\boldsymbol{b}} = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \min_{\boldsymbol{h}} \|\boldsymbol{y} - \mathbb{B}\boldsymbol{h}\|^2 = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\boldsymbol{y} - \mathbb{B}\hat{\boldsymbol{h}}\|^2 = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y}\|^2, \tag{3.2}$$

where $\mathbb{P}_B \triangleq \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T$. Note that the mapping from a codeword $\boldsymbol{b}$ to the corresponding transformed codeword $\mathbb{P}_B$ is not one-to-one unless b_1 is fixed. For convenience, we will always set $b_1 = -1$ for the codes we construct.[1]

[1] Under the setting, it is obvious that the largest code rate attainable by our code design is $(N-1)/N$.

3.2 Code Construction

3.2.1 Code Constraint that Maximizes the Average SNR Regardless of Channel Statistics

From the system model in (3.1), it can be derived that the average SNR conditional on the input $\mathbb{B}$ satisfies

$$\frac{E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]} = \frac{N}{L\sigma_n^2}\text{tr}\left[E[\boldsymbol{h}\boldsymbol{h}^H]\left(\frac{1}{N}\mathbb{B}^T\mathbb{B}\right)\right]. \tag{3.3}$$

Since both transmitter and receiver know nothing about the channel coefficients $\boldsymbol{h}$, the average SNR can be as worse as

$$\min_{\{\boldsymbol{h}\,:\,\text{tr}(E[\boldsymbol{h}\boldsymbol{h}^H])=\tau\}} \frac{E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]},$$

where τ is a certain (possibly unknown) power level on the channel coefficients $\boldsymbol{h}$. We then found that such a worst-case SNR can be upper-bounded by a constant, i.e.,

$$\min_{\{\boldsymbol{h}\,:\,\text{tr}(E[\boldsymbol{h}\boldsymbol{h}^H])=\tau\}} \frac{E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]} \leq \frac{E[\|\mathbb{B}\tilde{\boldsymbol{h}}\|^2|\mathbb{B}]}{E[\|\boldsymbol{n}\|^2]} = \left(\frac{N}{L\sigma_n^2}\right)\tau,$$

where the above inequality holds since an upper bound can be resulted by taking any $\boldsymbol{h}$ that satisfies $\text{tr}(E[\boldsymbol{h}\boldsymbol{h}^H]) = \tau$ into $E[\|\mathbb{B}\boldsymbol{h}\|^2|\mathbb{B}]/E[\|\boldsymbol{n}\|^2]$, and here we take $\tilde{\boldsymbol{h}}$ to be zero-mean i.i.d. with $\text{tr}(E[\tilde{\boldsymbol{h}}\tilde{\boldsymbol{h}}^H]) = \tau$. It is thus straightforward from (3.3) that this constant SNR bound can be achieved even if the system is totally blind on channel coefficients $\boldsymbol{h}$ (as well as the power level τ), when the codeword is designed to be *self-orthogonal* in the sense that

$$\frac{1}{N}\mathbb{B}^T\mathbb{B} = \mathbb{I}_P. \tag{3.4}$$

Condition (3.4) actually has an operational meaning. It ensures that every codeword is orthogonal to the shifted version of itself, and hence temporal diversity can be implicitly realized even under completely no knowledge on channel statistics. We henceforth say that codewords constrained on (3.4) maximize the average SNR attainable regardless of the statistics of $\boldsymbol{h}$ [14].

Unfortunately, a codeword sequence satisfying (3.4) is only guaranteed to exist for $P = 2$ with N odd (and trivially, for $P = 1$). In some other cases, such as $P = 3$, one can only design codes to approximately satisfy (3.4). For example,

$$\frac{1}{N}\mathbb{B}^T\mathbb{B} = \frac{1}{N}\begin{bmatrix} N & \pm 1 & 0 \\ \pm 1 & N & \pm 1 \\ 0 & \pm 1 & N \end{bmatrix} \text{ for } N \text{ even,}$$

and

$$\frac{1}{N}\mathbb{B}^T\mathbb{B} = \frac{1}{N}\begin{bmatrix} N & 0 & \pm 1 \\ 0 & N & 0 \\ \pm 1 & 0 & N \end{bmatrix} \text{ for } N \text{ odd.}$$

We therefore relax (3.4) and allow some off-diagonal entries in $\mathbb{B}^T\mathbb{B}$ to be either -1 or 1 whenever it is impossible to strictly satisfy (3.4). We will denote such a matrix as $\mathbb{G}$.

After the establishment of (3.4), we find that this particular structure of $\mathbb{G}$ can really be observed in the simulated-annealing-based computer-searched codes. Specifically, for $4 \le N \le 18$ and N even, the best computer-searched half-rate codes that minimize the sum of PEPs under complex zero-mean Gaussian distributed $\boldsymbol{h}$ with $E[\boldsymbol{h}\boldsymbol{h}^H] = (1/2)\mathbb{I}_P$ and $P = 2$ all satisfy the relation

$$\mathbb{B}^T\mathbb{B} = \begin{bmatrix} N & \pm 1 \\ \pm 1 & N \end{bmatrix}. \tag{3.5}$$

We have also obtained and examined the computer-searched code used in [30] for $N = 22$, and found as anticipated that every codeword satisfies (3.5).

We close this subsection by stating some existing results in the literature that correspond to condition (3.4). The authors in [8] suggest that for an optimal channel estimation, the training sequences $\boldsymbol{b}$ can be chosen such that $\mathbb{B}^T\mathbb{B}$ is proportional to $\mathbb{I}_P$. Their observation agrees with what we obtained in (3.4). Moreover, condition (3.4) also has been identified in [13] where the authors remark [13, pp. 1591] that a code sequence with a certain aperiodic autocorrelation property possibly could be exploited in future code design approaches. This is indeed one of the main research goals of this chapter.

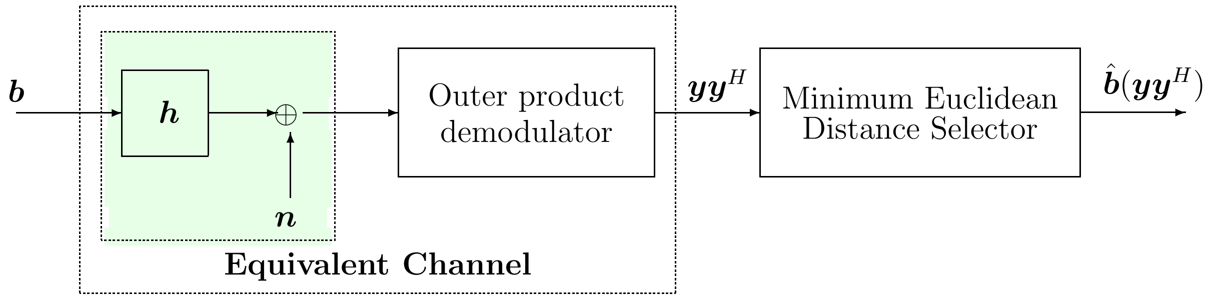

Figure 3.1: Equivalent system model for combined channel estimation and error protection codes.

3.2.2 Equivalent System Model for Joint Channel and Data Estimation

By noting that $\mathbb{P}_B$ is idempotent and symmetric, and both $\mathrm{tr}(\mathbb{P}_B)$ and $\|\mathrm{vec}(\mathbb{P}_B)\|^2$ equal P, where $\mathrm{vec}(\cdot)$ denotes the operation to transform a matrix into a vector, the joint ML decision in (3.2) can be reformulated as

$$\begin{aligned}\hat{\boldsymbol{b}} &= \arg\min_{\boldsymbol{b}\in\mathcal{C}} (\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y})^H(\boldsymbol{y} - \mathbb{P}_B\boldsymbol{y}) \\ &= \arg\min_{\boldsymbol{b}\in\mathcal{C}} -\mathrm{tr}(\mathbb{P}_B\boldsymbol{y}\boldsymbol{y}^H) \\ &= \arg\min_{\boldsymbol{b}\in\mathcal{C}} \left(\|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H)\|^2 - \mathrm{vec}(\mathbb{P}_B)^T\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H) - \mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H)^H\mathrm{vec}(\mathbb{P}_B) + \|\mathrm{vec}(\mathbb{P}_B)\|^2\right) \\ &= \arg\min_{\boldsymbol{b}\in\mathcal{C}} \|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H) - \mathrm{vec}(\mathbb{P}_B)\|^2.\end{aligned} \tag{3.6}$$

This implies that the ML decision can be obtained by finding the codeword $\mathbb{P}_B$ whose Euclidean distance to $\boldsymbol{y}\boldsymbol{y}^H$ is the smallest.

We therefore transform the original system in (3.1) to an equivalent system model that contains an *outer product demodulator* and a *minimum Euclidean distance selector* at the $\mathbb{P}_B$-domain as shown in Fig. 3.1. As the outer product demodulator can be viewed as a generalization of the *square-law combining* that is of popular use in non-coherent detection for both slow and fast fading [28], the above equivalent transformation suggests a potential application of combined channel estimate and error protection codes for the non-coherent system in which the fading is rapid enough to preclude a good estimate of the channel coefficients. Further discussion on how to design codes for unknown fast-fading channels will be continued in subsequent chapters.

We can then bound the ML error probability by

$$P_{\mathrm{e}} \leq \frac{1}{2^K}\sum_{i=1}^{2^K}\sum_{\substack{j=1\\j\neq i}}^{2^K} \Pr\left(\|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H)-\mathrm{vec}(\mathbb{P}_{B_j})\|^2 < \|\mathrm{vec}(\boldsymbol{y}\boldsymbol{y}^H)-\mathrm{vec}(\mathbb{P}_{B_i})\|^2 \,\middle|\, \boldsymbol{b}_i \text{ transmitted}\right), \tag{3.7}$$

where $\boldsymbol{b}_i$ is the ith codeword of an (N,K) block code, and $\mathbb{P}_{B_i}$ denotes the equivalent ith codeword in the $\mathbb{P}_B$-domain. By the self-orthogonal property, $\mathbb{P}_B = \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T = \frac{1}{N}\mathbb{B}\mathbb{B}^T$. The PEP-based upper bound in (3.7) then suggests that a good self-orthogonal code design should have an adequately large pairwise Euclidean distance

$$\left\|\mathrm{vec}(\mathbb{B}_i\mathbb{B}_i^T)-\mathrm{vec}(\mathbb{B}_j\mathbb{B}_j^T)\right\|^2 \tag{3.8}$$

between all codeword pairs $\mathbb{B}_i$ and $\mathbb{B}_j$, where $\mathbb{B}_i$ is the equivalent ith codeword in the $\mathbb{B}$-domain. Based on this observation, we may infer under equal prior probabilities that a uniform draw of codewords satisfying $\mathbb{B}^T\mathbb{B} = N\cdot\mathbb{I}_P$ may asymptotically result in a good code. This is conceptually equivalent to a uniform pick of codewords in a set of self-orthogonal binary sequences.

We recall that our initial research query is how to construct an efficiently decodable code that supports joint channel estimation and error correction. In order to achieve this goal for the priority-first search decoding algorithm, we need an efficient and systematic way to generate the successor paths of the top path. In particular, we would like to have a code tree that can be spanned in an on-the-fly or bit-by-bit fashion. The uniform pick principle then suggests that considering only the self-orthogonal sequences with the same prefix $\boldsymbol{b}_{(\ell-1)}$, the ratio of the number of self-orthogonal codewords satisfying $b_\ell = -1$ to the number of all self-orthogonal sequences having the same b_ℓ must be made equal to the similar ratio for self-orthogonal codewords satisfying $b_\ell = 1$, whenever possible. Mathematically, this can be expressed as

$$\frac{|\mathcal{C}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = 1)|}{|\mathcal{A}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = 1|\mathbb{G})|} \approx \frac{|\mathcal{C}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = -1)|}{|\mathcal{A}(b_1, b_2, \ldots, b_{\ell-1}, b_\ell = -1|\mathbb{G})|}, \tag{3.9}$$

where $\mathcal{C}(\boldsymbol{b}_{(\ell)})$ is the set of all codewords whose first ℓ bits equal $b_1, b_2, \ldots, b_\ell$, and $\mathcal{A}(\boldsymbol{b}_{(\ell)}|\mathbb{G})$ is the set of all binary sequences of length N, whose first ℓ bits equal $b_1, b_2, \ldots, b_\ell$, and whose $\mathbb{B}$-representation satisfies $\mathbb{B}^T\mathbb{B} = \mathbb{G}$. Accordingly, given the index i of the codeword, where $0 \le i \le 2^K - 1$, and given the previous $\ell - 1$ bits $b_1, b_2, \ldots, b_{\ell-1}$, whether the next code bit b_ℓ is -1 or $+1$ can be determined conceptually by checking whether i is less than or larger than $\sum_{\tilde{b}_1+\tilde{b}_2\cdot 2+\ldots+\tilde{b}_{\ell-1}\cdot 2^{\ell-2}<b_1+b_2\cdot 2+\ldots+b_{\ell-1}\cdot 2^{\ell-2}} |\mathcal{C}(\tilde{\boldsymbol{b}}_{(\ell-1)}|\mathbb{G})| + |\mathcal{C}(\boldsymbol{b}_{(\ell-1)}, b_\ell = -1|\mathbb{G})|$. A specific code design algorithm will be given in the next subsection.

3.2.3 Exemplified Code Design Algorithm for Channels of Memory Order One

In this subsection, we provide an exemplified code design algorithm based on the uniform pick principle for channels of memory order 1, namely, $P = 2$. The code design algorithm for channels with higher memory order can be similarly built.

For $\theta \in \{-1, 0, +1\}$, we define

$$\mathbb{G}_\theta \triangleq \begin{bmatrix} N & \theta \\ \theta & N \end{bmatrix}.$$

Note that when $\mathbb{B}^T\mathbb{B} = \mathbb{G}_0$ $(= N \cdot \mathbb{I}_2)$ cannot be satisfied as aforementioned for N even, $\mathbb{G}_{-1}$ and $\mathbb{G}_1$ will be used instead to define the relaxed self-orthogonal codewords. In such case, the uniform pick principle again suggests that half of the codewords should be uniformly drawn from binary sequences satisfying $\mathbb{B}^T\mathbb{B} = \mathbb{G}_{-1}$, and the other half of codewords are selected according to $\mathbb{B}^T\mathbb{B} = \mathbb{G}_1$. The proposed codeword selection process is simply to list all the sequences satisfying the desired self-orthogonal property in binary-alphabetical order, starting from zero, and uniformly pick the codewords from the ordered list in every Δ_θ interval with

$$\Delta_\theta = \frac{|\mathcal{A}(b_1 = -1|\mathbb{G}_\theta)| - 1}{2^K/|\Theta| - 1} \quad \text{for } \theta \in \Theta, \tag{3.10}$$

where $\Theta = \{0\}$ for N odd, and $\Theta = \{-1, 1\}$ for N even. As a result, the selected codewords are those sequences with indices closest to $\lfloor (i \bmod (2^K/|\Theta|)) \cdot \Delta_\theta \rfloor$ for $0 \le i \le 2^K - 1$. The codeword mapping algorithm is summarized by the following list:

Step 1. Input the index i of the requested codeword in the (N, K) block code, where $0 \le i \le 2^K - 1$.

Step 2. Set $\Theta = \{0\}$ for N odd, and $\Theta = \{-1, 1\}$ for N even. Also, set $\theta = ((N + 1) \bmod 2) \cdot (-1)^{\lceil (i+1)/(2^K/|\Theta|) \rceil}$. Compute Δ_θ according to (3.10). Initialize $b_1 = -1$, $\ell = 1$ and $\rho = \lfloor (i \bmod (2^K/|\Theta|)) \cdot \Delta_\theta \rfloor$. Let the minimum sequence index $\rho_{\min} = 0$.

Step 3. Execute $\ell = \ell + 1$, and compute $\gamma_\ell = |\mathcal{A}(\boldsymbol{b}_{(\ell-1)}, b_\ell = -1 | \mathbb{G}_\theta)|$.

If $\rho < \rho_{\min} + \gamma_\ell$, then choose the next code bit $b_\ell = -1$;

otherwise choose the next code bit $b_\ell = 1$, and readjust $\rho_{\min} = \rho_{\min} + \gamma_\ell$.

Step 4. If $\ell = N$, output the corresponding codeword $\boldsymbol{b}$, and the algorithm stops; otherwise, go to Step 3.

In implementing the above algorithm, it is perhaps more convenient to calculate γ_ℓ recursively[2] such that the codeword mapping can be performed in an on-the-fly or bit-by-bit systematic fashion with respect to the given codeword index i. This recursive nature also facilitates the priority-first decoding search at the receiver, since branches of the code tree will only be spanned when necessary.

[2] Initializing $b_0 = 0$, $m_0 = \theta$ and $\gamma_1 = |\mathcal{A}(b_1 | \mathbb{G}_\theta)|$, and setting $m_{\ell+1} = m_\ell - b_\ell b_{\ell+1}$ for $0 \le \ell < N$, we obtain for $P = 2$ that if $|m_{\ell-1} + b_{\ell-1}| \le N - \ell$,

$$\gamma_{\ell+1} = \gamma_\ell \cdot \frac{1}{2(N-\ell)} \cdot \begin{cases} \left(\dfrac{(N - \ell - m_{\ell-1})^2 - 1}{N - \ell + m_{\ell-1} + 1} \right) \cdot \mathbf{1}\{|m_{\ell-1} + 2| \le N - \ell - 1\}, & \text{for } (b_{\ell-1}, b_\ell) = (-1, 1), \\ (N - \ell + m_{\ell-1} + 1 - b_{\ell-1} b_\ell + b_\ell) \cdot \mathbf{1}\{|m_{\ell-1} - b_{\ell-1} b_\ell + b_\ell| \le N - \ell - 1\}, & \text{otherwise,} \end{cases}$$

where $\mathbf{1}\{\cdot\}$ is the set indicator function. If however $|m_{\ell-1} + b_{\ell-1}| > N - \ell$, then

$$\gamma_{\ell+1} = \begin{cases} 0, & \text{for } (b_{\ell-1}, b_\ell) \ne (-1, 1) \text{ or } \big((b_{\ell-1}, b_\ell) = (-1, 1) \text{ and } m_{\ell-1} \ne -N + \ell - 1\big), \\ 1, & \text{otherwise.} \end{cases}$$

3.3 Maximum-Likelihood Metrics For Priority-First Search Decoding

In this section, we will establish two different metric functions to be used by the priority-first search algorithm. The first metric is

$$f_1(\boldsymbol{b}_{(\ell)}) = g(\boldsymbol{b}_{(\ell)}) + \varphi_1(\boldsymbol{b}_{(\ell)}), \tag{3.11}$$

where $g(\boldsymbol{b}_{(\ell)})$ is derived in Section 3.3.1, and $\varphi_1(\boldsymbol{b}_{(\ell)}) = 0$ is the all-zero function (cf. Section 3.3.2). The second metric is

$$f_2(\boldsymbol{b}_{(\ell)}) = g(\boldsymbol{b}_{(\ell)}) + \varphi_2(\boldsymbol{b}_{(\ell)}), \tag{3.12}$$

with $g(\boldsymbol{y})$ the same as in f_1, and with $\varphi_2(\boldsymbol{b}_{(\ell)})$ defined in Section 3.3.3. Both metrics will lead to an ML decoding. The difference is that f_1 can be computed on-the-fly, and will therefore cause much less delay in the decoding. For the evaluation of f_2, however, one needs to know *all* received symbols, but the computational complexity of f_2 is one order of magnitude less than that of f_1.

