UNIVERSITY OF CALIFORNIA, SAN DIEGO

Synchronization Error Channels & Windowed Decoding in Theory & Practice

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Electrical Engineering
(Communication Theory & Systems)

by

Aravind Raghava Iyengar

Committee in charge:

Professor Paul H. Siegel, Chair
Professor Patrick J. Fitzsimmons
Professor Young-Han Kim
Professor Alexander Vardy
Professor Ruth J. Williams

2012

The dissertation of Aravind Raghava Iyengar is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____

Chair

University of California, San Diego

2012

DEDICATION

To my parents, sister Anu & wife Ramya.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I am deeply indebted to my advisors, Prof. Paul H. Siegel and Prof. Jack K. Wolf, without whose guidance, encouragement and timely advice, this work would not be possible. Their patience and willingness to explore strange-sounding ideas have been as responsible for the fruition of my doctoral studies as their deep understanding and experience in the topic. The belief they reposed in me was sometimes more than my own, and this was of tremendous help during tough times. They struck a perfect balance between keeping me on track in my research and giving me the freedom to explore problems that interested me. I consider myself extremely lucky to have been able to work with them. Toby Wolf has been very encouraging and I am thankful to her for all her feedback, advice and help.

Thanks go to my mentor during my visit to EPFL, Prof. Rüdiger Urbanke. Rudi has been a great inspiration. His help during and after my visit is deeply appreciated. I would like to extend my gratitude to my committee members Prof. Patrick J. Fitzsimmons, Prof. Young-Han Kim, Prof. Alexander Vardy, and Prof. Ruth J. Williams for their stimulating discussions. Special thanks to Prof. Kim for recommending my fellowship in the first year, and for accepting to be on my committee at short notice.

The staff at the Center for Magnetic Recording Research has been completely responsible for taking care of all official matters, and thereby allowing me to focus on my research. I thank Betty and Iris for taking care of all my needs at work, and answering my endless list of frivolous questions. Jan was very helpful in tracking down old and arcane reference materials, and Ray in handling all the computing and technical issues with ease. I thank them both. I thank the staff of the ECE department for developing a friendly and refreshing atmosphere through their frequent coffee and happy hours. I thank Muriel for taking care of all the logistics during my visits to EPFL.

My friends at UCSD have been responsible for making me feel at home away from home. Hirakendu's help right from the time I secured an admit at UCSD has been invaluable and is very appreciated. Arun, Harish & Balaji, who accompanied me to UCSD from IIT have been a great gang. My roommate Bharath has been a tremendous source of inspiration, and has grown to be a close friend and confidante. I thank him for his company during many sleepless nights of music, discussions, movies, and work.

Vikas, who has been my neighbor since elementary school, has been a great friend. I am thankful for his constant guidance and for all the thought-provoking discussions we have shared. Jayadev was enjoyable company during the classes in the first year. Thanks to Aditya for being a great roommate and friend, and to Sraavan and Saurabh for lightening the mood many a times. Thanks go to my friends from IIT who have gone on similar journeys through grad school—they have been a very enjoyable company.

I am also very grateful to my collaborators, Marco and Pablo, with whom I have thoroughly enjoyed working. Also, I am thankful to them for showing me to balance work and play. Thanks go to my lab neighbor Aman, with whom I have brainstormed often. I am grateful to my seniors in the lab—Seyhan, Eitan, Ido and Brian, as well as alumni—Hossein, Junsheng, Hadi, Joseph and Henry for their guidance and help. The competitive atmosphere in the lab due to them and Minghai, Scott, Xiaojie has also motivated my work, and I thank them all for this.

Last, but not the least, I thank my family for their constant support, enthusiastic encouragement, and unconditional love. This journey would be impossible without the backing of my parents, and sister Anu. My wife Ramya has joined in this bandwagon over the last year and I thank her for her love and support. My brother-in-law, Thejaswi, has been a constant mentor and help, and I am extremely grateful to him.

This research was supported in part by the Center for Magnetic Recording Research, and by the National Science Foundation under grant CCF-0829865.

Chapter 2 contains material from the paper "On the Capacity of Channels with Synchronization Errors", A.R. Iyengar, P. H. Siegel, J. K. Wolf; which is being prepared for submission to the IEEE Transactions on Information Theory. Chapter 3 contains material from "Write Channel Model for Bit-Patterned Media Recording", A.R. Iyengar, P. H. Siegel, J. K. Wolf; published in the IEEE Transactions on Magnetics, vol. 47, no. 1, Jan. 2011. Chapter 4 is a reprint of "Windowed Decoding of Spatially Coupled Codes", A.R. Iyengar, P. H. Siegel, R. L. Urbanke, J. K. Wolf; which has been submitted for publication in the IEEE Transactions on Information Theory. Chapter 5 contains material from "Windowed Decoding of Protograph-based LDPC Convolutional Codes over Erasure Channels", A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, G. E. Corazza; published in the IEEE Transactions on Information Theory, vol.

58, no. 4, Apr. 2012. The dissertation author was the primary investigator and author of all of these papers.

VITA

2007	B. Tech. in Electrical Engineering, Indian Institute of Technology Madras, Chennai
2008-2012	Graduate Student Researcher, University of California, San Diego
2009	M. S. in Electrical Engineering (Communication Theory & Systems), University of California, San Diego
2012	Ph. D. in Electrical Engineering (Communication Theory & Systems), University of California, San Diego

PUBLICATIONS

A. R. Iyengar, P. H. Siegel, J. K. Wolf, "LDPC Codes for the Cascaded BSC-BAWGN Channel", *Proc. 47th Allerton Conference on Communication, Control, and Computing*, 30 Sep.-2 Oct. 2009, Monticello, IL, USA.

M. Papaleo, A. R. Iyengar, P. H. Siegel, J. K. Wolf, G. E. Corazza, "Windowed Erasure Decoding of LDPC Convolutional Codes", *Proc. IEEE Information Theory Workshop*, 6-8 Jan. 2010, Cairo, Egypt.

A. R. Iyengar, M. Papaleo, G. Liva, P. H. Siegel, J. K. Wolf, G. E. Corazza, "Protograph-based LDPC Convolutional Codes for Correlated Erasure Channels", *Proc. IEEE International Conference on Communications*, 23-27 May 2010, Cape Town, South Africa.

A. R. Iyengar, P. H. Siegel, J. K. Wolf, "Data-Dependent Write Channel Model for Magnetic Recording", *Proc. IEEE International Symposium on Information Theory*, 13-18 Jun. 2010, Austin, TX, USA.

G. E. Corazza, A. R. Iyengar, M. Papaleo, P. H. Siegel, A. Vanelli-Coralli, J. K. Wolf, "Latency-Constrained Protograph-based LDPC-CC", *Proc. 6th International Symposium on Turbo Codes and Iterative Information Processing*, 6-10 Sep. 2010, Brest, France.

A. R. Iyengar, P. H. Siegel, J. K. Wolf, "Write Channel Model for Bit-Patterned Media Recording", *IEEE Transactions on Magnetics*, vol. 47, no. 1, pp. 35-45, Jan. 2011.

A. R. Iyengar, P. H. Siegel, R. L. Urbanke, J. K. Wolf, "Windowed Decoding of Spatially Coupled Codes", *Proc. IEEE International Symposium on Information Theory*, 31 Jul.-5 Aug. 2011, St. Petersburg, Russia.

A. R. Iyengar, P. H. Siegel, J. K. Wolf, "Modeling and Information Rates for Synchronization Error Channels", *Proc. IEEE International Symposium on Information Theory*, 31 Jul.-5 Aug. 2011, St. Petersburg, Russia.

A. Bhatia, A. R. Iyengar, P. H. Siegel, "Enhancing Binary Images of Non-Binary LDPC Codes", *Proc. IEEE Global Communications Conference*, 5-9 Dec. 2011, Houston, TX, USA.

A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, G. E. Corazza, "Windowed Decoding of Protograph-based LDPC Convolutional Codes over Erasure Channels", *IEEE Transactions on Information Theory*, vol. 58, no. 4, pp. , Apr. 2012.

A. R. Iyengar, P. H. Siegel, R. L. Urbanke, J. K. Wolf, "Windowed Decoding of Spatially Coupled Codes", *Submitted to the IEEE Transactions on Information Theory*, Mar. 2012.

A. R. Iyengar, P. H. Siegel, J. K. Wolf, "On the Capacity of Channels with Synchronization Errors", *To be submitted to the IEEE Transactions on Information Theory*, 2012.

ABSTRACT OF THE DISSERTATION

Synchronization Error Channels & Windowed Decoding in Theory & Practice

by

Aravind Raghava Iyengar

Doctor of Philosophy in Electrical Engineering
(Communication Theory & Systems)

University of California, San Diego, 2012

Professor Paul H. Siegel, Chair

Right from the birth of communication theory, synchronization errors have been a challenge. In the first part of this dissertation, we will consider a class of synchronization error channels and develop a rigorous information theoretic analysis. We provide analytical bounds on the capacity of channels that introduce deletions or replications. For channels that introduce deletions and replications, we develop methods to approximately estimate the achievable information rates. Following this, we consider specific applications in magnetic recording where synchronization errors play a key role. For these applications, we provide bounds and numerical estimates of the channel capacity as well as the zero-error capacity.

In the second part of the dissertation, we will focus on a coding theoretic problem of analyzing a low-complexity decoding scheme for spatially coupled codes over the erasure channel. We describe the operation of windowed decoding, and analytically establish its asymptotic performance limits. For protograph-based LDPC convolutional codes, which are a variant of the spatially coupled codes, we identify characteristics of code ensembles that result in good performance with the windowed decoding algorithm over erasure channels with and without memory.

Chapter 1

Introduction

The advent of digital communications paved way for fundamental changes in the way people communicate. With the birth of *information theory*, a meaningful quantification of information was possible, which led to a well defined notion of the "best speed" of communication. More importantly, it also gave rise to methods of handling signal distortion, which was one of the big roadblocks in analog communications. In other words, digitally communicated information was correctable in the presence of errors due to physical impairments of devices used for communications. *Coding theory* quickly emerged as a crucial topic in communications.

Despite the simultaneous development of information and coding theories, and despite their common motivation, it was not until the 90s that they converged with the discovery of codes and efficient decoding algorithms that approached the limits imposed by information theory. It is to this development that today's fast communication and storage devices, the internet, and all communication networks largely owe their birth.

1.1 Digital Communication System

A digital communication system basically consists of a message transmitter, a *channel* through which the message is communicated, and a receiver. Based on the channel of communication, the transmitter and the receiver are required to perform various operations to realize successful communication. For example, in a wireless communication system, the information is typically *modulated* before being transmitted. This is

done for various reasons including facilitating reception of signal with a small enough antenna and using frequencies that do not undergo heavy attenuation during transmission.

Most physical communication channels are *noisy*, i.e., the transmitted signals are typically impaired due to physical limitations or imperfections in the device. Communication devices are therefore required to provide mechanisms for error correction. The exact mechanism to be used is designed based on the specific application of the device and is known both to the transmitter and the receiver prior to transmission. The transmitter *encodes* information, and the receiver *decodes* it on reception based on its knowledge of the encoding.

1.2 Data Storage System

Although the description so far pertains to the case of communications, the same model applies for data storage as well. The storing of data is the transmission, the device on which data is stored is the channel, and subsequent reading of the data on the device is the reception. Thus, information is communicated from one time to a subsequent time. Devices on which data is stored, e.g., hard disks in computers, flash memories in tablets, phones etc., are also noisy. The data storing process is hence accompanied by encoding, and data retrieval by decoding. Due to this parallel, we will not distinguish between data communication and storage devices henceforth in this chapter, and refer to them as communication systems.

1.3 Information & Coding Theory

In order to study a digital communication system, a mathematical model that mimics the behavior of the communication system in consideration is first framed. Typically, communication systems are modeled probabilistically, where the randomness arises from the noise in the system. Information theory allows us to formulate fundamental limits on the performance of such a system in terms of the number of messages that can be sent each time the channel is used. Efficient coding schemes are then devised

for the system. Codes are ranked based on their ability to correct errors introduced in the channel, the complexity of the encoding and decoding operations, as well as the gap between the rate at which messages can be sent using these codes and the limit dictated by information theory.

For a class of communication systems, Shannon [103] showed that there is a maximum rate up to which reliable information transfer is possible, i.e., the probability of decoding to a wrong symbol vanishes to zero as the number of times the channel is used goes to infinity, and beyond which no reliable information transfer is possible, i.e., the error probability is strictly bounded away from zero. This limit on the rate of information transfer is called the *capacity* of the communication channel. Since this seminal work, similar results have been shown for various communication channels.

Capacity-achieving codes are those for which efficient encoding and decoding schemes are known for rates that are close to the capacity of the channel for which they are designed. Such codes had not been discovered until the advent of *Turbo codes* [13] and the rediscovery of *low-density parity-check* (LDPC) codes [80]. Recent developments have produced various types of capacity-achieving codes.

1.4 Dissertation Overview

This dissertation can be divided into two main parts. The first part, comprising Chapters 2 and 3, is information theoretic in nature. In Chapter 2, we study a class of communication channels that introduce *synchronization errors*, i.e., the number of symbols received is a random function of the number of symbols transmitted over the channel. Such a channel introduces *deletions* and *replications* of symbols transmitted over it. We establish bounds on the capacities of such channels. In Chapter 3, we consider synchronization error channels that arise in the context of data storage, and in particular, in magnetic recording. We employ similar strategies as in Chapter 2 to establish bounds on the capacities of these channels.

In the second part of the dissertation, comprising Chapters 4 and 5, we focus on a coding theoretic problem. We consider transmission on channels that introduce erasures and consider a low-complexity decoding algorithm for a class of LDPC codes. In Chap-

ter 4, we theoretically guarantee the good performance of the algorithm by establishing that the windowed decoding threshold approaches the belief propagation threshold at least exponentially fast in the size of the window. In Chapter 5, we identify characteristics of specific code ensembles that have good performance with the proposed decoding algorithm.

We start with a note on the notation used in the rest of the dissertation.

Notation

Non-random variables are typically written as lowercase letters, e.g., n. We denote sets by double-stroke uppercase letters, e.g., $\mathbb{X}$. We will reserve $\mathbb{N}$, $\mathbb{Z}$ and $\mathbb{R}$ to denote the sets of natural numbers, integers and real numbers, respectively. $\mathbb{Z}^+$ denotes the set of non-negative integers. We define

$$
\begin{aligned}
[n] &\triangleq \{1, 2, \cdots, n\}, n \in \mathbb{N}, \\
[0] &\triangleq \emptyset, \\
[m:n] &\triangleq \begin{cases} \{m, m+1, \cdots, n\}, & m \leq n, \\ \emptyset, & n < m. \end{cases} \text{ and} \\
\mathbb{Z}_{\pm m} &\triangleq \{-m, -m+1, \cdots, 0, 1, \cdots, m\} \; \forall \, m \in \mathbb{Z}^+.
\end{aligned}
$$

For some $n \in \mathbb{N}$, we will let $\mathbb{X}^n$ denote the set of vectors of dimension n with elements from $\mathbb{X}$. Vectors are denoted either underlined, $\underline{x}$, or in bold face, $\mathbf{x}$. Matrices are also typically written in bold face, e.g., $\mathbf{H}$. We will write $\bar{x}$ to denote a string, and λ to denote the *empty string*. The *length* of a string, denoted $|x|$, is the number of symbols in it, and by definition, $|\lambda| = 0$. With some abuse of notation, we will use "vectors of dimension n" and "strings of length n" interchangeably with the understanding that the string corresponding to a vector is just the concatenation of the symbols of the vector in order. The set of all strings of length n over the alphabet $\mathbb{X}$ is hence also denoted $\mathbb{X}^n$, and $\mathbb{X}^0 = \{\lambda\}$. We write $\overline{\mathbb{X}}$ to denote the set of all strings over the set $\mathbb{X}$, i.e.,

$$
\overline{\mathbb{X}} = \bigcup_{i=0}^{\infty} \mathbb{X}^i.
$$

The bar "$\overline{\cdot}$" will denote the concatenation operation, so that $\overline{\overline{x}\cdot\overline{y}}$ is the concatenation of strings $\overline{x}$ and $\overline{y}$.

In general, we assume an underlying probability space $(\mathbb{S},\mathscr{B},\mathsf{P})$ over which random variables, denoted by uppercase letters, e.g., X, are defined. Random vectors are denoted by uppercase letters with the *multiset* of indices as subscripts, e.g., $X_{[n]} = (X_1, X_2, \cdots, X_n)$, or $X_{Y_{[n]}}$ when the multiset of indices is itself the elements of a random vector $Y_{[n]}$. Random processes (assumed discrete-time) are denoted by script letters $\mathscr{X}$, or subscripted by the set of natural numbers, $X_{\mathbb{N}}$.

We will use the asymptotic notations $O(\cdot)$, $o(\cdot)$, $\omega(\cdot)$ as in [17, 59].

Chapter 2

Synchronization Error Channels : Theory

Channels with synchronization errors have been familiar to information and coding theorists and practitioners alike ever since the advent of the digital information era. Although Dobrushin [27] established the coding theorem for such channels as early as 1967, tackling these channels in terms of estimating information rates and constructing codes with good performance have proved to be very tough. In the last decade, significant progress has been made in estimating achievable information rates for certain channels with synchronization errors. However, a coding scheme with provably "good" performance remains elusive thus far.

In this chapter, we start with Dobrushin's model of channels with synchronization errors, henceforth referred to as the *synchronization error channel* (SEC), and convert it into an equivalent channel with states. Using this alternative model, we establish certain bounds on achievable information rates for the special cases of channels that introduce only deletions or only replications of input symbols. For the case of SECs that introduce deletions as well as replications, we construct sequences of channels that "approximate" the SEC, and whose limit is related to the SEC in terms of mutual information rates. Thus, through the use of these approximate channels, we derive some results about information rates achievable over the SEC. Although the motivation behind the formulation of SECs as channels with states is straightforward, its use to obtain non-trivial bounds on the capacity of the SEC has not been found in literature. While

this work concerns only a few asymptotic results on information rates of the SEC, we think that the model presented here can be utilized to design codes for SECs in general.

The remainder of this chapter is organized as follows. In Section 2.1, we revisit Dobrushin's model of an SEC and recall the main results on capacity of SECs. Through much of this chapter, we consider a special case of the generic SEC—the deletion, replication channel (DRC)—and construct an equivalent channel by viewing the DRC as a channel with states in Section 2.2. Under further special cases of channels with only deletions or only replications, we give some simple, non-trivial and sometimes tight bounds on the capacity in Sections 2.3.1 and 2.3.2. We then construct sequences of approximate channels for the DRC and establish certain properties of this sequence of channels that allow us to bound the capacity of the DRC in Section 2.4. In Section 2.5, we note the application of similar strategies to more general SECs, and we conclude with summary and remarks in Section 2.6.

2.1 Synchronization Error Channels

Remark 2.1 (Notation). Non-random variables are written as lowercase letters, e.g., n. We denote sets by double-stroke uppercase letters, e.g., $\mathbb{X}$. We will reserve $\mathbb{N}$, $\mathbb{Z}$ and $\mathbb{R}$ to denote the sets of natural numbers, integers and real numbers, respectively. $\mathbb{Z}^+$ denotes the set of non-negative integers. We define

$$
\begin{aligned}
[n] &\triangleq \{1,2,\cdots,n\}, n \in \mathbb{N},\\
[0] &\triangleq \emptyset,\\
[m:n] &\triangleq \begin{cases} \{m, m+1, \cdots, n\}, & m \leq n,\\ \emptyset, & n < m. \end{cases} \text{ and}\\
\mathbb{Z}_{\pm m} &\triangleq \{-m, -m+1, \cdots, 0, 1, \cdots, m\} \; \forall \; m \in \mathbb{Z}^+.
\end{aligned}
$$

For some $n \in \mathbb{N}$, we will let $\mathbb{X}^n$ denote the set of vectors of dimension n with elements from $\mathbb{X}$. We will write $\bar{x}$ to denote a string, and λ to denote the *empty string*. The *length* of a string, denoted $|x|$, is the number of symbols in it, and by definition, $|\lambda| = 0$. With some abuse of notation, we will use "vectors of dimension n" and "strings of length n" interchageably. The set of all strings of length n over the alphabet $\mathbb{X}$ is

hence also denoted $\mathbb{X}^n$, and $\mathbb{X}^0 = \{\lambda\}$. We write $\overline{\mathbb{X}}$ to denote the set of all strings over the set $\mathbb{X}$, i.e.,

$$\overline{\mathbb{X}} = \bigcup_{i=0}^{\infty} \mathbb{X}^i.$$

The bar "$\overline{\cdot}$" will denote the concatenation operation, so that $\overline{\overline{x} \cdot \overline{y}}$ is the concatenation of strings $\overline{x}$ and $\overline{y}$.

Throughout this chapter, we assume an underlying probability space $(\mathbb{S}, \mathscr{B}, \mathsf{P})$ over which random variables, denoted by uppercase letters, e.g., X, are defined. Random vectors are denoted by uppercase letters with the *multiset* of indices as subscripts, e.g., $X_{[n]} = (X_1, X_2, \cdots, X_n)$, or $X_{Y_{[n]}}$ when the multiset of indices is itself the elements of a random vector $Y_{[n]}$. Random processes (assumed discrete-time) are denoted by script letters $\mathscr{X}$, or subscripted by the set of natural numbers, $X_{\mathbb{N}}$.

We will use the asymptotic notations $O(\cdot)$, $o(\cdot)$, $\omega(\cdot)$ as in [17, 59]. □

We start by defining the synchronization error channels as considered by Dobrushin [27].

Definition 2.1 (Memoryless SECs)**.** *Let $\mathbb{X}$ be a finite set. A* memoryless *synchronization error channel is specified by a stochastic matrix*

$$\{q(\overline{y}|x), \overline{y} \in \overline{\mathbb{Y}}, x \in \mathbb{X}\}$$

where $\mathbb{Y}$ is the output alphabet. From the properties of a stochastic matrix, we have

$$0 \leq q(\overline{y}|x) \leq 1, \qquad \sum_{\overline{y} \in \overline{\mathbb{Y}}} q(\overline{y}|x) = 1 \ \forall\, x \in \mathbb{X}. \tag{2.1}$$

Further, we will assume that the mean value of the length of the output string arising from one input symbol is strictly positive and finite, i.e.,

$$0 < \sum_{\overline{y} \in \overline{\mathbb{Y}}} |\overline{y}| q(\overline{y}|x) < \infty. \tag{2.2}$$

For $x_{[n]} = (x_1, x_2, \cdots, x_n) \in \mathbb{X}^n$ and $\overline{y}_{[n]} = (\overline{y}_1, \overline{y}_2, \cdots, \overline{y}_n) \in \overline{\mathbb{Y}}^n$, we write

$$q_n(\overline{y}_{[n]}|x_{[n]}) = \prod_{i=1}^{n} q(\overline{y}_i|x_i).$$

Let $\overline{\overline{y}_{[n]}}$ denote the concatenation of strings $\overline{y}_i, i \in [n]$. Then the transition probabilities of the memoryless SEC are defined as

$$Q_n(\overline{y}|x_{[n]}) = \sum_{\overline{\overline{y}_{[n]}}=\overline{y}} q_n(\overline{y}_{[n]}|x_{[n]}) \tag{2.3}$$

for $\overline{y} \in \overline{\mathbb{Y}}$ and $x_{[n]} \in \mathbb{X}^n$. The memoryless SEC is given by the triplet $\mathbf{Q}_n \triangleq (\mathbb{X}, Q_n, \mathbb{Y})$, the input and the output alphabets, and the transition probabilities between input strings of length n and all output strings. □

Consider the sequence of memoryless SECs $\{\mathbf{Q}_n\}_{n=1}^{\infty}$. Then, we have the following.

Theorem 2.1 (Capacity [27])**.** *Let $X_{[n]}$ and $\overline{Y}$ denote the input and the output of the SEC $\mathbf{Q}_n$. Let*

$$C_n = \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I(X_{[n]}; \overline{Y}).$$

Then,

$$C = \lim_{n\to\infty} C_n = \inf_{n\geq 1} C_n$$

exists and is equal to the capacity of the sequence of SECs. □

The quantity C represents the maximum rate at which information can be transferred over the SEC with vanishing error probability. Furthermore, the following result shows that, in estimating the capacity of the SEC, we can restrict ourselves to a subclass of possible input processes $\mathscr{X}$.

Proposition 2.2 (Markov Capacity [27])**.** *Let $\mathscr{X}_{\mathscr{M}}$ be a stationary, ergodic, Markov process over $\mathbb{X}$. Then the capacity of the sequence $\{\mathbf{Q}_n\}_{n=1}^{\infty}$ is*

$$C = \sup_{\mathscr{X}_{\mathscr{M}}} \lim_{n\to\infty} \frac{1}{n} I(X_{[n]}; \overline{Y}).$$

The capacity is therefore the supremum of the rates achievable through stationary, ergodic, Markov processes $\mathscr{X}_{\mathscr{M}}$. □

We will now give an example of a memoryless SEC. For convenience, we will assume that the input alphabet for the SECs is $\mathbb{X} = \{0, 1\}$, i.e., the channels considered are *binary* memoryless SECs. However, we note that all the results presented here can be straightforwardly extended to the case where $\mathbb{X}$ is any finite set.

