UNIVERSITY OF CALIFORNIA
Los Angeles

Efficient Reliable Communication in the Short Blocklength Regime Through List Decoding and Through Feedback

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Electrical and Computer Engineering

by

Hengjie Yang

2022

Abstract of the Dissertation

Efficient Reliable Communication in the Short Blocklength Regime

Through List Decoding and Through Feedback

by

Hengjie Yang
Doctor of Philosophy in Electrical and Computer Engineering
University of California, Los Angeles, 2022
Professor Richard Wesel, Chair

This dissertation consists of three parts investigating the efficient reliable communication in the short blocklength regime for classical channels in three different settings: (i) no feedback, (ii) full, noiseless feedback, and (iii) finite, stop feedback.

The first part focuses on the non-feedback binary-input additive white Gaussian noise (AWGN) channel. A long-standing research problem is to design good linear block codes for this channel. As its primary contribution, we propose the cyclic-redundancy-check-aided (CRC-aided) convolutional code under serial list Viterbi decoding (SLVD). To design a good CRC-aided convolutional code, we propose the distance-spectrum optimal (DSO) CRC polynomial and provide an efficient search algorithm for a given convolutional code. We then analyze the performance and complexity of the SLVD for the CRC-aided convolutional code. For transmitting 64 information bits, simulation shows that some CRC-aided convolutional codes beat the random-coding union (RCU) bound at short blocklength.

The second part of the dissertation focuses on the binary asymmetric channel (BAC) with full, noiseless feedback, including the binary symmetric channel (BSC) as a special case. Building on the small-enough-difference (SED) coding scheme of Naghshvar *et al.* originally proposed for symmetric binary-input channels with feedback, we generalize the coding scheme to the class of BACs with feedback, and establish a non-asymptotic achievability bound for the deterministic variable-length feedback (VLF) code constructed from the generalized SED coding scheme. In the specific case of the BSC, we present a refined non-asymptotic VLF achievability bound. Despite the extreme use of feedback, Naghshvar *et al.*'s results on the BSC with full feedback appear to be inferior to Polyanskiy's bound for codes with a limited use of feedback, known as the variable-length stop-feedback (VLSF) codes. In contrast, numerical evaluations show that our VLF achievability bounds outperform Polyanskiy's VLSF achievability bound for both BAC and BSC cases.

The third part of the dissertation focuses on the performance of VLSF codes with finite optimal decoding times for the BI-AWGN channel. We first develop tight approximations on the tail probability of length-n cumulative information density which will play an important role in numerical evaluations. Building on a recent result of Yavas *et al.* on VLSF codes with finite decoding times, the problem reduces to an integer program of minimizing the upper bound of average blocklength subject to the average error probability, minimum gap, and integer constraints. By allowing real-valued decoding times and using a two-step minimization, we derive the gap-constrained sequential differential optimization procedure to numerically evaluate the achievability bound. Numerical evaluations show that Polyanskiy's bound for VLSF codes, which assumes infinite decoding times, can be closely approached with a finite (and relatively small) number of decoding times.

The dissertation of Hengjie Yang is approved.

Dariush Divsalar

Alexander Sherstov

Christina Fragouli

Lara Dolecek

Richard Wesel, Committee Chair

University of California, Los Angeles

2022

To my parents

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Acknowledgments

This dissertation would not have been possible without the support from my advisor, family, and friends. First of all, I would like to thank my advisor, Prof. Richard Wesel, for his omnipresent guidance and support throughout the past five years. His enthusiasm led me to the world of channel coding and information theory and taught me to see and contemplate the beauty behind the wall of definitions and theorems. The research meeting has always been a great source of inspiration. I enjoyed our numerous stimulating discussions and Rick's remarkable insight in identifying the impact of a result. Rick's kindness and patience also influenced me by osmosis.

I would like to thank all faculty members of Electrical and Computer Engineering Department for the wonderful graduate courses; special thanks go to Professors Suhas Diggavi, Christina Fragouli, Lara Dolecek, Lieven Vandenberghe, and Richard Wesel as your classes built the foundation of my research. I am also indebted to the faculty of the Mathematics Department whose analysis class helped shape my academic rigor and thinking. Finally, I also would like to thank Deeona Columbia, Ryo Arreola, and all other staff members whose services I have benefited from while I was a graduate student. To Ryo, I am saddened by your untimely passing. You will always be missed.

In addition, I would like to thank Prof. Dariush Divsalar (JPL), Prof. Aaron B. Wagner (Cornell), Prof. Tara Javidi (UCSD), Dr. Peter Kazakov, Dr. Gianluigi Liva, and Dr. Fabian Steiner for the inspiring conversations and correspondences we have shared. I was fortunate to have collaborated with Hanwen Yao and Prof. Alexander Vardy from UCSD. However, I am saddened by Prof. Vardy's untimely passing while I am preparing this dissertation. Recently, I have had the great pleasure of working with Recep Can Yavas and Prof. Victoria Kostina from Caltech. I am deeply impressed by Recep's profound knowledge in probability theory and his rigor in both academic writing and research.

I would like to thank colleagues I have collaborated with at UCLA: Ethan Liang, Ming-

hao Pan, Linfang Wang, Sudarsan V. S. Ranganathan, Amaael Antonini, and many others. Especially, I am fortunate enough to have worked with two talented and motivated undergraduate students: Ethan Liang and Minghao Pan. Together, we have produced several papers. I wish them the best of luck in their future academic endeavor. To the junior doctoral students in the lab – Amaael Antonini and Semira Galijasevic – I wish you success over the course of your graduate career.

I was fortunate to have interned at Qualcomm during summer twice. I would like to thank all colleagues at the company who have taught me practical industry knowledge that is not covered in a graduate coursework. In particular, I would like to extend my gratitude to my internship mentors: Ming Kang, Taemin Kim, and Hobin Kim.

I cannot emphasize enough the importance of all the wonderful friends I was lucky to meet during these years, whose company has greatly enriched my life in Los Angeles: Weinan Song, Linfang Wang, Bijie Bai, Yun Liao, Minghao Pan, Jingbo Wang, Yannan Li, Tingyi Zhou, Siyou Pei, Weixin Liang, Dr. Lin Zhou, and many others. To all of you, I am thankful. Especially, I would like to thank Yun Liao for all the enlightening and stimulating discussions we have shared.

Finally, my deepest gratitude goes to my family: to my parents for their unconditional love and support, without which I could not have achieved any of my serious goals.

Vita

2017	B.S. in Telecommunications Engineering Xidian University, Xi'an, China
2017 – 2022	Graduate Student Researcher Electrical and Computer Engineering Department University of California, Los Angeles
2018	M.S. in Electrical and Computer Engineering University of California, Los Angeles
2020	Teaching Assistant Electrical and Computer Engineering Department University of California, Los Angeles
2020	Modem Systems Engineering Intern Qualcomm Technologies, Inc.
2021	Modem Systems Engineering Intern Qualcomm Technologies, Inc.
2021 – 2022	Dissertation Year Fellow University of California, Los Angeles

Selected Publications

H. Yang, S. V. S. Ranganathan and R. D. Wesel, "Serial List Viterbi Decoding with CRC: Managing Errors, Erasures, and Complexity", in *Proc. IEEE Global Commun. Conf. (GLOBECOM),* Abu Dhabi, UAE, December 2018.

H. Yang and R. D. Wesel, "On the Most Informative Boolean Functions of the Very Noisy Channel", in *Proc. IEEE Int. Symp. Inf. Theory (ISIT),* Paris, France, July 2019.

H. Yang, E. Liang, H. Yao, A. Vardy, D. Divsalar, and R. D. Wesel, "A List-Decoding Approach to Low-Complexity Soft Maximum-Likelihood Decoding of Cyclic Codes", in *Proc. IEEE Global Commun. Conf. (GLOBECOM),* Waikoloa, HI, USA, December 2019.

H. Yang and R. D. Wesel, "Finite-Blocklength Performance of Sequential Transmission over BSC with Noiseless Feedback", in *Proc. IEEE Int. Symp. Inf. Theory (ISIT),* Los Angeles, CA, USA, June 2020.

H. Yang, L. Wang, V. Lau and R. D. Wesel, "An Efficient Algorithm for Designing Optimal CRCs for Tail-Biting Convolutional Codes", in *Proc. IEEE Int. Symp. Inf. Theory (ISIT),* Los Angeles, CA, USA, June 2020.

H. Yang, E. Liang, M. Pan, and R. D. Wesel, "CRC-Aided List Decoding of Convolutional Codes in the Short Blocklength Regime", *IEEE Trans. Inf. Theory*, vol. 68, no. 6, pp. 3744 - 3766, June 2022.

H. Yang, M. Pan, A. Antonini, and R. D. Wesel, "Sequential Transmission Over Binary Asymmetric Channels With Feedback", to appear in *IEEE Trans. Inf. Theory*, May 2022.

H. Yang, R. C. Yavas, V. Kostina, and R. D. Wesel, "Variable-Length Stop-Feedback Codes With Finite Optimal Decoding Times for BI-AWGN Channels", to appear in *Proc. IEEE Int. Symp. Inf. Theory (ISIT),* Espoo, Finland, June 2022.

CHAPTER 1

Introduction

In recent decades, the ultra-reliable low-latency communication (URLLC) in 5G calls for powerful short blocklength codes with low probability of error. Depending on the availability and types of feedback, there are three major cases worthy of investigation: channels without feedback, channels with full noiseless feedback, and channels with finite, stop feedback.

In the first case, we focus on the non-feedback binary-input additive white Gaussian noise (BI-AWGN) channel. A long-standing research problem is to construct good block codes at short information length (e.g., a thousand or fewer information bits) for this channel. Thanks to the advances in finite-blocklength information theory developed by Polyanskiy, Poor, and Verdú, the probability of error of the best (n, M) fixed-length code of blocklength n and message size M is tightly upper bounded by the *random-coding union* (RCU) bound, and is tightly lower bounded by the *meta-converse* (MC) bound [PPV10]. These two bounds serve as the benchmark to assess the performance of a given short blocklength code for a broad class of channels, including the discrete memoryless channel (DMC) and the BI-AWGN channel. For the BI-AWGN channel, this dissertation proposes the *cyclic-redundancy-check (CRC)-aided convolutional code* as a good short blocklength code capable of achieving the RCU bound at low decoding complexity through *serial list Viterbi decoding (SLVD)*.

In the second case where full noiseless feedback is present, we turn our attention to the binary asymmetric channel (BAC) with feedback, including the binary symmetric channel (BSC) as a special case. Naghshvar *et al.* [NWJ12] proposed an intriguing coding scheme which we term as the *small-enough-difference* (SED) coding scheme. The deterministic variable-length feedback (VLF) code constructed with the SED coding scheme asymptotically attains both capacity and Burnashev's optimal error exponent for the symmetric binary-input channels with feedback. As the name suggests, although the capacity-achieving distribution is Bern(1/2), the SED coding scheme only seeks to partition the message set into two subsets such that the probability difference of the two subsets is small enough rather than strictly zero. The SED coding scheme allows Naghshvar *et al.* to develop a non-asymptotic VLF acheivability bound for the symmetric binary-input channels.

However, for the BSC, despite the extreme use of feedback, Naghshvar *et al.*'s non-asymptotic VLF achievability bound lies beneath Polyanskiy's achievability bound for codes with a limited use of feedback, known as the variable-length stop-feedback (VLSF) codes [PPV11]. This suggests that there is still significant room to improve previous results. In this dissertation, we extend the SED coding scheme to the BAC with feedback, and develop a non-asymptotic VLF achievability bound that asymptotically attains the capacity of the BAC and Burnashev's optimal error exponent. In the specific case of the BSC, we develop a refined non-asymptotic VLF achievability bound using a two-phase analysis. Numerical evaluations show that our VLF bounds outperform Polyanskiy's VLSF achievability bound in both the BAC and BSC cases, as desired.

In practice, however, the feedback is often an ACK/NACK signal that informs the encoder of whether to terminate transmission. This type of feedback is known as the stop feedback which only aims at terminating the transmission and does not affect the transmitted symbol. Very often, the decoding opportunities are also limited. In [PPV11], Polyanskiy *et al.* formally defined the VLSF code and showed that the VLSF code is sufficient to dramatically improve the maximal achievable rate, compared to the fixed-length code at the same blocklength and error probability. However, their bound assumes *infinite* decoding times. In contrast, this dissertation investigates the performance of VLSF codes with *finite* decoding times for the BI-AWGN channel. A key question is whether infinite decoding times are necessary for VLSF codes to approach Polyanskiy's achievability bound.

Yavas *et al.* [YKE21b] recently developed a non-asymptotic achievability bound for VLSF codes with finite decoding times for Gaussian channels. We first develop tight approximations to the tail probability of cumulative information density which will play an important role in numerical evaluations. Next, building upon Yavas *et al.*'s result, the problem of evaluating the achievable rate of a VLSF code reduces to an integer program of minimizing the upper bound on the average blocklength, subject to the average error probability, minimum gap, and integer constraints. By allowing real-valued decoding times and utilizing a two-step minimization, we develop the gap-constrained sequential differential optimization (SDO) procedure to numerically estimate the achievability bound. Numerical evaluations show that Polyanskiy's VLSF bound, which assumes infinite decoding times, can be closely approached with a finite and relatively small number of decoding times.

1.1 Summary of Contributions

We summarize the contributions of each chapter below. We remark that Chapters 2, 3 and 4 are independent of each other. Chapter 5 discusses the possible connections between these topics and some interesting open problems that are worth further investigation.

Chapter 2 Contributions

In Chapter 2, we consider the concatenation of a convolutional code (CC) with an optimized CRC code as a promising paradigm for good short blocklength codes for the BI-AWGN channel. The resulting CRC-aided convolutional code naturally permits the use of SLVD to achieve maximum-likelihood decoding. The convolutional encoder of interest is of rate-$1/\omega$ and the convolutional code is either zero-terminated (ZT) or tail-biting (TB). The resulting CRC-aided convolutional code is called a CRC-ZTCC or a CRC-TBCC.

To design a good CRC-aided convolutional code, we propose the *distance-spectrum optimal (DSO)* CRC polynomial. For a low target error probability, the DSO CRC polynomial corresponds to the one that maximizes the minimum distance of the resulting concatenated code. In this case, we provide an efficient DSO CRC search algorithm for the TBCC. The algorithm can be easily adapted to the ZTCC case.

To assess the performance of the CRC-aided convolutional code under SLVD, our analysis reveals that the complexity of SLVD is governed by the expected list rank which converges to 1 at high SNR. This allows a good performance to be achieved with a small increase in complexity. In this chapter, we focus on transmitting 64 information bits with a rate-1/2 convolutional encoder. For a target error probability 10^{-4}, simulations show that the best CRC-ZTCC approaches the RCU bound within 0.4 dB. Several CRC-TBCCs outperform the RCU bound at moderate SNR values.

Chapter 3 Contributions

In Chapter 3, we consider the problem of variable-length coding over the class of memoryless BAC with noiseless feedback, including the BSC as a special case. In 2012, Naghshvar *et al.* introduced a deterministic, one-phase coding scheme, which we refer to as the SED coding scheme. The deterministic VLF code constructed with the SED coding scheme asymptotically achieves both capacity and Burnashev's optimal error exponent for symmetric binary-input channels. Building on the work of Naghshvar *et al.*, this chapter extends the SED encoding scheme to the class of BACs and develops a non-asymptotic upper bound on the average blocklength that is shown to achieve both capacity and the optimal error exponent.

For the specific case of the BSC, we develop an additional non-asymptotic bound using a two-phase analysis that leverages both a submartingale synthesis and a Markov chain time of first passage analysis. Unlike Naghshvar *et al.*'s achievability bound which still lies beneath Polyanskiy's bound for VLSF codes, numerical evaluations show that both new achievability bounds exceed Polyanskiy's bound for VLSF codes.

Chapter 4 Contributions

In Chapter 4, we are interested in the performance of a VLSF code with m optimal decoding times for the binary-input additive white Gaussian noise channel. We first develop tight approximations to the tail probability of length-n cumulative information density by means of Edgeworth expansion and Petrov expansion. This will play an important role in the numerical evaluation of the upper bound on the average blocklength of a VLSF code.

Next, building on the work of Yavas *et al.*, for a given information density threshold, we formulate the integer program of minimizing the upper bound on average blocklength over all decoding times subject to the average error probability, minimum gap and integer constraints. Eventually, minimization of locally optimal upper bounds over all thresholds yields the globally minimum upper bound and this is called the two-step minimization.

For the integer program, by allowing positive real-valued decoding times, we develop the gap-constrained SDO procedure that sequentially produces the optimal, real-valued decoding times. We identify the error regime in which Polyanskiy's scheme of stopping at zero does not improve the achievability bound. In this error regime, the achievability bounds estimated by the two-step minimization and gap-constrained SDO show that Polyanskiy's achievability bound for VLSF codes can be approached with a small number of decoding times.

CHAPTER 2

CRC-Aided Convolutional Codes Under Serial List Viterbi Decoding

2.1 Introduction

Recently, the coding theory community has witnessed a growing interest in designing powerful short blocklength codes (e.g., codes with a thousand or fewer information bits). This renewed interest is mainly driven by the stringent requirement of new ultra-reliable low-latency communication in 5G [JPY18, SMA19], and advances in the finite-blocklength information theory developed by Polyanskiy, Poor, and Verdú [PPV10]. The basic question of finite-blocklength information theory asks: what is the maximal channel coding rate achievable at a given blocklength n and error probability ϵ? To answer this question, Polyanskiy *et al.* developed the *random-coding union (RCU) bound* $\mathrm{rcu}(n, M)$ [PPV10, Theorem 16] and the *meta-converse (MC) bound* $\mathrm{mc}(n, M))$ [PPV10, Theorem 27] that provide, respectively, tight upper and lower bounds on the error probability $P_e^*(n, M)$ of the best (n, M) code of length n and M codewords. Namely,

$$\mathrm{mc}(n, M) \leq P_e^*(n, M) \leq \mathrm{rcu}(n, M). \tag{2.1}$$

They also provide the *normal approximation* [PPV10, Eq. (223)] that tightly approximates the performance of the best (n, M) code. Thereafter, these bounds serve as benchmarks to assess the performance of a given finite-blocklength code over a broad class of channels, including the discrete memoryless channel (DMC) and the additive white Gaussian noise (AWGN) channel. Due to the prohibitive complexity of an exact computation of the RCU and MC bounds, saddlepoint approximations of these two bounds were developed that are shown to be numerically accurate [FVM18].

For coding theorists, a central task is to construct *structured* short-blocklength codes for the binary-input AWGN channel such that the probability of error falls into the region delimited by the RCU bound and the MC bound at a reasonable decoding complexity. There are numerous approaches to achieve this goal. As a comprehensive overview, Coşkun *et al.* surveyed in detail the contemporary short-blocklength code designs developed in recent decades [CDJ19]. Important examples include extended BCH codes under ordered statistics decoding (OSD) [FS95, YSV21], tail-biting convolutional codes under wrap-around Viterbi algorithm (WAVA) [GNJ17], non-binary low-density parity-check codes [DDS14, RDW19], non-binary turbo codes [LPM13, JM16], and polar codes [Ar09, TV15]. Recent advances also include the polarization-adjusted convolutional codes proposed by Arıkan [Ari19, YFV20]. It is worth noting that if no restrictions are imposed on what kind of codes should be used for the AWGN channel, Shannon [Sha59] has ingeniously shown that the optimal (n, M) code should be placed on a sphere in the n-dimensional Euclidean space such that the total solid angle is evenly split between the M Voronoi regions and every Voronoi region is a perfect circular cone in order to achieve the minimum probability of error.

While there are many possible structures for short-blocklength coding, this chapter focuses on the concatenation of a convolutional code with a cyclic redundancy check (CRC) code. The resulting concatenated code is called the *CRC-aided convolutional code.* Convolutional codes were first introduced by Elias [Eli55]. Viterbi decoding of convolutional codes was developed by Viterbi [Vit67] and its maximum-likelihood (ML) nature was recognized by Forney [For73, For74]. Advantages of convolutional codes include low decoding latency [HH09, MCF12] and good error correction performance at short blocklength. The term “CRC” stems from the use of cyclic codes for error detection [PB61], where the cyclic codeword can be put into systematic form with the parity bits easily generated by a linear sequential circuit. As explained in [BK19], CRC codes are possibly shortened cyclic codes generated by a polynomial whose leading and zero coefficients are nonzero. The order of the generator polynomial defines the blocklength of the associated cyclic code. However, in practice, the CRC code is a subcode of this cyclic code whose blocklength is less than the polynomial order.

The structure of concatenating a convolutional code with a CRC code was first proposed in the context of hybrid automatic repeat request (ARQ) [Ric94] and is used in numerous practical systems where the convolutional code serves as an inner error-correcting code to combat channel errors and the CRC code serves as an error-detecting code to verify if a codeword has been correctly received. Examples include the 3GPP cellular communication standards of both 3G [3GP06] and 4G LTE [3GP18].

The classical decoding approach for a CRC-aided convolutional code in a hybrid ARQ setting is Viterbi decoding with CRC verification. The input sequence identified by Viterbi decoding is checked to determine whether it is divisible by the CRC polynomial. This indicates whether a valid message has been decoded. If the decoded sequence is divisible by the CRC polynomial, the message segment of the decoded sequence is declared as the most likely message. Otherwise, a negative acknowledgement (NACK) is declared and perhaps a retransmission request is sent to the transmitter.

Unfortunately, the classical approach of Viterbi decoding with CRC verification conceals the true potential of the CRC-aided convolutional code. Performing a single Viterbi decoding step causes the decoder to give up too early, often before encountering a convolutional codeword whose input sequence passes the CRC verification. To unleash the power of the CRC-aided convolutional code, we consider the serial list Viterbi decoding (SLVD) pioneered by Seshadri and Sundberg [SS94]. SLVD sequentially produces a rank ordered list of codewords according to their likelihoods. Hence, CRC verification can naturally be used as a termination criterion for this list decoding.

Practical implementation of the SLVD typically assumes a *constrained maximum list size* Ψ to limit the peak decoding complexity. The SLVD terminates either when an input sequence passes the CRC verification or when the list rank reaches Ψ. The list rank at which the decoder stops is called the *terminating list rank* L. However, it is not always possible to have $L = \Psi$. This is because Ψ can be set arbitrarily large, yet only finitely many codewords exist. This implies that L has an intrinsic maximum achievable value independent of Ψ which is referred to as the *supremum list rank* λ. Consequently, L is a bounded random variable between 1 and $\min\{\lambda, \Psi\}$. Since the decoding complexity is a function of L, the average decoding complexity is a function of the average list rank $\mathbb{E}[L]$.

Assume that $\Psi < \lambda$. In this case, there are three possible outcomes associated with the SLVD: 1) a correct decoding if SLVD identifies the transmitted message within Ψ trials; 2) an undetected error (UE) if an erroneous input sequence found by SLVD passes the CRC verification within Ψ trials; and 3) a NACK and the forced termination of the decoder if the SLVD fails to find an input sequence that passes CRC verification within Ψ trials. In contrast, any value of Ψ with $\Psi \geq \lambda$ gives the same decoder behavior where no NACK is produced. In this case, the SLVD is an implementation of ML decoding of the CRC-aided convolutional code. In the extreme case where $\Psi = 1$, the SLVD reduces to the classical Viterbi decoding with CRC verification.

A classical list decoder [Eli57] assumes a fixed list size and declares decoding success as long as the transmitted codeword is in the list. In contrast, the SLVD has a more stringent requirement for success that can lead to a higher error probability than for the classical list decoder. Several upper bounds on error probability were developed for the classical list decoder, e.g., [BJK08, HSS10]. However, these results are not directly applicable to the SLVD.

This chapter focuses on the concatenation of a rate-$1/\omega$ convolutional code with an optimized CRC code. We explore both zero-terminated convolutional code (ZTCC) and tail-biting convolutional code (TBCC) [MW86]. The resulting concatenated code is called a *CRC-ZTCC* in the first case and a *CRC-TBCC* in the second case. For CRC-ZTCCs, Lou *et al.* [LDW15] realized that previous designs of CRC polynomials typically ignore the structure of the inner error-correcting code, which leads to suboptimal performance. Lou *et al.* designed optimal CRC polynomials for a given ZTCC such that the probability of UE is minimized for a single Viterbi decoding attempt followed by CRC verification. A key point in their analysis is that when the target probability of UE is low enough, the design principle is equivalent to maximizing the minimum distance of the CRC-ZTCC. However, Lou *et al.* did not address the optimal CRC design for a TBCC and did not consider SLVD.

Compared to the ZTCC, the TBCC has the advantage of avoiding the rate loss incurred by the overhead associated with the zero tail that follows the information sequence, but this overhead reduction comes with an increase in decoding complexity. A TB codeword requires that the initial and terminating states be the same, which can be achieved, for example, by setting the initial encoder memory to be the final bits of the information sequence. However, this requirement increases the difficulty of efficiently identifying the ML path on the trellis because the common value of the initial and terminating states is unknown at the decoder.

One approach to ML decoding of a TBCC is to perform Viterbi decoding from every possible initial state [MW86]. Various *approximate* algorithms are proposed for decoding the TBCC based on either ML or maximum *a posteriori* probability criterion, e.g., [WB89, CS94,AH98,SSF03]. Among these algorithms, the WAVA [SSF03] proves to be both efficient and near-ML. Shankar *et al.* [SKS] introduced an efficient, iterative, two-phase algorithm for *exact* ML decoding of TBCC, where an A* algorithm is applied in the second phase, using information from the first phase to compute the heuristic function. To make the exact SLVD of TBCC possible and efficient, this chapter extends Shankar *et al.*'s algorithm to accommodate the CRC polynomial. Specifically, if a traceback identifies a TB path, the CRC of the corresponding input sequence is checked. If the input sequence passes the CRC verification, the algorithm terminates. Otherwise, the algorithm locates the next rank ordered path.

2.1.1 Contributions

This chapter provides a design paradigm for both CRC-ZTCCs and CRC-TBCCs, a suite of tools for performance analysis of these codes, and a complexity analysis showing that SLVD allows low-complexity decoding at low probability of UE for $\Psi \geq \lambda$, i.e., an average decoding complexity similar to standard Viterbi decoding of the convolutional code alone. These contributions combine to yield, for example, CRC-aided convolutional codes that closely approach the RCU bound while requiring decoding complexity similar to Viterbi decoding on a convolutional code trellis with 2^8 states.

The main contributions of this chapter are summarized below.

1) *CRC-Aided Convolutional Code Design:* This chapter introduces the concept of the *distance-spectrum optimal (DSO) CRC polynomial*, which minimizes the theoretical union bound of the probability of UE for $\Psi \geq \lambda$. Theorem 1 shows that for high SNR, the DSO CRC polynomial reduces to the one that obtains the best minimum distance $d^l_{\min}$. Theorem 2 provides a sharp upper bound on the achievable $d^l_{\min}$ based on the distance spectrum of the convolutional code. For low target probability of UE, we present an efficient algorithm for finding DSO CRC polynomials for TBCCs of arbitrary rate, and provide these polynomials for ZTCCs and TBCCs for optimum rate-1/2 convolutional encoders in [LC04] at 64 information bits.

2) *CRC-Aided Convolutional Code Performance Analysis:* The performance of a CRC-aided convolutional code with the constrained maximum list size Ψ is measured by three probabilities: probability of correct decoding $P_{c,\Psi}$, probability of UE $P_{e,\Psi}$, and probability of NACK $P_{NACK,\Psi}$, where $P_{c,\Psi} + P_{e,\Psi} + P_{NACK,\Psi} = 1$. This chapter provides bounds, approximations, and simulation results characterizing how these probabilities vary with Ψ and with SNR. Theorems $4 - 6$ describe how performance evolves as Ψ increases, the existence and behavior of the supremum list rank λ, and performance (in terms of $P_{c,\Psi}$, $P_{e,\Psi}$, and $P_{NACK,\Psi}$) as a function of SNR for extreme values of $\Psi = 1$

and $\Psi = \lambda$.

3) *CRC-Aided Convolutional Code Decoding Complexity:* This chapter provides expressions for the complexity of SLVD for CRC-ZTCCs and CRC-TBCCs. These expressions reveal that complexity is a function of the expected list rank $\mathbb{E}[L]$. This chapter characterizes $\mathbb{E}[L]$ including a new approach to computing $\mathbb{E}[L]$ in the limit of low SNR, a new analysis of conditional expected list rank given the noise magnitude, and two new approaches for approximating the conditional expected list rank. Our parametric approximation on the conditional expected list rank naturally leads to an accurate approximation of $\mathbb{E}[L]$ as a function of $P_{e,\lambda}$ which shows that as $P_{e,\lambda}$ converges to 0, $\mathbb{E}[L]$ converges to 1 (see Approximation 3 to follow). We see that for practically interesting operating points of $P_{e,\lambda}$ such as 10^{-6}, $\mathbb{E}[L] \approx 1$ for typical CRC lengths. This implies that for an interesting range of CRC lengths, the CRC length can be increased with negligible impact on complexity. Moreover, for these CRC lengths, the complexity of SLVD for the CRC-aided convolutional code is very similar to that of standard Viterbi decoding of the convolutional code alone.

4) *Achieving the RCU Bound with Practical Complexity:* This chapter focuses on designing good CRC-aided convolutional codes for transmitting 64 information bits. Simulation results show that at target error probability 10^{-4}, the CRC-ZTCC with 8 memory elements can approach the RCU bound within 0.42 dB with decoding complexity similar to standard Viterbi decoding of the ZTCC. The best CRC-TBCC with 8 memory elements almost achieves the RCU bound, but requires increased decoding complexity.

2.1.2 Organization

This chapter is organized as follows: Section 4.2 introduces notation, the system architecture, TB trellises, Polyanskiy *et al.*'s finite-blocklength bounds, and the related saddlepoint approximations. Section 2.3 introduces the concept of the DSO CRC polynomial, shows that at high SNR the DSO CRC can be obtained by maximizing $d^l_{\min}$, provides an upper bound on $d^l_{\min}$, and gives a DSO CRC design algorithm for TBCCs of arbitrary rate at high SNR. Section 2.4 presents the performance and complexity analyses of SLVD of a given CRC-aided convolutional code. Section 2.5 presents simulation results of our designed CRC-aided convolutional codes and a comparison of $(128, 64)$ linear block codes. Section 2.6 concludes the chapter.

2.2 Preliminaries

2.2.1 Notation

Let $\mathbb{F}_2 = \{0, 1\}$ denote the binary field. $\mathbb{F}_2^n$ denotes the set of n-dimensional binary sequences. $\mathbb{F}_2[x]$ denotes the set of binary polynomials. The indicator function $\mathbf{1}_E$ takes the value 1 if the event E occurs, and 0 otherwise. The polynomial $u(x) = \sum_{i=0}^{n-1} u_i x^i \in \mathbb{F}_2[x]$ and its row vector form $\boldsymbol{u} = [u_0, u_1, \ldots, u_{n-1}] \in \mathbb{F}_2^n$ are used interchangeably. The CRC polynomial is represented in hexadecimal when its binary coefficients are written from the highest to lowest order. For instance, 0xD represents $x^3 + x^2 + 1$. The convolutional generator polynomial is represented in octal when the binary coefficients of each generator polynomial are written from the lowest to highest order. For instance, $(13, 17)$ represents $(1+x^2+x^3, 1+x+x^2+x^3)$. Let $w_H(\cdot)$, $d_H(\cdot, \cdot)$, and $\|\cdot\|$ denote the Hamming weight, Hamming distance, and Euclidean norm respectively. Finally, $\mathrm{cl}(S)$ and $\partial(S)$ denote the closure and the boundary of a subset $S \subseteq \mathbb{R}^n$, respectively.

2.2.2 Architecture

This chapter considers CRC-aided list decoding of convolutional codes, as depicted in Fig. 3.1. Let $u(x) = \sum_{i=0}^{k-1} u_i x^i \in \mathbb{F}_2[x]$ denote the k-bit binary information sequence, where u_{k-1} is the first bit entering the CRC encoder. The information sequence $u(x)$ is first encoded

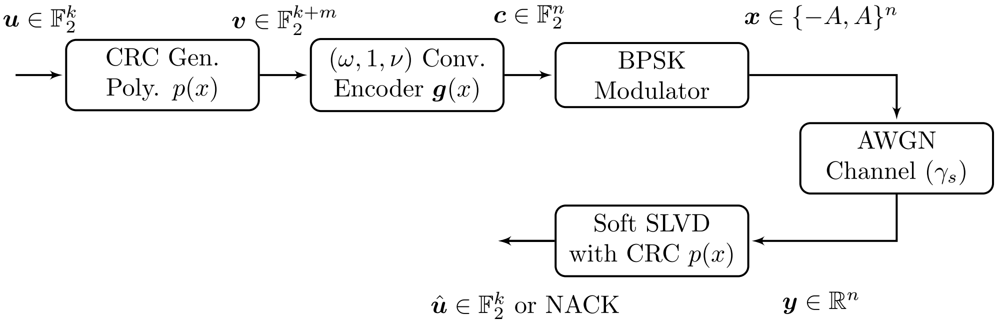
$\boldsymbol{u} \in \mathbb{F}_2^k$
$\boldsymbol{v} \in \mathbb{F}_2^{k+m}$
$\boldsymbol{c} \in \mathbb{F}_2^n$
$\boldsymbol{x} \in \{-A, A\}^n$
CRC Gen. Poly. $p(x)$
$(\omega, 1, \nu)$ Conv. Encoder $\boldsymbol{g}(x)$
BPSK Modulator
AWGN Channel (γ_s)
Soft SLVD with CRC $p(x)$
$\hat{\boldsymbol{u}} \in \mathbb{F}_2^k$ or NACK
$\boldsymbol{y} \in \mathbb{R}^n$

Figure 2.1: Block diagram of the CRC-aided list decoding of convolutional codes.

with a degree-m CRC generator polynomial $p(x) = 1 + p_1 x + \cdots + p_{m-1}x^{m-1} + x^m \in \mathbb{F}_2[x]$ to obtain m parity check bits $r(x) = x^m u(x) \mod p(x)$. Thus, we obtain $v^*(x) = x^m u(x) + r(x)$ which is divisible by the CRC polynomial $p(x)$. The final CRC-coded sequence $v(x)$ is produced by reversing $v^*(x)$, i.e., $v(x) = x^{k+m-1}v^*(x^{-1})$. This guarantees that the first bit entering the encoder, namely, u_{k-1} in $u(x)$, is always the lowest degree term of $v(x)$, consistent with common representation. The concatenated codeword $\boldsymbol{c} \in \mathbb{F}_2^n$ of blocklength n is obtained by convolutionally encoding $\boldsymbol{v}$ with a minimal, feedforward, $(\omega, 1, \nu)$ encoder $\boldsymbol{g}(x) = (g_1(x), g_2(x), \ldots, g_\omega(x))$, $g_i(x) = \sum_{j=0}^{\nu} g_{i,j}x^j$, with ν memory elements. To terminate a convolutional code into a linear block code, we consider either the ZT or TB method.

This chapter focuses on CRC-aided convolutional codes, but our analysis also involves the higher-rate convolutional code for which the CRC codeword $\boldsymbol{v}$ is the input message. To describe the two codes of interest as concisely as possible, define the higher-rate code $\mathcal{C}_h$ and the lower-rate code $\mathcal{C}_l$, where the latter is the CRC-aided convolutional code, as follows,

$$\mathcal{C}_h \triangleq \left\{\boldsymbol{c} \in \mathbb{F}_2^n : \boldsymbol{c} = \boldsymbol{v}\boldsymbol{G}, \forall \boldsymbol{v} \in \mathbb{F}_2^{k+m}\right\}, \tag{2.2}$$

$$\mathcal{C}_l \triangleq \left\{\boldsymbol{c} \in \mathbb{F}_2^n : \boldsymbol{c} = \boldsymbol{v}\boldsymbol{G}, \forall \boldsymbol{v} \in \mathbb{F}_2^{k+m} \text{ s.t. } p(x)|v^*(x)\right\}, \tag{2.3}$$

where $\boldsymbol{G} \in \mathbb{F}_2^{(k+m)\times n}$ is the matrix representation of the convolutional encoder. Intuitively, the effect of $p(x)$ is to obtain a subcode $\mathcal{C}_l$ from the given higher-rate code $\mathcal{C}_h$. The exact definition of $\mathcal{C}_h$ and $\mathcal{C}_l$ require the specification of the ZTCC or TBCC. For a ZTCC, $n =$

$\omega(k+m+\nu)$ and

$$\boldsymbol{G} = \begin{bmatrix} G_0 & G_1 & \cdots & G_\nu & & & \\ & G_0 & G_1 & \cdots & G_\nu & & \\ & & \ddots & \ddots & \ddots & \ddots & \\ & & & G_0 & G_1 & \cdots & G_\nu \end{bmatrix},$$

where

$$G_j = \begin{bmatrix} g_{1,j} & g_{2,j} & \cdots & g_{\omega,j} \end{bmatrix}, \quad j = 0, 1, \ldots, \nu.$$

Similarly, for a TBCC, $n = \omega(k+m)$ and

$$\boldsymbol{G} = \begin{bmatrix} G_0 & G_1 & \cdots & \cdots & G_\nu & & & \\ & G_0 & G_1 & \cdots & \cdots & G_\nu & & \\ & & \ddots & \ddots & \ddots & & \ddots & \\ & & & G_0 & G_1 & \cdots & \cdots & G_\nu \\ G_\nu & & & & G_0 & G_1 & \cdots & G_{\nu-1} \\ G_{\nu-1} & G_\nu & & & & \ddots & \ddots & \vdots \\ \vdots & & \ddots & & & & \ddots & G_1 \\ G_1 & G_2 & \cdots & G_\nu & & & & G_0 \end{bmatrix}.$$

Clearly, $\mathcal{C}_l \subseteq \mathcal{C}_h$, $|\mathcal{C}_h| = 2^{k+m}$, and $|\mathcal{C}_l| = 2^k$. The rate of the CRC-aided convolutional code (i.e., the lower-rate code) $R = k/n$. A fundamental quantity associated with a linear block code is its minimum distance. To aid our discussion, we define

$$d^h_{\text{min}} \triangleq \min\{w_H(\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}_h \setminus \{\boldsymbol{0}\}\}, \tag{2.4}$$

$$d^l_{\text{min}} \triangleq \min\{w_H(\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}_l \setminus \{\boldsymbol{0}\}\}. \tag{2.5}$$

As a corollary, $0 < d^h_{\text{min}} \leq d^l_{\text{min}}$. Note that for a ZTCC, d^h_{min} is in fact the order-$(k+m-1)$ row distance and is thus no less than the free distance of the convolutional code [JZ99].

The binary phase shift keying (BPSK) modulated sequence $\boldsymbol{x} = [x_0, x_1, \ldots, x_{n-1}]$ for codeword $\boldsymbol{c}$ is obtained via $x_i = (1-2c_i)A$, where A is the BPSK amplitude, and is then transmitted over the AWGN channel with channel SNR γ_s. Therefore, the channel model is

$$y_i = x_i + z_i, \quad i = 0, 1, \ldots, n-1, \tag{2.6}$$

where z_i's are independent and identically distributed (i.i.d.) according to the standard normal distribution. Thus, $\gamma_s = A^2$ or $A = \sqrt{\gamma_s}$.

Upon receiving the channel observations $\boldsymbol{y}$, the (soft) SLVD with a constrained maximum list size Ψ using CRC polynomial $p(x)$ is employed to determine the most likely information sequences $\hat{u}(x)$ from the trellis of the higher-rate code $\mathcal{C}_h$ based on $\boldsymbol{y}$ in a sequential manner using a maximum of Ψ trials. We assume that the SLVD sequentially produces rank ordered codewords that are also higher-rate codewords in $\mathcal{C}_h$. This is true when $\mathcal{C}_h$ is a ZTCC and may not be true when it is a TBCC in practice. If an input sequence $\hat{v}^*(x)$ associated with a higher-rate codeword passes the CRC verification, decoding terminates and the list stops growing. The corresponding list rank is marked as the terminating list rank L and the most likely information sequence $\hat{u}(x)$ is recovered from the last k bits of $\hat{v}^*(x)$. If an input sequence divisible by $p(x)$ is not found after Ψ attempts, the decoder terminates at list rank Ψ with a NACK as the output. As mentioned earlier, there exists a supremum list rank λ (whose formal definition will be given in (2.44)) which is independent of Ψ. If $\Psi \geq \lambda$, no NACK will occur. Consequently, L is always bounded between 1 and $\min\{\lambda, \Psi\}$.

A UE occurs if the SLVD erroneously identifies an input sequence $\hat{v}^*(x)$ that is divisible by $p(x)$ and $\hat{v}^*(x) \neq v^*(x)$. This is equivalent to the case where the UE polynomial $\hat{v}^*(x) - v^*(x) \in \mathbb{F}_2[x]$ is nonzero and is divisible by $p(x)$. Hence, an *error event* is given by the input-output pair $(\hat{v}(x) - v(x), \hat{c}(x) - c(x))$, where $\hat{v}(x) \neq v(x)$ and $\hat{c}(x)$ is a higher-rate codeword associated with $\hat{v}(x)$. By linearity, each error event corresponds to a pair of a nonzero input sequence $v(x)$ and its corresponding codeword $c(x)$. When restricted to convolutional codes, we can also use a trellis path to represent an error event.

The performance of the CRC-aided convolutional code is measured by three probabilities: probability of correct decoding $P_{c,\Psi}$, probability of UE $P_{e,\Psi}$, and probability of NACK

$P_{NACK,\Psi}$, where $P_{c,\Psi} + P_{e,\Psi} + P_{NACK,\Psi} = 1$. In the special case where $\Psi \geq \lambda$, $P_{c,\Psi} + P_{e,\Psi} = 1$. For ease of reference, we use $P_{e,\lambda}$ to represent $P_{e,\Psi}$ with $\Psi \geq \lambda$.

2.2.3 Tail-Biting Trellises

We follow [KV03] in describing a TB trellis. Let V be a set of vertices (or states). The set $\mathcal{A}$ is the output alphabet, and E is the set of edges described as ordered triples (v, a, v') with $v, v' \in V$, and $a \in \mathcal{A}$. In words, $(v, a, v') \in E$ denotes an edge that starts at v, ends at v', and has output a.

Definition 1 (Tail-biting trellises)**.** *A tail-biting trellis $T = (V, E, \mathcal{A})$ of depth N is an edge-labeled directed graph with the following property: the vertex set V can be partitioned as*

$$V = V_0 \cup V_1 \cup \cdots \cup V_{N-1} \tag{2.7}$$

such that every edge in T either begins at a vertex of V_i and ends at a vertex of V_{i+1} for some $i = 0, 1, \ldots, N-2$, or begins at a vertex of V_{N-1} and ends at a vertex of V_0.

Geometrically, a TB trellis can be viewed as a cylinder of N sections defined on some circular time axis. Alternatively, we can also define a TB trellis on a sequential time axis $\mathcal{I} = \{0, 1, \ldots, N\}$ with the restriction that $V_0 = V_N$ so that we obtain a conventional trellis.

For a trellis T of depth N, a trellis section connecting time i and $i+1$ is a subset $T_i \subseteq V_i \times \mathcal{A} \times V_{i+1} \subseteq E$ that specifies the allowed combination (s_i, a_i, s_{i+1}) of state $s_i \in V_i$, output symbol $a_i \in \mathcal{A}$, and state $s_{i+1} \in V_{i+1}$, $i = 0, 1, \ldots, N-1$. Such allowed combinations are called trellis branches. A trellis path $(\boldsymbol{s}, \boldsymbol{a}) \in T$ is a state/output sequence pair, where $\boldsymbol{s} \in V_0 \times V_1 \times \cdots \times V_N$, $\boldsymbol{a} \in \mathcal{A}^N$. Since $\boldsymbol{s}$ equivalently specifies the input sequence, an error event can also be described by its corresponding trellis path $(\boldsymbol{s}, \boldsymbol{a})$.

For a TB trellis T of depth N, a TB path $(\boldsymbol{s}, \boldsymbol{a})$ of length N on T is a *closed* path through N vertices. If T is defined on a sequential time axis $\mathcal{I} = \{0, 1, \ldots, N\}$, then any TB path $(\boldsymbol{s}, \boldsymbol{a})$ of length N satisfies $s_0 = s_N$.

2.2.4 Finite-Blocklength Bounds and Approximations

In [PPV10], Polyanskiy *et al.* derived the RCU bound and the MC bound that upper and lower bound the probability of error of the best (n, M) code, respectively. These two bounds serve as benchmarks to assess the performance of a given finite-blocklength code.

We follow the notation in [FVM18] to introduce the RCU bound and the MC bound. Let $W^n(\cdot|\cdot)$ denote a length-n channel. Let $\alpha_\beta(P, Q)$ denote the smallest type-I error probability among all tests discriminating between distributions P and Q, with a type-II error probability at most β [CT06a, Chapter 11.7]. For a random-coding ensemble defined over distribution P^n, the RCU bound is given by

$$\mathrm{rcu}(n, M) \triangleq \mathbb{E}[\min\{1, (M-1)\,\mathrm{pep}(X^n, Y^n)], \tag{2.8}$$

where $(X^n, Y^n) \sim P^n \times W^n$ and the pairwise error probability $\mathrm{pep}(x^n, y^n)$ is defined as

$$\mathrm{pep}(x^n, y^n) \triangleq \mathscr{P}\big(W^n(y^n|\bar{X}^n) \geq W^n(y^n|x^n)\big),$$

with $\bar{X}^n \sim P^n$. The MC bound is a minimax of a particular smallest type-I error probability

$$\mathrm{mc}(n, M) \triangleq \min_{P^n} \max_{Q^n} \left\{ \alpha_{\frac{1}{M}}(P^n \times W^n, P^n \times Q^n) \right\}, \tag{2.9}$$

where the minimization is over all input distributions P^n, and the maximization is over a set of auxiliary, independent of the input, output distributions Q^n.

An exact evaluation of the RCU bound and the MC bound involves integrating tail probabilities of n-dimensional random variables, which is computationally difficult even for simple channels and moderate values of n. In [FVM18], the authors provided saddlepoint approximations of these two bounds for memoryless symmetric channels, including the binary-input AWGN channel. These approximations are shown to be tight for a wide range of rates and blocklengths. Section 2.5 uses saddlepoint approximations to evaluate the RCU bound and the MC bound for the binary-input AWGN channel.

Approximation 1 (MC bound, [FVM18]). *For memoryless symmetric channels for which $Y \sim W(\cdot|x)$ is independent of x,*

$$\mathrm{mc}(n, M) \approx \max_{\rho \geq 0} \Big\{ e^{-n(E_0(\rho) - \rho E_0'(\rho))} \Big(\psi\big(\sqrt{nU(\rho)}\big) + \psi\big(\rho\sqrt{nU(\rho)}\big) - e^{-n(R - E_0'(\rho))} \Big) \Big\}, \tag{2.10}$$

where

$$E_0(\rho, P) = -\log \int_{\mathcal{Y}} \Big(\sum_{x \in \mathcal{X}} P(x) W(y|x)^{\frac{1}{1+\rho}} \Big)^{1+\rho} \mathrm{d}y, \tag{2.11}$$

$$E_0(\rho) = \max_P E_0(\rho, P), \tag{2.12}$$

$$\psi(x) = \frac{1}{2} \operatorname{erfc}\left(\frac{|x|}{\sqrt{2}} \right) e^{\frac{x^2}{2}} \operatorname{sign}(x), \tag{2.13}$$

$$U(\rho) = -(1+\rho) E_0''(\rho), \tag{2.14}$$

where $\mathcal{X}$ and $\mathcal{Y}$ denote the discrete input and continuous output alphabets of the channel, respectively. The maximization in (2.12) *is over all possible probability distributions on $\mathcal{X}$.*

Approximation 2 (RCU bound, [FVM18]). *For memoryless symmetric channels for which $Y \sim W(\cdot|x)$ is independent of x,*

$$\mathrm{rcu}(n, M) \approx \tilde{\xi}_n(\hat{\rho}) + \varphi_n(\hat{\rho}) e^{-n(E_0(\hat{\rho}, P) - \hat{\rho} R)}, \tag{2.15}$$

where $\hat{\rho}$ is the value for which $E_0'(\rho, P) = R$, and

$$Q_\rho(y) = \frac{1}{e^{-E_0(\rho, P)}} \Big(\sum_{x \in \mathcal{X}} P(x) W(y|x)^{\frac{1}{1+\rho}} \Big)^{1+\rho}, \tag{2.16}$$

$$\bar{\omega}''(\hat{\rho}) = \int_{\mathcal{Y}} Q_{\hat{\rho}}(y) \left[\frac{\partial^2}{\partial \tau^2} \Big(\log \sum_{x \in \mathcal{X}} P(x) W(y|x)^\tau \Big) \Big|_{\tau = \hat{\tau}} \right] \mathrm{d}y, \tag{2.17}$$

$$\theta_n(\hat{\rho}) = \frac{1}{\sqrt{1+\hat{\rho}}} \left(\frac{1+\hat{\rho}}{\sqrt{2\pi n \bar{\omega}''(\hat{\rho})}} \right)^{\hat{\rho}}, \tag{2.18}$$

$$\tilde{\xi}_n(\hat{\rho}) = \begin{cases} 1, & \hat{\rho} < 0 \\ 0, & 0 \leq \hat{\rho} \leq 1 \\ e^{-n(E_0(1,P) - R)} \theta_n(1), & \hat{\rho} > 1, \end{cases} \tag{2.19}$$

$$V(\hat{\rho}) = -E_0''(\hat{\rho}, P), \tag{2.20}$$

$$\varphi_n(\hat{\rho}) = \theta_n(\hat{\rho})\Big(\psi\big(\hat{\rho}\sqrt{nV(\hat{\rho})}\big) + \psi\big((1-\hat{\rho})\sqrt{nV(\hat{\rho})}\big)\Big). \tag{2.21}$$

2.3 The Search for the DSO CRC Polynomial

In this section, we seek to design good CRC-aided convolutional codes that provide the lowest possible probability of UE $P_{e,\lambda}$. To this end, for a given convolutional code, we design CRC polynomials that minimize the union bound on the probability of undetected error $P_{e,\lambda}$. The resulting CRC polynomial is known as the DSO CRC polynomial.

2.3.1 General Theory

For a given convolutional code and a desired CRC degree m, we wish to identify the degree-m CRC polynomial

$$p(x) = 1 + p_1x + \cdots + p_{m-1}x^{m-1} + x^m \in \mathbb{F}_2[x] \tag{2.22}$$

that minimizes the probability of UE $P_{e,\lambda}$. Since the exact probability $P_{e,\lambda}$ has no closed-form expression that can facilitate a design procedure, we use the union bound as the objective function that only involves the *distance spectrum*, $C_{d^l_{\min}}, \ldots, C_n$, of the lower-rate code $\mathcal{C}_l$, where C_d denotes the number of codewords in $\mathcal{C}_l$ of Hamming weight d, $d^l_{\min} \le d \le n$. The distance spectrum of the lower-rate code $\mathcal{C}_l$ is a function of both the CRC polynomial $p(x)$ and the higher-rate code $\mathcal{C}_h$. For any candidate CRC polynomial $p(x)$, the union bound on $P_{e,\lambda}$ is given by

$$\begin{aligned} P_{e,\lambda} &\le \sum_{\boldsymbol{c}\in\mathcal{C}_l\setminus\{\bar{\boldsymbol{c}}\}} \mathscr{P}\Big(Z > \frac{1}{2}\|\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{x}(\bar{\boldsymbol{c}})\|\,\big|\boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}})\Big) \\ &= \sum_{d=d^l_{\min}}^{n} C_d Q\big(A\sqrt{d}\big), \end{aligned} \tag{2.23}$$

where $\bar{\boldsymbol{c}} \in \mathcal{C}_l$ is the transmitted codeword, $\boldsymbol{x}(\boldsymbol{c}) \in \{-A, A\}^n$ is the BPSK-modulated point for codeword $\boldsymbol{c}$, $Z \sim \mathcal{N}(0,1)$, and

$$Q(x) \triangleq \int_x^{\infty} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} \, \mathrm{d}u \tag{2.24}$$

is the complementary Gaussian cumulative distribution function. $Q\big(A\sqrt{d}\big)$ computes the pairwise error probability of two codewords at distance d. For a given higher-rate code $\mathcal{C}_h$, a given SNR γ_s (i.e., $A = \sqrt{\gamma_s}$), and a CRC degree m, we define the degree-m *DSO CRC polynomial* as the one that minimizes the union bound on $P_{e,\lambda}$. Namely, the degree-m DSO CRC polynomial is the solution to the following optimization problem,

$$\min_{p(x)} \quad \sum_{d=d^l_{\min}}^{n} C_d Q\big(A\sqrt{d}\big). \tag{2.25}$$

Theoretically, the distance spectrum $C_{d^l_{\min}}, \dots, C_n$ of $\mathcal{C}_l$ can be found through a Viterbi search of the trellis of the higher-rate code $\mathcal{C}_h$, retaining only codewords whose input sequences are divisible by the candidate CRC polynomial $p(x)$. However, this approach requires the calculation of distance spectra for 2^{m-1} candidate CRC polynomials and quickly becomes computationally expensive as the information length k gets large. The degree-m DSO CRC polynomial depends on the specific higher-rate code and the SNR at which (2.23) is minimized. Note that the DSO CRC polynomial can be different for different values of k. In [LDW15], Lou *et al.* investigated how DSO CRC polynomials vary with information length k. Their essential finding is that a DSO CRC polynomial for a large k is usually "good" for shorter k. If the SNR is not sufficiently high, the CRC polynomial that minimizes the union bound in (2.23) may not minimize the actual $P_{e,\lambda}$.