3.3.1 Recursive Maximum-Likelihood Metric g

Let subcode $\mathcal{C}_\theta$ be the set of codewords that satisfy $\mathbb{B}^T\mathbb{B} = \mathbb{G}_\theta$, where θ takes value in Θ. Hence, $\mathcal{C} = \cup_{\theta\in\Theta}\mathcal{C}_\theta$, and $\mathcal{C}_\theta \cap \mathcal{C}_\eta = \emptyset$ whenever $\theta \neq \eta$. Since a transmitted codeword belongs to only one of the subcodes, to maintain *individual stacks* for priority-first codeword searching over *each* subcode will introduce considerable unnecessary decoding burden, especially for the subcodes that the transmitted codeword does not belong to. Hence, only one stack is maintained during the entire priority-first search, and the metric function values for different subcodes are compared and sorted in the same stack. The path to be expanded next is therefore the one whose metric function value is the smallest globally.

By denoting $\mathbb{D}_\theta = \mathbb{G}_\theta^{-1} = (\mathbb{B}^T\mathbb{B})^{-1}$, and letting the matrix entry of $\mathbb{D}_\theta$ be $\delta_{i,j}^{(\theta)}$, we can continue the derivation from (3.6) as follows:

$$\begin{aligned}\hat{\boldsymbol{b}} &= \arg\min_{\boldsymbol{b}\in\mathcal{C}}\sum_{\theta\in\Theta}\left[-\mathrm{tr}(\mathbb{B}\mathbb{D}_\theta\mathbb{B}^T\boldsymbol{y}\boldsymbol{y}^H)\right]\cdot\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\}\\ &= \arg\min_{\boldsymbol{b}\in\mathcal{C}}\sum_{\theta\in\Theta}\left[-\mathrm{vec}(\mathbb{D}_\theta)^T\mathrm{vec}(\mathbb{B}^T\boldsymbol{y}\boldsymbol{y}^H\mathbb{B})\right]\cdot\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\}\\ &= \arg\min_{\boldsymbol{b}\in\mathcal{C}}\sum_{\theta\in\Theta}\left[-\sum_{i=0}^{P-1}\sum_{j=0}^{P-1}\delta_{i,j}^{(\theta)}\sum_{m=1}^{L}\sum_{n=1}^{L}b_{m+i}b_{n+j}y_m y_n^*\right]\cdot\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\},\end{aligned}$$

where for convenience, we put $b_j = 0$ for $j > N$. After adjusting indices, the derivation can be resumed as

$$\hat{\boldsymbol{b}} = \arg\min_{\boldsymbol{b}\in\mathcal{C}}\frac{1}{2}\sum_{\theta\in\Theta}\left[\sum_{m=1}^{N}\sum_{n=1}^{N}\left(-w_{m,n}^{(\theta)}b_m b_n\right)\right]\cdot\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\}, \tag{3.13}$$

where

$$w_{m,n}^{(\theta)} = \sum_{i=0}^{P-1}\sum_{j=0}^{P-1}\delta_{i,j}^{(\theta)}\mathrm{Re}\{y_{m+i}y_{n+j}^*\}.$$

As the maximum-likelihood decision remains unchanged by adding a constant that is independent of the codeword $\boldsymbol{b}$, we add a constant to make the decision criterion nonnegative:[3]

$$\begin{aligned}\hat{\boldsymbol{b}} &= \arg\min_{\boldsymbol{b}\in\mathcal{C}}\left\{\sum_{m=1}^{N}\max_{\eta\in\Theta}\left(\sum_{n=1}^{m-1}|w_{m,n}^{(\eta)}|+\frac{1}{2}|w_{m,m}^{(\eta)}|\right)-\frac{1}{2}\sum_{\theta\in\Theta}\left[\sum_{m=1}^{N}\sum_{n=1}^{N}w_{m,n}^{(\theta)}b_m b_n\right]\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\}\right\}\\ &= \arg\min_{\boldsymbol{b}\in\mathcal{C}}\sum_{\theta\in\Theta}\left[\sum_{m=1}^{N}\max_{\eta\in\Theta}\left(\sum_{n=1}^{m-1}|w_{m,n}^{(\eta)}|+\frac{1}{2}|w_{m,m}^{(\eta)}|\right)-\frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N}w_{m,n}^{(\theta)}b_m b_n\right]\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\}.\end{aligned}$$

It remains to prove that the metric of

$$\sum_{m=1}^{N}\max_{\eta\in\Theta}\left(\sum_{n=1}^{m-1}|w_{m,n}^{(\eta)}|+\frac{1}{2}|w_{m,m}^{(\eta)}|\right)-\frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N}w_{m,n}^{(\theta)}b_m b_n$$

can be computed recursively. To that aim, we define for every path $\boldsymbol{b}_{(\ell)}$ over code tree θ

$$g(\boldsymbol{b}_{(\ell)}) \triangleq \sum_{m=1}^{\ell}\max_{\eta\in\Theta}\left(\sum_{n=1}^{m-1}|w_{m,n}^{(\eta)}|+\frac{1}{2}|w_{m,m}^{(\eta)}|\right)-\frac{1}{2}\sum_{m=1}^{\ell}\sum_{n=1}^{\ell}w_{m,n}^{(\theta)}b_m b_n.$$

[3]Here, a *nonnegative* maximum-likelihood criterion makes possible the later definition of path metric $g(\boldsymbol{b}_{(\ell)})$ to be *nondecreasing* along any path in the code tree. It can then be anticipated (cf. Section 3.3.2) that letting the heuristic function be zero for all paths in the code tree suffices to result in a metric function satisfying the condition (2.3) in Lemma 1.

Note that the additive constant that makes the metric function nondecreasing along any path in the code tree can also be obtained by first defining g based on (3.13), and then determining its respective φ according to (2.3). Such an approach however complicates the determination of the heuristic function φ when we additionally require the metric function to be recursive-computable. The alternative approach that directly defines a recursive-computable g based on a nonnegative maximum-likelihood criterion is accordingly adopted in this work.

Then, by $w_{m,n}^{(\theta)} = w_{n,m}^{(\theta)}$ for every $1 \le m, n \le N$ and $\theta \in \Theta$, we have for $1 \le \ell \le N-1$,

$$\begin{aligned} g(\boldsymbol{b}_{(\ell+1)}) &= g(\boldsymbol{b}_{(\ell)}) + \max_{\eta\in\Theta}\left(\sum_{n=1}^{\ell}|w_{\ell+1,n}^{(\eta)}| + \frac{1}{2}|w_{\ell+1,\ell+1}^{(\eta)}|\right) - \sum_{n=1}^{\ell} w_{\ell+1,n}^{(\theta)} b_{\ell+1} b_n - \frac{1}{2} w_{\ell+1,\ell+1}^{(\theta)} \\ &= g(\boldsymbol{b}_{(\ell)}) + \max_{\eta\in\Theta}\alpha_{\ell+1}^{(\eta)} - b_{\ell+1}\sum_{i=0}^{P-1}\sum_{j=0}^{P-1}\delta_{i,j}^{(\theta)}\mathrm{Re}\left\{y_{\ell+i+1}\cdot u_j(\boldsymbol{b}_{(\ell+1)})\right\}, \end{aligned}$$

where

$$\alpha_{\ell+1}^{(\eta)} \triangleq \sum_{n=1}^{\ell}|w_{\ell+1,n}^{(\eta)}| + \frac{1}{2}|w_{\ell+1,\ell+1}^{(\eta)}| \tag{3.14}$$

and for $0 \le j \le P-1$,

$$u_j(\boldsymbol{b}_{(\ell+1)}) \triangleq \sum_{n=1}^{\ell} b_n y_{n+j}^* + \frac{1}{2} b_{\ell+1} y_{\ell+j+1}^* = u_j(\boldsymbol{b}_{(\ell)}) + \frac{1}{2}\left(b_\ell y_{\ell+j}^* + b_{\ell+1} y_{\ell+1+j}^*\right).$$

This shows that we can recursively compute $g(\boldsymbol{b}_{(\ell+1)})$ and $\{u_j(\boldsymbol{b}_{(\ell+1)})\}_{0\le j\le P-1}$ from the previous $g(\boldsymbol{b}_{(\ell)})$ and $\{u_j(\boldsymbol{b}_{(\ell)})\}_{j=0}^{P-1}$ using $y_{\ell+1}, y_{\ell+2}, \ldots, y_{\ell+P}$ and $b_{\ell+1}$, and setting as initial condition $g(\boldsymbol{b}_{(0)}) = u_j(\boldsymbol{b}_{(0)}) = b_0 = 0$ for $0 \le j \le P-1$.

A final remark in this discussion is that although the computational burden of $\alpha_\ell^{(\eta)}$ in (3.14) increases linearly with ℓ, such a linearly increasing burden can be moderately compensated for by the fact that it is only necessary to compute $\alpha_\ell^{(\eta)}$ once for each ℓ and η, because it can be shared for all paths ending at level ℓ over the code tree η.

3.3.2 Heuristic Function φ_1

We next derive the first heuristic function that validates (2.3). Taking the maximum-likelihood metric g into the sufficient condition in (2.3) yields

$$\begin{aligned} &\sum_{m=1}^{\ell}\max_{\eta\in\Theta}\alpha_m^{(\eta)} - \frac{1}{2}\sum_{m=1}^{\ell}\sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_m b_n + \varphi(\boldsymbol{b}_{(\ell)}) \\ &\qquad \le \min_{\left\{\tilde{\boldsymbol{b}}\in\mathcal{C}\ :\ \tilde{\boldsymbol{b}}_{(\ell)}=\boldsymbol{b}_{(\ell)}\right\}}\left[\sum_{m=1}^{N}\max_{\eta\in\Theta}\alpha_m^{(\eta)} - \frac{1}{2}\sum_{m=1}^{N}\sum_{n=1}^{N} w_{m,n}^{(\theta)} b_m b_n + \varphi(\tilde{\boldsymbol{b}})\right]. \end{aligned}$$

Hence, in addition to $\varphi(\tilde{\boldsymbol{b}}) = 0$, the heuristic function should satisfy

$$\begin{aligned}\varphi(\boldsymbol{b}_{(\ell)}) &\leq \sum_{m=\ell+1}^{N} \max_{\eta\in\Theta} \alpha_m^{(\eta)} \\ &\quad - \max_{\left\{\tilde{\boldsymbol{b}}\in\mathcal{C}:\ \tilde{\boldsymbol{b}}_{(\ell)}=\boldsymbol{b}_{(\ell)}\right\}} \left(\sum_{m=\ell+1}^{N} \tilde{b}_m \sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_n + \frac{1}{2} \sum_{m=\ell+1}^{N} \sum_{n=\ell+1}^{N} w_{m,n}^{(\theta)} \tilde{b}_m \tilde{b}_n \right). \quad (3.15)\end{aligned}$$

It is apparent that the all-zero function is the largest one that satisfies this inequality subject to no dependence on the future route and future receptions, i.e., $\{\tilde{b}_m\}_{m\geq \ell+1}$ and $\{w_{m,n}^{(\theta)}\}_{m\geq\ell+1,n\geq\ell+1}$. Hence, we choose $\varphi_1(\boldsymbol{b}_{(\ell)}) = 0$.

Note that φ_1 is trivially on-the-fly computable, and hence so is f_1. In comparison with the exhaustive-search decoding, decoding based on recursive priority-first search shows a significant decrease in computational complexity especially at medium-to-high SNRs.

3.3.3 Heuristic Function φ_2

If we drop the requirement that the metric f must be independence of future receptions, we can further reduce the computational complexity. Upon reception of all $y_1, \ldots, y_L$, the heuristic function that satisfies (3.15) regardless of $\tilde{b}_{\ell+1}, \ldots, \tilde{b}_N$ can be increased to

$$\begin{aligned}\varphi_2(\boldsymbol{b}_{(\ell)}) &\triangleq \sum_{m=\ell+1}^{N} \max_{\eta\in\Theta} \alpha_m^{(\eta)} - \sum_{m=\ell+1}^{N} \left| \sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_n \right| - \frac{1}{2} \sum_{m=\ell+1}^{N} \sum_{n=\ell+1}^{N} \left| w_{m,n}^{(\theta)} \right| \\ &= \sum_{m=\ell+1}^{N} \max_{\eta\in\Theta} \alpha_m^{(\eta)} - \sum_{m=\ell+1}^{N} \left| v_m^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_\ell^{(\theta)}, \quad (3.16)\end{aligned}$$

where for $1 \leq \ell, m \leq N$ and $\theta \in \Theta$,

$$v_m^{(\theta)}(\boldsymbol{b}_{(\ell)}) \triangleq \sum_{n=1}^{\ell} w_{m,n}^{(\theta)} b_n = v_m^{(\theta)}(\boldsymbol{b}_{(\ell-1)}) + b_\ell w_{\ell,m}^{(\theta)}$$

and

$$\beta_\ell^{(\theta)} \triangleq \sum_{m=\ell+1}^{N} \left(\sum_{n=\ell+1}^{m-1} |w_{m,n}^{(\theta)}| + \frac{1}{2} |w_{m,m}^{(\theta)}| \right) = \beta_{\ell-1}^{(\theta)} - \sum_{n=\ell+1}^{N} |w_{\ell,n}^{(\theta)}| - \frac{1}{2} |w_{\ell,\ell}^{(\theta)}|$$

with initial conditions $v_m^{(\theta)}(\boldsymbol{b}_{(0)}) = b_0 = 0$, and $\beta_0^{(\theta)} = \sum_{m=1}^{N} \alpha_m^{(\theta)}$. Simulations show that compared to the zero-heuristic function φ_1, the heuristic function in (3.16) further reduces the number of path expansions during the decoding process up to one order of magnitude (cf. Table 3.1).

3.4 Simulation Results

In this section, we examine the performance of the codes proposed in Section 3.2. We also illustrate the decoding complexity of the maximum-likelihood priority-first search decoding algorithm presented in the previous section. For ease of comparison, the channel parameters used in our simulations follow those in [30], where $\boldsymbol{h}$ is zero-mean complex-Gaussian distributed with $E[\boldsymbol{h}\boldsymbol{h}^H] = (1/P)\mathbb{I}_P$ and $P = 2$. The average system SNR is thus given by

$$\frac{N}{L\sigma_n^2}\mathrm{tr}\left(E[\boldsymbol{h}\boldsymbol{h}^H]\frac{1}{N}\mathbb{B}^T\mathbb{B}\right) = \frac{N}{L\sigma_n^2}\mathrm{tr}\left(\frac{1}{NP}\mathbb{B}^T\mathbb{B}\right) = \frac{N}{L\sigma_n^2}, \tag{3.17}$$

since $\mathrm{tr}\left(\mathbb{B}^T\mathbb{B}\right) = NP$ for all simulated codewords.[4]

Figure 3.2 illustrates the simulation results of three codes: the computer-searched half-rate code obtained by the simulated annealing algorithm in [30] (SA-22), the constructed double-tree code with half of the codewords satisfying $\mathbb{B}^T\mathbb{B} = \mathbb{G}_{-1}$ and the remaining half satisfying $\mathbb{B}^T\mathbb{B} = \mathbb{G}_1$ (Double-22), and the constructed single-tree code whose codewords are all selected from the candidate sequences satisfying $\mathbb{B}^T\mathbb{B} = \mathbb{G}_{-1}$ (Single-22). We observe from Figure 3.2 that the Double-22 code performs almost the same as the SA-22 code. Actually, the simulations illustrated in Figure 3.3 provide evidence that the performance of the constructed double-tree half-rate codes is as good as the computer-searched half-rate codes for all $N > 12$. However, when $N \leq 12$, the Double-N code performs slightly worse

[4] The authors in [30] directly define the channel SNR as $1/\sigma_n^2$. It is apparent that their definition is exactly the limit of (3.17) as N approaches infinity.

Since it is assumed that an adequate guard period between two encoding blocks exists (so that there is no interference between two consecutive decoding blocks), the computation of the system SNR for finite N should be adjusted to account for this muting (but still part-of-the-decoding-block) guard period. For example, in comparison of the (6,3) and (20,10) codes over channels with memory order 1 (i.e., $P = 2$), one can easily observe that the former can only transmit 18 code bits in the time interval of 21 code bits, while the latter pushes out up to 20 code bits in the period of the same duration. Thus, under fixed code bit transmission power and fixed component noise power σ_n^2, it is reasonable for the (20,10) code to result in a higher SNR than the (6,3) code.

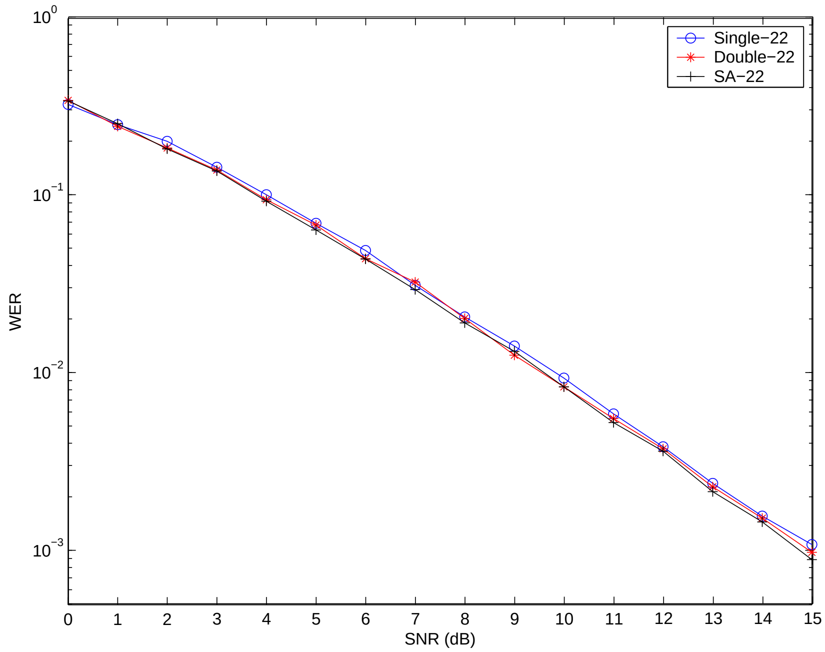

Figure 3.2: The maximum-likelihood word error rates (WERs) of the computer-searched half-rate code by simulated annealing in [30] (SA-22), the constructed half-rate code with double code trees (Double-22), and the constructed half-rate code with single code tree (Single-22). The codeword length is $N = 22$.