Example 2.1 (Deletion-Replication Channel (DRC)). Consider the binary SEC with $\mathbb{X} = \mathbb{Y} = \{0,1\}$ and the following stochastic matrix.

$$q(\bar{y}|x) = \begin{cases} p_{\mathsf{d}}, & \bar{y} = \lambda \\ p_{\mathsf{t}} p_{\mathsf{r}}^{\ell-1}, & \bar{y} = x^{\ell},\ \forall\ \ell \geq 1. \end{cases}$$

Intuitively, we can think of p_{d} as the deletion probability, p_{t} as the transmission probability, and p_{r} as the replication probability, i.e., when $x \in \mathbb{X}$ is sent, it is either deleted with probability p_{d}, or transmitted and replicated $(\ell-1)$ times with probability $p_{\mathsf{t}} p_{\mathsf{r}}^{\ell-1}$ for $\ell \geq 1$. From (2.1), we get for $p_{\mathsf{r}} < 1$

$$p_{\mathsf{d}} + \sum_{\ell=1}^{\infty} p_{\mathsf{t}} p_{\mathsf{r}}^{\ell-1} = p_{\mathsf{d}} + \frac{p_{\mathsf{t}}}{1-p_{\mathsf{r}}} = 1,$$

or equivalently,

$$p_{\mathsf{t}} = (1-p_{\mathsf{d}})(1-p_{\mathsf{r}}). \tag{2.4}$$

From (2.2), we have that

$$0 < \sum_{\ell=1}^{\infty} \ell p_{\mathsf{t}} p_{\mathsf{r}}^{\ell-1} = \frac{p_{\mathsf{t}}}{(1-p_{\mathsf{r}})^2} = \frac{1-p_{\mathsf{d}}}{1-p_{\mathsf{r}}} < \infty$$

where we use Equation (2.4). Hence $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in [0,1)^2$. Note that when $p_{\mathsf{r}} = 0$, the DRC is the same as the *binary deletion channel* (BDC); and when $p_{\mathsf{d}} = 0$, it is the *binary replication channel* (BRC), also referred to as the *geometric binary sticky channel* [84]. □

The BDC has been the most well-studied SEC. In [85], the author surveys the results that were known prior to 2009. To summarize, the best known lower bounds were obtained, chronologically, through bounds on the cutoff rate for sequential decoding [37], bounding the rate with a first-order Markov input [23], reduction to a Poisson-repeat channel [86], analyzing a "jigsaw-puzzle" coding scheme [28], or by directly bounding the information rate by analyzing the channel as a joint renewal process [58]. Recently, [55] and [54] independently gave the capacity of a BDC with small deletion probabilities, and showed that it is achieved by independent and uniformly distributed (i.u.d.) inputs. The known upper bounds for the BDC have been obtained by genie-aided decoder arguments [24, 35]. An idea from [35] was extended to obtain some

analytical lower bounds on the capacity of channels that involve substitution errors as well as insertions or deletions [96]. The idea in [55] was extended to obtain a better approximation for the capacity of the BDC with small deletion probabilities in [56].

In the case of the BDC, in contrast to these existing results, our approach explicitly characterizes the achievable information rates in terms of "subsequence-weights", which is a measure relevant in ML decoding for the BDC [85]. Additionally, the method proposed here gives the tight bound on capacity for small deletion probabilities obtained in [55] more directly[1].

For the BRC, [84] obtained lower bounds on the capacity by numerically estimating the capacity per unit cost of the equivalent channel of runs through optimization of 8 and 16 bit codes. Here, we obtain direct analytical lower bounds on the capacity. These, to the best of our knowledge, represent the only analytical bounds for the capacity of the BRC. Moreover, we obtain an exact expression for the Markov-1 rate for the BRC which conclusively disproves the conjecture that the capacity of SECs is a convex function of the channel parameter.

We will use the DRC as a running example of an SEC. In Section 2.5, we discuss the extension of the results presented to more general classes of SECs.

2.2 DRC as a Channel with States

We now construct a channel with states that is equivalent to the DRC introduced in Example 2.1. Dobrushin's model of SEC (cf. Definition 2.1) tracks the output string generated by each input symbol. In our model, we track the input symbol that gave rise to each output symbol.

2.2.1 Channel Model

Definition 2.2 (DRC with states)**.** *For a fixed $n \in \mathbb{N}$, we write*

$$Y_i = X_{\Gamma_i} = X_{i-Z_i}, i \in [N_n] \tag{2.5}$$

[1] Note that although we obtain the same lower bound for the capacity of the BDC as in [55], we do not prove a converse here.

$$
\begin{aligned}
P_n(\overline{y}|x_{[n]}, Z_0 = 0) &= \sum_{\{\overline{z}:|\overline{z}|=|\overline{y}|\}} \mathsf{P}(\overline{Z} = \overline{z}, \overline{Y} = \overline{y}|X_{[n]} = x_{[n]}, Z_0 = 0) \\
&= \sum_{\{\overline{z}:|\overline{z}|=|\overline{y}|\}} \mathsf{P}(\overline{Z} = \overline{z}|Z_0 = 0)\mathsf{P}(\overline{Y} = \overline{y}|X_{[n]} = x_{[n]}, Z_0 = 0, \overline{Z} = \overline{z}) \\
&= \sum_{\{\overline{z}:|\overline{z}|=|\overline{y}|\}} \prod_{i=1}^{|\overline{y}|} \Big(\mathsf{P}(Z_i = z_i|Z_{i-1} = z_{i-1}, Z_0 = 0) \\
&\qquad\qquad \cdot \mathsf{P}(Y_i = y_i|X_{[n]} = x_{[n]}, Z_i = z_i) \Big) \qquad (2.7) \\
&= \sum_{\{\overline{z}:|\overline{z}|=|\overline{y}|\}} \prod_{i=1}^{|\overline{y}|} \Big(\mathsf{P}(Z_i = z_i|Z_{i-1} = z_{i-1}) \mathbb{1}_{\{y_i = x_{i-z_i}\}} \Big).
\end{aligned}
$$

---

where $Z_i \in \mathbb{Z}$ is the "state" of the channel and

$$N_n \triangleq \sup\{i \geq 0 : \Gamma_i \leq n | \Gamma_0 = 0\}.$$

We will refer to the random variable N_n as the output length *corresponding to n input symbols. The state process $\mathscr{Z}$ is independent of the channel input process, and is a first-order Markov process over the set of integers $\mathbb{Z}$ with transition probabilities for each $i \in \mathbb{N}$ given by*

$$
\begin{aligned}
&\mathsf{P}(Z_i = z_i|Z_{i-1} = z_{i-1}) \\
&\qquad = \begin{cases} p_{\mathsf{r}}, & z_i = z_{i-1} + 1 \\ p_{\mathsf{t}} p_{\mathsf{d}}^{\ell}, & z_i = z_{i-1} - \ell, \ \forall\, \ell \geq 0, \end{cases} \qquad (2.6)
\end{aligned}
$$

where we define p_{t} as in Equation (2.4) *assuming $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in [0,1)^2$. We will refer to the process $\Gamma \triangleq \Gamma_{\mathbb{N}}$ where $\Gamma_i = i - Z_i$ as the* index process.

We also assume the boundary condition that $Z_0 = \Gamma_0 = 0$, i.e., the channel is perfectly synchronized before transmission commences. Note that the transition probabilities in (2.6) *indeed are well-defined since $\forall\, z_{i-1} \in \mathbb{Z}$, as $p_{\mathsf{d}} < 1$,*

$$\sum_{z_i} p(z_i|z_{i-1}) = p_{\mathsf{r}} + \sum_{\ell=0}^{\infty} p_{\mathsf{t}} p_{\mathsf{d}}^{\ell} = p_{\mathsf{r}} + \frac{p_{\mathsf{t}}}{1 - p_{\mathsf{d}}} = 1.$$

With the above definition, for $\overline{y} \in \overline{\mathbb{Y}}$ and $x_{[n]} \in \mathbb{X}^n$, the channel transition probabilities are given as in Equation (2.7). *Note that in the terms within the parenthesis*

on the right hand side of Equation (2.7)*, the first term is completely specified by the transition probabilities* (2.6) *of the channel state process* $\mathscr{Z}$*, and the second term is* 0 *or* 1 *accordingly as* $y_i \neq x_{i-z_i}$ *or* $y_i = x_{i-z_i}$ *respectively.*

For each $n \in \mathbb{N}$*, we define the DRC with states as the channel specified by the triplet* $\mathbf{P}_n \triangleq (\mathbb{X}, P_n, \mathbb{Y})$. □

We will start by proving a few properties of the output length N_n and the channel state $\mathscr{Z}$ and index processes Γ which will be made use of subsequently.

Lemma 2.3 (Properties of N_n)**.** *The output length* N_n *satisfies the following properties:*

(i) For any $n \in \mathbb{N}$*,* $N_n < \infty$ *a.s..*

(ii) $N_n \to \infty$ *as* $n \to \infty$ *a.s..*

(iii) $\frac{N_n}{n} \to \frac{1-p_\mathsf{d}}{1-p_\mathsf{r}}$ *a.s. as* $n \to \infty$*.*

Proof. (i) This is true since $p_\mathsf{r} < 1$.

(ii) Since $p_\mathsf{d} < 1$.

(iii) Notice that, for each $n \in \mathbb{N}$, we can write

$$\Gamma_n = n - Z_n = \sum_{i=1}^{n} \Delta_i$$

where the Δ_i's are i.i.d. with

$$\mathsf{P}(\Delta_1 = \delta) = \begin{cases} p_\mathsf{r}, & \delta = 0 \\ p_\mathsf{t} p_\mathsf{d}^{\delta-1}, & \delta \geq 1. \end{cases}$$

From the strong law of large numbers (SLLN), we therefore have $\frac{\Gamma_n}{n} \to \mathsf{E}(\Delta_1)$ a.s. as $n \to \infty$. We also have $N_n \to \infty$ a.s. as $n \to \infty$ from point (ii) above. Therefore, $\frac{\Gamma_{N_n}}{N_n} \to \mathsf{E}(\Delta_1)$ a.s. as $n \to \infty$. Further, by definition, we have $\Gamma_{N_n} \leq n < \Gamma_{N_n+1}$, i.e.,

$$\frac{\Gamma_{N_n}}{N_n} \leq \frac{n}{N_n} \leq \frac{\Gamma_{N_n+1}}{N_n+1}\Big(\frac{N_n+1}{N_n}\Big).$$

Thus $\frac{N_n}{n} \to \frac{1}{\mathsf{E}(\Delta_1)} = \frac{1-p_\mathsf{d}}{1-p_\mathsf{r}}$ a.s. as $n \to \infty$.

□

Lemma 2.4 (Properties of $\mathscr{Z},\Gamma$)**.** *The channel state process $\mathscr{Z}$ and the index process Γ satisfy the following properties:*

(i) *$\mathscr{Z}$ and Γ are first-order, time-homogeneous, shift-invariant Markov chains. Further, $\mathscr{Z}$ is irreducible and aperiodic.*

(ii) *Γ is almost surely non-decreasing, i.e.,*

$$\Gamma_{i+j} \geq \Gamma_i \ \forall\ j \geq 0, i \in \mathbb{N} \text{ a.s.}.$$

For any $n \in \mathbb{N}$, a realization of $Z_{[n]}$ such that the corresponding $\Gamma_{[n]}$ realization satisfies the above monotonicity property is called a compatible *state path.*

(iii) *For every $i \in \mathbb{N}$,*

$$H(Z_i|Z_{i-1}) = H(\Gamma_i|\Gamma_{i-1}) = h_2(p_{\mathsf{r}}) + \frac{1-p_{\mathsf{r}}}{1-p_{\mathsf{d}}} h_2(p_{\mathsf{d}}),$$

where $h_2(x) \triangleq -x\log_2 x - (1-x)\log_2(1-x)$, for $x \in [0,1]$, is the binary entropy function *[18]. Here, we assume from continuity that $0\log_2 0 = 0$. Consequently, for every $n \in \mathbb{N}$,*

$$H(Z_{[n]}) = H(\Gamma_{[n]}) = n\Big(h_2(p_{\mathsf{r}}) + \frac{1-p_{\mathsf{r}}}{1-p_{\mathsf{d}}} h_2(p_{\mathsf{d}})\Big).$$

Proof. (i) By definition, $\mathscr{Z}$ is a first-order Markov chain. Time-homogeneity implies that

$$\mathsf{P}(Z_i|Z_{i-1}) = \mathsf{P}(Z_1|Z_0) \ \forall\ i \geq 1.$$

This is true for the state process $\mathscr{Z}$ from the definition since the transition probabilities in Equation (2.6) do not depend on the time index i. Shift-invariance implies

$$\mathsf{P}(Z_1 = z_1|Z_0 = z_0) = \mathsf{P}(Z_1 = z_1 - z_0|Z_0 = 0).$$

This is true because the state transition probabilities in (2.6) depend only on the difference $z_i - z_{i-1}$.

The Γ process inherits these properties from $\mathscr{Z}$ through the bijection $\zeta : \mathbb{Z}^n \mapsto \mathbb{Z}^n$, where with some abuse of notation, we write $\Gamma_{[n]} = \zeta(Z_{[n]}) = (\zeta(Z_i)), i \in [n]$, with $\Gamma_i = \zeta(Z_i) = i - Z_i, i \in [n],\ \forall\ n \in \mathbb{N}$.

The irreducibility and aperiodicity of the $\mathscr{Z}$ process follow from the definition.

(ii) Note that from Equation (2.6), $Z_{i+1} \leq Z_i + 1$ a.s. for every $i \geq 0$. Hence $Z_{i+j} \leq Z_i + j$ a.s. for every $i \geq 0, j \geq 0$. Since $\Gamma_i = i - Z_i$, we have $\Gamma_{i+j} = i + j - Z_{i+j} \geq i + j - Z_i - j = \Gamma_i$ with probability 1.

(iii) From the bijection ζ (See point (i) above) and Lemma 2.3, we have

$$\begin{aligned} H(Z_i|Z_{i-1}) &= H(\Gamma_i|\Gamma_{i-1}) = H(\Gamma_{i-1} + \Delta_i|\Gamma_{i-1}) \\ &= H(\Delta_i|\Gamma_{i-1}) = H(\Delta_i) = H(\Delta_1) \\ &= h_2(p_{\mathsf{r}}) + \frac{1-p_{\mathsf{r}}}{1-p_{\mathsf{d}}} h_2(p_{\mathsf{d}}). \end{aligned}$$

Hence

$$H(Z_{[n]}) = \sum_{i=1}^{n} H(Z_i|Z_{[i-1]}) = \sum_{i=1}^{n} H(Z_i|Z_{i-1}) = n\Big(h_2(p_{\mathsf{r}}) + \frac{1-p_{\mathsf{r}}}{1-p_{\mathsf{d}}} h_2(p_{\mathsf{d}})\Big).$$

□

Note that the $\mathscr{Z}$ process is *not* stationary because we fix $Z_0 = 0$. The Γ process is clearly not stationary since Γ_i depends on i. From Lemma 2.4 (ii), we can show that for $1 \leq n \leq m < \infty$,

$$N_n \leq N_m \text{ a.s..} \tag{2.8}$$

Proposition 2.5 (Channel Equivalence)**.** *For each $n \in \mathbb{N}$, the channels $\mathbf{Q}_n$ and $\mathbf{P}_n$ are equivalent.*

Proof. Both $\mathbf{Q}_n$ and $\mathbf{P}_n$ have the same input and output alphabets $\mathbb{X}$ and $\mathbb{Y}$, respectively. The correspondence between the transition probabilities Q_n and P_n in Equations (2.3) and (2.7) is evident by the following observations:

(i) For every parsing of $\overline{y} \in \overline{\mathbb{Y}}$ as $\overline{y}_{[n]}$ in Equation (2.3), there is a corresponding state path $\overline{z} \in \overline{Z}$ in Equation (2.7).

(ii) For every compatible state path $\overline{z} \in \overline{Z}$ in Equation (2.7) (See Lemma 2.4), there is a corresponding parsing of $\overline{y} \in \overline{\mathbb{Y}}$ in Equation (2.3).

(iii) For these corresponding parsings of $\overline{y}$ and compatible state paths $\overline{z}$, the terms within the parenthesis on the right hand side of Equation 2.7, when grouped according to the output symbols arising from the same input symbol, spell out exactly the same probability as the terms $q(\overline{y}_i|x_i)$.

Therefore, except on a set of zero probability (state paths that are not compatible), the probability measures Q_n and P_n are equal. This implies the equivalence of the channels $\mathbf{Q}_n$ and $\mathbf{P}_n$. □

As a consequence of the above equivalence, the results of Theorem 2.1 and Proposition 2.2 carry forward to the sequence of channels $\{\mathbf{P}_n\}_{n=1}^{\infty}$ specified by Equations (2.5) and (2.6).

Corollary 2.6 (Dobrushin's results for $\{\mathbf{P}_n\}_{n=1}^{\infty}$)**.** *For input $X_{[n]}$ and output $Y_{[N_n]}$ of the channel $\mathbf{P}_n$, the quantity*

$$\begin{aligned} C &= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I(X_{[n]};Y_{[N_n]}) \\ &= \sup_{\mathscr{X}_{\mathscr{M}}} \lim_{n\to\infty} \frac{1}{n} I(X_{[n]};Y_{[N_n]}), \end{aligned}$$

where $\mathscr{X}_{\mathscr{M}}$ represents stationary, ergodic, Markov processes over $\mathbb{X}$, exists and is equal to the capacity of the sequence of channels $\{\mathbf{P}_n\}_{n=1}^{\infty}$. □

We will henceforth restrict our attention to this class of input processes.

Proposition 2.7 (Stationarity)**.** *The channel output process $\mathscr{Y}$ is stationary for stationary input processes $\mathscr{X}$.*

Proof. Let $Z_0 = 0$ and consider semi-infinite input, state and output processes. We first note that $\forall\ k,l \in \mathbb{N}, k \leq l$,

$$\begin{aligned} &\mathsf{P}(Y_{[k:l]} = y_{[k:l]} | Z_{k-1} = 0) \\ &\qquad = \sum_{\gamma_{[k:l]}} \mathsf{P}(\Gamma_{[k:l]} = \gamma_{[k:l]}, X_{\gamma_{[k:l]}} = y_{[k:l]} | \Gamma_{k-1} = k-1) \\ &\qquad \overset{(a)}{=} \sum_{\gamma_{[k:l]}} \mathsf{P}(\Gamma_{[k:l]} = \gamma_{[k:l]} - z, X_{\gamma_{[k:l]}} = y_{[k:l]} | \Gamma_{k-1} = k-1-z) \\ &\qquad \overset{(b)}{=} \sum_{\gamma_{[k:l]}} \mathsf{P}(\Gamma_{[k:l]} = \gamma_{[k:l]} - z, X_{\gamma_{[k:l]}-z} = y_{[k:l]} | \Gamma_{k-1} = k-1-z) \\ &\qquad = \mathsf{P}(Y_{[k:l]} = y_{[k:l]} | \Gamma_{k-1} = k-1-z) \\ &\qquad = \mathsf{P}(Y_{[k:l]} = y_{[k:l]} | Z_{k-1} = z)\ \forall\ z \leq k-1. \end{aligned}$$

Here, (a) follows from the shift-invariance of Γ (See Lemma 2.4) and (b) from the stationarity of $\mathscr{X}$. Therefore, we have

$$\begin{aligned}
\mathsf{P}(Y_{[k]} = y_{[k]}) &= \sum_{z\in\mathbb{Z}} \mathsf{P}(Z_0 = z)\mathsf{P}(Y_{[k]} = y_{[k]}|Z_0 = z) \\
&= \mathsf{P}(Y_{[k]} = y_{[k]}|Z_0 = 0) \\
&= \mathsf{P}(Y_{[k]} = y_{[k]}|\Gamma_0 = 0) \\
&= \sum_{\gamma_{[k]}} \mathsf{P}(\Gamma_{[k]} = \gamma_{[k]}, X_{\gamma_{[k]}} = y_{[k]}|\Gamma_0 = 0) \\
&\overset{(c)}{=} \sum_{\gamma_{[k]}} \mathsf{P}(\Gamma_{[j+1:j+k]} = \gamma_{[k]}, X_{\gamma_{[k]}} = y_{[k]}|\Gamma_j = 0) \\
&= \mathsf{P}(Y_{[j+1:j+k]} = y_{[k]}|Z_j = 0) \\
&= \mathsf{P}(Y_{[j+1:j+k]} = y_{[k]}) \ \forall\ j,k \in \mathbb{N}
\end{aligned}$$

where (c) follows form the time-homogeneity of Γ (Lemma 2.4). The last equality above follows from the observation made in the beginning of the proof. □

As a consequence of the above result, the *entropy rate* $\mathscr{H}(\mathscr{Y})$ of the output process is well-defined [18].

2.2.2 Bounds on the Capacity of the DRC

The formulation of the DRC as a channel with states allows us to immediately establish the following.

Proposition 2.8 (Simple bounds on C)**.** *For the DRC,*

$$(1-p_{\mathsf{d}})\Big(1-\frac{h_2(p_{\mathsf{r}})}{1-p_{\mathsf{r}}}\Big) - h_2(p_{\mathsf{d}}) \le C \le 1-p_{\mathsf{d}}.$$

Proof. We can write

$$\begin{aligned}
I(X_{[n]};Y_{[N_n]}) &= I(X_{[n]};Y_{[N_n]},Z_{[N_n]}) - I(X_{[n]};Z_{[N_n]}|Y_{[N_n]}) \\
&\overset{(a)}{=} I(X_{[n]};Y_{[N_n]}|Z_{[N_n]}) - I(X_{[n]};Z_{[N_n]}|Y_{[N_n]}) \\
&\overset{(b)}{=} (1-p_{\mathsf{d}})H(X_{[n]}) - I(X_{[n]};Z_{[N_n]}|Y_{[N_n]}),
\end{aligned} \tag{2.9}$$

where (a) is true because $\mathscr{X} \perp\!\!\!\perp \mathscr{Z}$ and (b) from the fact that the DRC, given the $\mathscr{Z}$ process realization, is equivalent to a binary erasure channel (BEC) with erasure rate p_{d}. Then,

$$n(1-p_{\mathsf{d}}) \geq I(X_{[n]};Y_{[N_n]}) \geq (1-p_{\mathsf{d}})H(X_{[n]}) - H(Z_{[N_n]}).$$

From Lemma 2.4 (iii), and since, for any finite n, we have the extra knowledge that $Z_i \geq i-n$ by definition of N_n, we can show that

$$H(Z_{[N_n]}) \leq \mathsf{E}(N_n)\Big(h_2(p_{\mathsf{r}}) + \frac{1-p_{\mathsf{r}}}{1-p_{\mathsf{d}}}h_2(p_{\mathsf{d}})\Big).$$

Note that the extra information $Z_i \geq i-n$ becomes tautological when $n \to \infty$, and hence

$$\lim_{n\to\infty}\frac{H(Z_{[N_n]})}{n} = \Big(\lim_{n\to\infty}\frac{\mathsf{E}(N_n)}{n}\Big)\Big(h_2(p_{\mathsf{r}}) + \frac{1-p_{\mathsf{r}}}{1-p_{\mathsf{d}}}h_2(p_{\mathsf{d}})\Big).$$

From Lemma 2.3, and for independent uniformly distributed inputs, the claim follows. □

Proposition 2.8 gives bounds on the capacity for $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in [0,1)^2$. Three special cases of the DRC are of particular interest: the binary deletion channel (BDC) with $p_{\mathsf{d}} = p, p_{\mathsf{r}} = 0$; the *symmetric* deletion-replication channel (SDRC) with $p_{\mathsf{d}} = p_{\mathsf{r}} = p$; and the binary replication channel (BRC) with $p_{\mathsf{d}} = 0, p_{\mathsf{r}} = p$. Specializing Proposition 2.8 to these cases gives us the following results.

Corollary 2.9 (Bounds on C for special cases)**.** *We have*

$$\begin{aligned} 1-p-h_2(p) \leq C_{\text{BDC}} &\leq 1-p, \\ 1-p-2h_2(p) \leq C_{\text{SDRC}} &\leq 1-p, \\ 1-\frac{h_2(p)}{1-p} \leq C_{\text{BRC}} &\leq 1. \end{aligned}$$

□

Although the bounds in Corollary 2.9 have simple closed-form expressions with well known information theoretic functions, they are loose compared to the best known (analytical or numerical) bounds for the capacity of these channels. We can, however, improve these bounds. We have from Equation (2.9),

$$\begin{aligned} I(X_{[n]};Y_{[N_n]}) = (1-p_{\mathsf{d}})H(X_{[n]}) &+ I(Y_{[N_n]};Z_{[N_n]}) \\ &- H(Z_{[N_n]}) + H(Z_{[N_n]}|X_{[n]},Y_{[N_n]}). \end{aligned} \tag{2.10}$$

Writing the entropy rate of the input process $\mathscr{X}$ as $\mathscr{H}(\mathscr{X})$ and defining

$$\hat{\mathscr{H}}(\mathscr{Z}) \triangleq \lim_{n\to\infty} \frac{H(Z_{[N_n]})}{n}, \quad \hat{\mathscr{H}}(\mathscr{Y}) \triangleq \lim_{n\to\infty} \frac{H(Y_{[N_n]})}{n},$$
$$\text{and} \quad \hat{\mathscr{H}}(\mathscr{Z}|\mathscr{X},\mathscr{Y}) \triangleq \lim_{n\to\infty} \frac{H(Z_{[N_n]}|X_{[n]},Y_{[N_n]})}{n},$$

from Lemma 2.4 and Equation (2.10), we can bound

$$C \geq \sup_{\mathscr{X}} \Big((1-p_{\mathsf{d}})\mathscr{H}(\mathscr{X}) + \hat{\mathscr{H}}(\mathscr{Z}|\mathscr{X},\mathscr{Y}) \Big) - \frac{1-p_{\mathsf{d}}}{1-p_{\mathsf{r}}} h_2(p_{\mathsf{r}}) - h_2(p_{\mathsf{d}}).$$

Lemma 2.10. *Let $H_n \triangleq \frac{1}{n} H(Z_{[N_n]}|X_{[n]},Y_{[N_n]})$ for $n \in \mathbb{N}$. Then, for the sequence $\{H_n\}_{n=1}^{\infty}$,*

$$\hat{\mathscr{H}}(\mathscr{Z}|\mathscr{X},\mathscr{Y}) = \lim_{n\to\infty} H_n = \sup_{n\geq 1} H_n.$$

Proof. From Equation (2.8), we can write

$$\begin{aligned}
(i+j)H_{i+j} &= H(Z_{[N_{i+j}]}|X_{[i+j]},Y_{[N_{i+j}]}) \\
&= H(Z_{[N_i]}|X_{[i+j]},Y_{[N_{i+j}]}) + H(Z_{[N_i+1:N_{i+j}]}|X_{[i+j]},Y_{[N_{i+j}]},Z_{[N_i]}) \\
&\geq H(Z_{[N_i]}|X_{[i+j]},Y_{[N_{i+j}]},N_i) + H(Z_{[N_i+1:N_{i+j}]}|X_{[i+j]},Y_{[N_{i+j}]},Z_{[N_i]},N_i) \\
&\overset{(a)}{=} H(Z_{[N_i]}|X_{[i]},Y_{[N_i]}) + H(Z_{[N_i+1:N_{i+j}]}|X_{[i+1:i+j]},Y_{[N_i+1:N_{i+j}]},Z_{N_i}) \\
&= iH_i + jH_j.
\end{aligned}$$

In the above, the equality labeled (a) follows from the conditional independence of $Z_{[N_i]}$ and $Z_{[N_i+1:N_{i+j}]}$ on $(X_{[i+1:i+j]},Y_{[N_i+1:N_{i+j}]})$ and $(X_{[i]},Y_{[N_i]},Z_{[N_i-1]})$ respectively, given N_i. From Fekete's Lemma [32, Appendix II], this superadditivity proves the claim. □

The above result implies that if we could evaluate (or lower bound) H_n for some n, that could be used to estimate a lower bound on C.