Nevertheless, when SNR is sufficiently high or equivalently when the target probability of UE $P_{e,\lambda}$ is sufficiently low (typically less than 10^{-6}), the union bound (2.23) will be dominated by its first term $C_{d^l_{\min}} Q\Big(A\sqrt{d^l_{\min}}\Big)$ which becomes asymptotically tight to $P_{e,\lambda}$. Furthermore, in most cases at high SNR where the operating A is large enough, the first term in (2.23) is only dominated by $d^l_{\min}$. The following theorem justifies this statement.

Theorem 1. *For a given higher-rate code $\mathcal{C}_h$, let $C_{d^l_{\min,1}}, \ldots, C_n$ and $C'_{d^l_{\min,2}}, \ldots, C'_n$ be two distance spectra associated with lower-rate codes generated by CRC polynomials $p_1(x)$ and $p_2(x)$, respectively. If $d^l_{\min,1} < d^l_{\min,2}$, there exists a positive threshold A^* such that if $A > A^*$,*

$$\sum_{d=d^l_{\min,1}}^{n} C_d Q\big(A\sqrt{d}\big) > \sum_{d=d^l_{\min,2}}^{n} C'_d Q\big(A\sqrt{d}\big). \tag{2.26}$$

In the special case where $d^l_{\min,1} = d^l_{\min,2}$ and $C_{d^l_{\min,1}} > C'_{d^l_{\min,2}}$, the above conclusion still holds.

Proof. Assume that $d^l_{\min,1} < d^l_{\min,2}$. Since coefficients $C_{d^l_{\min,1}}, C'_{d^l_{\min,2}}$ are positive and bounded,

$$\begin{aligned}
&\lim_{A\to\infty} \frac{\sum_{d=d^l_{\min,1}}^{n} C_d Q(A\sqrt{d})}{\sum_{d=d^l_{\min,2}}^{n} C'_d Q(A\sqrt{d})} && (2.27)\\
&= \lim_{A\to\infty} \frac{C_{d^l_{\min,1}} \exp\Big(-\frac{A^2 d^l_{\min,1}}{2}\Big)\Big[1+\sum_{d=d^l_{\min,1}+1}^{n} \frac{C_d}{C_{d^l_{\min,1}}}\exp\Big(-\frac{A^2(d-d^l_{\min,1})}{2}\Big)\Big]}{C'_{d^l_{\min,2}} \exp\Big(-\frac{A^2 d^l_{\min,2}}{2}\Big)\Big[1+\sum_{d=d^l_{\min,2}+1}^{n} \frac{C'_d}{C'_{d^l_{\min,2}}}\exp\Big(-\frac{A^2(d-d^l_{\min,2})}{2}\Big)\Big]} && (2.28)\\
&= \lim_{A\to\infty} \frac{C_{d^l_{\min,1}}}{C'_{d^l_{\min,2}}} \exp\left(\frac{A^2}{2}(d^l_{\min,2} - d^l_{\min,1})\right) && (2.29)\\
&= \infty.
\end{aligned}$$

Hence, there exists a threshold A^* such that when $A > A^*$, (2.26) holds. In the special case where $d^l_{\min,1} = d^l_{\min,2}$ and $C_{d^l_{\min,1}} > C'_{d^l_{\min,2}}$, the limit in (2.29) is still greater than 1. Thus, the same conclusion follows. □

For sufficiently low target $P_{e,\lambda}$, the operating amplitude A is typically large enough such that $A > A^*$ is easily met in practice. In these common situations, the DSO CRC design principle reduces to maximizing the minimum distance $d^l_{\min}$ of the lower-rate code.

As an illustrative example, Fig. 2.2 shows the union bounds (2.23) for three degree-5 CRC polynomials among the 16 candidates for $k = 10$ and ZTCC $(13, 17)$. The CRC polynomial 0x37 minimizes the union bound at low SNR, whereas the CRC polynomial 0x2D minimizes the union bound at high SNR. On the contrary, the CRC polynomial 0x33 yields the worst possible union bound among all candidates. A detailed computation reveals that $d^l_{\min} = 11$, $C_{d^l_{\min}} = 17$ for 0x37, $d^l_{\min} = 12$, $C_{d^l_{\min}} = 76$ for 0x2D. Thus, the DSO CRC polynomial may not necessarily have the best minimum distance. The worst CRC polynomial 0x33 has $d^l_{\min} = 8$, $C_{d^l_{\min}} = 10$. In this example, the threshold at which the DSO CRC polynomial switches from 0x37 to 0x2D is -0.2398 dB. However, the gap between the performance of the two CRC polynomials is minimal, especially at low SNR. Nevertheless, both 0x37 and 0x2D achieve a gain of 0.5 dB compared to 0x33 at 10^{-2}, showing that the optimal CRC polynomial is crucial to achieving good performance.

For a given convolutional code and a specified CRC degree m, one may ask: how large can $d^l_{\min}$ be? The next theorem gives a tight upper bound on $d^l_{\min}$ in terms of the distance spectrum of the higher-rate code $\mathcal{C}_h$.

Theorem 2. *Given a specified CRC degree m and a higher-rate code $\mathcal{C}_h$ with distance spectrum $B_{d^h_{\min}}, \dots, B_n$, define w^* as the minimum w for which $\sum_{d=d^h_{\min}}^{w} B_d \geq 2^m$. For any degree-$m$ CRC polynomial, we have $d^l_{\min} \leq 2w^*$.*

Proof. Define the set $V(\boldsymbol{c})$ to be the set of codewords from the higher-rate code $\mathcal{C}_h$ that unambiguously decode to codeword $\boldsymbol{c}$ of the lower-rate code $\mathcal{C}_l$. Specifically, for each $\boldsymbol{c} \in \mathcal{C}_l$, define

$$V(\boldsymbol{c}) \triangleq \left\{\boldsymbol{r} \in \mathcal{C}_h : d_H(\boldsymbol{r}, \boldsymbol{c}) < d_H(\boldsymbol{r}, \boldsymbol{c}'),\ \forall \boldsymbol{c}' \in \mathcal{C}_l \setminus \{\boldsymbol{c}\}\right\}. \tag{2.30}$$

Hence, by linearity of the higher-rate code $\mathcal{C}_h$, the cardinality of $V(\boldsymbol{c})$ for every $\boldsymbol{c} \in \mathcal{C}_l$ is exactly the same. Hence,

$$|V(\boldsymbol{c})| \leq \frac{|\mathcal{C}_h|}{|\mathcal{C}_l|} = 2^m, \tag{2.31}$$

where (2.31) is an inequality because some codewords $\boldsymbol{r} \in \mathcal{C}_h$ may be equidistant from two or more lower-rate codewords.

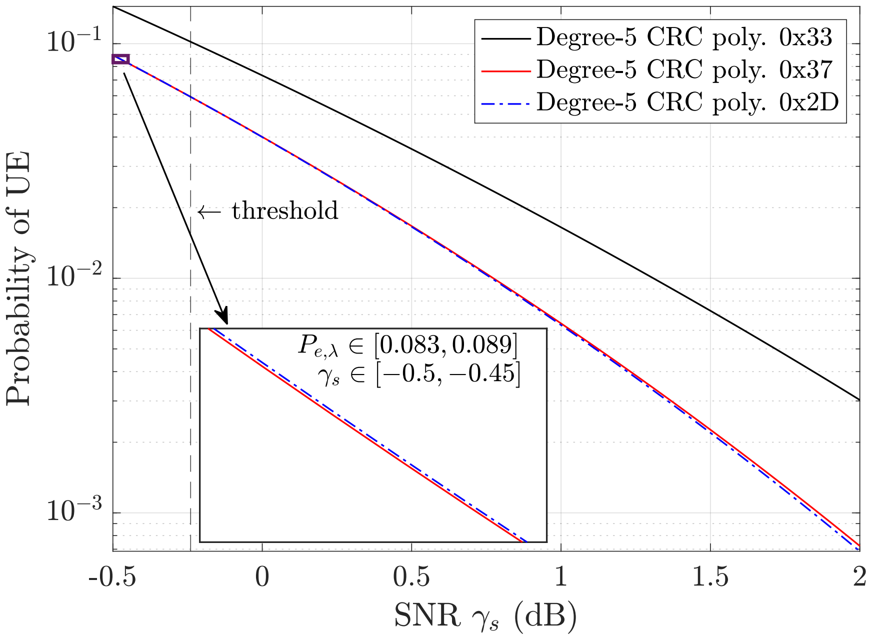

Figure 2.2: Comparison of the DSO CRC polynomials for $k = 10$, $m = 5$, and ZTCC $(13, 17)$. The blocklength of the CRC-ZTCC $n = 36$. The threshold value is -0.2398 dB.

Next, we show that for a given $\boldsymbol{c} \in \mathcal{C}_l$, $d_H(\boldsymbol{r}, \boldsymbol{c}) < \frac{1}{2}d^l_{\text{min}}$ implies that $\boldsymbol{r} \in V(\boldsymbol{c})$. By definition of the minimum distance, for two arbitrary distinct codewords $\boldsymbol{c}, \boldsymbol{c}' \in \mathcal{C}_l$, $d_H(\boldsymbol{c}, \boldsymbol{c}') \geq d^l_{\text{min}}$. Hence, for any $\boldsymbol{r} \in \mathcal{C}_h$, by triangle inequality,

$$d_H(\boldsymbol{r}, \boldsymbol{c}) + d_H(\boldsymbol{r}, \boldsymbol{c}') \geq d_H(\boldsymbol{c}, \boldsymbol{c}') \geq d^l_{\text{min}}. \tag{2.32}$$

Thus, if $d_H(\boldsymbol{r}, \boldsymbol{c}) < \frac{1}{2}d^l_{\text{min}}$, this implies that $d_H(\boldsymbol{r}, \boldsymbol{c}') > \frac{1}{2}d^l_{\text{min}}$ for any other $\boldsymbol{c}' \in \mathcal{C}_l$, i.e., $d_H(\boldsymbol{r}, \boldsymbol{c}) < d_H(\boldsymbol{r}, \boldsymbol{c}')$ for all $\boldsymbol{c}' \in \mathcal{C}_l \setminus \{\boldsymbol{c}\}$. By definition of $V(\boldsymbol{c})$, we conclude that $\boldsymbol{r} \in V(\boldsymbol{c})$.

By law of contraposition, if $\boldsymbol{r} \notin V(\boldsymbol{c})$, then $d_H(\boldsymbol{r}, \boldsymbol{c}) \geq \frac{1}{2}d^l_{\text{min}}$. Indeed, when $\sum_{d=d^h_{\text{min}}}^{w} B_d \geq 2^m$ (i.e., $\sum_{d=0}^{w} B_d \geq 2^m + 1$), by pigeonhole principle, there exists a codeword $\boldsymbol{r} \in \mathcal{C}_h$ that is outside of $V(\boldsymbol{c})$ and whose distance from $\boldsymbol{c}$ satisfies $d_H(\boldsymbol{r}, \boldsymbol{c}) \leq w$. Therefore, for this codeword $\boldsymbol{r}$, $w \geq d_H(\boldsymbol{r}, \boldsymbol{c}) \geq \frac{1}{2}d^l_{\text{min}}$ or equivalently, $d^l_{\text{min}} \leq 2d_H(\boldsymbol{r}, \boldsymbol{c}) \leq 2w$. Since this holds for any w satisfying $\sum_{d=d^h_{\text{min}}}^{w} B_d \geq 2^m$, the minimum such value w^* yields the tightest upper bound. □

Table 2.1: Comparison Between $d^l_{\min}$ Associated With the DSO CRC Polynomial and $2w^*$ Computed From Theorem 2 for $k = 64$

m	ZTCC $(13, 17)$			TBCC $(13, 17)$		
	$p(x)$	$d^l_{\min}$	$2w^*$	$p(x)$	$d^l_{\min}$	$2w^*$
0	0x1	6	12	0x1	6	12
3	0x9	10	12	0xF	8	12
4	0x1B	10	12	0x1F	9	12
5	0x2D	12	12	0x2D	10	12
6	0x43	12	12	0x63	12	12
7	0xB5	13	14	0xED	12	14
8	0x107	14	14	0x107	12	14
9	0x313	14	16	0x349	14	16
10	0x50B	15	18	0x49D	14	18

Table 2.1 shows the comparison between $d^l_{\min}$ and the upper bound $2w^*$ in Theorem 2 for both ZTCC and TBCC generated with the rate-1/2 convolutional encoder $(13, 17)$ at $k = 64$. We see that the upper bound is sharp as there exist DSO CRC polynomials that achieve this bound.

2.3.2 A Two-Phase DSO CRC Design Algorithm for TBCCs

We focus on finding the DSO CRC polynomial for low target $P_{e,\lambda}$. As discussed earlier, the design principle under this circumstance conveniently reduces to maximizing $d^l_{\min}$ of the lower-rate code. Thus, the optimal CRC polynomial depends on the convolutional code but not the SNR.

In principle, the DSO CRC design algorithm for low target $P_{e,\lambda}$ comprises a *collection phase* that gathers error events of the higher-rate code $\mathcal{C}_h$ up to a certain distance $\tilde{d}$, and a *search phase* that identifies the degree-m DSO CRC polynomial using the error events gathered in the collection phase. In this section, we propose a two-phase DSO CRC design algorithm particularized to TBCCs of arbitrary rate (including rate $1/\omega$). Later, we point out that our algorithm is also applicable to ZTCCs of arbitrary rate with a few distinctions.

The difficulty of designing DSO CRC polynomials for a TB trellis lies in the fact that a TB trellis is a union of 2^ν subtrellises that share trellis branches in the middle. Thus, to collect error events that meet the TB condition, a straightforward collection method is to perform Viterbi search separately at each possible start state to identify the *irreducible error event* (IEE) that leaves the start state once and rejoins it once, and then use them to reconstruct length-N TB paths with distance less than $\tilde{d}$. These IEEs constitute the error events of interest. However, this scheme will be *inefficient* in that for each nonzero start state, there exists a catastrophic IEE that spends a majority of time in the self-loop of the zero state. Such an IEE has the catastrophic property that its length grows unbounded with a finite weight. As a consequence, they are rarely used during reconstruction yet occupy a significant portion of total IEEs.

The algorithm we are about to propose follows the straightforward algorithm with the distinction in collecting IEEs. To circumvent the aforementioned catastrophic IEEs, we wish to identify IEEs whose weight is proportional to its length. To this end, we first partition the TB trellis into several sets that are closed under cyclic shifts. Next, all elements in each set are reconstructed via the concatenation of the corresponding IEEs and circular shifts of the resulting path.

For a given length-N TB trellis associated with a minimal convolutional encoder $\boldsymbol{g}(x)$, let $V_0 = \{0, 1, \dots, 2^\nu - 1\}$ be the set of possible encoder states. We seek a partition of the TB trellis, i.e., mutually exclusive sets that, together, contain all length-N TB paths. To do this, we define TBP(0) as the set that contains all TB paths that traverse state 0; TBP(1) contains the TB paths that traverse state 1 but not state 0; and so on. In general, the set TBP(σ) for $\sigma \in V_0$ is defined as follows,

$$\begin{aligned}\mathrm{TBP}(\sigma) \triangleq \big\{(\boldsymbol{s},\boldsymbol{a}) \in V_0^{N+1} \times \mathcal{A}^N :& s_0 = s_N;\\ &\exists i \in \mathcal{I} \text{ s.t. } s_i = \sigma;\ \forall i \in \mathcal{I},\ s_i \notin \{0,1,\dots,\sigma-1\}\big\}. \qquad (2.33)\end{aligned}$$

An important property of the above decomposition is that each set $\mathrm{TBP}(\sigma)$ is closed under cyclic shifts, as circularly shifting a TB path preserves the sequence of states that it traverses. Furthermore, such a partition of the TB trellis motivates the following IEE.

Definition 2 (Irreducible error events)**.** *For a TB trellis T on sequential time axis $\mathcal{I} = \{0, 1, \ldots, N\}$, the set of irreducible error events $(\boldsymbol{s}, \boldsymbol{a})$ at state $\sigma \in V_0$ is defined as*

$$\mathrm{IEE}(\sigma) \triangleq \bigcup_{l=1,2,\ldots,N} \overline{\mathrm{IEE}}(\sigma, l), \tag{2.34}$$

where

$$\overline{\mathrm{IEE}}(\sigma, l) \triangleq \big\{(\boldsymbol{s}, \boldsymbol{a}) \in V_0^{l+1} \times \mathcal{A}^l : s_0 = s_l = \sigma; \forall j, 0 < j < l,\ s_j \notin \{0, 1, \ldots, \sigma\}\big\}. \tag{2.35}$$

For ZTCCs, Lou *et al.* [LDW15] considered finding IEEs that start and end at the zero state and counting the allowed combinations. Hence, the IEE defined above generalizes Lou *et al.*'s IEEs. Since for a nonzero start state, no IEE can traverse the zero state, this guarantees that the weight of the IEE grows proportionally with its length, thus avoiding the catastrophic IEEs incurred in the straightforward algorithm.

With the set $\mathrm{TBP}(\sigma)$ defined as above, the following theorem describes how to efficiently find all elements in each $\mathrm{TBP}(\sigma)$ via the corresponding IEEs.

Theorem 3. *Every TB path $(\boldsymbol{s}, \boldsymbol{a}) \in \mathrm{TBP}(\sigma)$ can be constructed from the IEEs in* $\mathrm{IEE}(\sigma)$ *via concatenation and subsequent cyclic shifts.*

Proof. Let us consider T as a TB trellis defined on a sequential time axis $\mathcal{I} = \{0, 1, \ldots, N\}$. For any TB path $(\boldsymbol{s}, \boldsymbol{a}) \in \mathrm{TBP}(\sigma)$ of length N on T, we can first circularly shift it to the TB path $(\boldsymbol{s}^{(0)}, \boldsymbol{a}^{(0)}) \in \mathrm{TBP}(\sigma)$ on T satisfying $s_0^{(0)} = s_N^{(0)} = \sigma$.

Now, we examine $(\boldsymbol{s}^{(0)}, \boldsymbol{a}^{(0)})$. If $(\boldsymbol{s}^{(0)}, \boldsymbol{a}^{(0)})$ is already an element of $\mathrm{IEE}(\sigma)$, then there is nothing to prove. Otherwise, there exists a time index j, $0 < j < N$, such that $s_j = \sigma$. In this case, we break the TB path $(\boldsymbol{s}^{(0)}, \boldsymbol{a}^{(0)})$ at time j into two subpaths $(\boldsymbol{s}^{(1)}, \boldsymbol{a}^{(1)})$ and

$(\boldsymbol{s}^{(2)}, \boldsymbol{a}^{(2)})$, where

$$\begin{aligned}\boldsymbol{s}^{(1)} =&(s_0, s_1, \ldots, s_j),\ \boldsymbol{a}^{(1)} = (a_0, a_1, \ldots, a_{j-1}),\\ \boldsymbol{s}^{(2)} =&(s_j, s_{j+1}, \ldots, s_N),\ \boldsymbol{a}^{(2)} = (a_j, a_{j+1}, \ldots, a_{N-1}).\end{aligned}$$

Note that after segmentation of $(\boldsymbol{s}^{(0)}, \boldsymbol{a}^{(0)})$, the resultant two subpaths, $(\boldsymbol{s}^{(1)}, \boldsymbol{a}^{(1)})$ and $(\boldsymbol{s}^{(2)}, \boldsymbol{a}^{(2)})$, still meet the TB condition. Repeat the above procedure on $(\boldsymbol{s}^{(1)}, \boldsymbol{a}^{(1)})$ and $(\boldsymbol{s}^{(2)}, \boldsymbol{a}^{(2)})$. Since the length of a new subpath is strictly decreasing after each segmentation, the boundary case is the atomic subpath $(\boldsymbol{s}, \boldsymbol{a})$ of some length j^* satisfying $s_0 = s_{j*} = \sigma$, $s_{j'} \neq \sigma$, $\forall j' \in (0, j^*)$. Clearly, this atomic path is an element of IEE(σ). Thus, we successfully decompose a length-N TB path into elements of IEE(σ). Hence, reversing the above procedures will assemble elements of IEE(σ) into a length-N TB path. □

Algorithm 1 The Collection Procedure

```
Input: The TB trellis $T$, threshold $\tilde{d}$
Output: The list of IEEs $\mathcal{L}_{\text{IEE}}(\tilde{d}) = \{(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})\}$
Initialize empty lists $\mathcal{L}_\sigma$ for all $\sigma \in V_0$;
for $\sigma \leftarrow 0, 1, \ldots, |V_0| - 1$ do
    Perform Viterbi search at $\sigma$ on $T$ to collect list $\mathcal{L}_\sigma(\tilde{d})$ of all IEEs of distance less than $\tilde{d}$;
end for
return $\mathcal{L}_{\text{IEE}}(\tilde{d}) \leftarrow \bigcup_{\sigma \in V_0} \mathcal{L}_\sigma(\tilde{d})$;
```

Algorithm 2 The Search Procedure

```
Input: The trellis length $N$, degree $m$, list of IEEs $\mathcal{L}_{\text{IEE}}(\tilde{d})$
Output: The degree-$m$ DSO CRC polynomial $p(x)$
Initialize the list $\mathcal{L}_{\text{CRC}}$ of $2^{m-1}$ CRC candidates and empty lists $\mathcal{L}_{\text{TBP}}(d)$ of TBPs, $d = 1, \ldots, \tilde{d}-1$;
for $d \leftarrow 1, 2 \ldots, \tilde{d}-1$ do
    Construct all TBPs $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})$ from $\mathcal{L}_{\text{IEE}}(\tilde{d})$ s.t. $w_H(\boldsymbol{a}) = d$, $|\boldsymbol{v}| = N$, via concatenation and cyclic shifts; for each TBP, $\mathcal{L}_{\text{TBP}}(d) \leftarrow \mathcal{L}_{\text{TBP}}(d) \cup \{(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})\}$;
end for
$\text{Candi}(1) \leftarrow \mathcal{L}_{\text{CRC}}$;
for $d \leftarrow 1, \ldots, \tilde{d}-1$ do
    for $p_i(x) \in \text{Candi}(d)$ do
        Pass all $\boldsymbol{v}(x) \in \mathcal{L}_{\text{TBP}}(d)$ to $p_i(x)$;
        $C^{(i)} \leftarrow$ the number of divisible $\boldsymbol{v}(x)$ of dist. $d$;
    end for
    $C^* \leftarrow \min_{i \in \text{Candi}(d)} C^{(i)}$;
    $\text{Candi}(d+1) \leftarrow \{p_i(x) \in \text{Candi}(d) : C^{(i)} = C^*\}$;
    if $|\text{Candi}(d+1)| = 1$ then
        return $\text{Candi}(d+1)$;
    end if
end for
```

We now present our two-phase DSO CRC polynomial design algorithm for TBCCs of arbitrary rate (including rate $1/\omega$) at low target $P_{e,\lambda}$ that consists of the collection procedure described in Algorithm 1 and the search procedure described in Algorithm 2. In the collection procedure, $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})$ denotes the triple of states $\boldsymbol{s}$, outputs $\boldsymbol{a}$, and inputs $\boldsymbol{v}$, where the inputs $\boldsymbol{v}$ are uniquely determined by state transitions $s_i \rightarrow s_{i+1}$, $i = 0, 1, \ldots, N-1$. The TB trellis considered in the collection procedure should set a sufficiently large trellis length so that all

IEEs with distance less than $\tilde{d}$ are identified. Once the collection procedure is completed, one can reuse the collected IEEs in the search procedure for various trellis lengths. For a given higher-rate code $\mathcal{C}_h$ and a specified CRC degree m, by Theorem 2, it suffices to consider distance threshold $\tilde{d} \leq 2w^* + 1$ to identify the degree-m DSO CRC polynomial, where w^* is the minimum weight determined in Theorem 2.

In the search procedure, let $|\boldsymbol{v}|$ denote the length of $\boldsymbol{v}$. Steps from lines 2 to 4 use the IEEs to build all length-N trellis paths with distance less than $\tilde{d}$. In practice, this can be accomplished with dynamic programming. Specifically, for a given state $\sigma \in V_0$, let $\mathcal{L}_\sigma(w, l)$ denote the list of TB paths of weight w, of length l, and with initial state σ, $0 \leq w < \tilde{d}$, $1 \leq l \leq N$. Then, the update rule of $\mathcal{L}_\sigma(w, l)$ is as follows: given an IEE $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v}) \in \text{IEE}(\sigma)$ with $w_H(\boldsymbol{a}) \leq w$ and $|\boldsymbol{v}| < l$,

$$\mathcal{L}_\sigma(w, l) \leftarrow \mathcal{L}_\sigma(w, l) \cup \{\mathcal{L}_\sigma(w - w_H(\boldsymbol{a}), l - |\boldsymbol{v}|) \oplus (\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})\},$$

where $\mathcal{L}_\sigma(w, l) \oplus (\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})$ denotes appending $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{v})$ to the rear of each element in $\mathcal{L}_\sigma(w, l)$. The update rule inherently requires that w, l be enumerated in ascending order and $w_H(\boldsymbol{a}), |\boldsymbol{v}|$ in descending order. Finally, the set of length-N TB paths of distance less than $\tilde{d}$ via direct concatenation are given by $\bigcup_{\sigma \in V_0} \mathcal{L}_\sigma(\tilde{d} - 1, N)$. The rest of the TB paths are obtained by circularly shifting elements in $\bigcup_{\sigma \in V_0} \mathcal{L}_\sigma(\tilde{d} - 1, N)$.

Table 2.2: Optimum Rate-1/2 ZTCCs and Their DSO CRC Polynomials for $k = 64$ at Sufficiently Low Probability of UE $P_{e,\lambda}$

ν	ZTCC $\boldsymbol{g}(x)$	DSO CRC Polynomials							
		$m = 3$	4	5	6	7	8	9	10
3	$(13, 17)$	9	1B	2D	43	B5	107	313	50B
4	$(27, 31)$	F	15	33	4F	D3	13F	2AD	709
5	$(53, 75)$	9	11	25	49	EF	131	23F	73D
6	$(133, 171)$	F	1B	23	41	8F	113	2EF	629
7	$(247, 371)$	9	13	3F	5B	E9	17F	2A5	61D
8	$(561, 753)$	F	11	33	49	8B	19D	27B	4CF
9	$(1131, 1537)$	D	15	21	51	B7	1D5	20F	50D
10	$(2473, 3217)$	F	13	3D	5B	BB	105	20D	6BB

Table 2.3: Optimum Rate-1/2 TBCCs and Their DSO CRC Polynomials for $k = 64$ at Sufficiently Low Probability of UE $P_{e,\lambda}$

ν	TBCC $\boldsymbol{g}(x)$	DSO CRC Polynomials							
		$m = 3$	4	5	6	7	8	9	10
3	$(13, 17)$	F	1F	2D	63	ED	107	349	49D
4	$(27, 31)$	F	11	33	4F	B5	1AB	265	4D1
5	$(53, 75)$	9	11	3F	63	BD	16D	349	41B
6	$(133, 171)$	F	1B	3D	7F	FF	145	2BD	571
7	$(247, 371)$	F	11	33	63	EF	145	3A1	5D7
8	$(561, 753)$	F	11	33	7F	FF	1AB	301	4F5
9	$(1131, 1537)$	D	15	33	51	C5	1FF	349	583
10	$(2473, 3217)$	F	1B	33	79	BB	199	217	4DD

We remark that our algorithm can be generalized to ZTCCs of arbitrary rate by carrying out the following distinctions: the collection procedure only collects IEEs that start and terminate at the zero state; the search procedure only performs dynamic programming to reconstruct all ZT paths with the target trellis length N and distances less than $\tilde{d}$; termination tails of each ZT path should be removed before CRC verification. For interested readers, the DSO CRC design MATLAB routines are available for ZTCCs [Yanb] and for TBCCs [Yana].

Table 2.2 presents the DSO CRC polynomials of degree m from 3 to 10 that maximize $d^l_{\min}$ of CRC-ZTCCs based on a family of optimum rate-1/2 convolutional encoders in [LC04, Table 12.1(c)] with constraint length v from 3 to 10 for $k = 64$. These DSO CRC polynomials are designed for a sufficiently low $P_{e,\lambda}$. Table 2.3 presents the TBCC counterpart in the same setting. The code generated by the DSO CRC polynomial and convolutional encoder in the above tables is our designed CRC-aided convolutional code. In Section 2.5, we will present the performance and complexity trade-off of these codes.

2.4 Performance and Complexity of SLVD

This section explores the performance and complexity of SLVD. For a specified CRC-aided convolutional code, performance under SLVD is characterized by three probabilities: $P_{c,\Psi}$, $P_{e,\Psi}$, and $P_{NACK,\Psi}$. The average decoding complexity of SLVD is a function of expected list rank $\mathbb{E}[L]$. In order to understand the performance-complexity trade-off, we investigate how these quantities vary with system parameters including the SNR γ_s and the constrained maximum list size Ψ.

Geometrically speaking, the process of SLVD is to draw a list decoding sphere around the received sequence $\boldsymbol{y}$ with an increasing radius until the sphere touches the closest lower-rate codeword. To formalize this procedure, let us consider the set of received sequences $\boldsymbol{y} \in \mathbb{R}^n \setminus \mathcal{N}$ where $\mathcal{N}$ is the probability-zero set defined by $\mathcal{N} \triangleq \{\boldsymbol{y} \in \mathbb{R}^n : \exists\, \boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C}_h \text{ s.t. } \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_1)\| = \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_2)\|\}$. For every $\boldsymbol{y} \in \mathbb{R}^n \setminus \mathcal{N}$, let

$$\boldsymbol{c}_1(\boldsymbol{y}), \boldsymbol{c}_2(\boldsymbol{y}), \ldots, \boldsymbol{c}_{|\mathcal{C}_h|}(\boldsymbol{y}) \tag{2.36}$$

be an enumeration of $\mathcal{C}_h$ such that

$$\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_1(\boldsymbol{y}))\| < \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_2(\boldsymbol{y}))\| < \cdots < \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_{|\mathcal{C}_h|}(\boldsymbol{y}))\|.$$

Using the above enumeration, we formally define the terminating list rank $L(\boldsymbol{y})$ and the terminating Euclidean distance $d_t(\boldsymbol{y})$ for $\boldsymbol{y}$ as follows:

$$L(\boldsymbol{y}) \triangleq \min\{s \in \{1, 2, \dots, |\mathcal{C}_h|\} : \boldsymbol{c}_s(\boldsymbol{y}) \in \mathcal{C}_l\}, \tag{2.37}$$

$$d_t(\boldsymbol{y}) \triangleq \min_{\boldsymbol{c} \in \mathcal{C}_l} \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|. \tag{2.38}$$

Thus, the list decoding sphere of $\boldsymbol{y}$ can be expressed as

$$\mathcal{B}_{\mathrm{SLVD}}(\boldsymbol{y}) = \{\boldsymbol{c} \in \mathcal{C}_h : \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\| \leq d_t(\boldsymbol{y})\}. \tag{2.39}$$

Clearly, $L(\boldsymbol{y}) = |\mathcal{B}_{\mathrm{SLVD}}(\boldsymbol{y})|$.

The concepts above are defined for each individual received point $\boldsymbol{y} \in \mathbb{R}^n \setminus \mathcal{N}$. Alternatively, we can also consider the decoding region $\mathcal{Y}(\boldsymbol{c})$ (i.e., the Voronoi region) of each lower-rate codeword $\boldsymbol{c} \in \mathcal{C}_l$:

$$\mathcal{Y}(\boldsymbol{c}) \triangleq \big\{\boldsymbol{y} \in \mathbb{R}^n \setminus \mathcal{N} : \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\| < \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}')\|, \forall \boldsymbol{c}' \in \mathcal{C}_l \setminus \{\boldsymbol{c}\}\big\}. \tag{2.40}$$

For SLVD, the decoding region $\mathcal{Y}(\boldsymbol{c})$ can be further decomposed into finer subsets according to the list rank. Namely, for each $\boldsymbol{c} \in \mathcal{C}_l$ and a particular list rank $s \in \{1, 2, \dots, |\mathcal{C}_h| - |\mathcal{C}_l| + 1\}$,

$$\begin{aligned} \mathcal{Z}_s(\boldsymbol{c}) \triangleq \Big\{ & \boldsymbol{y} \in \mathbb{R}^n \setminus \mathcal{N} : \exists \boldsymbol{c}_1, \dots, \boldsymbol{c}_{s-1} \in \mathcal{C}_h \setminus \mathcal{C}_l \text{ s.t. } \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\| > \max_{j=1,2,\dots,s-1} \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_j)\| \\ & \text{and } \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\| < \min_{\boldsymbol{c}' \notin \mathcal{C}_h \setminus \{\boldsymbol{c}, \boldsymbol{c}_1, \dots, \boldsymbol{c}_{s-1}\}} \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}')\| \Big\}. \end{aligned} \tag{2.41}$$

Here, each $\mathcal{Z}_s(\boldsymbol{c})$ is referred to as the *order-s decoding region of* $\boldsymbol{c}$. Obviously, for each $\boldsymbol{c} \in \mathcal{C}_l$, we have

$$\mathcal{Z}_{s_1}(\boldsymbol{c}) \cap \mathcal{Z}_{s_2}(\boldsymbol{c}) = \varnothing, \quad \text{if } s_1 \neq s_2 \tag{2.42}$$

$$\mathcal{Y}(\boldsymbol{c}) = \bigcup_{s=1,2,\dots,|\mathcal{C}_h|-|\mathcal{C}_l|+1} \mathcal{Z}_s(\boldsymbol{c}). \tag{2.43}$$

By linearity of the code, the order-s decoding regions of all lower-rate codewords are isomorphic. With BPSK modulation, the bisection hyperplane of any two codewords passes through the origin of $\mathbb{R}^n$, making each order-s decoding region a polyhedron. Note that there exists a *supremum list rank* λ

$$\lambda \triangleq \max\{s \in \mathbb{Z}^+ : \mathcal{Z}_s(\boldsymbol{c}) \neq \varnothing, \forall \boldsymbol{c} \in \mathcal{C}_l\}. \tag{2.44}$$

Here, the supremum list rank λ only depends on $\mathcal{C}_l$ and $\mathcal{C}_h$ and is independent of Ψ. Hence, if $\Psi \geq \lambda$, the possible outcomes of SLVD only include correct decoding and UE. Namely, NACKs are not possible.

2.4.1 Performance Analysis

We first give our results on how $P_{c,\Psi}, P_{e,\Psi}$, and $P_{NACK,\Psi}$ vary with Ψ for a fixed SNR. Each of these probabilities may be understood as the probability of an event defined as a set of received sequences $\boldsymbol{y}$. For example, with $\bar{\boldsymbol{c}} \in \mathcal{C}_l$ as the transmitted codeword, by linearity, we have

$$P_{c,\Psi} = \mathscr{P}\left(\bigcup_{s=1,2\ldots,\lambda\wedge\Psi} \mathcal{Z}_s(\bar{\boldsymbol{c}}) \middle| \boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}}) \right) = \sum_{s=1}^{\lambda\wedge\Psi} \mathscr{P}\big(\mathcal{Z}_s(\bar{\boldsymbol{c}}) | \boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}})\big), \tag{2.45}$$

$$P_{e,\Psi} = \sum_{\boldsymbol{c}\in\mathcal{C}_l\setminus\{\bar{\boldsymbol{c}}\}} \mathscr{P}\left(\bigcup_{s=1,2,\ldots\lambda\wedge\Psi} \mathcal{Z}_s(\boldsymbol{c}) \middle| \boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}}) \right) = \sum_{s=1}^{\lambda\wedge\Psi} \sum_{\boldsymbol{c}\in\mathcal{C}_l\setminus\{\bar{\boldsymbol{c}}\}} \mathscr{P}\big(\mathcal{Z}_s(\boldsymbol{c}) | \boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}})\big), \tag{2.46}$$

where $\lambda \wedge \Psi \triangleq \min\{\lambda, \Psi\}$.

Theorem 4. *For a given CRC-aided convolutional code decoded with SLVD at a fixed SNR, $P_{c,\Psi}$ and $P_{e,\Psi}$ are both strictly increasing in Ψ and will converge to $P_{c,\lambda}$ and $P_{e,\lambda}$ respectively, where $P_{c,\lambda} + P_{e,\lambda} = 1$.*

Proof. According to (4.41) and (4.42), $P_{c,\Psi}$ and $P_{e,\Psi}$ are summations of the order-s decoding regions $\mathscr{P}(\mathcal{Z}_s(\boldsymbol{c})|\boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}}))$, thus are non-decreasing in Ψ. For each $\boldsymbol{c} \in \mathcal{C}_l$ and $s = 1, 2, \ldots, \lambda$, $\mathscr{P}(\mathcal{Z}_s(\boldsymbol{c})|\boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}}))$ is solely determined by the SNR value and is independent of Ψ. Since every order-s decoding region $\mathcal{Z}_s(\boldsymbol{c})$ is the intersection of halfplanes, it follows that each $\mathcal{Z}_s(\boldsymbol{c})$ is an open set. Hence, to show the strict increasing property, it suffices to show that each $\mathcal{Z}_s(\boldsymbol{c})$ is nonempty. To this end, we use induction to show that all $\mathcal{Z}_s(\boldsymbol{c})$, $s = 1, 2, \ldots, \lambda$, are open and nonempty.

By definition, $\mathcal{Z}_\lambda(\boldsymbol{c})$ is nonempty. Assume $\mathcal{Z}_s(\boldsymbol{c})$ is nonempty for some $2 \leq s \leq \lambda$, we need to show that $\mathcal{Z}_{s-1}(\boldsymbol{c})$ is also nonempty. By assumption, there exists $\boldsymbol{y} \in \mathcal{Z}_s(\boldsymbol{c})$ with

$\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{s-1}, \boldsymbol{c} \in \mathcal{B}_{\mathrm{SLVD}}(\boldsymbol{y})$, where $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{s-1} \in \mathcal{C}_h \setminus \mathcal{C}_l$ and $\boldsymbol{c} \in \mathcal{C}_l$. Next, we show that with probability 1, a point $\boldsymbol{y}'$ can be constructed from $\boldsymbol{y}$ such that there exists a single $j \in \{1, 2, \ldots, s-1\}$ such that $\boldsymbol{x}(\boldsymbol{c}_j)$ and $\boldsymbol{x}(\boldsymbol{c})$ are respectively the furthest and second furthest points from $\boldsymbol{y}'$. This implies that $\boldsymbol{y}' \in \mathcal{Z}_{s-1}(\boldsymbol{c})$.

We construct the new point $\boldsymbol{y}'$ as $\boldsymbol{y}' = \boldsymbol{y} + t(\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{y})$, where $t \in [0, 1]$. Hence,

$$\|\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{y}'\| = (1-t)\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|. \tag{2.47}$$

Therefore, it is equivalent to showing that there exists $t \in (0, 1)$ such that for some $j \in \{1, 2, \ldots, s-1\}$,

$$\|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_j)\| > (1-t)\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|, \tag{2.48}$$

$$\max_{i \in \{1,\ldots,s-1\} \setminus \{j\}} \|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_i)\| < (1-t)\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|. \tag{2.49}$$

To this end, we show that the set of $\boldsymbol{y}$ for which no such t exists has a probability of zero. First, consider function

$$F(t) \triangleq \max_{i=1,2,\ldots,s-1} \|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_i)\| - (1-t)\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|.$$

Since each $\|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_i)\|$, $i = 1, 2, \ldots, s-1$, is a continuous function in t, $F(t)$ is also a continuous function in $t \in [0, 1]$. Note that

$$F(0) = \max_{i=1,2,\ldots,s-1} \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_i)\| - \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\| < 0, \tag{2.50}$$

$$F(1) = \max_{i=1,2,\ldots,s-1} \|\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{x}(\boldsymbol{c}_i)\| > 0. \tag{2.51}$$

By the intermediate value theorem, there exists a $t^* \in (0, 1)$ such that

$$\max_{i=1,2,\ldots,s-1} \|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_i)\| = (1-t^*)\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|. \tag{2.52}$$

Hence, the converse case can only occur if there exist two codewords $\boldsymbol{c}_{j_1}$ and $\boldsymbol{c}_{j_2}$, $j_1 \neq j_2$, such that

$$\|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_{j_1})\| = \|\boldsymbol{y}' - \boldsymbol{x}(\boldsymbol{c}_{j_2})\| = (1-t^*)\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c})\|. \tag{2.53}$$

Namely, the converse case only occurs if there exist two distinct points in $\mathcal{B}_{\text{SLVD}}(\boldsymbol{y})$ that are equally furthest from $\boldsymbol{y}'$. If (3.78) holds, this implies that $\boldsymbol{y}'$ lies on the intersection of two hyperplanes: one that bisects $\boldsymbol{x}(\boldsymbol{c}_{j_1})\boldsymbol{x}(\boldsymbol{c})$ and the other that bisects $\boldsymbol{x}(\boldsymbol{c}_{j_2})\boldsymbol{x}(\boldsymbol{c})$. Namely, $\boldsymbol{y}'$ lies on an $(n-2)$-dimensional hyperplane that crosses the origin. Hence, such $\boldsymbol{y}'$ only occurs if line segment $\boldsymbol{y}\boldsymbol{x}(\boldsymbol{c})$ intersects with any of these $(n-2)$-dimensional hyperplanes. Therefore, the set of $\boldsymbol{y}$ for which the converse case occurs is the union of finitely many $(n-1)$-dimensional hyperplanes, and thus has zero probability. Hence, we can construct a $\boldsymbol{y}'$ from $\boldsymbol{y} \in \mathcal{Z}_s(\boldsymbol{c})$ such that $\boldsymbol{y}' \in \mathcal{Z}_{s-1}(\boldsymbol{c})$ with probability 1. Namely, $\mathcal{Z}_{s-1}(\boldsymbol{c})$ is also nonempty.

By induction, every order-s decoding region $\mathcal{Z}_s(\boldsymbol{c})$, $s = 1, 2, \ldots, \lambda$, is open and nonempty. Thus, $P_{c,\Psi}$ and $P_{e,\Psi}$ are both strictly increasing in Ψ and will converge to $P_{c,\lambda}$ and $P_{e,\lambda}$, respectively, provided that $\Psi \geq \lambda$. □

As an example, Fig. 2.3 shows the probability of UE $P_{e,\Psi}$ and probability of NACK $P_{NACK,\Psi}$ vs. the constrained maximum list size Ψ for $k = 64$, degree-6 DSO CRC polynomial 0x43 and ZTCC $(13, 17)$. It can be seen that $P_{e,\Psi}$ quickly increases and converges to $P_{e,\lambda}$ when Ψ is relatively small.

The monotone property of $P_{e,\Psi}$ with Ψ in Theorem 4 indicates that for a fixed SNR value,

$$P_{e,1} \leq P_{e,\Psi} \leq P_{e,\lambda}, \quad \forall \Psi \in \mathbb{Z}^+. \tag{2.54}$$

The proof of Theorem 4 also implies that the closure of the order-λ decoding region must intersect with the boundary of $\mathcal{Y}(\boldsymbol{c})$, $\boldsymbol{c} \in \mathcal{C}_l$. We formalize this notion in Theorem 5.

Theorem 5. *For any lower-rate codeword* $\boldsymbol{c} \in \mathcal{C}_l$*,* $\text{cl}(\mathcal{Z}_\lambda(\boldsymbol{c})) \cap \partial\mathcal{Y}(\boldsymbol{c}) \neq \varnothing$*.*

Proof. Fix a lower-rate codeword $\boldsymbol{c} \in \mathcal{C}_l$. Let $\boldsymbol{y} \in \mathcal{Z}_\lambda(\boldsymbol{c})$. Consider $\boldsymbol{y}' = \boldsymbol{y} + t(\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}))$, $t \geq 0$. By the proof in Theorem 4, if $\boldsymbol{y}' \in \mathcal{Y}(\boldsymbol{c})$, $L(\boldsymbol{y}') \geq L(\boldsymbol{y}) = \lambda$. Since λ is the supremum list rank, $L(\boldsymbol{y}') = \lambda$ for all $0 \leq t < t^*$, where t^* is the threshold at which $\boldsymbol{y}' \in \partial\mathcal{Y}(\boldsymbol{c})$. This implies that $\text{cl}(\mathcal{Z}_\lambda(\boldsymbol{c})) \cap \partial\mathcal{Y}(\boldsymbol{c}) \neq \varnothing$. □

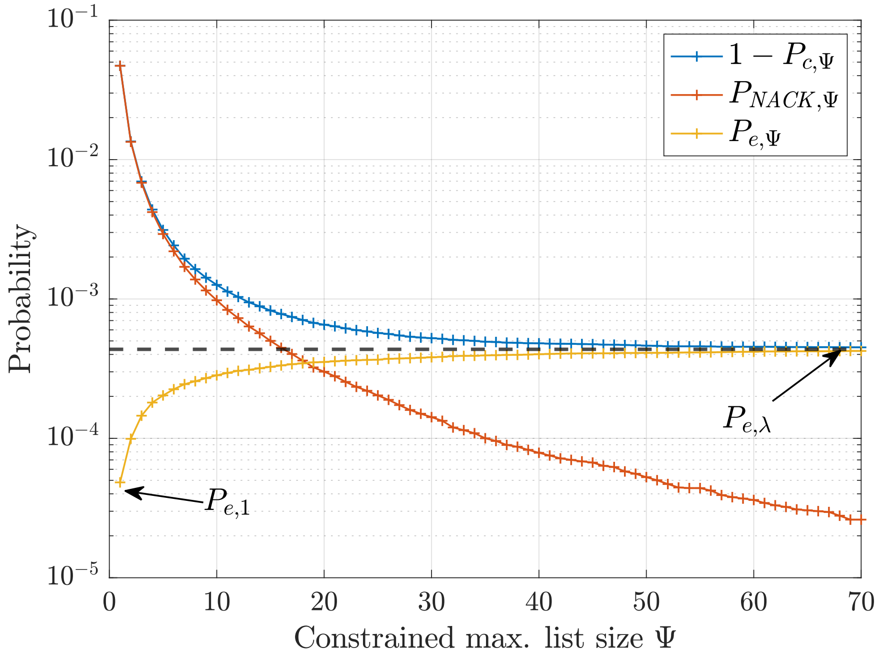

Figure 2.3: $1 - P_{c,\Psi}, P_{NACK,\Psi}, P_{e,\Psi}$ vs. the constraint maximum list size Ψ at SNR $\gamma_s = 3$ dB for ZTCC $(13, 17)$, degree-6 DSO CRC polynomial 0x43, and $k = 64$ in Table 2.2. The black, dashed line represents $P_{e,\lambda}$.

Theorem 5 indicates that one can find λ by following along the boundary of $\mathcal{Y}(\boldsymbol{c})$ and making a slight deviation towards the decoding region $\mathcal{Y}(\boldsymbol{c})$. This approach is computationally challenging in $\mathbb{R}^n$ for interesting values of n. While $|\mathcal{C}_h| - |\mathcal{C}_l| + 1$ is a straightforward upper bound on λ, it remains an open problem to identify a tighter bound on λ and to develop an efficient algorithm to compute λ.

We next direct our attention to quantifying $P_{e,1}$, $P_{e,\lambda}$ in terms of the SNR (or equivalently in terms of amplitude A) and the distance spectra of both the lower-rate code $\mathcal{C}_l$ and the higher-rate code $\mathcal{C}_h$.

Theorem 6. *Under SLVD of a CRC-aided convolutional code with higher-rate distance spec-*

trum $B_{d^h_{\min}}, \ldots, B_n$ *and lower-rate distance spectrum* $C_{d^l_{\min}}, \ldots, C_n$*,*

$$P_{e,1} \leq \min\left\{2^{-m},\ \sum_{d=d^l_{\min}}^{n} C_d Q\big(A\sqrt{d}\big)\right\} \tag{2.55}$$

$$\approx \min\left\{2^{-m},\ C_{d^l_{\min}} Q\Big(A\sqrt{d^l_{\min}}\Big)\right\}, \tag{2.56}$$

$$P_{e,\lambda} \leq \min\left\{1,\ \sum_{d=d^l_{\min}}^{n} C_d Q\big(A\sqrt{d}\big)\right\} \tag{2.57}$$

$$\approx \min\left\{1,\ \sum_{d=d^l_{\min}}^{\tilde{d}} C_d Q\big(A\sqrt{d}\big)\right\}, \tag{2.58}$$

$$P_{NACK,1} \approx \min\left\{1 - 2^{-m},\ \sum_{d=d^h_{\min}}^{\tilde{d}} B_d Q\big(A\sqrt{d}\big) - C_{d^l_{\min}} Q\Big(A\sqrt{d^l_{\min}}\Big)\right\}, \tag{2.59}$$

where the second approximation in braces in (2.56) *is called the nearest neighbor approximation, and the second approximation in* (2.58) *is called the truncated union bound (TUB) at distance* $\tilde{d} \in \mathbb{Z}^+$*.*

Proof. First, note that $P_{e,\Psi}$ is a monotonically decreasing function of A for any Ψ. This can be seen from (4.42) where as A increases, the center of the Gaussian density is moving away from every $\boldsymbol{x}(\boldsymbol{c})$ for $\boldsymbol{c} \in \mathcal{C}_l \setminus \{\bar{\boldsymbol{c}}\}$. Hence, the corresponding probability $\mathscr{P}(\mathcal{Z}_s(\boldsymbol{c})|\boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}}))$ decreases with A, causing $P_{e,\Psi}$ to decrease with A.

Now we focus on the $\Psi = 1$ case. The previous paragraph reveals that $P_{e,1}$ has its maximum value at $A = 0$. As $A \to 0$, the transmitted point converges to the origin $\boldsymbol{O}$ in $\mathbb{R}^n$. At the limit where $\boldsymbol{x}(\bar{\boldsymbol{c}}) = \boldsymbol{O}$, the symmetry of the Gaussian density and linearity of the code ensures that each order-1 decoding region has a probability of $2^{-(k+m)}$. Hence,

$$P_{e,1} = \sum_{\boldsymbol{c} \in \mathcal{C}_l \setminus \{\bar{\boldsymbol{c}}\}} \mathscr{P}(\mathcal{Z}_1(\boldsymbol{c})|\boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}})) \tag{2.60}$$

$$\leq \lim_{A \to 0} \sum_{\boldsymbol{c} \in \mathcal{C}_l \setminus \{\bar{\boldsymbol{c}}\}} \mathscr{P}(\mathcal{Z}_1(\boldsymbol{c})|\boldsymbol{X} = \boldsymbol{x}(\bar{\boldsymbol{c}})) \tag{2.61}$$

$$= \sum_{\boldsymbol{c} \in \mathcal{C}_l \setminus \{\bar{\boldsymbol{c}}\}} \mathscr{P}(\mathscr{Z}_1(\boldsymbol{c}) | \boldsymbol{X} = \boldsymbol{O})) \tag{2.62}$$

$$= (2^k - 1)2^{-(k+m)} \leq 2^{-m}. \tag{2.63}$$

For any SNR value, $P_{e,1} < P_{e,\lambda}$ so that the union bound (2.23) is also an upper bound for $P_{e,1}$. Hence, the minimum between the two is an upper bound on $P_{e,1}$. As SNR increases, the majority of probability will concentrate on the nearest neighbors of $\bar{\boldsymbol{c}}$, hence, we approximate $P_{e,1}$ only using the nearest neighbors.