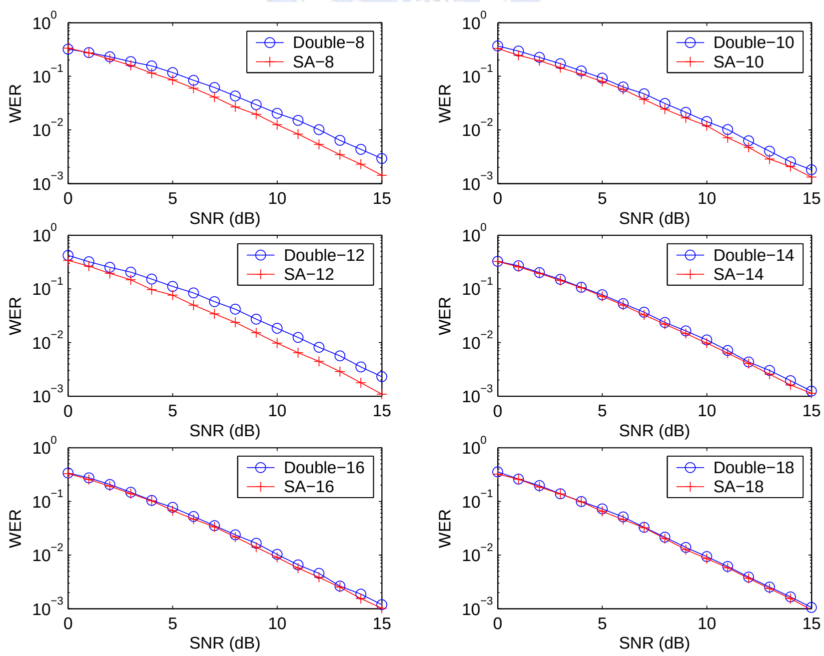

Figure 3.3: The maximum-likelihood word error rates (WERs) of the computer-searched code by simulated annealing (SA-N) and the constructed half-rate code with double code trees (Double-N).

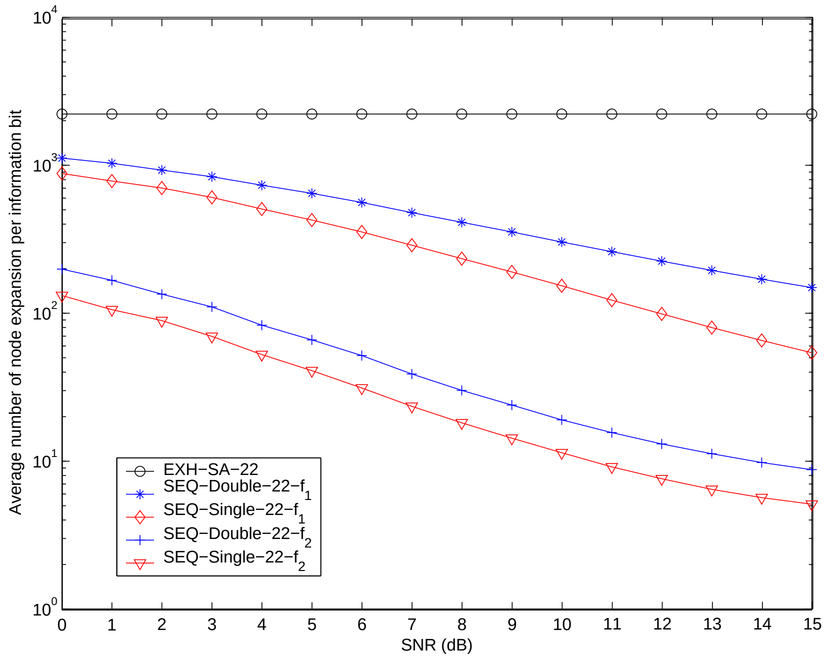

Figure 3.4: The average numbers of node expansions per information bit for the simulated-annealing-based computer-searched code in [30] by exhaustive decoding (EXH-SA-22), and the constructed single-tree (SEQ-Single-22) and double-tree (SEQ-Double-22) codes using the priority-first search decoding guided by either metric function f_1 or metric function f_2.

than the SA-N code. This is because for $N \leq 12$ the approximation in (3.9) can no longer be well maintained due to the restriction that $|\mathcal{A}(\boldsymbol{b}_{(\ell)}|\mathbb{G})|$ must be an integer.

In addition to the Double-22 code, Figure 3.2 also depicts simulation results of the Single-22 code. Since the pairwise codeword distance in the sense of (3.8) for the Single-22 code is in general smaller than that of the Double-22 code, its performance has a 0.2 dB degradation compared with that of the Double-22 code. However, we will see in Figure 3.4 that the Single-22 code actually has the smallest decoding complexity among the three codes. This suggests that to select codewords uniformly from a single code tree should not be ruled out as a candidate design, especially when the decoding complexity becomes the main system concern.

In Figure 3.4, the average numbers of node expansions per information bit are illustrated for the codes examined in Figure 3.2. Since the number of node expansions is exactly equal to the number of tree branch metrics (i.e., one recursion of f-function values) computed,

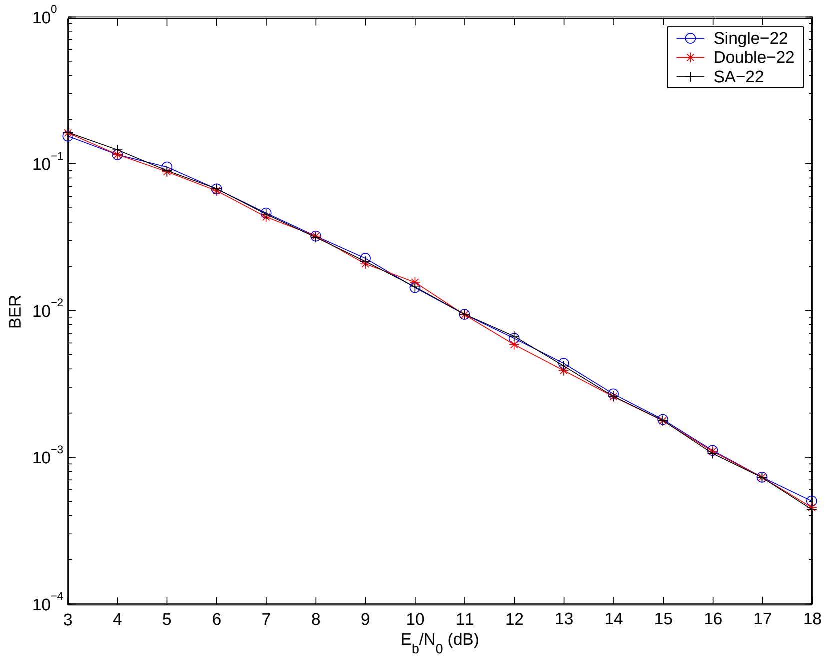

Figure 3.5: Bit error rates (BERs) for the simulation of codes illustrated in Figure 3.2.

the equivalent complexity of exhaustive decoding is correspondingly plotted. It can then be observed that in comparison with the exhaustive decoder, a significant reduction in computational burden is achieved at moderate-to-high SNRs by adopting the Double-22 code and the priority-first search decoder with on-the-fly computable metric f_1 (see (3.11)). Further reduction can be achieved if the Double-22 code is replaced with the Single-22 code. This is because performing the sequential search over multiple code trees introduces extra node expansions for those code trees that the transmitted codeword does not belong to. An additional order-of-magnitude reduction in node expansions can be achieved when the metric $f_2 = g + \varphi_2$ (see (3.12)) is used instead.

The authors in [7] and [30] only focus on the word error rate (WER). No bit error rate (BER) performances that involve the mapping design between the information bit patterns and the codewords are presented. Yet, in certain applications, such as voice transmission and digital radio broadcasting, the BER is generally considered a more critical performance index. In addition, the adoption of the BER performance index, as well as the signal-to-noise ratio per information bit, facilitates the comparison of codes of different code rates.

Figure 3.5 depicts the BER performance of the same codes whose WER performances were depicted in Figure 3.2. The corresponding E_b/N_0 is computed according to $E_b/N_0 = \text{SNR}/R$, where $R = K/N$ is the code rate. The mapping between the bit patterns and the codewords of the given computer-searched code is obtained through simulated annealing by minimizing the upper bound of

$$\text{BER} \le \frac{1}{2^K}\sum_{i=1}^{2^K}\sum_{j=1, j\neq i}^{2^K}\frac{d(\boldsymbol{m}_i, \boldsymbol{m}_j)}{K}\Pr\left(\hat{\boldsymbol{b}} = \boldsymbol{b}_j \middle| \boldsymbol{b}_i \text{ transmitted}\right),$$

where, other than the notations defined in (3.7), $\boldsymbol{m}_i$ is the information sequence corresponding to the i-th codeword, and $d(\cdot,\cdot)$ is the Hamming distance. For the constructed codes of Section 3.2.3, the binary representation of the index of the requested codeword in Step 1 is directly taken as the information bit pattern corresponding to the requested codeword. The result illustrated in Figure 3.5 then indicates that the BER performance of the three curves are almost the same. Hence we conclude that taking the binary representation of the requested codeword index as the information bit pattern for the constructed code not only makes its implementation easy, but also yields a BER performance similar to that of the best simulated-annealing-based computer-searched codes.

Lastly, we demonstrate the WER and BER performances, respectively, of Single-26, Double-26, Single-30, and Double-30 codes, together with those of Single-22 and Double-22 codes, over the quasi-static fading channels in Figures 3.6 and 3.7. Both figures show that the Double-30 code has the best maximum-likelihood performance not only in WER but also in BER. This result concurs with the intuition that a longer code will perform better provided that the channel coefficients remain unchanged in a coding block. The decoding complexities of the codes are listed in Table 3.1, from which we observe that the saving of decoding complexity of metric f_2 with respect to metric f_1 increases as the codeword length increases. It is worth mentioning that at very high SNR, the priority-first search decoding over the AWGN channels will directly go all the way down to the terminal nodes, and result in a decoding complexity of approximately two node expansions per information

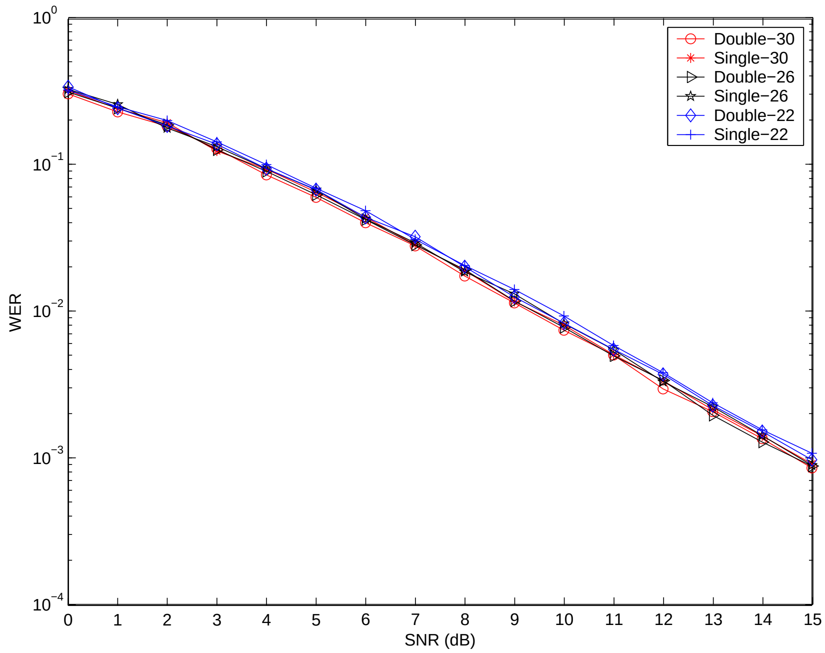

Figure 3.6: Word error rates (WERs) for the codes of Single-22, Double-22, Single-26, Double-26, Single-30 and Double-30.

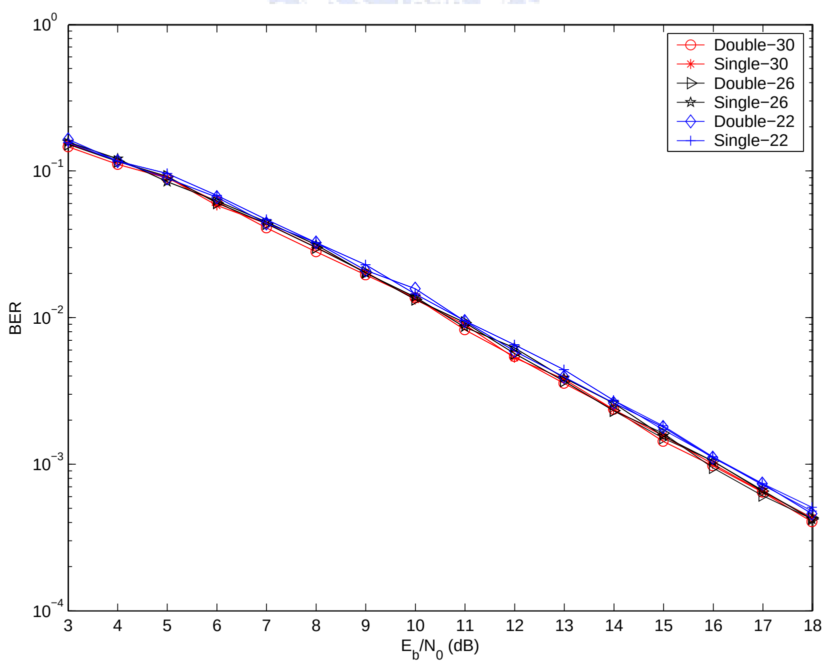

Figure 3.7: Bit error rates (BERs) for the codes of Single-22, Double-22, Single-26, Double-26, Single-30 and Double-30.

bit. However, for fading channels, the decoding complexity cannot reach the ideal two node expansions per information bit even with zero additive noise, as shown in the last column of Table 3.1. In this regard, metric f_2 still reaches a better ultimate decoding complexity than metric f_1.

Table 3.1: Average number of node expansions per information bit for the priority-first search decoding of the constructed half-rate codes of length 22, 26, and 30.

SNR	5dB	6dB	7dB	8dB	9dB	10dB	11dB	12dB	13dB	14dB	15dB	∞ dB
Double-22-f_1	671	590	506	436	375	320	274	236	204	178	156	54
Double-22-f_2	68	55	42	32	26	20	17	14	12	10	9	6
ratio of f_1/f_2	9.8	10.7	12.0	13.6	14.4	16.0	16.1	16.8	17.0	17.8	17.3	9.0
Double-26-f_1	2361	2006	1695	1416	1189	981	813	677	523	499	392	105
Double-26-f_2	175	130	94	69	53	39	29	23	18	15	13	6
ratio of f_1/f_2	13.5	15.4	18.0	20.5	22.4	25.2	28.0	29.4	29.1	33.3	30.2	17.5
Double-30-f_1	8455	7073	5760	5133	3759	3430	2644	1996	1765	1368	1081	192
Double-30-f_2	459	332	232	166	119	86	60	44	33	25	20	7
ratio of f_1/f_2	18.4	21.3	24.8	30.9	31.6	39.9	44.1	45.4	53.4	54.7	54.1	27.4
Single-22-f_1	460	371	308	250	200	163	130	105	85	69	57	12
Single-22-f_2	45	33	26	20	15	12	10	8	7	6	5	4
ratio of f_1/f_2	10.2	11.2	11.8	12.5	13.3	13.5	13.0	13.1	12.1	11.5	11.4	3.0
Single-26-f_1	1635	1328	1061	839	666	522	403	312	244	191	152	21
Single-26-f_2	112	79	57	42	31	23	17	13	11	9	7	4
ratio of f_1/f_2	14.6	16.8	18.6	20.0	21.5	22.7	23.7	23.9	22.2	21.2	21.7	5.3
Single-30-f_1	5871	4695	3857	2924	2335	1813	1328	884	805	572	416	39
Single-30-f_2	284	199	144	101	72	51	35	26	18	14	11	4
ratio of f_1/f_2	20.6	23.6	26.8	29.0	32.4	35.5	38.0	34.0	44.7	40.9	37.8	9.8

Table 3.2: The attained diversity levels of codes, which are least-square-approximated based on WER performance curves within 8–15 dBs.

Diversity	$N = 8$	$N = 10$	$N = 12$	$N = 14$	$N = 16$	$N = 18$	$N = 22$	$N = 26$	$N = 30$
SA-N	1.84	1.84	1.88	1.87	1.89	1.87	1.91	—	—
Single-N	—	—	—	—	—	—	1.89	1.90	1.87
Double-N	1.67	1.80	1.79	1.85	1.88	1.87	1.87	1.89	1.87

We close this section by commenting on the attained diversity level d of the simulated codes. The diversity level d serves as approximation of the word error probability at high SNR, i.e., $P_{\mathrm{e}} \approx \mathrm{SNR}^{-d}$. From Table 3.2, we observe that the attained diversities of codes of length 22 are around 1.9, which is close to the anticipated value of $P = 2$. The tables also suggest that the diversities degrade at small N, and the computer-searched codes have somewhat higher diversities within the considered SNR range. We conclude that under the constraint of the self-orthogonal structure, the simulated codes can turn the second delayed channel path into another diversity. This results in a blind detection performance of diversity level close to P.