Proposition 2.11. *For the DRC,*

$$C \geq \sup_{\mathscr{X}} \Big(\mathscr{H}(\mathscr{X}) + \frac{H(Z_1|\mathscr{X},\mathscr{Y})}{1-p_{\mathsf{r}}} \Big)(1-p_{\mathsf{d}}) - \frac{1-p_{\mathsf{d}}}{1-p_{\mathsf{r}}} h_2(p_{\mathsf{r}}) - h_2(p_{\mathsf{d}}).$$

Proof. We have

$$\begin{aligned}
H_n &= \frac{1}{n} H(Z_{[N_n]}|X_{[n]},Y_{[N_n]}) \\
&= \frac{1}{n} \mathsf{E}\Big(\sum_{i=1}^{N_n} H(Z_i|Z_{[i-1]}=z_{[i-1]},X_{[n]},Y_{[N_n]}) \Big) \\
&= \frac{1}{n} \mathsf{E}\Big(\sum_{i=1}^{N_n} H(Z_i|Z_{i-1}=z_{i-1},X_{[i-1-z_{i-1}:n]},Y_{[i:N_n]}) \Big)
\end{aligned}$$

where the last equality follows from the conditional independence of Z_i on $Z_{[i-2]}$ and $X_{[i-2-Z_{i-1}]}$ and $Y_{[i-1]}$, given Z_{i-1}. From the time-homogeneity and shift-invariance of the $\mathscr{Z}$ process (See Lemma 2.4), as $n \to \infty$, the summand in the above expression

$$H(Z_i|Z_{i-1} = z_{i-1}, X_{[i-1-z_{i-1}:n]}, Y_{[i:N_n]}) \to H(Z_1|Z_0 = 0, X_{\mathbb{N}}, Y_{\mathbb{N}}) \triangleq H(Z_1|\mathscr{X}, \mathscr{Y}).$$

Since $\frac{\mathsf{E}(N_n)}{n} \to \frac{1-p_{\mathsf{d}}}{1-p_{\mathsf{r}}}$, optimizing over input processes $\mathscr{X}$ gives us the desired result. □

It is not easy to evaluate the bound in Proposition 2.11. However, we can further lower bound the capacity by introducing some conditioning.

Lemma 2.12. *The sequence of lower bounds* $\{D_i^{\mathscr{X}}\}_{i=1}^{\infty}$*, where*

$$D_i^{\mathscr{X}} \triangleq \left(\mathscr{H}(\mathscr{X}) + \frac{H(Z_1|Z_i, \mathscr{X}, \mathscr{Y})}{1-p_{\mathsf{r}}}\right)(1-p_{\mathsf{d}}) - \frac{1-p_{\mathsf{d}}}{1-p_{\mathsf{r}}} h_2(p_{\mathsf{r}}) - h_2(p_{\mathsf{d}}),\ i \in \mathbb{N}$$

is non-decreasing.

Proof. Since we have introduced extra conditioning, the $D_i^{\mathscr{X}}$s are lower bounds. We have

$$\begin{aligned}
H(Z_1|Z_{i+1}) &= H(Z_1, Z_i|Z_{i+1}) - H(Z_i|Z_1, Z_{i+1}) \\
&= H(Z_i|Z_{i+1}) + H(Z_1|Z_{[i:i+1]}) - H(Z_i|Z_1, Z_{i+1}) \\
&\overset{(a)}{=} H(Z_i|Z_{i+1}) + H(Z_1|Z_i) - H(Z_i|Z_1, Z_{i+1}) \\
&= H(Z_1|Z_i) + I(Z_1; Z_i|Z_{i+1}) \\
&\geq H(Z_1|Z_i)
\end{aligned}$$

where (a) follows from the Markovity of the $\mathscr{Z}$ process. Since conditioning on $\mathscr{X}$ and $\mathscr{Y}$ preserves the above chain of inequalities, we have

$$H(Z_1|Z_{i+1}, \mathscr{X}, \mathscr{Y}) \geq H(Z_1|Z_i, \mathscr{X}, \mathscr{Y})\ \forall\ i \geq 1.$$

Hence $\{D_i^{\mathscr{X}}\}_{i=1}^{\infty}$ is non-decreasing.

Optimizing $D_1^{\mathscr{X}}$ over stationary, ergodic, Markov input processes $\mathscr{X}$ gives the bound in Proposition 2.8. Therefore, for increasing i, we have bounds better than the one in Proposition 2.8. In particular, as $i \to \infty$, following the proof of Lemma 2.3 (iii),

we can see that $\frac{Z_i}{i} \to \frac{p_\mathsf{r}-p_\mathsf{d}}{1-p_\mathsf{d}}$ a.s., so that the knowledge of Z_i becomes tautological in the limit, and consequently,

$$\sup_{\mathscr{X}} \lim_{i\to\infty} D_i^{\mathscr{X}} = \sup_{\mathscr{X}} \sup_{i\geq 1} D_i^{\mathscr{X}}$$

gives us the bound in Proposition 2.11. □

Alternatively, instead of bounding the information rate as in Proposition 2.11, we can write the following as an immediate consequence of Equation (2.10) and an argument similar to the one made in the proof of Proposition 2.11.

Proposition 2.13 (Information rates for the DRC)**.** *For the DRC,*

$$\begin{aligned} C &= \sup_{\mathscr{X}} \Big((1-p_\mathsf{d})\mathscr{H}(\mathscr{X}) - \hat{\mathscr{H}}(\mathscr{Z}|\mathscr{Y}) + \hat{\mathscr{H}}(\mathscr{Z}|\mathscr{X},\mathscr{Y}) \Big) \\ &= (1-p_\mathsf{d}) \Big[\sup_{\mathscr{X}} \Big(\mathscr{H}(\mathscr{X}) + \frac{H(Z_1|\mathscr{X},\mathscr{Y}) - H(Z_1|\mathscr{Y})}{1-p_\mathsf{r}} \Big) \Big]. \end{aligned}$$ □

Following arguments similar to the ones used in Lemma 2.12, we can show the following.

Lemma 2.14. *The sequence of lower bounds* $\{R_i^{\mathscr{X}}\}_{i=1}^{\infty}$*, where*

$$R_i^{\mathscr{X}} \triangleq (1-p_\mathsf{d}) \Big(\mathscr{H}(\mathscr{X}) + \frac{H(Z_1|Z_i,\mathscr{X},\mathscr{Y}) - H(Z_1|\mathscr{Y})}{1-p_\mathsf{r}} \Big)$$

is non-decreasing, and

$$C = \sup_{\mathscr{X}} \lim_{i\to\infty} R_i^{\mathscr{X}} = \sup_{\mathscr{X}} \sup_{i\geq 1} R_i^{\mathscr{X}}.$$ □

The task of finding the rate-maximizing input distributions appears to be tough, with no theoretical insights[2] or efficient numerical algorithms. Often, to establish lower bounds on achievable rates, special classes of input processes are considered, and we will resort to a similar strategy here to obtain some expressions for the bounds we have so far developed. The following section will consider special cases of the DRC wherein there are either only deletions, i.e., the BDC, or only replications, i.e., the BRC. In a subsequent section, the DRC will be studied. The bounds developed in the next section are similar to the generic bounds developed thus far.

[2] A new result on BDC with small deletion probability [56] provides a partial answer to this question.

2.3 Channels with Deletions or Replications

For the case of the BDC or the BRC, evaluating some of the bounds developed in the previous section is somewhat easy, owing to the fact that the $\mathscr{Z}$ process is monotonic in these two special cases, i.e., it is non-increasing or non-decreasing with increments of at most one, respectively. This monotonicity in $\mathscr{Z}$ implies that the Γ process is strictly increasing for the BDC and non-decreasing with increments of at most one for the BRC. This translates to the output being a subsequence of the input sequence for the BDC and vice versa for the BRC.

2.3.1 Information Rates for the BDC

In this subsection we estimate the information rates possible over the BDC, i.e., $p_{\mathsf{d}} = p, p_{\mathsf{r}} = 0$, when the input process is either independent and uniformly distributed (i.u.d.) or when it is a first-order Markov process.

For the BDC with i.u.d. inputs, we can easily show that $\mathscr{Y}$ is also an i.u.d. sequence. Consequently,

$$\frac{I(Y_{[N_n]};Z_{[N_n]})}{n} \to 0 \text{ as } n \to \infty$$

because the only information obtained from $Y_{[N_n]}$ about $Z_{[N_n]}$ is the length of the vector, and this information vanishes in the limit as $n \to \infty$. Therefore, we have from Equation (2.10) that the lower bound in Proposition 2.11 for i.u.d. inputs is actually the *symmetric information rate* (SIR). We are hence interested in evaluating D_i^{iud} as defined in Lemma 2.12. In particular, we have the SIR

$$C_{\text{BDC}}^{\text{iud}} = \lim_{i\to\infty} D_i^{\text{iud}} = \sup_{i\geq 1} D_i^{\text{iud}}. \tag{2.11}$$

We start with some definitions and notation.

Definition 2.3 (Subsequence weights)**.** *We call a vector x_A a* subsequence *of a vector x_B if $A \subset B$ and the order of the elements in A is the same as the order in which those elements appear in B. For ease of notation, we will write $w_{y_{[i]}}(x_{[j]})$ to denote the number of subsequences of $x_{[j]} \in \mathbb{X}^j$ that are the same as $y_{[i]} \in \mathbb{X}^i$, which is referred to as the*

$$\mathfrak{H}_m^{(i)} = \sum_{x_{[m+i-1]} \in \mathbb{X}^{m+i-1}} \frac{1}{2^{m+i-1}} \mathsf{H}(x_{[m+i-1]}),$$

$$\mathsf{H}(x_{[m+i-1]}) = \sum_{y_{[i-1]} \in \mathbb{Y}^{i-1}} \frac{w_{y_{[i-1]}}(x_{[m+i-1]})}{\binom{m+i-1}{m}} \mathfrak{h}(x_{[m+i-1]}, y_{[i-1]}), \tag{2.12}$$

$$\mathfrak{h}(x_{[m+i-1]}, y_{[i-1]}) = -\Big[\sum_{z=-m}^{0} \mathbb{1}_{\{x_{1-z}=y_1\}} \frac{w_{y_{[2:i-1]}}(x_{[2-z:m+i-1]})}{w_{y_{[i-1]}}(x_{[m+i-1]})} \times \log_2 \Big(\mathbb{1}_{\{x_{1-z}=y_1\}} \frac{w_{y_{[2:i-1]}}(x_{[2-z:m+i-1]})}{w_{y_{[i-1]}}(x_{[m+i-1]})} \Big) \Big].$$

$y_{[i]}$-subsequence weight *of the vector* $x_{[j]}$. *We can write*

$$w_{y_{[i]}}(x_{[j]}) = \sum_{S \subset [j]:|S|=i} \mathbb{1}_{\{x_S = y_{[i]}\}}$$

where the elements of the set S are arranged in ascending order. Clearly, $w_{y_{[i]}}(x_{[j]}) = 0$ *for* $i > j$. *We define* $w_\lambda(x_{[j]}) = 1 \; \forall \, x_{[j]} \in \mathbb{X}^j$ *for* $j \geq 0$. □

Definition 2.4 (Runs and run-lengths)**.** *For a binary sequence, a* run *is a maximal block of contiguous* 0*s or* 1*s. The* run-length *of a run is the number of symbols in it. We denote by* $r_1(x_{[j]})$ *the length of the first run in the vector* $x_{[j]} \in \mathbb{X}^j, j \geq 1$. *Clearly,* $1 \leq r_1(x_{[j]}) \leq |x_{[j]}| = j$. □

We will denote by $\mathbb{Z}_\uparrow^i$ and $\mathbb{Z}_\downarrow^i$ the sets of non-decreasing and non-increasing vectors of length i, respectively, for $i \geq 1$.

Theorem 2.15 (SIR for the BDC)**.** *For the BDC,*

$$C_{\text{BDC}}^{\text{iud}} = 1 - p - h_2(p) + (1-p)\Big(\lim_{i \to \infty} \sum_{m \geq 0} \psi_{i,m} p^m (1-p)^i \Big),$$

where $\psi_{i,m} \triangleq \binom{m+i-1}{m} \mathfrak{H}_m^{(i)}$, *with* $\mathfrak{H}_m^{(i)} = H(Z_1 | Z_i = -m, \mathscr{X}, \mathscr{Y})$ *is as given in Equation* (2.12).

Proof. For the BDC, we have from Lemma 2.12 that

$$D_i^{\text{iud}} = 1 - p - h_2(p) + (1-p) H(Z_1 | Z_i, \mathscr{X}, \mathscr{Y}).$$

From Equation (2.11), we need to show that

$$H(Z_1|Z_i,\mathscr{X},\mathscr{Y}) = \sum_{m\geq 0} \psi_{i,m} p^m (1-p)^i.$$

We first note that $H(Z_1|Z_i,\mathscr{X},\mathscr{Y}) = H(Z_1|Z_i,X_{[i-Z_i-1]},Y_{[i-1]})$. Clearly, the above entropy term is zero for $i=1$. For $i\geq 2$, given $Z_i=-m, X_{[m+i-1]}=x_{[m+i-1]}$ and $Y_{[i-1]}=y_{[i-1]}$, it is easy to see that

$$Z_1 \in \{z\in\{0,-1,\cdots,-m\} : x_{1-z}=y_1, w_{y_{[2:i-1]}}(x_{[2-z:m+i-1]})>0\}.$$

That is, $Z_1=z$ only if x_{1-z} and y_1 match, and the subsequent part of the output vector $y_{[2:i-1]}$ is a subsequence of the subsequent part of the input vector $x_{[2-z:m+i-1]}$. Also, for $z_{[i-1]}\in\mathbb{Z}_{\downarrow}^{i-1}$ (which, as noted earlier, is true for the BDC),

$$\begin{aligned}\mathsf{P}(Z_{[i-1]}=z_{[i-1]},Z_i=-m|X_{[m+i-1]}=x_{[m+i-1]},&Y_{[i-1]}=y_{[i-1]})\\ &= \mathbb{1}_{\{x_{[i-1]-z_{[i-1]}}=y_{[i-1]}\}} p_{\mathsf{t}}^i p_{\mathsf{d}}^m,\end{aligned}$$

where $p_{\mathsf{d}}=1-p_{\mathsf{t}}=p$, so that for $0\geq z\geq -m$,

$$\begin{aligned}\mathsf{P}(Z_1=z,Z_i=-m|X_{[m+i-1]}=x_{[m+i-1]},&Y_{[i-1]}=y_{[i-1]})\\ &= \mathbb{1}_{\{x_{1-z}=y_1\}} p_{\mathsf{t}}^i p_{\mathsf{d}}^m \cdot w_{y_{[2:i-1]}}(x_{[2-z:m+i-1]}),\end{aligned}$$

and

$$\mathsf{P}(Z_i=-m|X_{[m+i-1]}=x_{[m+i-1]},Y_{[i-1]}=y_{[i-1]}) = w_{y_{[i-1]}}(x_{[m+i-1]}) p_{\mathsf{t}}^i p_{\mathsf{d}}^m.$$

Hence, when $w_{y_{[i-1]}}(x_{[m+i-1]})>0$,

$$\begin{aligned}\mathsf{P}(Z_1=z|Z_i=-m,X_{[m+i-1]}=x_{[m+i-1]},&Y_{[i-1]}=y_{[i-1]})\\ &= \frac{\mathbb{1}_{\{x_{1-z}=y_1\}} w_{y_{[2:i-1]}}(x_{[2-z_1:m+i-1]}) p_{\mathsf{t}}^i p_{\mathsf{d}}^m}{w_{y_{[i-1]}}(x_{[m+i-1]}) p_{\mathsf{t}}^i p_{\mathsf{d}}^m}\\ &= \frac{\mathbb{1}_{\{x_{1-z}=y_1\}} w_{y_{[2:i-1]}}(x_{[2-z_1:m+i-1]})}{w_{y_{[i-1]}}(x_{[m+i-1]})}.\end{aligned}$$

Since, with i.u.d. inputs, $\mathsf{P}(X_{[m+i-1]}=x_{[m+i-1]}|Z_i=-m)=2^{-(m+i-1)}$ and

$$\mathsf{P}(Y_{[i-1]}=y_{[i-1]}|X_{[m+i-1]}=x_{[m+i-1]},Z_i=-m) = \frac{w_{y_{[i-1]}}(x_{[m+i-1]})}{\binom{m+i-1}{m}},$$

we have that $H(Z_1|Z_i = -m, \mathscr{X}, \mathscr{Y}) = \mathfrak{H}_m^{(i)}$ as in Equation (2.12). By noting that

$$\mathsf{P}(Z_i = -m|Z_0 = 0) = \binom{m+i-1}{m} p^m (1-p)^i$$

from Equation (2.6) (with $p_{\mathsf{d}} = p, p_{\mathsf{r}} = 0, p_{\mathsf{t}} = 1-p$), we have the desired result. □

Although evaluating $\mathfrak{H}_m^{(i)}$ in general is hard since we are required to count subsequence weights of sequences, we can evaluate it in two specific cases: for every m when $i = 2$ (when all but a single bit are deleted) and for all i when $m = 1$ (when only a single bit is deleted).

Corollary 2.16 (Lower bound for $C_{\mathrm{BDC}}^{\mathrm{iud}}$)**.** *For the BDC,*

$$C_{\mathrm{BDC}}^{\mathrm{iud}} \geq D_2^{\mathrm{iud}} \geq \frac{4(1-p)^3}{(2-p)^2} - h_2(p) + (1-p)^3 \Big(\sum_{m \geq 2} m p^{m-1} \log_2 m \Big).$$

Proof. It is easy to see that when $i = 2$, Equation (2.12) reduces to

$$\begin{aligned} \mathfrak{H}_m^{(2)} &= H(Z_1|Z_2 = -m, \mathscr{X}, \mathscr{Y}) \\ &= \log_2(m+1) - \frac{1}{2^{m+1}} \sum_{x_{[m+1]}} h_2\Big(\frac{w(x_{[m+1]})}{m+1}\Big) \\ &= \log_2(m+1) - \frac{1}{2^{m+1}} \sum_{j=0}^{m+1} \binom{m+1}{j} h_2\Big(\frac{j}{m+1}\Big), \end{aligned} \tag{2.13}$$

where $w(\cdot)$ denotes *Hamming weight*. Hence

$$D_2^{\mathrm{iud}} = 1 - p - h_2(p) + (1-p)^3 \sum_{m \geq 0} (m+1) \mathfrak{H}_m^{(2)}. \tag{2.14}$$

For numerically estimating $\mathfrak{H}_m^{(2)}$ for large m, we can use the upper bound [81]

$$\binom{m+1}{j} \leq 2^{(m+1)h_2(\frac{j}{m+1})} \sqrt{\frac{m+1}{2\pi j(m+1-j)}}$$

to get a further lower bound[3] on $\mathfrak{H}_m^{(2)}$. On the other hand, to obtain a looser analytic lower bound, we can bound

$$\frac{1}{2^{m+1}} \sum_{j=0}^{m+1} \binom{m+1}{j} h_2\Big(\frac{j}{m+1}\Big) \leq 1 - 2^{-m},$$

[3]We would like to get a lower bound on D_2^{iud} since this will be a lower bound for C^{iud} as well (cf. Equation (2.11)).

to get $\mathfrak{H}_m^{(2)} \geq \log_2(m+1) - 1 + 2^{-m}$. This gives us

$$D_2^{\text{iud}} \geq (1-p)^3 \Big(\frac{4}{(2-p)^2} + \sum_{m \geq 0} (m+1) p^m \log_2(m+1) \Big) - h_2(p).$$

This proves the claim. Unfortunately, it is not easy to evaluate the series

$$\kappa = \sum_{m=2}^{\infty} m p^{m-1} \log_2 m$$

on the right hand side of the above inequality. Consider the function

$$f(x) = x p^{x-1} \ln x,$$

where $\ln(\cdot)$ is the natural logarithm. The m^{th} term in the series can then be written as $(\log_2 e) f(m), m \geq 2$. It turns out that for

$$p < p^* \triangleq \exp\Big(-\frac{1+\ln 2}{2 \ln 2} \Big) \approx 0.294832606,$$

we can lower bound the series κ by the integral

$$\begin{aligned} \kappa &\geq \log_2 e \int_2^{\infty} f(x) dx = \log_2 e \int_2^{\infty} x p^{x-1} \ln x dx \\ &= \frac{\log_2 e}{\ln p} \Big(\frac{p}{\ln p} (1 + \ln 2) - 2p \ln 2 - \frac{1}{p} \text{Ei}(2 \ln p) \Big), \end{aligned}$$

where $\text{Ei}(x)$ is the *exponential integral* function defined as

$$\text{Ei}(x) = \int_{-\infty}^{x} \frac{e^t}{t} dt,$$

which can be numerically evaluated to arbitrary accuracy through a Taylor series expansion. Therefore, for $p < p^*$,

$$D_2^{\text{iud}} \geq \frac{4(1-p)^3}{(2-p)^2} - h_2(p) + (1-p)^3 \frac{\log_2 e}{\ln p} \Big(\frac{p}{\ln p} (1 + \ln 2) - 2p \ln 2 - \frac{1}{p} \text{Ei}(2 \ln p) \Big).$$

With $\mathfrak{H}_m^{(2)}$ as given in Equation (2.13), we can write

$$D_2^{\text{iud}} = 1 + p \log_2 p - p \log_2(2e) + O(p^2)$$

for small p. This is loose compared to the bound obtained in [55]. This can be attributed to the fact that we evaluated $H(Z_1 | Z_2, \mathscr{X}, \mathscr{Y})$ rather than $H(Z_1 | \mathscr{X}, \mathscr{Y})$ to obtain D_2^{iud}. In fact, this small-p series expansion of D_2^{iud} is no better than that of the lower bound for the BDC in Corollary 2.9. We will improve this bound for small p in the next subsection. □

Corollary 2.17 (Small deletion probability SIR)**.** *For the BDC,*

$$C_{\mathrm{BDC}}^{\mathrm{iud}} = 1 + p\log_2 p - \mathtt{d}p + O(p^2)$$

where $\mathtt{d} \approx 1.154163765$.

Proof. We can prove the claim by pursuing the other case where (2.12) is easy to evaluate. Instead of evaluating D_i^{iud} exactly, we can further lower bound it as follows.

$$\begin{aligned}
D_i^{\mathrm{iud}} &= 1 - p - h_2(p) + (1-p)H(Z_1|Z_i, \mathscr{X}, \mathscr{Y}) \\
&= 1 - p - h_2(p) + (1-p)\Big(\sum_{m\geq 0} \mathsf{P}(Z_i = -m)H(Z_1|Z_i = -m, \mathscr{X}, \mathscr{Y})\Big) \\
&= 1 - p - h_2(p) + (1-p)\Big(\sum_{m=0}^{j} \mathsf{P}(Z_i = -m)H(Z_1|Z_i = -m, \mathscr{X}, \mathscr{Y})\Big) \\
&\triangleq 1 - p - h_2(p) + (1-p)\Psi_j^{(i)} \\
&\triangleq \mathfrak{D}_j^{(i)} \ \forall\ j \geq 0, i \geq 1.
\end{aligned}$$

We are essentially writing a series expansion for D_i^{iud} and lower bounding[4] it by the j^{th} partial sum. Note that we can write

$$\begin{aligned}
\Psi_j^{(i)} &= \Psi_{j-1}^{(i)} + \mathsf{P}(Z_i = -j)H(Z_1|Z_i = -j, \mathscr{X}, \mathscr{Y}) \\
&= \Psi_{j-1}^{(i)} + \psi_{i,j}p^j(1-p)^i
\end{aligned} \tag{2.15}$$

where $\psi_{i,m}$ was defined in Theorem 2.15. Clearly, the sequence $\{\Psi_j^{(i)}\}_{j\geq 0}$ is non-decreasing, and, in turn, so is the sequence $\{\mathfrak{D}_j^{(i)}\}_{j\geq 0}$. Since $\Psi_0^{(i)} = \psi_{i,0} = 0$, we have $\Psi_1^{(i)} = p(1-p)^i\psi_{i,1}$. Further, by definition,

$$D_i^{\mathrm{iud}} = \lim_{j\to\infty} \mathfrak{D}_j^{(i)} = \sup_{j\geq 0} \mathfrak{D}_j^{(i)}.$$

Thus for every $j \geq 0$, we can write

$$C_{\mathrm{BDC}}^{\mathrm{iud}} = \sup_{i\geq 1} D_i^{\mathrm{iud}} = \sup_{i\geq 1}\sup_{j\geq 0} \mathfrak{D}_j^{(i)} \overset{(a)}{=} \sup_{j\geq 0}\sup_{i\geq 1} \mathfrak{D}_j^{(i)} \geq \sup_{i\geq 1} \mathfrak{D}_j^{(i)} \triangleq \mathfrak{D}_j^{\mathrm{iud}},$$

where (a) is true since $\mathfrak{D}_j^{(i)} \in [0,1]\ \forall\ i \geq 1, j \geq 0$.