For $P_{e,\lambda}$, we upper bound it by the union bound (2.23). For ease of computation, we can consider the TUB up to a sufficient distance $\tilde{d}$ to approximate the original union bound.

For $P_{NACK,1}$, in the extremely low SNR regime (i.e., when A is close to 0), $P_{c,1} \approx 2^{-(k+m)}$ and $P_{e,1} \approx 2^{-m}(1 - 2^{-k})$. It follows that

$$P_{NACK,1} = 1 - P_{e,1} - P_{c,1} \approx 1 - 2^{-m}. \tag{2.64}$$

For an arbitrary SNR, invoking the union bound on $P_{NACK,1} + P_{e,1}$ yields

$$P_{NACK,1} + P_{e,1} \leq \sum_{d=d^h_{\min}}^{n} B_d Q\big(A\sqrt{d}\big) \approx \sum_{d=d^h_{\min}}^{\tilde{d}} B_d Q\big(A\sqrt{d}\big).$$

Hence,

$$P_{NACK,1} \approx \sum_{d=d^h_{\min}}^{\tilde{d}} B_d Q\big(A\sqrt{d}\big) - C_{d^l_{\min}} Q\Big(A\sqrt{d^l_{\min}}\Big). \tag{2.65}$$

This concludes the proof of Theorem 6. □

Fig. 2.4 shows simulation results and approximations for the three probabilities stated in Theorem 6: $P_{NACK,1}$, $P_{e,1}$, and $P_{e,\lambda}$. As SNR increases, all three approximations become asymptotically tight to the respective $P_{e,1}$, $P_{NACK,1}$, and $P_{e,\lambda}$. The nearest neighbor approximation will eventually become asymptotically tight for $P_{e,\lambda}$, but is a tight approximation for $P_{e,1}$ at a much lower SNR.

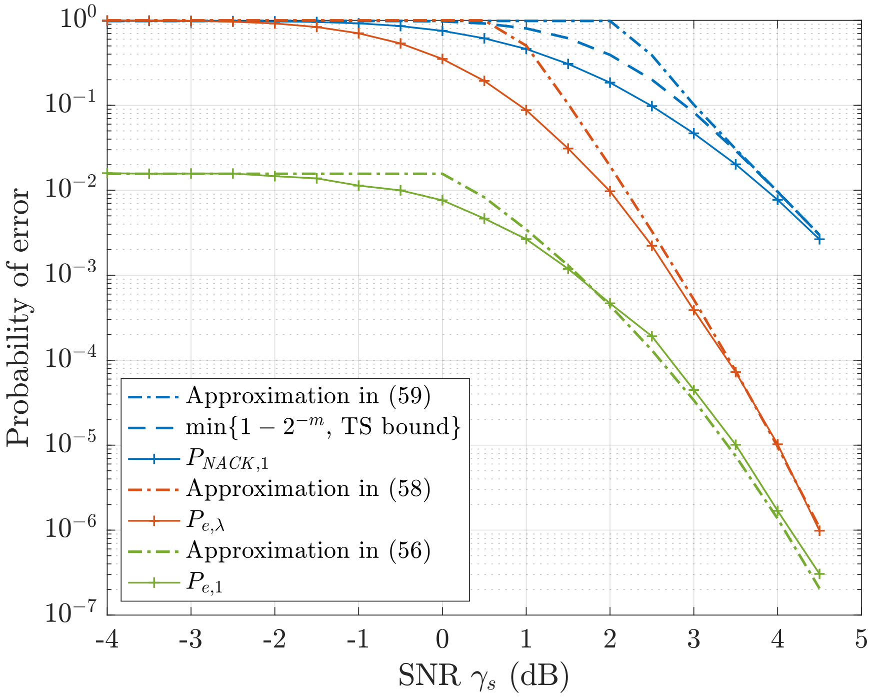

Figure 2.4: $P_{NACK,1}$, $P_{e,\lambda}$, and $P_{e,1}$ vs. SNR γ_s for ZTCC $(13, 17)$, degree-6 DSO CRC polynomial 0x43 and $k = 64$ in Table 2.2. The TUBs in (2.58) and (2.59) are obtained at $\tilde{d} = 24$. The TS bound on $P_{NACK,1}$ is plotted using [YK04a, Eq. (14)].

We remark that improved upper bounds on $P_{NACK,1}$ and $P_{e,\lambda}$ can be derived using Gallager's first bounding technique [Gal63], provided that the full distance spectra of $\mathcal{C}_h$ and $\mathcal{C}_l$ are known, respectively. Some classical examples include the tangential bound [Ber80], the tangential sphere (TS) bound [Pol94,YK04a], and the added-hyperplane bound [YK04b]. These bounds provide a tight estimation at high noise levels and converge to the union bound at low noise levels. As an example, in Fig. 2.4, we plot the minimum between $(1-2^{-m})$ and the TS bound for $P_{NACK,1}$ following [YK04a, Eq. (14)]. It can be seen that the TS bound quickly converges to the TUB as SNR increases. Since this chapter mainly focuses on low target error probability, we only consider the TUB for estimating $P_{NACK,1}$ and $P_{e,\lambda}$.

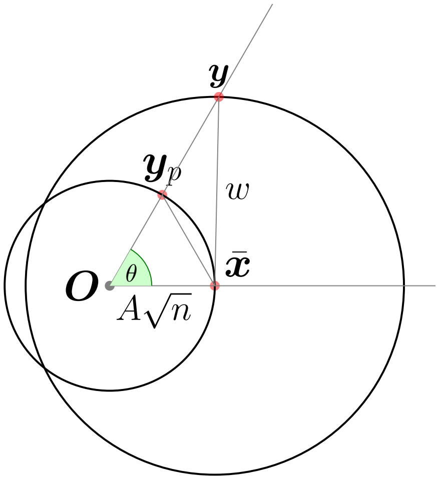

Figure 2.5: An illustration of the projection method.

2.4.2 Analysis of the Expected List Rank

For a fixed transmitted point $\bar{\boldsymbol{x}}$, observe that $\mathscr{P}(L = s|\boldsymbol{X} = \bar{\boldsymbol{x}}) = \sum_{\boldsymbol{c}\in\mathcal{C}_l} \mathscr{P}(Z_s(\boldsymbol{c})|\boldsymbol{X} = \bar{\boldsymbol{x}})$ is independent of Ψ. Combining with the linearity $\mathbb{E}[L] = \mathbb{E}[L|\boldsymbol{X} = \bar{\boldsymbol{x}}]$, it follows that $\mathbb{E}[L]$ is a strictly increasing function in Ψ. In the following analysis, we assume that $\Psi \geq \lambda$ unless otherwise specified. Thus, the terminating list rank L ranges from 1 to λ.

Theorem 7. *For a given CRC-aided convolutional code decoded with SLVD,* $\lim_{\gamma_s\to 0} \mathbb{E}[L] = \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$.

Proof. We use the projection method to show the convergence of $\mathbb{E}[L]$ in the low SNR regime.

For ease of discussion, let $\mathcal{B}(\boldsymbol{a}, r)$ denote the *spherical surface* of center $\boldsymbol{a}$ and radius r in $\mathbb{R}^n$, where $\boldsymbol{a} \in \mathbb{R}^n$, $r \geq 0$. With BPSK modulation, all codewords sit on the *codeword sphere* $\mathcal{B}(\boldsymbol{O}, A\sqrt{n})$, whereas the received point $\boldsymbol{y}$ lies on the *noise sphere* $\mathcal{B}(\bar{\boldsymbol{x}}, w)$ for some noise vector with Euclidean norm w added to the transmitted point $\bar{\boldsymbol{x}}$. The projection method projects the received point $\boldsymbol{y}$ onto the codeword sphere. Namely, the projected point $\boldsymbol{y}_p$ of $\boldsymbol{y}$ is given by $\boldsymbol{y}_p = (A\sqrt{n}/\|\boldsymbol{y}\|)\boldsymbol{y}$. Fig. 2.5 illustrates the geometry of the projection method.

The significance of the projection method introduced above lies in the fact that it preserves the order of list decoded codewords. By law of cosines at angle θ in Fig. 2.5, we obtain

$$\|\boldsymbol{y}_p - \bar{\boldsymbol{x}}\| = \begin{cases} \sqrt{\frac{\|\boldsymbol{y}-\bar{\boldsymbol{x}}\|^2 - \|\boldsymbol{y}-\boldsymbol{y}_p\|^2}{1+\frac{\|\boldsymbol{y}-\boldsymbol{y}_p\|}{A\sqrt{n}}}}, & \text{if } \boldsymbol{y}_p \text{ in between } \boldsymbol{O}, \boldsymbol{y} \\ \sqrt{\frac{\|\boldsymbol{y}-\bar{\boldsymbol{x}}\|^2 - \|\boldsymbol{y}-\boldsymbol{y}_p\|^2}{1-\frac{\|\boldsymbol{y}-\boldsymbol{y}_p\|}{A\sqrt{n}}}}, & \text{otherwise.} \end{cases} \tag{2.66}$$

Hence, the monotone relation between $\|\boldsymbol{y}_p - \bar{\boldsymbol{x}}\|$ and $\|\boldsymbol{y} - \bar{\boldsymbol{x}}\|$ ensures that performing SLVD over $\boldsymbol{y}$ is equivalent to that over $\boldsymbol{y}_p$. The essential motivation of projecting points onto the codeword sphere is to transfer the computation from the noise sphere to the codeword sphere.

To see how the projection method helps to show the convergence of $\mathbb{E}[L]$, we first decompose the expected list rank $\mathbb{E}[L]$ according to the noise vector norm $W = w$. By linearity of the code,

$$\mathbb{E}[L] = \mathbb{E}[L|\boldsymbol{X} = \bar{\boldsymbol{x}}] = \int_0^\infty f_W(w)\mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}]\,\mathrm{d}w, \tag{2.67}$$

where $f_W(w)$ denotes the density function of norm $W = w$. To find $f_W(w)$, let

$$\phi_n(w) = \frac{1}{(\sqrt{2\pi})^n} e^{-w^2/2}, \tag{2.68}$$

$$S_{n-1}(w) = \frac{2\pi^{\frac{n}{2}}}{\Gamma(\frac{n}{2})} w^{n-1}, \tag{2.69}$$

be the n-dimensional standard normal density function and the spherical area of $\mathcal{B}(\bar{\boldsymbol{x}}, w)$ in $\mathbb{R}^n$, respectively. Then,

$$f_W(w) = \phi_n(w)S_{n-1}(w) = \frac{w^{n-1}}{2^{\frac{n-2}{2}}\Gamma(\frac{n}{2})} e^{-w^2/2}. \tag{2.70}$$

For a given norm $W = w$, it follows that

$$\mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}] = \frac{1}{S_{n-1}(w)} \int_{\boldsymbol{y} \in \mathcal{B}(\bar{\boldsymbol{x}}, w) \setminus \mathcal{N}} L(\boldsymbol{y})\,\mathrm{d}\boldsymbol{\sigma}, \tag{2.71}$$

where $\boldsymbol{\sigma}$ denotes the spherical measure on $\mathcal{B}(\bar{\boldsymbol{x}}, w)$. Using the projection method, the integral in (2.71) can be transferred to the codeword sphere at the cost of introducing an induced density function $g_w(\boldsymbol{y}_p)$. Namely,

$$\mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}] = \int_{\boldsymbol{y}_p \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n}) \setminus \mathcal{N}} L(\boldsymbol{y}_p) g_w(\boldsymbol{y}_p)\,\mathrm{d}\boldsymbol{\sigma}. \tag{2.72}$$

In the Appendix, the induced density function, for $w \geq A\sqrt{n}$, is given by

$$g_w(\boldsymbol{y}_p) = \left(\frac{\|\boldsymbol{y}(\boldsymbol{y}_p)\|}{w}\right)^{n-1} \frac{1}{\cos\angle\bar{\boldsymbol{x}}\boldsymbol{y}(\boldsymbol{y}_p)\boldsymbol{O}} \frac{1}{S_{n-1}(A\sqrt{n})}, \tag{2.73}$$

where $\boldsymbol{y}(\boldsymbol{y}_p)$ is the preimage of $\boldsymbol{y}_p$ on the noise sphere $\mathcal{B}(\bar{\boldsymbol{x}}, w)$. Note that $g_w(\boldsymbol{y}_p)$ is rotationally symmetric with respect to axis $\boldsymbol{O}\bar{\boldsymbol{x}}$. The Appendix also shows that

$$g_w(\boldsymbol{y}_p) \geq \frac{1}{S_{n-1}(A\sqrt{n})}\left(1 - \frac{A\sqrt{n}}{w}\right)^{n-1}, \tag{2.74}$$

$$g_w(\boldsymbol{y}_p) \leq \frac{1}{S_{n-1}(A\sqrt{n})}\left(1 + \frac{A\sqrt{n}}{w}\right)^{n-1}. \tag{2.75}$$

This implies that for a fixed norm w,

$$\lim_{A\to 0} \frac{g_w(\boldsymbol{y}_p)}{(S_{n-1}(A\sqrt{n}))^{-1}} = 1. \tag{2.76}$$

Hence, for a fixed norm w, it follows that

$$\begin{aligned}
\lim_{A\to 0} \mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}] &= \lim_{A\to 0} \int_{\boldsymbol{y}_p \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})\setminus\mathcal{N}} L(\boldsymbol{y}_p) g_w(\boldsymbol{y}_p)\, \mathrm{d}\boldsymbol{\sigma} && (2.77)\\
&= \lim_{A\to 0} \int_{\boldsymbol{y}_p \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})\setminus\mathcal{N}} L(\boldsymbol{y}_p) \frac{1}{S_{n-1}(A\sqrt{n})}\, \mathrm{d}\boldsymbol{\sigma} && (2.78)\\
&= \lim_{A\to 0} \mathbb{E}[L|W = A\sqrt{n}, \boldsymbol{X} = \boldsymbol{O}] && (2.79)\\
&= \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}], && (2.80)
\end{aligned}$$

where we have used the fact that $\mathbb{E}[L|W = w, \boldsymbol{X} = \boldsymbol{O}] = \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$ for all $w > 0$. Similarly, we can also show that, for a fixed amplitude A,

$$\lim_{w\to\infty} \mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}] = \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]. \tag{2.81}$$

As a consequence,

$$\begin{aligned}
\lim_{\gamma_s\to 0} \mathbb{E}[L] &= \lim_{A\to 0} \int_0^\infty f_W(w) \mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}]\, \mathrm{d}w\\
&= \int_0^\infty f(w) \lim_{A\to 0} \mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}]\, \mathrm{d}w\\
&= \int_0^\infty f(w) \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]\, \mathrm{d}w
\end{aligned}$$

$$= \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]. \tag{2.82}$$

This completes the proof. □

In general, $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$ depends on the geometric structure of the lower-rate code $\mathcal{C}_l$ and the higher-rate code $\mathcal{C}_h$ on $\mathcal{B}(\boldsymbol{O}, A\sqrt{n})$, and it is not easy to obtain an analytic expression. Still, using a simple random coding argument, we show that a good concatenated code could achieve $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}] \leq 2^m$.

Theorem 8. *For a given higher-rate code $\mathcal{C}_h$ with $|\mathcal{C}_h| = 2^{k+m}$, let $\mathcal{A}_l \triangleq \{\mathcal{C}' \subset \mathcal{C}_h : |\mathcal{C}'| = 2^k\}$. Let $\mathscr{P}(\mathcal{C}') = \frac{1}{|\mathcal{A}_l|}$ be the uniform distribution defined over $\mathcal{A}_l$. Assume $\mathcal{C}'$ is drawn according to $\mathscr{P}(\mathcal{C}')$. Then,*

$$\mathbb{E}_{\mathcal{C}'}\big[\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}, \mathcal{C}']\big] \leq 2^m. \tag{2.83}$$

Thus, there exists a lower-rate code $\mathcal{C}'$ (which may not be a linear code) such that $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}, \mathcal{C}'] \leq 2^m$.

Proof. Let $L(\boldsymbol{y}, \mathcal{C}')$ be the terminating list rank for received point $\boldsymbol{y} \in \mathbb{R}^n$ when the lower-rate code $\mathcal{C}' \in \mathcal{A}_l$ is selected. During the SLVD over $\boldsymbol{y}$ using code $\mathcal{C}'$, if there exist two codewords $\boldsymbol{c}_{j_1}$ and $\boldsymbol{c}_{j_2}$ that are equidistant from $\boldsymbol{y}$, we assume that the decoder adopts a predetermined order relation between $\boldsymbol{c}_{j_1}$ and $\boldsymbol{c}_{j_2}$. Hence, we obtain

$$\begin{aligned}
\mathbb{E}_{\mathcal{C}'}\big[\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}, \mathcal{C}']\big] &= \sum_{\mathcal{C}' \in \mathcal{A}_l} \mathscr{P}(\mathcal{C}') \frac{1}{S_{n-1}(A\sqrt{n})} \int_{\boldsymbol{y} \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})} L(\boldsymbol{y}, \mathcal{C}') \, \mathrm{d}\boldsymbol{\sigma} \\
&= \frac{1}{S_{n-1}(A\sqrt{n})} \int_{\boldsymbol{y} \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})} \sum_{\mathcal{C}' \in \mathcal{A}_l} \mathscr{P}(\mathcal{C}') L(\boldsymbol{y}, \mathcal{C}') \, \mathrm{d}\boldsymbol{\sigma} \\
&= \frac{1}{S_{n-1}(A\sqrt{n})} \int_{\boldsymbol{y} \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})} \mathbb{E}_{\mathcal{C}'}[L(\boldsymbol{y}, \mathcal{C}')|\boldsymbol{y}] \, \mathrm{d}\boldsymbol{\sigma}.
\end{aligned} \tag{2.84}$$

Next, we show that for any $\boldsymbol{y} \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})$,

$$\mathbb{E}_{\mathcal{C}'}[L(\boldsymbol{y}, \mathcal{C}')|\boldsymbol{y}] \leq 2^m \tag{2.85}$$

for $\mathcal{C}'$ uniformly drawn from $\mathcal{A}_l$. Fix a $\boldsymbol{y} \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})$ and let $\boldsymbol{c}_1(\boldsymbol{y}), \boldsymbol{c}_2(\boldsymbol{y}), \ldots, \boldsymbol{c}_{|\mathcal{C}_h|}(\boldsymbol{y})$ be an enumeration of $\mathcal{C}_h$ such that

$$\|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_1(\boldsymbol{y}))\| \leq \cdots \leq \|\boldsymbol{y} - \boldsymbol{x}(\boldsymbol{c}_{|\mathcal{C}_h|}(\boldsymbol{y}))\|.$$

Hence, the terminating list rank $L(\boldsymbol{y}, \mathcal{C}')$ of $\boldsymbol{y}$ is given by

$$L(\boldsymbol{y}, \mathcal{C}') = \min\left\{s \in \{1, 2, \ldots, |\mathcal{C}_h|\} : \boldsymbol{c}_s(\boldsymbol{y}) \in \mathcal{C}'\right\}. \tag{2.86}$$

For $\mathcal{C}'$ uniformly drawn in $\mathcal{A}_l$, computing $\mathbb{E}_{\mathcal{C}'}[L(\boldsymbol{y}, \mathcal{C}')|\boldsymbol{y}]$ is equivalent to solving the following problem: there are $|\mathcal{C}_h|$ balls in a basket, among which $|\mathcal{C}'|$ of them are red and the rest are white. Balls are picked up $|\mathcal{C}_h|$ times without replacement, and the time at which the first red ball emerges is marked as the terminating list rank. Since every ordering of ball picking is equiprobable and is bijective with $\mathcal{A}_l$, the expected list rank in ball picking problem is equal to $\mathbb{E}_{\mathcal{C}'}[L(\boldsymbol{y}, \mathcal{C}')|\boldsymbol{y}]$. Hence,

$$\begin{aligned}\mathbb{E}_{\mathcal{C}'}[L(\boldsymbol{y}, \mathcal{C}')|\boldsymbol{y}] &= \sum_{s=1}^{|\mathcal{C}_h|-|\mathcal{C}'|+1} s \frac{\binom{|\mathcal{C}_h|-s}{|\mathcal{C}'|-1}}{\binom{|\mathcal{C}_h|}{|\mathcal{C}'|}} && (2.87)\\ &= \frac{|\mathcal{C}_h|+1}{|\mathcal{C}'|+1} && (2.88)\\ &\leq 2^m,\end{aligned}$$

where (3.116) follows from a variant of the Chu-Vandermonde identity.

Finally, substituting (2.85) into (2.84) proves Theorem 8. □

In (2.67), it is shown that $\mathbb{E}[L]$ can be fully characterized by its conditional expectation $\mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}]$. For a given w and A, let $\bar{\boldsymbol{x}}_e = \bar{\boldsymbol{x}}/A$ be the transmitted point with unit amplitude per dimension. Then it can be shown that

$$\mathbb{E}[L|W = w, \boldsymbol{X} = \bar{\boldsymbol{x}}] = \mathbb{E}[L|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e], \tag{2.89}$$

where $\eta \triangleq w/A$ is called the *normalized norm*. Hence, it suffices to compute $\mathbb{E}[L|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e]$. The SNR (equivalently, the BPSK amplitude A) only exhibits a scaling effect.

To evaluate $\mathbb{E}[L|W=\eta, \boldsymbol{X}=\bar{\boldsymbol{x}}_e]$, let $\mathcal{C}_l^- \triangleq \mathcal{C}_l \setminus \{\bar{\boldsymbol{c}}\}$ and define the conditional probability of UE conditioned on the sphere $\mathcal{B}(\bar{\boldsymbol{x}}_e, \eta)$ as

$$P_{e,\lambda}(\eta) \triangleq \sum_{\boldsymbol{c}\in\mathcal{C}_l^-} \mathscr{P}(\mathcal{Y}(\boldsymbol{c})|W=\eta, \boldsymbol{X}=\bar{\boldsymbol{x}}_e). \tag{2.90}$$

In general, it is difficult to know the conditional probability of UE $P_{e,\lambda}(\eta)$. Assuming the knowledge of parametric information $P_{e,\lambda}(\eta)$, we first show an approximation that represents $\mathbb{E}[L|W=\eta, \boldsymbol{X}=\bar{\boldsymbol{x}}_e]$ as a linear combination between $L=1$ and $L=\mathbb{E}[L|\boldsymbol{X}=\boldsymbol{O}]$ with coefficient given by $P_{e,\lambda}(\eta)$.

Approximation 3 (Parametric approximation)**.** *Define $\bar{L} \triangleq \mathbb{E}[L|\boldsymbol{X}=\boldsymbol{O}]$. For a CRC-aided convolutional code with corresponding parameters $\bar{L}$ and $P_{e,\lambda}(\eta)$,*

$$\mathbb{E}[L|W=\eta, \boldsymbol{X}=\bar{\boldsymbol{x}}_e] \approx 1 - P_{e,\lambda}(\eta) + P_{e,\lambda}(\eta)\bar{L}. \tag{2.91}$$

Furthermore, averaging over $W=\eta$ on both sides of (2.91) *yields the parametric approximation of $\mathbb{E}[L]$, i.e.,*

$$\mathbb{E}[L] \approx 1 - P_{e,\lambda} + P_{e,\lambda}\bar{L}. \tag{2.92}$$

Proof. [Justification] For ease of discussion, we use the shorthand notation $\mathscr{P}(\cdot|\eta, \bar{\boldsymbol{x}}_e) \triangleq \mathscr{P}(\cdot|W=\eta, \boldsymbol{X}=\bar{\boldsymbol{x}}_e)$ and $\mathscr{P}(\cdot|\boldsymbol{O}) = \mathscr{P}(\cdot|\boldsymbol{X}=\boldsymbol{O})$. Let us consider η for which $P_{e,\lambda}(\eta) > 0$. Hence,

$$\begin{aligned}
\mathbb{E}[L|W=\eta, \boldsymbol{X}=\bar{\boldsymbol{x}}_e] &= \sum_{s=1}^{\lambda} s\mathscr{P}(L=s|\eta, \bar{\boldsymbol{x}}_e) \\
&= \mathscr{P}(\mathcal{Y}(\bar{\boldsymbol{c}})|\eta, \bar{\boldsymbol{x}}_e) + \sum_{s=1}^{\lambda} s\mathscr{P}(L=s|\eta, \bar{\boldsymbol{x}}_e) - \sum_{s=1}^{\lambda} \mathscr{P}(\mathcal{Z}_s(\bar{\boldsymbol{c}})|\eta, \bar{\boldsymbol{x}}_e) \\
&\geq 1 - P_{e,\lambda}(\eta) + \sum_{s=1}^{\lambda} s\big(\mathscr{P}(L=s|\eta, \bar{\boldsymbol{x}}_e) - \mathscr{P}(\mathcal{Z}_s(\bar{\boldsymbol{c}})|\eta, \bar{\boldsymbol{x}}_e)\big) \\
&= 1 - P_{e,\lambda}(\eta) + P_{e,\lambda}(\eta)\left(\sum_{s=1}^{\lambda} s\frac{\sum_{\boldsymbol{c}\in\mathcal{C}_l^-} \mathscr{P}(\mathcal{Z}_s(\boldsymbol{c})|\eta, \bar{\boldsymbol{x}}_e)}{\sum_{\boldsymbol{c}\in\mathcal{C}_l^-} \mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\eta, \bar{\boldsymbol{x}}_e)}\right)
\end{aligned} \tag{2.93}$$

$$\begin{aligned}&\approx 1 - P_{e,\lambda}(\eta) + P_{e,\lambda}(\eta)\left(\sum_{s=1}^{\lambda} s\mathscr{P}(L = s|\boldsymbol{O})\right) && (2.94)\\ &= 1 - P_{e,\lambda}(\eta) + P_{e,\lambda}(\eta)\bar{L},\end{aligned}$$

where (2.94) follows from the substitution below. Consider the conditional list rank distribution

$$\boldsymbol{P}_\eta = \left(\frac{\sum_{\boldsymbol{c}\in\mathcal{C}_l^-}\mathscr{P}(\mathcal{Z}_1(\boldsymbol{c})|\eta,\bar{\boldsymbol{x}}_e)}{\sum_{\boldsymbol{c}\in\mathcal{C}_l^-}\mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\eta,\bar{\boldsymbol{x}}_e)},\ldots,\frac{\sum_{\boldsymbol{c}\in\mathcal{C}_l^-}\mathscr{P}(\mathcal{Z}_\lambda(\boldsymbol{c})|\eta,\bar{\boldsymbol{x}}_e)}{\sum_{\boldsymbol{c}\in\mathcal{C}_l^-}\mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\eta,\bar{\boldsymbol{x}}_e)}\right). \tag{2.95}$$

Using the fact that $\lim_{\eta\to\infty} g_\eta(\boldsymbol{y}_p) = 1/S_{n-1}(\sqrt{n})$, as $\eta \to \infty$, the conditional list rank distribution $\boldsymbol{P}_\eta$ converges to

$$\begin{aligned}\boldsymbol{P}_\infty &= \left(\frac{\mathscr{P}(\mathcal{Z}_1(\boldsymbol{c})|\boldsymbol{O})}{\mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\boldsymbol{O})},\ldots,\frac{\mathscr{P}(\mathcal{Z}_\lambda(\boldsymbol{c})|\boldsymbol{O})}{\mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\boldsymbol{O})}\right) && (2.96)\\ &= \left(\frac{\sum_{\boldsymbol{c}\in\mathcal{C}_l}\mathscr{P}(\mathcal{Z}_1(\boldsymbol{c})|\boldsymbol{O})}{\sum_{\boldsymbol{c}\in\mathcal{C}_l}\mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\boldsymbol{O})},\ldots,\frac{\sum_{\boldsymbol{c}\in\mathcal{C}_l}\mathscr{P}(\mathcal{Z}_\lambda(\boldsymbol{c})|\boldsymbol{O})}{\sum_{\boldsymbol{c}\in\mathcal{C}_l}\mathscr{P}(\mathcal{Y}(\boldsymbol{c})|\boldsymbol{O})}\right)\\ &= (\mathscr{P}(L = 1|\boldsymbol{O}),\ldots,\mathscr{P}(L = \lambda|\boldsymbol{O})), && (2.97)\end{aligned}$$

where $\boldsymbol{c}$ is any lower-rate codeword in (2.96). Hence, we directly replace $\boldsymbol{P}_\eta$ with the limit distribution $\boldsymbol{P}_\infty$ in (3.121). Finally, averaging over $W = \eta$ on both sides of (2.91) yields (2.92). □

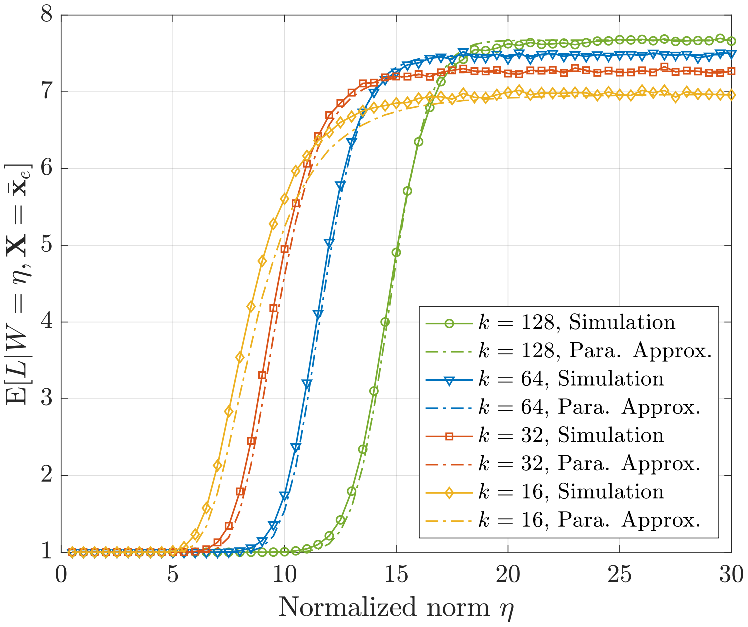

Figure 2.6: The conditional expected list rank $\mathbb{E}[L|W=\eta,\boldsymbol{X}=\bar{\boldsymbol{x}}_e]$ vs. the normalized norm η for the CRC-ZTCC generated with the degree-3 DSO CRC polynomial 0x9 and ZTCC $(13,17)$.

Fig. 2.6 shows the simulation results of the conditional expected list rank $\mathbb{E}[L|W=\eta,\boldsymbol{X}=\bar{\boldsymbol{x}}_e]$ vs. the normalized norm η for CRC-ZTCCs with various information lengths. The corresponding parametric approximation is also provided, where $\bar{L}=\mathbb{E}[L|\boldsymbol{X}=\boldsymbol{O}]$ is obtained from simulation. We see that the parametric approximation exhibits a remarkable accuracy that improves as k increases. Observe that for large values of k, the convergent $\mathbb{E}[L|W=\eta,\boldsymbol{X}=\bar{\boldsymbol{x}}_e]$ at sufficiently large η is close to 2^m.

Using (2.67) and (2.89), we can produce $\mathbb{E}[L]$ as a function of SNR γ_s. Fig. 2.7 shows $\mathbb{E}[L]$ vs. SNR along with its parametric approximations for ZTCC $(13,17)$ and various DSO CRC polynomials of degree $m=3,4,\ldots,6$, where $\bar{L}=\mathbb{E}[L|\boldsymbol{X}=\boldsymbol{O}]$ is obtained from simulation. We see that the parametric approximation on $\mathbb{E}[L]$ remains extremely tight.

The parametric approximation provides a practically useful quantitative connection between performance and complexity. Specifically, for CRC-ZTCCs with a target probabil-

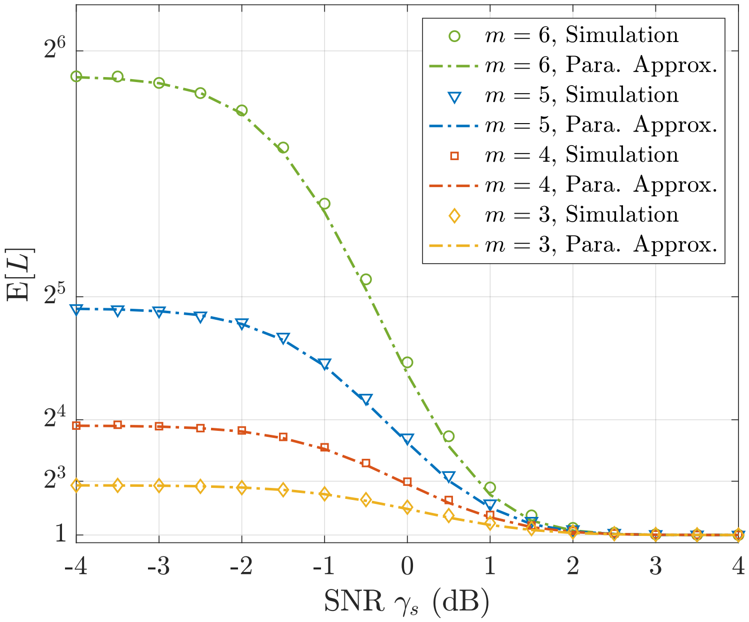

Figure 2.7: The expected list rank $\mathbb{E}[L]$ vs. SNR for various CRC-ZTCCs, where ZTCC is $(13, 17)$ and the DSO CRC polynomials are from Table 2.2 with degree $m = 3, 4, \ldots, 6$. The information length $k = 64$.

ity of UE $P^*_{e,\lambda}$ and $\bar{L} \approx 2^m$ for CRC degree m, (2.92) implies that a CRC with degree $m \leq -\log(P^*_{e,\lambda})$ is sufficient to maintain $\mathbb{E}[L] \leq 2$, which ensures that the average complexity for SLVD to achieve $P^*_{e,\lambda}$ is at most one more traceback than the standard Viterbi decoding.

As an alternative to Approximation 3, we provide a higher-order approximation formula for a good CRC-aided convolutional code that only requires the knowledge of $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$. This alternative approximation is motivated by Shannon's result [Sha59] that an optimal (n, M) code places its codewords on the surface of a sphere such that the total solid angle Ω_0 is evenly divided among the M Voronoi regions, one for each codeword, and that each Voronoi region is a circular cone. Hence, if the CRC-aided convolutional code is good enough, the union of order-1 to order-μ decoding regions $\mathcal{Z}_s(\boldsymbol{c})$ for a lower-rate codeword $\boldsymbol{c} \in \mathcal{C}_l$ should resemble circular cones, where μ is a parameter to be optimized. From this perspective, we propose the *onion model* for the order-1 to the order-μ decoding regions based on the following assumptions.

1. The union $\bigcup_{i=1}^{s} \mathcal{Z}_i(\boldsymbol{c})$ of order-1 to order-s decoding regions, $1 \le s \le \mu$, is a circular cone with half-angle α_s. This implies that each order-s decoding region, $2 \le s \le \mu$ is an *annulus* in between two circular cones.

2. The solid angle $\Omega(\alpha_s)$ of $\bigcup_{i=1}^{s} \mathcal{Z}_i(\boldsymbol{c})$ is equal to $\frac{s}{2^{k+m}}\Omega_0$, $1 \le s \le \mu$, where Ω_0 is the total solid angle (i.e., the area of a unit sphere in $\mathbb{R}^n$).

3. The conditional expected list rank beyond $\bigcup_{i=1}^{\mu} \mathcal{Z}_i(\bar{\boldsymbol{c}})$ is equal to $\bar{L}$ (i.e., $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$).

Approximation 4 (Higher-order approximation)**.** *For a given CRC-aided convolutional code, let* $\bar{L} = \mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$*. With the onion model assumptions and parameter* $\mu \in \mathbb{Z}^+$*,* $\mathbb{E}[L|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e]$ *is approximated by*

$$\mathbb{E}[L|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e] \approx \begin{cases} 1, \quad \textit{if } \eta < \sqrt{n}\sin\alpha_1 \\ \dots \\ s - \sum_{l=1}^{s-1} F_{\bar{\boldsymbol{x}}_e}(l), \ \textit{if } \sqrt{n}\sin\alpha_{s-1} \le \eta < \sqrt{n}\sin\alpha_s \\ \dots \\ \bar{L} - (\bar{L} - \mu)F_{\bar{\boldsymbol{x}}_e}(\mu) - \sum_{l=1}^{\mu-1} F_{\bar{\boldsymbol{x}}_e}(l), \ \textit{if } \eta \ge \sqrt{n}\sin\alpha_\mu, \end{cases} \tag{2.98}$$

where assuming $\eta \ge \sqrt{n}\sin\alpha_s$*,*

$$F_{\bar{\boldsymbol{x}}_e}(s) = \frac{\Gamma\left(\frac{n}{2}\right)}{\sqrt{\pi}\Gamma\left(\frac{n-1}{2}\right)} \left(\int_0^{\beta_{s,1}} \sin^{n-2}\theta \, \mathrm{d}\theta + \int_0^{\beta_{s,2}} \sin^{n-2}\theta \, \mathrm{d}\theta \right), \tag{2.99}$$

$$\beta_{s,1} = \frac{\pi}{2} + \alpha_s - \arcsin\left(\frac{\sqrt{\eta^2 - n\sin^2\alpha_s}}{\eta} \right), \tag{2.100}$$

$$\beta_{s,2} = \left(\frac{\pi}{2} - \alpha_s - \arcsin\left(\frac{\sqrt{\eta^2 - n\sin^2\alpha_s}}{\eta} \right) \right) \mathbf{1}_{\{\eta \le \sqrt{n}\}}, \tag{2.101}$$

and α_s *is the half-angle for which*

$$\frac{\Omega(\alpha_s)}{\Omega_0} = \frac{\Gamma\left(\frac{n}{2}\right)}{\sqrt{\pi}\Gamma\left(\frac{n-1}{2}\right)} \int_0^{\alpha_s} \sin^{n-2}\theta \, \mathrm{d}\theta = \frac{s}{2^{k+m}}. \tag{2.102}$$

Justification. The onion model assumptions imply that each higher order decoding region $\mathcal{Z}_s(\boldsymbol{c})$, $2 \leq s \leq \mu$, is an annulus in between two circular cones. Hence, $\mathscr{P}(L = s|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e)$ is simply the spherical area of $\mathcal{B}(\bar{\boldsymbol{x}}_e, \eta)$ cut out by the annulus. To evaluate this quantity, consider the cumulative probability function of $L = s$,

$$F_{\bar{\boldsymbol{x}}_e}(s) \triangleq \mathscr{P}(L \leq s, \boldsymbol{X} = \bar{\boldsymbol{x}}_e). \tag{2.103}$$

Thus,

$$\mathscr{P}(L = s|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e) = F_{\bar{\boldsymbol{x}}_e}(s) - F_{\bar{\boldsymbol{x}}_e}(s-1). \tag{2.104}$$

By the onion model assumptions, for $\eta \geq \sqrt{n}\sin\alpha_\mu$,

$$\begin{aligned}\mathbb{E}[L|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e] &\approx \sum_{l=1}^{\mu} l(F_{\bar{\boldsymbol{x}}_e}(l) - F_{\bar{\boldsymbol{x}}_e}(l-1)) + \bar{L}(1 - F_{\bar{\boldsymbol{x}}_e}(\mu)) &(2.105)\\ &= \bar{L} - (\bar{L} - \mu)F_{\bar{\boldsymbol{x}}_e}(\mu) - \sum_{l=1}^{\mu-1} F_{\bar{\boldsymbol{x}}_e}(l). &(2.106)\end{aligned}$$

In the similar fashion, for $\sqrt{n}\sin\alpha_{s-1} \leq \eta < \sqrt{n}\sin\alpha_s$, $1 \leq s \leq \mu$,

$$\mathbb{E}[L|W = \eta, \boldsymbol{X} = \bar{\boldsymbol{x}}_e] \approx s - \sum_{l=1}^{s-1} F_{\bar{\boldsymbol{x}}_e}(l). \tag{2.107}$$

Next, we derive the cumulative probability function $F_{\bar{\boldsymbol{x}}_e}(s)$. Geometrically, $F_{\bar{\boldsymbol{x}}_e}(s)$ is the fraction of the spherical area of $\mathcal{B}(\bar{\boldsymbol{x}}_e, \eta)$ cut out by the circular cone $\bigcup_{i=1}^{s} \mathcal{Z}_s(\bar{\boldsymbol{c}})$ with half-angle α_s to the total noise spherical area. Assume that $\sqrt{n}\sin\alpha_s \leq \eta \leq \sqrt{n}$. Fig. 2.8 shows the side view of this scenario in $\mathbb{R}^3$, in which the blue arc represents the spherical area contained in $\bigcup_{i=1}^{s} \mathcal{Z}_s(\bar{\boldsymbol{c}})$. It can be seen that α_s will induce two possible half-angles $\beta_{s,1}$ and $\beta_{s,2}$. By law of cosines,

$$\begin{aligned}\beta &= \frac{\pi}{2} \pm \alpha_s - \arcsin\left(\frac{r_2 - r_1}{2\eta}\right) &(2.108)\\ &= \frac{\pi}{2} \pm \alpha_s - \arcsin\left(\frac{\sqrt{\eta^2 - n\sin^2\alpha_s}}{\eta}\right), &(2.109)\end{aligned}$$

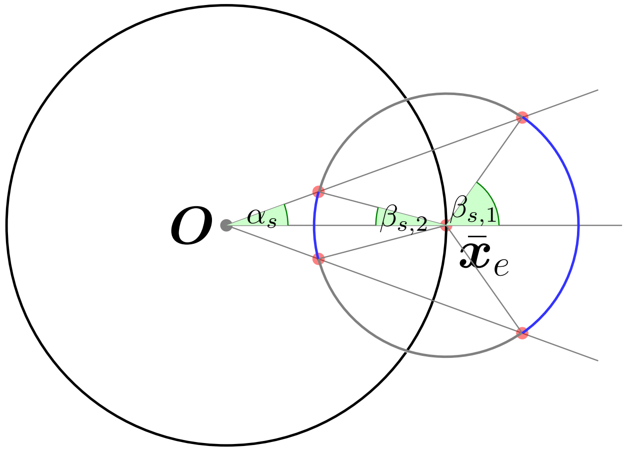

Figure 2.8: The geometry of the cumulative probability function $F_{\bar{\boldsymbol{x}}_e}(s)$, assuming that $\sqrt{n}\sin\alpha_s \leq \eta \leq \sqrt{n}$.

where r_1, r_2 are solutions to

$$r^2 - (2\sqrt{n}\cos\alpha_s)r + (n - \eta^2) = 0. \tag{2.110}$$

The induced half-angle β becomes unique once $\eta > \sqrt{n}$.

From [Sha59, Eq. (21)], the solid angle $\Omega(\alpha)$ of a circular cone with center $\boldsymbol{O}$ and half-angle α in n-dimensional Euclidean space is given by

$$\Omega(\alpha) = \frac{2\pi^{\frac{n-1}{2}}}{\Gamma\left(\frac{n-1}{2}\right)} \int_0^\alpha \sin^{n-2}\theta \, \mathrm{d}\theta. \tag{2.111}$$

The total solid angle Ω_0 in n-dimensional Euclidean space is given by

$$\Omega_0 = \frac{2\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}\right)}. \tag{2.112}$$

Thus, using (2.111), (2.112), we can solve α_s from assumption 2 of the onion model. Namely, α_s is the solution to

$$\frac{\Omega(\alpha)}{\Omega_0} = \frac{\Gamma\left(\frac{n}{2}\right)}{\sqrt{\pi}\Gamma\left(\frac{n-1}{2}\right)} \int_0^\alpha \sin^{n-2}\theta \, \mathrm{d}\theta = \frac{s}{2^{k+m}}. \tag{2.113}$$

By geometry in Fig. 2.8, $F_{\bar{\boldsymbol{x}}_e}(s)$ in (2.103) is given by

$$F_{\bar{\boldsymbol{x}}_e}(s) = \frac{\Omega(\beta_{s,1}) + \Omega(\beta_{s,2})}{\Omega_0}. \tag{2.114}$$

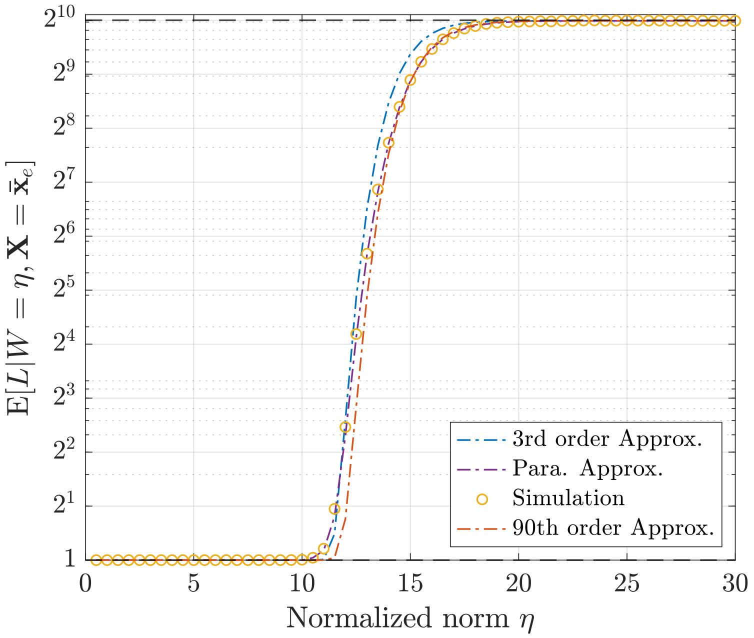

Figure 2.9: The higher-order and parametric approximations of $\mathbb{E}[L|W=\eta,\boldsymbol{X}=\bar{\boldsymbol{x}}_e]$ for ZTCC $(561,753)$ used with the degree-10 DSO CRC polynomial 0x4CF at $k=64$. Both the higher-order and parametric approximations assume the knowledge of $\bar{L}=1017$. The back, dashed line corresponds to 2^{10}.

This concludes the justification of Approximation 4. □

To demonstrate the tightness of Approximation 4 for suffi-ciently good CRC-aided convolutional codes, Fig. 2.9 shows approximations of $\mathbb{E}[L|W=\eta,\boldsymbol{X}=\bar{\boldsymbol{x}}_e]$ for ZTCC $(561,753)$ used with the degree-10 DSO CRC polynomial 0x4CF at $k=64$ with $\mu=3$ and 90. This concatenated code has a minimum distance $d^l_{\text{min}}=20$ and thus can be deemed good enough. When $\mu=3$, the third order approximation accurately gives the smaller values of the actual conditional expected list rank. As μ increases, the accuracy of the approximation shifts towards large values of conditional expected list rank. Fig. 2.10 illustrates approximations of $\mathbb{E}[L]$ vs. SNR via (2.67) and (2.89). The 3rd order and 90th order approximations still behave in the similar fashion as in Fig. 2.9.

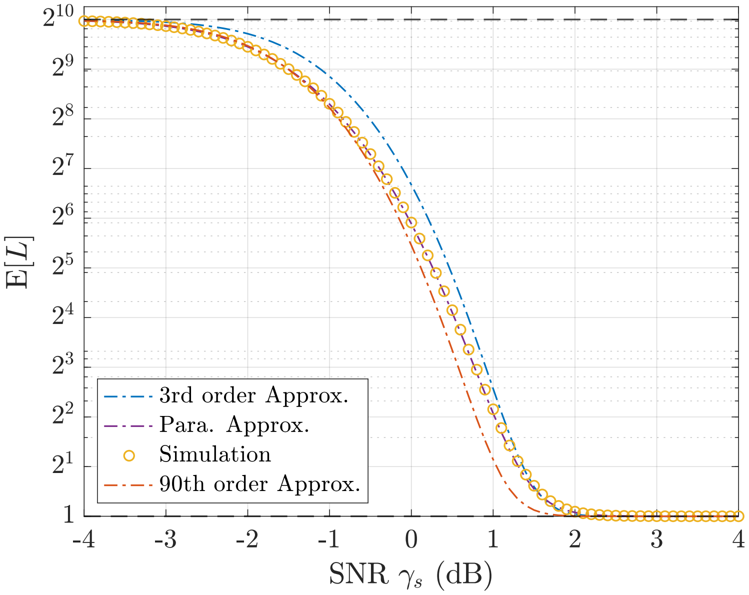

Figure 2.10: The expected list rank $\mathbb{E}[L]$ vs. SNR via (2.67) and (2.89) for ZTCC $(561, 753)$, degree-10 DSO CRC polynomial 0x4CF at $k = 64$. The back, dashed line corresponds to 2^{10}.

2.4.3 Complexity Analysis

There are a variety of implementations of list decoding of convolutional codes as described in, e.g., [BMK04,LCC04,RH06,KTK18]. In this chapter, the SLVD implementation maintains a list of path metric differences by using a red-black tree described in [RH06], which provides the fastest runtime we found among the data structures that support full floating-point precision. The literature mentioned above also analyzed the number of bit operations or the asymptotic complexity of the algorithms presented, but those complexity metrics are not directly connected with the actual runtime. To compare the complexity of SLVD of a CRC-aided convolutional code with that of the standard soft Viterbi (SSV) decoding, we develop an average complexity expression that closely approximates our empirical runtime.

For our specific implementation, three components comprise the average complexity of

SLVD, given by

$$C_{\text{SLVD}} = C_{\text{SSV}} + C_{\text{trace}} + C_{\text{list}}. \tag{2.115}$$

The first component C_{SSV} is the complexity required to perform the add-compare-select (ACS) operation on the trellis of the given convolutional code and perform the initial traceback associated with SSV. Specifically, for CRC-ZTCCs, this quantity is given by

$$\begin{aligned} C_{\text{SSV}} = {} & (2^{\nu+1} - 2) + 1.5(2^{\nu+1} - 2) + 1.5(k + m - \nu)2^{\nu+1} \\ & + c_1[2(k + m + \nu) + 1.5(k + m)]. \end{aligned} \tag{2.116}$$

For CRC-TBCCs, this quantity is given by

$$C_{\text{SSV}} = 1.5(k + m)2^{\nu+1} + 2^{\nu} + 3.5c_1(k + m). \tag{2.117}$$

In order to measure the decoding complexity, define 1 unit of complexity as the complexity required to perform one addition. In (2.116) and (2.117), we assign 1 unit of complexity to each addition per branch and 0.5 units of complexity to each compare-select operation per branch. In the first and second terms of (2.116), $(2^{\nu+1} - 2)$ counts the number of edges in the initial ν sections and the final ν termination sections of a ZT trellis. In the third term of (2.116), $(k + m - \nu)2^{\nu+1}$ counts the number of edges in the middle $(k + m - \nu)$ sections of a ZT trellis. The fourth term in (2.116) approximates the complexity of the traceback operation, assigning 2 units of complexity for accessing the parent node per trellis stage and 1.5 units of complexity per codeword symbol for the CRC verification on the decoded sequence $\hat{\boldsymbol{v}}$. In (2.117), the second term stems from that it takes 2^{ν} operations to identify the optimal termination state with the minimum metric before the first traceback.

The second component C_{trace} represents the complexity of *additional* traceback operations required by SLVD. Specifically, for a given CRC-ZTCC,

$$C_{\text{trace}} = c_1(\mathbb{E}[L] - 1)[2(k + m + \nu) + 1.5(k + m)]. \tag{2.118}$$

For CRC-TBCCs,

$$C_{\text{trace}} = 3.5c_1(\mathbb{E}[L]-1)(k+m). \tag{2.119}$$

The third component C_{list} represents the average complexity of inserting new elements to maintain an ordered list of path metric differences. For both CRC-ZTCCs and CRC-TBCCs,

$$C_{\text{list}} = c_2\mathbb{E}[I]\log(\mathbb{E}[I]), \tag{2.120}$$

where $\mathbb{E}[I]$ is the expected number of insertions to maintain the sorted list of path metric differences. According to the mechanism of insertion, for CRC-ZTCCs,

$$\mathbb{E}[I] \leq (k+m)\mathbb{E}[L], \tag{2.121}$$

and for CRC-TBCCs,

$$\mathbb{E}[I] \leq (k+m)\mathbb{E}[L] + 2^{\nu} - 1, \tag{2.122}$$

where $2^{\nu}-1$ denotes the number of path metric differences between the optimal terminating state and any of the remaining $2^{\nu}-1$ terminating states.