3.5 Summary

In this chapter, we introduce an algorithm to construct codes that allow joint channel estimation and error correction at the receiver side of a block fading channel. In contrast to previously published codes, our codes are designed systematically and allow for an ML decoding with a much smaller computational complexity than the operation-intensive exhaustive decoding that was previously used in [7,13,30] to decode the structureless computer-searched codes. The given algorithm is based on the optimal signal-to-noise ratio framework and requires every codeword to satisfy a self-orthogonal property that helps to counter the effects of multi-path fading.

The improved decoding algorithm is a tree-based priority-first search decoding algorithm that uses a recursive maximum-likelihood metric. Simulations demonstrate that the constructed codes have almost identical performance as the best computer-searched codes, but with much smaller decoding complexity.

Moreover, we propose two different maximum-likelihood decoding metrics. The first one can be used in an on-the-fly fashion, while the second one that results in a much lower decoding complexity requires the knowledge of all channel outputs. We hence have a trade-off between decoding complexity versus decoding delay.

Chapter 4

Code Designs for Frequency-Selective Varying Fading Channels

In previous chapter, also in [4], [30] and [13], it is assumed that the channel coefficients $\boldsymbol{h}$ are invariant in each coding block of length $L = N + P - 1$. In this chapter, we will show that the approaches employed in previous sections can also be applicable to the situation that $\boldsymbol{h}$ may change in every Q symbol, where $Q < L$. Our simulation results suggest that our code that takes into consideration the varying characteristic of channels can achieve better performance at median-to-high signal-to-noise ratio over the computer-searched, union-bound-minimized code of length less than the varying subblock size. A side advantage of our code construction scheme is that its systematic structure makes it maximum-likelihoodly decodable by the priority-first search algorithm. The decoding complexity is therefore significantly decreased in contrast to that of exhaustive decoder for the structureless computer-searched codes.

The remainder of this chapter is organized as follows. After introducing the system model, we presents our code design scheme that is devised for $Q < L$. Then, the maximum-likelihood metric that can be used by priority-first search decoding is derived in Section 4.3. Simulations are summarized in Section 4.4 and concluding remarks are given in Section 4.5.

4.1 System Model

For $1 \leq k \leq M = \lceil L/Q \rceil$, let $\boldsymbol{h}_k \triangleq [h_{1,k} \quad h_{2,k} \quad \cdots \quad h_{P,k}]^T$ be the constant channel coefficients at the kth sub-block. Denote by $\boldsymbol{b}_k = [b_{(k-1)Q-P+2} \quad \cdots \quad b_{(k-1)Q+1} \quad \cdots \quad b_{kQ}]^T$ the portion of $\boldsymbol{b}$, which will affect the output portion $\boldsymbol{y}_k = [y_{(k-1)Q+1} \quad y_{(k-1)Q+2} \quad \cdots \quad y_{kQ}]$, where we assume $b_j = 0$ for $j \leq 0$ and $j > N$ for notational convenience. Then, for a channel whose coefficients change in every Q symbol, the system model defined in (3.1) remains as $\boldsymbol{y} = \mathbb{B}\boldsymbol{h} + \boldsymbol{n}$ except that both $\boldsymbol{y}$ and $\boldsymbol{n}$ extend as $MQ \times 1$ vectors with $y_j = n_j = 0$ for $j > L$, and $\mathbb{B}$ and $\boldsymbol{h}$ have to be re-defined as

$$\mathbb{B} \triangleq \mathbb{B}_1 \oplus \mathbb{B}_2 \oplus \cdots \oplus \mathbb{B}_M \quad \text{and} \quad \boldsymbol{h} \triangleq \begin{bmatrix} \boldsymbol{h}_1^H & \boldsymbol{h}_2^H & \cdots & \boldsymbol{h}_M^H \end{bmatrix}^H,$$

where $\mathbb{B}_k = [\mathbf{0}_{Q\times(P-1)} \quad \mathbb{I}_Q][\boldsymbol{b}_k \quad \tilde{\mathbb{E}}\boldsymbol{b}_k \quad \cdots \quad \tilde{\mathbb{E}}^{P-1}\boldsymbol{b}_k]$ is a $Q \times P$ matrix,

$$\tilde{\mathbb{E}} \triangleq \begin{bmatrix} 0 & 0 \cdots & 0 & 0 \\ 1 & 0 \ddots & 0 & 0 \\ 0 & 1 \ddots & 0 & 0 \\ 0 & 0 \cdots & 1 & 0 \end{bmatrix}_{(Q+P-1)\times(Q+P-1)}$$

and "$\oplus$" is the direct sum operator of two matrices.

Based on the new system model, we have $\mathbb{P}_B = \mathbb{P}_{B_1} \oplus \mathbb{P}_{B_2} \oplus \cdots \oplus \mathbb{P}_{B_M}$, where $\mathbb{P}_{B_k} = \mathbb{B}_k(\mathbb{B}_k^T\mathbb{B}_k)^{-1}\mathbb{B}_k^T$, and Eq. (3.6) becomes:

$$\hat{\boldsymbol{b}} = \arg\max_{\boldsymbol{b}\in\mathcal{C}} \sum_{k=1}^{M} \left\| \text{vec}(\boldsymbol{y}_k\boldsymbol{y}_k^H) - \text{vec}(\mathbb{P}_{B_k}) \right\|^2 \tag{4.1}$$

Again, codeword $\boldsymbol{b}$ and transformed codeword $\mathbb{P}_B$ is not one-to-one corresponding unless the first element of $\boldsymbol{b}$, namely b_1, is fixed.[1]

[1]It can be derived that given $Q \geq P$ and $\mathbb{B}_k^T\mathbb{B}_k = \mathbb{G}_k$ for $1 \leq k \leq M$,

$$\begin{cases} b_{Q-P+2} &= b_1 \times (-1)^{(Q-P+1-\gamma_{P,P-1,1})/2} \\ b_{kQ-P+2} &= b_{(k-1)Q-P+2} \times (-1)^{(Q-\gamma_{P,P-1,k})/2} \text{ for } k = 2, \cdots, M-1 \end{cases}$$

where $\gamma_{i,j,k}$ is the (i,j)th entry of the symmetric matrix $\mathbb{G}_k$ for $1 \leq i,j \leq P$, and, in our setting, $\gamma_{P,P-1,k} \in \{0, \pm 1\}$ should be chosen to make the exponent of (-1) an integer. Therefore, the first bit in each $\boldsymbol{b}_k$ is fixed once b_1 is set, which indicates that with the knowledge of b_1, codeword $\boldsymbol{b}$ can be uniquely determined by transformed codeword P_B.

4.2 Code Design

We summarize the proposed code construction scheme [35] in the following algorithm.

Step 1. Fix $b_1 = -1$,[2] *and choose a certain integer* Δ *defined later. Find* 2^K *codewords of the* (N, K) *code by repeating Steps 2–4 for* $0 \le i \le 2^K - 1$.

Step 2. Let $\rho_{\min} = 0$ *and* $\rho = i \cdot \Delta$.

Step 3. For $\ell = 2$ *to* N*, assign the* ℓ*-th bit of the* i*-th codeword,* b_ℓ*, according to that if* $\rho < \rho_{\min} + \gamma_\ell$*, then* $b_\ell = -1$*; else,* $b_\ell = 1$ *and* $\rho_{\min} = \rho_{\min} + \gamma_\ell$*, where*

$$\gamma_\ell = |\mathcal{A}_P(b_1, \ldots, b_{\ell-1}, b_\ell = -1)|,$$

which will be defined shortly.

Step 4. Store the i*th codeword* $\boldsymbol{b}$*, and goto Step 2 for the next codeword until all* 2^K *codewords are selected.*

Since $\mathbb{B}^T\mathbb{B} = (\mathbb{B}_1^T\mathbb{B}_1) \oplus (\mathbb{B}_2^T\mathbb{B}_2) \oplus \cdots \oplus (\mathbb{B}_M^T\mathbb{B}_M)$, the maximization of system SNR can be achieved simply by assigning

$$\mathbb{B}_1^T\mathbb{B}_1 = \mathbb{B}_2^T\mathbb{B}_2 = \cdots = \mathbb{B}_M^T\mathbb{B}_M = Q \cdot \mathbb{I}_Q \tag{4.2}$$

regardless of the statistics of $\boldsymbol{h}$ if such assignment is possible. Due to the same reason mentioned in Section 3.2.1, approximation to (4.2) will have to be taken in the true code design. Then, for channels of memory order 1, i.e., $P = 2$, $\mathcal{A}_P(b_1, \ldots, b_{\ell-1}, b_\ell = -1)$ is the set of all binary ± 1-sequences of length N, whose first bits are assigned as the arguments indicate, and which at the same time satisfy (4.2).

It remains to determine the number of all possible ± 1-sequences of length N, whose first ℓ bits equal b_1, b_2, …, b_ℓ subject to $\mathbb{B}_k^T\mathbb{B}_k = \mathbb{G}_k$ for $1 \le k \le M$.

[2] Codeword $\boldsymbol{b}$ and $\{\mathbb{P}_{B_k}\}_{k=1}^M$ is not one-to-one corresponding unless the first element of $\boldsymbol{b}$, namely b_1, is fixed. We thus fix $b_1 = -1$ in our code design.

Lemma 2. *Fix $P = 2$ and $Q \geq P$, and put*

$$\mathbb{B}_1^T\mathbb{B}_1 = \begin{bmatrix} Q & c_1 \\ c_1 & Q-1 \end{bmatrix}, \quad \mathbb{B}_k^T\mathbb{B}_k = \begin{bmatrix} Q & c_k \\ c_k & Q \end{bmatrix} \text{ for } 2 \leq k \leq M-1,$$
$$\text{and } \mathbb{B}_M^T\mathbb{B}_M = \begin{bmatrix} N-(M-1)Q & c_M \\ c_M & N-(M-1)Q+1 \end{bmatrix}, \tag{4.3}$$

where in our code selection process, $[c_1, c_2, \cdots, c_M] \in \{0, \pm 1\}^M$ will be chosen such that $Q - 1 + c_1$, $Q + c_k$ for $2 \leq k \leq M-1$, and $N - (M-1)Q + c_M$ are all even. Then, the number of all possible ± 1-sequences of length N, whose first ℓ bits equal b_1, b_2, …, b_ℓ subject to (4.3) is given by:

$$\begin{cases} \binom{Q-(\ell \bmod Q)}{\frac{Q-(\ell \bmod Q)+c_\tau - m_\ell}{2}} \left[\prod_{k=\tau+1}^{M-1} \binom{Q}{\frac{Q+c_{k+1}}{2}}\right] \binom{N-(M-1)Q}{\frac{N-(M-1)Q+c_M}{2}} \mathbf{1}\left\{|c_\tau - m_\ell| \leq Q - (\ell \bmod Q)\right\}, \\ \hfill \text{for } 1 \leq \tau < M; \\ \binom{N-(M-1)Q}{\frac{N-(M-1)Q+c_M-m_\ell}{2}} \mathbf{1}\left\{|c_M - m_\ell| \leq N-(M-1)Q\right\}, \hfill \text{for } \tau = M \end{cases}$$

where $\tau = \lfloor \ell/Q \rfloor + 1$, and

$$m_\ell = \begin{cases} 0, & \ell = 1 \text{ or } (\ell = (\tau-1)Q \text{ and } 2 \leq \tau \leq M); \\ b_1 b_2 + \cdots + b_{\ell-1}b_\ell, & 1 < \ell < Q; \\ b_{(\tau-1)Q}b_{(\tau-1)Q+1} + \cdots + b_{\ell-1}b_\ell, & (\tau-1)Q < \ell < \tau Q \text{ and } 2 \leq \tau \leq M. \end{cases}$$

Proof. It requires

$$\begin{cases} c_1 &= b_1 b_2 + \cdots + b_{Q-1}b_Q \\ & \vdots \\ c_\tau &= b_{(\tau-1)Q}b_{(\tau-1)Q+1} + \cdots + b_\ell b_{\ell+1} + \cdots + b_{\tau Q-1}b_{\tau Q} \\ &= m_\ell + b_\ell b_{\ell+1} + \cdots + b_{\tau Q-1}b_{\tau Q} \\ & \vdots \\ c_M &= b_{(M-1)Q}b_{(M-1)Q+1} + \cdots + b_{N-1}b_N \end{cases}$$

the number of all possible ± 1-sequences of length N, whose first ℓ bits equal b_1, b_2, …, b_ℓ subject to (4.3), is given by:

$$\begin{aligned} &\binom{kQ-\ell}{(kQ-\ell+c_k-m_\ell)/2} \mathbf{1}\left\{|c_k - m_\ell| \leq kQ-\ell\right\} \\ &\times \binom{Q}{(Q+c_{k+1})/2} \mathbf{1}\left\{|c_{k+1}| \leq Q\right\} \times \cdots \times \binom{Q}{(Q+c_{M-1})/2} \mathbf{1}\left\{|c_{M-1}| \leq Q\right\} \\ &\times \binom{N-(M-1)Q}{(N-(M-1)Q+c_M)/2} \mathbf{1}\left\{|c_M| \leq N-(M-1)Q\right\}. \end{aligned}$$

The proof is completed by noting that ℓ is equal to $(\tau-1)Q+(\ell \bmod Q)$, and $|c_k| \le Q$ and $|c_M| \le N-(M-1)Q$ are always valid. $\square$

It remains to determine the integer Δ. In order to have adequate number of codewords selected, Δ must satisfy

$$\Delta \le \frac{|\mathcal{A}_P(b_1=-1)|}{2^K-1}. \tag{4.4}$$

Therefore, we let Δ be the largest integer satisfying (4.4), i.e., $\Delta = |\mathcal{A}_P(b_1=-1)|/(2^K-1)$. With the availability of the above lemma and Δ, the code construction algorithm mentioned above can be performed.

4.3 Optimal Priority-First Search Decoding

Next, we re-derive the maximum-likelihood decoding metric for use of priority-first search decoding algorithm. Continuing the derivation from (4.1) based on $\mathbb{B}_k^T\mathbb{B}_k = \mathbb{G}_{\theta,k}$ for $1 \le k \le M$ and $1 \le \theta \le \Theta$, we can establish in terms of similar procedure as in Section 3.3.1 that:

$$\hat{\boldsymbol{b}} = \arg\min_{\boldsymbol{b}\in\mathcal{C}} \frac{1}{2}\sum_{k=1}^{M}\sum_{m=1}^{Q+P-1}\sum_{n=1}^{Q+P-1}\left[-w_{m,n,k}^{(\theta)} b_{(k-1)Q-P+m+1} b_{(k-1)Q-P+n+1}\right]\mathbf{1}\{\boldsymbol{b}\in\mathcal{C}_\theta\}$$

where for $1 \le m,n \le Q+P-1$,

$$w_{m,n,k}^{(\theta)} = \sum_{i=0}^{P-1}\sum_{j=0}^{P-1}\delta_{i,j,k}^{(\theta)}\mathrm{Re}\{\tilde{y}_{m+i,k}\tilde{y}_{n+j,k}^*\},$$

$\delta_{i,j,k}^{(\theta)}$ is the (i,j)th entry[3] of $\mathbb{D}_{\theta,k} = \mathbb{G}_{\theta,k}^{-1}$, and $\tilde{\boldsymbol{y}}_k = [\mathbf{0}_{1\times(P-1)}\ \boldsymbol{y}_k^H\ \mathbf{0}_{1\times(P-1)}]^H = [\tilde{y}_{1,k}\ \cdots\ \tilde{y}_{Q+2P-2,k}]^T$. As it turns out, the recursive on-the-fly metric for the priority-first search decoding algorithm

[3] Under the assumption that $Q \ge P$, the ith diagonal element of the target $\mathbb{G}_{\theta,1}$ is given by $Q-i+1$, and the diagonal elements of the target $\mathbb{G}_{\theta,k}$ are equal to Q for $2 \le k < M$; hence, their inverse matrices exist. However, when $P \le N-(M-1)Q$, $\mathbb{G}_{\theta,M}$ has no inverse. In such case, we re-define $\mathbb{D}_{\theta,M}$ as:

$$\mathbb{D}_{\theta,M} \triangleq \mathbf{0}_{[N-(M-1)Q]\times[N-(M-1)Q]} \oplus \mathbb{G}_{\theta,M}^{-1}(N-(M-1)Q+1),$$

where $\mathbb{G}_{\theta,M}(j)$ is a $(P-j+1)\times(P-j+1)$ matrix that contains the jth to Pth rows and the jth to Pth columns of $\mathbb{G}_{\theta,M}$.