[4] All terms in the series expansion are non-negative.

From the channel model, $\mathsf{P}(X_{[i]} = x_{[i]}|Z_i = -1) = 2^{-i}$ since $\mathscr{X} \perp\!\!\!\perp \mathscr{Z}$ and $\mathscr{X}$ is i.u.d., and

$$\mathsf{P}(Y_{[i-1]} = y_{[i-1]}|X_{[i]} = x_{[i]}, Z_i = -1) = \frac{w_{y_{[i-1]}}(x_{[i]})}{i}.$$

For $y_{[i-1]} = x_{[i-1]-z_{[i-1]}}$ for some realization $z_{[i-1]} \in \mathbb{Z}_{\downarrow}^{i-1}$ with the boundary conditions $z_0 = 0$ and $z_i = -1$,

$$H(Z_1|Z_i = -1, X_{[i]} = x_{[i]}, Y_{[i-1]} = y_{[i-1]}) = h_2\Big(\frac{1}{r_1(x_{[i]})}\Big)\mathbb{1}_{\mathscr{R}_1(x_{[i]}, y_{[i-1]})}$$

where $\mathscr{R}_1(x_{[i]}, y_{[i-1]})$ is the event that the single deletion occurred in the first run of $x_{[i]}$ to result in $y_{[i-1]}$. To see this, let $y_{[i-1]}$ represent a received word resulting from a single deletion upon transmission of $x_{[i]}$. Consider the two mutually exclusive and exhaustive cases in this scenario:

- The single deletion occurs in a run other than the first run of $x_{[i]}$. In this case, there is no ambiguity that $Z_1 = 0$, and the first run of $y_{[i-1]}$ is either the same or larger than[5] that of $x_{[i]}$.
- The single deletion occurs in the first run of $x_{[i]}$.
  - If $r_1(x_{[i]}) = 1$, there is no ambiguity that $Z_1 = -1$.
  - If $r_1(x_{[i]}) > 1$, the deleted symbol could be, with equal likelihood, one of the symbols comprising the first run of $x_{[i]}$. The uncertainty in Z_1 is $h_2\Big(\frac{1}{r_1(x_{[i]})}\Big)$.

  In both the above sub-cases, the uncertainty can be written as $h_2\Big(\frac{1}{r_1(x_{[i]})}\Big)$.

Therefore,

$$\begin{aligned}
\psi_{i,1} &= i\sum_{x_{[i]}}\frac{1}{2^i}\sum_{y_{[i-1]}}\frac{w_{y_{[i-1]}}(x_{[i]})}{i}h_2\Big(\frac{1}{r_1(x_{[i]})}\Big)\mathbb{1}_{\mathscr{R}_1(x_{[i]}, y_{[i-1]})} \\
&= i\sum_{x_{[i]}}\frac{1}{2^i}\frac{r_1(x_{[i]})}{i}h_2\Big(\frac{1}{r_1(x_{[i]})}\Big) = \frac{1}{2^i}\sum_{j=1}^{i} j2^{i-j}h_2\Big(\frac{1}{j}\Big) + \frac{1}{2^i}h_2\Big(\frac{1}{i}\Big) \\
&= \sum_{j=1}^{i}\frac{j}{2^j}h_2\Big(\frac{1}{j}\Big) + \frac{1}{2^i}h_2\Big(\frac{1}{i}\Big) \\
&= \frac{1}{2}\sum_{j=1}^{i-2}\frac{j}{2^j}\log_2 j + 2\frac{i}{2^i}\log_2 i. \qquad (2.16)
\end{aligned}$$

[5] When the second run of $x_{[i]}$ disappears.

We observe that $\psi_{i,1}$ is non-decreasing in i, and converges exponentially to the value $\psi_1 \approx 1.288531275$. From (2.15) and (2.16), we have

$$\begin{aligned}\mathfrak{D}_1^{(i)} &= 1 - p - h_2(p) + p(1-p)^{i+1}\psi_{i,1} \\ &= 1 + p\log_2 p - p\log_2(2e) + \psi_{i,1}p + O(p^2).\end{aligned}$$

Since $D_i^{\text{iud}} = \mathfrak{D}_1^{(i)} + \sum_{j\geq 2} p^j(1-p)^{i+1}\psi_{i,j}$, we have

$$D_i^{\text{iud}} = 1 + p\log_2 p - p\log_2(2e) + \psi_{i,1}p + O(p^2).$$

Thus, from Equation (2.11)

$$C_{\text{BDC}}^{\text{iud}} = 1 + p\log_2 p - \mathtt{d}p + O(p^2) \tag{2.17}$$

where $\mathtt{d} = \log_2(2e) - \psi_1 \approx 1.154163765$. We note here that this is exactly the same bound obtained in [55] with a completely different technique. Since this bound was shown to be tight for small p, we have that the capacity of the BDC itself is given by the above expression for small p. □

Discussion : The advantage in the evaluation of the above bound was that, when we restrict to the case of a single deletion, the ambiguity in the first channel state Z_1 arises only when $r_1(x_{[i]}) > 1$, in which case the uncertainty is exactly $h_2\Big(\frac{1}{r_1(x_{[i]})}\Big)$. This, however, is not true when there are 2 or more deletions, wherein we will have to count subsequence weights of sequences.

Similar bounds for symmetric first-order Markov input processes can be obtained as follows. Since the channel has no bias for the input symbols, we can confine our attention to *symmetric* Markov inputs. Proceeding along the same lines as above, we can write for $\mathsf{P}(X_i = x \oplus 1 | X_{i-1} = x) = \alpha \in [0,1]$,

$$D_2^{\mathscr{M}1} = \Big[\max_{\alpha}\Big(h_2(\alpha) + (1-p)^2 \sum_{m\geq 0}(m+1)p^m \ell_m(\alpha)\Big)\Big](1-p) - h_2(p),$$

where

$$\ell_m(\alpha) = \log_2(m+1) - \sum_{j=0}^{m+1} h_2\Big(\frac{j}{m+1}\Big)\eta(\alpha, j, m+1),$$

and $\eta(\cdot)$ is defined recursively as

$$\begin{aligned}\eta(\alpha,j,m) &= \eta_0(\alpha,j,m)+\eta_1(\alpha,j,m)\\ \eta_0(\alpha,j,m) &= (1-\alpha)\eta_0(\alpha,j,m-1)+\alpha\eta_1(\alpha,j,m-1)\\ \eta_1(\alpha,j,m) &= (1-\alpha)\eta_1(\alpha,j-1,m-1)+\alpha\eta_0(\alpha,j-1,m-1)\end{aligned}$$

with $\eta_k(\alpha,km,m)=\frac{1}{2}(1-\alpha)^{m-1}$, $\eta_k(\alpha,(1-k)m,m)=0$ and $\eta_k(\alpha,j,m)=0\ \forall\ j\notin[m]$ for $k\in\{0,1\}$.

Similarly, we can also evaluate

$$\begin{aligned}&\mathfrak{D}_1^{\mathscr{M}1} = -h_2(p)+(1-p)\times\\ &\max_{\alpha}\Big[h_2(\alpha)+p\cdot\sup_{i\geq 1}(1-p)^i\Big(\alpha\sum_{j=1}^{i}j(1-\alpha)^{j-1}h_2\Big(\frac{1}{j}\Big)+i(1-\alpha)^i h_2\Big(\frac{1}{i}\Big)\Big)\Big].\end{aligned}$$

However, both $D_2^{\mathscr{M}1}$ and $\mathfrak{D}_1^{\mathscr{M}1}$ turn out to be better than their SIR counterparts by less than 2%.

Discussion : Although first-order Markov inputs are expected to perform better than i.u.d. inputs, we see that the bounds we obtained are almost the same in the two cases. This is because we are considering two special cases, the first when $i=2$ wherein all but a single symbol were deleted, and the second when $m=1$ wherein a single symbol was deleted; and in these cases, a first-order Markov input is not significantly different than i.u.d. inputs. □

Figure 2.1 plots the bounds on the capacity for C_{BDC}.

2.3.2 Information Rates for the BRC

In this subsection, we will consider information rates for the BRC, i.e., $p_{\mathsf{d}}=0, p_{\mathsf{r}}=p$. As in the previous subsection, we will consider i.u.d. and symmetric first-order Markov inputs.

For the BRC, the $\mathscr{Z}$ process is non-decreasing. Moreover, when it increases, the increment is at most 1 at each time instant. This simplifies the evaluation of information

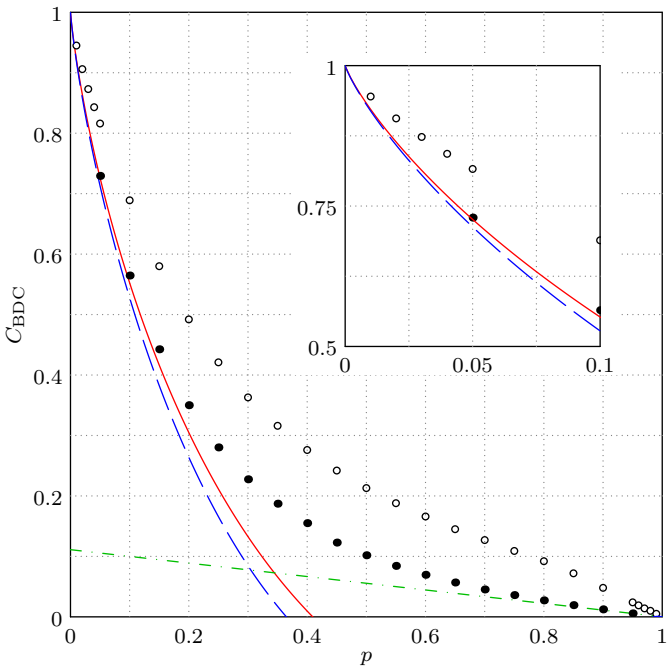

Figure 2.1: Bounds on the capacity of the BDC in bits per channel use as a function of the deletion probability p. D_2^{iud} (cf. Equation (2.14)) is shown as the long-dashed blue line and C^{iud} (or equivalently $\mathfrak{D}_1^{\mathrm{iud}}$, cf. Corollary 2.17) with the $O(p^2)$ term dropped as the solid red line (cf. Equation (2.17)). The best known numerical lower [58] and upper bounds [35] are shown as black and white circles respectively. The best known lower bound as p approaches 1 [86] is shown as the dash-dotted green line. The inset shows the bounds for small p values where the red solid curve is known to be tight from [55].

rates and we will, in fact, be able to write exact expressions for the Markov-1 rates, as will be shown shortly. In this case, even when the input is i.u.d., the term

$$\frac{I(Y_{[N_n]};Z_{[N_n]})}{n} \nrightarrow 0 \text{ as } n \to \infty$$

in the normalized version of Equation (2.10). Hence the expression for the information rate in Proposition 2.13 will prove to be more useful in this subsection.

Theorem 2.18 (Markov-1 Rates for the BRC)**.** *For the BRC, the Markov-*1 *rate is given as*

$$C_{\mathrm{BRC}}^{\mathscr{M}1} = \max_{\alpha} \Big[h_2(\alpha) + \alpha \sum_{l\geq 1} \Big((1-\alpha)\frac{1-p}{p} \Big)^l \Big(\sum_{k\geq l} \binom{k}{l} p^k h_2(\tfrac{l}{k}) \Big) - \frac{p+(1-\alpha)(1-p)}{1-p} h_2\Big(\frac{p}{p+(1-\alpha)(1-p)} \Big) \Big]. \tag{2.18}$$

Proof. First we note that since $Z_1 \in \{0,1\}$, we have

$$H(Z_1|\mathscr{X},\mathscr{Y}) = \mathsf{E}[h_2(\mathsf{P}(Z_1 = 0 \mid x_{\mathbb{N}},\, y_{\mathbb{N}}))]$$

and $H(Z_1|\mathscr{Y}) = \mathsf{E}[h_2(\mathsf{P}(Z_1 = 0|y_{\mathbb{N}}))]$. Further we have for the BRC that whenever $Y_i \neq Y_{i-1}$, we must have $Z_i = Z_{i-1}$ or equivalently $\Gamma_i = \Gamma_{i-1} + 1$. This means that Z_1 is independent of subsequent runs of $\mathscr{Y}$ (and $\mathscr{X}$) given the first run of $\mathscr{Y}$ (and $\mathscr{X}$) since we can achieve synchronization at the end of each run. Thus we can write the conditional probabilities $\mathsf{P}(Z_1 = 0 \mid \mathscr{X},\, \mathscr{Y})$ and $\mathsf{P}(Z_1 = 0 \mid \mathscr{Y})$ in terms of the first runs of $\mathscr{X}$ and $\mathscr{Y}$, i.e., $\mathsf{P}(Z_1 = 0 \mid x_{\mathbb{N}},\, y_{\mathbb{N}}) = \mathsf{P}(Z_1 = 0 \mid r_1(x_{\mathbb{N}}),\, r_1(y_{\mathbb{N}}))$ and $\mathsf{P}(Z_1 = 0 \mid y_{\mathbb{N}}) = \mathsf{P}(Z_1 = 0 \mid r_1(y_{\mathbb{N}}))$. Note that we assume that $Z_0 = 0$ so that $Y_0 = X_0$. Thus, if $x_1 \neq x_0$, then Z_1 is 0 or 1 accordingly as y_1 is not equal or equal to y_0, respectively. This means that there is no uncertainty in Z_1 given the output sequence (and the assumption that $x_0 = y_0 = 0$, which can be made without loss of generality). Therefore, in estimating the entropy of Z_1 given the output sequence, or the output and the input sequences, we can confine our attention to those sequences $x_{\mathbb{N}}$ and $y_{\mathbb{N}}$ whose first runs are comprised of zeros. We shall denote such runs as $r_1^0(\cdot)$. For a first-order Markov input process, we have, for $l \geq 0$

$$\mathsf{P}(r_1^0(x_{\mathbb{N}}) = l) = (1-\alpha)^l \alpha,$$

and we can get from the definition of the BRC that

$$\mathsf{P}(r_1^0(y_{\mathbb{N}}) = k | r_1^0(x_{\mathbb{N}}) = l) = \binom{k}{l}(1-p)^{l+1}p^{k-l}$$

for $k \geq l$. Consequently, we have

$$\begin{aligned}\mathsf{P}(r_1^0(y_{\mathbb{N}}) = k) &= \sum_{l=0}^{k}(1-\alpha)^l\alpha \cdot \binom{k}{l}(1-p)^{l+1}p^{k-l} \\ &= \alpha(1-p)\Big(p + (1-\alpha)(1-p)\Big)^k.\end{aligned}$$

Since $Z_1 = 0$ excludes the first bit in the received sequence from being a replication, we can easily obtain

$$\mathsf{P}(Z_1 = 0 | r_1^0(x_{\mathbb{N}}) = l, r_1^0(y_{\mathbb{N}}) = k) = \frac{l}{k}$$

for $k \geq l + \mathbb{1}_{\{l=0\}}$. For $k \geq 1$,

$$\begin{aligned}
\mathsf{P}(Z_1 = 0 | r_1^0(y_{\mathbb{N}}) = k) &= \frac{\sum_{l=0}^{k} \mathsf{P}(Z_1 = 0, r_1^0(x_{\mathbb{N}}) = l, r_1^0(y_{\mathbb{N}}) = k)}{\mathsf{P}(r_1^0(y_{\mathbb{N}}) = k)} \\
&= \frac{\sum_{l=0}^{k} (1-\alpha)^l \alpha \binom{k}{l} (1-p)^{l+1} p^{k-l} (\frac{l}{k})}{(1-p)\alpha\Big(p + (1-\alpha)(1-p)\Big)^k} \\
&= \frac{(1-\alpha)(1-p)}{p + (1-\alpha)(1-p)}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
H(Z_1 | \mathscr{X}, \mathscr{Y}) &= \sum_{l \geq 0} (1-\alpha)^l \alpha \Big(\sum_{k \geq l + \mathbb{1}_{\{l=0\}}} \binom{k}{l} (1-p)^{l+1} p^{k-l} h_2(\frac{l}{k}) \Big) \\
&= \alpha(1-p) \sum_{l \geq 1} \Big((1-\alpha) \frac{1-p}{p} \Big)^l \Big(\sum_{k \geq l} \binom{k}{l} p^k h_2(\frac{l}{k}) \Big)
\end{aligned}$$

and

$$\begin{aligned}
H(Z_1 | \mathscr{Y}) &= \sum_{k \geq 1} \alpha(1-p) \Big(p + (1-\alpha)(1-p) \Big)^k h_2\Big(\frac{(1-\alpha)(1-p)}{p + (1-\alpha)(1-p)} \Big) \\
&= (p + (1-\alpha)(1-p)) h_2\Big(\frac{p}{p + (1-\alpha)(1-p)} \Big).
\end{aligned}$$

Substituting these in Proposition 2.13 specialized to the BRC and first-order Markov inputs, we have the desired result. □

Corollary 2.19 (Lower bound for $C_{\mathrm{BRC}}^{\mathscr{M}1}$)**.** *For the BRC,*

$$\begin{aligned}
C_{\mathrm{BRC}}^{\mathscr{M}1} \geq R_2^{\mathscr{M}1} = h_2\Big(\frac{1}{(1-p)(4^p+1)} \Big) + \Big(\frac{2p}{1-p} \Big) \Big(\frac{(1-p)4^p - p}{4^p + 1} \Big) \\
- \Big(\frac{1}{1-p} \Big) \Big(\frac{4^p}{4^p+1} \Big) h_2\Big(\frac{p(4^p+1)}{4^p} \Big)
\end{aligned}$$

for $0 \leq p \leq p_* \approx 0.734675821$.

Proof. We have from Proposition 2.13 and Lemma 2.14 that

$$\begin{aligned}
C_{\mathrm{BRC}}^{\mathscr{M}1} \geq R_2^{\mathscr{M}1} \triangleq \max_{\alpha} \Big[h_2(\alpha) + \frac{H(Z_1 | Z_2, \mathscr{X}, \mathscr{Y})}{1-p} \\
- \frac{p + (1-\alpha)(1-p)}{1-p} h_2\Big(\frac{p}{p + (1-\alpha)(1-p)} \Big) \Big]
\end{aligned}$$

where we have used the expression for $H(Z_1|\mathscr{Y})$ from the proof of Theorem 2.18. Observe that $Z_2 \in \{0,1,2\}$, and among these possibilities, the only event wherein there is an ambiguity in the value of Z_1 is when $Z_2 = 1$. Thus, we can see easily that $H(Z_1|Z_2,\mathscr{X},\mathscr{Y}) = 2p(1-p)(1-\alpha)$. Hence

$$R_2^{\mathscr{M}1} = \max_{\alpha}\Big[h_2(\alpha) + 2p(1-\alpha) - \frac{p+(1-\alpha)(1-p)}{1-p}h_2\Big(\frac{p}{p+(1-\alpha)(1-p)}\Big)\Big].$$

It can be shown that the optimal α in the above is given by

$$\alpha^* = \frac{1}{(1-p)(2^{2p}+1)}.$$

Note that α^* is always larger than $\frac{1}{2}$, and $\alpha^* \le 1$ for $p \le p_*$ where $p_* \approx 0.734675821$. Plugging this back in the expression for $R_2^{\mathscr{M}1}$ ends the proof. □

Corollary 2.20 (Small replication probability SIR)**.** *For the BRC,*

$$C_{\mathrm{BRC}}^{\mathrm{iud}} = 1 + p\log_2 p + \mathtt{r}p + O(p^2)$$

where $\mathtt{r} \approx 0.845836235$.

Proof. From Proposition 2.13 and Lemma 2.14, for i.u.d. inputs,

$$\begin{aligned}
C_{\mathrm{BRC}}^{\mathrm{iud}} &= 1 - \frac{H(Z_1|\mathscr{Y})}{1-p} + \sup_{i\ge 1}\frac{H(Z_1|Z_i,\mathscr{X},\mathscr{Y})}{1-p} \\
&= 1 - \frac{1+p}{2(1-p)}h_2\Big(\frac{2p}{1+p}\Big) + \sup_{i\ge 1}\frac{\Big(\sum_{m=0}^{i}\mathsf{P}(Z_i=m)H(Z_1|Z_i=m,\mathscr{X},\mathscr{Y})\Big)}{1-p} \\
&= 1 - \frac{1+p}{2(1-p)}h_2\Big(\frac{2p}{1+p}\Big) + \sup_{i\ge 1}\Big(\sum_{m=0}^{i}\binom{i}{m}p^m(1-p)^{i-m-1} \\
&\qquad\qquad \times H(Z_1|Z_i=m,\mathscr{X},\mathscr{Y})\Big) \\
&= 1 - \frac{1+p}{2(1-p)}h_2\Big(\frac{2p}{1+p}\Big) + \sup_{i\ge 1}\Big(ip(1-p)^{i-2}H(Z_1|Z_i=1,\mathscr{X},\mathscr{Y})\Big) + O(p^2),
\end{aligned}$$

where we have used the expression for $H(Z_1|\mathscr{Y})$ from the proof of Theorem 2.18 for $\alpha = \frac{1}{2}$. The last equality is true since $H(Z_1|Z_i=0,\mathscr{X},\mathscr{Y}) = 0$.

As shown in the proof of Theorem 2.18, we can write

$$H(Z_1|Z_i=1,\mathscr{X},\mathscr{Y}) = \mathsf{E}[h_2(\mathsf{P}(Z_1=0|Z_i=1,r_1^0(x_{\mathbb{N}}),r_1^0(y_{\mathbb{N}})))].$$

Further, there is no ambiguity in Z_1 if the single replication does not occur in the first run of $x_{\mathbb{N}}$. Therefore, for a first-order Markov input process,

$$\begin{aligned} H(Z_1|Z_i=1,\mathscr{X},\mathscr{Y}) &= \mathsf{E}[h_2(\mathsf{P}(Z_1=0|Z_i=1,r_1^0(x_{\mathbb{N}})=l,r_1^0(y_{\mathbb{N}})=l+1))] \\ &= \sum_{l=1}^{i-1}(1-\alpha)^l\alpha\frac{l+1}{i}h_2\Big(\frac{1}{l+1}\Big)+(1-\alpha)^i h_2\Big(\frac{1}{i}\Big). \end{aligned}$$

For $\alpha=\frac{1}{2}$, we get

$$iH(Z_1|Z_i=1,\mathscr{X},\mathscr{Y}) = \frac{1}{2}\sum_{l=1}^{i-2}\frac{l}{2^l}\log_2 l + 2\frac{i}{2^i}\log_2 i = \psi_{i,1}$$

from Equation (2.16). Thus,

$$\begin{aligned} C_{\mathrm{BRC}}^{\mathrm{iud}} &= 1-\frac{1+p}{2(1-p)}h_2\Big(\frac{2p}{1+p}\Big)+\sup_{i\geq 1}\Big(p(1-p)^{i-2}\psi_{i,1}\Big)+O(p^2), \\ &= 1+p\log_2 p+\log_2\Big(\frac{2}{e}\Big)p+\sup_{i\geq 1}\Big(p(1-p)^{i-2}\psi_{i,1}\Big)+O(p^2), \\ &= 1+p\log_2 p+\mathtt{r}p+O(p^2), \end{aligned}$$

where $\mathtt{r}=\log_2(\frac{2}{e})+\psi_1 = 2-\mathtt{d}\approx 0.845836235$. As was the case for the BDC, we expect this to be a tight bound for the capacity for small p. □

Figure 2.2 plots these bounds. Note that the SIR and the Markov-1 rate are non-convex in p. Further, it appears that the Markov-1 rate (and the SIR) are zero for some values of $p<1$. However, this behavior is due to the fact that the term

$$\sum_{k\geq l}\binom{k}{l}p^k h_2\Big(\frac{l}{k}\Big)$$

in Equation (2.18) is computed only up to a finite value of k (the curves in Figure 2.2 are, therefore, lower bounds for the Markov-1 rate and the SIR). For values of p close to 1, more terms in this sum need to be considered to get a better estimate of the achievable rates.

Remark 2.2. It was expected that the capacity of a memoryless SEC was a convex function of the channel parameters. Although this conjecture seems to be true for the BDC [20], we see that this conjecture is false for the BRC. Note that the lower bounds in [84] themselves lead one to question the conjecture (cf. Figure 2.2). However, the

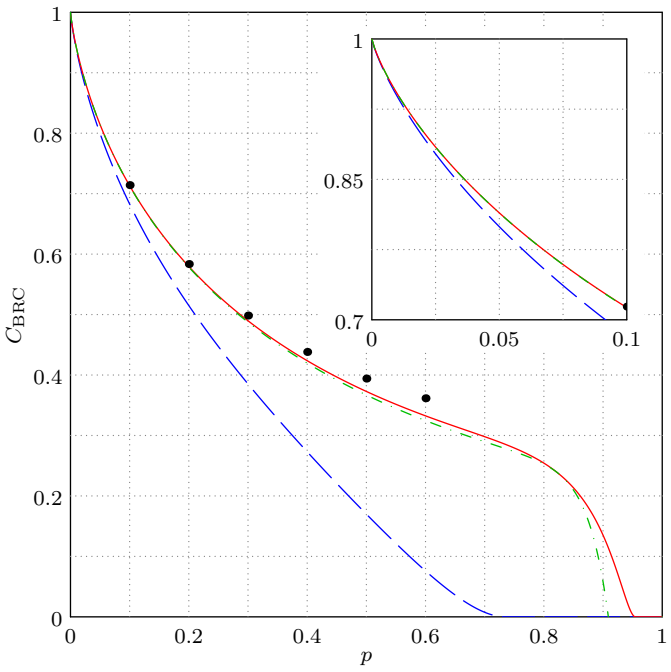

Figure 2.2: Lower bounds on the capacity of the BRC. The bound $R_2^{\mathscr{M}^1}$ from Corollary 2.19 is shown as the long-dashed blue line and the Markov-1 rate in Equation (2.18) is shown as the solid red line. The SIR ($\alpha = \frac{1}{2}$ in Equation(2.18)) is the dash-dotted green line. The numerical lower bounds in [84] are shown as black circles. The inset shows the bounds for small replication probabilities.