In (2.116), (2.117), (2.118), (2.119), and (2.120), the constants c_1 and c_2 characterize implementation-specific differences in the implemented complexity of traceback and list insertion, respectively, as compared to the ACS operations of Viterbi decoding. For our implementation, we found $c_1 = 1.5$ and $c_2 = 2.2$.

The additional complexity of the SLVD over SSV decoding is completely characterized by the additional tracebacks along the trellis and the maintenance of an ordered list of path metric differences. We define the *normalized complexity* $\bar{C}_{\text{SLVD}}$ as the complexity of SLVD divided by the complexity of SSV decoding, i.e.,

$$\bar{C}_{\text{SLVD}} = \frac{C_{\text{SLVD}}}{C_{\text{SSV}}} = 1 + \bar{C}_{\text{trace}} + \bar{C}_{\text{list}}. \tag{2.123}$$

The normalized complexity provides a measure for the additional complexity of operations associated with the SLVD relative to the complexity of the SSV algorithm.

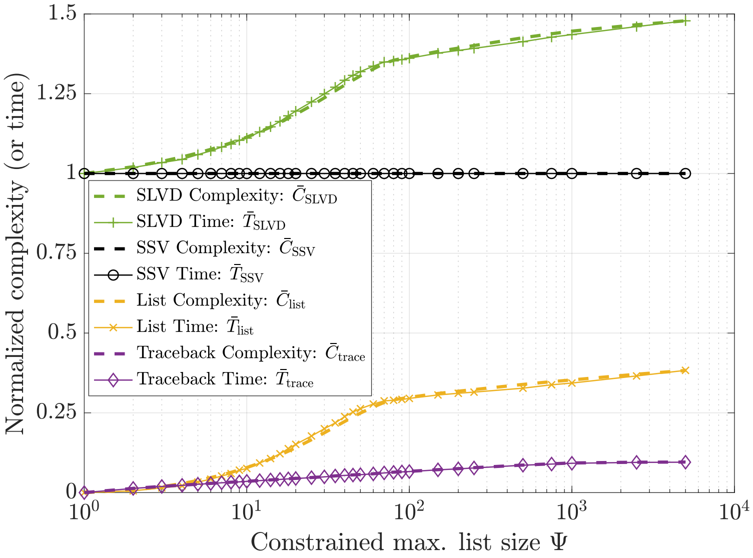

Figure 2.11: The complexity of SLVD with different constrained maximum list sizes for ZTCC $(27, 31)$ and degree-10 DSO CRC polynomial 0x709, with $k = 64$ at SNR $\gamma_s = 2$ dB. All variables are normalized by time or the complexity of SSV decoding. In the simulation, $c_1 = 1.5$ and $c_2 = 2.2$.

We recorded the runtime T_{SLVD}, T_{SSV}, T_{trace}, and T_{list} on the Intel Core i7-4720HQ using Visual C++. We then divided all of these terms by T_{SSV} to compute the normalized runtime $\bar{T}$. Fig. 2.11 shows the normalized complexity based on (2.123) and the normalized runtime. In both cases, the normalization is computed by dividing by the complexity or runtime associated with SSV, i.e., performing all ACS operations on the trellis and a single traceback from the state with the best metric. Fig. 2.11 shows that the normalized complexity and normalized runtime curves are indistinguishable. It also shows that the additional complexity of SLVD is primarily from maintaining an ordered list of path metric differences.

2.5 Simulation Results

In this section, we present our simulation results of CRC-ZTCCs in Table 2.2 and CRC-TBCCs in Table 2.3 for the binary-input AWGN channel at $k = 64$. Finally, we compare

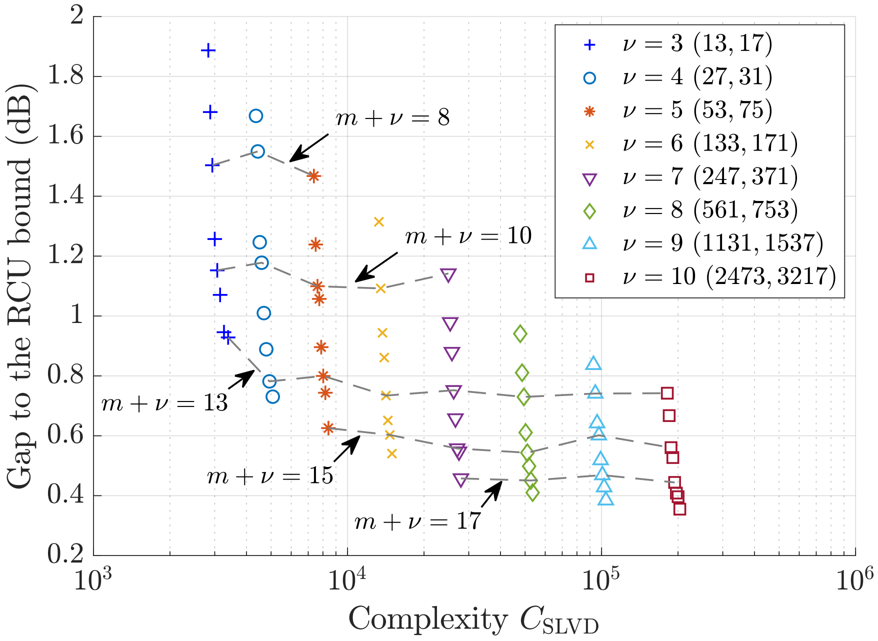

Figure 2.12: The SNR gap to the RCU bound vs. the average complexity of SLVD for the family of CRC-ZTCCs in Table 2.2 at target $P_{e,\lambda} = 10^{-4}$. Each color represents a specific ZTCC shown in parenthesis. Markers from top to bottom with the same color correspond to the DSO CRC polynomials with $m = 3, 4, \ldots, 10$ in Table 2.2. The information length and blocklength are given by $k = 64$ and $n = 2(64 + m + \nu)$, respectively.

the punctured CRC-TBCC with $k = 64$ and $n = 128$ designed in our precursor conference paper [LYD19] with several $(128, 64)$ linear block codes presented in [CDJ19].

2.5.1 Simulation Results for CRC-ZTCCs

Fig. 2.12 shows the trade-off between the SNR gap to the RCU bound and the average decoding complexity computed using (2.115) for target probability of UE $P_{e,\lambda} = 10^{-4}$. It is shown that for a given ZTCC, increasing the degree m of DSO CRC polynomials can significantly diminish the SNR gap to the RCU bound at a relatively small complexity increase. This SNR gap reduction is especially considerable when ν is small and becomes less significant as ν becomes large. For all ZTCCs, the complexity cost of increasing m from

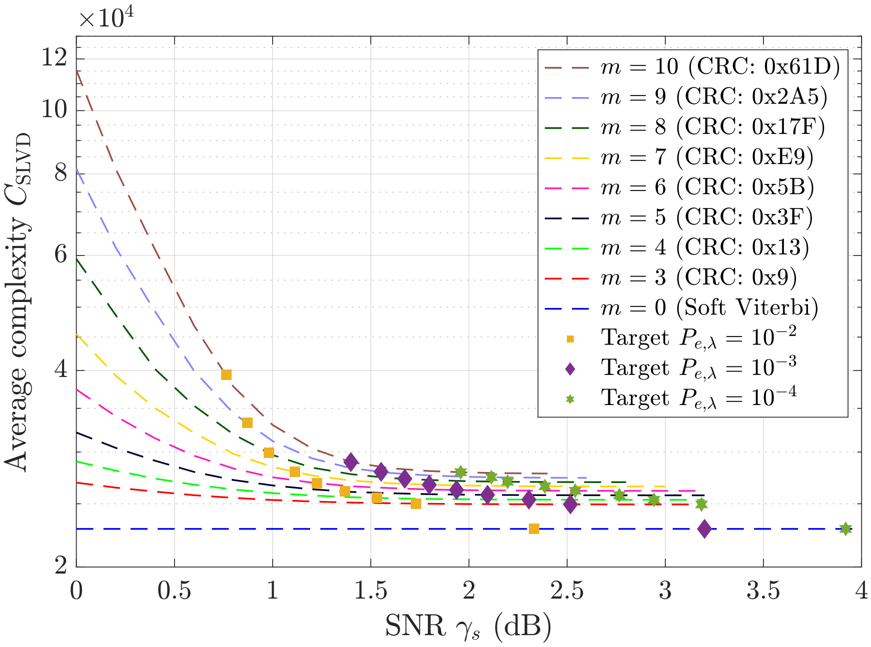

Figure 2.13: The average complexity vs. SNR for ZTCC $(247, 371)$ used with the corresponding DSO CRC polynomials. The ZTCC with no CRC using soft Viterbi decoding is also given as a reference.

3 to 10 is within a factor of 2. This is consistent with Fig. 2.11 in which the complexity increases by a factor less than 1.5 even for a very large constrained maximum list size Ψ.

A CRC-ZTCC could be decoded using the Viterbi algorithm alone, without list decoding, on a trellis with $2^{m+\nu}$ states per trellis stage. The dashed lines in Fig. 2.12 show that the gap to the RCU bound remains roughly constant for a constant value of $m + \nu$. However, list decoding with a well chosen (m, ν) pair achieves this performance with a minimum complexity C_{SLVD}. Thus, for a given target error probability $P_{e,\lambda}$ and a fixed value of $m+\nu$, the inclusion of CRC-aided list decoding will generally reduce complexity compared to Viterbi decoding alone on a convolutional code with $2^{m+\nu}$ states per trellis stage.

Fig. 2.13 shows the complexity C_{SLVD} computed using (2.115) as a function of SNR for ZTCC $(247, 371)$ and the corresponding DSO CRC polynomials with degree m from 3 to 10 from Table 2.2. The ZTCC using soft Viterbi decoding with no CRC is also shown. Here, the

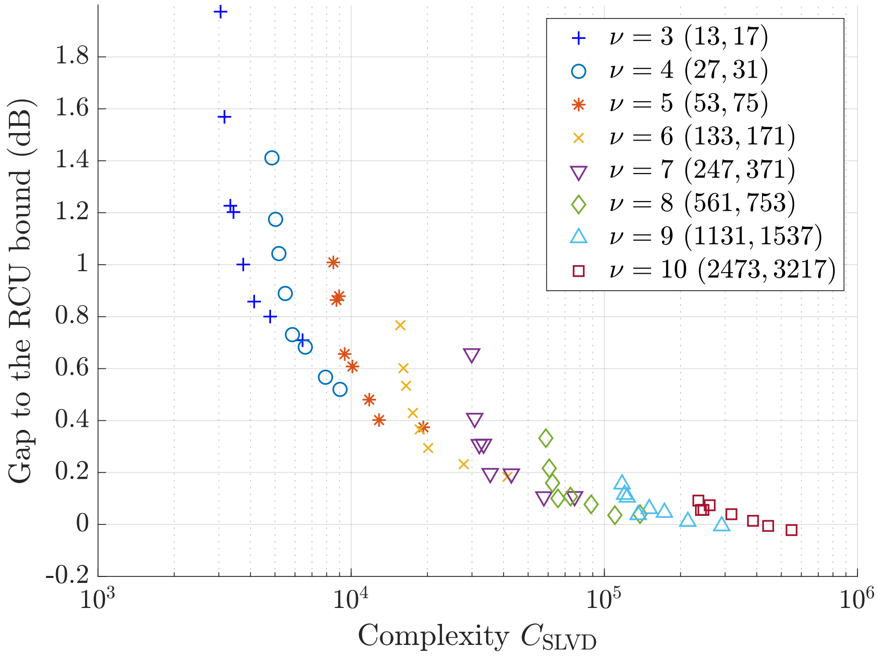

Figure 2.14: The SNR gap to the RCU bound vs. the average complexity of SLVD for the family of CRC-TBCCs in Table 2.3 at target $P_{e,\lambda} = 10^{-4}$. Each color represents a specific TBCC shown in parenthesis. Markers from top to bottom with the same color correspond to the DSO CRC polynomials with $m = 3, 4, \ldots, 10$ in Table 2.3. The information length and blocklength are given by $k = 64$ and $n = 2(64 + m)$, respectively.

target probabilities of UE at $10^{-2}, 10^{-3}, 10^{-4}$ for each CRC-ZTCC are marked by squares, diamonds, and stars, respectively. For each target probability of UE, the corresponding complexity is within a factor of 2 compared to the soft Viterbi decoding of ZTCC $(247, 371)$.

The termination overhead associated with the ZTCC induces a gap from the RCU bound, which can be closed by using the corresponding TBCC as we will see below.

2.5.2 Simulation Results for CRC-TBCCs

In Section 4.2 we use the fact that for a CRC-ZTCC, each traceback operation in SLVD yields a valid higher-rate codeword, i.e., a ZT convolutional codeword. However, for a CRC-

TBCC, traceback operations in SLVD do not always yield a valid higher-rate codeword, i.e., a TB convolutional codeword, because the TB condition is often not met. In view of this, we can no longer assume that $\bar{L} \approx 2^m$. Nevertheless, $\bar{L}$ can still be obtained from simulation and Approximations 3 and 4 still apply.

The increased value of $\bar{L}$ may be understood by considering the higher-rate code $\mathcal{C}_h$ to be the pseudo code represented by all paths on the trellis regardless of whether they meet the TB condition. Due to the additional complexity required to check the TB condition, $\mathbb{E}[I]$ is significantly increased compared to that for the CRC-ZTCC. While we identified the empirical value of $\mathbb{E}[I]$ for CRC-ZTCCs, in this section we simply assume $\mathbb{E}[I]$ attains the upper bound in (2.122) for CRC-TBCCs. Hence, using (2.117), (2.119) with $c_1 = 1.5$, (2.120) with $c_2 = 2.2$, together with the aforementioned assumption on $\mathbb{E}[I]$, we can compute an estimate of the average complexity C_{SLVD} of our implementation of SLVD of CRC-TBCCs.

Fig. 2.14 shows the SNR gap to the RCU bound vs. the average complexity for target probability of UE $P_{e,\lambda} = 10^{-4}$ for all CRC-TBCCs designed in Table 2.3. Compared to Fig. 2.12, TB encoding significantly reduces the SNR gap to the RCU bound, because the overhead of termination is avoided. However, this reduction of the SNR gap comes at the expense of a slight increase in average complexity for checking the TB condition. Note the exciting result that some CRC-TBCCs outperform the RCU bound for $\nu = 9$ and 10. Another phenomenon distinct from CRC-ZTCCs is that for TBCCs with large ν, increasing the DSO CRC polynomial degree from $m = 3$ to 10 only provides a small benefit. Note, however, that the degree-3 DSO CRC polynomial does provide a benefit over a TBCC used with no CRC at all.

Fig. 2.15 shows the trade-off comparison for CRC-TBCCs with $m = 10$ and ν from 3 to 10 at $P_{e,\lambda} = 10^{-4}$ and $P_{e,\lambda} = 10^{-5}$. We see that as $P_{e,\lambda}$ decreases, the SNR gap to the RCU bound is increased and the average complexity of SLVD is further reduced. However, the increase in SNR gap depends on ν. As ν increases, the increase in SNR gap is reduced. We see that at $m = 10$ and $\nu = 10$, this SNR gap increase becomes negligible.

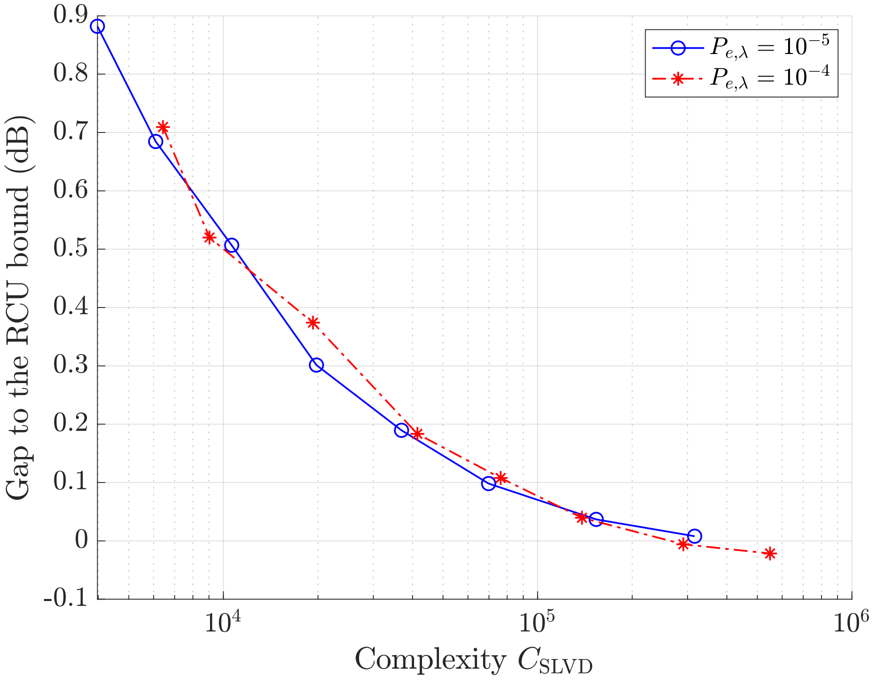

Figure 2.15: The SNR gap to the RCU bound vs. the average complexity of SLVD for the family of CRC-TBCCs in Table 2.3 with $m = 10$. For a fixed $P_{e,\lambda}$, the points from the left to the right correspond to ν from 3 to 10. In this example, $k = 64$ and $n = 148$.

To illustrate the performance of the best CRC-TBCCs designed in Table 2.3, we select $\nu = 9$ and $\nu = 10$ TBCCs as an example. Fig. 2.16 shows two cases: $R = 64/134$ corresponding to $m = 3$ and $R = 64/146$ corresponding to $m = 9$. The MC bound and the RCU bounds for these rates are plotted using the saddlepoint approximations provided in Approximations 1 and 2, respectively. We see that in these two cases, the CRC-TBCCs in Fig. 2.16 beat the RCU bound at low SNR values. However, this superiority gradually fades away as SNR increases, although for $m = 9$, the performance is very close to the RCU bound even at $P_{e,\lambda} = 10^{-5}$. Simulations also suggest that it is extremely difficult to further improve the code performance once beyond the RCU bound at low probability of UE.

Fig. 2.17 shows the family of CRC-TBCCs with $k = 64$ and $n = 148$ (corresponding to $m = 10$). For small ν, we see a visible improvement as ν increases. However, once perfor-

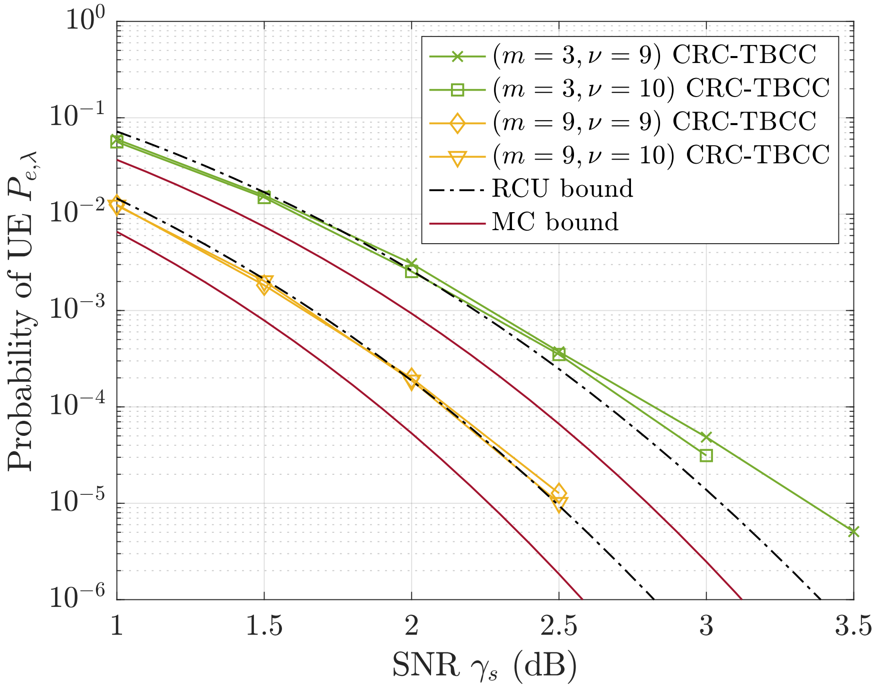

Figure 2.16: Comparison of $P_{e,\lambda}$ with RCU and MC bounds at rates $R = 64/134$ ($m = 3$) and $R = 64/146$ ($m = 9$) for the CRC-TBCCs designed in Table 2.3. For the sake of clarity, only $\nu = 9, 10$ TBCCs are displayed.

mance reaches the RCU bound, further increases in ν provide little benefit. For example, with $m = 10$, the CRC-TBCC with $\nu = 9$ attains similar performance to that with $\nu = 10$.

2.5.3 Comparison of $(128, 64)$ Linear Block Codes

Direct comparison of CRC-TBCCs with other codes often requires puncturing to match rates. For simplicity, we have excluded puncturing from analysis in this chapter. However, our precursor conference paper [LYD19] designed a $v = 8$, $m = 10$ punctured CRC-TBCC with $k = 64$ and $n = 128$ whose frame error rate (FER) performance can be directly compared to the $(128, 64)$ linear block codes presented in [CDJ19], as shown in Fig. 2.18.

At SNR of 3 dB, the $v = 8$, $m = 10$ punctured CRC-TBCC in [LYD19] and the best codes studied in [CDJ19] all perform similarly. Specifically, the four codes in [CDJ19] with similar

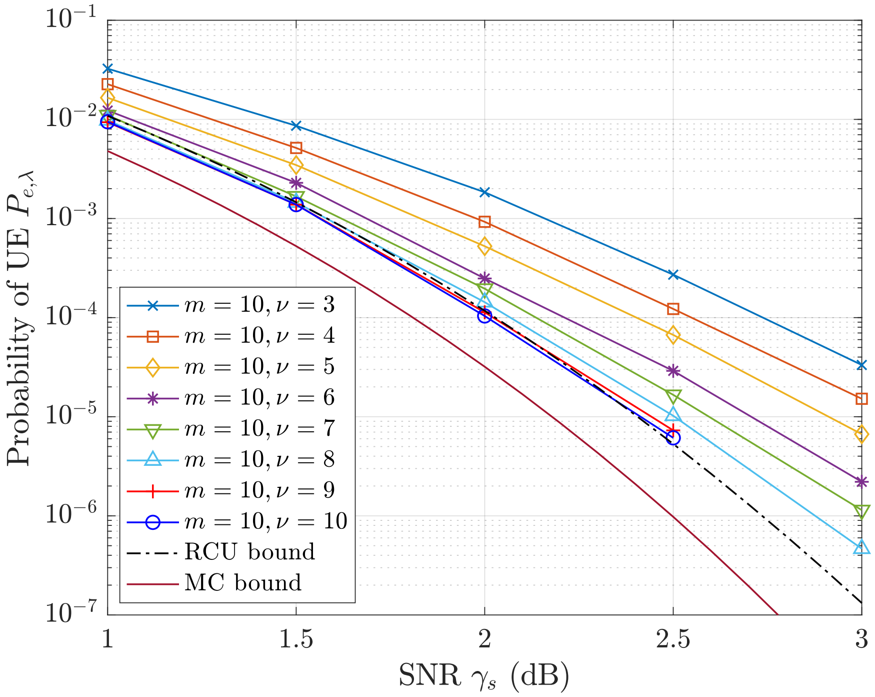

Figure 2.17: Comparison of $P_{e,\lambda}$ with RCU and MC bounds at rate $R = 64/148$ (i.e., $m = 10$) for the CRC-TBCCs designed in Table 2.3.

performance at 3 dB to the $v = 8$, $m = 10$ punctured CRC-TBCC are the following: the $v = 14$ and $v = 11$ TBCCs decoded with WAVA, the extended BCH code with order-4 OSD, and a non-binary LDPC code over $\mathbb{F}_{256}$ with order-4 OSD. As shown in Fig. 2.18, at higher SNR, the FER performance is more differentiated with the best performance provided by the $v = 14$ TBCC, slightly worse performance provided by the $v = 8$, $m = 10$ punctured CRC-TBCC and the extended BCH code with order-4 OSD and further degraded performance by the $v = 11$ TBCC and the non-binary LDPC code over $\mathbb{F}_{256}$ with order-4 OSD.

We now consider the decoding complexity of the three best codes described above at 3 dB, excluding the discussion of the non-binary LDPC code due to its further degraded performance. Actual complexity depends on specific implementation choices, here we consider the total number of computations per codeword as a way to give some flavor of the complexity differences between these approaches. At SNR of 3 dB, simulation shows that

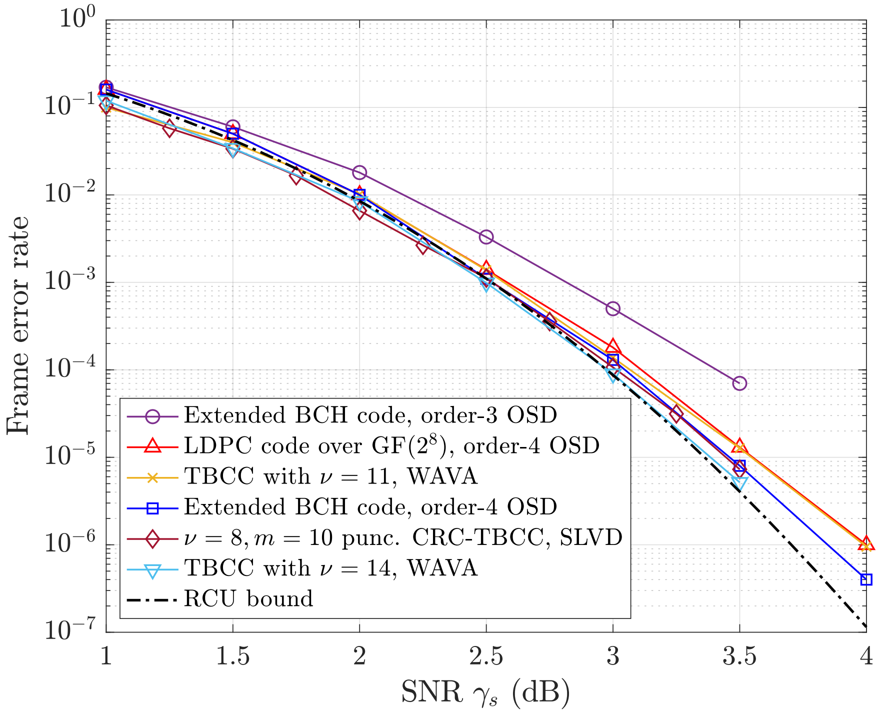

Figure 2.18: Comparison of $(128, 64)$ linear block codes.

$\mathbb{E}[L] = 44.41$ for the $v = 8$, $m = 10$ punctured CRC-TBCC. Using (2.117), (2.119), (2.120) together with (2.122), we obtain $C_{\text{SLVD}} \leq 1.67 \times 10^5$.

In terms of WAVA complexity, let I be the number of iterations in WAVA. By assuming 0.5 units of complexity for compare-select operation per branch and 1 unit of complexity for one addition, the WAVA complexity for a rate-$1/\omega$ TBCC with ν memory elements at information length k is given by

$$C_{\text{WAVA}} = kI(0.5 \cdot 2^{\nu} + 2^{\nu+1}). \tag{2.124}$$

Using (2.124), the complexity of 3-round WAVA for $v = 11$ TBCC in [CDJ19] is 9.83×10^5, which is higher than for the $v = 8, m = 10$ punctured CRC-TBCC. The best $v = 14$ TBCC in [CDJ19] under 3-round WAVA achieves a complexity of 7.86×10^6.

A direct complexity comparison of SLVD with OSD is more difficult, but Table V in [FS95] indicates that at 3 dB, the order-3 OSD of the $(128, 64)$ extended BCH code requires 2.83×10^5

operations per codeword on average, which indicates that the order-4 OSD would likely have a higher complexity than the SLVD of $v = 8$, $m = 10$ punctured CRC-TBCC. Based on this analysis, the CRC-TBCC paradigm appears to be competitive with the existing approaches that provide similarly excellent FER performance at short blocklength.

2.6 Conclusion

In this chapter, we consider the CRC-aided convolutional code as a promising good short blocklength code. The concatenated nature of the code permits the use of SLVD that allows the code to attain the ML decoding performance at low complexity. For $k = 64$ and the binary-input AWGN channel, we identified the DSO CRC polynomial for a family of ZTCCs and TBCCs generated with the optimum rate-1/2 convolutional encoders in [LC04] at sufficiently low target probability of UE. Several CRC-TBCCs beat the RCU bound at practically interesting values of SNR. In a recent work [Sch21], Schiavone confirmed that the CRC-TBCC is indeed a powerful short blocklength code by showing that its performance matches the expurgated ensemble.

All CRC-aided convolutional codes considered in this chapter are designed based on an optimum convolutional encoder. It would be interesting to investigate whether a suboptimal convolutional code used with the DSO CRC polynomial can also lead to a good concatenated code. Another interesting direction is to explore the performance of CRC-aided convolutional codes in the moderately short blocklength regime, e.g., $128 \leq k < 1000$. If puncturing is introduced in the code design, the problem of how to jointly design the puncturing pattern and the optimal CRC polynomial for a given convolutional code still remains open.

The beauty of SLVD lies in the fact that the average complexity is governed by the expected list rank $\mathbb{E}[L]$, a quantity that is inversely proportional to the SNR value. This allows a huge complexity reduction at operating SNRs of interest that guarantee a low target probability of UE. In particular, the parametric approximation of $\mathbb{E}[L]$ provides an explicit characterization of the performance-complexity trade-off. It shows that for CRC-ZTCCs with a target error probability $P^*_{e,\lambda}$ and $\bar{L} \approx 2^m$, a CRC degree $m \leq -\log(P^*_{e,\lambda})$ is sufficient to maintain $\mathbb{E}[L] \leq 2$. In closing, several theoretical problems are still open, for instance, how to develop tight bounds on $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$ and $P_{e,1}$ using only the weight spectrum. In addition, the behavior of the supremum list rank λ is also less understood and is worth future investigation.

Acknowledgment

Chapter 2 is largely a reprint of [YLP22] except Fig. 2.15 and its discussion. An earlier version of this work was presented in part at the 2018 and 2019 IEEE Global Communications Conferences [YRW18, LYD19] and the 2020 IEEE International Symposium on Information Theory [YWL20].

2.7 Appendix: Derivation of the Induced Density Function

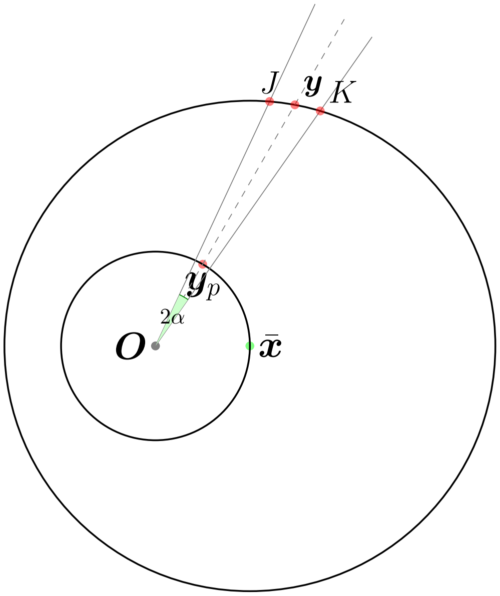

Figure 2.19: Derivation of the induced density function $g_w(\boldsymbol{y}_p)$ in $\mathbb{R}^n$.

Let $\mathcal{B}(\boldsymbol{a}, r)$ denote the spherical surface of center $\boldsymbol{a}$ and radius r in $\mathbb{R}^n$, where $\boldsymbol{a} \in \mathbb{R}^n$ and $r \geq 0$. In this section, we derive the induced density function $g_w(\boldsymbol{y}_p)$ incurred when projecting a received point $\boldsymbol{y}$ uniformly distributed on $\mathcal{B}(\bar{\boldsymbol{x}}, w)$ to point $\boldsymbol{y}_p = (A\sqrt{n}/\|\boldsymbol{y}\|)\boldsymbol{y}$ that lies on the codeword sphere $\mathcal{B}(\boldsymbol{O}, A\sqrt{n})$ in $\mathbb{R}^n$. As an illustration, Fig. 2.19 depicts this scenario in $\mathbb{R}^2$. For our purposes, we assume that $w \geq A\sqrt{n}$ to ensure the bijective relationship between $\boldsymbol{y}$ and $\boldsymbol{y}_p$.

Let us consider a circular cone Q_α in $\mathbb{R}^n$ with apex at the origin $\boldsymbol{O}$, axis along $\boldsymbol{O}\boldsymbol{y}_p$, and half-angle α. Algebraically, define the direction vectors

$$\boldsymbol{y}_e \triangleq \frac{\boldsymbol{y}}{\|\boldsymbol{y}\|}, \tag{2.125}$$

$$\boldsymbol{z}_e \triangleq \frac{\boldsymbol{y} - \bar{\boldsymbol{x}}}{\|\boldsymbol{y} - \bar{\boldsymbol{x}}\|}. \tag{2.126}$$

Hence, the circular cone Q_α is given by

$$\begin{aligned} Q_\alpha &= \left\{ \boldsymbol{r} \in \mathbb{R}^n : \frac{\boldsymbol{r}^\top \boldsymbol{y}_e}{\|\boldsymbol{r}\|} \geq \cos\alpha \right\} \\ &= \left\{ \boldsymbol{r} \in \mathbb{R}^n : (\boldsymbol{r} - \boldsymbol{0})^\top (I - \epsilon^2(\alpha) \boldsymbol{y}_e \boldsymbol{y}_e^\top)(\boldsymbol{r} - \boldsymbol{0}) \leq 0 \right\}, \end{aligned} \tag{2.127}$$

where $\epsilon(\alpha) \triangleq 1/\cos\alpha$ denotes the eccentricity of the cone. Cone Q_α intersects with the noise sphere $\mathcal{B}(\bar{\boldsymbol{x}}, w)$, thus producing a surface area $Q_\alpha \cap \mathcal{B}(\bar{\boldsymbol{x}}, w)$ delimited by J and K on Fig. 2.19. Thus, the induced density at $\boldsymbol{y}_p$ is given by

$$g_w(\boldsymbol{y}_p) = \lim_{\alpha \to 0} \frac{S(Q_\alpha \cap \mathcal{B}(\bar{\boldsymbol{x}}, w))/S_{n-1}(w)}{S(Q_\alpha \cap \mathcal{B}(\boldsymbol{O}, A\sqrt{n}))}, \tag{2.128}$$

where $S(\cdot)$ denotes the surface area in $\mathbb{R}^n$. Note that for sufficiently small α, the spherical surface around $\boldsymbol{y}$ is equivalent to the tangent hyperplane at $\boldsymbol{y}$, given by

$$\begin{aligned} H &= \left\{ \boldsymbol{r} \in \mathbb{R}^n : \boldsymbol{z}_e^\top (\boldsymbol{r} - \boldsymbol{y}) = 0 \right\} \\ &= \left\{ \boldsymbol{r} \in \mathbb{R}^n : \boldsymbol{z}_e^\top (\boldsymbol{r} - \boldsymbol{0}) = \hat{h} \right\}, \end{aligned} \tag{2.129}$$

where $\hat{h} \triangleq \boldsymbol{z}_e^\top \boldsymbol{y}$. Define $\rho \triangleq \sqrt{1 - (\boldsymbol{z}_e^\top \boldsymbol{y}_e)^2}$. Thus, using the result by Dearing [Dea, Eq. (15)], if $\epsilon(\alpha)\rho < 1$, the intersection of hyperplane H and cone Q_α is an ellipsoid of dimension $(n-1)$, which, after proper rotation T around $\boldsymbol{O}$, can be written as

$$T(Q_\alpha) \cap T(H) = \left\{ (r_1, \ldots, r_{n-1}, \hat{h}) : \frac{(r_1 - \hat{c}_1)^2}{\hat{a}^2} + \frac{\sum_{j=2}^{n-1} (r_j - \hat{c}_j)^2}{\tilde{b}} = 1 \right\},$$

where

$$\sigma = \boldsymbol{z}_e^\top \boldsymbol{y}_e, \tag{2.130}$$

$$\hat{c}_1 = \frac{\epsilon^2(\alpha)\rho\sigma\hat{h}}{1 - \epsilon^2(\alpha)\rho^2}, \quad \hat{c}_j = 0, \; j = 2, \ldots, n-1, \tag{2.131}$$

$$\hat{a}^2 = \frac{(\epsilon^2(\alpha) - 1)\hat{h}^2}{(1 - \epsilon^2(\alpha)\rho^2)^2}, \tag{2.132}$$

$$\tilde{b} = \hat{a}^2(1 - \epsilon^2(\alpha)\rho^2). \tag{2.133}$$

Since $\boldsymbol{z}_e$ and $\boldsymbol{y}_e$ are non-orthogonal, $1/\rho > 1$. Hence, for sufficiently small α, $\epsilon(\alpha) < 1/\rho$ and thus Dearing's result follows. Summarizing the analysis above, we obtain

$$\begin{aligned}
\lim_{\alpha\to 0} S\big(Q_\alpha \cap \mathcal{B}(\bar{\boldsymbol{x}}, w)\big) &= \lim_{\alpha\to 0} S\big(T(Q_\alpha) \cap T(H)\big) \qquad (2.134)\\
&= \lim_{\alpha\to 0} \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n+1}{2})} \hat{a} \Big(\sqrt{\tilde{b}}\Big)^{n-2}\\
&= \lim_{\alpha\to 0} \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n+1}{2})} \left(\frac{(\epsilon^2(\alpha)-1)\hat{h}^2}{(1-\epsilon^2(\alpha)\rho^2)^2}\right)^{\frac{n-1}{2}} \big(1-\epsilon^2(\alpha)\rho^2\big)^{\frac{n-2}{2}}\\
&= \lim_{\alpha\to 0} \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n+1}{2})} 2^{\frac{n-1}{2}} (\epsilon(\alpha)-1)^{\frac{n-1}{2}} \hat{h}^{n-1} (\boldsymbol{z}_e^\top \boldsymbol{y}_e)^{-n}\\
&= \lim_{\alpha\to 0} \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n+1}{2})} 2^{\frac{n-1}{2}} \Big(\frac{1-\cos\alpha}{\cos\alpha}\Big)^{\frac{n-1}{2}} \Big(\frac{\boldsymbol{z}_e^\top \boldsymbol{y}}{\boldsymbol{z}_e^\top \boldsymbol{y}_e}\Big)^{n-1} \frac{1}{\boldsymbol{z}_e^\top \boldsymbol{y}_e}\\
&= \lim_{\alpha\to 0} \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n+1}{2})} 2^{\frac{n-1}{2}} \Big(2\sin^2\Big(\frac{\alpha}{2}\Big)\Big)^{\frac{n-1}{2}} \frac{\|\boldsymbol{y}\|^{n-1}}{\cos\angle\bar{\boldsymbol{x}}\boldsymbol{y}\boldsymbol{O}}\\
&= \lim_{\alpha\to 0} \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n+1}{2})} \alpha^{n-1} \frac{\|\boldsymbol{y}\|^{n-1}}{\cos\angle\bar{\boldsymbol{x}}\boldsymbol{y}\boldsymbol{O}}, \qquad (2.135)
\end{aligned}$$

where (2.134) follows since for sufficiently small half-angle, the spherical surface around $\boldsymbol{y}$ is equivalent to that of the tangent hyperplane H at $\boldsymbol{y}$. From [Sha59, Eq. (21)], the area of the spherical cap $S(Q_\alpha \cap \mathcal{B}(\boldsymbol{O}, A\sqrt{n}))$ is given by

$$S(Q_\alpha \cap \mathcal{B}(\boldsymbol{O}, A\sqrt{n})) = \frac{(n-1)\pi^{\frac{n-1}{2}}(A\sqrt{n})^{n-1}}{\Gamma\big(\frac{n+1}{2}\big)} \int_0^\alpha \sin^{n-2}\theta \, \mathrm{d}\theta. \qquad (2.136)$$

Substituting (3.180), (3.181) into (3.119), we obtain

$$\begin{aligned}
g_w(\boldsymbol{y}_p) =& \lim_{\alpha\to 0} \frac{S(Q_\alpha \cap \mathcal{B}(\bar{\boldsymbol{x}}, w))}{S(Q_\alpha \cap \mathcal{B}(\boldsymbol{O}, A\sqrt{n}))} \frac{S_{n-1}(A\sqrt{n})}{S_{n-1}(w) S_{n-1}(A\sqrt{n})}\\
=& \lim_{\alpha\to 0} \frac{\alpha^{n-1} \frac{\|\boldsymbol{y}(\boldsymbol{y}_p)\|^{n-1}}{\cos\angle\bar{\boldsymbol{x}}\boldsymbol{y}(\boldsymbol{y}_p)\boldsymbol{O}}}{(n-1)\int_0^\alpha \theta^{n-2}\,\mathrm{d}\theta} \frac{1}{w^{n-1}} \frac{1}{S_{n-1}(A\sqrt{n})}\\
=& \Big(\frac{\|\boldsymbol{y}(\boldsymbol{y}_p)\|}{w}\Big)^{n-1} \frac{1}{\cos\angle\bar{\boldsymbol{x}}\boldsymbol{y}(\boldsymbol{y}_p)\boldsymbol{O}} \frac{1}{S_{n-1}(A\sqrt{n})}, \qquad (2.137)
\end{aligned}$$

where $\boldsymbol{y}(\boldsymbol{y}_p)$ is the preimage of $\boldsymbol{y}_p$ on the noise sphere $\mathcal{B}(\bar{\boldsymbol{x}}, w)$. Here, (3.182) is the induced density function of $\boldsymbol{y}_p \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})$. Observe that it is rotationally symmetric with respect to axis $\boldsymbol{O}\bar{\boldsymbol{x}}$.

Next, we develop an alternative expression for $g_w(\boldsymbol{y}_p)$ to derive its upper and lower bounds. First, we rotate the coordinate system such that axis $\boldsymbol{O\bar{x}}$ is the first coordinate and the remaining $(n-1)$ coordinates are orthogonal to $\boldsymbol{O\bar{x}}$. In the new coordinate system, let $\bar{\boldsymbol{x}} = (A\sqrt{n}, 0, \ldots, 0) \in \mathbb{R}^n$. Hence, for an arbitrary projected point $\boldsymbol{y}_p = (y_1, y_2, \ldots, y_n) \in \mathcal{B}(\boldsymbol{O}, A\sqrt{n})$, assume that $\rho \triangleq \|\boldsymbol{y}(\boldsymbol{y}_p)\|$. Thus,

$$\boldsymbol{y}(\boldsymbol{y}_p) = \frac{\rho}{A\sqrt{n}}(y_1, y_2, \ldots, y_n). \tag{2.138}$$

Since $\boldsymbol{y}(\boldsymbol{y}_p) \in \mathcal{B}(\bar{\boldsymbol{x}}, w)$,

$$\left(\frac{\rho}{A\sqrt{n}}y_1 - A\sqrt{n}\right)^2 + \left(\frac{\rho}{A\sqrt{n}}\right)^2 \sum_{i=2}^{n} y_i^2 = w^2. \tag{2.139}$$

Solving for ρ yields

$$\rho = y_1 + \sqrt{y_1^2 + w^2 - A^2 n}. \tag{2.140}$$

By law of cosines, it is shown that

$$\cos\angle\bar{\boldsymbol{x}}\boldsymbol{y}\boldsymbol{O} = \frac{\rho^2 + w^2 - A^2 n}{2\rho w} = \frac{\sqrt{y_1^2 + w^2 - A^2 n}}{w}. \tag{2.141}$$

Hence, substituting (3.185) and (3.186) into (3.182) and expressing $g_w(\boldsymbol{y}_p)$ in terms of $y_1 \in [-A\sqrt{n}, A\sqrt{n}]$, we obtain

$$g_w(y_1) = \frac{1}{S_{n-1}(A\sqrt{n})} \frac{(y_1 + \sqrt{y_1^2 + w^2 - A^2 n})^{n-2}}{w^{n-2}} \left(1 + \frac{y_1}{\sqrt{y_1^2 + w^2 - A^2 n}}\right). \tag{2.142}$$

Clearly, $g_w(y_1)$ is monotonically increasing in y_1. Hence,

$$g_w(y_1) \geq g_w(-A\sqrt{n}) = \frac{1}{S_{n-1}(A\sqrt{n})}\left(1 - \frac{A\sqrt{n}}{w}\right)^{n-1}, \tag{2.143}$$

$$g_w(y_1) \leq g_w(A\sqrt{n}) = \frac{1}{S_{n-1}(A\sqrt{n})}\left(1 + \frac{A\sqrt{n}}{w}\right)^{n-1}. \tag{2.144}$$

Geometrically, this implies that the maximum induced density is attained at the transmitted point $\bar{\boldsymbol{x}}$, whereas the minimum induced density is attained at $-\bar{\boldsymbol{x}}$.

CHAPTER 3

Binary Asymmetric Channels With Full Feedback

3.1 Introduction

Feedback does not increase the capacity of memoryless channels [Sha56], but it significantly reduces communication complexity and probability of error, provided that variable-length feedback (VLF) codes are allowed. For a discrete memoryless channel (DMC) with full noiseless feedback, Burnashev [Bur76] proposed a pioneering two-phase transmission scheme that produces the *exact* optimal error exponent of VLF codes for all rates below capacity. The first phase is called the *communication phase*, during which the transmitter seeks to increase receiver's posterior probability for the transmitted message. The system transitions from the communication phase to the *confirmation phase* when the largest posterior probability at the receiver exceeds a certain threshold. During the confirmation phase, two most distinguishable input symbols are used: one for the message with the largest posterior probability, and the other for the rest of messages. The confirmation phase continues until either the transmission terminates or the system returns to the communication phase. This two-phase coding scheme allows Burnashev to obtain a VLF achievability bound that coincides asymptotically with the VLF converse bound, thus producing the optimal error exponent. However, Burnashev did not provide an explicit non-asymptotic VLF achievability bound for the DMC.

For the binary symmetric channel (BSC) with noiseless feedback, Horstein [Hor63] developed a simple, one-phase scheme that maps each message to a subinterval in $[0, 1]$. The transmitter sends a 0 if the subinterval of the true message lies entirely beneath the median and a 1 if it lies entirely above the median. If the subinterval includes the median point, which will eventually happen as the subinterval of the highest posterior probability grows, then randomized encoding is employed. Horstein did not provide a rigorous proof to show that his scheme achieves capacity. In [BZ74], Burnashev and Zigangirov showed that Horstein's scheme achieves the capacity of the BSC in the fixed blocklength setting. In [SF11], Shayevitz and Feder generalized Horstein's scheme to the concept of posterior matching, thus validating the capacity-achieving property of Horstein's scheme. Since Horstein's work, several authors, e.g., [Sch71, SP73, TT02, TT06], have constructed coding schemes for the BSC with noiseless feedback under various assumptions in order to attain capacity or Burnashev's optimal error exponent.

In the non-asymptotic regime, Polyanskiy *et al.* [PPV11] showed that for a target average blocklength l and error probability ϵ, the maximal message size $M^*(l, \epsilon)$ for the VLF code is significantly improved compared to that of fixed-length codes. Polyanskiy *et al.* demonstrated this benefit by establishing a non-asymptotic achievability bound for the *variable-length stop-feedback (VLSF)* code, a VLF code that makes a very limited use of feedback [PPV11, Theorem 3]. More specifically, all VLSF codewords are designed and fixed before transmission. A stop-feedback symbol is only used to inform the encoder of when to terminate transmission and does not affect the value of the transmitted codeword. A compelling example of the advantage offered by variable-length coding can be seen for the BSC with capacity 1/2 and target error probability 10^{-3}. With variable-length coding and stop feedback, the average blocklength required to achieve 90% of capacity is less than 200, compared to at least 3100 for the best fixed-blocklength code with noiseless feedback.

In [NWJ12], Naghshvar *et al.* asked the question of whether having two separate phases of operations and randomized encoding are necessary to achieve Burnashev's optimal error exponent. As a negative response to this question, they presented a deterministic, one-phase coding scheme that achieves the optimal error exponent for any symmetric binary-input channels (including the BSC) with full noiseless feedback. The most appealing feature of their scheme is that at each time instant, the common codebook available at the encoder and decoder is generated “on the fly” by a bi-partitioning of the message set such that the probability difference of the two subsets is *small enough* (see Sec. IV in [NWJ12]), and this is sufficient for their scheme to achieve both capacity and the optimal error exponent. Since the authors did not provide a name for their scheme, here we term their scheme as the *small-enough-difference (SED)* coding scheme[1]. In a subsequent work [NJW15], Naghshvar *et al.* applied the extrinsic Jensen-Shannon (EJS) divergence and submartingale synthesis technique to develop a non-asymptotic VLF achievability bound for the deterministic VLF code constructed with the SED coding scheme operated over a symmetric binary-input channel. Recently, Guo *et al.* [GK21] developed an instantaneous SED code for the symmetric binary-input channel with feedback for real-time communication.

While Naghshvar *et al.* obtained a non-asymptotic VLF achievability bound for the symmetric binary-input channel, this bound appears to be inferior to Polyanskiy's non-asymptotic achievability bound for the VLSF code. In general, a system that employs full noiseless feedback, such as the SED coding scheme, should achieve a rate much better than that of a VLSF code. This indicates that there is an opportunity to develop a tighter achievability bound that outperforms Polyanskiy's VLSF achievability bound. On the other hand, despite the simplicity of the SED coding scheme for the symmetric binary-input channel, the extension of this scheme to a general binary-input channel with feedback still remains unknown, let alone a general multi-input channel with feedback.

As its primary contributions, this chapter extends Naghshvar *et al.*'s SED coding scheme to the binary asymmetric channel (BAC) with feedback, including the BSC as a special case. In general, a BAC has binary input and output alphabets and is specified by two crossover probabilities: $p_0 \triangleq P(Y=1|X=0)$ and $p_1 \triangleq P(Y=0|X=1)$ that are allowed to be equal. Therefore, the word “asymmetric” does not exclude the symmetric case. In [MCL10, Section 2.3], it is argued that every BAC can be equivalently transformed into a *regularized BAC*

[1] We first coined this name in our conference chapter [YWL20].

satisfying $0 < p_0 < 1/2$ and $p_0 \leq p_1 \leq 1 - p_0$ by flipping either the input or the output, or both. Therefore, our generalized SED coding scheme is mainly for the regularized BAC. In the definition of a regularized BAC, we exclude $p_0 = 0$ and $p_0 = p_1 = 1/2$ cases, in order to guarantee that important quantities C_1 and C_2 defined in (3.7) and (3.8) are always positive and finite. Nonetheless, it has been known that in the degenerate case where $p_0 = 0$, zero-error VLF capacity is equal to the conventional capacity; see [Bur76, Sec. 6] or [PPV11, Eq. 133].

Unlike Naghshvar *et al.*'s one-phase SED coding scheme, our SED coding scheme for the regularized BAC is a deterministic, two-phase coding scheme. More specifically, assume that (π_0^*, π_1^*) is the capacity-achieving input distribution for a regularized BAC. In the communication phase where all posterior probabilities are less than π_1^*, the message set is partitioned into two subsets with probabilities π_0 and π_1 such that the difference $\frac{\pi_0}{\pi_0^*} - \frac{\pi_1}{\pi_1^*}$ is small enough. In the confirmation phase where the largest posterior probability among all messages exceeds π_1^*, we exclusively assign input symbol 1 to the message with the largest posterior probability and 0 to all remaining messages.