is:

$$g(\boldsymbol{b}_{(\ell)}) - g(\boldsymbol{b}_{(\ell-1)}) = \begin{cases} \max\limits_{1\le\eta\le\Theta} \alpha_{s,k}^{(\eta)} - b_\ell \sum\limits_{i=0}^{P-1}\sum\limits_{j=0}^{P-1} \delta_{i,j,k}^{(\theta)} \mathrm{Re}\{\tilde{y}_{s+i,k}\cdot u_{j,k}(\boldsymbol{b}_{(\ell)})\}, \text{ for } P \le s \le Q \\ \max\limits_{1\le\eta\le\Theta} \alpha_{r,k}^{(\eta)} + \max\limits_{1\le\eta\le\Theta} \alpha_{s,k+1}^{(\eta)} - b_\ell \sum\limits_{i=0}^{P-1}\sum\limits_{j=0}^{P-1} \Big(\delta_{i,j,k}^{(\theta)} \mathrm{Re}\{\tilde{y}_{r+i,k}\cdot u_{j,k}(\boldsymbol{b}_{(\ell)})\} \\ \qquad + \delta_{i,j,k+1}^{(\theta)} \mathrm{Re}\{\tilde{y}_{s+i,k+1}\cdot u_{j,k+1}(\boldsymbol{b}_{(\ell)})\}\Big), \text{ otherwise.} \end{cases}$$

where $-P+2 \le \ell \le N$, $s = [(\ell+P-2) \bmod Q]+1$, $r = s+Q$, $k = \max\{\lceil \ell/Q \rceil, 1\}$,

$$\alpha_{s,k}^{(\eta)} \triangleq \sum_{n=1}^{s-1} \left| w_{s,n,k}^{(\eta)} \right| + \frac{1}{2}\left| w_{s,s,k}^{(\eta)} \right|$$

and

$$u_{j,k}(\boldsymbol{b}_{(\ell+1)}) \triangleq u_{j,k}(\boldsymbol{b}_{(\ell)}) + \frac{1}{2}\left(b_\ell \tilde{y}_{s+j,k}^* + b_{\ell+1}\tilde{y}_{s+j+1,k}^* \right)$$

with initial values $g(\boldsymbol{b}_{(-P+1)}) = 0$ and $u_{j,k}(\boldsymbol{b}_{((k-1)Q-P+2)}) = 0$ for $0 \le j \le P-1$ and $1 \le k \le M$. In addition, the low-complexity heuristic function is given by:

$$\varphi_2(\boldsymbol{b}_{(\ell)}) \triangleq \begin{cases} \sum\limits_{m=s+1}^{Q+P-1} \max\limits_{1\le\eta\le\Theta} \alpha_{m,k}^{(\eta)} - \sum\limits_{m=s+1}^{Q+P-1} \left| v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k}^{(\theta)} \\ \quad + \sum\limits_{\kappa=k+1}^{M} \left(\sum\limits_{m=1}^{Q+P-1} \max\limits_{1\le\eta\le\theta} \alpha_{m,\kappa}^{(\eta)} - \beta_{0,\kappa}^{(\theta)} \right), & \text{for } P \le s \le Q; \\ \sum\limits_{m=s+1}^{Q+P-1} \max\limits_{1\le\eta\le\Theta} \alpha_{m,k+1}^{(\eta)} - \sum\limits_{m=s+1}^{Q+P-1} \left| v_{m,k+1}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k+1}^{(\theta)} \\ \quad + \sum\limits_{m=r+1}^{Q+P-1} \max\limits_{1\le\eta\le\Theta} \alpha_{m,k}^{(\eta)} - \sum\limits_{m=r+1}^{Q+P-1} \left| v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{r,k}^{(\theta)} \\ \quad + \sum\limits_{\kappa=k+2}^{M} \left(\sum\limits_{m=1}^{Q+P-1} \max\limits_{1\le\eta\le\theta} \alpha_{m,\kappa}^{(\eta)} - \beta_{0,\kappa}^{(\theta)} \right), & \text{otherwise,} \end{cases}$$

where s, r and k are defined the same as for $g(\cdot)$,

$$v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell)}) \triangleq \sum_{n=1}^{s} w_{m,n,k}^{(\theta)} b_{(k-1)Q+P+n-1} = v_{m,k}^{(\theta)}(\boldsymbol{b}_{(\ell-1)}) + w_{s,m,k}^{(\theta)} b_\ell,$$

and

$$\beta_{s,k}^{(\theta)} \triangleq \sum_{m=s+1}^{Q+P-1} \left(\sum_{n=s+1}^{m-1} \left| w_{m,n,k}^{(\theta)} \right| + \frac{1}{2}\left| w_{m,m,k}^{(\theta)} \right| \right) = \beta_{s-1,k}^{(\theta)} - \sum_{n=s+1}^{Q+P-1} \left| w_{s,n,k}^{(\theta)} \right| - \frac{1}{2}\left| w_{s,s,k}^{(\theta)} \right|$$

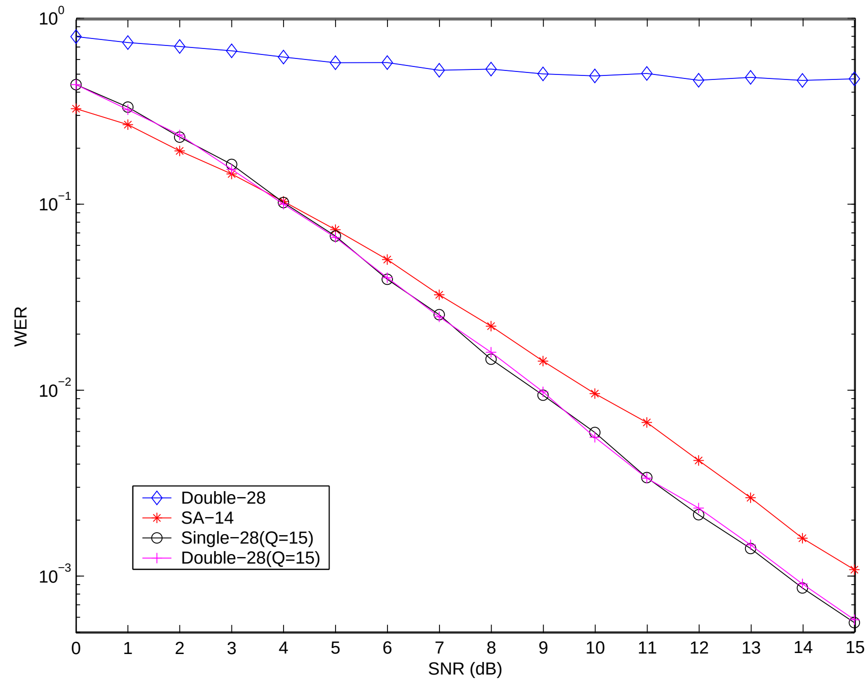

Figure 4.1: Word error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} = 15$.

with initial values $v_{m,k}^{(\theta)}(\boldsymbol{b}_{(k-1)Q-P+2}) = 0$ and $\beta_{0,k}^{(\theta)} = \sum_{m=1}^{Q+P-1} \alpha_{m,k}^{(\theta)}$.

It is worth mentioning that if the single-tree code is adopted, $\varphi_2(\cdot)$ can be further reduced to:

$$\varphi_2(\boldsymbol{b}_{(\ell)}) \triangleq \begin{cases} \displaystyle\sum_{m=s+1}^{Q+P-1} \alpha_{m,k}^{(1)} - \sum_{m=s+1}^{Q+P-1} \left| v_{m,k}^{(1)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k}^{(1)} & \text{for } P \le s \le Q; \\ \displaystyle\sum_{m=s+1}^{Q+P-1} \alpha_{m,k+1}^{(1)} - \sum_{m=s+1}^{Q+P-1} \left| v_{m,k+1}^{(1)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{s,k+1}^{(1)} & \\ \displaystyle\quad + \sum_{m=r+1}^{Q+P-1} \alpha_{m,k}^{(1)} - \sum_{m=r+1}^{Q+P-1} \left| v_{m,k}^{(1)}(\boldsymbol{b}_{(\ell)}) \right| - \beta_{r,k}^{(1)} & \text{otherwise,} \end{cases}$$

since $\sum_{m=1}^{Q+P-1} \max_{1\le\eta\le\theta} \alpha_{m,\kappa}^{(\eta)} - \beta_{0,\kappa}^{(\theta)} = \sum_{m=1}^{Q+P-1} \alpha_{m,\kappa}^{(1)} - \beta_{0,\kappa}^{(1)} = 0$; hence, a sub-blockwise low-complexity on-the-fly decoding can indeed be conducted under the single code tree condition.

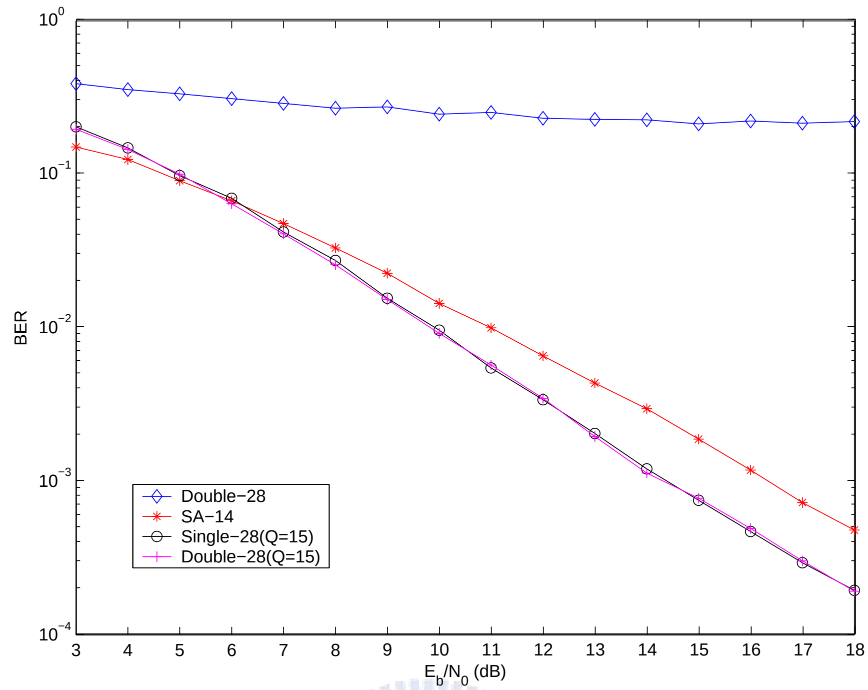

Figure 4.2: Bit error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} = 15$.

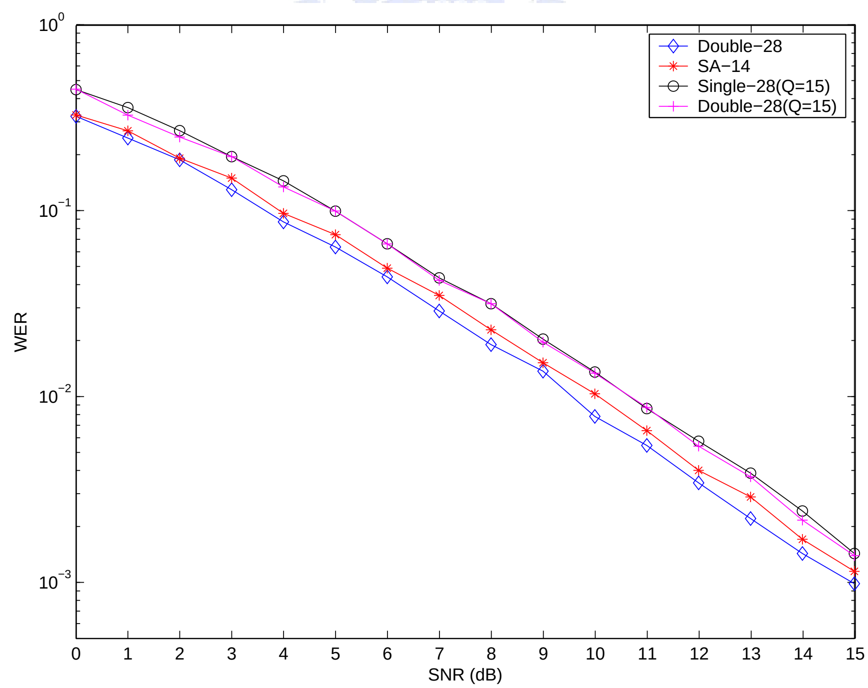

Figure 4.3: Word error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} \geq L$.

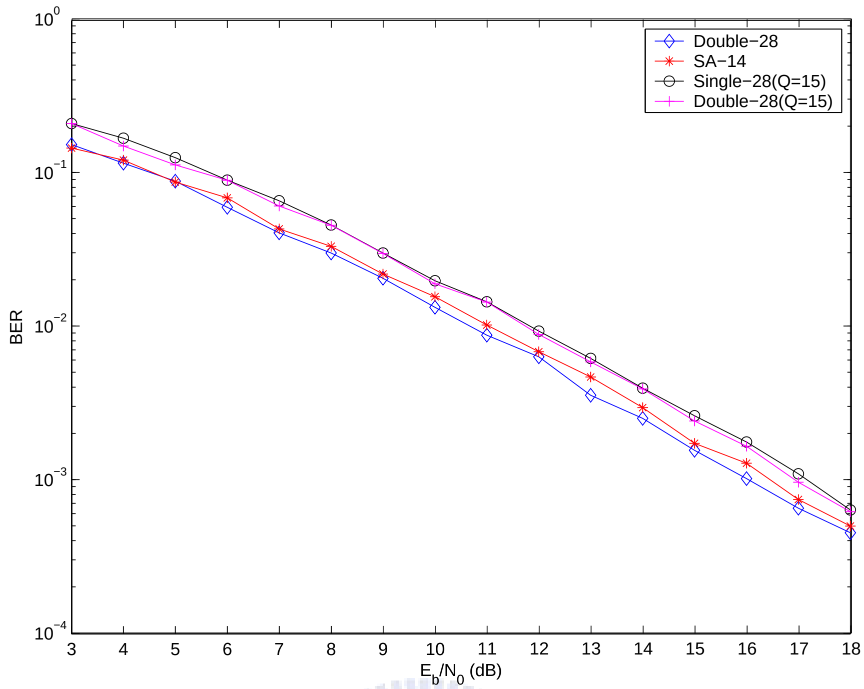

Figure 4.4: Bit error rates (BERs) for the codes of Double-28, SA-14, Single-28(Q=15) and Double-28(Q=15) over the quasi-static channel with $Q_{\text{chan}} \geq L$.

Table 4.1: Average numbers of node expansions per information bit for the codes of Single-28(Q=15) and Double-28(Q=15) using the priority-first search decoding guided by either evaluation function f_1 or evaluation function f_2 over the quasi-static channel with $Q_{\text{chan}} = 15$.

SNR	3dB	4dB	5dB	6dB	7dB	8dB	9dB
Double-28(Q=15)-f_1	2860	2440	2076	1790	1564	1359	1200
Double-28(Q=15)-f_2	1271	1029	877	685	582	484	413
ratio of f_1/f_2	2.3	2.4	2.4	2.6	2.7	2.8	2.9
Single-28(Q=15)-f_1	1658	1367	1074	899	701	593	488
Single-28(Q=15)-f_2	766	625	482	392	321	254	219
ratio of f_1/f_2	2.2	2.2	2.2	2.3	2.2	2.3	2.2

SNR	10dB	11dB	12dB	13dB	14dB	15dB
Double-28(Q=15)-f_1	1040	958	899	811	780	723
Double-28(Q=15)-f_2	353	312	277	250	229	207
ratio of f_1/f_2	2.9	3.1	3.2	3.2	3.4	3.5
Single-28(Q=15)-f_1	448	356	309	277	244	232
Single-28(Q=15)-f_2	177	149	133	121	104	92
ratio of f_1/f_2	2.5	2.4	2.3	2.3	2.3	2.5

4.4 Simulation Results

Figures 4.1 and 4.2 compare four codes over fast-fading channels whose channel coefficients vary in every 15-symbol period. Notably, we will use Q_{chan} to denote the varying period of the channel coefficients $\boldsymbol{h}$, and retain Q as the design parameter for the nonlinear codes. In notations, "Double-28" and "SA-14" denote the codes defined in the previous sections, and "Single-28(Q=15)" and "Double-28(Q=15)" are the codes constructed based on the rule introduced in this section under the design parameter $Q = 15$. Again, the mapping between the bit patterns and codewords for the SA-14 code is defined by simulated annealing.

Both Figs. 4.1 and 4.2 show that the Double-28 code seriously degrades when the channel coefficients unexpectedly vary in an intra-codeword fashion. This hints that the assumption that the channel coefficients remain constant in a coding block is critical in the code design in Section 3.2. Figures 4.3 and 4.4 then indicate that the codes taking into considerations the varying nature of the channel coefficients within a codeword is robust in its performance when being applied to channels with constant coefficients. Thus, we may conclude that for a channel whose coefficients vary more often than a coding block, it is advantageous to use the code design for a fast-fading environment considered in the section.

A more striking result from Fig. 4.1 is that even if the codeword length of the Single-28(Q=15) and the Double-28(Q=15) codes is twice of the SA-14 code, their word error rates are still markedly superior at medium-to-high SNRs. Note that the SA-14 code is the computer-optimized code specifically for $Q_{\text{chan}} = 15$ channel. This hints that when the channel memory order is known, performance gain can be obtained by considering the inter-subblock correlation, and favors a longer code design. The gain can be regarded as obtaining from a time diversity due to varying channels.

The decoding complexity, measured in terms of average number of node expansions per information bit, for codes of Single-28(Q=15) and Double-28(Q=15) are illustrated in Tab. 4.1. Similar observation is attained that the decoding metric f_2 yields less decoding complexity than the on-the-fly decoding one f_1; however, the saving in complexity reduces when channels with fast-fading are considered.

4.5 Summary

An extension of the code design for combined channel estimation and error correction to channels with independently varying fading subblocks is established in this chapter. This design can directly construct a code of any desired code length and code rate, of which the performance is shown to be comparable to the best computer-searched code for the channels simulated. Although we only derive the coding scheme and its decoding metric for a fixed Q, further extension to the situation that the channel coefficients $\boldsymbol{h}$ vary nonstationarily as the periods Q_1, Q_2, ..., Q_M are not equal is straightforward. Such design may be suitable for, e.g., the frequency-hopping scheme of Global System for Mobile communications (GSM) and Universal Mobile Telecommunications System (UMTS), or the time-hopping scheme in IS-54, in which cases the channel coefficients change (or hop) at protocol-aware scheduled time [19].

The performance of our constructed code can be further (slightly) improved if the codewords are selected uniformly from all feasible $(c_1, c_2, \cdots, c_M) \in \{-1, 0, 1\}^M$. For example, select only half (i.e., 2^{13}) of the codewords according to $c_1 = 0$ and $c_2 = -1$ for the $(28, 14)(Q = 15)$ code, and pick the remaining half of the codewords from those binary sequences satisfying (4.3) with $c_1 = 0$ and $c_2 = 1$. This however will slightly increase the decoding complexity. The trade-off between selecting codewords from fixed $(c_1, \ldots, c_M)$ or multiple $(c_1, \ldots, c_M)$'s is thus evident.

Chapter 5

Code Designs for Frequency-Nonselective Varying Fading Channels

The error correcting code design that jointly considers channel estimation is especially useful in situation when either the fading is rapid enough to preclude a good estimate of channel taps or the cost of implementing the channel estimators is high. One example is the reliable delivery of often short-in-length control signal such as channel quality indicator (CQI) in a highly mobile environment.

At this background, Xu *et al.* proposed a novel nonlinear coding scheme suitable for blind noncoherent detection of the transmitted control signal to the 802.16m standard body [39]. In the proposal, the uplink CQI information is encoded using a $(12, 6)$ code. The codeword will then be repeatedly transmitted three times (perhaps through different OFDM channels) in order to further benefit from diversity gain (which can be equivalently regarded as a $(36, 6)$ coding scheme).

Since most of the existing blind-detectable noncoherent codes are designed with the help of computer search, they exhibit no apparent structure for efficient decoding. The operation-intensive exhaustive search therefore becomes the only decoding option, of which the dramatically increasing decoding complexity prevents its practical use for codes of long codeword length or high code rate.

In this chapter, we take a different approach in such code design. Based on self-orthogonality framework, we propose a *systematic* (N, K) coding scheme that can deal with any given N and K for channels with possibly varying channel coefficients in a coding block. It is an extension of our previous work that targets the blind detection over channels with static (i.e., constant) channel coefficients during the transceiving of a codeword [35]. Simulations show that our constructed $(36, 6)$ code has almost the same performance as Xu's three-times-repetitive $(12, 6)$ code when the channel independently varies its coefficients three times in a coding block. In case the channel remains constant during the entire coding block, our constructed code has 0.7 dB performance improvement over Xu's code.