Markov-1 rate for the BRC in Equation (2.18) settles this conjecture as being false for the BRC. This is because if the capacity were convex in the replication probability, no rate larger than $(1-p)$ would be achievable, which is clearly not the case as can be seen from Figure 2.2. This implies that, in general, in presence of synchronization errors, the capacity is not convex in the channel parameters. In particular, it is possible that the capacity for the BDC is non-convex as well.

2.4 Channels with Deletions and Replications

Although the bounds in the previous section provide us some idea of the achievable information rates for the BDC and the BRC, they do not generalize in a straightforward manner for an SEC with both deletions and replications[6]. In order to obtain

[6]It is possible to obtain, albeit with a lot more effort than in the cases of the BDC or the BRC, the lower bound $D_2^{\mathscr{X}}$ for a first-order Markov input process for a DRC. We shall omit this here.

bounds when both deletions and replications are present, we take a different approach.

2.4.1 Approximate Non-Stationary Channels

We construct a sequence of channels that approximate the DRC $\mathbf{P}_n$. To this end, we fix $m \in \mathbb{Z}^+$ and let

$$Z_i^{(m)} = \begin{cases} Z_i, & |Z_i| \leq m \\ m \cdot \mathrm{sgn}(Z_i), & |Z_i| > m \end{cases} \tag{2.19}$$

for Z_is given by the $\mathscr{Z}$ process defined in Section 2.2 with $\mathrm{sgn} : \mathbb{R} \mapsto \{\pm 1\}$ defined as

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

We then define the channel model for the m^{th}-approximating channel $\mathbf{P}_{n,m}^{\dagger} = (\mathbb{X}, \mathbb{Y}, P_{n,m}^{\dagger})$ as was done for $\mathbf{P}_n$,

$$Y_i^{(m)} = X_{\Gamma_i^{(m)}} = X_{i - Z_i^{(m)}}, i \in [N_n^{(m)}]$$

where $N_n^{(m)} = \sup\{i \geq 0 : \Gamma_i^{(m)} \leq n | \Gamma_0^{(m)} = 0\}$. It is clear that

$$n - m \leq N_n^{(m)} \leq n + m. \tag{2.20}$$

The input and output alphabets of the channel $\mathbf{P}_{n,m}^{\dagger}$ are $\mathbb{X}$ and $\mathbb{Y}$ respectively, same as those of $\mathbf{P}_n$. The transition probability $P_{n,m}^{\dagger}$ for the channel $\mathbf{P}_{n,m}^{\dagger}$ is defined as in Equation (2.7), but with the channel states defined by the process $\mathscr{Z}^{(m)} \triangleq \{Z_i^{(m)}\}_{i \geq 1}$. The transition probability of the $\mathscr{Z}^{(m)}$ process itself is defined as that induced by Equations (2.19) and (2.6).

We now establish a few properties of the $\mathscr{Z}^{(m)}$ process and the approximating channels $\mathbf{P}_{n,m}^{\dagger}$. We start with some properties of the state process $\mathscr{Z}^{(m)}$ and the index process $\Gamma^{(m)}$ that will be useful in proving subsequent results. The following property establishes the non-stationarity of the sequence of channels $\{\mathbf{P}_{n,m}^{\dagger}\}_{m \geq 0}$.

Lemma 2.21 (Properties of $\mathscr{Z}^{(m)}$)**.** *The state process $\mathscr{Z}^{(m)}$ is a finite, time - inhomogeneous Markov chain. Moreover, the boundary states $\{\pm m\}$ are eventually* absorbing *states, under the measure* P*, in the following two cases.*

(i) For $m = o(n)$ when $p_{\mathsf{d}} \neq p_{\mathsf{r}}$.

(ii) For $m = o(\sqrt{n})$ when $p_{\mathsf{d}} = p_{\mathsf{r}}$.

Proof. From the definition of the $\mathscr{Z}^{(m)}$ process in Equation (2.19), it is clear that $Z_i^{(m)} \in \mathbb{Z}_{\pm m}$ for every $i \in \mathbb{N}$, and that it is a Markov chain. It is therefore a finite Markov chain. The time-inhomogeneity follows by noting the transition probabilities between states, which can be easily shown to be given as follows. For $-m < j < m$, $i \geq 1$,

$$\mathsf{P}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = j) = \begin{cases} p_{\mathsf{r}}, & k = j+1 \\ (1-p_{\mathsf{r}})(1-p_{\mathsf{d}})p_{\mathsf{d}}^{j-k}, & -m < k \leq j \\ (1-p_{\mathsf{r}})p_{\mathsf{d}}^{j+m}, & k = -m \\ 0, & \text{otherwise.} \end{cases}$$

From the states $\{\pm m\}$, the transition probabilities are

$$\mathsf{P}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = -m) = \begin{cases} 1 - p_{\mathsf{r}}\underline{\mathsf{p}}(i,m), & k = -m \\ p_{\mathsf{r}}\underline{\mathsf{p}}(i,m), & k = -m+1 \\ 0, & \text{otherwise,} \end{cases}$$

where

$$\underline{\mathsf{p}}(i,m) = \frac{\mathsf{P}(Z_{i-1} = -m)}{\mathsf{P}(Z_{i-1} \leq -m)},$$

and, for i such that $\mathsf{P}(Z_{i-1} \geq m) > 0$,

$$\mathsf{P}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = m) = \begin{cases} 1 - p_{\mathsf{d}}(1-p_{\mathsf{r}})\overline{\mathsf{p}}(i,m,p_{\mathsf{d}}), & k = m \\ (1-p_{\mathsf{d}})(1-p_{\mathsf{r}})p_{\mathsf{d}}^{m-k}\overline{\mathsf{p}}(i,m,p_{\mathsf{d}}), & -m < k < m \\ (1-p_{\mathsf{r}})p_{\mathsf{d}}^{2m}\overline{\mathsf{p}}(i,m,p_{\mathsf{d}}), & k = -m \\ 0, & \text{otherwise,} \end{cases}$$

where

$$\overline{\mathsf{p}}(i,m,p) = \frac{\sum_{l=0}^{\infty} \mathsf{P}(Z_{i-1} = m+l)p^l}{\mathsf{P}(Z_{i-1} \geq m)}.$$

Note that it is only transitions from the boundary states $\{\pm m\}$ that have time-dependent probabilities.

As was noted in the proofs of Lemmas 2.3 and 2.12, we can write the $\mathscr{Z}$ process as $Z_n = \sum_{i=1}^{n} \Xi_i$ where $\{\Xi_i\}_{i\geq 1}$ is an i.i.d. process with

$$\mathsf{P}(\Xi_1 = \xi) = \begin{cases} p_{\mathsf{r}}, & \xi = 1 \\ (1-p_{\mathsf{d}})(1-p_{\mathsf{r}})p_{\mathsf{d}}^{-\xi}, & \forall\ \xi \leq 0 \\ 0, & \text{otherwise.} \end{cases}$$

As noted before, from the SLLN, $\frac{Z_n}{n} \to \mathsf{E}[\Xi_1] = \frac{p_{\mathsf{r}}-p_{\mathsf{d}}}{1-p_{\mathsf{d}}} \triangleq \chi$ a.s. as $n \to \infty$. Let us write

$$\mathsf{Var}[\Xi_1] = \mathsf{E}[\Xi_1^2] - \mathsf{E}[\Xi_1]^2 = \frac{p_{\mathsf{r}} + p_{\mathsf{d}} + p_{\mathsf{d}}^2 - 3p_{\mathsf{d}}p_{\mathsf{r}}}{(1-p_{\mathsf{d}})^2} \triangleq \nu^2.$$

From the central limit theorem (CLT), we have

$$\mathsf{P}(Z_n \geq m) = \mathsf{P}\Big(\frac{Z_n - n\chi}{\sqrt{n}\nu} \geq \frac{m-n\chi}{\sqrt{n}\nu}\Big) \overset{n\to\infty}{\longrightarrow} \mathrm{Q}\Big(\frac{\mathtt{t}}{\nu}\Big),$$
$$\mathsf{P}(Z_n \leq -m) = \mathsf{P}\Big(\frac{Z_n - n\chi}{\sqrt{n}\nu} \leq \frac{-m-n\chi}{\sqrt{n}\nu}\Big) \overset{n\to\infty}{\longrightarrow} \mathrm{Q}\Big(\frac{\mathtt{b}}{\nu}\Big),$$

where

$$\mathtt{t} \triangleq \lim_{n\to\infty} \frac{m-n\chi}{\sqrt{n}} \quad \text{and} \quad \mathtt{b} \triangleq \lim_{n\to\infty} \frac{m+n\chi}{\sqrt{n}}$$

with $\mathtt{t}, \mathtt{b} \in \mathbb{R} \cup \{\pm\infty\}$, and

$$\mathrm{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}}\, du.$$

Writing $\mathtt{m} \triangleq \lim_{n\to\infty} \frac{m}{n}$, we can say that when $\chi > 0$,

$$\mathsf{P}(Z_n^{(m)} = m) = \mathsf{P}(Z_n \geq m) \overset{n\to\infty}{\longrightarrow} \begin{cases} 1, & \mathtt{m} < \chi \\ \frac{1}{2}, & \mathtt{m} = \chi \\ 0, & \mathtt{m} > \chi, \end{cases}$$

and when $\chi < 0$,

$$\mathsf{P}(Z_n^{(m)} = -m) = \mathsf{P}(Z_n \leq -m) \overset{n\to\infty}{\longrightarrow} \begin{cases} 1, & \mathtt{m} < -\chi \\ \frac{1}{2}, & \mathtt{m} = -\chi \\ 0, & \mathtt{m} > -\chi. \end{cases}$$

When $\chi = 0$, for $m = o(\sqrt{n})$,

$$\left.\begin{array}{l} \mathsf{P}(Z_n^{(m)} = m) \\ \mathsf{P}(Z_n^{(m)} = -m) \end{array}\right\} \xrightarrow{n\to\infty} \frac{1}{2}.$$

Hence

$$\mathsf{P}(Z_n^{(m)} \in \{\pm m\}) \xrightarrow{n\to\infty} 1 \begin{cases} \text{for } m = o(n) \text{ when } \chi \neq 0, \\ \text{for } m = o(\sqrt{n}) \text{ when } \chi = 0. \end{cases}$$

The result then follows by noting that χ is equal (or not equal) to 0 accordingly as p_{d} is equal (or not equal, respectively) to p_{r}. □

Lemma 2.22 (Coarseness of $\{\Gamma^{(m)}\}_{m\geq 0}$)**.** *For a fixed $n \in \mathbb{N}$, for every $m \in \mathbb{Z}^+$,*

$$\{\Gamma_{[N_n]}\} \cap [n] \subset \{\Gamma^{(m)}_{[N_n^{(m)}]}\} \cap [n] \text{ a.s.}$$

and

$$\{\Gamma^{(m+1)}_{[N_n^{(m+1)}]}\} \cap [n] \subset \{\Gamma^{(m)}_{[N_n^{(m)}]}\} \cap [n] \text{ a.s.},$$

where $\{\Gamma_{\mathbb{U}}\}$ denotes the set of elements of the random vector $\Gamma_{\mathbb{U}}$, i.e., where the random variables are not repeated.

Proof. We first note that although N_n and $N_n^{(m)}$ might themselves differ, both sets $\{\Gamma_{[N_n]}\}$ and $\{\Gamma^{(m)}_{[N_n^{(m)}]}\}$ are subsets of $[n] \cup \{0\}$. Therefore, assuming that all random variables X_i where $i \notin [n]$ are constants (in particular, we assume that these random variables are all equal to 0), we can consider the above sets of indices to be $\{\Gamma_{[N(m,n)]}\}$ and $\{\Gamma^{(m)}_{[N(m,n)]}\}$ respectively, where we define

$$N(m,n) \triangleq N_n \vee N_n^{(m)} = \max\{N_n, N_n^{(m)}\}.$$

We have $N(m,n) < \infty$ a.s. for every $n \in \mathbb{N}, m \in \mathbb{Z}^+$.

Let $S_1^+ = \inf\{i > 0 : Z_{i+1} > m\}$ and $T_1^+ = \inf\{i > 0 : Z_{S^+ + 1 + i} \leq m\}$, where we define $\inf\emptyset = \infty$. Then, $\{S_1^+ + 1, S_1^+ + 2, \cdots, T_1^+\} \cap [N(m,n)]$ is the set of instances where Z_i and $Z_i^{(m)}$ differ for the first time as a result of Z_i exceeding m. In this case, $Z_{S_1^+} = m, Z_{S_1^+ + 1} = m + 1$ with probability 1 from the definition of the $\mathscr{Z}$ process. Further, $m + 1 \leq Z_{S_1^+ + j} \leq m + j, j = 1, 2, \cdots, T_1^+$ a.s., implying for this range of js that $S_1^+ - m \leq$

$\Gamma_{S_1^+ + j} \leq S_1^+ + j - m - 1$ a.s.. But, by definition, $Z^{(m)}_{S_1^+ + j} = m, j = 0, 1, \cdots, T_1^+$, and hence $\Gamma^{(m)}_{S_1^+ + j} = S_1^+ + j - m$. Thus, if we write $\mathbb{U}_1^+ = \{S_1^+, S_1^+ + 1, \cdots, S_1^+ + T_1^+\}$, then we have

$$\{\Gamma_{\mathbb{U}_1^+}\} \subset \{\Gamma^{(m)}_{\mathbb{U}_1^+}\} \text{ a.s..}$$

Since $Z_{S_1^+} = Z^{(m)}_{S_1^+} = m$, $\Gamma_i \leq \Gamma_{S_1^+} = S_1^+ - m \; \forall \; i \leq S_1^+$ a.s. from Lemma 2.4. Similarly, since $Z_{S_1^+ + T_1^+ + 1} = Z^{(m)}_{S_1^+ + T_1^+ + 1} \leq m, \Gamma_i \geq \Gamma_{S_1^+ + T_1^+ + 1} \geq S_1^+ + T_1^+ + 1 - m \; \forall \; i \geq S_1^+ + T_1^+ + 1$ a.s. from Lemma 2.4. It follows that the indices in $\{\Gamma^{(m)}_{\mathbb{U}_1^+}\} \setminus \{\Gamma_{\mathbb{U}_1^+}\}$ cannot appear in $\{\Gamma_{\mathbb{U}}\}$ for any $\mathbb{U} \subset [N(m,n)] \setminus \mathbb{U}_1^+$. Using similar arguments, by recursively defining for $i \geq 2$

$$\begin{aligned} S_i^+ &= \inf\{i > S_{i-1}^+ + T_{i-1}^+ : Z_{i+1} > m\}, \\ T_i^+ &= \inf\{i > 0 : Z_{S_i^+ + 1 + i} \leq m\}, \end{aligned}$$

and letting $\mathbb{U}_i^+ = \{S_i^+, S_i^+ + 1, \cdots, S_i^+ + T_i^+\}$, we can show that

$$\{\Gamma_{\mathbb{U}_i^+}\} \subset \{\Gamma^{(m)}_{\mathbb{U}_i^+}\} \text{ a.s.} \forall \; i \geq 1.$$

Similarly, consider

$$\begin{aligned} S_0^- &= T_0^- = 0, \\ S_i^- &= \inf\{i > S_{i-1}^- + T_{i-1}^- : Z_i < -m\} \text{ and} \\ T_i^- &= \inf\{i > 0 : Z_{S_i^- + i} \geq -m\}, i \geq 1. \end{aligned}$$

Then, $Z_{S_i^- + T_i^-} = -m$ and $Z_{S_i^- + T_i^- - 1} = -m - 1$ with probability 1. Further, with probability 1, $-m - j \leq Z_{S_i^- + T_i^- - j} \leq -m - 1, j = 1, 2, \cdots, T_i^-$ a.s., implying $S_i^- + T_i^- - j + m + 1 \leq \Gamma_{S_i^- + T_i^- - j} \leq S_i^- + T_i^- + m$ a.s.. By definition, $\Gamma^{(m)}_{S_i^- + T_i^- - j} = S_i^- + T_i^- - j + m, j = 0, 1, \cdots, T_i^-$, and consequently

$$\{\Gamma_{\mathbb{U}_i^-}\} \subset \{\Gamma^{(m)}_{\mathbb{U}_i^-}\} \text{ a.s.} \forall \; i \geq 1$$

where $\mathbb{U}_i^- = \{S_i^-, S_i^- + 1, \cdots, S_i^- + T_i^-\}$. As before, the missing indices cannot appear in $\Gamma_{\mathbb{U}}$ for any $\mathbb{U} \subset [N(m,n)] \setminus \mathbb{U}_i^-$.

Therefore, writing $\mathbb{U}^{\pm} = \bigcup_{i \geq 1} (\mathbb{U}_i^+ \cup \mathbb{U}_i^-)$, we have

$$\{\Gamma_{\mathbb{U}^{\pm}}\} \subset \{\Gamma^{(m)}_{\mathbb{U}^{\pm}}\} \text{ a.s.}$$

Let $\mathbb{U}^0 \triangleq [N(m,n)] \setminus \mathbb{U}^{\pm}$. Since $\mathbb{U}^{\pm}$ consists of all indices i where Γ_i and $\Gamma_i^{(m)}$ differ, we have from the above relation that almost surely,

$$\begin{aligned}
&\{\Gamma_{\mathbb{U}^{\pm}}\} \cup \{\Gamma_{\mathbb{U}^0}\} \subset \{\Gamma^{(m)}_{\mathbb{U}^{\pm}}\} \cup \{\Gamma^{(m)}_{\mathbb{U}^0}\} \\
&\Rightarrow \{\Gamma_{[N(m,n)]}\} \subset \{\Gamma^{(m)}_{[N(m.n)]}\} \\
&\Rightarrow \{\Gamma_{[N_n]}\} \cap [n] \subset \{\Gamma^{(m)}_{[N_n^{(m)}]}\} \cap [n].
\end{aligned}$$

We are interested in the intersection in the last step above since only indices in the set $[n]$ are indices of non-constant random variables.

We can use an argument similar to the one above to show that $\forall\, m \in \mathbb{Z}^+$,

$$\{\Gamma^{(m+1)}_{[N_n^{(m+1)}]}\} \cap [n] \subset \{\Gamma^{(m)}_{[N_n^{(m)}]}\} \cap [n] \text{ a.s..}$$

□

Thus, the process $\Gamma^{(m)}$ gets "coarser" as m is increased.The following is an immediate consequence of this property.

Proposition 2.23. *For every $m \in \mathbb{Z}^+$,*

$$I(X_{[n]};Y_{[N_n]}) \leq I(X_{[n]};Y^{(m)}_{[N_n^{(m)}]}) \quad \textit{and} \quad I(X_{[n]};Y^{(m+1)}_{[N_n^{(m+1)}]}) \leq I(X_{[n]};Y^{(m)}_{[N_n^{(m)}]}).$$

Proof. We use the result from Lemma 2.22. Let us define

$$\begin{aligned}
\mathbb{S}_m &= \bigcap_{n\geq 1} \{\vartheta \in \mathbb{S} : \{\Gamma_{[N_n]}\} \cap [n] \subset \{\Gamma^{(m)}_{[N_n^{(m)}]}\} \cap [n]\}, \text{ and} \\
\hat{\mathbb{S}}_m &= \bigcap_{n\geq 1} \{\vartheta \in \mathbb{S} : \{\Gamma^{(m+1)}_{[N_n^{(m+1)}]}\} \cap [n] \subset \{\Gamma^{(m)}_{[N_n^{(m)}]}\} \cap [n]\}
\end{aligned}$$

and let

$$\mathbb{S}^* = \bigcap_{m\in\mathbb{Z}^+} (\mathbb{S}_m \cap \hat{\mathbb{S}}_m).$$

Clearly, $\mathsf{P}(\mathbb{S}^*) = 1$. Then, confining the expectations over the set $\mathbb{S}^*$,

$$I(X_{[n]};Y^{(m)}_{[N_n^{(m)}]}) - I(X_{[n]};Y_{[N_n]}) = I_{\mathbb{S}^*}(X_{[n]};X_{\Gamma^{(m)}_{[N_n^{(m)}]}\setminus\Gamma_{[N_n]}}|X_{\Gamma_{[N_n]}}) \geq 0,$$

where $I_{\mathbb{S}^*}(\cdot)$ denotes the mutual information obtained after confining the expectations to the set $\mathbb{S}^*$. Similarly, we have $I(X_{[n]};Y^{(m+1)}_{[N_n^{(m+1)}]}) \leq I(X_{[n]};Y^{(m)}_{[N_n^{(m)}]})$. □

Intuitively, the above result is true because the "drift" between the input and the output processes is bounded by m for the approximating channel $\mathbf{P}^{\dagger}_{n,m}$, whereas it is unbounded for the DRC $\mathbf{P}_n$ (or equivalently $\mathbf{Q}_n$). The result below, which gives a total ordering of the sequence of channels $\{\mathbf{P}^{\dagger}_{n,m}\}_{m\geq 0}$ in terms of their mutual information rates, follows immediately from Proposition 2.23.

Corollary 2.24 (Total Ordering of $\{\mathbf{P}^{\dagger}_{n,m}\}_{m\geq 0}$)**.** *For any $n \in \mathbb{N}$, the sequence $\{I^{\dagger}_{n,m}\}_{m\geq 0}$, where*

$$I^{\dagger}_{n,m} \triangleq \frac{1}{n} I(X_{[n]}; Y^{(m)}_{[N^{(m)}_n]}),$$

is non-increasing. Since $I^{\dagger}_{n,m} \in [0,1]\ \forall\ n \in \mathbb{N}$ and $m \in \mathbb{Z}^+$, $\lim_{m\to\infty} I^{\dagger}_{n,m}$ exists and is equal to $\inf_{m\geq 0} I^{\dagger}_{n,m}$. □

Proposition 2.25 (Information limits)**.** *For any $n \in \mathbb{N}$, we have*

$$I_n \triangleq \frac{1}{n} I(X_{[n]}; Y_{[N_n]}) = I^{\dagger}_n \triangleq \lim_{m\to\infty} \downarrow I^{\dagger}_{n,m} = \inf_{m\geq 0} I^{\dagger}_{n,m}.$$

Consequently, for a stationary, ergodic input process $\mathscr{X}$,

$$I_{\mathscr{X}} \triangleq \lim_{n\to\infty} I_n = \inf_{n\geq 1} I_n = I^{\dagger}_{\mathscr{X}} \triangleq \lim_{n\to\infty} I^{\dagger}_n = \inf_{n\geq 1} I^{\dagger}_n,$$

so that

$$C = \sup_{\mathscr{X}} I_{\mathscr{X}} = \sup_{\mathscr{X}} I^{\dagger}_{\mathscr{X}}$$

for stationary, ergodic, Markov processes $\mathscr{X}$.

Proof. The last equality in the first line is from Corollary 2.24. From Proposition 2.23, we have $I_n \leq I^{\dagger}_{n,m}\ \forall\ m \in \mathbb{Z}^+$, from which $I_n \leq I^{\dagger}_n$ follows. The equality is true because of the following. If we let $\mathscr{F}_{n,m} \triangleq \sigma(\{X_{[n]}, Y^{(m)}_{[N^{(m)}_n]}\})$, the sigma-algebra generated by the random variables $\{X_{[n]}, Y^{(m)}_{[N^{(m)}_n]}\}$, then $\{\mathscr{F}_{n,m}\}_{m\geq 0}$ is a *filtration* [123, §10.1], i.e.,

$$\mathscr{F}_{n,m} \subset \mathscr{F}_{n,m+1}\ \forall\ m \geq 0.$$

Thus, $\mathsf{P}(X_{[n]}, Y^{(m)}_{[N^{(m)}_n]})$ is the restriction of P to $\mathscr{F}_{n,m}$. From [11, Theorem 2], we have that $I_n = I^{\dagger}_n$.

The limit of I_n as n goes to infinity exists and is equal to the infimum of the sequence from the subadditivity of the sequence $\{nI_n\}_{n\geq 1}$ and Fekete's Lemma (cf. [32, Appendix II]). The last claim made is true from Proposition 2.2. □

Corollary 2.26. *For any $n \in \mathbb{N}$, we have that*

$$C_n = \sup_{\mathsf{P}(X_{[n]})} I_n = C_n^\dagger \triangleq \sup_{\mathsf{P}(X_{[n]})} I_n^\dagger$$

where C_n is as defined in Theorem 2.1. Therefore

$$C = \lim_{n\to\infty} C_n = \inf_{n\geq 1} C_n = C^\dagger \triangleq \lim_{n\to\infty} C_n^\dagger. \qquad \square$$

Although $\{\mathbf{P}_{n,m}^\dagger\}_{m\geq 0}$ is a sequence of channels that approximate $\mathbf{P}_n$, and have the properties discussed so far in this subsection, they are not useful as finite-state channels (FSCs), as shown below.

Lemma 2.27 (FSCs $\mathbf{P}_{n,m}^\dagger$)**.** *For any $m \in \mathbb{Z}^+$, for a stationary, ergodic input process $\mathscr{X}$,*

$$I_{\mathscr{X}}^\dagger(m) \triangleq \lim_{n\to\infty} I_{n,m}^\dagger = \mathscr{H}(\mathscr{X})$$

so that

$$C^\dagger(m) \triangleq \sup_{\mathscr{X}} I_{\mathscr{X}}^\dagger(m) = 1.$$

Proof. From Lemma 2.21, the states $\{\pm m\}$ are eventually absorbing for any $m \in \mathbb{Z}^+$. Hence, in the limit as $n \to \infty$, the channel only has a delay of $\pm m$, and hence the result. $\square$

Thus, the approximating channels $\{\mathbf{P}_{n,m}^\dagger\}_{m\geq 0}$ do not provide useful bounds for information rates achievable on the DRC $\mathbf{P}_n$. We now attempt to obtain approximate channels that are stationary and, as FSCs, can be used to estimate achievable rates for the DRC.