Using the generalized SED coding scheme, we develop a non-asymptotic VLF achievability bound that outperforms Polyanskiy's VLSF achievability bound for a regularized BAC. In particular, for the BSC with feedback, we develop a refined non-asymptotic VLF achievability bound. Numerical evaluations show that for BSC with capacity 1/2 and error probability 10^{-3}, both VLF achievability bounds exceed Polyanskiy *et al.*'s VLSF achievability bound, which is expected since a system with full noiseless feedback should perform better than the system that only employs stop feedback.

In our analysis, the technique for obtaining a VLF achievability bound for a regularized BAC involves a submartingale synthesis with optimal parameters and a variant of Doob's optional stopping theorem. For the specific case of the BSC, the confirmation phase can be modeled as a Markov chain with possible fallbacks to the communication phase. This facilitates a decomposition of the random process concerning the transmitted message into two components: a submartingale describing the first communication phase and a generalized Markov chain that describes the subsequent behavior (see Section 3.5.6). This decomposition allows a separate upper bound on the average blocklength to be computed for each of the two components. The upper bound for the first component is obtained using a surrogate submartingale construction and a variant of Doob's optional stopping theorem. The upper bound for the second component is obtained using time of first-passage analysis on a generalized Markov chain. Finally, the sum of the two upper bounds yields an upper bound on the overall average blocklength that turns out to be tighter than the bound developed using a pure submartingale synthesis when the crossover probability is small.

The remainder of this chapter is organized as follows. In Section 3.2, we introduce basic notation, the regularized BAC and some useful facts, a VLF code for a memoryless channel, and Naghshvar *et al.*'s SED coding scheme for symmetric binary-input channels. Next, we review some important previous results. In Section 3.3, we present the generalized SED coding scheme for a regularized BAC with full noiseless feedback and a non-asymptotic VLF achievability bound for a regularized BAC. Section 3.4 presents a refined VLF achievability bound for the BSC with full noiseless feedback. Section 3.5 contains proofs of our main results. In Section 3.6, we numerically compare our VLF achievability bounds with the simulated performance of the SED coding scheme and with some previously known results. In Section 3.7, we show that for a regularized BAC, our generalized SED coding scheme achieves both capacity and the optimal error exponent. Section 4.5 concludes the chapter.

3.2 Preliminaries

3.2.1 Notation and Definitions

Let $\mathbb{N} = \{0, 1, 2, \dots\}$ denote the set of natural numbers, and $\mathbb{N}_+ = \mathbb{N} \setminus \{0\}$ denote the set of positive integers. Let $[M] \triangleq \{1, 2, \dots, M\}$. We denote by $\log(\cdot), \ln(\cdot)$ the base-2 and the natural logarithms, respectively. $h(p) \triangleq -p\log(p) - (1-p)\log(1-p)$, $p \in [0, 1]$, denotes the binary entropy function. Let P_Y, Q_Y be two distributions over a finite alphabet $\mathcal{Y}$, the *Kullback-Leibler (KL) divergence* between P_Y and Q_Y is defined as $D(P_Y \| Q_Y) \triangleq \sum_{y\in\mathcal{Y}} P_Y(y) \log \frac{P_Y(y)}{Q_Y(y)}$ with the convention that $0 \log \frac{0}{a} = 0$ and $b \log \frac{b}{0} = \infty$ for $a, b \in [0,1]$ with $b \neq 0$. Let $[x]^+ = \max\{0, x\}$. We denote the collection of all subsets of $\mathcal{X}$ by $2^{\mathcal{X}}$.

3.2.2 The Regularized BAC and Some Useful Facts

A BAC consists of binary input and output alphabets, i.e., $\mathcal{X} = \mathcal{Y} = \{0,1\}$, and two crossover probabilities, $p_0 \triangleq P_{Y|X}(1|0) \in [0,1]$ and $p_1 \triangleq P_{Y|X}(0|1) \in [0,1]$. As noted in [MCL10], it suffices to restrict our attention to the regularized case where $p_0 \in [0, 1/2]$ and $p_0 \leq p_1 \leq 1 - p_0$, as any other case can be transformed into this case by swapping either the input or the output, or both. In this chapter, we say that a BAC(p_0, p_1) is *regularized* if $p_0 \in (0, 1/2)$ and $p_0 \leq p_1 \leq 1 - p_0$. Namely, we exclude the degenerate case $p_0 = 0$ and $p_0 = p_1 = 1/2$ to guarantee that the quantities C_1 and C_2 defined in (3.7) and (3.8) are positive and finite. If $p_0 = p_1 = p \in (0, 1/2)$, we simply write BSC$(p)$.

Let C be the capacity of the BAC(p_0, p_1) and let (π_0^*, π_1^*) be the corresponding capacity-achieving input distribution. The following results will be useful in our proofs.

Fact 1. *Consider a BAC(p_0, p_1) with capacity-achieving input distribution (π_0^*, π_1^*). Then,*

$$C = \frac{p_0 h(p_1)}{1 - p_0 - p_1} - \frac{(1-p_1) h(p_0)}{1 - p_0 - p_1} + \log(1+z), \tag{3.1}$$

$$\pi_0^* = \frac{1 - p_1(1+z)}{(1 - p_0 - p_1)(1+z)}, \tag{3.2}$$

$$\pi_1^* = \frac{(1-p_0)(1+z) - 1}{(1 - p_0 - p_1)(1+z)}, \tag{3.3}$$

where $z = 2^{\frac{h(p_0) - h(p_1)}{1 - p_0 - p_1}}$. Furthermore, if $p_0 \in (0, 1/2)$ and $p_0 \leq p_1 \leq 1 - p_0$, then $0 < \pi_1^* \leq \pi_0^* < 1$.

The proof of Fact 1 is given in Appendix 3.9.1.

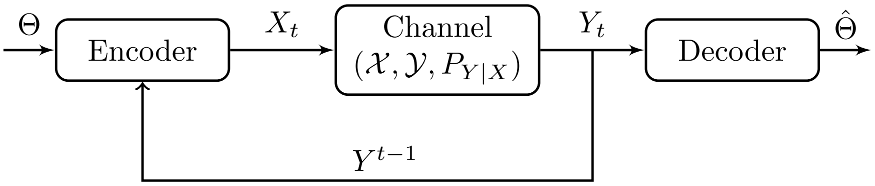

Figure 3.1: Variable-length coding over a memoryless channel $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$ with full noiseless feedback link.

Fact 2 (Theorem 4.5.1, [Gal68])**.** *Consider a DMC $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$ with capacity-achieving input distribution $(\pi_0^*, \pi_1^*, \ldots, \pi_{|\mathcal{X}|-1}^*)$. For each $k \in \{0, 1, \ldots, |\mathcal{X}|-1\}$, if $\pi_k^* > 0$, then,*

$$D\left(P(Y|X=k) \middle\| \sum_{l=0}^{|\mathcal{X}|-1} \pi_l^* P(Y|X=l)\right) = C. \tag{3.4}$$

Let C_1 be the maximal KL divergence between two conditional output distributions defined by

$$C_1 \triangleq \max_{x, x' \in \mathcal{X}} D\big(P(Y|X=x) \| P(Y|X=x')\big). \tag{3.5}$$

We also denote

$$C_2 \triangleq \max_{y \in \mathcal{Y}} \log \frac{\max_{x \in \mathcal{X}} P_{Y|X}(y|x)}{\min_{x \in \mathcal{X}} P_{Y|X}(y|x)}. \tag{3.6}$$

Fact 3. *For a regularized BAC(p_0, p_1),*

$$C_1 = D\Big(P(Y|X=1) \| P(Y|X=0)\Big), \tag{3.7}$$

$$C_2 = \log \frac{P_{Y|X}(1|1)}{P_{Y|X}(1|0)} = \log \frac{1-p_1}{p_0}. \tag{3.8}$$

The proof of Fact 3 is given in Appendix 3.9.2.

For a regularized BAC, it always holds that $0 < C \leq C_1 \leq C_2 < \infty$. Later, we will see how these quantities are used in our result.

3.2.3 VLF Codes for a Memoryless Channel

We follow [PPV11] in defining a VLF code for a memoryless channel $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$ with full feedback. Fig. 3.1 depicts the system model of variable-length coding over a memoryless channel with a full noiseless feedback link.

Definition 3. *An (l, M, ϵ) VLF code for a memoryless channel $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$, where $l > 0$, $M \in \mathbb{N}_+$, and $\epsilon \in (0, 1)$, is defined by:*

1) A random variable $\mathcal{C}$, defined on a set $\mathbb{C}$ with $|\mathbb{C}| \leq 2$, whose realization is revealed to both the encoder and decoder before the start of transmission. The realization of $\mathcal{C}$ is the common codebook.

2) A sequence of encoding functions $e_t : \mathbb{C} \times [M] \times \mathcal{Y}^{t-1} \to \mathcal{X}$, $t \in \mathbb{N}_+$, defining channel inputs

$$X_t = e_t(\mathcal{C}, \Theta, Y^{t-1}), \tag{3.9}$$

where Θ is uniformly distributed over $[M]$.

3) A sequence of decoding functions $g_t : \mathbb{C} \times \mathcal{Y}^t \to [M]$, $t \in \mathbb{N}_+$, providing the best estimate of Θ at time t.

4) A random variable $\tau \in \mathbb{N}$, a stopping time of the filtration $\mathcal{F}_t = \sigma\{\mathcal{C}, Y^t\}$ satisfying $\mathbb{E}[\tau] \leq l$. The final decision $\hat{\Theta}$ is computed at stopping time τ, given by

$$\hat{\Theta} = g_\tau(Y^\tau). \tag{3.10}$$

In addition, τ also needs to satisfy

$$P_e \triangleq \mathscr{P}\{\Theta \neq \hat{\Theta}\} \leq \epsilon. \tag{3.11}$$

The rate of an (l, M, ϵ) VLF code is defined as

$$R \triangleq \frac{\log M}{\mathbb{E}[\tau]}. \tag{3.12}$$

In [Bur76], Burnashev, for the first time, derived the *reliability function* $E(R)$ of variable-length coding over a fixed DMC for all rates $R < C$:

$$E(R) = C_1 \left(1 - \frac{R}{C}\right). \tag{3.13}$$

3.2.4 Naghshvar et al.'s SED Coding Scheme

In [NWJ12], Naghshvar *et al.* introduced a novel SED coding scheme that produces a deterministic VLF code for a symmetric binary-input channel with full feedback. We briefly describe their scheme below.

Let $\rho_i(t) \triangleq \mathscr{P}\{\Theta = i | Y^t\}$, $t \in \mathbb{N}$, denote the posterior probability of $\Theta = i$. Since Θ is uniformly distributed before transmission, $\rho_i(0) = 1/M$ for all $i \in [M]$. As noted in [NJW15], a sufficient statistic for estimating Θ is the *belief state vector* defined by

$$\boldsymbol{\rho}(t) \triangleq [\rho_1(t), \rho_2(t), \ldots, \rho_M(t)], \quad t \in \mathbb{N}. \tag{3.14}$$

According to Bayes' rule, upon receiving $Y_t = y_t$, each $\rho_i(t)$, $i \in [M]$, can be updated from $\boldsymbol{\rho}(t-1)$ by

$$\rho_i(t) = \frac{\rho_i(t-1) P_{Y|X}\big(y_t | x_{t,i}\big)}{\sum_{j \in [M]} \rho_j(t-1) P_{Y|X}\big(y_t | x_{t,j}\big)}, \tag{3.15}$$

where $x_{t,j} \triangleq e_t(\mathcal{C}, j, Y^{t-1}) \in \{0, 1\}$ denotes the input symbol for message $j \in [M]$. Thanks to the full noiseless feedback, the transmitter will be informed of y_t at time instant $t+1$ and thus can calculate the same $\boldsymbol{\rho}(t)$ which will be used to produce X_{t+1}.

We follow Definition 3 to describe the deterministic VLF code generated with Naghshvar *et al.*'s SED coding scheme that guarantees target error probability $\epsilon \in (0, 1/2)$.

1) *A common codebook generated by SED bipartition*: If $t = 1$, $\boldsymbol{\rho}(0) = (1/M)\mathbf{1}$. If $t \geq 2$, $\boldsymbol{\rho}(t-1)$ is obtained from Y_{t-1}, $\boldsymbol{\rho}(t-2)$, and previous input symbol assignments $\{x_{t-1,i}\}_{i \in [M]}$ using Bayes' rule (3.15). At time $t \in \mathbb{N}_+$, upon obtaining $\boldsymbol{\rho}(t-1)$, the

message set $[M]$ is partitioned into two subsets $S_0(t-1)$ and $S_1(t-1)$ such that

$$0 \le \pi_0(t-1) - \pi_1(t-1) \le \min_{i \in S_0(t-1)} \rho_i(t-1), \tag{3.16}$$

where $\pi_x(t-1) \triangleq \sum_{i \in S_x(t-1)} \rho_i(t-1)$, $x \in \{0,1\}$. Then, the input symbol $x_{t,i}$ for message $i \in [M]$ is 0 if $i \in S_0(t-1)$ and is 1 if $i \in S_1(t-1)$.

2) *Encoding function*: At time $t \in \mathbb{N}_+$, the encoder obtains $S_1(t-1)$ from $\boldsymbol{\rho}(t-1)$ according to step 1). The encoding function at time t is given by

$$e_t(\mathcal{C}, \Theta, Y^{t-1}) \triangleq \mathbf{1}_{\{\Theta \in S_1(t-1)\}}. \tag{3.17}$$

3) *Decoding function*: At time $t \in \mathbb{N}_+$, upon receiving y_t, the decoder first obtains input symbol assignments $\{x_{t,i}\}_{i \in [M]}$ from $\boldsymbol{\rho}(t-1)$ according to step 1). Next, the decoder computes new belief state vector $\boldsymbol{\rho}(t)$ using y_t, $\boldsymbol{\rho}(t-1)$ and $\{x_{t,i}\}_{i \in [M]}$ according to Bayes' rule (3.15). The decoding function at time t is given by

$$g_t(\mathcal{C}, Y^t) \triangleq \arg\max_{i \in [M]} \rho_i(t). \tag{3.18}$$

If there are multiple solutions in (3.18), the decoder arbitrarily selects a message among them.

4) *Stopping time* τ: The decoder adopts the following stopping time, which is a function of filtration $\mathcal{F}_t = \sigma\{Y^t\}$,

$$\tau \triangleq \min\left\{t \in \mathbb{N} : \max_{i \in [M]} \rho_i(t) \ge 1 - \epsilon\right\}, \tag{3.19}$$

where the computation of $\boldsymbol{\rho}(t)$ is described in 3). The final estimate $\hat{\Theta}$ is thus given by

$$\hat{\Theta} = \arg\max_{i \in [M]} \rho_i(\tau). \tag{3.20}$$

Clearly, the stopping time τ satisfies (4.6) because

$$P_e = \mathbb{E}\big[\mathscr{P}\{\Theta \ne \hat{\Theta} | Y^\tau\}\big] = \mathbb{E}\left[1 - \max_{i \in [M]} \rho_i(\tau)\right] \le \epsilon. \tag{3.21}$$

Note that in 1), there are many deterministic partitioning algorithms that achieve (4.10), yet both the encoder and decoder must agree on the same algorithm. As can be seen, the common codebook $\mathcal{C}$ at time t is a function of Y^{t-1}, hence the resulting VLF code is deterministic. Since the target error probability $\epsilon \in (0, 1/2)$, it follows that (3.20) always has a unique solution. Later, we show that τ defined in (3.19) under the SED coding scheme is almost surely (a.s.) finite, i.e., $\mathscr{P}\{\tau < \infty\} = 1$; see Lemma 10. Our goal is to determine a non-asymptotic upper bound on $\mathbb{E}[\tau]$.

To analyze the deterministic VLF code constructed with the SED coding scheme, we examine the log-likelihood ratio defined by

$$U_j(t) \triangleq \log \frac{\rho_j(t)}{1-\rho_j(t)}, \quad j \in [M]. \tag{3.22}$$

Using the log-likelihood ratio, the stopping time (3.19) can be equivalently written as

$$\tau = \min\left\{t \in \mathbb{N} : \max_{i \in [M]} U_i(t) \geq \log \frac{1-\epsilon}{\epsilon}\right\}. \tag{3.23}$$

3.2.5 Previous Results on Average Blocklength of VLF Codes

In [NJW15], for a given message size $M \geq 2$ and target error probability $\epsilon \in (0, 1/2)$, Naghshvar *et al.* used the EJS divergence and submartingale synthesis technique to obtain a non-asymptotic upper bound on $\mathbb{E}[\tau]$ for the deterministic VLF code constructed with the SED coding scheme operated over the symmetric binary-input channel.

Theorem 9 (Remark 7, [NJW15]). *For a given integer $M \geq 2$ and $\epsilon \in (0, 1/2)$, the deterministic (l, M, ϵ) VLF code constructed with the SED coding scheme in Sec. 3.2.4 for the symmetric binary-input channel $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$ satisfies*

$$l \leq \frac{\log M + \log\log \frac{M}{\epsilon}}{C} + \frac{\log \frac{1}{\epsilon} + 1}{C_1} + \frac{96 \cdot 2^{2C_2}}{CC_1}. \tag{3.24}$$

The technique that underlies this result is a two-stage submartingale resulted from the SED coding rule described in Sec. 3.2.4.

Lemma 1 ([NWJ12]). *Fix BSC(p), $p \in (0, 1/2)$ and $\Theta = i \in [M]$. The SED coding scheme in Sec. 3.2.4 induces a submartingale $\{U_i(t)\}_{t=0}^{\infty}$ with respect to the filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$ satisfying*

$$\mathbb{E}[U_i(t+1)|\mathcal{F}_t, \Theta = i] \geq U_i(t) + C, \quad \textit{if } U_i(t) < 0, \tag{3.25a}$$

$$\mathbb{E}[U_i(t+1)|\mathcal{F}_t, \Theta = i] = U_i(t) + C_1, \quad \textit{if } U_i(t) \geq 0, \tag{3.25b}$$

$$|U_i(t+1) - U_i(t)| \leq C_2. \tag{3.25c}$$

The proof of Lemma 1 can be found in [NWJ12, Appendix A]. We remark that the key step that links the SED coding scheme to the two-stage submartingale is the introduction and analysis of *extrinsic probabilities*; see [NWJ12, Eq. 19]. The next step is to synthesize the two-stage submartingale in Lemma 1 into a single submartingale and then apply Doob's optional stopping theorem. In [NJW15], Naghshvar *et al.* generalized [BZ75, Lemma 1] to obtain the following result.

Lemma 2 (Lemma 8, [NJW15]). *Assume that the sequence $\{\xi_t\}_{t=0}^{\infty}$ forms a submartingale with respect to filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. Furthermore, assume there exist positive constants K_1, K_2 and K_3 such that*

$$\mathbb{E}[\xi_{t+1}|\mathcal{F}_t] \geq \xi_t + K_1, \quad \textit{if } \xi_t < 0, \tag{3.26a}$$

$$\mathbb{E}[\xi_{t+1}|\mathcal{F}_t] \geq \xi_t + K_2, \quad \textit{if } \xi_t \geq 0, \tag{3.26b}$$

$$|\xi_{t+1} - \xi_t| \leq K_3, \quad \textit{if } \max\{\xi_{t+1}, \xi_t\} \geq 0. \tag{3.26c}$$

Consider the stopping time $v = \min\{t : \xi_t \geq B\}$, $B > 0$. Then, we have the inequality,

$$\mathbb{E}[v] \leq \frac{B - \xi_0}{K_2} + \xi_0 \mathbf{1}_{\{\xi_0 < 0\}} \left(\frac{1}{K_2} - \frac{1}{K_1} \right) + \frac{3K_3^2}{K_1 K_2}. \tag{3.27}$$

Observe that if $U_i(t)$ in Lemma 1 plays the role of ξ_t in Lemma 2, the sequence $\{U_i(t)\}_{t=0}^{\infty}$ meets the conditions in Lemma 2 by setting $K_1 = C$, $K_2 = C_1$ and $K_3 = C_2$. Thus, by setting $B = \log \frac{1-\epsilon}{\epsilon}$, the stopping rule in Lemma 2 coincides with that in (3.23) and we have the following corollary.

Corollary 1. *For a given integer $M \geq 2$ and $\epsilon \in (0, 1/2)$, the deterministic (l, M, ϵ) VLF code constructed with the SED coding scheme in Sec. 3.2.4 for the symmetric binary-input channel satisfies*

$$l \leq \frac{\log M}{C} + \frac{\log \frac{1-\epsilon}{\epsilon}}{C_1} + \frac{3C_2^2}{CC_1}. \tag{3.28}$$

Remark 1. *In [NJW15], Naghshvar* et al. *proved a two-stage submartingale similar to Lemma 1 by considering the average log-likelihood ratio $\tilde{U}(t)$ of the belief state $\boldsymbol{\rho}(t)$ rather than that of the transmitted message (see [NJW15, Appendix II]). They showed that the average drift of $\tilde{U}(t)$ is characterized by the EJS divergence, which is lower bounded by C or $\tilde{\rho}C_1$ depending on whether the sign of $\tilde{U}(t)$ is negative, where $\tilde{\rho} \in (0, 1)$ is some constant. Combining their two-stage submartingale with Lemma 2, they obtained Theorem 18. However, a direct comparison of the third terms in* (4.24) *and* (4.29) *immediately reveals that* (4.29) *is a significantly tighter upper bound on l.*

Next, we recall Polyanskiy's achievability bound for an (l, M, ϵ') VLSF code operated over an arbitrary DMC.

Theorem 10 (Theorem 3, [PPV11])**.** *Consider a DMC $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$. Fix a scalar $\gamma > 0$. Let X^n and $\bar{X}^n$ be independent copies from the same process and let Y^n be the output of the DMC when X^n is the input. Define a sequence of information density functions*

$$\iota(a^n; b^n) \triangleq \log \frac{P_{Y^n|X^n}(b^n|a^n)}{P_{Y^n}(b^n)} \tag{3.29}$$

and a pair of hitting times

$$\psi \triangleq \min\{n \geq 0 : \iota(X^n; Y^n) \geq \gamma\}, \tag{3.30}$$

$$\bar{\psi} \triangleq \min\{n \geq 0 : \iota(\bar{X}^n; Y^n) \geq \gamma\}. \tag{3.31}$$

Then, for an integer $M \geq 2$, there exists an (l, M, ϵ') VLSF code satisfying

$$l \leq \mathbb{E}[\psi], \tag{3.32}$$

$$\epsilon' \leq (M-1)\mathscr{P}\{\bar{\psi} \leq \psi\}. \tag{3.33}$$

In general, it is still difficult to compute $\mathbb{E}[\psi]$ and $\mathscr{P}\{\bar{\psi} \leq \psi\}$. Nevertheless, for a DMC with bounded information density, i.e., $a_0 \triangleq \sup_{x\in\mathcal{X},y\in\mathcal{Y}} \iota(x;y) < \infty$, Polyanskiy *et al.* proved the following useful relaxations by drawing independent and identically distributed (i.i.d.) X^n from capacity-achieving input distribution P_X^*

$$\mathbb{E}[\psi] \leq \frac{\gamma + a_0}{C}, \tag{3.34}$$

$$\mathscr{P}\{\bar{\psi} \leq \psi\} \leq 2^{-\gamma}. \tag{3.35}$$

Therefore, given a target error probability $\epsilon \in (0,1)$, by setting $\gamma = \log \frac{M-1}{\epsilon}$ in (3.34) and (3.35), (3.32) and (3.33) are further relaxed to

$$l \leq \frac{\log \frac{M-1}{\epsilon} + a_0}{C}, \tag{3.36}$$

$$\epsilon' \leq \epsilon. \tag{3.37}$$

In this chapter, we use the relaxed upper bounds (3.36), (3.37) to compute Polyanskiy's VLSF achievability bound on rate for a regularized BAC.

Finally, we recall Polyanskiy's converse bound for an (l, M, ϵ) VLF code operated over a DMC.

Theorem 11 (Theorems 4 and 6, [PPV11])**.** *Consider a DMC with* $0 < C \leq C_1 < \infty$*. Then any* (l, M, ϵ) *VLF code with* $0 < \epsilon \leq 1 - 1/M$*, satisfies both*

$$l \geq \sup_{0<\xi\leq\frac{M-1}{M}} \left[\frac{1}{C}\Big(\log M - F_M(\xi) - \min\Big\{F_M(\epsilon), \frac{\epsilon \log M}{\xi}\Big\}\Big) + \left[\frac{1-\epsilon}{C_1}\log\frac{\lambda_1\xi}{\epsilon(1-\xi)} - \frac{h(\epsilon)}{C_1}\right]^+\right], \tag{3.38}$$

and

$$l \geq \frac{(1-\epsilon)\log M - h(\epsilon)}{C}, \tag{3.39}$$

where

$$F_M(x) \triangleq x\log(M-1) + h(x), \quad x \in [0,1], \tag{3.40}$$

$$\lambda_1 \triangleq \min_{y,x_1,x_2} \frac{P_{Y|X}(y|x_1)}{P_{Y|X}(y|x_2)} \in (0,1). \tag{3.41}$$

3.3 Achievable Rates for BAC With Feedback

For a regularized BAC(p_0, p_1) with $p_0 \neq p_1$, Naghshvar *et al.*'s SED coding scheme no longer applies. In this section, we introduce the generalized SED coding scheme for a regularized BAC(p_0, p_1) with full noiseless feedback and develop a non-asymptotic achievability bound.

Intuitively speaking, in order to achieve capacity, the posterior matching principle [SF11] suggests that the coding scheme should shape the belief state vector $\boldsymbol{\rho}(t)$ into a Bernoulli distribution $(\pi_0(t), \pi_1(t))$ such that it is close to the capacity-achieving input distribution (π_0^*, π_1^*). That is, we wish $\pi_x(t)/\pi_x^* \approx 1$ for $x \in \{0, 1\}$. One way to ask for this is that the difference $\pi_0(t)/\pi_0^* - \pi_1(t)/\pi_1^*$ is close to zero. In analogy with Naghshvar *et al.*'s analysis [NWJ12], it suffices to require that this difference be *small enough*. This motivates our generalized SED coding scheme for the regularized BAC below.

For a regularized BAC(p_0, p_1), recall from Fact 1 that $0 < \pi_1^* \leq \pi_0^* < 1$. Using this, we propose the following generalized, deterministic, two-phase SED coding scheme for a regularized BAC(p_0, p_1) that is similar to Naghshvar *et al.*'s SED coding scheme described in Sec. 3.2.4 with an exception that 1) is replaced by

1') *A common codebook generated by the generalized SED bipartition*: If $t = 1$, $\boldsymbol{\rho}(0) = (1/M)\mathbf{1}$. If $t \geq 2$, $\boldsymbol{\rho}(t-1)$ is obtained from Y_{t-1}, $\boldsymbol{\rho}(t-2)$, and previous input symbol assignments $\{x_{t-1,i}\}_{i\in[M]}$ using Bayes' rule (3.15). At time $t \in \mathbb{N}_+$, upon obtaining $\boldsymbol{\rho}(t-1)$, let $\hat{i} = \arg\max_{j\in[M]} \rho_j(t-1)$. If $\rho_{\hat{i}}(t-1) < \pi_1^*$, the message set $[M]$ is partitioned into two subsets $S_0(t-1)$ and $S_1(t-1)$ such that

$$\frac{\pi_0(t-1)}{\pi_0^*} - \frac{\pi_1(t-1)}{\pi_1^*} \geq -\frac{\min_{i\in S_1(t-1)} \rho_i(t-1)}{\pi_1^*}, \tag{3.42a}$$

$$\frac{\pi_0(t-1)}{\pi_0^*} - \frac{\pi_1(t-1)}{\pi_1^*} \leq \frac{\min_{i\in S_0(t-1)} \rho_i(t-1)}{\pi_0^*}, \tag{3.42b}$$

where $\pi_x(t-1) = \sum_{i\in S_x(t-1)} \rho_i(t-1)$, $x \in \{0, 1\}$. If $\rho_{\hat{i}}(t-1) \geq \pi_1^*$, the message set $[M]$ is exclusively partitioned into $S_1(t-1) = \{\hat{i}\}$ and $S_0(t-1) = [M] \setminus \{\hat{i}\}$. Then, the input symbol $x_{t,i}$ for message $i \in [M]$ is 0 if $i \in S_0(t-1)$ and is 1 if $i \in S_1(t-1)$.

Remark 2. *First, we see that in the second case where $\rho_{\hat{i}}(t-1) \geq \pi_1^*$, the partition $S_1(t-1) = \{\hat{i}\}$, $S_0(t-1) = [M] \setminus \{\hat{i}\}$ still meets* (3.42)*. Second, if $p_0 = p_1$, then $\pi_0^* = \pi_1^* = 1/2$. Thus,* (3.42) *simplifies to*

$$\pi_0(t-1) - \pi_1(t-1) \geq -\min_{i \in S_1(t-1)} \rho_i(t-1), \tag{3.43a}$$

$$\pi_0(t-1) - \pi_1(t-1) \leq \min_{i \in S_0(t-1)} \rho_i(t-1). \tag{3.43b}$$

Clearly, this is a relaxation of (4.10)*. If $\rho_{\hat{i}}(t-1) \geq 1/2$,* (4.10) *is met if and only if $S_0(t-1) = \{\hat{i}\}$ and $S_1(t-1) = [M] \setminus \{\hat{i}\}$. In [NWJ12], Naghshvar* et al. *showed that this partition achieves C_1 defined in* (3.5)*. However, by symmetry of the BSC, one can show that the partition $S_1(t-1) = \{\hat{i}\}$, $S_0(t-1) = [M] \setminus \{\hat{i}\}$, which corresponds to the second case in 1'), also achieves the same C_1. Therefore, our SED coding scheme serves as a generalization of Naghshvar* et al.*'s SED coding scheme.*

The significance of our generalized SED coding scheme is that Lemma 1 now holds for the regularized BAC. For the sake of completeness, we state this result in a separate lemma.

Lemma 3. *Fix a regularized BAC(p_0, p_1) and $\Theta = i \in [M]$. The generalized SED coding scheme induces a submartingale $\{U_i(t)\}_{t=0}^{\infty}$ with respect to the filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$ satisfying*

$$\mathbb{E}[U_i(t+1)|\mathcal{F}_t, \Theta = i] \geq U_i(t) + C, \quad \textit{if } U_i(t) < 0, \tag{3.44a}$$

$$\mathbb{E}[U_i(t+1)|\mathcal{F}_t, \Theta = i] = U_i(t) + C_1, \quad \textit{if } U_i(t) \geq 0, \tag{3.44b}$$

$$|U_i(t+1) - U_i(t)| \leq C_2. \tag{3.44c}$$

Proof. The proof fully exploits the properties of extrinsic probabilities originally proposed in [NWJ12]. See Section 3.5.1 for the complete proof. □

Since Lemma 2 is developed from a poor choice of parameters, here we perform a submartingale synthesis with optimized parameters to obtain the best possible achievability bound for a regularized BAC with feedback.

Theorem 12. *For a given integer $M \geq 2$ and $\epsilon \in (0, 1/2)$, the deterministic (l, M, ϵ) VLF code constructed with the generalized SED coding scheme for the regularized $BAC(p_0, p_1)$ satisfies*

$$l < \frac{\log M}{C} + \frac{\log \frac{1-\epsilon}{\epsilon} + C_2}{C_1} + C_2 \left(\frac{1}{C} - \frac{1}{C_1} \right) \frac{1 - \frac{\epsilon}{1-\epsilon} 2^{-C_2}}{1 - 2^{-C_2}}. \tag{3.45}$$

Proof. See Section 3.5.2. □

```
**Algorithm 3** Original SED Partitioning Algorithm
**Require:** $\max_{i \in [M]} \rho_i < \pi_1^*$;
$S_0 \leftarrow \{1, 2, \ldots, M\}$ and $S_1 \leftarrow \varnothing$;
$\pi_0 \leftarrow 1$, $\pi_1 \leftarrow 0$, $\lambda \leftarrow \pi_1^* / \pi_0^*$, $\delta \leftarrow \lambda$, $\rho_{\min,0} \leftarrow \min_{i \in S_0} \rho_i$, and $\rho_{\min,1} \leftarrow 0$;
**while** $(\delta < -\rho_{\min,1}) \,||\, (\delta > \lambda \rho_{\min,0})$ **do**
    **if** $\delta < -\rho_{\min,1}$ **then**
        $j \leftarrow \arg\min_{i \in S_1} \rho_i$;
        $S_0 \leftarrow S_0 \cup \{j\}$ and $S_1 \leftarrow S_1 \setminus \{j\}$;
        $\pi_0 \leftarrow \pi_0 + \rho_j$ and $\pi_1 \leftarrow \pi_1 - \rho_j$;
    **end if**
    **if** $\delta > \lambda \rho_{\min,0}$ **then**
        $j \leftarrow \arg\min_{i \in S_0} \rho_i$;
        $S_0 \leftarrow S_0 \setminus \{j\}$ and $S_1 \leftarrow S_1 \cup \{j\}$;
        $\pi_0 \leftarrow \pi_0 - \rho_j$ and $\pi_1 \leftarrow \pi_1 + \rho_j$;
    **end if**
    $\delta \leftarrow \lambda \pi_0 - \pi_1$, $\rho_{\min,0} \leftarrow \min_{i \in S_0} \rho_i$, $\rho_{\min,1} \leftarrow \min_{i \in S_1} \rho_i$;
**end while**
**for** $i \leftarrow 1, 2, \ldots, M$ **do**
    $x_{t,i} = \begin{cases} 0, & \text{if } i \in S_0 \\ 1, & \text{if } i \in S_1 \end{cases}$
**end for**
```

Algorithm 4 Greedy SED Partitioning Algorithm

```
Require: $\max_{i\in[M]} \rho_i < \pi_1^*$;
$j_1 \leftarrow \arg\max_{i\in[M]} \rho_i$;
$S_0 \leftarrow \{j_1\}$ and $S_1 \leftarrow \varnothing$;
$\pi_0 \leftarrow \rho_{j_1}$, $\pi_1 \leftarrow 0$ and $\lambda \leftarrow \pi_1^*/\pi_0^*$;
for $s \leftarrow 2, 3, \ldots, M$ do
    $j_s \leftarrow \arg\max_{i\in[M]\setminus\{j_1,\ldots,j_{s-1}\}} \rho_i$;
    if $\pi_1 \geq \lambda\pi_0$ then
        $S_0 \leftarrow S_0 \cup \{j_s\}$;
        $\pi_0 \leftarrow \pi_0 + \rho_{j_s}$;
    else
        $S_1 \leftarrow S_1 \cup \{j_s\}$;
        $\pi_1 \leftarrow \pi_1 + \rho_{j_s}$;
    end if
end for
for $i \leftarrow 1, 2, \ldots, M$ do
    $x_{t,i} = \begin{cases} 0, & \text{if } i \in S_0 \\ 1, & \text{if } i \in S_1 \end{cases}$
end for
```

In the following, we show that the condition (3.42) required by the generalized SED coding scheme is always attainable at each time t. This is accomplished by solving a particular minimization problem.

Theorem 13. *For a regularized BAC with capacity-achieving input distribution* (π_0^*, π_1^*), *let* $\lambda \triangleq \pi_1^*/\pi_0^* \in (0, 1]$ *according to Fact 1. For a given belief state vector* $\boldsymbol{\rho} = [\rho_1, \rho_2, \ldots, \rho_M]$ *satisfying* $\max_{i\in[M]} \rho_i < \pi_1^*$, *define the following objective function* $f : 2^{[M]} \to \mathbb{R}$*:*

$$f(S) \triangleq \lambda\big(\pi_1(S) - \lambda\pi_0(S)\big)\mathbf{1}_{\{\pi_1(S)\geq\lambda\pi_0(S)\}} + \big(\lambda\pi_0(S) - \pi_1(S)\big)\mathbf{1}_{\{\pi_1(S)<\lambda\pi_0(S)\}}, \tag{3.46}$$

where $\pi_0(S) \triangleq \sum_{i\in S} \rho_i$ *and* $\pi_1(S) \triangleq \sum_{i\in[M]\setminus S} \rho_i$. *Assume* $S_0^* \subseteq [M]$ *minimizes* (3.46). *Then, the partition* $(S_0^*, [M] \setminus S_0^*)$ *satisfies* (3.42).

Proof. See Section 3.5.3. □

Theorem 13 implies that when all posterior probabilities in $\boldsymbol{\rho}(t)$ is less than π_1^*, it is always possible to identify a bipartition of $[M]$ that satisfies (3.42). In fact, the proof of Theorem 13 already reveals such a partitioning algorithm described in Algorithm 3. The algorithm is initialized with a partition of $[M]$ that fails to meet (3.42) and then successively constructs a new partition from the previous one that reduces $f(S)$. The termination condition is given by (3.42). Theorem 13 ensures that the termination will always be triggered at some point.

Finally, we present a greedy SED partitioning algorithm described in Algorithm 4 that provably meets (3.42). We state this result in the following theorem.

Theorem 14. *Let* (π_0^*, π_1^*) *be the capacity-achieving input distribution for a regularized BAC. Let* $\boldsymbol{\rho} = [\rho_1, \rho_2, \ldots, \rho_M]$ *be the belief state vector for Algorithm 4 satisfying* $\max_{i\in[M]} \rho_i < \pi_1^*$. *Let* (S_0, S_1) *be the bipartition of* $[M]$ *generated by Algorithm 4. Then,* (S_0, S_1) *satisfies* (3.42).

Proof. See Section 3.5.4. □

Remark 3. *Both Algorithms 3 and 4 have complexity* $O(M \log M)$*, making them not suitable for practical implementation. It still remains open how to further reduce the complexity of the SED partitioning algorithm for a regularized BAC. Nevertheless, in [AYW20], Antonini et al. proposed a type-based partitioning algorithm for the BSC based on a relaxed SED condition with a reduced complexity* $O(\log^2 M)$. *Simulations show that the coding scheme based on their relaxed SED condition achieves a similar performance as the original SED coding scheme.*

3.4 Achievable Rates for BSC With Feedback

In this section, we present our refined non-asymptotic achievability bound for the deterministic VLF code constructed with the SED coding scheme for the BSC(p), $p \in (0, 1/2)$, with full noiseless feedback. Both Naghshvar *et al.*'s and our SED coding scheme yield the same achievability bound. Therefore, the SED coding scheme refers to either of them.

Theorem 15. *For a given integer $M \geq 2$ and $\epsilon \in (0, 1/2)$, the deterministic (l, M, ϵ) VLF code constructed with the SED coding scheme for the BSC(p), $p \in (0, 1/2)$, satisfies*

$$l < \frac{\log M + \frac{1}{q}\log 2q}{C} + \frac{\log\frac{1-\epsilon}{\epsilon} + C_2}{C_1} + 2^{-C_2}C_2\left(\frac{1}{C} - \frac{1}{C_1} + \frac{\frac{1}{q}\log 2q}{CC_2}\right)\frac{1 - \frac{\epsilon}{1-\epsilon}2^{-C_2}}{1 - 2^{-C_2}}. \quad (3.47)$$

This result is a consequence of two supporting lemmas. To aid our discussion, let $q = 1-p$ and let us consider two stopping times for $\Theta = i$ when the log-likelihood ratio $U_i(t)$ first crosses 0 and $\log\frac{1-\epsilon}{\epsilon}$, respectively. Namely,

$$\nu_i \triangleq \min\{t \in \mathbb{N} : U_i(t) \geq 0\}, \quad (3.48)$$

$$\tau_i \triangleq \min\left\{t \in \mathbb{N} : U_i(t) \geq \log\frac{1-\epsilon}{\epsilon}\right\}. \quad (3.49)$$

Clearly, $\nu_i \leq \tau_i$. Equivalently, ν_i and τ_i also represent the stopping times when $\rho_i(t)$ first crosses $1/2$ and $1-\epsilon$, respectively.

We are now in a position to introduce the two supporting lemmas. First, note that

$$\mathbb{E}[\tau] = \frac{1}{M}\sum_{i=1}^{M}\mathbb{E}[\tau|\Theta = i] \leq \frac{1}{M}\sum_{i=1}^{M}\mathbb{E}[\tau_i|\Theta = i], \quad (3.50)$$

where the inequality follows since $\tau \leq \tau_i$ for all $i \in [M]$. Next, for $\mathbb{E}[\tau_i|\Theta = i]$, it can be decomposed into

$$\begin{aligned}\mathbb{E}[\tau_i|\Theta = i] &= \mathbb{E}[\nu_i|\Theta = i] + \mathbb{E}[\tau_i - \nu_i|\Theta = i]\\ &= \mathbb{E}[\nu_i|\Theta = i] + \mathbb{E}\big[\mathbb{E}[\tau_i - \nu_i|\Theta = i, U_i(\nu_i) = u]|\Theta = i\big]. \quad (3.51)\end{aligned}$$

The intuition behind decomposition (3.51) is that $\mathbb{E}[\nu_i|\Theta = i]$ corresponds to the average blocklength in the *first communication phase* (i.e., $U_i(t)$ traversing from $\log \frac{\rho_i(0)}{1-\rho_i(0)}$ to 0), and $\mathbb{E}[\tau_i - \nu_i|\Theta = i, U_i(\nu_i) = u]$ corresponds to the expected additional time spent in the confirmation phase with fallbacks to the communication phase. Here, u represents the value at which $U_i(t)$ arrives when it crosses threshold 0 for the first time.

Our next step is to develop upper bounds on $\mathbb{E}[\nu_i|\Theta = i]$ and $\mathbb{E}[\tau_i - \nu_i|\Theta = i, U_i(\nu_i) = u]$ that are independent from $\Theta = i$ and $U_i(\nu_i) = u$. We state each upper bound in Lemmas 4 and 5, respectively. Thus, summing up the two bounds will yield an upper bound on $\mathbb{E}[\tau_i|\Theta = i]$, hence an upper bound on $\mathbb{E}[\tau]$ using (3.50).

We remark that the technique for developing an upper bound on $\mathbb{E}[\nu_i|\Theta = i]$ makes use of a *surrogate submartingale*, thus allowing us to obtain a tighter constant term. In order to upper bound $\mathbb{E}[\tau_i - \nu_i|\Theta = i, U_i(\nu_i) = u]$, we first observe that the behavior of $U_i(t)$ in the confirmation phase can be modeled as a Markov chain with a *fallback self loop* on the initial state. This loop represents the probability that $U_i(t)$ first falls back to the communication phase and then returns to the confirmation phase. We first show that $\mathbb{E}[\tau_i - \nu_i|\Theta = i, U_i(\nu_i) = u]$ is upper bounded by a particular expected first-passage time on a generalized Markov chain. Further upper bounding the expected first-passage time yields the desired upper bound. Detailed analysis can be found in the proofs of Lemmas 4 and 5.

Lemma 4. *Fix a BSC(p), $p \in (0, 1/2)$. The stopping time ν_i defined in* (3.48) *under the SED coding scheme satisfies*

$$\mathbb{E}[\nu_i|\Theta = i] < \frac{\log M + \frac{1}{q}\log 2q}{C}. \tag{3.52}$$

Proof. See Section 3.5.5. □

Lemma 5. *Fix a BSC(p), $p \in (0, 1/2)$. The stopping times ν_i and τ_i defined in* (3.48) *and*

(3.49) *under the SED coding scheme satisfy*

$$\mathbb{E}[\tau_i-\nu_i|\Theta=i,U_i(\nu_i)=u]\le\frac{\log\frac{1-\epsilon}{\epsilon}+C_2}{C_1}+C_2 2^{-C_2}\left(\frac{1}{C}-\frac{1}{C_1}+\frac{\frac{1}{q}\log 2q}{CC_2}\right)\frac{1-\frac{\epsilon}{1-\epsilon}2^{-C_2}}{1-2^{-C_2}}. \tag{3.53}$$

Proof. See Section 3.5.6. □

3.5 Proofs

In this section, we prove our main results.

3.5.1 Proof of Lemma 3

Several steps in the proof of Lemma 3 are analogous to that in [NWJ12], for instance, the introduction of the extrinsic probabilities. However, the distinction is that our generalized SED coding scheme generalizes the analysis in [NWJ12, Appendix A2] which only works for symmetric binary-input channels.

Let $\Theta=i\in[M]$ be fixed. For brevity, let x_i be the input symbol for $\Theta=i$ at time $t+1$. Let $\mathcal{F}_t=\sigma\{\mathcal{C},Y^t\}$ denote the filtration generated by both the codebook $\mathcal{C}$ and Y^t. Thus, given $\mathcal{F}_t$ and $\Theta=i$, Y_{t+1} is distributed according to $P(Y|X=x_i)$. Hence, by letting $\bar{x}=1-x$,

$$\begin{aligned}
&\mathbb{E}[U_i(t+1)-U_i(t)|\mathcal{F}_t,\Theta=i]\\
&=\sum_{y\in\mathcal{Y}}P_{Y|X}(y|x_i)\left(\log\frac{\rho_i(t+1)}{1-\rho_i(t+1)}-\log\frac{\rho_i(t)}{1-\rho_i(t)}\right)\\
&=\sum_{y\in\mathcal{Y}}P_{Y|X}(y|x_i)\left(\log\frac{\frac{\rho_i(t)P_{Y|X}(y|x_i)}{\sum_{x\in\mathcal{X}}\pi_x(t)P_{Y|X}(y|x)}}{1-\frac{\rho_i(t)P_{Y|X}(y|x_i)}{\sum_{x\in\mathcal{X}}\pi_x(t)P_{Y|X}(y|x)}}-\log\frac{\rho_i(t)}{1-\rho_i(t)}\right)\\
&=\sum_{y\in\mathcal{Y}}P_{Y|X}(y|x_i)\log\frac{P_{Y|X}(y|x_i)}{\frac{\pi_{x_i}(t)-\rho_i(t)}{1-\rho_i(t)}P_{Y|X}(y|x_i)+\frac{\pi_{\bar{x}_i}(t)}{1-\rho_i(t)}P_{Y|X}(y|\bar{x}_i)}
\end{aligned}$$

$$= \sum_{y\in\mathcal{Y}} P_{Y|X}(y|x_i) \log \frac{P_{Y|X}(y|x_i)}{\sum_{x\in\mathcal{X}} \tilde{\pi}_{x,i}(t) P_{Y|X}(y|x)} \tag{3.54}$$

$$= D\big(P(Y|X=x_i)\|P(\tilde{Y})\big), \tag{3.55}$$

where in (3.54), we define the *extrinsic probabilities* by

$$\tilde{\pi}_{x_i,i}(t) \triangleq \frac{\pi_{x_i}(t) - \rho_i(t)}{1-\rho_i(t)}, \tag{3.56}$$

$$\tilde{\pi}_{\bar{x}_i,i}(t) \triangleq \frac{\pi_{\bar{x}_i}(t)}{1-\rho_i(t)}. \tag{3.57}$$

Clearly, $\tilde{\pi}_{x_i,i}(t) + \tilde{\pi}_{\bar{x}_i,i}(t) = 1$. Note that $\tilde{\pi}_{x_i,i}(t) \geq 0$ because $i \in S_{x_i}(t)$. In (3.55), $\tilde{Y}$ is the output induced by the channel $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$ for an input $\tilde{X}$ distributed as $(\tilde{\pi}_{0,i}(t), \tilde{\pi}_{1,i}(t))$.

Next, Lemmas 6 and 7 play a key role in connecting our generalized SED coding scheme to the two-stage submartingale in Lemma 3.

Lemma 6. *The extrinsic probability $\tilde{\pi}_{x_i,i}(t)$ under the generalized SED coding scheme in Sec. 3.3 for a BAC(p_0, p_1) with capacity-achieving input distribution (π_0^*, π_1^*) satisfies $\tilde{\pi}_{x_i,i}(t) \leq \pi^*_{x_i}$, where x_i is the input symbol for $\Theta = i$ at time $t+1$.*

Proof. Let $\hat{i} = \arg\max_{j\in[M]} \rho_j(t)$. We distinguish two cases: $\rho_{\hat{i}}(t) < \pi_1^*$ and $\rho_{\hat{i}}(t) \geq \pi_1^*$.

When $\rho_{\hat{i}}(t) < \pi_1^*$, we further discuss two subcases: $x_i = 0$ and $x_i = 1$. If $x_i = 0$, then $i \in S_0(t)$. Invoking (3.42b), we have

$$\begin{aligned}
\tilde{\pi}_{0,i}(t) - \pi_0^* &= (\pi_0^* + \pi_1^*)\tilde{\pi}_{0,i}(t) - \pi_0^* \\
&= \pi_1^*\tilde{\pi}_{0,i}(t) - \pi_0^*(1 - \tilde{\pi}_{0,i}(t)) \\
&= \pi_1^*\tilde{\pi}_{0,i}(t) - \pi_0^*\tilde{\pi}_{1,i}(t) \\
&= \frac{\pi_0^*\pi_1^*}{1-\rho_i(t)} \left(\frac{\pi_0(t) - \rho_i(t)}{\pi_0^*} - \frac{\pi_1(t)}{\pi_1^*} \right) \\
&\leq \frac{\pi_0^*\pi_1^*}{1-\rho_i(t)} \left(\frac{\pi_0(t) - \min_{j\in S_0(t)} \rho_j(t)}{\pi_0^*} - \frac{\pi_1(t)}{\pi_1^*} \right) \\
&\leq 0.
\end{aligned} \tag{3.58}$$

If $x_i = 1$, then $i \in S_1(t)$. By invoking (3.42a), we show in a similar fashion that

$$\begin{aligned}\tilde{\pi}_{1,i}(t) - \pi_1^* &= \frac{\pi_0^* \pi_1^*}{1-\rho_i(t)} \left(\frac{\pi_1(t) - \rho_i(t)}{\pi_1^*} - \frac{\pi_0(t)}{\pi_0^*} \right) \\ &\leq \frac{\pi_0^* \pi_1^*}{1-\rho_i(t)} \left(\frac{\pi_1(t) - \min_{j\in S_1(t)} \rho_j(t)}{\pi_1^*} - \frac{\pi_0(t)}{\pi_0^*} \right) \\ &\leq 0. \end{aligned} \tag{3.59}$$

Therefore, Lemma 6 holds for $\rho_{\hat{i}}(t) < \pi_1^*$.

When $\rho_{\hat{i}}(t) \geq \pi_1^*$, by the encoding rule, $S_1(t) = \{\hat{i}\}$ and $S_0(t) = [M] \setminus \{\hat{i}\}$. If $\hat{i} = i$, then $S_1(t) = \{i\}$ and $S_0(t) = [M] \setminus \{i\}$. Thus, $\tilde{\pi}_{1,i}(t) = 0 < \pi_1^*$. If $\hat{i} \neq i$, then $i \in S_0(t)$. Since $\pi_1(t) = \rho_{\hat{i}}(t) \geq \pi_1^*$, it follows that $\pi_0(t) \leq \pi_0^*$. Combining with the fact that $\tilde{\pi}_{0,i}(t) \leq \pi_0(t)$, we conclude that $\tilde{\pi}_{0,i}(t) \leq \pi_0^*$. Therefore, Lemma 6 also holds in this case.

Summarizing the above two cases, we conclude that Lemma 6 holds in general. □

Note that Lemma 6 does not require that the BAC be regularized. However, the regularized BAC is needed when we prove (3.44b). Next, we borrow a useful lemma on the KL divergence proved in [NJW15].

Lemma 7 (Lemma 1, [NJW15]). *For any two distributions P and Q on a set $\mathcal{Y}$ and $\alpha \in [0,1]$, $D(P\|\alpha P + (1-\alpha)Q)$ is decreasing in α.*

By Lemma 7, let $P \equiv P(Y|X = x_i)$, $Q \equiv P(Y|X = \bar{x}_i)$, and $\alpha = \tilde{\pi}_{x_i,i}(t)$. Then, (3.55) is lower bounded by

$$D\big(P(Y|X = x_i)\|P(\tilde{Y})\big) = D\big(P\|\alpha P + (1-\alpha)Q\big) \tag{3.60}$$

$$\geq D\big(P\|\pi_{x_i}^* P + \pi_{\bar{x}_i}^* Q\big) \tag{3.61}$$

$$= C, \tag{3.62}$$

where (3.61) follows from Lemma 6 and (3.62) follows from Fact 2. Therefore, with the generalized SED coding scheme, it always holds that

$$\mathbb{E}[U_i(t+1)|\mathcal{F}_t, \Theta = i] \geq U_i(t) + C. \tag{3.63}$$

As a result, (3.44a) is proved.