Xu's code is specifically designed for a frequency-nonselective OFDM system, while our systematic code construction scheme can also be applied in a frequency selective environment. Our simulation results indicate that with a proper design, a blind-detectable noncohrent code can be made robust for channels whose taps may vary more often than a coding block.

A side advantage of our code construction scheme is that its systematic structure makes it maximum-likelihoodly decodable by the priority-first search algorithm. Thus, when being compared with the operation-intensive exhaustive decoder, the decoding complexity is greatly reduced especially when codes of longer code length is adopted.

This chapter includes the following sections. Section 5.1 describes the system model we consider. Section 5.2 mentions our systematical codeword-selection procedure to construct codes for joint channel and data estimation for frequency-nonselective fading channels. Simulations are discussed in Section 5.3. Finally, Section 5.4 summarizes the chapter.

5.1 System Model

Suppose that a codeword $\boldsymbol{b} = [b_1 \ \cdots \ b_N]^T$ is transmitted over a block fading channel of memory order 0, of which channel coefficients may vary in every Q symbols, where $b_i \in \{\pm 1\}$ and $Q > P$. Then, the channel model we consider in this chapter actually is a special case of the model in Chapter 4 with $P = 1$. It can be derived that the joint maximum-likelihood decoder [4, 30] upon the reception of $\boldsymbol{y}$ is given by (2.2):

$$\begin{aligned} \hat{\boldsymbol{b}} &= \arg\min_{\boldsymbol{b}\in\mathcal{C}} \min_{\boldsymbol{h}} \|\boldsymbol{y} - \mathbb{B}\boldsymbol{h}\|^2 \\ &= \arg\max_{\boldsymbol{b}\in\mathcal{C}} \sum_{k=1}^{M} \left\|\boldsymbol{y}_k\boldsymbol{y}_k^H - \mathbb{P}_{B_k}\right\|^2 \\ &= \arg\max_{\boldsymbol{b}\in\mathcal{C}} \sum_{k=1}^{M} \|\boldsymbol{y}_k^H\boldsymbol{b}_k\|^2, \end{aligned} \tag{5.1}$$

where $\boldsymbol{y}_k \triangleq [y_{(k-1)Q+1} \ \ y_{(k-1)Q+2} \ \ \cdots \ \ y_{kQ}]$ is the output portion affected by $\boldsymbol{b}_k$, and $\mathbb{P}_{B_k} \triangleq \mathbb{B}_k(\mathbb{B}_k^T\mathbb{B}_k)^{-1}\mathbb{B}_k^T$. In the above derivation, we assume that the receiver, although it knows nothing about $\boldsymbol{h}$, has perfect knowledge about the values of P and Q.

5.2 Code Design

Now, as far as the code design for frequency nonselective channels is concerned, $\mathcal{A}_0(b_1, \ldots, b_\ell)$ is simply the set of all binary ± 1-sequences of length N, whose first ℓ bits are assigned as the arguments indicate, and which at the same time satisfies that

$$\begin{cases} \mathbb{B}_k^T\mathbb{B}_k = Q \text{ for } 1 \le k < M = \lceil L/Q \rceil \\ \mathbb{B}_M^T\mathbb{B}_M = N - (M-1)Q. \end{cases} \tag{5.2}$$

Again, the next step is to determine the integer Δ that satisfies (4.4). We however found that letting Δ be the largest integer satisfying (4.4) as we did in Section 4.2 may not generate the alphabetically uniform-pick code with the best error performance. In certain cases, the second largest integer satisfying (4.4) is indeed a better choice. Further investigation that follows along this direction suggests that a better choice of Δ will yield a code with larger minimum pairwise distance in the sense of $\sum_{k=1}^{M} \|\mathbb{P}_{\bar{B}_k} - \mathbb{P}_{B_k}\|^2$, where $\{\bar{B}_k\}_{k=1}^{M}$ and $\{B_k\}_{k=1}^{M}$ respectively correspond to codewords $\bar{\boldsymbol{b}}$ and $\boldsymbol{b}$.

It may not be practical to examine the minimum pairwise distance for all 2^K codewords for the determination of the best Δ. Instead, we choose K codewords as representatives. These representative codewords correspond to $\rho = 2^j\Delta$ for $0 \leq j \leq K-1$. We then adopt the Δ that minimizes the pairwise distance among these K codewords subject to (4.4).

When $N > K+4$ and $P = 0$, the proposed process of determining Δ is indeed equivalent to that the Δ-th codeword must be of the form

$$[\underbrace{-1 \cdots -1}_{K+1}\ 1\ 1\ \boldsymbol{u}\ 1],$$

where $\boldsymbol{u}$ is a maximum-length shift-register sequence. In other words, the first $K+3$ bits are fixed as $[-1\ \cdots\ -1\ \ 1\ \ 1]$, and the last bit is always equal to 1. This is because under $P = 0$, all binary ± 1-sequences satisfy (5.2), which results in that $(2^{j+1}\Delta)$-th codeword is exactly the logical left-shift of $(2^j\Delta)$-th codeword.

We close this section by pointing out that the size of set $\mathcal{A}_P(b_1, \ldots, b_\ell)$ for $P = 1$ has explicit formula as

$$|\mathcal{A}_P(b_1, \ldots, b_\ell)| = 2^{N-\ell}.$$

5.3 Simulation Results

In our simulations, the channel parameters follow those in [30], where $\boldsymbol{h}$ is zero-mean complex-Gaussian distributed with $E[\boldsymbol{h}\boldsymbol{h}^H] = (1/(P+1))\mathbb{I}_{P+1}$.

We first compare our constructed $(36, 6)$ code with Xu's three-times-repetitive $(12, 6)$ code over frequency nonselective channels. As shown in Figure 5.1, the two codes has comparable performance when channel coefficients vary independently in every 12 symbols. In case the

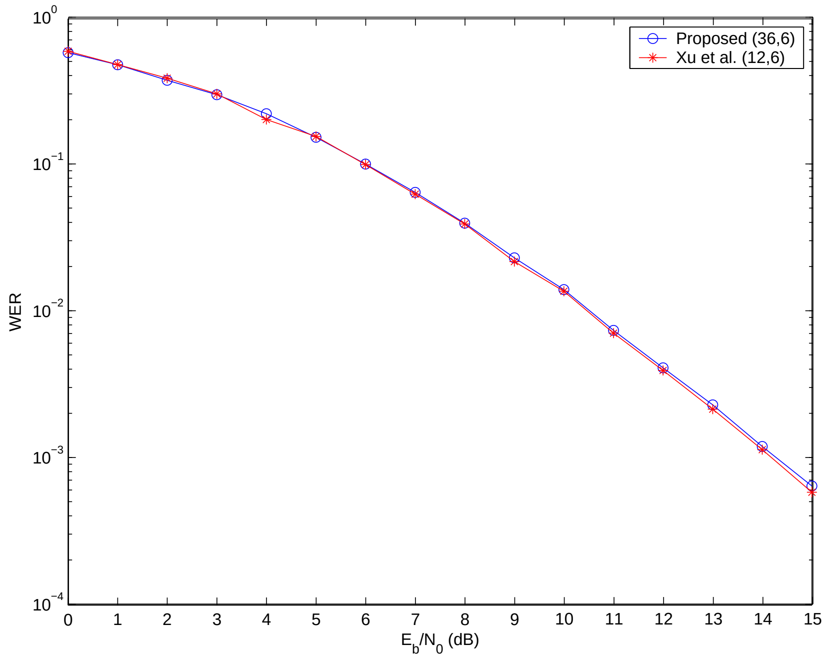

Figure 5.1: Word error rates (WERs) for the constructed $(36, 6)$ code and Xu's three-times-repetitive $(12, 6)$ code over flat fading channel with coefficients varying independently in every 12 symbols.

channel coefficients remain constant over the entire coding block, the proposed $(36, 6)$ code performs 0.7 dB better than Xu's code as shown in Figure 5.2. It should be emphasized that when $P = 0$, $\mathcal{A}_P(b_1, \ldots, b_\ell)$ is irrelevant to the design parameter Q; hence, the $(36, 6)$ code in Figure 5.1 is identical to the one used in Figure 5.2. This indicates that the proposed $(36, 6)$ code can adapt more robustly to the two simulated scenarios than Xu's code.

5.4 Summary

In this chapter, we propose a novel systematic code construction scheme for joint channel estimation and error correction for channels with independently varying fading subblocks. Unlike the existing noncoherent codes that are designed with the help of computer search, a code of desired code length and code rate can be directly generated with our coding scheme. We then compare our codes with the three-times-repetitive $(12, 6)$ code proposed by Xu *et al.* for use of channel quality indicator (CQI) in uplink control for IEEE 802.16m.

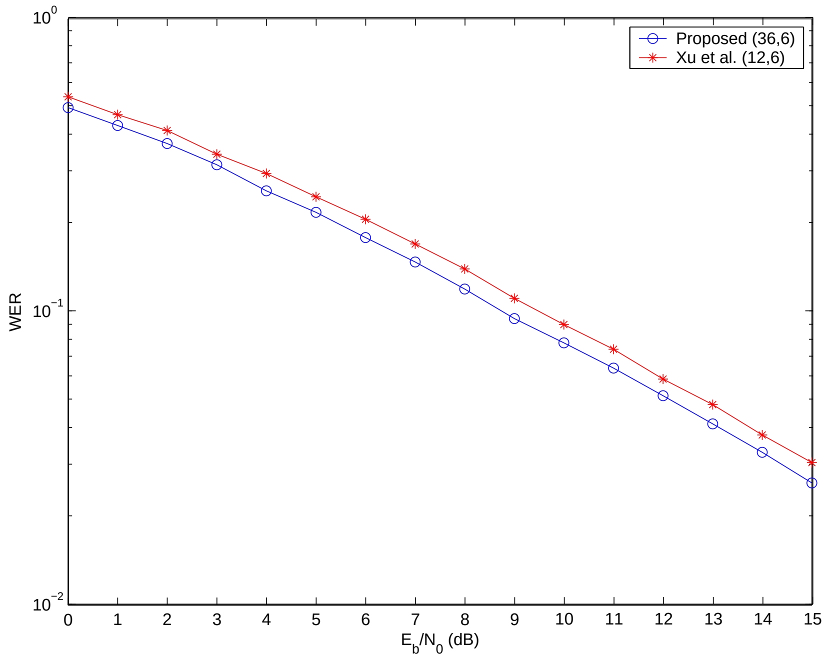

Figure 5.2: Word error rates (WERs) for the constructed $(36, 6)$ code and Xu's three-times-repetitive $(12, 6)$ code over flat fading channel with coefficients unchanged during the transmission of a codeword.

Simulations show that our constructed $(36, 6)$ code has comparable performance to Xu's code when channel coefficients changes randomly in every 12 symbols. If the channel taps remain constant in the entire coding block of length 36, our code outperforms Xu's code by 0.7 dB. This indicates that the new constructed code adapts more robustly to the two simulated scenarios.

Chapter 6

A Systematic Space-Time Code Design

Coding and transmission schemes for noncoherent receivers used in multiple-input multiple-output (MIMO) flat-fading channels can be roughly classified into two categories.[1] Schemes in the first category devise the space-time *constellations* for a given noncoherent receiver structure using computer search [1, 2, 16], while schemes in the second category couple the well-known space-time block codes with blind detection [21,22,31]. A brief summary of these schemes is as follows.

In [1], Beko *et al.* propose a two-phase code design approach, where the first phase produces a rough space-time code constellation that is subsequently refined in the second phase through a search-based geodesic descent optimization algorithm (GDA). In [2], Borran *et al.* uses the Kullback-Leibler distance as a design criterion to partition the signal space into several subsets, resulting in a reduction of number of parameters to be computer-searched. The authors in [16] construct unitary space-time signals by random search upon a Fourier-based structure, which only requires optimizing $L-1$ parameters instead of $L(L-1)/2$ in

[1]There are some notable papers that deal with similar problems, but cannot be classified into the two categories. For example, both [9] and [10] consider the so-called *training codes* that incorporate training symbols into their codewords. As anticipated, the receiver estimates the channel coefficients via training symbols. Such designs are very different from ours, which combines channel estimation and error correction by adopting joint maximum-likelihood decoding at the receiver. In [5], a noncoherent code is constructed through a mapping from coherent code. The code structure however only allows for a suboptimal efficient decoder.

the correlation matrix, where L is the number of space-time signals.

On the other hand, [21], [22] and [31] incorporate blind detection to existing space-time block codes. Based on the semidefinite relaxation (SDR) approach, an efficient suboptimal blind detection scheme is also suggested by Ma *et al.* in [21]. Later in [22], Ma further addresses the necessary properties for the family of orthogonal space-time block codes that can well co-work with blind detection.

Two main problems of designing codes or signal constellations based on unconstrained computer-search are that the design complexity is in general high, especially for codes of long block length, and the codes often need to be redesigned when design assumptions change. Moreover, their decoding depends mostly on operationally intensive exhaustive search, which further prevents their practical use in the case of long block lengths. Obviously, these problems can be solved by realizing a systematic code construction and its respective low-complexity decoder. Such an approach designed under two-transmit-antenna and half-rate condition is presented in this paper.

Furthermore, one main difference between our work and the existing works on combining known space-time block codes with blind detection, is that we aim at achieving a coding gain in contrast to targeting only improved diversity gains at maximum rate.

The chapter is organized in the following fashion. Section 6.1 introduces the system model. Section 6.2 presents our code design scheme that is devised based on the unitary and full-rank properties. Section 6.3 derives the maximum-likelihood metric that can be used by priority-first search decoding. Simulations are summarized and discussed in Section 6.4.

6.1 System Model

We consider an MIMO system with A_{T} transmit antennas and A_{R} receive antennas. The $N \times A_{\mathrm{R}}$ complex received matrix $\mathbb{Y} = [\boldsymbol{y}_1\ \boldsymbol{y}_2\ \ldots\ \boldsymbol{y}_{A_{\mathrm{R}}}]$ is then given by

$$\mathbb{Y} = \mathbb{B}\mathbb{H} + \mathbb{N},$$

where $\mathbb{B} = [\boldsymbol{b}_1\ \boldsymbol{b}_2\ \ldots\ \boldsymbol{b}_{A_\mathrm{T}}]$ is the $N \times A_\mathrm{T}$ transmitted code matrix, and $\mathbb{N} = [\boldsymbol{n}_1\ \boldsymbol{n}_2\ \ldots\ \boldsymbol{n}_{A_\mathrm{R}}]$ is an $N \times A_\mathrm{R}$ zero-mean complex Gaussian matrix with independent and identically distributed elements and covariance matrix

$$E[\boldsymbol{n}_i \boldsymbol{n}_i^H] = \sigma^2 \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}_{N \times N}.$$

Also, $\boldsymbol{b}_i = [b_{1,i}\ b_{2,i}\ \ldots\ b_{N,i}]^T$ is the bipolar codeword transmitted by antenna i with each $b_{i,n} \in \{\pm 1/\sqrt{A_\mathrm{T}}\}$. Likewise, $\boldsymbol{y}_j = [y_{1,j}\ y_{2,j}\ \ldots\ y_{N,j}]^T$ is the received vector at the jth receive antenna.

Because $\mathbb{H}$ is assumed an unknown constant matrix, the Gaussian assumption on the additive noise matrix $\mathbb{N}$ immediately gives that the maximum-likelihood (ML) decision about the transmitted codeword should be made based on the generalized likelihood ratio test (GLRT) as

$$\begin{aligned} \hat{\mathbb{B}} &= \arg\min_{\mathbb{B}} \min_{\mathbb{H}} \|\mathbb{Y} - \mathbb{B}\mathbb{H}\|^2 \\ &= \arg\min_{\mathbb{B}} \|\mathbb{Y} - \mathbb{B}\hat{\mathbb{H}}\|^2 \\ &= \arg\min_{\mathbb{B}} \|(\mathbb{I}_N - \mathbb{P}_B)\mathbb{Y}\|^2, \end{aligned} \tag{6.1}$$

where $\hat{\mathbb{H}} \triangleq (\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T\mathbb{Y}$ is the least-square estimate of $\mathbb{H}$ with respect to codeword $\mathbb{B}$ and received matrix $\mathbb{Y}$, and

$$\mathbb{P}_B \triangleq \mathbb{B}(\mathbb{B}^T\mathbb{B})^{-1}\mathbb{B}^T$$

is a function of the codeword $\mathbb{B}$. Here, $\mathbb{I}_N$ denotes an $N \times N$ identity matrix.

6.2 Code Design

6.2.1 Criteria for Good Codes

Several criteria for good codes have been proposed in the literature [1, 12, 13, 36]. We will in particular center on two of them: unitary and pairwise full-rank.

Firstly, it has been derived in [36] that unitary codewords, i.e., $\mathbb{B}^T\mathbb{B} = (N/A_\mathrm{T}) \cdot \mathbb{I}_{A_\mathrm{T}}$, can maximize the average signal-to-noise ratio (SNR) regardless of the statistics on $\mathbb{H}$. It has also been shown that when $\mathbb{H}$ is zero-mean complex Gaussian distributed, a unitary signal maximizes the capacity [24] and minimizes the union bound of word error rate (WER) [3] at high SNR. These results suggest that a good code can perhaps be constructed by collecting unitary codewords.

Secondly, it is better to have full-rank codeword pairs, where a pair of codewords, $\mathbb{B}(i)$ and $\mathbb{B}(j)$, is said to be *pair-wisely full-rank* if

$$\mathrm{rank}\left(\begin{bmatrix}\mathbb{B}(i) & \mathbb{B}(j)\end{bmatrix}\right) = 2A_\mathrm{T},$$

subject to $N \geq 2A_\mathrm{T}$. This is because at fairly high SNR, the average error probability is well approximated by the sum of pair-wise word error rates, namely, the union bound [1]. Also at fairly high SNR, the pair-wise word error is in turn well approximated by

$$\Pr\left(\hat{\mathbb{B}} = \mathbb{B}(j) \middle| \mathbb{B}(i) \text{ transmitted}\right) \approx \mathcal{Q}\left(\frac{1}{\sqrt{2}}\|\mathbb{H}\|\sqrt{\lambda_{\min}(\mathbb{L}_{ij})}\right)$$

where

$$\mathbb{L}_{ij} \triangleq \mathbb{I}_{A_\mathrm{R}} \otimes \left(\mathbb{B}(i)^T\left(\mathbb{I}_N - \mathbb{P}_{B(j)}\right)\mathbb{B}(i)\right),$$

and $\lambda_{\min}(\mathbb{L}_{ij})$ is the smallest eigenvalue of $\mathbb{L}_{ij}$. Here, "$\otimes$" indicates the Kronecker product, and $\mathcal{Q}(x) \triangleq \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-t^2/2}dt$ is the area under the tail of a standard Gaussian probability density function. Hence, if $[\mathbb{B}(i)\ \mathbb{B}(j)]$ do not achieve full column rank, we can obtain by [13] that

$$\det\left|\mathbb{B}(i)^T\left(\mathbb{I}_N - \mathbb{P}_{B(j)}\right)\mathbb{B}(i)\right| = 0.$$

This subsequently implies that $\lambda_{\min}(\mathbb{L}_{ij}) = 0$, and (6.2) will be close to 1/2 at fairly high SNR, which is a situation that a good code should avoid.