2.4.2 Approximate Stationary Channels

Let $m \in \mathbb{Z}^+$. Fix $n \in \mathbb{N}$. Consider the channel $\mathbf{P}_{n,m}^\star = (\mathbb{X}, \mathbb{Y}, P_{n,m}^\star)$ where

$$Y_i^{(m)} = X_{\Gamma_i^{(m)}} = X_{i - Z_i^{(m)}}, i \in [N_n^{(m)}]$$

with $N_n^{(m)}, \Gamma_{[N_n^{(m)}]}^{(m)}$ and $Z_{[N_n^{(m)}]}^{(m)}$ as defined for the channel $\mathbf{P}_{n,m}^\dagger$. The difference will be in the underlying measure $\mathsf{P}_{\langle m\rangle}$. Let the measure $\mathsf{P}_{\langle m\rangle}$ be such that the $\mathscr{Z}^{(m)}$ process is a finite, time-homogeneous, first-order Markov chain with transition probabilities

$$\mathsf{P}_{\langle m\rangle}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = j) = \mathsf{P}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = j)$$

when $-m < j < m$,

$$\mathsf{P}_{\langle m\rangle}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = -m) = \begin{cases} 1-p_{\mathsf{r}}, & k = -m \\ p_{\mathsf{r}}, & k = -m+1 \\ 0, & \text{otherwise,} \end{cases}$$

and

$$\mathsf{P}_{\langle m\rangle}(Z_i^{(m)} = k | Z_{i-1}^{(m)} = m) = \begin{cases} 1-p_{\mathsf{d}}(1-p_{\mathsf{r}}), & k = m \\ (1-p_{\mathsf{d}})(1-p_{\mathsf{r}})p_{\mathsf{d}}^{m-k}, & -m < k < m \\ (1-p_{\mathsf{r}})p_{\mathsf{d}}^{2m}, & k = -m \\ 0, & \text{otherwise.} \end{cases}$$

Note that the measure $\mathsf{P}_{\langle m\rangle}$ differs from P only for state paths that reach beyond the states $\{\pm m\}$. The transition probabilities $P^{\star}_{n,m}$ for the channel $\mathbf{P}^{\star}_{n,m}$ can now be defined as in Equation (2.7), but under the measure $\mathsf{P}_{\langle m\rangle}$. The stationarity of the channels $\mathbf{P}^{\star}_{n,m}$ follows from the time-homogeneity of the $\mathscr{Z}^{(m)}$ process.

Remark 2.3. Note that the sequence of sigma-algebras $\{\mathscr{G}_m\}_{m\geq 0}$ where $\mathscr{G}_m \triangleq \sigma(\mathscr{Z}^{(m)})$ forms a filtration. The sequence of measures $\mathsf{P}_{\langle m\rangle}$ as defined above seem to be defined only on the corresponding sigma-algebras $\mathscr{G}_m$s for each $m \in \mathbb{Z}^+$. However, we can extend these measures to the sigma-algebra $\mathscr{B}$ as follows.

We will first assume that $\mathbb{S} = \mathbb{S}_{\mathscr{X}} \times \mathbb{S}_{\mathscr{Z}}$ and that $\mathscr{B} = \mathscr{B}_{\mathscr{X}} \times \mathscr{B}_{\mathscr{Z}}$ with $\mathscr{B}_{\mathscr{X}} = \sigma(\mathscr{X})$ and $\mathscr{B}_{\mathscr{Z}} = \sigma(\mathscr{Z})$, i.e., the space $(\mathbb{S}, \mathscr{B})$ is a product space. Since in our model $\mathscr{X} \perp\!\!\!\perp \mathscr{Z}$, there is no loss of generality in this assumption.

By defining the stationary transition probabilities $\mathsf{P}_{\langle m\rangle}(Z_1|Z_0)$ as in Section 2.4.2, the measures $\mathsf{P}_{\langle m\rangle}$ are well-defined over $\mathscr{G}_m = \sigma(\mathscr{Z}^{(m)})$. Let $(\mathscr{Z}^{(m)})^{-1}(\bar{z}) \triangleq \{\vartheta \in \mathbb{S}_{\mathscr{Z}} : \mathscr{Z}^{(m)}(\vartheta) = \bar{z}\}$ for $\bar{z} \in \overline{\mathbb{Z}}_{\pm m}$, and similarly $\mathscr{Z}^{-1}(\bar{z}) \triangleq \{\vartheta \in \mathbb{S}_{\mathscr{Z}} : \mathscr{Z}(\vartheta) = \bar{z}\}$. Then, clearly

$$\mathscr{Z}^{-1}(\bar{z}) \subset (\mathscr{Z}^{(m)})^{-1}(\bar{z}) \; \forall \, \bar{z} \in \overline{\mathbb{Z}}_{\pm m}$$

and

$$\mathscr{Z}^{-1}(\bar{z}) \in \mathscr{B}_{\mathscr{Z}}, (\mathscr{Z}^{(m)})^{-1}(\bar{z}) \in \mathscr{G}_m \; \forall \, \bar{z} \in \overline{\mathbb{Z}}_{\pm m}.$$

Then, we define

$$\mathsf{P}_{\langle m\rangle}(\mathscr{Z}^{-1}(\bar{z})) = \mathsf{P}_{\langle m\rangle}((\mathscr{Z}^{(m)})^{-1}(\bar{z})) \; \forall \, \bar{z} \in \overline{\mathbb{Z}}_{\pm m}. \tag{2.21}$$

This will imply that for every $\bar{z} \in \overline{\mathbb{Z}}_{\pm m}$,

$$\mathsf{P}_{\langle m \rangle}((\mathscr{Z}^{(m)})^{-1}(\bar{z}) \setminus \mathscr{Z}^{-1}(\bar{z})) = 0.$$

By definition, we also have for $\bar{z} \in \overline{\mathbb{Z}} \setminus \overline{\mathbb{Z}}_{\pm m}$ that $(\mathscr{Z}^{(m)})^{-1}(\bar{z}) = \emptyset$ so that the associated probability is zero under any measure $\mathsf{P}, \mathsf{P}_{\langle m \rangle}$. We can now consider the space $(\mathbb{S}_{\mathscr{Z}}, \mathscr{B}_{\mathscr{Z}}, \mathsf{P}_{\langle m \rangle})$ to be obtained from $(\mathbb{S}_{\mathscr{Z}}, \mathscr{G}_m, \mathsf{P}_{\langle m \rangle})$ along with the definition (2.21) and subsequent *completion* [97, §2.6.19].

By now defining $\mathsf{P}_{\langle m \rangle}(\mathscr{X}) = \mathsf{P}(\mathscr{X})$ independent of m, we can extend the measure $\mathsf{P}_{\langle m \rangle}$ to $\mathscr{B} = \sigma(\{\mathscr{X}, \mathscr{Z}\})$ for each $m \in \mathbb{Z}^+$ as required. □

The lemma below shows that for a fixed $m \in \mathbb{Z}^+$, the FSC $\mathbf{P}^{\star}_{n,m}$ is an indecomposable FSC [39, §4.6].

Lemma 2.28 ($\mathbf{P}^{\star}_{n,m}$ Indecomposable)**.** *The FSC $\mathbf{P}^{\star}_{n,m}$ is indecomposable for every $m \in \mathbb{Z}^+$ for $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in (0,1)^2$.*

Proof. Fix $m \in \mathbb{Z}^+$. We need to make a couple of modifications to put the channels $\{\mathbf{P}^{\star}_{n,m}\}_{n \geq 1}$ in the parlance of discrete FSCs. First, we set

$$\acute{Y}_i^{(m)} = Y_{i-m}^{(m)} = X_{i-m-Z_{i-m}^{(m)}} = X_{i-\acute{Z}_i^{(m)}} \text{ for } i \in [n].$$

Note that $\acute{Z}_i^{(m)} = m + Z_{i-m}^{(m)} \in [0:2m]$, and hence the channel producing $\acute{Y}_{[n]}^{(m)}$ is "causal". Let the "state" $W_i^{(m)}$ of the channel $\mathbf{P}^{\star}_{n,m}$ at time $i \in [n]$ be defined as

$$W_i^{(m)} = (X_{[i-2m:i-1]}, \acute{Z}_i^{(m)}) \in \mathbb{X}^{2m} \times [0:2m],$$

where we set $X_i = 0$ for $i \notin [n]$. Note that we need to redefine the state of the channel in this case to keep the factorization

$$\mathsf{P}_{\langle m \rangle}(\acute{Y}_i^{(m)}, W_{i+1}^{(m)} | X_i, W_i^{(m)}) = \mathsf{P}_{\langle m \rangle}(\acute{Y}_i^{(m)} | X_i, W_i^{(m)}) \cdot \mathsf{P}_{\langle m \rangle}(W_{i+1}^{(m)} | X_i, W_i^{(m)}).$$

Since $\acute{\mathscr{Z}}^{(m)}$ is a finitely delayed, finitely shifted version of $\mathscr{Z}^{(m)}$, and because $\mathscr{Z}^{(m)}$ is an irreducible, aperiodic Markov chain under the measure $\mathsf{P}_{\langle m \rangle}$ [66, Chapter 1] as long as $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in (0,1)^2$, so is $\acute{\mathscr{Z}}^{(m)}$. In particular, we have that for every $i \geq 2m$,

$$\min_{z \in [0:2m]} \mathsf{P}_{\langle m \rangle}(\acute{Z}_i^{(m)} = z | \acute{Z}_0^{(m)} = z') > 0 \; \forall \; z' \in [0:2m].$$

This implies that for $i = 2m$ and $\bar{x} \in \overline{\mathbb{X}}$, by choosing[7] $w = (x_{[0:2m-1]}, z)$ for any $z \in [0 : 2m]$, we see that

$$\mathsf{P}_{\langle m \rangle}(W_{2m}^{(m)} = w | \overline{X} = \bar{x}, W_0^{(m)} = w') > 0$$

for every $w' \in \mathbb{X}^{2m} \times [0 : 2m]$. From [39, Theorem 4.6.3], we have the desired result. □

Remark 2.4. Note that in the description of the causal channel in the proof above, we have discarded the part of the output $Y^{(m)}_{[n-m+1:N_n^{(m)}]}$ by considering the causal output $\acute{Y}^{(m)}_{[n]}$. This will however not matter in the estimation of the information rate since

$$0 \leq \frac{1}{n} I(X_{[n]}; Y^{(m)}_{[N_n^{(m)}]}) - \frac{1}{n} I(X_{[n]}; \acute{Y}^{(m)}_{[n]}) \leq \frac{m}{n},$$

and since $m \in \mathbb{Z}^+$ is fixed, the rates are the same in the limit as $n \to \infty$. □

Corollary 2.29 (Capacity of FSC $\mathbf{P}^{\star}_{n,m}$)**.** *For $m \in \mathbb{Z}^+$ and the sequence of FSCs $\{\mathbf{P}^{\star}_{n,m}\}_{n \geq 1}$ with $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in (0,1)^2$, the capacity is given by*

$$C^{\star}(m) = \lim_{n \to \infty} \sup_{\mathsf{P}_{\langle m \rangle}(X_{[n]})} \frac{1}{n} I(X_{[n]}; Y^{(m)}_{[N_n^{(m)}]}) \triangleq \lim_{n \to \infty} \sup_{\mathsf{P}_{\langle m \rangle}(X_{[n]})} I^{\star}_{n,m}.$$

Proof. From Lemma 2.28 and Remark 2.4, we have $C^{\star}(m)$ as defined in the statement can be written as

$$C^{\star}(m) = \lim_{n \to \infty} \sup_{\mathsf{P}_{\langle m \rangle}(X_{[n]})} \frac{1}{n} I(X_{[n]}; \acute{Y}^{(m)}_{[n]}).$$

Now since

$$I(X_{[n]}; \acute{Y}^{(m)}_{[n]}) = I(X_{[n]}; W_0^{(m)}) + I(X_{[n]}; \acute{Y}^{(m)}_{[n]} | W_0^{(m)}) - I(X_{[n]}; W_0^{(m)} | \acute{Y}^{(m)}_{[n]}),$$

we have

$$\begin{aligned} |I(X_{[n]}; \acute{Y}^{(m)}_{[n]}) - I(X_{[n]}; \acute{Y}^{(m)}_{[n]} | W_0^{(m)})| &\leq \log_2 \left((2m+1) |\mathbb{X}|^{2m} \right) \\ &= 2m \log_2 |\mathbb{X}| + \log_2 (2m+1). \end{aligned}$$

Therefore

$$C^{\star}(m) = \lim_{n \to \infty} \sup_{\mathsf{P}_{\langle m \rangle}(X_{[n]})} \frac{1}{n} I(X_{[n]}; \acute{Y}^{(m)}_{[n]} | W_0^{(m)}).$$

From [39, Theorem 4.6.4], the quantity on the right hand side of the above equality exists and is the capacity of the indecomposable FSCs $\{\mathbf{P}^{\star}_{n,m}\}_{n \geq 1}$. □

[7] Set $x_0 = 0$ by convention.

Corollary 2.30 (Markov Capacity of $\mathbf{P}^{\star}_{n,m}$)**.** *For the FSCs $\{\mathbf{P}^{\star}_{n,m}\}_{n\geq 1}$, the capacity $C^{\star}(m)$ can be written [32] as*

$$C^{\star}(m) = \sup_{\mathscr{X}} \lim_{n\to\infty} \frac{1}{n} I(X_{[n]}; Y^{(m)}_{[N^{(m)}_n]}) = \sup_{\mathscr{X}} \lim_{n\to\infty} I^{\star}_{n,m} \triangleq \sup_{\mathscr{X}} I^{\star}_{\mathscr{X}}(m)$$

where the supremum is over all stationary, ergodic input sources $\mathscr{X}$. Further, since the FSCs are indecomposable, we have [16]

$$C^{\star}(m) = \sup_{\mathscr{X}_{\mathscr{M}}} \lim_{n\to\infty} \frac{1}{n} I(X_{[n]}; Y^{(m)}_{[N^{(m)}_n]}) = \sup_{\mathscr{X}_{\mathscr{M}}} I^{\star}_{\mathscr{X}_{\mathscr{M}}}(m)$$

where $\mathscr{X}_{\mathscr{M}}$ denotes stationary, ergodic, Markov sources over the alphabet $\mathbb{X}$. □

From Lemma 2.28, since $\{\mathbf{P}^{\star}_{n,m}\}_{n\geq 1}$ are indecomposable FSCs, we have from [14] that

$$-\frac{1}{n}\log_2 \mathsf{P}_{\langle m\rangle}(X_{[n]}, Y^{(m)}_{[N^{(m)}_n]}) \to \lim_{n\to\infty} \frac{H(X_{[n]}, Y^{(m)}_{[N^{(m)}_n]})}{n} = \hat{\mathscr{H}}(\mathscr{X}, \mathscr{Y}^{(m)}),$$

$$-\frac{1}{n}\log_2 \mathsf{P}_{\langle m\rangle}(Y^{(m)}_{[N^{(m)}_n]}) \to \lim_{n\to\infty} \frac{H(Y^{(m)}_{[N^{(m)}_n]})}{n} = \hat{\mathscr{H}}(\mathscr{Y}^{(m)}),$$

as $n \to \infty$ a.s., where the entropies are calculated with respect to the measure $\mathsf{P}_{\langle m\rangle}$. Therefore

$$I^{\star}_{\mathscr{X}}(m) = \mathscr{H}(\mathscr{X}) + \hat{\mathscr{H}}(\mathscr{Y}^{(m)}) - \hat{\mathscr{H}}(\mathscr{X}, \mathscr{Y}^{(m)})$$

can be estimated numerically using the forward passes of the BCJR algorithm [10] to estimate $\hat{\mathscr{H}}(\mathscr{X}, \mathscr{Y}^{(m)})$ and $\hat{\mathscr{H}}(\mathscr{Y}^{(m)})$, as in [7, 94]. Moreover, optimizing Markov input sources numerically is possible [57, 116] for these FSCs.

In Figure 2.3, we plot the SIRs, $C^{\star}_{\text{iud}}(m)$, for the indecomposable FSCs $\{\mathbf{P}^{\star}_{n,m}\}_{n\geq 1}$ obtained through numerical simulations for $1 \leq m \leq 8$ and $p_{\mathsf{d}} = p_{\mathsf{r}} = p \in [0, \frac{1}{2}]$. The value of n used for the estimation was 5×10^5. The error in estimation is consequently upper bounded by 0.15%.

A couple of observations are worthwhile noting. First, the SIRs $\{C^{\star}_{\text{iud}}(m)\}_{m\geq 0}$ are non-increasing. This hints at a total ordering of the FSCs $\{\mathbf{P}^{\star}_{n,m}\}_{m\geq 0}$ with respect to the information rates similar to what we had in Corollary 2.24. Second, we see that for small values of p, the SIRs get bunched up as m increases, i.e., the SIRs $C^{\star}_{\text{iud}}(m)$

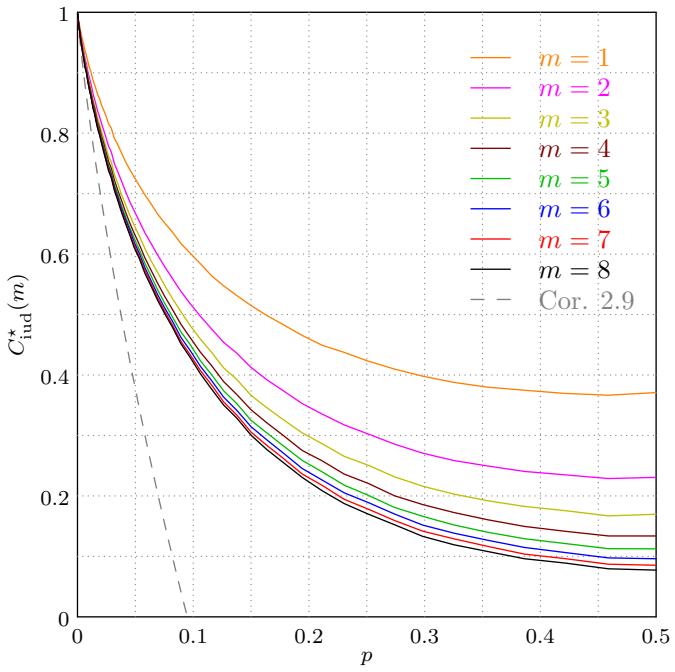

Figure 2.3: SIR estimates for the FSCs $\{\mathbf{P}^{\star}_{n,m}\}_{n\geq 1}$ with $p_{\mathsf{d}} = p_{\mathsf{r}} = p \in [0, \frac{1}{2}]$ for different m values are shown in solid lines. The lower bound on the capacity of the SDRC from Corollary 2.9 is also shown as the dashed line.

converge quickly, so that we have a good estimate of

$$C^{\star}_{\mathrm{iud}}(\infty) \triangleq \lim_{m\to\infty} C^{\star}_{\mathrm{iud}}(m)$$

for p close to 0.

Proposition 2.31. *For $n \in \mathbb{N}$, we have*

$$I^{\star}_{n} \triangleq \liminf_{m\to\infty} I^{\star}_{n,m} = I_n.$$

Thus,

$$C = \sup_{\mathscr{X}} \inf_{n\geq 1} \liminf_{m\to\infty} I^{\star}_{n,m}.$$

Proof. For a fixed $n \in \mathbb{N}$, since we have that $\mathsf{P}_{\langle m\rangle}(X_{[n]}, Y^{(m)}_{[N_n^{(m)}]}) \to \mathsf{P}(X_{[n]}, Y_{[N_n]})$ as $m \to \infty$ for every $\vartheta \in \mathbb{S}$, $\mathsf{P}_{\langle m\rangle}(X_{[n]}, Y^{(m)}_{[N_n^{(m)}]})$ converges to $\mathsf{P}(X_{[n]}, Y_{[N_n]})$ in total variation as $m \to \infty$. Consequently, from [26, Corollary 1′], we have the desired result. □

Remark 2.5. We conjecture that $I^{\star}_{n,m} \to I^{\star}_{n} = I_n$ as $m \to \infty$. From [26, Corollary 1′], one needs to show uniform integrability of the *information densities* $i(X_{[n]}, Y^{(m)}_{[N^{(m)}_n]})$ for the conjecture to be true. Alternatively, if the sequence of channels $\{\mathbf{P}^{\star}_{n,m}\}_{m\geq 0}$ is totally ordered for every $n \in \mathbb{N}$ with respect to the mutual information rates, as was the case for the sequence $\{\mathbf{P}^{\dagger}_{n,m}\}_{m\geq 0}$ (cf. Corollary 2.24), i.e., if $\{I^{\star}_{n,m}\}_{m\geq 0}$ is a non-increasing sequence for every $n \in \mathbb{N}$, then we know that

$$\lim_{m\to\infty} I^{\star}_{n,m} = I^{\star}_{n},$$

and from Proposition 2.31, $I^{\star}_{n,m} \downarrow I_n$ follows. Unfortunately, we are not able to show this monotonicity in the sequence $\{I^{\star}_{n,m}\}_{m\geq 0}$ as we argued in the case of the sequence $\{I^{\dagger}_{n,m}\}_{m\geq 0}$. Although the coarseness property of the $\Gamma^{(m)}$ process (cf. Lemma 2.22) still holds, the different measures $\mathsf{P}_{\langle m\rangle}$ being used for each $m \in \mathbb{Z}^{+}$ do not allow us to generalize the result of Corollary 2.24. However, Figure 2.3 provides sufficient empirical evidence for this monotonicity conjecture. □

2.4.3 Approximating Channels for the SDRC

In this subsection, we consider the SDRC, i.e., the case when $p_{\mathsf{d}} = p_{\mathsf{r}} = p \in [0, 1)$. This channel is of interest since in practice, systems prone to mis-synchronization are usually not biased to produce more deletions or replications. For the case of the SDRC, we can fix m to be a *sublinear* function of n satisfying a simple condition and define a sequence of approximating channels whose information rates satisfy a relation similar to Proposition 2.31.

Lemma 2.32. *For the SDRC, for every $n \in \mathbb{N}$, let $m \in \mathbb{N}$. Then,*

$$\mathsf{P}_{\langle m\rangle}\Big(\max_{i=1}^{N^{(m)}_n} |Z_i| \geq m\Big) = \mathsf{P}\Big(\max_{i=1}^{N^{(m)}_n} |Z_i| \geq m\Big) = O\Big(\frac{n+m}{m^2}\Big). \tag{2.22}$$

Proof. As noted in the proof of Lemma 2.21, we have for every $n \in \mathbb{N}$ that Z_n is the n^{th} partial sum of the i.i.d. process $\{\Xi_i\}_{i\geq 1}$. For the SDRC, we have $\mathsf{E}[\Xi_1] = \chi = 0$ and $\mathsf{Var}[\Xi_1] = \nu^2 = \frac{2p}{1-p} < \infty$ since $p \in [0, 1)$.

Let $\mathscr{S}_n = \sigma(\{Z_n\}) \subset \mathscr{B}$, the sigma-algebra generated by Z_n, for every $n \in \mathbb{N}$. Clearly, $\mathscr{S}_n = \sigma(\{\Xi_{[n]}\})$ so that $\{\mathscr{S}_n\}_{n\geq 1}$ is a filtration, and $Z_n \in \mathscr{S}_n$ by definition. Let

$\mathscr{S}_n \uparrow \mathscr{S} \subset \mathscr{B}$ as $n \to \infty$. Then for every $n \in \mathbb{N}$, $Z_n \in L^2(\mathbb{S}, \mathscr{B}, \mathsf{P})$ since

$$\begin{aligned}\mathsf{E}[|Z_n|^2] = \mathsf{E}[Z_n^2] = \mathsf{E}\Big[\Big(\sum_{i=1}^{n} \Xi_i\Big)^2\Big] &= \sum_{i=1}^{n} \mathsf{E}[\Xi_i^2] + \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \mathsf{E}[\Xi_i \Xi_j] \\ &= n \cdot \mathsf{Var}[\Xi_1] + \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} \mathsf{E}[\Xi_i]\mathsf{E}[\Xi_j] = n\frac{2p}{1-p} < \infty.\end{aligned}$$

Further, $\mathsf{E}[Z_n|\mathscr{S}_{n-1}] = \mathsf{E}[Z_{n-1} + \Xi_n|\mathscr{S}_{n-1}] = Z_{n-1}$. Therefore, $\{Z_n, \mathscr{S}_n\}_{n\geq 1}$ is a martingale under the measure P. Consequently, $\{|Z_n|, \mathscr{S}_n\}_{n\geq 1}$ is a submartingale.

Since $|Z_n| \in L^2(\mathbb{S}, \mathscr{B}, \mathsf{P})$, from Doob's submartingale inequality [123, §14.6], we have

$$\mathsf{P}(\max_{i=1}^{n} |Z_i| \geq m) \leq \frac{\mathsf{E}[|Z_n|^2]}{m^2} = \Big(\frac{2p}{1-p}\Big)\frac{n}{m^2}.$$

We have from Equation (2.20) that $N_n^{(m)} \leq n + m$. The result then follows by noting that

$$\mathsf{P}\Big(\max_{i=1}^{N_n^{(m)}} |Z_i| \geq m\Big) \leq \mathsf{P}(\max_{i=1}^{n+m} |Z_i| \geq m)$$

and the above result. The bound with respect to the measure $\mathsf{P}_{\langle m \rangle}$ is true because

$$\mathsf{P}_{\langle m \rangle}\Big(\vartheta \in \mathbb{S} : \max_{i=1}^{N_n^{(m)}(\vartheta)} |Z_i(\vartheta)| \geq m\Big) = \mathsf{P}\Big(\vartheta \in \mathbb{S} : \max_{i=1}^{N_n^{(m)}(\vartheta)} |Z_i(\vartheta)| \geq m\Big)$$

from the definition of the measure $\mathsf{P}_{\langle m \rangle}$ (See Section 2.4.2 and Remark 2.3). □

The significance of the above result can be seen by noticing that, for the SDRC, the probability (under measure P or $\mathsf{P}_{\langle m \rangle}$) with which the approximating channels introduced in the previous two subsections differ from the actual channel can be made arbitrarily small by setting $m(n) = \omega(\sqrt{n})$, i.e., $\lim_{n\to\infty} \frac{m(n)}{\sqrt{n}} = \infty$, and choosing a large enough n. For the so-chosen sequence of approximating channels, we can conclude that the limiting channel characterizes the SDRC from the following result.