In particular, if $U_i(t) \geq 0$, this is equivalent to $\rho_i(t) \geq 1/2$ and thus i is the index with the maximum posterior. Using Fact 1 that $\pi_1^* \leq 1/2$, it follows that $\max_{j\in[M]} \rho_j(t) = \rho_i(t) \geq \pi_1^*$. Thus, according to the generalized SED coding scheme in Sec. 3.3, the message set $[M]$ is exclusively partitioned into $S_1(t) = \{i\}$ and $S_0(t) = [M] \setminus \{i\}$, resulting in $\tilde{\pi}_{1,i}(t) = 0$ and

$$\begin{aligned} D\big(P(Y|X = x_i)\|P(\tilde{Y})\big) &= D\big(P(Y|X = 1)\|P(Y|X = 0)\big) \\ &= C_1, \end{aligned} \tag{3.64}$$

where (3.64) follows from Fact 3. Therefore, (3.44b) is proved. We also remark that for $Y_{t+1} = y$,

$$U_i(t+1) = U_i(t) + \log \frac{P_{Y|X}(y|1)}{P_{Y|X}(y|0)}, \quad \text{if } U_i(t) \geq 0. \tag{3.65}$$

Hence, C_1 can be thought of as the average drift of $U_i(t)$ whenever $U_i(t) \geq 0$.

To prove (3.44c), we note that when $Y_{t+1} = y$, by (3.15),

$$\begin{aligned} |U_i(t+1) - U_i(t)| &= \left|\log \frac{\rho_i(t+1)}{1-\rho_i(t+1)} - \log \frac{\rho_i(t)}{1-\rho_i(t)}\right| \\ &= \left|\log \left(\frac{\rho_i(t) P_{Y|X}(y|x_{t+1,i})}{\sum_{j\neq i} \rho_j(t) P_{Y|X}(y|x_{t+1,j})} \cdot \frac{1-\rho_i(t)}{\rho_i(t)}\right)\right| \\ &= \left|\log \frac{P_{Y|X}(y|x_{t+1,i}))}{\sum_{j\neq i} \frac{\rho_j(t)}{1-\rho_i(t)} P_{Y|X}(y|x_{t+1,j}))}\right| & (3.66) \\ &\leq \log \frac{\max_{x\in\mathcal{X}} P_{Y|X}(y|x)}{\min_{x\in\mathcal{X}} P_{Y|X}(y|x)}. & (3.67) \end{aligned}$$

Hence, we have

$$\begin{aligned} |U_i(t+1) - U_i(t)| &\leq \max_{y\in\mathcal{Y}} \log \frac{\max_{x\in\mathcal{X}} P_{Y|X}(y|x)}{\min_{x\in\mathcal{X}} P_{Y|X}(y|x)}, \\ &= C_2, \end{aligned} \tag{3.68}$$

which completes the proof of (3.44c).

3.5.2 Proof of Theorem 12

The proof of Theorem 12 involves a submartingale synthesis with optimized parameters (Lemma 8) and a variant of Doob's optional stopping theorem (Lemma 9). We establish auxiliary Lemmas 11 and 10 to show that the sufficient conditions in Lemma 9 hold. Throughout the proof, we fix $\Theta = i \in [M]$ to avoid writing the conditioning $\Theta = i$ unless otherwise specified.

Let the sequence $\{U_i(t)\}_{t=0}^{\infty}$ be the two-stage submartingale defined in (3.44) with respect to filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$ as a result of the generalized SED coding scheme over a regularized BAC(p_0, p_1). Let us consider a sequence $\{\eta(t)\}_{t=0}^{\infty}$ defined as

$$\eta(t) \triangleq \begin{cases} -A + \frac{U_i(t)}{C} - t, & \text{if } U_i(t) < 0, \\ -Ae^{-sU_i(t)} + \frac{U_i(t)}{C_1} - t, & \text{if } U_i(t) \geq 0, \end{cases} \tag{3.69}$$

where $s > 0$ and $A > 0$ are two parameters to be chosen. This particular sequence is originally considered by Burnashev and Zigangirov that facilitates a general upper bound on the expected stopping time [BZ75, Eq. 4.9]. For our purposes, we require that parameters s and A meet the following two equations,

$$A(1 - e^{-sC_2}) - C_2\left(\frac{1}{C} - \frac{1}{C_1}\right) = 0, \tag{3.70}$$

$$p_1 e^{-s\log\frac{p_1}{1-p_0}} + (1-p_1)e^{-s\log\frac{1-p_1}{p_0}} = 1. \tag{3.71}$$

The motivation behind these equations is to select the best parameters that make $\{\eta(t)\}_{t=0}^{\infty}$ just a submartingale. This will become clearer as our proof proceeds. Solving (3.70) and (3.71) yields

$$s = \ln 2, \tag{3.72}$$

$$A = \frac{C_2}{1 - 2^{-C_2}}\left(\frac{1}{C} - \frac{1}{C_1}\right). \tag{3.73}$$

Lemma 8. *The sequence $\{\eta(t)\}_{t=0}^{\infty}$ with parameters s and A satisfying* (3.70) *and* (3.71) *forms a submartingale with respect to the filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$.*

Proof. We will show that $\mathbb{E}[\eta(t+1)|\mathcal{F}_t] \geq \eta(t)$. There are two cases.

Case 1 ($U_i(t) < 0$): there are two subcases. If $U_i(t+1) \geq 0$, then from (3.44c), $U_i(t+1) < C_2$. Consider the function

$$f(x) \triangleq A - Ae^{-sx} - \left(\frac{1}{C} - \frac{1}{C_1}\right) x, \tag{3.74}$$

where s and A satisfy equations (3.70) and (3.71). Since $f(0) = 0$, $f(C_2) = 0$ due to (3.70), and $f(x)$ is a concave function, it follows that $f(x) > 0$ for $x \in (0, C_2)$. Let $U_i(t+1)$ play the role of x. Using $f(U_i(t+1)) > 0$, we obtain

$$\begin{aligned}\eta(t+1) &= -Ae^{-sU_i(t+1)} + \frac{U_i(t+1)}{C_1} - (t+1) \\ &> -A + \frac{U_i(t+1)}{C} - (t+1). \end{aligned} \tag{3.75}$$

If $U_i(t+1) < 0$, then

$$\eta(t+1) = -A + \frac{U_i(t+1)}{C} - (t+1). \tag{3.76}$$

Hence, regardless of the sign of $U_i(t+1)$, it holds that

$$\begin{aligned}\mathbb{E}[\eta(t+1)|\mathcal{F}_t] &\geq \mathbb{E}\Big[-A + \frac{U_i(t+1)}{C} - (t+1)\Big|\mathcal{F}_t\Big] && (3.77)\\ &\geq -A + \frac{U_i(t)+C}{C} - (t+1) && (3.78)\\ &= \eta(t), && (3.79)\end{aligned}$$

where (3.78) follows from (3.44a).

Case 2 ($U_i(t) \geq 0$): there are two subcases. If $U_i(t+1) < 0$, using $f(x)$ defined in (3.74), $f(U_i(t+1)) < 0$. Therefore,

$$\begin{aligned}\eta(t+1) &= -A + \frac{U_i(t+1)}{C} - (t+1) && (3.80)\\ &\geq -Ae^{-sU_i(t+1)} + \frac{U_i(t+1)}{C_1} - (t+1). && (3.81)\end{aligned}$$

If $U_i(t+1) \geq 0$, then

$$\eta(t+1) = -Ae^{-sU_i(t+1)} + \frac{U_i(t+1)}{C_1} - (t+1). \tag{3.82}$$

Hence, regardless of the sign of $U_i(t+1)$, it holds that

$$\mathbb{E}[\eta(t+1)|\mathcal{F}_t] \geq \mathbb{E}\Big[-Ae^{-sU_i(t+1)} + \frac{U_i(t+1)}{C_1} - (t+1)\Big|\mathcal{F}_t\Big] \tag{3.83}$$

$$= -A\mathbb{E}[e^{-sU_i(t+1)}|\mathcal{F}_t] + \frac{\mathbb{E}[U_i(t+1)|\mathcal{F}_t]}{C_1} - (t+1) \tag{3.84}$$

$$= -A\Big(p_1 e^{-s\log\frac{p_1}{1-p_0}} + (1-p_1)e^{-s\log\frac{1-p_1}{p_0}}\Big)e^{-sU_i(t)} + \frac{U_i(t)}{C_1} - t \tag{3.85}$$

$$= \eta(t), \tag{3.86}$$

where (3.85) follows from (3.44b) and (3.65), and (3.86) follows from (3.71).

Summarizing the above two cases, we conclude that $\mathbb{E}[\eta(t+1)|\mathcal{F}_t] \geq \eta(t)$. □

Next, we follow [BZ75] to prove a variant of Doob's optional stopping theorem which will be useful in proving the main result. For a given submartingale $\{U(t)\}_{t=0}^{\infty}$, the original Doob's optional stopping theorem [Wil91, Sec. 10.10] requires that the stopping time T satisfy $\mathbb{E}[T] < \infty$ and $|U(t+1) - U(t)|$ be bounded. In contrast, we show that if T is an upper-threshold-crossing stopping time, then it suffices to ask for T being a.s. finite and $|U(t+1) - U(t)|$ being bounded.

Lemma 9 (Variant of Doob's Optional Stopping Theorem). *Let $\{U(t)\}_{t=0}^{\infty}$ be a submartingale with respect to filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$ satisfying $|U(t+1) - U(t)| \leq K$ for some positive constant K. Let $T = \min\{t : U(t) \geq \zeta\}$, $\zeta > 0$ be a stopping time and assume that T is a.s. finite. Then,*

$$U(0) \leq \mathbb{E}[U(T)]. \tag{3.87}$$

Proof. Let $t \wedge T \triangleq \min\{t, T\}$. From the martingale theory [Wil91, Sec. 10.9], if $\{U(t)\}_{t=0}^{\infty}$ is a submartingale, then the stopped process $\{U(t \wedge T)\}_{t=0}^{\infty}$ is also a submartingale. Thus, we obtain

$$U(0) \leq \mathbb{E}[U(t \wedge T)] \tag{3.88}$$

$$\leq \lim_{t\to\infty} \mathbb{E}[U(t \wedge T)] \tag{3.89}$$

$$
\begin{aligned}
&\leq \mathbb{E}[\lim_{t\to\infty} U(t \wedge T)] \qquad (3.90)\\
&= \mathbb{E}[U(T)].
\end{aligned}
$$

In the above,

- (3.88) follows from applying Doob's optional stopping theorem [Wil91, Sec. 10.10] to the stopped process $\{U(t \wedge T)\}_{t=0}^{\infty}$.
- (3.89) follows from that $\mathbb{E}[U(t \wedge T)] \leq \mathbb{E}[U((t+1) \wedge T)]$ for submartingales. This can be seen by noting that

$$
\begin{aligned}
&\mathbb{E}[U((t+1) \wedge T)] - \mathbb{E}[U(t \wedge T)]\\
&= \mathbb{E}[(U(t+1) - U(t))\mathbf{1}_{\{T \geq t+1\}}] + \mathbb{E}[0 \cdot \mathbf{1}_{\{T \leq n\}}]\\
&= \mathbb{E}\big[\mathbb{E}[(U(t+1) - U(t))\mathbf{1}_{\{T \geq t+1\}} | \mathcal{F}_t]\big]\\
&= \mathbb{E}\big[\mathbf{1}_{\{T \geq t+1\}}\mathbb{E}[(U(t+1) - U(t)) | \mathcal{F}_t]\big]\\
&\geq 0,
\end{aligned}
$$

  where the last step follows from submartingale property $\mathbb{E}[U(t+1)|\mathcal{F}_t] \geq U(t)$.
- (3.90) follows from the fact that $U(t \wedge T)$ is uniformly bounded above, the assumption that T is a.s. finite, and the reverse Fatou's lemma.

This concludes the proof of Lemma 9. □

In the next two lemmas, we show that the sufficient conditions in Lemma 9 indeed holds for the submartingale $\{\eta(t)\}_{t=0}^{\infty}$ in Lemma 8.

Lemma 10. *Let $\{U(t)\}_{t=0}^{\infty}$ be the submartingale in* (3.44) *with respect to filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. Consider the stopping time $T \triangleq \min\{t : U(t) \geq \zeta\}$, where $\zeta > 0$ is some constant. Then, $\mathscr{P}\{T < \infty\} = 1$. Namely, T is a.s. finite.*

Proof. We first recall Azuma-Hoeffding inequality for a general submartingale $\{\xi(t)\}_{t=0}^{\infty}$: If $\{\xi(t)\}_{t=0}^{\infty}$ is a submartingale that satisfies $|\xi(t+1)-\xi(t)| \leq K$ for all $t \geq 0$, then for a given $\sigma > 0$,

$$\mathscr{P}\{\xi(t)-\xi(0) \leq -\sigma\} \leq \exp\left(\frac{-\sigma^2}{2tK^2}\right). \tag{3.91}$$

Let us consider $\xi(t) \triangleq \frac{U(t)}{C} - t$. We show that $\{\xi(t)\}_{t=0}^{\infty}$ is also a submartingale with respect to filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. Specifically, if $U(t) < 0$, then

$$\begin{aligned}
\mathbb{E}[\xi(t+1)|\mathcal{F}_t] &= \frac{\mathbb{E}[U(t+1)|\mathcal{F}_t]}{C} - (t+1) && (3.92)\\
&\geq \frac{U(t)+C}{C} - (t+1) && (3.93)\\
&= \xi(t). && (3.94)
\end{aligned}$$

If $U(t) \geq 0$, using the fact that $C_1 \geq C$, we can also show that $\mathbb{E}[\xi(t+1)|\mathcal{F}_t] \geq \xi(t)$. Hence, $\{\xi(t)\}_{t=0}^{\infty}$ is a submartingale with respect to filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. Furthermore, for any $t \geq 0$,

$$|\xi(t+1)-\xi(t)| = \left|\frac{U(t+1)-U(t)}{C} - 1\right| \leq \frac{C_2}{C} + 1. \tag{3.95}$$

Let $K = \frac{C_2}{C} + 1$ for shorthand notation. Thus, appealing to Azuma-Hoeffding inequality (3.91),

$$\begin{aligned}
\mathscr{P}\left\{U(t) \leq (t-\sigma)C + U(0)\right\} &= \mathscr{P}\left\{\frac{U(t)}{C} - t - \frac{U(0)}{C} \leq -\sigma\right\} && (3.96)\\
&= \mathscr{P}\left\{\xi(t)-\xi(0) \leq -\sigma\right\} && (3.97)\\
&\leq \exp\left(\frac{-\sigma^2}{2tK^2}\right). && (3.98)
\end{aligned}$$

Equating $\zeta = (t-\sigma)C + U(0)$ yields $\sigma = t - \frac{\zeta - U(0)}{C}$, $t > \frac{\zeta - U(0)}{C}$. Hence,

$$\begin{aligned}
\mathscr{P}\left\{U(t) \leq \zeta\right\} &\leq \exp\left(\frac{-(t-\frac{\zeta-U(0)}{C})^2}{2tK^2}\right) && (3.99)\\
&= \exp\left(-\frac{t}{2K^2} + O(t^{-1})\right). && (3.100)
\end{aligned}$$

It follows that

$$\lim_{t\to\infty} \mathscr{P}\{U(t) \leq \zeta\} \leq \lim_{t\to\infty} \exp\left(-\frac{t}{2K^2} + O(t^{-1})\right) = 0. \tag{3.101}$$

This implies that

$$\begin{aligned} \mathscr{P}\{T = \infty\} &= \lim_{t\to\infty} \mathscr{P}\left(\bigcap_{k=1}^{t} \{U(k) < \zeta\}\right) && (3.102)\\ &\leq \lim_{t\to\infty} \mathscr{P}\{U(t) \leq \zeta\} && (3.103)\\ &= 0. && (3.104)\end{aligned}$$

Namely, $\mathscr{P}\{T < \infty\} = 1$. □

Lemma 11. *The sequence* $\{\eta(t)\}_{t=0}^{\infty}$ *with parameters* s *and* A *satisfying* (3.70) *and* (3.71) *has the property that the difference between* $\eta(t+1)$ *and* $\eta(t)$ *is absolutely bounded. More specifically,*

$$|\eta(t+1) - \eta(t)| \leq A + \frac{2C_2}{C} + 1. \tag{3.105}$$

Proof. We distinguish four cases.

Case 1: $U_i(t) < 0$ and $U_i(t+1) < 0$. In this case,

$$\begin{aligned} |\eta(t+1) - \eta(t)| &= \left|\frac{U_i(t+1) - U_i(t)}{C} - 1\right| \\ &\leq \frac{|U_i(t+1) - U_i(t)|}{C} + 1 \\ &\leq \frac{C_2}{C} + 1 && (3.106)\end{aligned}$$

Case 2: $U_i(t) < 0$ and $U_i(t+1) \geq 0$. In this case, $U_i(t+1) \leq C_2$ by (3.44c), and

$$\begin{aligned} |\eta(t+1) - \eta(t)| &= \left|A(1 - e^{-sU_i(t+1)}) + \frac{U_i(t+1)}{C_1} - \frac{U_i(t)}{C} - 1\right| \\ &\leq A(1 - e^{-sC_2}) + \frac{C_2}{C} + 1. && (3.107)\end{aligned}$$

Case 3: $U_i(t) \geq 0$ and $U_i(t+1) < 0$. In this case, $U_i(t) \leq C_2$ by (3.44c), and

$$\begin{aligned}|\eta(t+1) - \eta(t)| &= \left| A(e^{-sU_i(t)} - 1) + \frac{U_i(t+1)}{C} - \frac{U_i(t)}{C_1} - 1 \right| \\ &\leq A|1 - e^{-sU_i(t)}| + \left| \frac{U_i(t+1) - U_i(t)}{C} + \left(\frac{1}{C} - \frac{1}{C_1} \right) U_i(t) \right| + 1 \\ &\leq A(1 - e^{-sC_2}) + \frac{C_2}{C} + \left(\frac{1}{C} - \frac{1}{C_1} \right) C_2 + 1. \end{aligned} \tag{3.108}$$

Case 4: $U_i(t) \geq 0$ and $U_i(t+1) \geq 0$. In this case,

$$\begin{aligned}|\eta(t+1) - \eta(t)| &= \left| -A\big(e^{-sU_i(t+1)} - e^{-sU_i(t)}\big) + \frac{U_i(t+1) - U_i(t)}{C_1} - 1 \right| \\ &\leq A\big|e^{-sU_i(t+1)} - e^{-sU_i(t)}\big| + \frac{|U_i(t+1) - U_i(t)|}{C_1} + 1 \\ &\leq A(1 - e^{-sC_2}) + \frac{C_2}{C_1} + 1, \end{aligned} \tag{3.109}$$

where (3.109) follows from the inequality $|e^{-sy} - e^{-sx}| \leq 1 - e^{-s|y-x|}$ for $s \geq 0$, $x \geq 0$ and $y \geq 0$.

Note that the upper bounds in (3.106), (3.107), (3.108) and (3.109) are no greater than $A + \frac{2C_2}{C} + 1$. The proof is completed. □

We are now in a position to derive a non-asymptotic upper bound on $\mathbb{E}[\tau]$. Let us consider the stopping time

$$\tau_i \triangleq \min \left\{ t \in \mathbb{N} : U_i(t) \geq \log \frac{1-\epsilon}{\epsilon} \right\}. \tag{3.110}$$

Lemmas 10 and 11 indicate that the submartingale $\{\eta(t)\}_{t=0}^{\infty}$ in (3.69) with parameters s and A given by (3.72) and (3.73) and the stopping time τ_i in (3.110) meet the conditions in Lemma 9. Hence, by Lemma 9,

$$\begin{aligned}\eta(0) &\leq \mathbb{E}[\eta(\tau_i)|\Theta = i] \\ &= \mathbb{E}\left[-Ae^{-sU_i(\tau_i)} + \frac{U_i(\tau_i)}{C_1} - \tau \middle| \Theta = i \right] \end{aligned} \tag{3.111}$$

$$\leq -Ae^{-s(\log\frac{1-\epsilon}{\epsilon}+C_2)} + \frac{\log\frac{1-\epsilon}{\epsilon}+C_2}{C_1} - \mathbb{E}[\tau_i|\Theta = i], \tag{3.112}$$

where (3.112) follows since

$$\mathbb{E}[U_i(\tau)] = \mathbb{E}[U_i(\tau_i - 1)] + \mathbb{E}[U_i(\tau_i) - U_i(\tau_i - 1)] \tag{3.113}$$

$$< \log\frac{1-\epsilon}{\epsilon} + C_2. \tag{3.114}$$

Rewriting (3.112) and substituting s and A with (3.72) and (3.73) respectively yield

$$\begin{aligned}\mathbb{E}[\tau_i|\Theta = i] &\leq -Ae^{-s(\log\frac{1-\epsilon}{\epsilon}+C_2)} + \frac{\log\frac{1-\epsilon}{\epsilon}+C_2}{C_1} - \eta(0)\\ &= -Ae^{-s(\log\frac{1-\epsilon}{\epsilon}+C_2)} + \frac{\log\frac{1-\epsilon}{\epsilon}+C_2}{C_1} + A - \frac{U_i(0)}{C}\\ &< \frac{\log M}{C} + \frac{\log\frac{1-\epsilon}{\epsilon}+C_2}{C_1} + C_2\left(\frac{1}{C} - \frac{1}{C_1}\right)\frac{1-\frac{\epsilon}{1-\epsilon}2^{-C_2}}{1-2^{-C_2}},\end{aligned} \tag{3.115}$$

where we have used the fact that $U_i(0) = -\log(M-1) \leq 0$. Finally,

$$\mathbb{E}[\tau] = \frac{1}{M}\sum_{i=1}^{M}\mathbb{E}[\tau|\Theta = i] \leq \frac{1}{M}\sum_{i=1}^{M}\mathbb{E}[\tau_i|\Theta = i], \tag{3.116}$$

where the last inequality follows since $\tau \leq \tau_i$ for all $i \in [M]$. Finally, combining (3.115) and (3.116) completes the proof of Theorem 12.

3.5.3 Proof of Theorem 13

We prove Theorem 13 by contradiction. Let $S_0^* \subseteq [M]$ be an optimal subset of $[M]$ that minimizes $f(S)$ in (3.46). If the partition $(S_0^*, [M] \setminus S_0^*)$ does not meet (3.42), one can construct another subset $S_0' \subseteq [M]$ from S_0^* such that $f(S_0') < f(S_0^*)$, thus contradicting the assumption that S_0^* minimizes $f(S)$.

Assume that the partition $(S_0^*, [M] \setminus S_0^*)$ does not meet (3.42), there are two cases.

Case 1: the partition $(S_0^*, [M] \setminus S_0^*)$ satisfies $\lambda\pi_0(S_0^*) - \pi_1(S_0^*) < -\min_{i\in[M]\setminus S_0^*}\rho_i$. Let $i^* = \arg\min_{i\in[M]\setminus S_0^*}\rho_i$. Then,

$$f(S_0^*) = \lambda\big(\pi_1(S_0^*) - \lambda\pi_0(S_0^*)\big) > \lambda\rho_{i^*}. \tag{3.117}$$

Consider a new subset $S_0' \triangleq S_0^* \cup \{i^*\}$. Next, we show that $f(S_0') < f(S_0^*)$. There are two subcases. If $\pi_1(S_0') \geq \lambda\pi_0(S_0')$, then

$$\begin{aligned} f(S_0') &= \lambda\big(\pi_1(S_0') - \lambda\pi_0(S_0')\big) \\ &= \lambda\big(\pi_1(S_0^*) - \rho_{i^*} - \lambda\pi_0(S_0^*) - \lambda\rho_{i^*}\big) \\ &< \lambda\big(\pi_1(S_0^*) - \lambda\pi_0(S_0^*)\big) \qquad (3.118) \\ &= f(S_0^*), \end{aligned}$$

where (3.118) follows since all elements in $\boldsymbol{\rho}$ remain strictly positive during Bayes' update. If $\pi_1(S_0') < \lambda\pi_0(S_0')$, then

$$\begin{aligned} f(S_0') &= \lambda\pi_0(S_0') - \pi_1(S_0') \\ &= \lambda\big(\pi_0(S_0^*) + \rho_{i^*}\big) - \big(\pi_1(S_0^*) - \rho_{i^*}\big) \\ &= \lambda\rho_{i^*} - \big(\pi_1(S_0^*) - \lambda\pi_0(S_0^*) - \rho_{i^*}\big) \\ &< f(S_0^*), \qquad (3.119) \end{aligned}$$

where (3.119) follows from the assumption that $\lambda\pi_0(S_0^*) - \pi_1(S_0^*) < -\rho_{i^*}$ and (3.117). Hence, the optimality assumption of S_0^* is contradicted in Case 1.

Case 2: the partition $(S_0^*, [M] \setminus S_0^*)$ satisfies $\lambda\pi_0(S_0^*) - \pi_1(S_0^*) > \lambda \min_{i \in S_0^*} \rho_i$. Let $i^* = \arg\min_{i \in S_0^*} \rho_i$. Then,

$$f(S_0^*) = \lambda\pi_0(S_0^*) - \pi_1(S_0^*) > \lambda\rho_{i^*}. \qquad (3.120)$$

Consider a new subset $S_0' \triangleq S_0^* \setminus \{i^*\}$. We next show that $f(S_0') < f(S_0^*)$. There are two subcases. If $\pi_1(S_0') \geq \lambda\pi_0(S_0')$, then

$$\begin{aligned} f(S_0') &= \lambda\big(\pi_1(S_0') - \lambda\pi_0(S_0')\big) \\ &= \lambda\big(\pi_1(S_0^*) + \rho_{i^*} - \lambda\pi_0(S_0^*) + \lambda\rho_{i^*}\big) \\ &= \lambda\rho_{i^*} - \lambda\big(\lambda\pi_0(S_0^*) - \pi_1(S_0^*) - \lambda\rho_{i^*}\big) \\ &< f(S_0^*), \qquad (3.121) \end{aligned}$$

where (3.121) follows from the assumption that $\lambda\pi_0(S_0^*) - \pi_1(S_0^*) > \lambda\rho_{i^*}$ and (3.120). If $\pi_1(S_0') < \lambda\pi_0(S_0')$, then

$$\begin{aligned}
f(S_0') &= \lambda\pi_0(S_0') - \pi_1(S_0') \\
&= \lambda\big(\pi_0(S_0^*) - \rho_{i^*}\big) - \pi_1(S_0^*) - \rho_{i^*} \\
&< \lambda\pi_0(S_0^*) - \pi_1(S_0^*) \\
&= f(S_0^*). \qquad (3.122)
\end{aligned}$$

Hence, the optimality assumption of S_0^* is contradicted in Case 2.

In summary, we have shown that if the partition $(S_0^*, [M] \setminus S_0^*)$ does not meet (3.42), the optimality assumption of S_0^* will be contradicted. Therefore, the partition $(S_0^*, [M] \setminus S_0^*)$ must satisfy (3.42). This concludes the proof of Theorem 13.

3.5.4 Proof of Theorem 14

Let us write $\pi_0^{(s)}$ and $\pi_1^{(s)}$ to denote the probabilities of $S_0^{(s)}$ and $S_1^{(s)}$ at iteration s, $s = 1, 2, \dots, M$. We prove Theorem 14 by induction.

Base case: For $s = 1$, $\pi_0^{(1)} = \rho_{j_1}$ and $\pi_1^{(1)} = 0$. Clearly,

$$\lambda\pi_0^{(1)} - \pi_1^{(1)} = \lambda\rho_{j_1} \in [0, \lambda\rho_{j_1}]. \qquad (3.123)$$

Hence, $\pi_0^{(1)}$ and $\pi_1^{(1)}$ meet the condition in (3.42).

Inductive step: Assume that for $s = k$, (3.42) holds for $\pi_0^{(k)}$ and $\pi_1^{(k)}$. We will show that (3.42) will also hold for $\pi_0^{(k+1)}$ and $\pi_1^{(k+1)}$. There are two cases.

Case 1: $\pi_1^{(k)} \geq \lambda\pi_0^{(k)}$. According to Algorithm 4, $\pi_0^{(k+1)} = \pi_0^{(k)} + \rho_{j_{k+1}}$ and $\pi_1^{(k+1)} = \pi_1^{(k)}$. Meanwhile, $\rho_{j_{k+1}} = \min_{i \in S_0^{(k+1)}} \rho_i$ and $\min_{i \in S_1^{(k)}} \rho_i = \min_{i \in S_1^{(k+1)}} \rho_i$. Therefore,

$$\begin{aligned}
\lambda\pi_0^{(k+1)} - \pi_1^{(k+1)} &= \big(\lambda\pi_0^{(k)} - \pi_1^{(k)}\big) + \lambda\rho_{j_{k+1}} \\
&\leq \lambda \min_{i \in S_0^{(k+1)}} \rho_i, \qquad (3.124)
\end{aligned}$$

and

$$\begin{aligned}\lambda\pi_0^{(k+1)} - \pi_1^{(k+1)} &= \left(\lambda\pi_0^{(k)} - \pi_1^{(k)}\right) + \lambda\rho_{j_{k+1}} \\ &\geq -\min_{i\in S_1^{(k+1)}} \rho_i. \end{aligned} \tag{3.125}$$

Hence, (3.42) holds for $\pi_0^{(k+1)}$ and $\pi_1^{(k+1)}$ in Case 1.

Case 2: $\pi_1^{(k)} < \lambda\pi_0^{(k)}$. According to Algorithm 4, $\pi_0^{(k+1)} = \pi_0^{(k)}$ and $\pi_1^{(k+1)} = \pi_1^{(k)} + \rho_{j_{k+1}}$. Meanwhile, $\rho_{j_{k+1}} = \min_{i\in S_1^{(k+1)}} \rho_i$ and $\min_{i\in S_0^{(k)}} \rho_i = \min_{i\in S_0^{(k+1)}} \rho_i$. Therefore,

$$\begin{aligned}\lambda\pi_0^{(k+1)} - \pi_1^{(k+1)} &= \left(\lambda\pi_0^{(k)} - \pi_1^{(k)}\right) - \rho_{j_{k+1}} \\ &\leq \lambda \min_{i\in S_0^{(k+1)}} \rho_i, \end{aligned} \tag{3.126}$$

and

$$\begin{aligned}\lambda\pi_0^{(k+1)} - \pi_1^{(k+1)} &= \left(\lambda\pi_0^{(k)} - \pi_1^{(k)}\right) - \rho_{j_{k+1}} \\ &> -\min_{i\in S_1^{(k+1)}} \rho_i. \end{aligned} \tag{3.127}$$

Hence, (3.42) holds for $\pi_0^{(k+1)}$ and $\pi_1^{(k+1)}$ in Case 2.

In summary, (3.42) holds for $\pi_0^{(k+1)}$ and $\pi_1^{(k+1)}$ at iteration $s = k + 1$. Therefore, when the algorithm terminates, a bipartition of $[M]$ will be formed and the corresponding $\pi_0^{(M)}$ and $\pi_1^{(M)}$ will satisfy (3.42). This completes the proof of Theorem 14.

3.5.5 Proof of Lemma 4

The proof of Lemma 4 includes a construction of a surrogate submartingale and an application of the variant of Doob's optional stopping theorem (Lemma 9).

Let x_i be the input symbol for $\Theta = i$ at time $t + 1$ and define $\bar{x}_i \triangleq 1 - x_i$. Following the derivation of (3.54), for $Y_{t+1} = y$,

$$U_i(t+1) = \log\frac{\rho_i(t+1)}{1-\rho_i(t+1)} \tag{3.128}$$

$$= U_i(t) + \log \frac{P_{Y|X}(y|x_i)}{\sum_{x\in\mathcal{X}} \tilde{\pi}_{x,i}(t)P(y|x)}, \tag{3.129}$$

where $\tilde{\pi}_{x,i}(t)$, $x \in \{0,1\}$, are the extrinsic probabilities defined in (3.56) and (3.57). For brevity, let us define the instantaneous step size

$$w_i(t,y) \triangleq \log \frac{P_{Y|X}(y|x_i)}{\sum_{x\in\mathcal{X}} \tilde{\pi}_{x,i}(t)P(y|x)}. \tag{3.130}$$

From previous analysis in Section 3.5.1, we showed in (3.62) that with the SED encoding rule,

$$\mathbb{E}[W_i(t,Y)|\mathcal{F}_t] \geq C. \tag{3.131}$$

where C is the capacity of the BSC(p).

Here, we seek a *surrogate submartingale* $U_i'(t)$ satisfying the following two conditions:

1. $\forall t \geq 0$ and $\forall y^t$, $U_i'(t) \leq U_i(t)$ with $U_i'(0) = U_i(0)$;

2. $\mathbb{E}[U_i'(t+1)|\mathcal{F}_t] = U_i'(t) + C$.

The motivation behind Condition 1) is that if we consider stopping time

$$\nu_i' \triangleq \min\{t : U_i'(t) \geq 0\}, \tag{3.132}$$

then, Condition 1) implies that $\nu_i \leq \nu_i'$.

Construction of $\{U_i'(t)\}_{t=0}^{\infty}$: Let $U_i'(0) = U_i(0)$. For $t \geq 0$ and $Y_{t+1} = y$,

$$U_i'(t+1) \triangleq U_i'(t) + w_i'(t,y), \tag{3.133}$$

where $w_i'(t,y)$ is defined as

$$w_i'(t,x_i) \triangleq \log 2P_{Y|X}(x_i|x_i) - \frac{P_{Y|X}(\bar{x}_i|x_i)}{P_{Y|X}(x_i|x_i)} \log \frac{1/2}{\sum_{x\in\mathcal{X}} \tilde{\pi}_{x,i}(t)P(\bar{x}_i|x)}, \tag{3.134}$$

$$w_i'(t,\bar{x}_i) \triangleq \log 2P_{Y|X}(\bar{x}_i|x_i) + \log \frac{1/2}{\sum_{x\in\mathcal{X}} \tilde{\pi}_{x,i}(t)P(\bar{x}_i|x)}. \tag{3.135}$$

We see that the only distinction between $w'_i(t,y)$ and $w_i(t,y)$ defined in (3.130) lies in $w'_i(t,x_i) \neq w_i(t,x_i)$. We now show that $\{U'_i(t)\}_{t=0}^{\infty}$ in (3.133) indeed satisfies the two conditions aforementioned. First,

$$\begin{aligned}
\mathbb{E}[W'_i(t,Y)|\mathcal{F}_t] &= P_{Y|X}(x_i|x_i)w'_i(t,x_i) + P_{Y|X}(\bar{x}_i|x_i)w'_i(t,\bar{x}_i) \tag{3.136}\\
&= P_{Y|X}(x_i|x_i)\log 2P_{Y|X}(x_i|x_i) + P_{Y|X}(\bar{x}_i|x_i)\log 2P_{Y|X}(\bar{x}_i|x_i) \tag{3.137}\\
&= C. \tag{3.138}
\end{aligned}$$

This implies that $\{U'_i(t)\}_{t=0}^{\infty}$ is a submartingale satisfying Condition 2). Specifically, $\mathbb{E}[U'_i(t+1)|\mathcal{F}_t] = U'_i(t) + C$.

Next, we show Condition 1) also holds for $\{U'_i(t)\}_{t=0}^{\infty}$. Note that the only difference between $w_i(t,y)$ and $w'_i(t,y)$ is when $y = x_i$, thus, it suffices to show $w'_i(t,x_i) \leq w_i(t,x_i)$. Indeed,

$$\begin{aligned}
w_i(t,x_i) &= \log 2P_{Y|X}(x_i|x_i) + \log\frac{1/2}{\sum_{x\in\mathcal{X}}\tilde{\pi}_{x,i}(t)P(x_i|x)}\\
&\geq \log 2P_{Y|X}(x_i|x_i) + \frac{P_{Y|X}(\bar{x}_i|x_i)}{P_{Y|X}(x_i|x_i)}\log\frac{1/2}{\sum_{x\in\mathcal{X}}\tilde{\pi}_{x,i}(t)P(x_i|x)}\\
&\geq \log 2P_{Y|X}(x_i|x_i) + \frac{P_{Y|X}(\bar{x}_i|x_i)}{P_{Y|X}(x_i|x_i)}\log\frac{\sum_{x\in\mathcal{X}}\tilde{\pi}_{x,i}(t)P(\bar{x}_i|x)}{1/2} \tag{3.139}\\
&= w'_i(t,x_i), \tag{3.140}
\end{aligned}$$

where (3.139) follows from the inequality below. Let us use the shorthand notation $\pi_{x,i} = \pi_{x,i}(t)$, $p = P_{Y|X}(\bar{x}|x)$ and $q = P_{Y|X}(x|x)$. Then,

$$(\tilde{\pi}_{x_i,i}q + \tilde{\pi}_{\bar{x}_i,i}p)(\tilde{\pi}_{x_i,i}p + \tilde{\pi}_{\bar{x}_i,i}q) = (\tilde{\pi}_{x_i,i}q + \tilde{\pi}_{\bar{x}_i,i}p)\big[1 - (\tilde{\pi}_{x_i,i}q + \tilde{\pi}_{\bar{x}_i,i}p)\big] \leq \frac{1}{4} \tag{3.141}$$

with equality if and only if $\tilde{\pi}_{x_i,i} = 1/2$. Note that by Lemma 6, $\tilde{\pi}_{x_i,i}(t) \in [0,1/2]$ for the BSC. As a result, $\sum_{x\in\mathcal{X}}\tilde{\pi}_{x,i}(t)P(\bar{x}_i|x) = -(q-p)\tilde{\pi}_{x_i,i}(t) + q \in [1/2,q]$, implying that $w'_i(t,x_i) \geq \log 2q > 0$. Thus, $w'_i(t,y) \leq w_i(t,y)$ for $y \in \{0,1\}$ and Condition 1) follows.

Finally, we apply Lemma 9 to the surrogate submartingale $\{U'_i(t)\}_{t=0}^{\infty}$ to obtain an upper bound on $\mathbb{E}[\nu'_i]$. Observe that for any $t \geq 0$ and y^t,

$$|w'_i(t,y)| \leq |w_i(t,y)| \leq C_2. \tag{3.142}$$

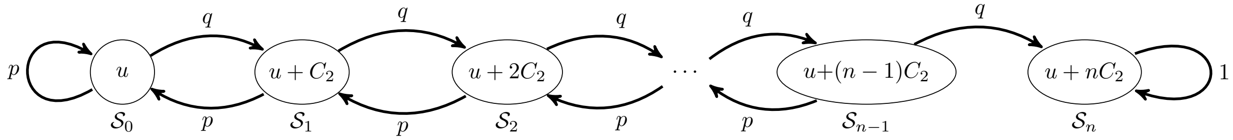

Figure 3.2: An example of the generalized Markov chain with initial value u, $u \in [0, C_2)$, assuming that $U_i(t)$ arrives at u when crossing threshold 0 and remains nonnegative all the time. The value beside each branch denotes the transition probability. The value inside the jth circle represents the unique active value in $\mathcal{S}_j$, $j \in [n]$.

Hence, the conditions in Lemma 9 are met.

Consider a normalized sequence $\{\eta(t)\}_{t=0}^{\infty}$ defined as

$$\eta(t) \triangleq \frac{U_i'(t)}{C} - t. \tag{3.143}$$

It is straightforward to show that $\{\eta(t)\}_{t=0}^{\infty}$ is a martingale with the difference $|\eta(t+1)-\eta(t)|$ bounded from above. Therefore, by Lemma 9,

$$\begin{aligned}
\frac{U_i(0)}{C} &= \eta(0) \\
&\leq \mathbb{E}[\eta(\nu_i')] \\
&= \frac{\mathbb{E}[U_i'(\nu_i') - U_i'(\nu_i'-1)] + \mathbb{E}[U_i'(\nu_i'-1)]}{C} - \mathbb{E}[\nu_i'] \\
&\leq \frac{w_i'(t, x_i) + 0}{C} - \mathbb{E}[\nu_i'] && (3.144) \\
&\leq \frac{\frac{1}{q}\log 2q}{C} - \mathbb{E}[\nu_i'] && (3.145)
\end{aligned}$$

where (3.144) follows from the fact that $U_i'(t)$ has to cross the threshold 0 from $t = \nu_i' - 1$ to $t = \nu_i'$ and that $w_i'(t, x_i)$ is the only positive step size, (3.145) follows from the fact that

$$w_i'(t, x_i) \leq \log 2q - \frac{p}{q}\log\frac{1/2}{q} = \frac{1}{q}\log 2q. \tag{3.146}$$

Combining (3.145) with the fact that $\nu_i \leq \nu_i'$,

$$\mathbb{E}[\nu_i] \leq \mathbb{E}[\nu_i'] \leq \frac{\frac{1}{q}\log 2q}{C} - \frac{\log\frac{1/M}{1-1/M}}{C} \tag{3.147}$$

$$< \frac{\log M + \frac{1}{q}\log 2q}{C}. \tag{3.148}$$

This completes the proof of Lemma 4.

3.5.6 Proof of Lemma 5

The proof requires several steps. First, we show that when $U_i(t) \geq 0$, the behavior of $U_i(t)$ can be modeled as a Markov chain with a fallback self loop. This self loop represents the probability that $U_i(t)$ first falls back to the communication phase and eventually returns to the confirmation phase. Next, we show that $\mathbb{E}[\tau_i - \nu_i | \Theta = i, U_i(\nu_i) = u]$ can be upper bounded by the expected first-passage time from the initial state to the terminating state on a generalized Markov chain. Further upper bounding this expected first-passage time yields the desired upper bound.

Let $q \triangleq 1 - p$. For BSC(p), $p \in (0, 1/2)$, by Fact 3, $C_2 = \log(q/p)$ and $C_1 = (q-p)C_2$. In the following analysis, we fix $\Theta = i \in [M]$ unless otherwise specified.

Recall that with the SED coding scheme, the one-step update for $U_i(t)$ when $U_i(t) \geq 0$ is given by (3.65). In the case of BSC(p), we have

$$U_i(t+1) = U_i(t) + W, \tag{3.149}$$

where $W = C_2$ with probability $q = P_{Y|X}(1|1)$ and $W = -C_2$ with probability $p = P_{Y|X}(0|1)$. Assume that $U_i(\nu_i) = u \in [0, C_2)$. Clearly, the behavior of $U_i(t)$ is a Markov chain with initial value u, provided that $U_i(t) \geq u$ for all $t \geq \nu_i$.

Unfortunately, the above Markov chain is too simple to capture the reality. First, $U_i(t)$ can fall back to the communication phase (i.e., $U_i(t) < 0$) at some time t' where $t' > \nu_i$. Second, if $U_i(t)$ falls back and then returns to the confirmation phase at time t'', $t'' > \nu_i$, the value at which $U_i(t'') \geq 0$ might be different from u.

Nevertheless, we make two important observations. First, the prior probability that $U_i(t)$ falls back to the communication phase is p. Given that $U_i(t)$ falls back, the conditional probability that $U_i(t)$ eventually returns to the confirmation phase is 1, since τ_i is a.s. finite by Lemma 10. Hence, the transition probability from the initial state value u at which $U_i(t)$ falls back to another initial state value u' at which $U_i(t)$ returns is p. Second, assume u is the initial value when $U_i(t)$ first enters the confirmation phase. By (3.149), the subsequent values that $U_i(t)$ assumes are of the form $u + jC_2$, $j \in \mathbb{N}$, provided that $U_i(t) \geq 0$ all the time. These observations motivate the definition of a *generalized Markov chain.*

Definition 4. *Let $\mathcal{S}_0 = [0, C_2)$ represent the set of values of $U_i(t)$ when transitioning from below 0 to above 0 for the first time. Let $n \triangleq \lceil \frac{1}{C_2} \log \frac{1-\epsilon}{\epsilon} \rceil$. Define $\mathcal{S}_j \triangleq [jC_2, jC_2 + C_2)$, $j \in [n]$. The generalized Markov chain consists of a sequence of states $\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_n$ satisfying*

$$\begin{aligned}
\mathscr{P}\{\mathcal{S}_{j+1}|\mathcal{S}_j\} &\triangleq P_{V|U}(u + C_2|u),\ u \in \mathcal{S}_j,\ 0 \leq j \leq n-1,\\
\mathscr{P}\{\mathcal{S}_{j-1}|\mathcal{S}_j\} &\triangleq P_{V|U}(u - C_2|u),\ u \in \mathcal{S}_j,\ 1 \leq j \leq n,\\
\mathscr{P}\{\mathcal{S}_0|\mathcal{S}_0\} &\triangleq P(V \in \mathcal{S}_0|U = u),\ u \in \mathcal{S}_0,\\
\mathscr{P}\{\mathcal{S}_n|\mathcal{S}_n\} &\triangleq 1,
\end{aligned}$$

where if $u \in \mathcal{S}_0$, $P(V \in \mathcal{S}_0|U = u) = p$ and $P(V = u + C_2|U = u) = q$. If $u \in \mathcal{S}_j$, $j \geq 1$, $P(V = u + C_2|U = u) = q$ and $P(V = u + C_2|U = u) = p$.

The distinction between the generalized Markov chain and a conventional Markov chain discussed above is that each state is an interval rather than a single value. However, as soon as $U_i(t) \geq 0$, only a single value in each set $\mathcal{S}_j$ remains active and is uniquely determined by the initial value in $\mathcal{S}_0$. Specifically, if the initial value is u, then the only active value in $\mathcal{S}_j$ is given by $u + jC_2$, $j \in [n]$. For this reason, each state $\mathcal{S}_j$, albeit defined as an interval, resembles a "single value", and one can directly define transition probabilities between two consecutive states. Fig. 3.2 illustrates an example of the generalized Markov chain with an initial value $u \in [0, C_2)$.

Let us consider a new stopping time

$$\tau_i^* \triangleq \min \left\{ t : \left\lfloor \frac{U_i(t)}{C_2} \right\rfloor \geq \left\lceil \frac{\log \frac{1-\epsilon}{\epsilon}}{C_2} \right\rceil \right\}. \tag{3.150}$$

By definition, τ_i^* is independent from the initial value $U_i(t) - \lfloor U_i(t)/C_2 \rfloor C_2$ and is achieved whenever $U_i(t)$ enters $\mathcal{S}_n$ for the first time. Moreover,

$$\frac{U_i(\tau_i^*)}{C_2} \geq \left\lfloor \frac{U_i(\tau_i^*)}{C_2} \right\rfloor \geq \left\lceil \frac{\log \frac{1-\epsilon}{\epsilon}}{C_2} \right\rceil \geq \frac{\log \frac{1-\epsilon}{\epsilon}}{C_2}. \tag{3.151}$$

Hence, by definition of τ_i in (3.49), we obtain

$$\tau_i \leq \tau_i^*. \tag{3.152}$$

This implies that

$$\mathbb{E}[\tau_i - \nu_i | \Theta = i, U_i(\nu_i) = u] \leq \mathbb{E}[\tau_i^* - \nu_i | \Theta = i, U_i(\nu_i) = u]. \tag{3.153}$$

Note that $\mathbb{E}[\tau_i^* - \nu_i | \Theta = i, U_i(\nu_i) = u]$ represents the expected first-passage time from initial state $u \in \mathcal{S}_0$ when $U_i(t)$ first crosses threshold 0 at time ν_i to state $\mathcal{S}_n$. In Appendix 3.9.3, the time of first passage analysis shows that

$$\begin{aligned}\mathbb{E}[\tau_i^* - \nu_i | \Theta = i, U_i(\nu_i) = u] &= \frac{n}{1-2p} + \frac{p}{1-2p}\left(1 - \left(\frac{p}{q}\right)^n\right)\left(\Delta_0 - \frac{2q}{1-2p}\right) &(3.154)\\ &= \frac{nC_2}{C_1} + \frac{2^{-C_2}}{1-2^{-C_2}}(1 - 2^{-nC_2})\left(\Delta_0 - 1 - \frac{C_2}{C_1}\right), &(3.155)\end{aligned}$$

where Δ_0 represents the expected self loop time of $U_i(t)$ from $\mathcal{S}_0$ to $\mathcal{S}_0$. Assume that after fallback, $U_i(t) = u - C_2 < 0$. Following (3.145) in Section 3.5.5, we immediately obtain

$$\begin{aligned}\Delta_0 &\leq 1 + \frac{\frac{1}{q}\log 2q}{C} - \frac{u - C_2}{C} &(3.156)\\ &= 1 + \frac{\frac{1}{q}\log 2q + C_2 - u}{C} &(3.157)\\ &\leq 1 + \frac{\frac{1}{q}\log 2q + C_2}{C}. &(3.158)\end{aligned}$$

Substituting (3.158) into (3.155) yields

$$\mathbb{E}[\tau_i^* - \nu_i | \Theta = i, U_i(\nu_i) = u] \leq \frac{nC_2}{C_1} + C_2 2^{-C_2}\left(\frac{1}{C} - \frac{1}{C_1} + \frac{\frac{1}{q}\log 2q}{CC_2}\right)\frac{1 - 2^{-nC_2}}{1 - 2^{-C_2}}. \tag{3.159}$$

Using $n = \lceil \frac{1}{C_2} \log \frac{1-\epsilon}{\epsilon} \rceil \leq \frac{1}{C_2} \log \frac{1-\epsilon}{\epsilon} + 1$, we obtain the desired upper bound

$$\mathbb{E}[\tau_i^* - \nu_i | \Theta = i, U_i(\nu_i) = u] \leq \frac{\log \frac{1-\epsilon}{\epsilon} + C_2}{C_1} + C_2 2^{-C_2} \left(\frac{1}{C} - \frac{1}{C_1} + \frac{\frac{1}{q} \log 2q}{C C_2} \right) \frac{1 - \frac{\epsilon}{1-\epsilon} 2^{-C_2}}{1 - 2^{-C_2}}. \tag{3.160}$$

Combining (3.153) and (3.160) completes the proof of Lemma 5.

3.6 Numerical Simulation

In this section, we simulate the performance of the generalized SED coding scheme in Sec. 3.3 for a regularized BAC(p_0, p_1) with full noiseless feedback. The empirical rate is computed with (3.12). The achievability bound on rate can be obtained from the non-asymptotic upper bound on $\mathbb{E}[\tau]$. We consider target error probability $\epsilon = 10^{-3}$ and a regularized BAC and a BSC both with capacity 1/2. In each case, we compare the empirical rate attained by the SED coding scheme with our VLF achievability bounds and with previous achievability bounds.

3.6.1 The Regularized BAC with Feedback

Consider the target error probability $\epsilon = 10^{-3}$ and the regularized BAC$(0.03, 0.22)$ with feedback. Using Facts 1 and 3, one can compute

$$C = 0.5, \quad C_1 = 3.1954, \quad C_2 = 4.7. \tag{3.161}$$

For $\epsilon = 10^{-3}$ and the BAC$(0.03, 0.22)$ with feedback, Fig. 3.3 shows the empirical rate achieved by the generalized SED coding scheme, along with the VLF converse bound in Theorem 11, our VLF achievability bound in Theorem 12 and Polyanskiy's VLSF achievability bound in Theorem 10 evaluated with (3.36), (3.37). Since the generalized SED partitioning algorithm has an exponential complexity in message length k, we were only able to simulate message lengths k from 1 to 20. We see in Fig. 3.3 that the empirical rate achieved by

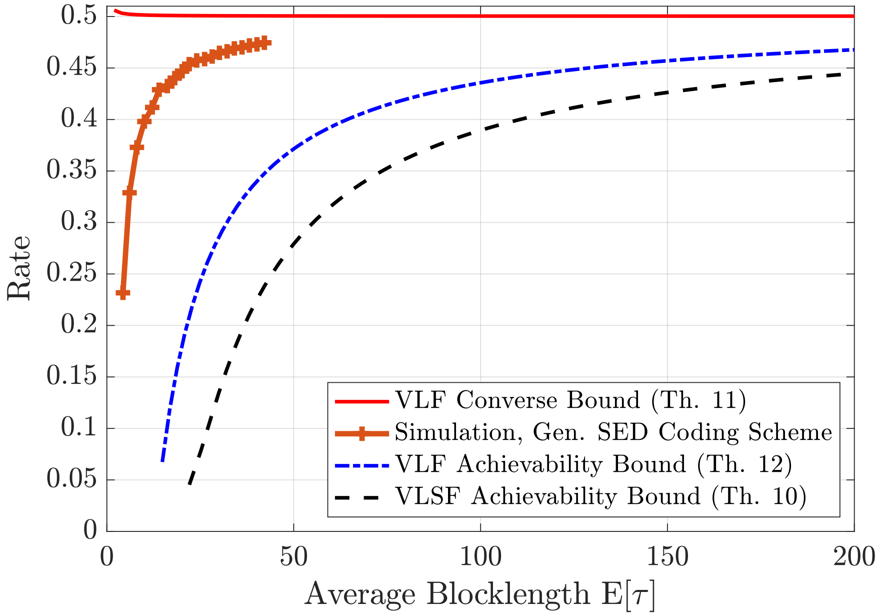

Figure 3.3: The rate as a function of average blocklength for the BAC$(0.03, 0.22)$ with noiseless feedback. Target error probability $\epsilon = 10^{-3}$. In this example, the message length k in the simulation ranges from 1 to 20.

the SED coding scheme is much better than our VLF achievability bound, implying that there is still room for improvement. Nevertheless, our VLF achievability bound outperforms Polyanskiy's VLSF achievability bound as desired.