Therefore, a code that satisfies both the above criteria should guarantee a good pairwise-error-based union bound (which in turn hints to have a good performance). This viewpoint will be confirmed by the subsequent simulations.

6.2.2 The Proposed Code Design

Denote the information sequence by $\boldsymbol{k} = [k_1\ k_2\ \ldots\ k_K]^T$, where $k_i \in \{\pm 1\}$. The corresponding codeword is then proposed to be

$$\mathbb{B} = \frac{1}{\sqrt{A_{\mathrm{T}}}} \begin{bmatrix} \boldsymbol{k} & \boldsymbol{k} \odot \boldsymbol{s} \\ -\boldsymbol{k} \odot \boldsymbol{s} & \boldsymbol{k} \end{bmatrix}$$

where "$\odot$" denotes the Hadamard product, and

$$\boldsymbol{s} = \begin{cases} \begin{bmatrix} \mathbf{1}_{K-\lceil K/2 \rceil} \\ -\mathbf{1}_{\lceil K/2 \rceil} \end{bmatrix}, & \text{if } k_1 = -1 \\ \begin{bmatrix} \mathbf{1}_{K-\lceil K/2 \rceil} \\ -\mathbf{1}_{\lceil K/2 \rceil} \end{bmatrix} \odot \boldsymbol{d}, & \text{otherwise.} \end{cases}$$

In the above equation, $\mathbf{1}_k$ represents a $k \times 1$ all-one vector, and $\boldsymbol{d} \triangleq [(-1)^0\ (-1)^1\ \ldots\ (-1)^{K-1}]^T$.

It can be easily examined that the unitary criterion is satisfied, i.e., $\mathbb{B}^T \mathbb{B} = (N/A_{\mathrm{T}}) \cdot \mathbb{I}_{A_{\mathrm{T}}}$. It remains to show that the code just introduced satisfies pair-wise full-rank criterion.

Let $\mathbb{A}_{i,j} \triangleq \mathbb{B}(i)^T \mathbb{B}(j)$. Then for the validity of the pair-wise full-rank criterion, it suffices to prove that

$$\det \left| \mathbb{I}_{A_{\mathrm{T}}} - \frac{1}{(N/A_{\mathrm{T}})^2} \mathbb{A}_{i,j} \mathbb{A}_{i,j}^T \right| \neq 0, \tag{6.2}$$

for $1 \leq i, j \leq 2^K$ with $i \neq j$. By denoting respectively the kth eigenvalue and kth eigenvector of $\mathbb{A}_{i,j}\mathbb{A}_{i,j}^T$ by λ_k and $\boldsymbol{u}_k$, the validity of (6.2) can be verified by showing that $\lambda_k \neq (N/A_{\mathrm{T}})^2$ for every k because

$$\begin{aligned} (6.2) \quad &\Leftrightarrow \quad \det \left| \mathbb{I}_{A_{\mathrm{T}}} - \frac{1}{(N/A_{\mathrm{T}})^2} \sum_{k=1}^{A_{\mathrm{T}}} \lambda_k \boldsymbol{u}_k \boldsymbol{u}_k^T \right| \neq 0 \\ &\Leftrightarrow \quad \det \left| \sum_{k=1}^{A_{\mathrm{T}}} \left(1 - \frac{\lambda_k}{(N/A_{\mathrm{T}})^2} \right) \boldsymbol{u}_k \boldsymbol{u}_k^T \right| \neq 0. \end{aligned}$$

Now, let $(\boldsymbol{k}_i, \boldsymbol{s}_i)$ and $(\boldsymbol{k}_j, \boldsymbol{s}_j)$ be respectively the vector pairs that define codewords $\mathbb{B}(i)$ and $\mathbb{B}(j)$. Denote for convenience $\boldsymbol{c}_{j,i} = \boldsymbol{k}_j \odot \boldsymbol{s}_i$. We then prove that $\lambda_1 \neq (N/A_{\mathrm{T}})^2$ and $\lambda_2 \neq (N/A_{\mathrm{T}})^2$ by differentiating the following two cases.

Case 1: $\boldsymbol{s}_i = \boldsymbol{s}_j = \boldsymbol{s}$.

In this case,

$$\begin{aligned}\mathbb{B}(i)^T\mathbb{B}(j) &= \frac{1}{A_\mathrm{T}}\begin{bmatrix}\boldsymbol{k}_i^T\boldsymbol{k}_j + \boldsymbol{c}_{i,i}^T\boldsymbol{c}_{j,i} & \boldsymbol{k}_i^T\boldsymbol{c}_{j,i} - \boldsymbol{c}_{i,i}^T\boldsymbol{k}_j \\ \boldsymbol{c}_{i,i}^T\boldsymbol{k}_j - \boldsymbol{k}_i^T\boldsymbol{c}_{j,i} & \boldsymbol{k}_i^T\boldsymbol{k}_j + \boldsymbol{c}_{i,i}\boldsymbol{c}_{j,i}\end{bmatrix} \\ &= \begin{bmatrix}\boldsymbol{k}_i^T\boldsymbol{k}_j & 0 \\ 0 & \boldsymbol{k}_i^T\boldsymbol{k}_j\end{bmatrix}\end{aligned}$$

Then,

$$\mathbb{A}_{i,j}\mathbb{A}_{i,j}^T = \begin{bmatrix}(\boldsymbol{k}_i^T\boldsymbol{k}_j)^2 & 0 \\ 0 & (\boldsymbol{k}_i^T\boldsymbol{k}_j)^2\end{bmatrix}$$

So, $\lambda_1 = \lambda_2 = (\boldsymbol{k}_i^T\boldsymbol{k}_j)^2 < (N/A_\mathrm{T})^2$.

Case 2: $\boldsymbol{s}_i \neq \boldsymbol{s}_j$.

In this case,

$$\begin{aligned}\mathbb{B}(i)^T\mathbb{B}(j) &= \frac{1}{A_\mathrm{T}}\begin{bmatrix}\boldsymbol{k}_i^T\boldsymbol{k}_j + \boldsymbol{c}_{i,i}^T\boldsymbol{c}_{j,j} & \boldsymbol{k}_i^T\boldsymbol{c}_{j,j} - \boldsymbol{c}_{i,i}^T\boldsymbol{k}_j \\ \boldsymbol{c}_{i,i}^T\boldsymbol{k}_j - \boldsymbol{k}_i^T\boldsymbol{c}_{j,j} & \boldsymbol{k}_i^T\boldsymbol{k}_j + \boldsymbol{c}_{i,i}^T\boldsymbol{c}_{j,j}\end{bmatrix} \\ &= \frac{1}{A_\mathrm{T}}\begin{bmatrix}\boldsymbol{k}_i^T(\boldsymbol{k}_j + \boldsymbol{k}_j \odot \boldsymbol{d}) & \boldsymbol{k}_i^T(\boldsymbol{c}_{j,j} - \boldsymbol{c}_{j,i}) \\ -\boldsymbol{k}_i^T(\boldsymbol{c}_{j,j} - \boldsymbol{c}_{j,i}) & \boldsymbol{k}_i^T(\boldsymbol{k}_j + \boldsymbol{k}_j \odot \boldsymbol{d})\end{bmatrix},\end{aligned}$$

which gives

$$\mathbb{A}_{i,j}\mathbb{A}_{i,j}^T = \begin{bmatrix}c & 0 \\ 0 & c\end{bmatrix},$$

where $c \triangleq (1/A_\mathrm{T})^2(|\boldsymbol{k}_i^T(\boldsymbol{k}_j + \boldsymbol{k}_j \odot \boldsymbol{d})|^2 + |\boldsymbol{k}_i^T(\boldsymbol{c}_{j,j} - \boldsymbol{c}_{j,i})|^2)$. Accordingly, $\lambda_1 = \lambda_2 = c < (N/A_\mathrm{T})^2$.

We end this section by commenting that our design can be viewed as a high-dimensional variation of Alamouti codes. Hence, the unitary property is satisfied simply by the Alamouti code structure. By properly introducing the additional Hadamard product, our code can further fulfill the pairwise full-rank property.

6.3 Priority-First Search Decoding

In this section, we will derive the recursive decoding metric that can be used by the priority-first search algorithm [15]. Since the metric proposed is nondecreasing along every path in the code tree, the optimality of the decoding result is certified [36].

Continuing the derivation in (6.1) by noting that $\|\mathbb{P}_B\|^2 = A_{\mathrm{T}}$, we obtain

$$\begin{aligned}
\hat{\mathbb{B}} &= \arg\min_{\mathbb{B}} - \sum_{j=1}^{A_{\mathrm{R}}} \boldsymbol{y}_j^H \mathbb{P}_B \boldsymbol{y}_j \\
&= \arg\min_{\mathbb{B}} -\mathrm{tr}\left(\mathbb{P}_B \sum_{j=1}^{A_{\mathrm{R}}} \boldsymbol{y}_j \boldsymbol{y}_j^H\right) \\
&= \arg\min_{\mathbb{B}} -\mathrm{tr}\left(\frac{A_{\mathrm{T}}}{N}\mathbb{B}\mathbb{B}^T \sum_{j=1}^{A_{\mathrm{R}}} \boldsymbol{y}_j \boldsymbol{y}_j^H\right) \\
&= \arg\min_{\mathbb{B}} -\mathrm{tr}\left(\mathbb{W} \sum_{i=1}^{A_{\mathrm{T}}} \boldsymbol{b}_i \boldsymbol{b}_i^T\right),
\end{aligned}$$

where $\mathbb{W} \triangleq \mathbf{Re}\{(A_{\mathrm{T}}/N)\sum_{j=1}^{A_{\mathrm{R}}} \boldsymbol{y}_j \boldsymbol{y}_j^H\}$, and $\mathrm{tr}(\cdot)$ is the trace matrix operation. By letting

$$\mathbb{M}_1 \triangleq \boldsymbol{k}\boldsymbol{k}^T + (\boldsymbol{k}\odot\boldsymbol{s})(\boldsymbol{k}\odot\boldsymbol{s})^T,$$

and

$$\mathbb{M}_2 \triangleq \boldsymbol{k}(\boldsymbol{k}\odot\boldsymbol{s})^T - (\boldsymbol{k}\odot\boldsymbol{s})\boldsymbol{k}^T,$$

we have

$$\sum_{i=1}^{A_{\mathrm{T}}} \boldsymbol{b}_i \boldsymbol{b}_i^T = \frac{1}{A_{\mathrm{T}}}\begin{bmatrix}\mathbb{M}_1 & \mathbb{M}_2^T \\ \mathbb{M}_2 & \mathbb{M}_1\end{bmatrix}.$$

This reduces the decoding criterion to

$$\hat{\boldsymbol{k}} = \arg\min_{\boldsymbol{k}} \left\{-\mathrm{tr}\left(\mathbb{M}_1\mathbb{D}\right) - \mathrm{tr}\left(\mathbb{M}_2\mathbb{E}\right)\right\},$$

where $\mathbb{D} \triangleq \mathbb{W}_{1,1} + \mathbb{W}_{2,2}$, $\mathbb{E} \triangleq \mathbb{W}_{1,2} - \mathbb{W}_{1,2}^H$, and $\mathbb{W}_{1,1}$, $\mathbb{W}_{1,2}$ and $\mathbb{W}_{2,2}$ are the corresponding submatrices of

$$\mathbb{W} = \begin{bmatrix}\mathbb{W}_{1,1} & \mathbb{W}_{1,2} \\ \mathbb{W}_{1,2}^H & \mathbb{W}_{2,2}\end{bmatrix}.$$

Since the decision criterion is intact by adding a constant independent of the codewords,

$$\hat{\boldsymbol{k}} = \arg\min_{\boldsymbol{k}} \left(\sum_{m=1}^{K} C_m - \frac{1}{4}\sum_{m=1}^{K}\sum_{n=1}^{K} k_m k_n (1 + s_m s_n) d_{m,n} - \frac{1}{4}\sum_{m=1}^{K}\sum_{n=1}^{K} k_m k_n (s_m - s_n) e_{m,n}\right), \tag{6.3}$$

where

$$C_m \triangleq \sum_{n=1}^{m-1}(|d_{m,n}| + |e_{m,n}|) + \frac{1}{2}|d_{m,m}|,$$

and $d_{m,n}$ and $e_{m,n}$ are respectively the elements in matrices $\mathbb{D}$ and $\mathbb{E}$ and can be expressed as

$$d_{m,n} = \begin{cases} \frac{A_\mathrm{T}}{N}\sum_{j=1}^{A_\mathrm{R}} \mathbf{Re}\{y_{m,j}y_{n,j}^* + y_{m+K,j}y_{n+K,j}^*\}, & \\ & \text{for } 1 \le m,n \le N \\ 0, & \text{otherwise} \end{cases}$$

$$e_{m,n} = \begin{cases} \frac{A_\mathrm{T}}{N}\sum_{j=1}^{A_\mathrm{R}} \mathbf{Re}\{y_{m,j}y_{n+K,j}^* - y_{m+K,j}y_{n,j}^*\}, & \\ & \text{for } 1 \le m,n \le N \\ 0, & \text{otherwise} \end{cases}$$

Finally, the decoding metric f inside the parenthesis of (6.3) can be computed recursively as

$$f(\boldsymbol{k}_{(\ell)}) = g(\boldsymbol{k}_{(\ell)}) - \gamma(\boldsymbol{k}_{(\ell)}),$$

where $\boldsymbol{k}_{(\ell)} = [k_1\ k_2\ \ldots\ k_\ell]^T$,

$$g(\boldsymbol{k}_{(\ell+1)}) = g(\boldsymbol{k}_{(\ell)}) - \beta(\boldsymbol{k}_{(\ell+1)}),$$

$$\beta(\boldsymbol{k}_{(\ell+1)}) = k_{\ell+1}\frac{\sqrt{A_\mathrm{T}}}{N}\sum_{r=1}^{A_\mathrm{R}}\mathbf{Re}\left\{\sum_{t=0}^{1} y_{\ell+1+tK,r}\sum_{i=0}^{1}\sum_{j=0}^{1}(-1)^p s_{\ell+1}^q \cdot u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell)})\right\},$$

$$\begin{aligned}\gamma(\boldsymbol{k}_{(\ell)}) &= -\sum_{m=\ell+1}^{K}\sum_{n=\ell+1}^{m}\big(|d_{m,n}|\,\mathbf{1}\{s_m = s_n\} + |e_{m,n}|\,\mathbf{1}\{s_m \neq s_n\}\big) \\ &\quad + \frac{\sqrt{A_\mathrm{T}}}{N}\sum_{m=\ell+1}^{K}\left|\sum_{r=1}^{A_\mathrm{R}}\mathbf{Re}\left\{\sum_{t=0}^{1} y_{m+tK,r}\sum_{i=0}^{1}\sum_{j=0}^{1}(-1)^p s_m^q \cdot u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell)})\right\}\right|,\end{aligned}$$

$p = \lfloor t + (-1)^t(i+j)/2 \rfloor$, $q = t + |i-j|(-1)^t$, and

$$u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell+1)}) = u_{i,j}^{(r)}(\boldsymbol{k}_{(\ell)}) + \frac{1}{\sqrt{A_\mathrm{T}}}k_{\ell+1}s_{\ell+1}^i y_{\ell+1+jK,r}^*.$$

In the above equation, $\mathbf{1}\{\cdot\}$ denotes the set indicator function.

6.4 Simulation Results

In this section, we compare the performance of the code constructed in Section 6.2 with the codes obtained by computer search. The criterion used in the simulated annealing code search algorithm follows that in [3] (also, [12] and [13]). We take $A_{\mathrm{R}} = 1$ in our simulations, and assume that $\mathbb{H}$ is zero-mean complex Gaussian with $E[\mathbb{H}\mathbb{H}^H] = (1/A_{\mathrm{T}})\mathbb{I}_{A_{\mathrm{T}}}$. The average SNR is then give by

$$\frac{E[\mathbb{H}^H\mathbb{H}]}{(K/N)\sigma^2} = \frac{2}{\sigma^2}.$$

Figure 6.1 shows that the best (structureless) computer-searched codes only have about 0.4 dB advantage over the constructed codes for $N = 4, 6, \ldots, 12$.

We also compare our code with a multiple-antenna system that uses the $(17, 12, 3)$ nonlinear channel code[2] in combination with the Alamouti code and a 7-bit training sequence. In particular, the code bits are mapped to the two transmit antennas using the Alamouti code before its transmission, and the receiver will estimate $\mathbb{H}$ in terms of a least square estimator based on the 7 training bits. The result in Figure 6.2 illustrates that this communication system performs 0.7 dB worse than the constructed code. In a technically infeasible situation that assumes the receiver can achieve a perfect estimate of $\mathbb{H}$ with merely 7 training bits, the typical communication system outperforms the constructed code by only 0.5 dB.

We would like to emphasize that to search the best code by computers for codeword length greater than 14 is very operational intensive even if there are only two transmit antennas. For example, it took about three weeks to cool down the simulated-annealing search when $N = 14$ and $A_{\mathrm{T}} = 2$. It can be anticipated that the search time will grow exponentially with the code word length. Thus, the systematic code construction that we propose may be a good alternative as far as long code is concerned.

Figure 6.3 shows the decoding complexity of the priority-first search decoder for con-

[2]The $(17, 12, 3)$ code we adopt here is formed by taking out some code words from the $(17, 12 \cdot \log_2(20/16), 3)$ code in [23].