Proposition 2.33 (Approximating SDRC)**.** *For the SDRC,*

$$I_{\mathscr{X}} = \lim_{n\to\infty} I_n = \liminf_{n\to\infty} I^{\star}_{n,m(n)}$$

where $m(n) = \omega(\sqrt{n})$, for stationary, ergodic input process $\mathscr{X}$. □

We will need the following Lemma.

Lemma 2.34. *Let $(\mathbb{T}, \mathscr{A})$ be a measurable space, and let $\{\mathsf{Q}_n\}_{n\geq 1}$, Q all be probability measures on this space. Suppose that*

(i) For every $n \geq 1$, there is a set $\mathbb{B}_n \in \mathscr{A}$ such that $\mathsf{Q}_n(\mathbb{A}) = \mathsf{Q}(\mathbb{A})$ for every $\mathbb{A} \subset \mathbb{B}_n$, $\mathbb{A} \in \mathscr{A}$.

(ii) $\mathsf{Q}(\mathbb{B}_n) \to 1$ as $n \to \infty$.

Then the measures Q_n converge in total variation to Q, i.e., $\mathsf{Q}_n \xrightarrow{tv} \mathsf{Q}$ as $n \to \infty$.

Proof. From (ii), for every $\varepsilon > 0$, there exists $n'(\varepsilon) \in \mathbb{N}$ such that

$$\mathsf{Q}(\mathbb{B}_n) \geq 1 - \varepsilon \ \forall\ n \geq n'(\varepsilon).$$

From (i), $\mathsf{Q}_n(\mathbb{A} \cap \mathbb{B}_n) = \mathsf{Q}(\mathbb{A} \cap \mathbb{B}_n)$ for every $n \geq 1$, $\mathbb{A} \in \mathscr{A}$. Therefore, for every $\varepsilon > 0$,

$$\begin{aligned} ||\mathsf{Q}_n - \mathsf{Q}|| &= 2 \sup_{\mathbb{A}\in\mathscr{A}} |\mathsf{Q}_n(\mathbb{A}) - \mathsf{Q}(\mathbb{A})| \\ &= 2 \sup_{\mathbb{A}\in\mathscr{A}} |\mathsf{Q}_n(\mathbb{A} \cap \mathbb{B}_n^{\mathsf{C}}) - \mathsf{Q}(\mathbb{A} \cap \mathbb{B}_n^{\mathsf{C}})| \\ &\leq 2\varepsilon \ \forall\ n \geq n'(\varepsilon). \end{aligned}$$

Hence $\mathsf{Q}_n \xrightarrow{tv} \mathsf{Q}$ as $n \to \infty$. □

Proof of Proposition 2.33. Note that

$$\mathbb{D}_{n,m} \triangleq \left\{ \vartheta \in \mathbb{S} : \max_{i=1}^{N_n^{(m)}(\vartheta)} |Z_i(\vartheta)| \geq m \right\}$$

is the subset of $\mathbb{S}$ in $\mathscr{B}$ where $\mathsf{P}_{\langle m \rangle}(X_{[n]}, Y^{(m)}_{[N_n^{(m)}]})$ differs from $\mathsf{P}(X_{[n]}, Y_{[N_n]})$. From Lemma 2.32, we have

$$\mathsf{P}_{\langle m(n) \rangle}(\mathbb{D}_{n,m(n)}) = \mathsf{P}(\mathbb{D}_{n,m(n)}) \to 0$$

as $n \to \infty$, whenever $m(n) = \omega(\sqrt{n})$. Consider henceforth that $m(n)$ satisfies this condition. In Lemma 2.34 above, set $\mathbb{T} = \mathbb{S}$, $\mathscr{A} = \mathscr{B}$, $\mathsf{Q}_n = \mathsf{P}_{\langle m(n) \rangle}$ and $\mathsf{Q} = \mathsf{P}$. Note that although $\mathsf{P}_{\langle m(n) \rangle}$ is only defined on $\mathscr{F}_{n,m(n)}$ (cf. Proposition 2.25), we can extend it to $\mathscr{B}$ such that it agrees with the measure P on every subset of $\mathbb{B}_n$ for each $n \geq 1$. Then for each $n \in \mathbb{N}$, we see that by setting $\mathbb{B}_n = \mathbb{D}^{\mathsf{C}}_{n,m(n)}$, both conditions (i) and (ii) in Lemma 2.34 are satisfied. From this and [26, Corollary 1′], we have the desired result. □

The channels $\{\mathbf{P}^{\star}_{n,m}\}_{m\geq 0}$ give us a way to approach the problems of optimizing input distributions as well as designing coding schemes for the SDRC. We can optimize the inputs of $\mathbf{P}^{\star}_{n,m}$, starting with small values of m, under some input assumptions, e.g., for fixed-order Markov inputs [57, 116]. Note that the numerical estimation of $I^{\star}_{n,m}$ is possible (as described in the previous subsection) only when $m < n$, since setting the channels as indecomposable FSCs (cf. Lemma 2.28) is possible only in this case. Moreover, for a good estimate of the information rate, we will require $m \ll n$. For the SDRC, Proposition 2.33 allows us to consider some $\mathbf{P}^{\star}_{n,m(n)}$, where $m(n)$ is both $\omega(\sqrt{n})$ as well as $o(n)$, for which a good estimate of the information rate $I^{\star}_{n,m(n)}$ can be obtained. Note that due to the lack of a result analogous to Lemma 2.32 in the case of a general DRC for $m < n$, generalizing these arguments when $p_{\mathsf{d}} \neq p_{\mathsf{r}}$ is not completely justified.

Starting with some small values of m, we expect that the information rates and optimal distributions quickly converge (in m), giving us a way to characterize optimal inputs for the SDRC $\mathbf{P}_n$. For small values of p, as in Figure 2.3, the information rates for the SDRC can be characterized numerically for moderate values of m (much smaller than $\omega(\sqrt{n})$ guaranteed by Lemma 2.32). For optimizing the input distribution for an approximation $\mathbf{P}^{\star}_{n,m}$, we can start with optimizing inputs that are μ^{th}-order Markov processes, for $\mu \geq 1$. As was observed[8] in [117], the convergence of optimal information rates as a function of the order μ of the input Markov process is expected to be rapid. The authors in [117] hypothesized that this convergence was exponential in μ. Similar "diminishing returns" on increasing μ has also been observed by others [57, 116]. We think that a similar rapid convergence of $I^{\star}_{n,m}(\mathscr{X}^{*}_{\mathscr{M}\mu})$ to $C_n(\mathscr{X}^{*}_{\mathscr{M}\mu})$ also holds for m, where $I^{\star}_{n,m}(\mathscr{X}^{*}_{\mathscr{M}\mu})$ is the optimal information rate achieved by a μ^{th}-order Markov input process on the FSC $\mathbf{P}^{\star}_{n,m}$ with $p_{\mathsf{d}} = p_{\mathsf{r}} = p$, and $C_n(\mathscr{X}^{*}_{\mathscr{M}\mu})$ is the optimal information rate achieved by a μ^{th}-order Markov input process on the corresponding SDRC $\mathbf{P}_n$.

Apart from the above advantage of facilitating numerical estimation of information rates, the approximating channels $\mathbf{P}^{\star}_{n,m}$ have another important advantage. This is that since they have immediate factor-graph interpretations, there is a possibility of constructing sparse graph-based coding schemes and decoding over the joint graphical model representing the channels as well as the codes, as was done for joint detection and

[8] Although the validity of the bounds in [117] is unclear (See, e.g., [28]), the rapid convergence of information rates as a function of the order μ of the input Markov process is expected to be true.

decoding of LDPC codes on partial response channels [65]. Instead of trying to build codes for the SDRC $\mathbf{P}_n$, the problem can be reduced to designing good codes and efficient decoding schemes for the FSCs $\mathbf{P}^{\star}_{n,m}$. For small values of the deletion-replication probability p, we can expect these codes to perform well for the SDRC $\mathbf{P}_n$ as well.

2.5 Generalizations

In this section, we discuss different scenarios that can be modeled using the channel model introduced in Section 2.2 with appropriate modifications. Wherever possible, methods for information theoretic analysis for these cases through generalizations of the channel model presented here are highlighted.

2.5.1 Channels introducing Random Insertions

The channel model of Equation (2.5) allows us to handle deletions as well as replications. However, the class of SECs that introduce random insertions cannot be written in the form of Equation (2.5). A suitable modification for our model in this scenario is to let

$$Y_i = X_{i-Z_i} \oplus \mathbb{1}_{\{Z_i = Z_{i-1}+1\}} V_i,$$

where $\mathbb{X} = \mathbb{Y} = \mathbb{V} = \{0,1\}$, and $\mathscr{V} = \{V_i\}_{i \geq 1}$ is a Bernoulli sequence with parameter f (for "flip"). This means that the probability of a random insertion is $f p_{\mathsf{r}}$ and that of a replication is $(1-f)p_{\mathsf{r}}$. Note that this can be easily generalized to any finite sets $\mathbb{X}$, and arbitrary sets $\mathbb{Y}$ and $\mathbb{V}$ (with an appropriate notion of the addition operation "$\oplus$"). Analysis of this channel is, however, more complicated than analyzing the DRC itself due to the cascaded additive noise channel which also depends on the "shared" state process $\mathscr{Z}$. However, when the channel produces deletions, replications, random insertions as well as substitutions, we can write

$$Y_i = X_{i-Z_i} \oplus V_i,$$

which is just a cascade of the DRC and an additive noise channel. In the binary setting, this corresponds to a channel that deletes a bit x with probability p_{d}, or inserts a sequence

$\bar{y}$ with probability $(1-p_{\mathsf{d}})(1-p_{\mathsf{r}})p_{\mathsf{r}}^{|\bar{y}|-1}f^{w(\bar{y}+x)}$. This implies that the substitution error probability for a bit is given by $(1-p_{\mathsf{d}})(1-p_{\mathsf{r}})f$.

The capacity (or the information rate achievable by a given input process) and coding for a cascade of binary, memoryless channels without synchronization errors has been studied in, e.g., [21, 49, 104, 105]. Some lower bounds on the capacity of a cascade of a BDC and an additive noise channel were given in [34, 96]. The possibility of extending these results to general memoryless SECs using the model presented here remains a problem worth exploring.

2.5.2 SECs with Memory

An SEC with memory is defined as in Definition 2.1, with the only difference being that the transition probabilities $q_n(\bar{y}_{[n]}|x_{[n]})$ do not factorize as products of individual probabilities $q(\bar{y}_i|x_i)$. As an example, one could think of an SEC where the deletion of a symbol influences the likelihood of the following symbol being deleted, i.e., a channel that introduces a burst of deletions. Similarly, channels that introduce a limited number of deletions in every input subsequence of a certain length could also occur in practice. These have been studied under the name of *segmented* deletion channels [74, 119]. Note that the definition of the segmented deletion channel is slightly different in the references cited, where it is assumed that the input is divided into blocks of a certain length and at most one deletion occurs within each block. Our definition is more general and corresponds closer to reality, e.g., in the case of a frequency offset between the transmitter and receiver clocks.

The channel model in Equation (2.5) generalizes readily to the case of DRC with memory. Consider the $\mathscr{Z}$ process to be a non-increasing (so that only deletions occur), time-homogeneous, shift-invariant Markov process of order $\mathsf{z} \geq 2$ such that

$$\mathsf{P}(Z_i = z_i | Z_{[i-\mathsf{z}:i-1]} = z_{[i-\mathsf{z}:i-1]}) > 0$$

only for $z_{[i-\mathsf{z}:i-1]}$ such that $z_{i-\mathsf{z}} = z_{i-\mathsf{z}+1} = \cdots = z_{i-j} \geq z_{i-j+1} = \cdots = z_{i-1}$ for some $1 \leq j \leq \mathsf{z}$, and $z_{i-j} - z_{i-j+1} \leq 1$. Then, clearly at most one deletion occurs for every input subsequence of length z. The model in Equation (2.5) will then correspond to a segmented deletion channel where no more than 1 deletion occurs for every z input

symbols. Similarly, we can model other DRC with memory by suitably considering the $\mathscr{Z}$ process to be a Markov process of some order with specific transitions occurring with non-zero probability.

Although we have let $\mathsf{z} \geq 2$, not all second-order Markov processes $\mathscr{Z}$ result in SECs with memory. One example worth noting is when the $\mathscr{Z}$ process is non-decreasing with increments of at most 1, and is such that two consecutive increments occur with probability 0. This results in a replication channel where each symbol is transmitted noiselessly and possibly replicated once—this is exactly the *elementary sticky channel* introduced in [84], which is a memoryless SEC. We will refer to such channels that introduce a bounded number of inserted symbols per input symbol as *bounded*, memoryless SECs. This particular channel has also been studied in [96], where some analytical lower bounds on the capacity were given. Another example where $\mathsf{z} = 2$ does not result in an SEC with memory but is a bounded, memoryless SEC is Gallager's model [37] of the insertion-deletion channel. Some analytical lower bounds for the capacity of this channel (without deletions) were given in [96]. Achievable rates for a bounded, memoryless SEC were studied in [115], and those for a cascade of a bounded, memoryless SEC with an inter-symbol interference (ISI) channel in [43]. Some bounds on the capacity of a bounded, memoryless SEC with substitution errors were given in [36].

Note that the channel coding theorem for SECs with memory has not been established. The various works on the "capacity" of such channels is an indication of such SECs occurring widely in practice. Establishing the channel coding theorem for SECs with memory is, therefore, important both for the theory and in practice. For SECs with memory, since the channel model (2.5) will have the transition probabilities that still factorize as in Equation (2.7) (with the channel state transition probabilities replaced by the higher-order transition probabilities), it is more amenable to analysis and could potentially be used to establish the channel coding theorem.

2.5.3 Jitter, Bit-shift and Grain-error Channels

Channels that consider mis-synchronization due to *jitter* or *bit-shifts* have been studied in the context of magnetic recording and constrained coding [6, 9, 101]. These represent a variant of the general model of the DRC presented here. In particular, they

are characterized by a $\mathscr{Z}$ process where each valid state path $\bar{z} \in \overline{\mathbb{Z}}$ has increments and decrements of size at most 1, and the transition probabilities are data-dependent. The zeros and ones in the input correspond to the absence and presence, respectively, of a transition in the signal. Thus, the presence of a transition cannot be deleted, i.e., a 1 in the input stream cannot be deleted, whereas the 0s can be deleted or replicated (at most once). The authors of [101] gave bounds on the capacity and the zero-error capacity of bit-shift channels and also presented some bounds on achievable rates over a concatenation of the bit-shift channel with a binary symmetric channel. Similar analysis was performed in [9] for discrete and continuous channels with timing jitter. Numerical upper and lower bounds on the capacity of a binary channel with jitter where transitions could "cancel" each other were given in [6].

Another class of channels that resemble these channels are the "paired" insertion-deletion channels studied in the context of bit-patterned media recording [50], which will also be the application studied in the next chapter. Here, the channel is similar to the approximating FSC given in Section 2.4.2 with $m = 1$. In [50], the authors give bounds on the capacity and the zero-error capacity of the channel for varying sizes of the state space. A further specialization of this channel is the one-dimensional *granular media* recording channel. This has also been studied in [82], where some bounds on capacity and coding constructions have been proposed.

2.5.4 Permuting and Trapdoor Channels

The *trapdoor channel* introduced by Blackwell (See [8, §7.1]) is a channel where the input stream is fed to a buffer at the same rate as symbols from within the buffer are randomly drawn as the output stream. Using our model, we can define the trapdoor channel as follows. The multiset of indices of the buffer contents at time $i \geq 1$ is denoted as $B_i = \{\beta_1, \cdots, \beta_b\}$, which is of size b. We initialize

$$B_1 = \{\underbrace{0, \cdots, 0}_{b-1}, 1\}$$

and define the output at the i^{th} instant as $Y_i = X_{\Gamma_i}$ for $i \in [n]$, where Γ_i has the distribution $\mathsf{P}(\Gamma_i = \beta_j) = \frac{1}{b}, 1 \leq j \leq b$. The buffer multiset is updated as $B_{i+1} = B_i \setminus \{\Gamma_i\} \cup \{i+1\}$.

In this case, a further simplification of the channel model might be more useful since the channel depends not on the indices of the inputs in the buffer, but on the *type* [18, §12.1] of the buffer contents at any time. This channel was generalized to define *permuting channels* in [12].

Although the trapdoor channel is described easily, its capacity, even in the simplest case of $|\mathbb{X}| = 2$ and $b = 2$, has been an open problem ever since its introduction. In [1], the authors considered coding schemes for certain non-probabilistic models of the trapdoor channel. The capacity of the probabilistic trapdoor channel when $|\mathbb{X}| = 2$ and $b = 2$ is known to satisfy [93]

$$\frac{1}{2} < C(|\mathbb{X}| = 2, b = 2) \leq \log_2 \frac{1+\sqrt{5}}{2} \approx 0.694241914.$$

It is worthwhile to explore the possibility of obtaining better bounds on the capacity of the trapdoor and permuting channels using the model presented here.

2.5.5 Molecular and Chemical Channels

A simple model for molecular or chemical channels is as follows. The channel state Z_i at time instant i is a random variable on the alphabet $\{0\} \cup [m]$ and represents the delay introduced to the input at time i. The output at time i is given as

$$Y_i = \sum_{\mathsf{z}=0}^{m} X_{i-\mathsf{z}} \mathbb{1}_{\{Z_{i-\mathsf{z}}=\mathsf{z}\}},$$

i.e., the output is the sum of all the channel inputs that arrive at time i. This channel was studied in [19] as a *delay selector channel* and a lower bound on the capacity was given assuming that the state process is i.i.d.. In general, the state process can be modeled as a Markov process, and the channel might be amenable to a similar analysis as presented here.

2.5.6 Timing Channels

There is a link between discrete *timing channels* [4], where information is communicated not only in the signals but also in the timing of the signals and the randomness is in the arrival times of the signals, and "good" transmission sequences for SECs. This

is in the sense that a information-bearing transmission sequence for an SEC must not only be able to carry information within the sequence, but also contain information in the ordering of the symbols within the sequence, such that even in the presence of synchronization errors, the information about the symbol ordering is not completely lost. That is to say that the sequences $X_{[n]}$ must be such that under limited number of synchronization errors, the received sequence $Y_{[N_n]}$ must convey adequate information about the state sequence $Z_{[N_n]}$. Therefore, it might be of importance to study whether methods of coding over timing can be used to obtain efficient codes and decoding schemes for the SECs.

2.6 Conclusions

We introduced a new channel model for a class of SECs which formulated the SEC as a channel with states. This allowed us to obtain analytical lower bounds for the capacity of SECs with only deletions or only replications. For the case of the BDC, we were able to write the SIR in terms of subsequence weights of binary sequences. Subsequence weights are known to be a quantity of interest in the maximum-likelihood decoding of sequences for the BDC (cf. Equation (2.12)). Moreover, it is clear from Equation (2.12) that the dependence of information rates for the BDC on the input statistics only appears in the term $\mathfrak{H}_m^{(i)}$, whereas the subsequence weights influence $\mathsf{H}(\bar{x})$ independently of the input statistics. Thus, our result establishes a natural link between the capacity of the BDC and the metric relevant for ML decoding. We were also able to obtain lower bounds on the capacity of the BDC that are known to be tight for small deletion probabilities. For the BRC, we were able to exactly characterize the Markov-1 rate, which is the first analytic lower bound on the capacity of the BRC. In doing so, we were able to disprove the conjecture that the capacity of SECs is a convex function of the channel parameters, at least in the case of the BRC.

For the case of an SEC with deletions and replications, we were able to provide a sequence of approximating FSCs that are totally ordered with respect to the mutual information rates achievable, and therefore, with respect to capacities. These approximating FSCs were shown to be such that the mutual information rate achievable for the

SEC was equal to the limit of the mutual information rates achievable for the sequence of FSCs. To obtain numerical estimates of achievable rates on the DRC, we defined another sequence of indecomposable FSCs. Computing the mutual information rates for this sequence of FSCs allows us to relate the mutual information rate for the DRC to the limiting value of the mutual information rates of the sequence. For the particular case of the SDRC, we were able to show a stronger uniformity in the convergence of these mutual information rates.

The formulation in this work not only allows us to get estimates of mutual information rates achievable on SECs but also gives some insight into possible code constructions and decoding schemes for such channels. The approximations introduced for the DRC gives us a natural way to reduce these problems. One would therefore obtain progressively better performing codes for the DRC by designing good codes for the sequence of approximating FSCs. We expect that for a small values of the deletion-replication probability, a code constructed for an approximation with a moderate value of m will perform well over the DRC as well. Some coding schemes for special cases of the FSCs (with $m = 1$) have been known in various contexts (See Section 2.5.3). Extending these schemes to better approximations (larger m values) will prove crucial in designing good codes for the DRC.

The present formulation of the SECs also allows us to make the following remarks.

- In [56], the authors conjectured that the capacity of the BDC has a Taylor-like series expansion. We see from Theorem 2.15 that this is true for the SIR of the BDC. We expect that the capacity also has a similar formulation.

- The capacity of a general SEC might not be convex in the channel parameters (See Remark 2.2). It was shown in [20] that the capacity C_{BDC} of the BDC satisfies
$$\inf_{p\in[0,1]} \frac{C_{\mathrm{BDC}}(p)}{1-p} = \lim_{p\to 1} \frac{C_{\mathrm{BDC}}(p)}{1-p}.$$
It is therefore expected that $C_{\mathrm{BDC}}(p)$ is convex in p. From Theorem 2.15, we see that the SIR $C_{\mathrm{BDC}}^{\mathrm{iud}}$ of the BDC can be written as
$$C_{\mathrm{BDC}}^{\mathrm{iud}} = 1-p-h_2(p)+(1-p)\Big(\lim_{i\to\infty}\lim_{\upsilon\to\infty}\sum_{m=0}^{\upsilon} f(i,m,p)\Big),$$

where $f(i,m,p) = \psi_{i,m} p^m (1-p)^i$ is non-convex in p for $m \geq 1$. It is interesting to see if this double limit turns out to be convex despite

$$\sum_{m=0}^{\upsilon} f(i,m,p)$$

being non-convex for every finite $\upsilon \geq 1$. Extending this to the case of the capacity C_{BDC} is also of interest.

- In order to obtain bounds for the capacity of a BDC for p close to 1, one might typically consider the case where all but one (or a few) symbols are lost. The lower bound D_2 presented here (cf. Lemma 2.12) corresponds to this situation. However, since we considered this bound for a first-order Markov input, the bounds we obtained didn't prove to be useful for p close to 1. It might therefore be of interest to generalize this bound for a high-order Markov input which might give us a strictly positive (and thereby non-trivial) achievable rate.

- The approximations of the DRC presented have another useful feature. Via the estimation of the delayed feedback capacity of the approximating channels, tight bounds on the capacity can be obtained [126]. By estimating the capacity for moderate values of m, we can obtain a good estimate of the capacity of the DRC as guaranteed by Propositions 2.31 and 2.33.

Acknowledgement

This chapter contains material from the paper "On the Capacity of Channels with Synchronization Errors", A.R. Iyengar, P. H. Siegel, J. K. Wolf; which is being prepared for submission to the IEEE Transactions on Information Theory. The dissertation author was the primary investigator and author of this paper.

Chapter 3

Synchronization Error Channels in Magnetic Recording

In this chapter, we will focus on some practical scenarios where synchronization error channels arise naturally. We will typically be interested in applications of magnetic recording. However, many communication applications have physical limitations that result in similar channels that are of interest here. Much of the analysis will bear a similar flavor as the concepts introduced in the previous chapter.

Magnetic recording channels are typically modeled as binary-input intersymbol interference (ISI) channels, also called *partial response* channels [53, 60]. An implicit assumption made in these channel models is that the data are correctly written on the disk and that errors occur only during the readback process. While this is a fairly realistic assumption in conventional "low-density" recording technologies on continuous media, it is questionable in the context of certain advanced recording technologies, such as *bit-patterned media* (BPM) recording, that achieve higher storage densities. In this paper, we will examine some of the underlying causes of errors in the recording process, particularly in BPM recording, and then propose a new probabilistic write channel model that captures some of the data dependence of these write errors. Thus, the input to this write channel model is the data sequence to be recorded and the channel output is the "noisy" sequence that actually gets stored on the medium. This leads to a description of the full data recording and readback process as a cascade of an imperfect write channel and a noisy (partial response) readback channel.

Hu, Duman, Kurtas and Erden [44] proposed a model for the BPM recording channel in which the write channel was a binary symmetric channel (BSC), and the readback channel was a linear, intersymbol-interference (ISI) channel with additive noise. They proposed and evaluated detection methods for this channel, and they investigate achievable information rates for such a system. In [49], we considered an idealized cascaded channel model in which the write channel was again a BSC and the readback channel was a memoryless, binary-input, additive white Gaussian noise (AWGN) channel. We studied theoretical properties of this channel, as well as the decodable regions of LDPC codes under a number of decoding algorithms. Here, our focus is on a new write channel model which allows us to determine and compare bounds on several relevant information-theoretic limits: capacity, symmetric information rate (SIR), Markov-1 rate, and zero-error capacity.