3.6.2 The BSC with Feedback

Consider the target error probability $\epsilon = 10^{-3}$ and the BSC(0.11) with feedback. Using Facts 1 and 3,

$$C = 0.5, \quad C_1 = 2.3527, \quad C_2 = 3.0163. \tag{3.162}$$

One can verify that this setting satisfies the technical conditions in [NJW15]. Thus, by Theorem 18 of Naghshvar *et al.*,

$$\mathbb{E}[\tau] \leq \frac{\log M + \log\log M}{0.5} + 5352.67, \tag{3.163}$$

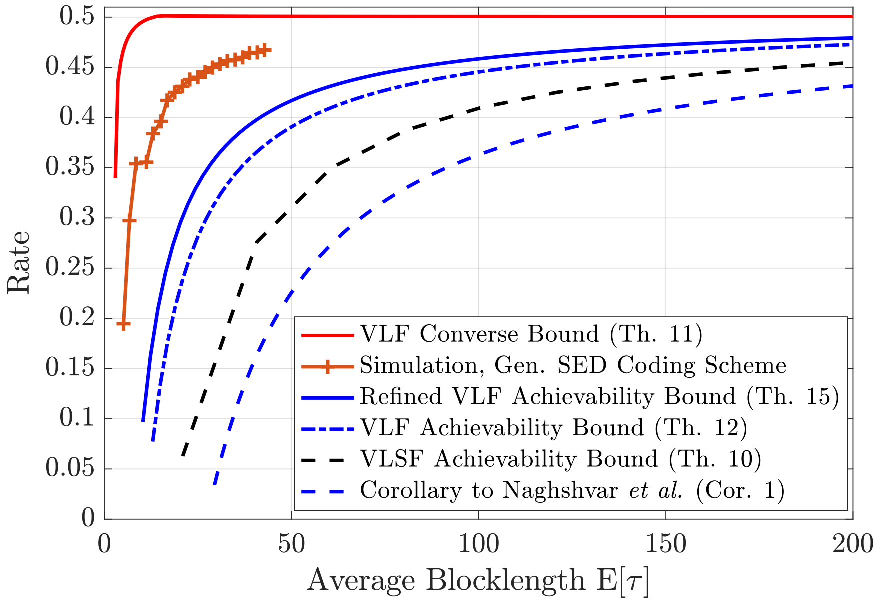

Figure 3.4: The rate as a function of average blocklength over the BSC(0.11) with noiseless feedback. Target error probability $\epsilon = 10^{-3}$.

which turns out to be an extremely loose upper bound on $\mathbb{E}[\tau]$. The corresponding achievability bound even falls out of the average blocklength region of interest, thus is omitted from the simulation plot.

For $\epsilon = 10^{-3}$ and BSC(0.11) with feedback, Fig. 3.4 shows the empirical rate achieved by the generalized SED coding scheme, along with the VLF converse bound in Theorem 11, our refined VLF achievability bound for BSC in Theorem 15, the VLF achievability bound for the BSC in Theorem 12, Polyanskiy's VLSF achievability bound in Theorem 10, and the VLF achievability bound in Corollary 1. Thanks to Polyanskiy, we directly evaluate the VLSF achievability bound in Theorem 10, rather than using relaxations in (3.36) and (3.37). Due to the exponential complexity of the SED partitioning algorithm, we were only able to simulate message lengths k from 1 to 20.

Despite that Corollary 1 is a better result compared to Theorem 18, the resulting VLF achievability bound still falls beneath Polyanskiy's VLSF achievability bound. In contrast, our VLF achievability bound in Theorem 12 exceeds Polyanskiy's VLSF achievability bound in Theorem 10 as desired. Indeed, this should be expected since a system that employs full noiseless feedback should perform better than a system that employs stop feedback. In particular, our refined VLF achievability bound for the BSC in Theorem 15 is a further improvement compared to Theorem 12.

3.7 Implications on the Reliability Function

In this section, we show that for a regularized BAC with feedback, the deterministic VLF code constructed with our generalized SED coding scheme described in Sec. 3.3 asymptotically achieves both capacity and Burnashev's optimal error exponent.

Let $\mathfrak{c}$ be a variable-length coding scheme such that for each positive number l, one out of $M_{\mathfrak{c}_l}$ equiprobable messages is transmitted at an error probability $P_{e,\mathfrak{c}}$ and with an average blocklength $\mathbb{E}_{\mathfrak{c}_l}[\tau]$. We say that the scheme $\mathfrak{c}$ achieves rate R if for any small numbers $\delta > 0$, $\epsilon \in [0, 1)$ and all sufficiently large l, the following three conditions hold:

$$P_{e,\mathfrak{c}_l} \leq \epsilon, \tag{3.164a}$$

$$M_{\mathfrak{c}_l} \geq 2^{l(R-\delta)}, \tag{3.164b}$$

$$\mathbb{E}_{\mathfrak{c}_l}[\tau] \leq l. \tag{3.164c}$$

Furthermore, if the scheme $\mathfrak{c}$ satisfies (3.164b), (3.164c) and a stronger condition

$$P_{e,\mathfrak{c}_l} \leq 2^{-l(E-\delta)}, \tag{3.165}$$

for some positive real number E, then we say the scheme $\mathfrak{c}$ achieves error exponent E at rate R.

We invoke a general result from [NJW15] to show our claim.

Lemma 12 (Lemma 4, [NJW15])**.** *Suppose that we have a VLF coding scheme $\mathfrak{c}$ that for each message size $M \in \mathbb{N}_+$ and each positive $\epsilon \in (0, 1)$, satisfies $P_{e,\mathfrak{c}} \leq \epsilon$ with expected* *stopping time*

$$\mathbb{E}_{\mathfrak{c}}[\tau] \leq \left(\frac{\log M}{R_{\min}} + \frac{\log \frac{1}{\epsilon}}{E_{\min}}\right)\big(1 + o(1)\big) \tag{3.166}$$

for some positive numbers $E_{\min}$ and $R_{\min}$, where $o(1) \to 0$ as $\epsilon \to 0$ or $M \to \infty$. Then, the scheme $\mathfrak{c}$ can achieve any rate $R \in [0, R_{\min}]$ with error exponent E, if

$$E \leq E_{\min}\left(1 - \frac{R}{R_{\min}}\right). \tag{3.167}$$

Observe that in Theorem 12 and Theorem 15, both upper bounds can be relaxed and written in the form of

$$\frac{\log M}{C} + \frac{\log \frac{1}{\epsilon}}{C_1} + \frac{K(C, C_1, C_2)}{CC_1}, \tag{3.168}$$

where $K(C, C_1, C_2)$ is a constant that only relies on C, C_1, C_2. Hence, for sufficiently large M or sufficiently small ϵ,

$$\frac{K(C, C_1, C_2)}{CC_1} \leq \frac{C_1 \log M + C \log \frac{1}{\epsilon}}{CC_1} = \frac{\log M}{C} + \frac{\log \frac{1}{\epsilon}}{C_1}.$$

Another way to write the above inequality is that

$$\frac{K(C, C_1, C_2)}{CC_1} = \left(\frac{\log M}{C} + \frac{\log \frac{1}{\epsilon}}{C_1}\right) o(1), \tag{3.169}$$

where $o(1) \to 0$ as $\epsilon \to 0$ or $M \to \infty$. This implies that the non-asymptotic upper bounds in both Theorem 12 and Theorem 15 meet the condition in Lemma 12. Therefore, the SED encoding scheme can achieve any rate $R \in [0, C]$ with error exponent

$$E \leq C_1 \left(1 - \frac{R}{C}\right), \tag{3.170}$$

thus the claim is proved.

3.8 Conclusion

In this chapter, we proposed a generalized SED coding scheme for the regularized BAC with full noiseless feedback. For a BSC with feedback, our generalized SED coding scheme is a relaxation of Naghshvar *et al.*'s SED coding scheme. This chapter develops a non-asymptotic VLF achievability bound for the deterministic VLF code constructed with the generalized SED coding scheme. For the BSC, we develop another refined VLF achievability bound using a two-phase analysis. Numerical evaluations show that our VLF achievability bounds outperform Polyanskiy's VLSF achievability bound as desired. In summary, the SED coding scheme is a powerful tool that helps facilitate stronger VLF achievability bounds.

Two important technical ingredients lead our generalized SED coding scheme to a non-asymptotic VLF achievability bound that asymptotically achieves both capacity and Burnashev's optimal error exponent. The first ingredient is that our scheme guarantees that at each time t, the extrinsic probability $\tilde{\pi}_{x_i,i}(t)$ for the input symbol x_i associated with the transmitted message $\Theta = i \in [M]$ is always upper bounded by $\pi^*_{x_i}$; see Lemma 6. This is key to achieving capacity C. The second ingredient is that for regularized BAC, $0 < \pi^*_1 \leq 1/2$ and transmitting input symbol 1 achieves C_1 i.e., $D(P(Y|X=1)\|P(Y|X=0)) > D(P(Y|X=0)\|P(Y|X=1))$. This guarantees that once the transmitted message $\Theta = i$ has posterior probability at least 1/2, our scheme exclusively partitions the message set $[M]$ into $S_1 = \{i\}$ and $S_0 = [M] \setminus \{i\}$, thus achieving the C_1 constant.

However, the challenge for extending the SED coding scheme to an arbitrary binary-input DMC (B-DMC) mainly lies in the second technical ingredient. Namely, does the input symbol $x \in \{0,1\}$ that achieves C_1 constant satisfy $\pi^*_x \leq 1/2$ for an arbitrary B-DMC? Unfortunately, we discovered counterexamples that give a negative answer to the above problem. For example, consider the following B-DMC with $\mathcal{X} = \{0,1\}$, $\mathcal{Y} = \{0,1,2\}$, and transition matrix

$$\begin{bmatrix} 0.5645 & 0.3687 & 0.0668 \\ 0.3714 & 0.0212 & 0.6074 \end{bmatrix}. \tag{3.171}$$

The capacity-achieving input distribution identified by the Blahut-Arimoto algorithm [Ari72, Bla72] is given by

$$(\pi^*_0, \pi^*_1) = (0.50507, 0.49493), \tag{3.172}$$

achieving capacity $C = 0.33$. Two KL divergences are computed as

$$D\big(P(Y|X=0)\|P(Y|X=1)\big) = 1.6474, \tag{3.173}$$

$$D\big(P(Y|X=1)\|P(Y|X=0)\big) = 1.6227. \tag{3.174}$$

This implies that transmitting input symbol 0 achieves C_1 constant, yet $\pi_0^* > 1/2$. This example suggests that new analysis is required in order to extend the SED coding scheme to an arbitrary B-DMC.

Acknowledgment

The material in this chapter is based on the revised version of preprint [YPA22] that was recently accepted for publication in the *IEEE Transactions on Information Theory.* An earlier version of this work was presented at the 2020 IEEE International Symposium on Information Theory [YW20].

3.9 Appendices

3.9.1 Proof of Fact 1

Let $(\pi_0, 1-\pi_0)$ be an input distribution to a BAC(p_0, p_1). Hence, Y is also a binary random variable with

$$\mathscr{P}\{Y=0\} = \pi_0(1-p_0) + (1-\pi_0)p_1. \tag{3.175}$$

Therefore, the mutual information $I(\pi_0)$ between X and Y is given by

$$\begin{aligned} I(\pi_0) &= h\big(\pi_0(1-p_0) + (1-\pi_0)p_1\big) - \pi_0 h(p_0) - (1-\pi_0)h(p_1) \\ &= h\big(\pi_0(1-p_0-p_1) + p_1\big) - \pi_0\big(h(p_0) - h(p_1)\big) - h(p_1). \end{aligned} \tag{3.176}$$

Since mutual information $I(\pi_0)$ is concave in $\pi_0 \in [0,1]$ [CT06b, Theorem 2.7.4], the optimal π_0^* satisfies $I'(\pi_0^*) = 0$. The first derivative of $I(\pi_0)$ is given by

$$I'(\pi_0) = (1-p_0-p_1)\log\left(\frac{1}{\pi_0(1-p_0-p_1)+p_1} - 1\right) - \big(h(p_0) - h(p_1)\big). \tag{3.177}$$

Clearly, $I'(\pi_0)$ is a monotonically decreasing function in $\pi_0 \in (0,1)$. Let $z \triangleq 2^{\frac{h(p_0)-h(p_1)}{1-p_0-p_1}}$. By setting $I'(\pi_0) = 0$, we obtain π_0^* in (3.2). Using (3.176) and the relation $\pi_1^* = 1 - \pi_0^*$, we obtain capacity C in (3.1) and π_1^* in (3.3).

If $p_0 \in (0, 1/2)$ and $p_0 \le p_1 \le 1-p_0$, it is straightforward to see that $C > 0$. Note that $I(0) = I(1) = 0$. This implies that $\pi_0^* \in (0,1)$ and $\pi_1^* \in (0,1)$.

To show that $p_0 \in (0,1/2)$ and $p_0 \le p_1 \le 1-p_0$ imply $\pi_0^* \ge 1/2$, it suffices to show that

$$I'\left(\frac{1}{2}\right) \ge 0. \tag{3.178}$$

Note that

$$I'\left(\frac{1}{2}\right) = -(1-p_0-p_1)\log\left(\frac{1}{\frac{(1-p_1)+p_0}{2}} - 1\right) - h(p_0) + h(1-p_1). \tag{3.179}$$

Therefore, it is equivalent to showing that

$$h(1-p_1) \geq h(p_0) + (1-p_1-p_0)\log\left(\frac{1}{\frac{(1-p_1)+p_0}{2}} - 1\right). \tag{3.180}$$

Let us fix $p_0 \in (0, 1/2)$ and define $x \triangleq 1 - p_1 \in [p_0, 1-p_0]$. Then, (3.180) simplifies to

$$h(x) \geq h(p_0) + (x-p_0)\log\left(\frac{1}{\frac{x+p_0}{2}} - 1\right). \tag{3.181}$$

In order to show (3.181), we introduce the following useful lemma.

Lemma 13. *Let* $f : (0,1) \to \mathbb{R}$ *be convex in* $(0, 1/2]$ *and be concave in* $[1/2, 1)$*. Additionally,* $f(x) = -f(1-x)$*. Then,* $\forall x, y \in (0,1)$ *with* $x + y < 1$*,*

$$f(x) + f(y) \geq 2f\left(\frac{x+y}{2}\right). \tag{3.182}$$

Proof. Without loss of generality, assume that $x < y$. If $y \leq 1/2$, (3.182) directly follows from convexity of $f(x)$ in $x \in (0, 1/2]$. Now consider $y > 1/2$. Therefore,

$$\begin{aligned}
f\left(\frac{x+y}{2}\right) - f(y) &= f\left(\frac{y+x}{2}\right) - f\left(1/2\right) + f\left(1/2\right) - f(y) \\
&\leq f\left(\frac{1-y+x}{2}\right) - f(1-y) + f\left(1/2\right) - f(y) \qquad (3.183)\\
&= f(1/2) - f\left(\frac{1+y-x}{2}\right) \qquad (3.184)\\
&\leq f\left(1 - \frac{x+y}{2}\right) - f(1-x) \qquad (3.185)\\
&= f(x) - f\left(\frac{x+y}{2}\right). \qquad (3.186)
\end{aligned}$$

In the above,

- (3.183) follows from the convexity property that for a fixed $\delta > 0$,

$$f(x) - f(x+\delta) \leq f(y) - f(y+\delta), \text{ whenever } x \geq y,$$

- (3.184) and (3.186) follow from $f(x) = -f(1-x)$,

- (3.185) follows from the concavity property that for a fixed $\delta > 0$,

$$f(x) - f(x+\delta) \leq f(y) - f(y+\delta), \text{ whenever } x \leq y.$$

This completes the proof of Lemma 13. □

We are now in a position to prove (3.181). Let $g(x) \triangleq \log(1/x - 1)$, $x \in [p_0, 1-p_0]$. Observe that $g(x)$ meets the conditions in Lemma 13 and $h'(x) = g(x)$. Hence, appealing to Lemma 13, we obtain

$$\begin{aligned}
h(x) &= h(p_0) + \int_{p_0}^{x} g(z)\,\mathrm{d}z && (3.187)\\
&= h(p_0) + \int_{p_0}^{\frac{x+p_0}{2}} \Big(g(z) + g(x+p_0-z)\Big)\,\mathrm{d}z && (3.188)\\
&\geq h(p_0) + \int_{p_0}^{\frac{x+p_0}{2}} 2g\left(\frac{x+p_0}{2}\right)\mathrm{d}z && (3.189)\\
&= h(p_0) + (x-p_0)\log\left(\frac{1}{\frac{x+p_0}{2}} - 1\right). && (3.190)
\end{aligned}$$

This implies that (3.181) indeed holds. Hence, $\pi_0^* \geq 1/2$, concluding the proof of Fact 1.

3.9.2 Proof of Fact 3

For brevity, let us define two distributions

$$\begin{aligned}
P &\triangleq P(Y|X=0) = [1-p_0, p_0], && (3.191)\\
Q &\triangleq P(Y|X=1) = [p_1, 1-p_1]. && (3.192)
\end{aligned}$$

Hence, it is equivalent to show that

$$D\big(Q\|P\big) \geq D\big(P\|Q\big). \qquad (3.193)$$

Let us define the function

$$f(p_0, p_1) \triangleq D\big(Q\|P\big) - D\big(P\|Q\big)$$

$$= (1 - p_0 + p_1) \log \frac{p_1}{1 - p_0} + (1 + p_0 - p_1) \log \frac{1 - p_1}{p_0}. \tag{3.194}$$

The first and second derivatives with respect to p_1 are, respectively, given by

$$\frac{\partial f}{\partial p_1} = \log \frac{p_1}{1 - p_0} - \log \frac{1 - p_1}{p_0} + (\log e)\Big(\frac{1 - p_0}{p_1} - \frac{p_0}{1 - p_1}\Big) \tag{3.195}$$

$$\frac{\partial^2 f}{\partial p_1^2} = \frac{-(\log e)(2p_1 - 1)(p_1 - 1 + p_0)}{p_1^2(1 - p_1)^2} \tag{3.196}$$

$$\begin{cases} < 0, & \text{if } p_1 \in [p_0, 1/2) \\ \geq 0, & \text{if } p_1 \in [1/2, 1 - p_0]. \end{cases} \tag{3.197}$$

Hence, for a given $p_0 \in (0, 1/2)$, $f(p_0, p_1)$ is concave in $p_1 \in [p_0, 1/2]$ and is convex in $p_1 \in [1/2, 1 - p_0]$. Next, we borrow a classical result in analysis [Rud76].

Lemma 14. *Consider a function $\phi : I \to \mathbb{R}$ defined on an interval $I \triangleq [a, b]$ with $a < b$. If the first derivative $\phi'(x)$ is continuous on I and the second derivative $\phi''(x)$ exists for every $x \in I^o \triangleq (a, b)$, then the following two properties hold*

1) if $\phi''(x) \geq 0$, $x \in I^o$, and $\phi'(x^*) = 0$ for some $x^* \in I$, then $\phi(x) \geq \phi(x^*)$ for all $x \in I$.

2) If $\phi''(x) \leq 0$, $x \in I^o$, then $\phi(x) \geq \min\{\phi(a), \phi(b)\}$ for all $x \in I$.

By Lemma 14, for $p_1 \in [p_0, 1/2]$, due to concavity,

$$f(p_0, p_1) \geq \min\{f(p_0, p_0), f(p_0, 1/2)\} \tag{3.198}$$

$$= \min\{0, f(p_0, 1/2)\}. \tag{3.199}$$

Similarly, for $p_1 \in [1/2, 1 - p_0]$, due to convexity and the fact that $\frac{\partial f}{\partial p_1}\big|_{p_1 = 1 - p_0} = 0$,

$$f(p_0, p_1) \geq f(p_0, 1 - p_0) = 0, \tag{3.200}$$

implying that $f(p_0, 1/2) \geq 0$. Combining this with (3.199) and (3.200), we conclude that $f(p_0, p_1) \geq 0$ for all $p_1 \in [p_0, 1 - p_0]$, thus establishing (3.193). This completes the proof of (3.7).

Next, we prove (3.8). This is equivalent to showing that $p_1(1 - p_1) \geq p_0(1 - p_0)$, which clearly holds when $p_0 \leq p_1 \leq 1 - p_0$.

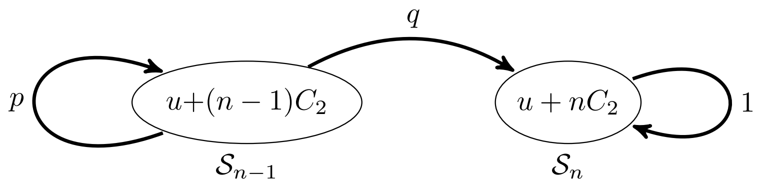

Figure 3.5: An equivalent Markov chain from $\mathcal{S}_{n-1}$ to $\mathcal{S}_n$.

3.9.3 Time of First Passage Analysis

In this section, we compute the expected first-passage time from $\mathcal{S}_0$ to $\mathcal{S}_n$ for the generalized Markov chain, as depicted in Fig. 3.2. Consider the general case where the self loop for state $\mathcal{S}_0$ has weight Δ_0 (i.e., the expected self loop time from $\mathcal{S}_0$ to $\mathcal{S}_0$), and all other transitions in graph has weight 1. Let v_i denote the expected first-passage time from $\mathcal{S}_i$ to $\mathcal{S}_n$, $0 \leq i \leq n-1$. Our goal is to compute v_0, which is equal to $\mathbb{E}[\tau_i^* - \nu_i | \Theta = i, U_i(\nu_i) = u]$.

This appendix computes v_0 by first simplifying the expected first-passage time node equations into an expression involving only v_0 and v_{n-1}. Characterizing the entire process to the left of $\mathcal{S}_{n-1}$ as a self loop with weight Δ_{n-1} yields an explicit expression for v_{n-1}. This eventually produces an expression for v_0 that naturally decomposes into the expected first-passage time for a classical random walk plus a differential term.

3.9.3.1 Simplifying Node Equations

Following [Gal13, Chapter 4.5.1], the node equations for the generalized Markov chain in Fig. 3.2 are as follows:

$$v_{n-1} = 1 + pv_{n-2}, \tag{3.201a}$$

$$v_{n-2} = 1 + pv_{n-3} + qv_{n-1}, \tag{3.201b}$$

$$v_{n-3} = 1 + pv_{n-4} + qv_{n-2}, \tag{3.201c}$$

$$\vdots$$

$$v_2 = 1 + pv_1 + qv_3, \tag{3.201d}$$

$$v_1 = 1 + pv_0 + qv_2, \tag{3.201e}$$

$$v_0 = q + pv_0 + qv_1 + p\Delta_0. \tag{3.201f}$$

Summing all equations in (3.201) yields

$$\sum_{i=0}^{n-1} v_i = n - 1 + q + \sum_{i=1}^{n-2} v_i + qv_{n-1} + 2pv_0 + p\Delta_0. \tag{3.202}$$

Solving for v_0 yields

$$v_0 = \frac{n}{1-2p} + \frac{p}{1-2p}(\Delta_0 - v_{n-1} - 1). \tag{3.203}$$

Thus, what remains to determine v_0 is to determine v_{n-1}.

3.9.3.2 Expressing v_{n-1} in Terms of Δ_0

In this subsection, we aim to express v_{n-1} in terms of Δ_0. By characterizing the entire process to the left of $\mathcal{S}_{n-1}$ as a self loop with weight Δ_{n-1} and transition probability p, the generalized Markov chain in Fig. 3.2 reduces to a two-state Markov chain as shown in Fig. 3.5. The node equation at $\mathcal{S}_{n-1}$ in Fig. 3.5 is given by

$$v_{n-1} = p\Delta_{n-1} + q + pv_{n-1}. \tag{3.204}$$

Solving for v_{n-1} yields

$$v_{n-1} = \frac{p}{q}\Delta_{n-1} + 1. \tag{3.205}$$

Next, we develop a recursive equation to solve Δ_{n-1}. Let Δ_i denote the expected self loop weight for $\mathcal{S}_i$, $0 \leq i \leq n-1$. Fig. 3.6 shows the transition between $\mathcal{S}_{i-1}$ and $\mathcal{S}_i$ conditioned on circling over $\mathcal{S}_i$ once. Thus, from Fig. 3.6, we obtain

$$\Delta_i = 1 + \sum_{k=0}^{\infty} p^k q(k\Delta_{i-1} + 1) \tag{3.206}$$

$$= 2 + \frac{p}{q}\Delta_{i-1}. \tag{3.207}$$

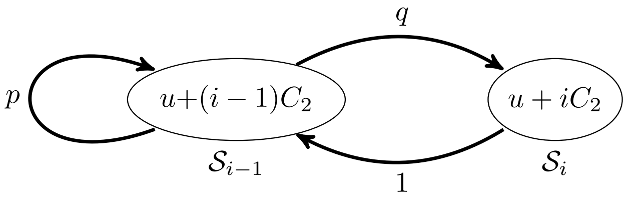

Figure 3.6: Recursive relation between Δ_i and Δ_{i-1}.

Since (3.207) holds for an arbitrary i, $0 \le i \le n-1$, applying (3.207) in a recursive manner yields

$$\Delta_{n-1} = \left(\frac{p}{q}\right)^{n-1} \Delta_0 + \frac{2q}{1-2p}\left[1-\left(\frac{p}{q}\right)^{n-1}\right]. \tag{3.208}$$

Substituting (3.208) into (3.205), we obtain

$$v_{n-1} = \left(\frac{p}{q}\right)^{n} \Delta_0 + \frac{2p}{1-2p}\left[1-\left(\frac{p}{q}\right)^{n-1}\right] + 1. \tag{3.209}$$

3.9.3.3 Finding the General Expression for v_0

Substituting (3.209) into (3.203),

$$\begin{aligned} v_0 &= \frac{n}{1-2p} + \frac{p}{1-2p}\left\{\left[1-\left(\frac{p}{q}\right)^{n}\right]\Delta_0 - \frac{2p}{1-2p}\left[1-\left(\frac{p}{q}\right)^{n-1}\right] - 2\right\} && (3.210)\\ &= \frac{n}{1-2p} + \frac{p}{1-2p}\left(1-\left(\frac{p}{q}\right)^{n}\right)\left(\Delta_0 - \frac{2q}{1-2p}\right). && (3.211)\end{aligned}$$

This completes the derivation of v_0.

Remark 4. *For an i.i.d. random walk that moves forward by* 1 *with probability* q *and moves backward by* 1 *with probability* p*, all* Δ_i*'s are identical. Using* (3.207)*, we obtain*

$$\Delta_i = \frac{2q}{1-2p}, \quad \forall i \in \mathbb{Z}. \tag{3.212}$$

Thus, (3.211) *can be thought of as the expected first-passage time for an i.i.d. random walk plus a differential term that depends on the difference between the self loop weight* Δ_0 *of the actual random process and the self loop weight of a standard i.i.d. random walk.*

CHAPTER 4

BI-AWGN Channels With Finite, Stop Feedback

4.1 Introduction

Feedback has been shown to be useful both in the variable-length and fixed-length regimes, even though it does not improve the capacity of a memoryless, point-to-point channel [Sha56]. In the variable-length regime, feedback has been shown to simplify the construction of coding schemes [Hor63, SK66, SF11, NWJ12, NJW15, YPA22], to significantly improve the optimal error exponent [Bur76], and to achieve universality [Lub02, DFK04, YKE21a]. In the fixed-length regime, feedback is shown to improve the second-order coding rate for the compound-dispersion discrete memoryless channels [WSA20].

In [PPV11], Polyanskiy *et al.* introduced variable-length feedback (VLF) codes, variable-length feedback with termination (VLFT) codes, and a special VLF code called a *variable-length stop-feedback (VLSF)* code. The infinite-length VLSF codewords are fixed before the start of transmission and feedback only affects the portion of a codeword being transmitted rather than the value of that codeword. During transmission, a feedback symbol “0” indicates that the decoder is not ready to decode and the transmission should continue, whereas a “1” signifies that the decoder is ready to decode and the transmitter must stop. Using VLSF codes, Polyanskiy *et al.* demonstrated that $\frac{C}{1-\epsilon}$ is achievable by stopping the code at $\tau = 0$ with a small probability, where C denotes channel capacity, and ϵ denotes the target error probability [PPV11].

The VLSF code defined in [PPV11] can be thought of as a VLF code with infinitely many decoding times, i.e., the number of decoding times $m = \infty$. However, in practical systems, the feedback opportunities are limited, i.e., $m < \infty$, and the decoder is only allowed to decode at time instants $n_1, n_2, \ldots, n_m$. In [KSL15], Kim *et al.* investigated VLSF codes with m periodic decoding times and derived a lower bound on throughput. In order to minimize the average blocklength, Vakilinia *et al.* [VRD16] developed the *sequential differential optimization* (SDO) procedure that produces decoding time n_{i+1} based on the knowledge of n_i, n_{i-1}, and their successful decoding probabilities approximated by a differentiable function. The SDO in [VRD16] uses the Gaussian tail probability to approximate the probability of successful decoding. Later, variations of SDO were developed to improve the Gaussian model accuracy [WWB17, WWB18]. The SDO algorithm is used to optimize systems that employ incremental redundancy and hybrid automatic repeat request (ARQ) [WVW17], and to code for the binary erasure channel [HCP18, HCW19]. However, in this chapter, we show that the Gaussian model is still imprecise for small values of n. Additionally, the existing SDO procedure fails to consider the inherent gap constraint that two decoding times must be separated by at least one.

In [YKE21b], Yavas *et al.* developed an achievability bound for VLSF codes with m decoding times for the additive white Gaussian noise channel with capacity C, dispersion V, and maximal power constraint P. The asymptotic expansion of the maximum message size M is given by $\ln M \approx \frac{lC}{1-\epsilon} - \sqrt{l \ln_{(m-1)}(l) \frac{V}{1-\epsilon}}$ where $\ln_{(k)}(\cdot)$ denotes the k-fold nested logarithm, l and ϵ are the upper bounds on average blocklength and error probability of the VLSF code, respectively. They showed that a slight increase in m can dramatically improve the achievable rate of VLSF codes. Unfortunately, due to the nested logarithm term, Yavas *et al.* were only able to show achievability bounds for $m \leq 4$, $\epsilon = 10^{-3}$, and $l \leq 2000$. They also demonstrated that within their code construction, the decoding times chosen by the SDO will yield the same second- and third-order coding rates as attained by their construction of decoding times.

In this chapter, we are interested in the performance of a VLSF code with m optimal decoding times for the binary-input additive white Gaussian noise (BI-AWGN) channel. We first develop tight approximations to the tail probability of length-n cumulative information density. Building on the result of Yavas *et al.* [YKE21b], for a fixed information density threshold γ, we formulate an integer program of minimizing the upper bound on average blocklength over all decoding times $n_1, n_2, \ldots, n_m$ subject to average error probability, minimum gap and integer constraints. Finally, minimization of locally minimum upper bounds over information density threshold γ yields the globally minimum upper bound, and this method is called the *two-step minimization.* For the integer program, we present a greedy algorithm that yields possibly suboptimal integer decoding times. By allowing positive real-valued decoding times, we develop the *gap-constrained SDO procedure* that captures the minimum gap constraint for the relaxed program. In [PPV11], Polyanskiy *et al.* demonstrated that the rate $\frac{C}{1-\epsilon}$ is achievable by allowing the VLSF code to stop at zero with a small probability. In this chapter, we identify the error regime where Polyanskiy's scheme of stopping at zero does not improve the achievability bound. In this error regime, the two-step minimization with the gap-constrained SDO procedure shows that a finite m suffices to attain Polyanskiy's bound for VLSF codes with $m = \infty$.

This chapter is organized as follows. Section 4.2 introduces the notation, the BI-AWGN channel model, and the VLSF code with m decoding times. Section 4.3 develops tight approximations to the tail probability of length-n cumulative information density. Section 4.4 introduces the integer program, the two-step minimization, and a greedy algorithm, develops the gap-constrained SDO procedure for the relaxed program, identifies the error regime where stopping at zero does not help, and shows numerical comparisons. Section 4.5 concludes the chapter.

4.2 Preliminaries

4.2.1 Notation

For $k \in \mathbb{Z}_+$, $[k] \triangleq \{1, 2, \ldots, k\}$. We use x_i^j to denote a sequence $(x_i, x_{i+1}, \ldots, x_j)$, $1 \leq i \leq j$. When the context is clear, x_1^n is abbreviated to x^n. All logarithms are taken to the base 2. We use $\phi(x), \Phi(x), Q(x)$ to respectively denote the probability density function (PDF), cumulative distribution function (CDF), and the tail probability of a standard normal $\mathcal{N}(0, 1)$.

4.2.2 Channel Model and VLSF Codes with m Decoding Times

Let X^n be a sequence of independent and identically distributed (i.i.d.) random variables, with each X_i uniformly distributed over $\{-\sqrt{P}, \sqrt{P}\}$, where $\sqrt{P}$ denotes the amplitude of binary-phase shift keying (BPSK). The output Y^n of a memoryless, point-to-point BI-AWGN channel in response to X^n is given by

$$Y^n = X^n + Z^n, \tag{4.1}$$

where $Z_1, Z_2, \ldots, Z_n$ are i.i.d. standard normal random variables. The SNR of the BI-AWGN channel is given by P.

For a BI-AWGN channel with a uniformly distributed input symbol, the information density $\imath(x; y) \triangleq \log \frac{P(y|x)}{P(y)}$ is given by

$$\imath(x; y) = 1 - \log\big(1 + \exp\big(- 2xy\big)\big) . \tag{4.2}$$

Furthermore, the channel capacity $C = \mathbb{E}[\imath(X; Y)]$ and dispersion $V = \text{var}(\imath(X; Y))$. Since the inputs are i.i.d. and the channel is memoryless, the cumulative information density for x^n and y^n is given by

$$\imath(x^n; y^n) \triangleq \log \frac{P(y^n|x^n)}{P(y^n)} = \sum_{i=1}^{n} \imath(x_i; y_i). \tag{4.3}$$

Next, we follow [YKE21b] in describing a VLSF code with m decoding times for the BI-AWGN channel. Due to BPSK, we omit the power constraint from the definition.

An (l, n_1^m, M, ϵ) VLSF code, where l is a positive real, n_1^m and M are non-negative integers satisfying $n_1 < n_2 < \cdots < n_m$, $\epsilon \in (0, 1)$, is defined by

1) A finite alphabet $\mathcal{U}$ and a probability distribution P_U on $\mathcal{U}$ defining the common randomness random variable U that is revealed to both the transmitter and the receiver before the start of transmission.

2) A sequence of encoders $f_n : \mathcal{U} \times [M] \to \mathcal{X}$, $n = 1, 2, \ldots, n_m$, defining channel inputs

$$X_n = f_n(U, W), \tag{4.4}$$

where $W \in [M]$ is the equiprobable message.

3) A non-negative integer-valued random stopping time $\tau \in \{n_1, n_2, \ldots, n_m\}$ of the filtration generated by $\{U, Y^{n_i}\}_{i=1}^m$ that satisfies an average decoding time constraint

$$\mathbb{E}[\tau] \leq l. \tag{4.5}$$

4) m decoding functions $g_{n_i} : \mathcal{U} \times \mathcal{Y}^{n_i} \to [M]$, providing the best estimate of W at time n_i, $i = 1, 2, \ldots, m$. The final decision $\hat{W}$ is computed at time instant τ, i.e., $\hat{W} = g_\tau(U, Y^\tau)$ and must satisfy

$$P_e \triangleq \mathscr{P}[\hat{W} \neq W] \leq \epsilon. \tag{4.6}$$

The rate of a VLSF code is given by $R \triangleq \log M / \mathbb{E}[\tau]$. In the above definition, the cardinality $\mathcal{U}$ specifies the number of deterministic codes under consideration to construct the random code. In [YKE21a, Appendix D], Yavas *et al.* showed that $|\mathcal{U}| \leq 2$ suffices.

4.3 Tight Approximations to $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$

In the analysis of (l, n_1^m, M, ϵ) VLSF codes, a key step is to develop a differentiable function $F_\gamma(n)$ to approximate or to bound the tail probability $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$ with a fixed γ. In [VRD16,WWB17,WVW17,WWB18], $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$ is approximated with a Gaussian tail probability, e.g., $Q\left(\frac{\gamma - nC}{\sqrt{nV}}\right)$ used in [WWB17]. However, we will show that for short blocklength n, the Gaussian model is imprecise and a better approximation is desired.

In probability theory, both the Edgeworth expansion [Edg05, Hal92] and the Petrov expansion [Pet75] have been known as powerful tools to approximate the distribution of the sum of n i.i.d. random variables. In this chapter, we apply these expansions to approximate $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$.

We first introduce the *cumulant* of a random variable, which will play an important role in the above expansions.

Definition 5. *Let $K_W(t) = \ln \mathbb{E}[e^{tW}] = \sum_{j=1}^{\infty} \kappa_j \frac{t^j}{j!}$ denote the cumulant generating function for a random variable W. The jth cumulant of W, κ_j, can be computed using the noncentral moments $\mathbb{E}[W^l]$, $1 \leq l \leq j$ [BM98, Eq. (32)],*

$$\kappa_j = j! \sum_{\{k_l\}} (-1)^{r-1}(r-1)! \prod_{l=1}^{j} \frac{1}{k_l!} \left(\frac{\mathbb{E}[W^l]}{l!}\right)^{k_l}, \tag{4.7}$$

where in (4.7), the set $\{k_l\}$ consists of all non-negative solutions to $\sum_{l=1}^{j} l k_l = j$, $r \triangleq \sum_{l=1}^{j} k_l$.

Theorem 16 (Edgeworth Expansion, Equation (2.18), [Hal92]). *Let $W_1, W_2, \ldots, W_n$ be a sequence of i.i.d. random variables with zero mean and a finite variance σ^2. Define $G_n(x) \triangleq \mathscr{P}[\sum_{i=1}^{n} W_i \leq x\sigma\sqrt{n}]$. Let $\chi_W(t) \triangleq \mathbb{E}[e^{\mathrm{i}tW}]$ be the characteristic function of W and let $\{\kappa_i\}_{i=1}^{\infty}$ be the cumulants of W. If $\mathbb{E}[|W|^{s+2}] < \infty$ for some $s \in \mathcal{Z}_+$ and* $\limsup_{|t|\to\infty} |\chi_W(t)| < 1$ *(known as Cramér's condition), then,*

$$G_n(x) = \Phi(x) + \phi(x) \sum_{j=1}^{s} n^{-\frac{j}{2}} p_j(x) + o(n^{-s/2}), \tag{4.8}$$

where by letting $\bar{\kappa}_m = \sigma^{-m}\kappa_m$ *be the mth cumulant for* W/σ,

$$p_j(x) = -\sum_{\{k_m\}} He_{j+2r-1}(x) \prod_{m=1}^{j} \frac{1}{k_m!} \left(\frac{\bar{\kappa}_{m+2}}{(m+2)!} \right)^{k_m}, \tag{4.9}$$

$$He_j(x) = j! \sum_{k=0}^{\lfloor j/2 \rfloor} \frac{(-1)^k x^{j-2k}}{k!(j-2k)!2^k}, \tag{4.10}$$

where the set $\{k_l\}$ *and* r *in* (4.9) *are defined analogously as in* (4.7)*.* $He_j(x)$ *is known as the degree-j Hermite polynomial. In [BM98], the authors presented the formula* (4.10) *and provided an efficient algorithm to compute the set* $\{k_m\}$ *in* (4.9)*.*

As an application of Theorem 16, let $W = 1 - \log\left(1 + e^{-2P - 2Z\sqrt{P}}\right) - C$, where $Z \sim \mathcal{N}(0,1)$. Clearly, W has a proper density function and $\mathbb{E}[|W|^{s+2}] < \infty$ holds for each $s \in \mathbb{Z}_+$. Hence, for moderate and large values of n, the differentiable function $F_\gamma(n)$ we use to approximate $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$ is given by the order-s Edgeworth expansion, i.e.,

$$F_\gamma(n) = Q\left(\frac{\gamma - nC}{\sqrt{nV}}\right) - \phi\left(\frac{\gamma - nC}{\sqrt{nV}}\right) \sum_{j=1}^{s} n^{-\frac{j}{2}} p_j\left(\frac{\gamma - nC}{\sqrt{nV}}\right). \tag{4.11}$$

By (4.9), we see that an order-s Edgeworth expansion utilizes order 3 to $s+2$ cumulants. In this chapter, we define the order of an expansion as the highest order of cumulants minus two.

A caveat of using the order-s Edgeworth expansion is that for small values of n, the order-s Edgeworth expansion oscillates around 0 due to truncation of an infinite series, making it no longer a suitable approximation function to the tail probability. To remedy the situation, we resort to the Petrov expansion [Pet75] for small n.

Theorem 17 (Theorem 1, [Pet75])**.** *Let* $W_1, W_2, \ldots, W_n$ *be a sequence of i.i.d. random variables with zero mean and a finite variance* σ^2*. Define* $G_n(x) \triangleq \mathscr{P}\left[\sum_{i=1}^{n} W_i \leq x\sigma\sqrt{n}\right]$*. If* $x \geq 0$*,* $x = o(\sqrt{n})$*, and the moment generating function* $\mathbb{E}[e^{tW}] < \infty$ *for* $|t| < H$ *for some* $H > 0$*, then*

$$G_n(x) = 1 - Q(x)\exp\left\{\frac{x^3}{\sqrt{n}}\Lambda\left(\frac{x}{\sqrt{n}}\right)\right\}\left[1 + O\left(\frac{x+1}{\sqrt{n}}\right)\right], \tag{4.12}$$

$$G_n(-x) = Q(x) \exp\left\{ \frac{-x^3}{\sqrt{n}} \Lambda\left(\frac{-x}{\sqrt{n}}\right) \right\} \left[1 + O\left(\frac{x+1}{\sqrt{n}} \right) \right], \tag{4.13}$$

where $\Lambda(t) = \sum_{k=0}^{\infty} a_k t^k$ is called the Cramér series[1]*. In [Pet75], Petrov provided the order-*3 *Cramér series $\Lambda^{[3]}(t)$,*

$$\Lambda^{[3]}(t) = \frac{\kappa_3}{6\kappa_2^{3/2}} + \frac{\kappa_4\kappa_2 - 3\kappa_3^2}{24\kappa_2^3} t + \frac{\kappa_5\kappa_2^2 - 10\kappa_4\kappa_3\kappa_2 + 15\kappa_3^3}{120\kappa_2^{9/2}} t^2. \tag{4.14}$$

where $\{\kappa_i\}_{i=1}^{\infty}$ denote the cumulants for W.

For small n satisfying $n < \gamma/C$, the function $F_\gamma(n)$ we use to approximate $\mathscr{P}[\iota(X^n; Y^n) \geq \gamma]$ is given by the order-3 Petrov expansion, where the order of 3 is determined by κ_5 in (4.14),

$$F_\gamma(n) = Q\left(\frac{\gamma - nC}{\sqrt{nV}} \right) \exp\left\{ \frac{(\gamma - nC)^3}{n^2 V^{3/2}} \Lambda^{[3]}\left(\frac{\gamma - nC}{n\sqrt{V}} \right) \right\}. \tag{4.15}$$

Note that at the 0th order, both expansions reduce to $\Phi(x)$.

We remark that both finite-order Edgeworth and Petrov expansions are approximations that are obtained by truncating an infinite series. Edgeworth expansion assumes a constant target probability compared to n, whereas Petrov expansion assumes that the target probability decays sub-exponentially to 0 as $n \to \infty$, defining a moderate deviation sequence in n. Therefore, the former performs better in the large n regime, while the latter performs better in small n regime.

In our implementation, we found that the order-5 Edgeworth expansion meets our desired approximation accuracy at large n. The switch from the order-5 Edgeworth expansion to the order-3 Petrov expansion occurs at the largest value for which two expansions are equal with a common value less than 1/2. Fig. 4.1 shows the comparison of different approximation models for $\mathscr{P}[\iota(X^n; Y^n) \geq \gamma]$ with $\gamma = 13.62$ for BI-AWGN channel at 0.2 dB. The Gaussian model in Fig. 4.1 is given by $Q\left(\frac{\gamma - nC}{\sqrt{nV}}\right)$. Fig. 4.1 shows that the Gaussian model fails to capture the true tail probability at small n. The order-5 Edgeworth expansion oscillates

[1] *Details on Cramér series can be found in the proof of [Pet75, Sec. VIII, Theorem 2].*

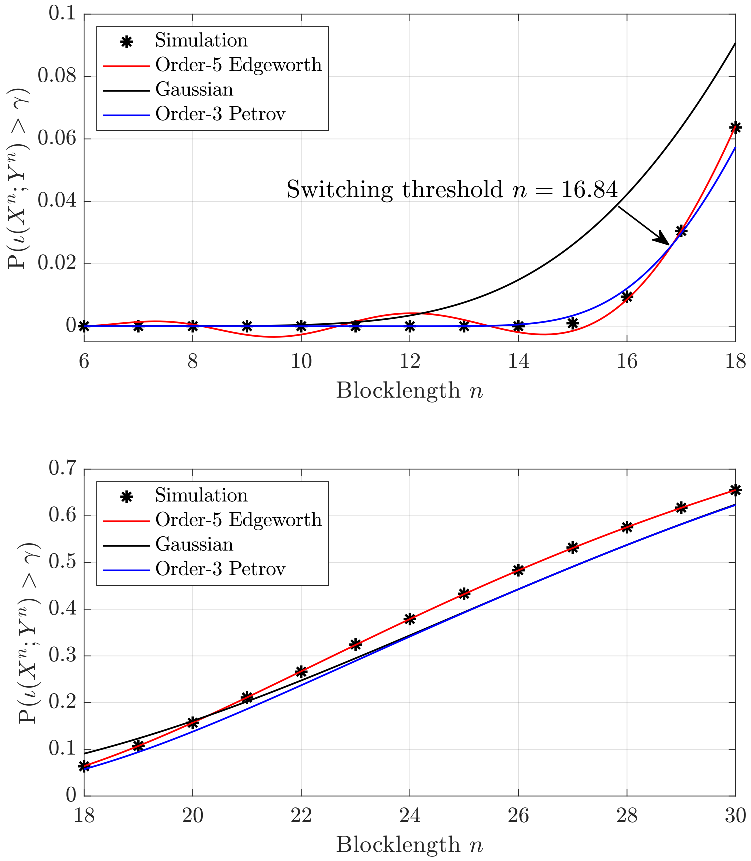

Figure 4.1: Comparison of various approximation models for $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$ with a fixed $\gamma > 0$. In this example, $k = 6$, $\epsilon = 10^{-2}$, $\gamma = \log \frac{2^k-1}{\epsilon/2} = 13.62$ for BI-AWGN channel at 0.2 dB.

around 0 when $n < 16$ and is extremely accurate when $n \geq 16$. In contrast, the order-3 Petrov expansion is loose yet close to the Gaussian model when $n \geq 24$ and becomes tight when $n \leq 14$. Therefore, the combination of the order-2 Petrov expansion and the order-5 Edgeworth expansion at switching threshold $n = 16.84$ provides a remarkably precise estimate of the tail probability $\mathscr{P}[\iota(X^n;Y^n) \geq \gamma]$.

4.4 VLSF Codes With m Optimal Decoding Times

In this section, we develop numerical tools to evaluate the achievable rate of a VLSF code with m optimal decoding times. We mainly consider the error regime where Polyanskiy's scheme of stopping at zero does not improve the achievability bound [PPV11].

4.4.1 An Integer Program and a Greedy Algorithm

In [YKE21b], Yavas *et al.* proved an achievability bound for an (l, n_1^m, M, ϵ) VLSF code for the AWGN channel. With a slight modification, this result is directly applicable to the BI-AWGN channel.

Theorem 18 (Theorem 3, [YKE21b]). *Fix a constant* $\gamma > 0$ *and decoding times* $n_1 < \cdots < n_m$. *For any positive numbers* l *and* $\epsilon \in (0,1)$, *there exists an* (l, n_1^m, M, ϵ') *VLSF code for the BI-AWGN channel* (4.1) *with*

$$l \leq n_m + \sum_{i=1}^{m-1} (n_i - n_{i+1}) \mathscr{P}\left[\bigcup_{j=1}^{i} \{\iota(X^{n_j}; Y^{n_j}) \geq \gamma\} \right], \tag{4.16}$$

$$\epsilon' \leq 1 - \mathscr{P}[\iota(X^{n_m}; Y^{n_m}) \geq \gamma] + (M-1)2^{-\gamma}, \tag{4.17}$$

where $P_{X^{n_m}}$ *is the product of distribution of* m *subvectors of length* $n_j - n_{j-1}$, $j \in [m]$, *with the convention* $n_0 = 0$. *Namely,*

$$P_{X^{n_m}}(x^{n_m}) = \prod_{j=1}^{m} P_{X_{n_{j-1}+1}^{n_j}}(x_{n_{j-1}+1}^{n_j}). \tag{4.18}$$

Remark 5. *In [YKE21b] (and its full version [YKE21c]), Yavas et al. obtained Theorem 18 by constructing a random VLSF code according to distribution* (4.18) *and applying an information density threshold decoder that favors the largest message index whose cumulative information density exceeds* γ *for the first time among any other message indices at decoding times* $\{n_1, n_2, \ldots, n_m\}$.

In (4.17)*, the first term upper bounds the probability that the true message never crosses* γ *and the second term upper bounds the probability that any other message crosses* γ *sooner than the true message.*

Interested readers can refer to the full version [YKE21c] of [YKE21b] for the proof of Theorem 18. By relaxing the probability of union events $\mathscr{P}\left[\bigcup_{j=1}^{i}\{\iota(X^{n_j};Y^{n_j}) \geq \gamma\}\right]$ in (4.16) to the marginal probability $\mathscr{P}[\iota(X^{n_i};Y^{n_i}) \geq \gamma]$, Theorem 18 motivates the following integer program. Define

$$N(\gamma, n_1^m) \triangleq n_m + \sum_{i=1}^{m-1}(n_i - n_{i+1})\mathscr{P}[\iota(X^{n_i};Y^{n_i}) \geq \gamma], \tag{4.19}$$

$$\begin{aligned}\mathcal{F}_m(\gamma, M, \epsilon) \triangleq \{&n_1^m : n_{i+1} - n_i \geq 1, i \in [m-1] \\ &\text{and } \mathscr{P}[\iota(X^{n_m};Y^{n_m}) \geq \gamma] \geq 1 - \epsilon + (M-1)2^{-\gamma}\}.\end{aligned} \tag{4.20}$$

For a given $m \in \mathbb{Z}_+, M \in \mathbb{Z}_+$, $\epsilon \in (0,1)$, and $\gamma \geq \log\frac{M-1}{\epsilon}$,

$$\begin{aligned}\min \quad & N(\gamma, n_1^m) \\ \text{s.t.} \quad & n_1^m \in \mathcal{F}_m(\gamma, M, \epsilon) \\ & n_1^m \in \mathbb{Z}_+^m.\end{aligned} \tag{4.21}$$

In the integer program (4.21), we consider the minimum gap and average error probability constraints as in (4.20), and the constraint that all decoding times must be integers.