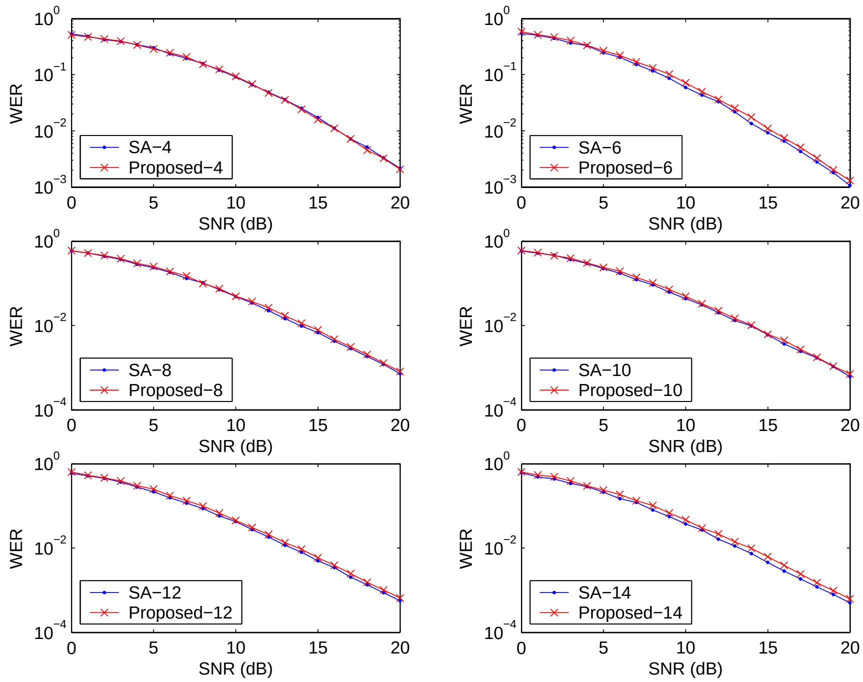

Figure 6.1: Comparison of word error rates (WERs) between the codes constructed in Section 6.2.2 (Proposed-N) and the codes obtained from simulated annealing search (SA-N). The codeword lengths are taken to be equal to $N = 4, 6, 8, 10, 12$ and 14.

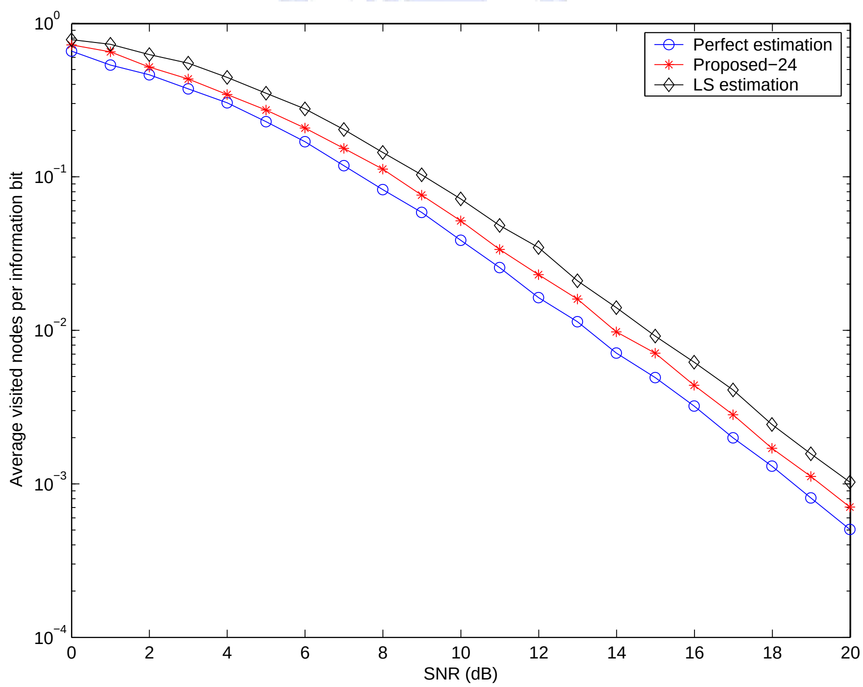

Figure 6.2: Comparison of WERs among the codes constructed in Section 6.2.2 (Proposed-24) and the system using a (17,12) nonlinear code in combination with the Alamouti code and a 7-bit training sequence. The codeword length is equal to $N = 24$.

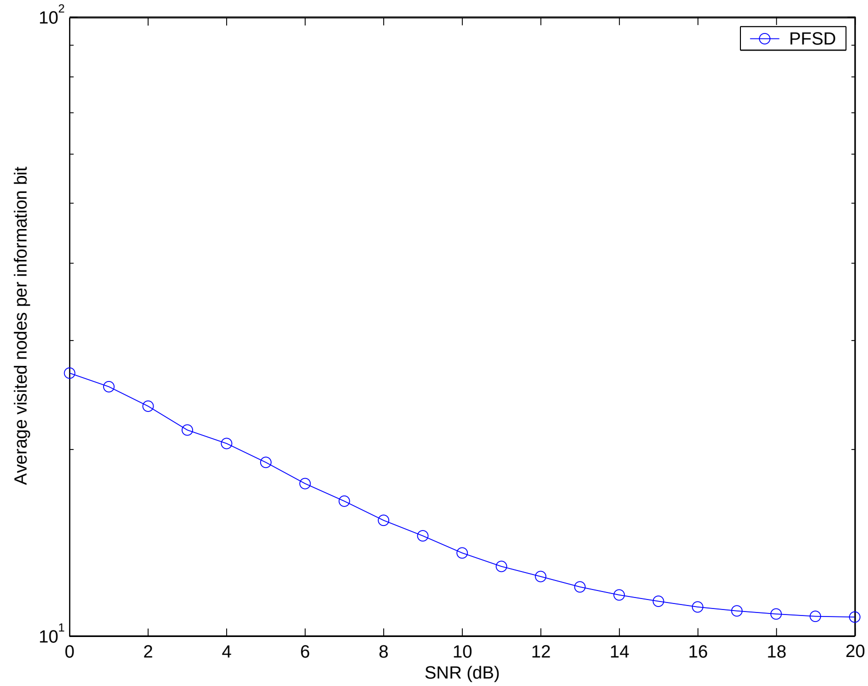

Figure 6.3: The decoding complexity of the priority-first search decoder (PFSD) for the constructed code of length $N = 24$.

structed code of length 24. The complexity is defined as the average number of node expansions per information bit. Since the number of node expansions is half of the number of tree branch metrics computed (i.e., two recursions of f-function values), the equivalent complexity of exhaustive decoding is correspondingly $(2^{K+1} - 1) \cdot A_{\mathrm{T}}/K$. In the case of $(24, 12)$ code with two transmit antennas, this number is equal to 1365.17. It is then clear from the figure that the priority-first search decoder significantly improve the decoding complexity when it is compared with the exhaustive decoder.

6.5 Summary

Conclusions of previous researches are however in general based on unstructured solutions obtained using computer search. The coding gain of these joint designs is therefore limited by both the computer-searchable "short" code length and the compromise between "suboptimal" performance and "high" complexity of their optimal decoding. Through our code design, we can solve the above problems partially. For codes of short block length, our simulations illustrate that the codes we propose have comparable performance to the best computer-searched codes. For codes of long block lengths that are almost beyond the searchable range of existing computer systems, our codes are still better than some reference designs based on separate channel estimation and error correction components.

Chapter 7

Conclusion and Future Works

In this dissertation, we propose several systematical code designs respectively for frequency-selective block fading channels, frequency-selective varying fading channels, frequency-non-selective varying fading channels, as well as space-time frequency-selective block fading channels, in which transmitter and receiver has no knowledge on CSI. When our codes are compared with the code designed by exhaustive computer searches in the literature, very limited performance difference can be observed.

Unlike the structureless computer-searched codes, the structure of the systematical code design we propose allows for maximum-likelihood priority-first search decoding, and hence, avoid the necessity of exhaustive decoding. This feature makes possible the decoding of longer codes of which the operational-intensive exhaustive decoding becomes infeasible in modern computer technology.

By simulations, the performances of our systematical designed codes are shown comparable to the best codes found by computer search. Also, the decoding complexity of the proposed maximum-likelihood priority-first search decoding is significantly smaller when it is compared with the exhaustive checking decoding.

We should point out at the end that we have thus far ignored an implicit problem for codes that absorb the training sequence into the error-correcting codewords: in traditional packet-switched systems, frame synchronization is often achieved by the same training sequence. Without synchronization of the codeword margins, decoding may become technically infeasible. Nevertheless, there are recent standards starting to consider to partly separate the tasks of frame synchronization and channel estimation. For example, in IEEE 802.16e, initial frame synchronization is performed by means of a preamble, and is later shared by all users. Pilots are then added amid user data for individual channel estimation during data transmission [41]. It is then fair to say that at this stage, the joint channel estimation and error correction codes may only fit well in an initial-sync, or circuit-switched, or TDD-based system environment. It will be an interesting, but quite challenging, future task to further enhance the proposed codes to possess self-synchronization capability.

Extensions of our coding system design can be done in several directions. As an example, how to systematically construct a "better" nonlinear code (as contrary to the traditional linear codes) for channels with known fading statistics should be an extension of interest. Another extension is to add a stop criterion in decoding such that the decoding complexity can be saved when deep fading occurs. The introduction of higher order modulation such as 16QAM and 64QAM in the code design is also a good future extension that is of practical use.

Bibliography

[1] M. Beko, J. Xavier and V. A. N. Barroso, "Noncoherent communication in multiple-antenna system: receiver design and codebook construction," *IEEE Trans. Signal Process.*, vol. 55, no. 12, pp. 5703-5715, Dec. 2007.

[2] M. J. Borran, A. Sabharwal and B. Aazhang, "On design criteria and construction of noncoherent space-time constellations," *IEEE Trans. Inform. Theory*, vol. 49, no. 10, pp. 2332-2351, Oct. 2003.

[3] M. Brehler and M. K. Varanasi, "Asymptotic error probability analysis of quadratic receivers in Rayleigh fading channels with applications to a unified analysis of coherent and noncoherent space-time receivers," *IEEE Trans. Inform. Theory*, vol. 47, no. 6, pp. 2383-2399, Sept. 2001.

[4] K. M. Chugg and A. Polydoros, "MLSE for an unknown channel-part I: Optimality considerations," *IEEE Trans. Commun.*, vol. 44, no. 7, pp. 836-846, July 1996.

[5] A.M. Cipriano, J.-C. Belfiore, "noncoherent Codes over the Grassmannian", *IEEE Trans Wireless Commun.*, vol. 6, no. 10, pp 3657-3667, Oct. 2007.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd edition, MIT Press, 2001.

[7] O. Coskun and K. M. Chugg, "Combined coding and training for unknown ISI channels," *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1310-1322, Aug. 2005.

[8] S. N. Crozier and D. D. Falconer and S. A. Mahmound, "Least sum of squared errors (LSSE) channel estimation," *Proc. Inst. Elect. Eng.*, vol. 138, pt. F, pp. 371-378, Auguest 1991.

[9] P. Dayal, M. Brehler and M. K. Varanasi, "Leveraging coherent space-time codes for noncoherent communication via training", *IEEE Trans. Inform. Theory*, vol. 50, no. 9, pp. 2058-2080, Sept. 2004.

[10] H. El Gamal, D. Aktas, M. O. Damen, "Noncoherent space-time coding: An algebraic perspective", *IEEE Trans. Inform. Theory*, vol. 51, no. 7, pp. 2380-2390, July 2005.

[11] J. Giese, "Constellation design under channel uncertainty", *doctoral thesis*, Royal Institute of Technology, 2005.

[12] J. Giese and M. Skoglund, "Space-time code design for unknown frequency-selective channels," *Proc. IEEE. Int. Conf. Acoust., Speech, Signal Process.*, Orlando, FL, USA, May 2002.

[13] J. Giese and M. Skoglund, "Single- and multiple-antenna constellations for communication over unknown frequency-selective fading channels," *IEEE Trans. Inform. Theory*, vol. 53, no. 4, pp. 1584-1594, April 2007.

[14] G. Ganesan and P. Stoica, "Space-time block codes: A maximum SNR approach," *IEEE Trans. Inform. Theory*, vol. 47, no. 4, pp. 1650-1659, May 2001.

[15] Y. S. Han and P.-N. Chen, "Sequential decoding of convolutional codes," *The Wiley Encyclopedia of Telecommunications*, edited J. Proakis, John Wiley and Sons, Inc., 2002.

[16] B. Hochwald, T. L. Marzetta, T. J. Richardson, W. Sweldens, and R. Urbanke, "Systematic design of unitary space-time constellations," *IEEE Trans. Inform. Theory*, vol. 46, no. 6, pp. 1962-1973, Sept. 2000.

[17] Y. S. Han, C. R. P. Hartmann and C.-C. Chen, "Efficient priority-first search maximum-likelihood soft-decision decoding of linear block codes," *IEEE Trans. Inform. Theory*, vol. 39, no. 5, pp. 1514–1523, Sep. 1993.

[18] D. Harville, *Matrix Algebra From a Statistician's Perspective*, 1st edition, Springer, 2000.

[19] R. Knopp and P. A. Humblet "On coding for block fading channels," *IEEE Trans. Inform. Theory* , vol. 46, no. 1, pp. 189-205, Jan. 2000.

[20] S. Lin and D. J. Costello, Jr., *Error Control Coding*, 2nd edition, Prentice-Hall, Inc., 2004.

[21] W.-K. Ma, B.-N. Vo, T. N. Davidson and P.-C. Ching, "Blind ML detection of orthogonal space-time block codes: efficient high-performance implementations," *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 738-751, Feb. 2006.

[22] W.-K. Ma, "Blind ML detection of orthogonal space-time block codes: Identifiability and code construction," *IEEE Trans. Signal. Process.*, vol. 55, no. 7, pp. 3312-3324, July 2007.

[23] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes Amsterdam*, The Netherlands: North-Holland, 1977.

[24] T. L. Marzetta and B. M. Hochwald, "Capacity of a mobile multiple-antenna communication link in Rayleigh flat-fading," *IEEE Trans. Inform. Theory*, vol. 45, no. 1, pp. 139-157, Jan. 1999.

[25] J. C. L. Ng, K. B. Letaief and R. D. Murch, "Complex optimal sequences with constant magnitude for fast channel estimation initialization," *IEEE Trans. commun.*, vol. 46, no. 3, pp. 305-308, March 1998.

[26] S. I. Park, S. R. Park, I. Song and N. Suehiro, "Multiple-access interference reduction for QS-CDMA systems with a novel class of polyphase sequence," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1448-1458, July 2000.

[27] J. Pearl, *Heuristics: Intelligent Search Strategies for Computer Problem Solving*, Addison-Wesley, 1984.

[28] J. Proakis, *Digital Communications*, McGraw-Hill, fourth edition, 2001.

[29] N. Seshadri, "Joint data and channel estimation using blind trellis search techniques," *IEEE Trans. Commun.* , vol. 42, no. 2/3/4, pp. 1000-1011, February/March/April 1994.

[30] M. Skoglund, J. Giese and S. Parkvall, "Code design for combined channel estimation and error protection," *IEEE Trans. Inform. Theory*, vol. 48, no. 5, pp. 1162-1171, May 2002.

[31] V. Tarokh and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inform. Theory*, vol. 45, no. 5, pp. 1456-1467, July 2007.

[32] C. Tellambura, Y. J. Guo, and S. K. Barton, "Channel estimation using aperiodic binary sequences," *IEEE Commun. Letters*, vol. 2, no. 5, pp. 140-142, May 1998.

[33] C. Tepedelenlioğlu, A. Abdi and G. B. Giannakis, "The Ricean K factor: estimation and performance analysis," *IEEE Trans. Wireless Commun.*, vol. 2, no. 4, pp. 799-810, July 2003.

[34] S.-H. Wu, "Effects of estimation errors on the noncoherent capacity over time-varying channels," *Proc. IEEE Vehicular Technology Conference*, Montreal, Canada, Sep. 2006.

[35] C.-L. Wu, P.-N. Chen, Yunghsiang S. Han and S.-W. Wang, "A self-orthogonal code and its maximum-likelihood decoder for combined channel estimation and error protection," *Proc. International Symposium on Information Theory and its Applications*, Auckland, New Zealand, Dec. 2008.

[36] C.-L. Wu, P.-N. Chen, Yunghsiang S. Han and M.-H. Kuo, "Maximum-likelihood priority-first search decodable codes for combined channel estimation and error correction," accepted by *IEEE Trans. Inform. Theory.*

[37] C.-L. Wu, P.-N. Chen, Yunghsiang S. Han and Y.-X. Zheng, "On the Coding Scheme for Joint Channel Estimation and Error Correction over Block Fading Channels," *Proc. Personal, Indoor and Mobile Radio Commun. Symposium*, Tokyo, Japan, Sept. 13-19, 2009.

[38] C.-L. Wu, M. Skoglund, P.-N. Chen and Yunghsiang S., "A Systematic Space-Time Code Design and Its Maximum-Likelihood Decoding for Combined Channel Estimation and Error Correction," *Proc. IEEE Int'l Symposium on Inform. Theory*, Seoul, Korea, June 28-July 3, 2009.

[39] C. Xu, H. Sun, J.-K. Fwu, Q. Li, Y. Zhu, J. Wang, S. Zhang, Y. Gao, H. Yin, R. Vannithamby and S. Ahmadi, *IEEE 802.16m UL PHY Control: CQI Feedback Channel Design*, C80216m-UL_PHY_Ctrl-08_065r1, Intel Corporation, IEEE 802.16 UL PHY Control RG, 11th, Nov. 2008.

[40] S.-A. Yang and J. Wu, "Optimal binary training sequence design for multiple-antenna systems over dispersive fading channels," *IEEE Trans. Vehicular Technology*, vol. 51, no. 5, pp. 1271-1276, September 2002.

[41] *Draft Standard for Local and Metropolitan Area Networks, Part 16: Air Interface for Broadcast Wireless Access Systems*, P802.16Rev2/D6, July 2008.

博士候選人資料

姓　名 ： 吳佳龍

性　別 ： 男

出生年月日 ： 民國 65 年 10 月 30 日

籍　貫 ： 台灣台南

學　歷 ： 中原大學 資訊工程系
交通大學 電信工程研究所 碩士
交通大學 電信工程研究所 博士候選人

經　歷 ：

論文題目 ：

(中文) 結合錯誤更正與通道估計的系統化編碼設計及其最大概度解碼

(英文) Systematic code design for combined error correction and channel estimation and its maximum-likelihood decoding