Let $\tilde{N}(\gamma)$ denote the locally minimum upper bound $N(\gamma, n_1^m)$ on $\mathbb{E}[\tau]$ for a given γ in program (4.21). Then, $\min_\gamma \tilde{N}(\gamma)$ yields the globally minimum upper bound $N(\gamma, n_1^m)$. In this chapter, we solve the globally minimum upper bound $N(\gamma, n_1^m)$ using this two-step minimization approach.

In general, an integer program is NP-complete. For the specific integer program (4.21), additional challenge is caused by that there is no closed-form expression for $\mathscr{P}[\iota(X^{n_k};Y^{n_k}) \geq \gamma]$ and $n_1, n_2, \ldots, n_m$ are required to be monotonically increasing integers.

While a complete solution to the integer program (4.21) remains open, we propose a greedy algorithm for a fixed γ: Start from $m = n^*$ where $n^* \triangleq \min\{n \in \mathbb{Z}_+ : \mathscr{P}[\imath(X^n; Y^n) \geq \gamma] \geq 1-\epsilon+(M-1)2^{-\gamma}\}$. Suppose that n_1^m is the solution for m. Then, the solution $\tilde{n}_1^{m-1}$ for $m-1$ is identified by removing the decoding time n_i in n_1^{m-1} that minimizes $N_\gamma(n_1^{i-1}, n_{i+1}^m)$. Note that the decoding time n_m is always retained to ensure that the target error probability is met via (4.17).

4.4.2 The Relaxed Program and the Gap-Constrained SDO

To facilitate a program that is computationally tractable, we consider the relaxed program that allows $n_1^m \in \mathbb{R}_+^m$: For a given $m \in \mathbb{Z}_+, M \in \mathbb{Z}_+$, $\epsilon \in (0,1)$ and $\gamma \geq \log \frac{M-1}{\epsilon}$,

$$\begin{aligned} \min \quad & N(\gamma, n_1^m) \\ \text{s.t.} \quad & n_1^m \in \mathcal{F}_m(\gamma, M, \epsilon), \end{aligned} \tag{4.22}$$

where the tail probability $\mathscr{P}[\imath(X^n; Y^n) \geq \gamma]$ is approximated by a monotonically increasing and differentiable function $F_\gamma(n)$ with $F_\gamma(0) = 0$ and $F_\gamma(\infty) = 1$, for instance, the piecewise function[2] introduced in Section 4.3. Let

$$f_\gamma(n) \triangleq \frac{dF_\gamma(n)}{dn}. \tag{4.23}$$

For the relaxed program (4.22) with a fixed γ, the optimal, real-valued decoding times $n_1^*, n_2^*, \ldots, n_m^*$ are given by the following theorem.

Theorem 19. *Assume $\imath(X;Y)$ is continuous and $F_\gamma(n)$ is an increasing, differentiable function. For a given $m \in \mathbb{Z}_+, M \in \mathbb{Z}_+$, $\epsilon \in (0,1)$ and $\gamma \geq \log \frac{M-1}{\epsilon}$, the optimal real-valued decoding times $n_1^*, n_2^*, \ldots, n_m^*$ in program (4.22) satisfy*

$$n_m^* = F_\gamma^{-1}\left(1 - \epsilon + (M-1)2^{-\gamma}\right), \tag{4.24}$$

[2] The first derivative of $F_\gamma(n)$ at the switching threshold does not exist. Nonetheless, one can assign the right (or left) derivative as the derivative for the switching threshold so that the solution is not affected significantly.

$$n^*_{k+1} = n^*_k + \max\left\{1, \frac{F_\gamma(n^*_k) - F_\gamma(n^*_{k-1}) - \lambda_{k-1}}{f_\gamma(n^*_k)}\right\}, \tag{4.25}$$

$$\lambda_k = \max\{\lambda_{k-1} + f_\gamma(n^*_k) - F_\gamma(n^*_k) + F_\gamma(n^*_{k-1}), 0\}, \tag{4.26}$$

where $k \in [m-1]$, $\lambda_0 \triangleq 0$ *and* $n^*_0 \triangleq 0$.

Proof. For brevity, define $\boldsymbol{n} \triangleq (n_1, n_2, \ldots, n_m)$. By introducing the Lagrangian multipliers ν, λ_1^{m-1}, the Lagrangian of program (4.22) is given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{n}, \nu, \lambda_1^{m-1}) = n_1 &+ \nu(1 - F_\gamma(n_m) + (M-1)2^{-\gamma} - \epsilon) \\ &+ \sum_{i=1}^{m-1}(n_{i+1} - n_i)(1 - F_\gamma(n_i)) + \sum_{i=1}^{m-1}\lambda_i(n_i - n_{i+1} + 1).\end{aligned}$$

By the Karush-Kuhn-Tucker (KKT) conditions, the optimal decoding times $\boldsymbol{n}^* = (n^*_1, n^*_2, \ldots, n^*_m)$ must satisfy

$$\frac{\partial \mathcal{L}}{\partial n_k}\Big|_{\boldsymbol{n}=\boldsymbol{n}^*} = F_\gamma(n^*_k) - F_\gamma(n^*_{k-1}) - (n^*_{k+1} - n^*_k)f_\gamma(n^*_k) + \lambda_k - \lambda_{k-1} = 0,\ k \in [m-1], \tag{4.27}$$

$$\frac{\partial \mathcal{L}}{\partial n_m}\Big|_{\boldsymbol{n}=\boldsymbol{n}^*} = 1 - F_\gamma(n^*_{m-1}) - \nu f_\gamma(n^*_m) = 0, \tag{4.28}$$

$$\nu(1 - F_\gamma(n^*_m) + (M-1)2^{-\gamma} - \epsilon) = 0, \tag{4.29}$$

$$\lambda_k(n^*_k - n^*_{k+1} + 1) = 0,\ k \in [m-1]. \tag{4.30}$$

Since $F_\gamma(n) \in (0,1)$ and $f_\gamma(n) > 0$ for $n > 0$, (4.28) indicates that $\nu > 0$. Hence, we obtain $n^*_m = F_\gamma^{-1}(1 - \epsilon + (M-1)2^{-\gamma})$ from (4.29).

Next, we analyze (4.30). There are two cases. If $\lambda_k > 0$, then $n^*_{k+1} = n^*_k + 1$. By (4.27), we obtain

$$\lambda_k = \lambda_{k-1} + f_\gamma(n^*_k) - F_\gamma(n^*_k) + F_\gamma(n^*_{k-1}). \tag{4.31}$$

If $n^*_{k+1} > n^*_k + 1$, then $\lambda_k = 0$. By (4.27), we obtain

$$n^*_{k+1} = n^*_k + \frac{F_\gamma(n^*_k) - F_\gamma(n^*_{k-1}) - \lambda_{k-1}}{f_\gamma(n^*_k)}. \tag{4.32}$$

Rewriting the above two cases in a compact form yields (4.25) and (4.26). □

The procedures (4.25) and (4.26) are called the *gap-constrained SDO procedure* for the relaxed program (4.22). In contrast, the *unconstrained SDO procedure* studied in [VRD16, WWB17, WWB18, WVW17, HCP18, HCW19] derived from the relaxed program (4.22) does not consider the gap constraint[3] and admits a simple recursion

$$n^*_{k+1} = n^*_k + \frac{F_\gamma(n^*_k) - F_\gamma(n^*_{k-1})}{f_\gamma(n^*_k)}, \quad k \in [m-1], \tag{4.33}$$

where $n^*_0 \triangleq 0$. We will show that for small values of m, the gap-constrained SDO procedure behaves indistinguishably as the unconstrained SDO procedure in (4.33). However, as m becomes large, the decoding times provided by these two algorithms differ noticeably.

In practice, after solving n^*_m via (4.24), one would apply a bisection search between 0.5 and $\lceil n^*_m \rceil - m + 0.5$ for n_1 and the SDO to identify n^*_1. This guarantees that the nearest integer to n^*_1 is at least 1.

When evaluating at small n, both $F_\gamma(n)$ and $f_\gamma(n)$ will become infinitesimally small. In this case, a direct numerical computation using (4.25) and (4.26) may cause the precision issue. Fortunately, the SDO described by (4.25) and (4.26) also admits a ratio form. Define $\lambda^{(r)}_k \triangleq \lambda_k / f_\gamma(n^*_k)$. Thus, (4.25) and (4.26) are equivalent to

$$n^*_{k+1} = n^*_k + \max\left\{1, \frac{F_\gamma(n^*_k)}{f_\gamma(n^*_k)} - \frac{F_\gamma(n^*_{k-1})}{f_\gamma(n^*_k)} - \lambda^{(r)}_{k-1}\frac{f_\gamma(n^*_{k-1})}{f_\gamma(n^*_k)}\right\},$$
$$\lambda^{(r)}_k = \max\left\{\lambda^{(r)}_{k-1}\frac{f_\gamma(n^*_{k-1})}{f_\gamma(n^*_k)} + 1 - \frac{F_\gamma(n^*_k)}{f_\gamma(n^*_k)} + \frac{F_\gamma(n^*_{k-1})}{f_\gamma(n^*_k)}, 0\right\}.$$

The purpose of using $F_\gamma(\tilde{n})/f_\gamma(n)$, $f_\gamma(\tilde{n})/f_\gamma(n)$, and $\lambda^{(r)}_k$ is that they have a closed-form expression that cancels out the common infinitesimal factor in both the numerator and denominator. In our implementation, we applied the ratio form of the gap-constrained SDO procedure.

[3] The error probability constraint is also different, yet it does not affect the SDO procedure.

4.4.3 Error Regime Where Stopping at Zero Does Not Help

In [PPV11], Polyanskiy *et al.* demonstrated that the VLSF code with infinitely many stopping times can achieve $\frac{C}{1-\epsilon}$. This is accomplished by the following scheme: With probability $p = \frac{\epsilon-\epsilon'}{1-\epsilon'}$, the code immediately stops at $\tau = 0$ without any channel use, and with probability $1-p$, employs an (l', M, ϵ') VLSF code satisfying $\log M = Cl' + \log \epsilon' - a_0$, where $a_0 \triangleq \sup_{x,y} \iota(x;y)$. The overall code has an error probability

$$1 \cdot p + \epsilon'(1-p) = \epsilon, \tag{4.34}$$

and average blocklength

$$0 \cdot p + l'(1-p) = l'(1-p). \tag{4.35}$$

In this section, we identify the error regime where Polyanskiy's scheme of stopping at $\tau = 0$ does not improve the achievability bound.

Theorem 20. *For a given $a_0 \in \mathbb{R}_+$, $M \in \mathbb{Z}_+$, define*

$$\epsilon^* \triangleq \operatorname*{arg\,min}_{x\in(0,1)} \frac{\log M + a_0 - \log x}{1-x}. \tag{4.36}$$

If $\epsilon \in (0, \epsilon^*]$, stopping at $\tau = 0$ does not improve the achievability bound for VLSF codes.

Proof. By Polyanskiy's scheme, solving the error regime where stopping at zero does not improve the achievability bound is equivalent to identifying the error regime in which $\epsilon' = \epsilon$ is the minimizer to the following program: For a given $C, a_0 \in \mathbb{R}_+$, $M \in \mathbb{Z}_+$, and $\epsilon \in (0,1)$,

$$\begin{aligned} \min_{\epsilon'} \quad & l'(1-p) \\ \text{s.t.} \quad & \log M = Cl' + \log \epsilon' - a_0 \\ & p = \frac{\epsilon-\epsilon'}{1-\epsilon'} \\ & \epsilon' \in (0, \epsilon]. \end{aligned} \tag{4.37}$$

The program (4.37) is equivalent to the following program

$$\begin{aligned} \min_{\epsilon'} \quad & \left(\frac{1-\epsilon}{C}\right) f(\epsilon') \\ \text{s.t.} \quad & \epsilon' \in (0, \epsilon], \end{aligned} \tag{4.38}$$

where

$$f(x) \triangleq \frac{\log M + a_0 - \log x}{1-x}. \tag{4.39}$$

Since $f(x)$ is convex in $(0,1)$, there exists a unique minimizer $\epsilon^* \in (0,1)$. Therefore, if $\epsilon \leq \epsilon^*$, then $\epsilon' = \epsilon$ minimizes the objective function in (4.38), giving $p = 0$. Namely, stopping at zero does not improve the achievability bound. □

We remark that there is no closed-form solution to ϵ^* in Theorem 20. Nonetheless, one can numerically solve ϵ^* for a given M and a_0.

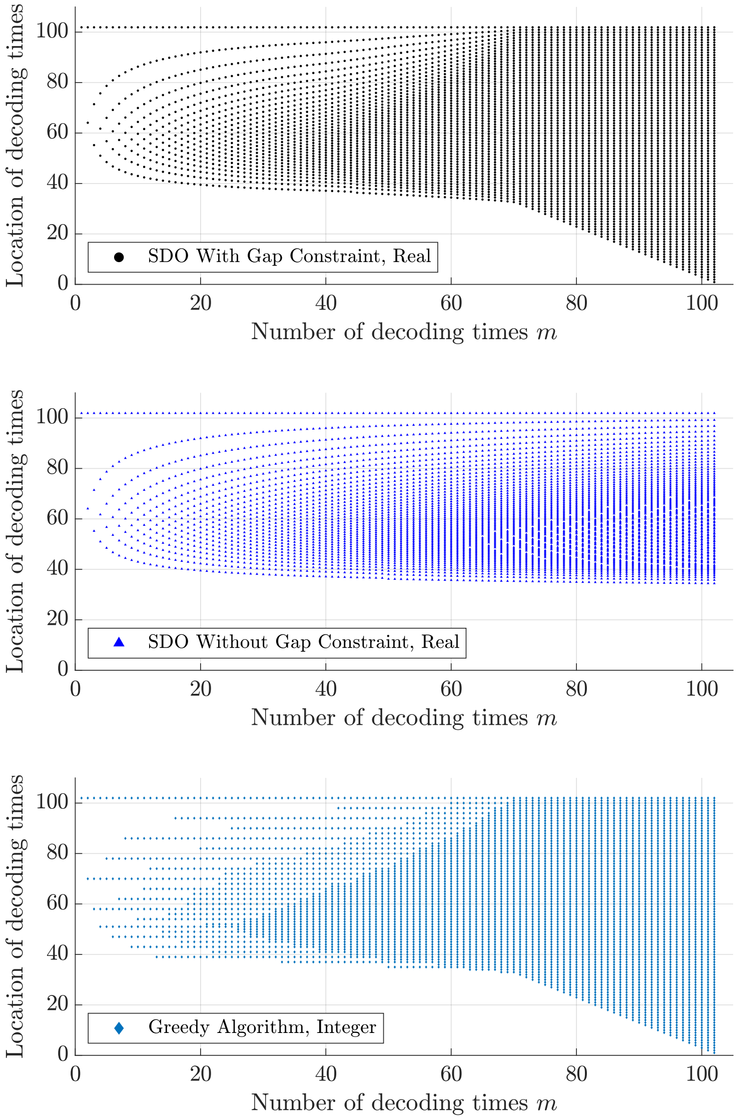

Figure 4.2: Comparison of the real-valued decoding times by the gap-constrained SDO procedure, the real-valued decoding times by the unconstrained SDO procedure, and the integer–valued decoding times by the greedy algorithm for $k = 20$, $\epsilon = 10^{-2}$, $\delta = 1/2$, $\gamma = \log \frac{2^k - 1}{\delta\epsilon}$ and BI-AWGN channel at 0.2 dB. m ranges from 1 to $\lceil n_m^* \rceil = 102$, where $n_m^* = 101.91$ is given by (4.24).

4.4.4 Numerical Evaluation

Let $M = 2^k$, $k \in \mathbb{Z}_+$. We consider the BI-AWGN channel at 0.2 dB with a capacity of 0.5 and the error regime in which stopping at zero does not improve the achievability bound. By Theorem 20, if $k \leq 100$, $\epsilon \leq 1.33 \cdot 10^{-2}$ is the error regime where stopping at zero does not help. In the following example, we consider $\epsilon = 10^{-2}$.

We consider the relaxed program and apply the two-step minimization and the gap-constrained SDO procedure introduced in Section 4.4 to obtain the globally minimum upper bound $N^*(\gamma, n_1^m)$. Thus, $k/N^*(\gamma, n_1^m)$ gives the achievability bound. In [PPV11], Polyanskiy *et al.* showed that the average blocklength $\mathbb{E}[\tau]$ of a VLSF code with $m = \infty$ and no stopping at $\tau = 0$ is upper bounded by

$$\mathbb{E}[\tau] \leq \frac{\log \frac{M-1}{\epsilon} + a_0}{C}, \tag{4.40}$$

where $a_0 \triangleq \sup_{x,y} \iota(x; y)$. This bound yields the *VLSF achievability bound* on rate. For BI-AWGN channel, $a_0 = 1$.

For a fixed γ at $k = 20$ and $\epsilon = 10^{-2}$, Fig. 4.2 shows how the decoding times evolve with m for the three algorithms: the gap-constrained SDO procedure, the unconstrained SDO procedure, and the greedy algorithm. For $m \leq 60$, the gap-constrained SDO procedure behaves indistinguishably as the unconstrained SDO procedure since the SDO solution naturally has gaps larger than one. The greedy algorithm is forced to choose from the remaining decoding times, leading to a possibly suboptimal solution. For large m, the unconstrained SDO procedure avoids early decoding times and instead adds later decoding times so densely that their separation is less than one. In contrast, the gap-constrained SDO procedure is forced to add early decoding times when all existing gaps become one.

The greedy algorithm lacks the optimality guarantee of the gap-constrained SDO procedure and is computationally more intensive. Despite their distinct design perspectives, the greedy algorithm and the gap-constrained SDO procedure arrive at essentially the same

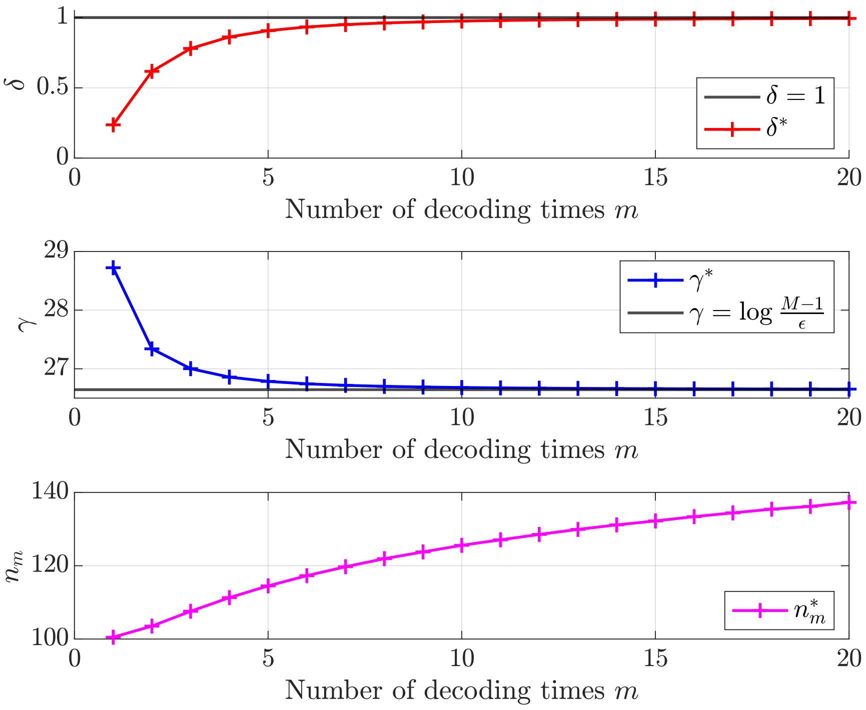

Figure 4.3: Globally optimal δ^*, γ^*, and n_m^* as a function of the number of decoding times m. In the VLSF achievability bound, $\delta = 1$ and $\gamma = \log \frac{M-1}{\epsilon}$. In this example, $\epsilon = 10^{-2}$, $k = 20$ for BI-AWGN channel at 0.2 dB.

solution for large m. For $k = 20$, $\epsilon = 10^{-2}$, and $\gamma = \log \frac{(2^k - 1)}{\epsilon/2}$, Fig. 4.2 shows that n_1 is never less than 37 when $m \leq 32$ and grows as the number of decoding times decreases.

We remark that Fig. 4.2 assumes a constant γ over all number of decoding times. However, in the two-step minimization with the gap-constrained SDO procedure, the globally optimal γ^* is a function of m. Therefore, the globally optimal n_m^* may not stay as constant as shown in Fig. 4.2.

Let $\delta \in [0, 1]$ and assume that the first and second terms in the right-hand side of (4.17) are equal to $(1-\delta)\epsilon$ and $\delta\epsilon$, respectively. Then, both γ^* and n_m^* can be thought as a function of δ^*, i.e.,

$$\gamma^*(\delta^*) = \log \frac{M-1}{\epsilon \delta^*}, \tag{4.41}$$

$$n_m^*(\delta^*) = F_{\gamma^*}^{-1}(1 - \epsilon + \epsilon\delta^*). \tag{4.42}$$

Thus, minimization over γ is equivalent to minimization over δ. For $k = 20$, $\epsilon = 10^{-2}$, and BI-AWGN channel at 0.2 dB, Fig. 4.3 shows how the globally optimal δ^* and the associated globally optimal γ^*, n_m^* vary with m during the two-step minimization. We see that when m is small, δ^* is far from 1, indicating a large value of γ^* and a small value of n_m^*. As m gets large, we observe that δ^* monotonically increases, which, by (4.41) and (4.42), implies that γ^* decreases and n_m^* increases. In particular, as $m \to \infty$, $\delta^* \to 1$, and consequently,

$$\lim_{\delta^* \to 1} \gamma^*(\delta^*) = \log \frac{M-1}{\epsilon}, \tag{4.43}$$

$$\lim_{\delta^* \to 1} n_m^*(\delta^*) = \infty. \tag{4.44}$$

In Polyanskiy's setting [PPV11], the first term in (4.17) is zero since $n_m = \infty$ and the optimal γ can thus be computed as $\gamma = \log \frac{M-1}{\epsilon}$ from (4.17), implying that $\delta = 1$. Fig. 4.3 shows that as m increases, δ^*, γ^*, and n_m^* rapidly approach those in Polyanskiy's setting.

Fig. 4.4 shows the achievable rate of a VLSF code with m optimal decoding times estimated by the two-step minimization with the gap-constrained SDO procedure. We see that a finite m suffices to achieve Polyanskiy's VLSF achievability bound derived from VLSF codes with infinitely many decoding times. For instance, for the BI-AWGN channel at 0.2 dB and $\epsilon = 10^{-2}$, the achievable rate estimated by the SDO for VLSF codes with $k \leq 6$ and $m = 32$ beats the VLSF bound. Additionally, for BI-AWGN channel at 0.2 dB, with 16 decoding times, the achievable rate by SDO is within 0.66% of the VLSF achievability bound for $k \leq 100$. With 32 decoding times, it becomes hard to distinguish the achievable rate by SDO from the VLSF achievability bound for $k \leq 30$.

4.5 Conclusion

This chapter provides a new SDO that includes the gap constraint. Using this improved SDO, the chapter demonstrates that Polyanskiy's VLSF achievability bound with infinitely

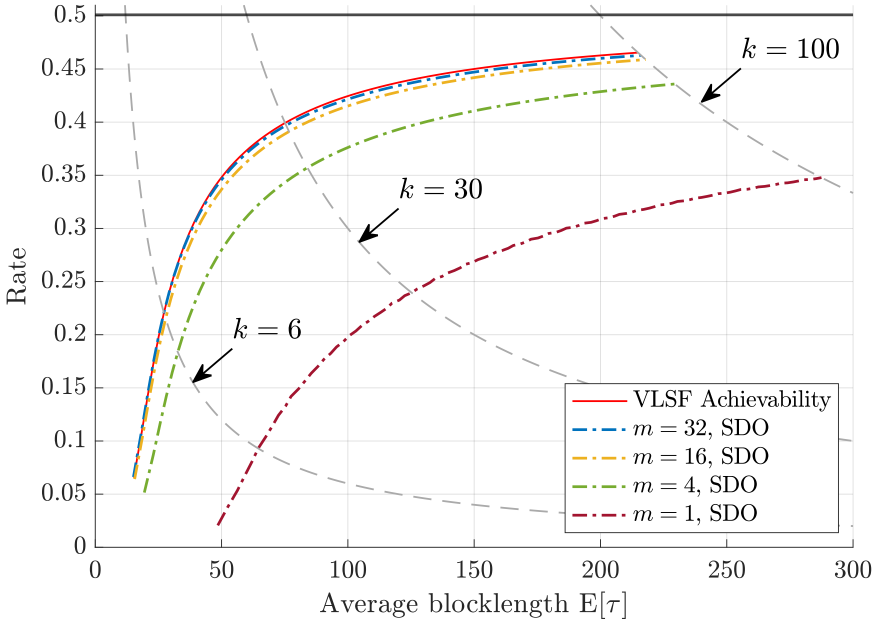

Figure 4.4: Comparison of achievable rate estimation by the gap-constrained SDO procedure and by the greedy algorithm for VLSF codes with m optimal decoding times. The gray dashed line represents the $(\mathbb{E}[\tau], R)$ pairs such that $R\mathbb{E}[\tau] = k$. In this example, $\epsilon = 10^{-2}$ and the BI-AWGN channel is at 0.2 dB.

many decoding times can be closely approached with a finite and relatively small number of decoding times.

Acknowledgment

The material in this chapter is based on the work [YYK22], a short version of which was accepted for presentation at the 2022 IEEE International Symposium on Information Theory.

CHAPTER 5

Conclusion

This dissertation investigated efficient reliable communication for the BI-AWGN channel without feedback, BAC (and BSC) with full noiseless feedback, and the BI-AWGN channel with finite, stop feedback. Three chapters are independent of each other and are interesting in its own right. Below we discuss open problems, possible extensions related to each chapter and their connections that could be further explored.

Chapter 2 investigated the design method, performance, and complexity of the proposed CRC-aided convolutional code under SLVD for the non-feedback BI-AWGN channel. As discussed in the Conclusion of Chapter 2, it would be interesting to study whether a suboptimal convolutional code used with the DSO CRC polynomial can also lead to a good concatenated code. Another interesting direction is to explore the performance of CRC-aided convolutional codes in the moderately short blocklength regime, e.g., $128 \leq k < 1000$. If puncturing is introduced in the code design, the problem of how to jointly design the puncturing pattern and the optimal CRC polynomial for a given convolutional code still remains open. Besides, there are several interesting theoretical open problems regarding SLVD, for instance, how to develop tight bounds on $\mathbb{E}[L|\boldsymbol{X} = \boldsymbol{O}]$ and $P_{e,1}$ using only the weight spectrum. In addition, the behavior of the supremum list rank λ, a quantity that governs the worst SLVD decoding complexity, is also less understood. Furthermore, the distribution of the terminating list rank L as a function of SNR remains unknown and is crucial to analyzing the SLVD decoding complexity. For 5G physical broadcast channel (PBCH), King *et al.* [KKY22] recently showed that the CRC-TBCC outperforms the PBCH polar codes in 5G standard both in terms of the frame error rate and the decoding runtime, suggesting that CRC-TBCCs are good candidates to be considered for 6G. However, the CRC polynomial used with the polar code is poorly designed. How to design optimal CRC generator polynomials for polar codes still remains largely open.

Chapter 3 extended Naghshvar *et al.*'s SED coding scheme for the symmetric binary-input channel with feedback to the BAC with feedback. The theoretical development of our generalized SED coding scheme and the associated non-asymptotic VLF achievability bound utilized the concept of extrinsic probabilities introduced by Naghshvar *et al.* and the fact that $\pi_1^* \leq \pi_0^*$ implies that transmitting symbol 1 attains the maximum relative entropy C_1. However, it remains open whether these observations also hold for a general binary-input channel with feedback. Furthermore, can we develop a similar SED coding scheme for multi-input DMCs, e,g., DMCs with $|\mathcal{X}| > 2$? In [AYW20], Antonini *et al.* considered systematic transmission followed by a type-based, relaxed SED coding scheme for the BSC (the systematic transmission automatically meets the original SED requirement) which significantly reduced the coding complexity. Simulations show that the type-based, relaxed SED coding scheme still achieve a similar performance as the original SED coding scheme. Yet it remains open to prove this phenomenon analytically.

Chapter 4 developed tight approximations for the tail probability of the cumulative information density that underlies the numerical evaluation of the VLSF code with finite decoding times for the BI-AWGN channel with feedback. In view of the inherent inaccuracy of approximations, it would be interesting to develop theoretically tight upper and lower bounds for the distribution of cumulative information density that would enable a more rigorous characterization of the performance of the VLSF code with finite decoding times. Another interesting direction is to extend our technique to more classical channels and to compare with the previously known achievability bounds. On a practical point of view, how to design a structured VLSF code with finite decoding times that outperforms our non-asymptotic VLSF achievability bound still remains open.

Although the three chapters in this dissertation are independent, tight connections among these topics exist and could be explored in future research. For example, the CRC-aided convolutional codes under SLVD can be applied to the variable-length coding with stop feedback framework by treating a NACK as a stop-feedback symbol that asks for retransmission. In this case, it still remains open how to determine the maximum list size Ψ for each decoding time such that the throughput is maximized while maintaining a target error probability. It also remains open how this system will perform by using the CRC-aided convolutional code as a VLSF code for the BI-AWGN channel with stop feedback. On the other hand, the CRC-aided convolutional code can be applied to probabilistic amplitude shaping (PAS) system that employs nonbinary constellations. In [WSA21], Wang *et al.* recently showed that the code generated by PAS and CRC-aided trellis coded modulation outperforms the RCU bound in the short blocklength regime.

The connection between channels with full feedback and channels with finite stop feedback can be seen as follows. Assume that the decoder is only allowed to send full feedback at time instants $n_1, n_2, \ldots, n_m$. At time n_i, the decoder can send a single non-binary symbol represented by $Y_{n_{i-1}+1}^{n_i}$ to the transmitter via the full feedback link. Clearly, this will result in an achievable rate upper bounded by that of the full-feedback system and lower bounded by that of the stop-feedback system. Then, for such a system, what is the maximal achievable rate and which coding scheme achieves the maximal achievable rate?

To summarize, the efficient reliable communication under different types of feedback opened a wide array of new directions that are of both theoretical interest and practical importance.

REFERENCES

[3GP06] "Universal Mobile Telecommunications System (UMTS); Multiplexing and channel coding (FDD); 3GPP TS 25.212 version 7.0.0 Release 7." Technical report, European Telecommunications Standards Institute, 2006.

[3GP18] "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding; 3GPP TS 36.212 version 15.2.1 Release 15." Technical report, European Telecommunications Standards Institute, 2018.

[AH98] J. B. Anderson and S. M. Hladik. "Tailbiting MAP decoders." *IEEE J. Sel. Areas Commun.*, **16**(2):297–302, Feb. 1998.

[Ar09] E. Arıkan. "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels." *IEEE Trans. Inf. Theory*, **55**(7):3051–3073, Jul. 2009.

[Ari72] S. Arimoto. "An algorithm for computing the capacity of arbitrary discrete memoryless channels." *IEEE Trans. Inf. Theory*, **18**(1):14–20, Jan. 1972.

[Ari19] E. Arıkan. "From sequential decoding to channel polarization and back again." *arXiv*, 2019.

[AYW20] A. Antonini, H. Yang, and R. D. Wesel. "Low Complexity Algorithms for Transmission of Short Blocks over the BSC with Full Feedback." In *2020 IEEE Int. Sym. Inf. Theory (ISIT)*, pp. 2173–2178, 2020.

[Ber80] E. R. Berlekamp. "The technology of error-correcting codes." *Proc. IEEE*, **68**(5):564–593, 1980.

[BJK08] I. E. Bocharova, R. Johannesson, B. D. Kudryashov, and M. Loncar. "An Improved Bound on the List Error Probability and List Distance Properties." *IEEE Trans. Inf. Theory*, **54**(1):13–32, Jan. 2008.

[BK19] T. Baicheva and P. Kazakov. "CRC Selection for Decoding of CRC-Polar Concatenated Codes." In *Proc. the 9th Balkan Conf. Inf.*, New York, NY, USA, Sep. 2019. ACM.

[Bla72] R. Blahut. "Computation of channel capacity and rate-distortion functions." *IEEE Trans. Inf. Theory*, **18**(4):460–473, Jul. 1972.

[BM98] S. Blinnikov and R. Moessner. "Expansions for nearly Gaussian distributions." *Astron. Astrophys. Suppl. Ser.*, **130**(1):193–205, 1998.

[BMK04] C. Bai, B. Mielczarek, W. A. Krzymien, and I. J. Fair. “Efficient list decoding for parallel concatenated convolutional codes.” In *2004 IEEE 15th Int. Symp. Personal, Indoor and Mobile Radio Commun.*, volume 4, pp. 2586–2590, Sep. 2004.

[Bur76] M. V. Burnashev. “Data Transmission over a Discrete Channel with Feedback. Random Transmission Time.” *Problemy Peredachi Inf.*, **12**(4):10–30, 1976.

[BZ74] M. V. Burnashev and K. S. Zigangirov. “An interval estimation problem for controlled observations.” *Problemy Peredachi Inf.*, **10**(3):51–61, 1974.

[BZ75] M. V. Burnashev and K. Sh. Zigangirov. “On One Problem of Observation Control.” *Problemy Peredachi Inf.*, **11**(3):44–52, 1975.

[CDJ19] M. C. Coşkun, G. Durisi, T. Jerkovits, G. Liva, W. Ryan, B. Stein, and F. Steiner. “Efficient error-correcting codes in the short blocklength regime.” *Physical Commun.*, **34**:66 – 79, Jun. 2019.

[CS94] R. V. Cox and C. E. W. Sundberg. “An efficient adaptive circular Viterbi algorithm for decoding generalized tailbiting convolutional codes.” *IEEE Trans. Veh. Technol.*, **43**(1):57–68, Feb. 1994.

[CT06a] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley, New Jersey, USA, 2nd edition, 2006.

[CT06b] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* Wiley, New York, NY, USA, 2nd ed edition, 2006.

[DDS14] L. Dolecek, D. Divsalar, Y. Sun, and B. Amiri. “Non-Binary Protograph-Based LDPC Codes: Enumerators, Analysis, and Designs.” *IEEE Trans. Inf. Theory*, **60**(7):3913–3941, Jul. 2014.

[Dea] P. M. Dearing. “Intersections of hyperplanes and conic sections in $\mathbb{R}^n$.”.

[DFK04] S.C. Draper, B.J. Frey, and F.R. Kschischang. “Efficient variable length channel coding for unknown DMCs.” In *2004 IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 379–379, Jun. 2004.

[Edg05] F. Y. Edgeworth. “The Law of Error.” *Cambridge Philos. Trans.*, **20**:36–66 and 113–141., 1905.

[Eli55] P. Elias. “Coding for noisy channels.” *Proc. IRE. Conv. Rec. pt. 4*, **3**:37 – 46, 1955.

[Eli57] P. Elias. “List decoding for noisy channels.” *Proc. IRE WESCON Conf. Rec.*, **2**:94–104, Sep. 1957.

[For73] G. D. Forney. "The Viterbi algorithm." *Proc. IEEE*, **61**(3):268–278, Mar. 1973.

[For74] G. D. Forney. "Convolutional codes II. Maximum-likelihood decoding." *Inf. Contr.*, **25**(3):222–266, Jul. 1974.

[FS95] M. P. C. Fossorier and S. Lin. "Soft-decision decoding of linear block codes based on ordered statistics." *IEEE Trans. Inf. Theory*, **41**(5):1379–1396, Sep. 1995.

[FVM18] J. Font-Segura, G. Vazquez-Vilar, A. Martinez, A. Guillén i Fàbregas, and A. Lancho. "Saddlepoint approximations of lower and upper bounds to the error probability in channel coding." In *2018 52nd Annual Conf. Inf. Sci. Syst. (CISS)*, pp. 1–6, Mar. 2018.

[Gal63] R. G. Gallager. *Low Density Parity Check Codes.* MIT Press, Cambridge, MA, USA, 1963.

[Gal68] R. G. Gallager. *Information Theory and Reliable Communication.* Wiley, New York, NY, USA, 1968.

[Gal13] R. G. Gallager. *Stochastic processes: theory for applications.* Cambridge University Press, Cambridge, United Kingdom, 2013.

[GK21] N. Guo and V. Kostina. "Instantaneous SED coding over a DMC." In *2021 IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 148–153, 2021.

[GNJ17] L. Gaudio, T. Ninacs, T. Jerkovits, and G. Liva. "On the Performance of Short Tail-Biting Convolutional Codes for Ultra-Reliable Communications." In *SCC 2017; 11th Int. ITG Conf. Syst., Commun., and Coding*, pp. 1–6, Feb. 2017.

[Hal92] P. Hall. *The Bootstrap and Edgeworth Expansion.* Springer, New York, NY, USA, 1992.

[HCP18] A. Heidarzadeh, J. F. Chamberland, P. Parag, and R. D. Wesel. "A Systematic Approach to Incremental Redundancy over Erasure Channels." In *2018 IEEE Int. Sym. Inf. Theory (ISIT)*, pp. 1176–1180, Jun. 2018.

[HCW19] A. Heidarzadeh, J. F. Chamberland, R. D. Wesel, and P. Parag. "A Systematic Approach to Incremental Redundancy With Application to Erasure Channels." *IEEE Trans. Commun.*, **67**(4):2620–2631, Apr. 2019.

[HH09] T. Hehn and J. B. Huber. "LDPC Codes and Convolutional Codes with Equal Structural Delay: A Comparison." *IEEE Trans. Commun.*, **57**(6):1683–1692, Jun. 2009.

[Hor63] M. Horstein. "Sequential transmission using noiseless feedback." *IEEE Trans. Inf. Theory*, **9**(3):136–143, July 1963.

[HSS10] E. Hof, I. Sason, and S. Shamai. "Performance Bounds for Erasure, List, and Decision Feedback Schemes With Linear Block Codes." *IEEE Trans. Inf. Theory*, **56**(8):3754–3778, Aug. 2010.

[JM16] T. Jerkovits and B. Matuz. "Turbo code design for short blocks." In *2016 8th Adv. Satellite Multimedia Syst. Conf. and the 14th Signal Processing for Space Commun. Workshop (ASMS/SPSC)*, pp. 1–6, Sep. 2016.

[JPY18] H. Ji, S. Park, J. Yeo, Y. Kim, J. Lee, and B. Shim. "Ultra-Reliable and Low-Latency Communications in 5G Downlink: Physical Layer Aspects." *IEEE Wireless Commun. Mag.*, **25**(3):124–130, Jun. 2018.

[JZ99] R. Johannesson and K. S. Zigangirov. *Fundamentals of Convolutional Coding.* IEEE Press, New Jersey, USA, 1999.

[KKY22] J. King, A. Kwon, H. Yang, W. Ryan, and R. D. Wesel. "CRC-Aided List Decoding of Convolutional and Polar Codes for Short Messages in 5G." In *Proc. IEEE Int. Conf. Commun. (ICC)*, Seoul, South Korea, May 2022.

[KSL15] S. H. Kim, D. K. Sung, and T. Le-Ngoc. "Variable-Length Feedback Codes Under a Strict Delay Constraint." *IEEE Commun. Lett.*, **19**(4):513–516, Apr. 2015.

[KTK18] J. Kim, J. Tak, H. Kwak, and J. No. "A New List Decoding Algorithm for Short-Length TBCCs With CRC." *IEEE Access*, **6**:35105–35111, Jun. 2018.

[KV03] R. Koetter and A. Vardy. "The structure of tail-biting trellises: minimality and basic principles." *IEEE Trans. Inf. Theory*, **49**(9):2081–2105, Sep. 2003.

[LC04] S. Lin and D. J. Costello. *Error Control Coding.* Pearson Education, New Jersey, USA, 2004.

[LCC04] L. Lijofi, D. Cooper, and B. Canpolat. "A reduced complexity list single-wrong-turn (SWT) Viterbi decoding algorithm." In *2004 IEEE 15th Int. Symp. Personal, Indoor and Mobile Radio Commun.*, volume 1, pp. 274–279, Sep. 2004.

[LDW15] C. Lou, B. Daneshrad, and R. D. Wesel. "Convolutional-Code-Specific CRC Code Design." *IEEE Trans. Commun.*, **63**(10):3459–3470, Oct. 2015.

[LPM13] G. Liva, E. Paolini, B. Matuz, S. Scalise, and M. Chiani. "Short Turbo Codes over High Order Fields." *IEEE Trans. Commun.*, **61**(6):2201–2211, Jun. 2013.

[Lub02] M. Luby. "LT codes." In *The 43rd Annual IEEE Symp. Foundations Comp. Sci., 2002. Proc.*, pp. 271–280, Nov. 2002.

[LYD19] E. Liang, H. Yang, D. Divsalar, and R. D. Wesel. "List-Decoded Tail-Biting Convolutional Codes with Distance-Spectrum Optimal CRCs for 5G." In *2019 IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec. 2019.

[MCF12] S. V. Maiya, D. J. Costello, and T. E. Fuja. "Low Latency Coding: Convolutional Codes vs. LDPC Codes." *IEEE Trans. Commun.*, **60**(5):1215–1225, May 2012.

[MCL10] S. Moser, P.-N. Chen, and H.-Y. Lin. "Error probability analysis of binary asymmetric channels." Technical report, National Chiao Tung University, 2010.

[MW86] H. Ma and J. Wolf. "On Tail Biting Convolutional Codes." *IEEE Trans. Commun.*, **34**(2):104–111, Feb. 1986.

[NJW15] M. Naghshvar, T. Javidi, and M. Wigger. "Extrinsic Jensen–Shannon Divergence: Applications to Variable-Length Coding." *IEEE Trans. Inf. Theory*, **61**(4):2148–2164, April 2015.

[NWJ12] M. Naghshvar, M. Wigger, and T. Javidi. "Optimal reliability over a class of binary-input channels with feedback." In *2012 IEEE Inf. Theory Workshop*, pp. 391–395, Sep. 2012.

[PB61] W. W. Peterson and D. T. Brown. "Cyclic Codes for Error Detection." *Proc. IRE*, **49**(1):228–235, Jan. 1961.

[Pet75] V. V. Petrov. *Sums of independent random variables.* Springer, Berlin, Heidelberg, New York, NY, USA, 1975.

[Pol94] G. Poltyrev. "Bounds on the decoding error probability of binary linear codes via their spectra." *IEEE Trans. Inf. Theory*, **40**(4):1284–1292, 1994.

[PPV10] Y. Polyanskiy, H. V. Poor, and S. Verdú. "Channel Coding Rate in the Finite Blocklength Regime." *IEEE Trans. Inf. Theory*, **56**(5):2307–2359, May 2010.

[PPV11] Y. Polyanskiy, H. V. Poor, and S. Verdú. "Feedback in the Non-Asymptotic Regime." *IEEE Trans. Inf. Theory*, **57**(8):4903–4925, Aug 2011.

[RDW19] S. V. S. Ranganathan, D. Divsalar, and R. D. Wesel. "Quasi-Cyclic Protograph-Based Raptor-Like LDPC Codes for Short Block-Lengths." *IEEE Trans. Inf. Theory*, **65**(6):3758–3777, Jun. 2019.

[RH06] M. Roder and R. Hamzaoui. "Fast tree-trellis list Viterbi decoding." *IEEE Trans. Commun.*, **54**(3):453–461, March 2006.

[Ric94] M. Rice. "Comparative analysis of two realizations for hybrid-ARQ error control." In *1994 IEEE Global Commun. Conf. (GLOBECOM)*, pp. 115–119, Nov. 1994.

[Rud76] W. Rudin. *Principles of Mathematical Analysis.* McGraw-Hill, 3rd edition, 1976.

[Sch71] J. Schalkwijk. "A class of simple and optimal strategies for block coding on the binary symmetric channel with noiseless feedback." *IEEE Trans. Inf. Theory*, **17**(3):283–287, May 1971.

[Sch21] R. Schiavone. *"Channel Coding for Massive IoT Satellite Systems."*. Master's thesis, Politechnic University of Turin (Polito), 2021.

[SF11] O. Shayevitz and M. Feder. "Optimal Feedback Communication Via Posterior Matching." *IEEE Trans. Inf. Theory*, **57**(3):1186–1222, March 2011.

[Sha56] C. Shannon. "The zero error capacity of a noisy channel." *IRE Trans. Inf. Theory*, **2**(3):8–19, September 1956.

[Sha59] C. E. Shannon. "Probability of error for optimal codes in a Gaussian channel." *Bell Syst. Tech. J.*, **38**(3):611–656, May 1959.

[SK66] J. Schalkwijk and T. Kailath. "A coding scheme for additive noise channels with feedback–I: No bandwidth constraint." *IEEE Trans. Inf. Theory*, **12**(2):172–182, Apr. 1966.

[SKS] P. Shankar, P. N. A. Kumar, K. Sasidharan, B. S. Rajan, and A. S. Madhu. "Efficient Convergent Maximum Likelihood Decoding on Tail-Biting Trellises.".

[SMA19] M. Shirvanimoghaddam, M. S. Mohammadi, R. Abbas, A. Minja, C. Yue, B. Matuz, G. Han, Z. Lin, W. Liu, Y. Li, S. Johnson, and B. Vucetic. "Short Block-Length Codes for Ultra-Reliable Low Latency Communications." *IEEE Commun. Mag.*, **57**(2):130–137, Feb. 2019.

[SP73] J. Schalkwijk and K. Post. "On the error probability for a class of binary recursive feedback strategies." *IEEE Trans. Inf. Theory*, **19**(4):498–511, July 1973.

[SS94] N. Seshadri and C. W. Sundberg. "List Viterbi decoding algorithms with applications." *IEEE Trans. Commun.*, **42**(234):313–323, Feb. 1994.

[SSF03] R. Y. Shao, Shu Lin, and M. P. C. Fossorier. "Two decoding algorithms for tailbiting codes." *IEEE Trans. Commun.*, **51**(10):1658–1665, Oct. 2003.

[TT02] A. Tchamkerten and E. Telatar. "A feedback strategy for binary symmetric channels." In *Proc. IEEE Int. Symp. Inf. Theory*, pp. 362–362, June 2002.

[TT06] A. Tchamkerten and I. E. Telatar. "Variable length coding over an unknown channel." *IEEE Trans. Inf. Theory*, **52**(5):2126–2145, May 2006.

[TV15] I. Tal and A. Vardy. "List Decoding of Polar Codes." *IEEE Trans. Inf. Theory*, **61**(5):2213–2226, May 2015.

[Vit67] A. Viterbi. "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm." *IEEE Trans. Inf. Theory*, **13**(2):260–269, Apr. 1967.

[VRD16] K. Vakilinia, S. V. S. Ranganathan, D. Divsalar, and R. D. Wesel. "Optimizing Transmission Lengths for Limited Feedback With Nonbinary LDPC Examples." *IEEE Trans. Commun.*, **64**(6):2245–2257, Jun. 2016.

[WB89] Q. Wang and V. K. Bhargava. "An efficient maximum likelihood decoding algorithm for generalized tail biting convolutional codes including quasicyclic codes." *IEEE Trans. Commun.*, **37**(8):875–879, Aug. 1989.

[Wil91] D. Williams. *Probability with Martingales.* Cambridge University Press, Cambridge, United Kingdom, 1991.

[WSA20] A. B. Wagner, N. V. Shende, and Y. Altuğ. "A New Method for Employing Feedback to Improve Coding Performance." *IEEE Trans. Inf. Theory*, **66**(11):6660–6681, Nov. 2020.

[WSA21] L. Wang, D. Song, F. Areces, and R. D. Wesel. "Achieving Short-Blocklength RCU bound via CRC List Decoding of TCM with Probabilistic Shaping." *arXiv*, Nov. 2021.

[WVW17] N. Wong, K. Vakilinia, H. Wang, S. V. S. Ranganathan, and R. D. Wesel. "Sequential differential optimization of incremental redundancy transmission lengths: An example with tail-biting convolutional codes." In *2017 Inf. Theory and App. Workshop (ITA)*, pp. 1–5, Feb. 2017.

[WWB17] H. Wang, N. Wong, A. M. Baldauf, C. K. Bachelor, S. V. S. Ranganathan, D. Divsalar, and R. D. Wesel. "An information density approach to analyzing and optimizing incremental redundancy with feedback." In *2017 IEEE Int. Sym. Inf. Theory (ISIT)*, pp. 261–265, Jun. 2017.

[WWB18] R. D. Wesel, N. Wong, A. Baldauf, A. Belhouchat, A. Heidarzadeh, and J. F. Chamberland. "Transmission Lengths That Maximize Throughput of Variable-Length Coding & ACK/NACK Feedback." In *2018 IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec. 2018.

[Yana] H. Yang. "GitHub repository: CRC Design for TBCCs." Accessed: May 20, 2021.

[Yanb] H. Yang. "GitHub repository: CRC Design for ZTCCs." Accessed: May 20, 2021.

[YFV20] H. Yao, A. Fazeli, and A. Vardy. "List Decoding of Arıkan's PAC Codes." In *2020 IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 443–448, Jun. 2020.

[YK04a] S. Yousefi and A.K. Khandani. "Generalized tangential sphere bound on the ML decoding error probability of linear binary block codes in AWGN interference." *IEEE Trans. Inf. Theory*, **50**(11):2810–2815, Nov. 2004.

[YK04b] S. Yousefi and A.K. Khandani. "A new upper bound on the ML decoding error probability of linear binary block codes in AWGN interference." *IEEE Trans. Inf. Theory*, **50**(12):3026–3036, Dec. 2004.

[YKE21a] R. C. Yavas, V. Kostina, and M. Effros. "Random Access Channel Coding in the Finite Blocklength Regime." *IEEE Trans. Inf. Theory*, **67**(4):2115–2140, Apr. 2021.

[YKE21b] R. C. Yavas, V. Kostina, and M. Effros. "Variable-length Feedback Codes with Several Decoding Times for the Gaussian Channel." In *2021 IEEE Int. Sym. Inf. Theory (ISIT)*, pp. 1883–1888, Jul. 2021.

[YKE21c] R. C. Yavas, V. Kostina, and M. Effros. "Variable-length Feedback Codes with Several Decoding Times for the Gaussian Channel." *arXiv*, Mar. 2021.

[YLP22] H. Yang, E. Liang, M. Pan, and R. D. Wesel. "CRC-Aided List Decoding of Convolutional Codes in the Short Blocklength Regime." *IEEE Trans. Inf. Theory*, **68**(6):3744–3766, 2022.

[YPA22] H. Yang, M. Pan, A. Antonini, and R. D. Wesel. "Sequential Transmission Over Binary Asymmetric Channels With Feedback." *IEEE Trans. Inf. Theory*, May 2022. in press.

[YRW18] H. Yang, S. V. S. Ranganathan, and R. D. Wesel. "Serial List Viterbi Decoding with CRC: Managing Errors, Erasures, and Complexity." In *2018 IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, Dec. 2018.

[YSV21] C. Yue, M. Shirvanimoghaddam, B. Vucetic, and Y. Li. "A Revisit to Ordered Statistics Decoding: Distance Distribution and Decoding Rules." *IEEE Trans. Inf. Theory*, **67**(7):4288–4337, Jul. 2021.

[YW20] H. Yang and R. D. Wesel. "Finite-Blocklength Performance of Sequential Transmission over BSC with Noiseless Feedback." In *2020 IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 2161–2166, 2020.

[YWL20] H. Yang, L. Wang, V. Lau, and R. D. Wesel. "An Efficient Algorithm for Designing Optimal CRCs for Tail-Biting Convolutional Codes." In *2020 IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 292–297, Jun. 2020.

[YYK22] H. Yang, R. C. Yavas, V. Kostina, and R. D. Wesel. "Variable-Length Stop-Feedback Codes With Finite Optimal Decoding Times for BI-AWGN Channels." *2022 IEEE Int. Sym. Inf. Theory (ISIT)*, Jun. 2022. in press.