The remainder of this chapter is organized as follows. In Section 3.1, we start with a brief description of the write process in bit-patterned media recording, highlighting the main factors leading to write errors. We present the data-dependent write channel model in Section 3.2. Two classes of channel state processes are introduced: a Bernoulli state process (Section 3.2) and a binary Markov state process (Section 3.2). In Section 3.2.1 we make the observation that, although proposed as a model for BPM recording, the Markov state channel model is also relevant to high-density magnetic recording using conventional granular media. In Section 3.3 we give bounds on the capacity, the SIR, and the Markov-1 rate of the Bernoulli state channel. Section 3.4 gives similar bounds on the capacity for the binary Markov state channel. The SIR is numerically computed for both of the channels considered in Section 3.2. In Section 3.5, we introduce a generalization of binary Markov state channels, namely the K-ary Markov state channels. For one such generalized channel, we numerically estimate the SIR and derive bounds on the channel capacity. Finally, in Section 3.6, we explore the zero-error capacity of the proposed class of write channel models. These results bear similar flavor as the information theoretic results in Chapter 2.

3.1 Bit-Patterned Media Recording

A conventional magnetic recording medium is a continuous film of magnetic *grains* that coats the surface of the disk substrate. Each grain is an atomic magnetic unit that assumes one of two possible magnetic states. A group of grains together form a *bit-cell*, an entity that stores one bit of information. Therefore, as the areal information density is increased, the number of grains forming a bit-cell reduces. One of the problems with high-density magnetic recording in conventional media is the *super-paramagnetic effect*, wherein the magnetic states of individual grains change due to the influence of neighboring grains or due to changes in temperature. When the areal information density is increased to point where there are only a few grains per bit, such uncontrolled changes in the magnetic states of grains are detrimental to reliable information storage.

Bit-patterned media recording (BPMR) proposes to get around this problem by making use of patterned magnetic islands separated by non-magnetic material [120]. However, this new structure of the magnetic medium introduces technical challenges not seen in recording on conventional media. An immediate requirement of this media structure is near-perfect synchronization of the write process to ensure that the write head is positioned correctly over the magnetic islands on this disk, i.e., to ensure that the head is positioned within the so-called *writing window zone* of the islands [77]. Assuming that this write synchronization is achieved through the use of timing synchronization patterns, there is a possibility of frequency and/or phase mismatch leading to incorrect writing. Furthermore, even without a timing mismatch, imperfections in the configuration of the patterned magnetic islands may cause writing errors. Finally, as in conventional magnetic recording, the *switching field distribution* of magnetic grains may contribute to errors in the write process. We will refer to write errors induced by any of these mechanisms as *written-in errors*.

Another important feature of BPMR is the geometry of the writing process. Along the *down-track* direction, the span of influence of the magnetic write field is typically larger than the spacing between the islands. Therefore, at any given time, the write head influences multiple adjacent islands. So, in the process of recording a bit on a specified island, the bit value is also recorded on a certain number of subsequent islands, with these islands themselves being overwritten in the future by subsequent bits.

Figure 3.1 gives an illustration of this idealized write process. We refer to the number of

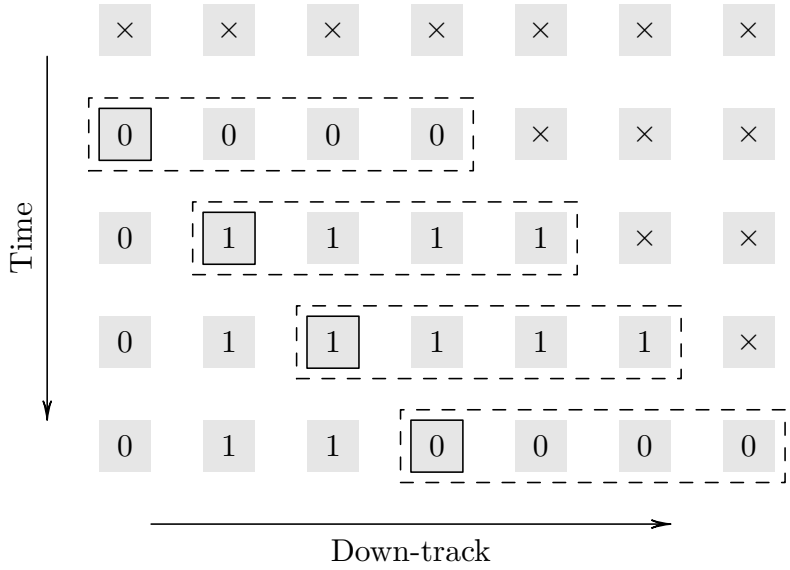

Figure 3.1: Example illustrating writing one bit on each island in BPMR with a write head of writing span $D = 3$. Depicted here are snapshots of the top view of the disk and write mechanism. Initially, all islands, represented here using gray squares, have unknown magnetic states represented by $\times$, which could be 0 or 1. At the first time instant, the write head, shown here using a dashed rectangle, is positioned over the first four islands. In this position, the first island is written with the data bit 0. The three subsequent islands are also written with the first data bit 0, before being overwritten with their own data. After each bit is written, the head is moved over to the next island along the down-track direction. At each position of the write head, the corresponding "current" island is highlighted with a black square boundary.

subsequent islands influenced by the write head as the *writing span*, D. We will assume throughout this paper that $D \geq 1$, which implies that write process has memory and, as a consequence, the write channel is data-dependent.

3.2 Write Channel Model

Representing the channel input and output at the i^{th} time instant as X_i and Y_i, respectively, defined over alphabets $\mathbb{X} = \mathbb{Y} = \{0, 1\}$, the write channel model can be written as

$$Y_i = X_i \oplus (X_i \oplus X_{i-1}) \otimes Z_i, \quad i \in [n] \tag{3.1}$$

where $Z_i \in \{0,1\}$ denotes the channel state at the i^{th} instant. Here, we assume that the state process $\mathscr{Z}$ is independent of the input process $\mathscr{X}$, and $\oplus$ and $\otimes$ represent addition and multiplication in $GF(2)$, respectively. The channel state Z_i is not to be confused with the magnetic state of the island.

It is natural to assume that the channel input and output alphabets are binary since the (intended and actual) magnetic states of the islands can be one of two possible states. The channel state Z_i is a random variable that represents a failure in writing due to one of the conditions mentioned in the previous section, i.e., when $Z_i = 1$ the write head can be assumed to have failed in writing the intended bit. However, this does not necessarily imply a written-in error because there would be no error if the bit to be written is the same as the existing magnetic state of the island. This is captured by the term $(X_i \oplus X_{i-1})$ that is multiplied with Z_i in (3.1). This "noise" term $(X_i \oplus X_{i-1}) \otimes Z_i$ is justified because we assume that the timing mismatch or the irregularity of island patterns can cause the write head to be positioned, in the worst case, on the island immediately following the correct island. We will continue with this assumption until Section 3.5, where we construct a more general write channel model. Also note that when the first bit is written late, i.e. when $Z_1 = 1$, we have $Y_1 = X_0$ which represents the pre-existing magnetic state of the first island. We will assume that X_0 is equally likely to be a 0 or a 1. Similarly, when the last bit is written late, the last island has a bit in error if the last and the penultimate bits are different. In either case, n magnetic islands are read and their contents are interpreted as the n data bits so that there is no blocklength mismatch.

Based on the Z_i sequence, the channel in (3.1) can be seen as producing different types of errors. When the Z_i sequence consists of isolated ones, the channel appears to produce *substitution* errors. This is illustrated in the Figure 3.2. Such substitution-like errors can occur when the write head misses islands at random and independently of its success in writing on previous islands. Noting that when $Z_i = 0$, $Y_i = X_i$ so that the output reproduces the input exactly, and when $Z_i = 1$, $Y_i = X_{i-1}$ so that the output reproduces the input with a delay of one time instant, we can see that another way to write the relation in (3.1) is

$$Y_i = X_{i-Z_i}, \quad i \in [n], \tag{3.2}$$

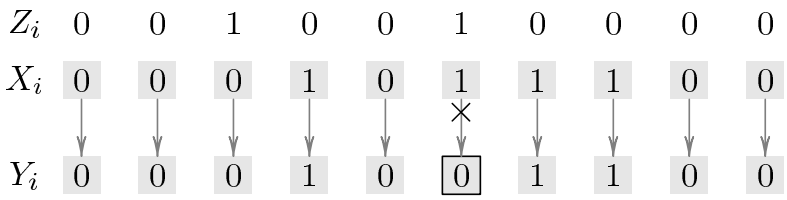

Figure 3.2: An example of substitution-like errors produced by the write channel. The sequence Z_i gives the channel state for each magnetic island. In this example, $Z_i = 1$ for the third and the sixth islands. The sequence X_i gives the intended magnetization of the islands, i.e., the data to be written. Taking into account the channel states Z_i, the sequence Y_i shows the resulting island magnetizations. Note that substitution-like errors occur only when $Z_i = 1$, as was the case with the sixth island here, highlighted with a box around the island in the Y_i sequence. However, not every $Z_i = 1$ results in an error, as illustrated by the third island.

which is exactly the same as the model for the DRC (See Chapter 2) in Equation (2.5). The main difference in the present scenario is the space of the channel state process $\mathscr{Z}$. Since the state space here is finite, unlike the DRC considered in Chapter 2, it suffices to consider outputs for time instants 1 through n in Equation (3.2). As argued for the finite-state channel approximations in Section 2.4, we have that this assumption does not influence the mutual information rates.

When the Z_i sequence consists of long runs of ones, the channel appears to produce "paired" deletion-replication errors, with replications accompanying $0 \to 1$ channel state transitions and deletions accompanying $1 \to 0$ channel state transitions, as shown in Figure 3.3. These deletion-replication errors can happen as a result of timing synchro-

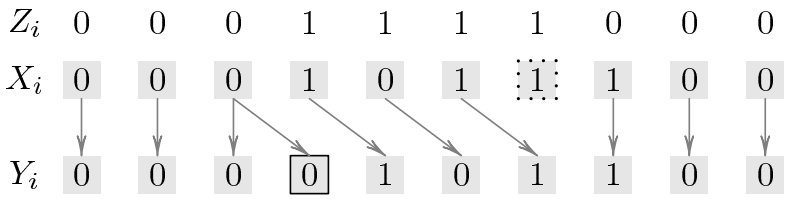

Figure 3.3: An example of an deletion-replication error. Due to the channel state sequence Z_i, the channel inputs are transformed as shown by the arrows between the X_i and Y_i sequences with the resulting magnetic states of the islands shown in the sequence Y_i. The inserted bit in the Y_i sequence accompanying the channel state transition $0 \to 1$ is shown with a box around the corresponding island. The deleted bit from the X_i sequence accompanying the channel state transition $1 \to 0$ is shown with a dotted box around the island.

nization errors, wherein there is a frequency mismatch between the islands and the write head; or as a result of a group of islands being separated farther than usual or having larger switching fields. We will consider the channel under two different statistical assumptions on the $\mathscr{Z}$ process in the following., and show how this difference in statistics changes the typical behavior of the channel.

Bernoulli state channel

When the channel state process $\mathscr{Z}$ is an i.i.d. Bernoulli process with parameter p, i.e., $\mathsf{P}(Z_i = 1) = p$, we will call the channel the *Bernoulli state channel*. In this case, the channel is completely specified by the parameter p. We shall henceforth denote the channel in Equation (3.1) (or (3.2)) by $\mathbf{W}_n(p)$. Typically, for small values of the parameter p we can expect this channel to produce errors resembling substitution errors (See Figure 3.2).

Binary Markov state channel

When the channel state process $\mathscr{Z}$ is a first-order binary Markov processwith $\mathsf{P}(Z_i = 0|Z_{i-1} = 1) = p_{\mathsf{d}}$ and $\mathsf{P}(Z_i = 1|Z_{i-1} = 0) = p_{\mathsf{r}}$, we will refer to the channel as the *binary Markov state channel* and denote it by $\mathbf{W}_n(p_{\mathsf{d}}, p_{\mathsf{r}})$. As noted earlier, we can expect such a channel to typically produce paired deletion-replication errors (See Figure 3.3). Hence, the parameters p_{d} and p_{r} can be thought of as deletion and replication probabilities, respectively, of the channel. However, this channel is different from the DRC in that the state space is finite and the state transitions are Markov as described above. In particular, this Markov process is distinct from the FSC approximations to the DRC (Section 2.4) as well.

3.2.1 High-density recording with granular media

As mentioned in Section 2.5, the channel model in Equation (3.2) can also be used to describe high-density magnetic recording on conventional granular magnetic media [124]. Although media granularity is typically considered as a two-dimensional phenomenon, we consider granularity only in one-dimension – along the down-track

direction, and assume adjacent tracks to be independent. This simplification allows us to establish lower bounds for performance over the two-dimensional channel as proposed in [124].

Media granularity in one-dimension results in written-in errors as follows. At storage densities of the order of 1 bit per grain, the variation in grain size plays an important role in deciding the reliability of data storage. In this case, bit-cells are at most as large as individual grains. We assume that the grains are all 1 or 2 bit-cells in size, and that the magnetic state of each grain is decided by the last bit written on them. This is depicted in Figure 3.4. Grains that are 2 bit-cells large will result in written-in

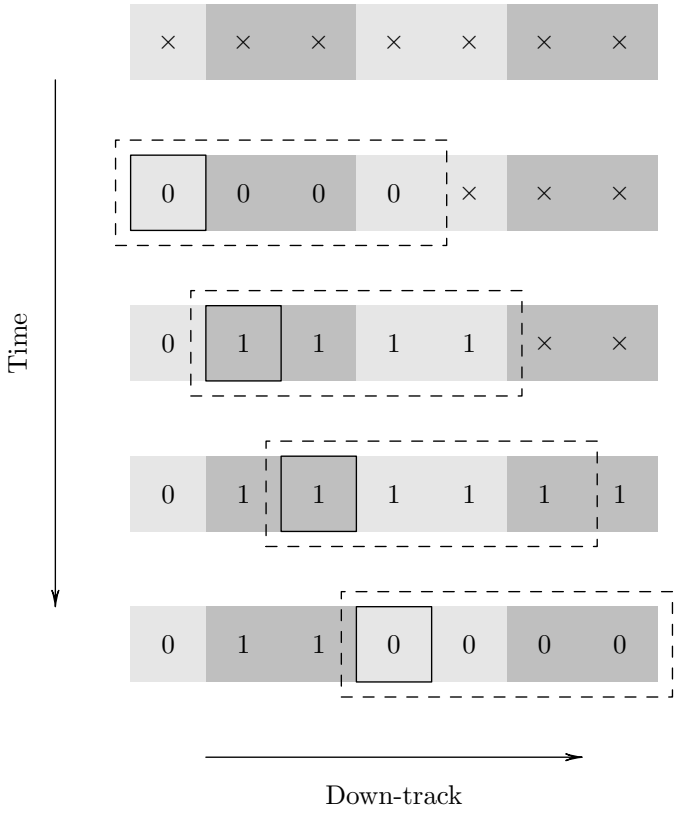

Figure 3.4: Writing on granular media. Individual bit-cells are shown as $\times$'s in the first row, corresponding to the magnetic states read from these cells, which could be 0's or 1's. Also shown in shades of gray are the magnetic grains: grains that comprise one bit-cell are shown in light gray and those that comprise two bit-cells are in a darker shade of gray. The two bit-cells comprising a grain of size 2 always have the same magnetic state. As in the case of BPMR, the write head spans multiple bit-cells, as shown by the dashed rectangles. The "current" bit-cell at each time instant is shown with a black square.

errors if the two bits written on them are different. Figure 3.5 gives an illustration of the written-in errors in this scenario. Comparing Figures 3.3 and 3.5, it is easy to see that

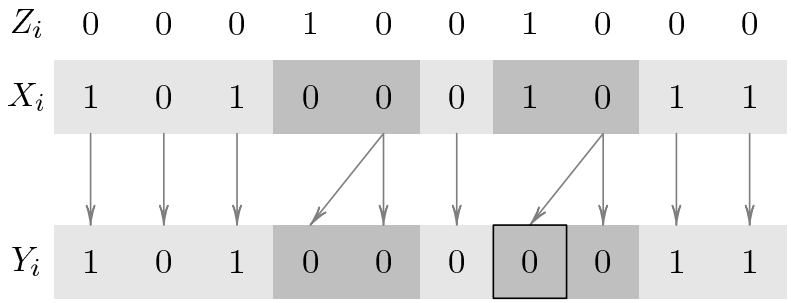

Figure 3.5: Written-in errors due to media granularity. The $\mathscr{Z}$ process represents the grain pattern with Z_i being 1 when the corresponding bit-cell is the first bit-cell in a grain that comprises two bit-cells. The X_i sequence shows the data to be written. The arrows between the X_i and Y_i sequences show the transformation of information according to the grain pattern, resulting in the sequence Y_i being stored. The written-in errors are shown with a box around the corresponding bit-cell.

the channel model in this case can be written as

$$Y_i = X_i \oplus (X_i \oplus X_{i+1}) \otimes Z_i.$$

Using a simple time-reversal argument, it can be seen that this is exactly the channel in (3.1). However, in this case, the Z_i sequence cannot have two consecutive ones, i.e., it satisfies the $(1,\infty)$ run-length constraint [46, Chap. 4]. We choose to model this as the $\mathbf{W}_n(1, p_{\mathsf{r}})$ channel since any realization of the state process for this channel satisfies the $(1,\infty)$ run-length constraint. In [83], the authors consider this channel and derive upper bounds for the achievable rates over the channel. Our focus here will be on obtaining bounds on the achievable information rates for the general channel in (3.1) in the context of the two channel state processes defined in Section 3.2.

3.3 Bernoulli State Channel

The channel space for the Bernoulli state channel defined in the previous section is parameterized by $p \in [0,1]$. As in Section 2.4.2, we can show that $\mathbf{W}_n(p)$ is an

indecomposable FSC, and therefore, its capacity $C(p)$ is given as

$$C(p) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I\left(X_{[n]}; Y_{[n]}\right).$$

The following symmetry result is straightforward.

Proposition 3.1 (Channel symmetry)**.** *For $p \in [0,1]$, $C(p) = C(1-p)$.*

Proof. Let us denote by $I_p(X_1^n; Y_1^n)$ the mutual information between the vectors X_1^n and Y_1^n when the channel parameter is p. Note that the channel output can be simultaneously written as

$$Y_i = (X_i \otimes Z_i^{\mathsf{C}}) \oplus (X_{i-1} \otimes Z_i)$$

and

$$Y_i = (S_i \otimes Z_i) \oplus (S_{i+1} \otimes Z_i^{\mathsf{C}})$$

where $S_i = X_{i-1}$ and $Z_i^{\mathsf{C}} = Z_i \oplus 1$. Thus

$$I_p(X_i; Y_{[n]} | X_{[i-1]}) = I_{1-p}(S_{n-i+1}; Y_{[n]} | S_{[n-i+2:n]})$$

for all but a vanishing fraction of indices $i \in \mathbb{N}$, as $n \to \infty$. Therefore,

$$\begin{aligned}
C(p) &= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_p(X_{[n]}; Y_{[n]}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} \sum_{i=1}^{n} I_p(X_i; Y_{[n]} | X_{[i-1]}) \\
&= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} \sum_{i=1}^{n} I_{1-p}(S_{n-i+1}; Y_{[n]} | S_{[n-i+2:n]}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{1-p}(S_{[n]}; Y_{[n]}) \\
&= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{1-p}(X_{[n-1]}; Y_{[n]}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{1-p}(X_{[n]}; Y_{[n]}) = C(1-p).
\end{aligned}$$

The channel space can therefore be reduced to the interval $p \in [0, \frac{1}{2}]$. Further note that the same symmetry argument holds for not just the rate-maximizing input distribution, but for all input distributions. □

The capacity of the Bernoulli state channel is upper bounded by the achievable rate for a genie-aided decoder, i.e., one with the $\mathscr{Z}$ process realization known. Given the realization of the $\mathscr{Z}$ process, the inserted bits and the positions of the deleted bits are known so that the Bernoulli state channel is equivalent to a correlated erasure channel with average erasure rate $\mathsf{P}(Z_{i-1} = 1, Z_i = 0) = p(1-p)$. The resulting erasure

channel is a correlated channel since two consecutive bits cannot be erased as erasures correspond to $1 \to 0$ transitions in the $\mathscr{Z}$ process. Therefore,

$$C(p) \leq 1 - p(1-p) \triangleq C_{g\varepsilon}(p) \tag{3.3}$$

since the capacity of a correlated erasure channel is the same as that of a memoryless erasure channel with the same erasure probability. We call this upper bound, $C_{g\varepsilon}(p)$, the *genie-erasure capacity* of the channel $\mathbf{W}_n(p)$.

Consider

$$\begin{aligned}\frac{1}{n}I\left(X_{[n]};Y_{[n]}\right) &= \frac{1}{n}H(Y_{[n]}) - \frac{1}{n}H(Y_{[n]}|X_{[n]}) \overset{(a)}{=} \frac{1}{n}H(Y_{[n]}) - \frac{1}{n}\sum_{i=1}^{n} H(Y_i|X_{[i-1:i]}) \\ &= \frac{1}{n}H(Y_{[n]}) - \frac{h_2(p)}{n}\sum_{i=1}^{n} \mathsf{P}(X_i \neq X_{i-1}),\end{aligned} \tag{3.4}$$

where the equality labeled (a) follows from the definition of Y_i in (3.1) and the fact that $\mathscr{Z}$ is i.i.d.. Since the information-rate-maximizing input distribution is unknown, we will now derive lower bounds to the capacity by making certain assumptions about the statistics of the input process $\mathscr{X}$.

3.3.1 Memoryless Input Process

We will first assume that the input process $\mathscr{X}$ is an i.i.d. Bernoulli process with parameter α process. With this assumption, the maximum achievable information rate, called the *i.i.d. capacity*, denoted $C^{\text{iid}}(\alpha, p)$, gives a lower bound to the capacity.

Proposition 3.2 (Input symmetry)**.** $C^{\text{iid}}(\alpha, p) = C^{\text{iid}}(1-\alpha, p)$.

Proof. Let $X_{[n]}$ and $Y_{[n]}$ be the input and output respectively of the channel $\mathbf{W}_n(p)$. Define $\hat{X}_{[n]} = (X_1 \oplus 1, X_2 \oplus 1, \cdots, X_n \oplus 1)$ and $\hat{Y}_{[n]} = (Y_1 \oplus 1, Y_2 \oplus 1, \cdots, Y_n \oplus 1)$. When $X_{[n]}$ is Bernoulli with parameter α, $\hat{X}_{[n]}$ is Bernoulli with parameter $1-\alpha$. Further since $X_{[n]} \leftrightarrow \hat{X}_{[n]}$ and $Y_{[n]} \leftrightarrow \hat{Y}_{[n]}$ are bijections, we have $I_p(X_{[n]};Y_{[n]}) = I(\hat{X}_{[n]};\hat{Y}_{[n]})$. Clearly, $\hat{X}_{[n]}$ and $\hat{Y}_{[n]}$ also satisfy the relation in (3.1) and consequently $I(\hat{X}_{[n]};\hat{Y}_{[n]}) = I_p(\hat{X}_{[n]};\hat{Y}_{[n]})$. □

As a consequence of Propositions 3.1 and 3.2, we have

$$C^{\text{iid}}(\alpha, p) = C^{\text{iid}}(\alpha, 1-p) = C^{\text{iid}}(1-\alpha, 1-p) = C^{\text{iid}}(1-\alpha, p).$$

From Proposition 3.2 and the fact that $C^{\text{iid}}(\alpha, p)$ is concave in α, we immediately have the following.

Corollary 3.3 (Rate-maximizing i.i.d. distribution)**.** *For the Bernoulli state channel* $\mathbf{W}_n(p)$ *with memoryless inputs*

$$\max_{\alpha\in[0,1]} C^{\text{iid}}(\alpha, p) = C^{\text{iid}}(\frac{1}{2}, p) = C^{\text{iud}}(p) \;\forall\; p \in [0,1],$$

where $C^{\text{iud}}(p)$ *is the* symmetric information rate *(SIR) of* $\mathbf{W}_n(p)$. □

We have from (3.4)

$$C(p) \geq C^{\text{iud}}(p) = \mathscr{H}(\mathscr{Y})\Big|_{\mathscr{X}\sim\text{iud}} - \frac{h_2(p)}{2} \tag{3.5}$$

We can lower bound the SIR by disregarding the data-dependence of the noise in the channel. This gives a channel equivalent to a BSC with crossover probability $\frac{p}{2}$ so that

$$C^{\text{iud}}(p) \geq 1 - h_2\left(\frac{p}{2}\right) \triangleq L_0^{\text{iud}}(p). \tag{3.6}$$

Further lower bounds can be obtained by conditioning the entropy of the output as follows.

$$\begin{aligned} C^{\text{iud}}(p) &= \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} H(Y_i|Y_{[i-1]}) - \frac{h_2(p)}{2} \geq \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} H(Y_i|Y_{[i-1]}, X_{i-1}) - \frac{h_2(p)}{2} \\ &= h_2\left(\frac{1-p}{2}\right) - \frac{h_2(p)}{2} \triangleq L_1^{\text{iud}}(p), \end{aligned} \tag{3.7}$$

where we have used the fact that Y_i depends only on X_{i-1} and X_i, and given X_{i-1}, Y_i is independent of Y_1^{i-1}. Continuing as above, we can obtain a tighter lower bound

$$\begin{aligned} C^{\text{iud}}(p) &\geq \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} H(Y_i|Y_{[i-1]}, X_{i-2}) - \frac{h_2(p)}{2} \\ &= \left(\frac{1+p}{2}\right) h_2\left(\frac{1+p^2}{2(1+p)}\right) + \left(\frac{1-p}{2}\right) h_2\left(\frac{1-p}{2}\right) - \frac{h_2(p)}{2} \triangleq L_2^{\text{iud}}(p). \end{aligned} \tag{3.8}$$

A straightforward upper bound for the SIR, implied by (3.5) is

$$C^{\text{iud}}(p) \leq 1 - \frac{h_2(p)}{2} \triangleq U_0^{\text{iud}}(p), \tag{3.9}$$

which follows because the entropy rate for a binary process $\mathscr{H}(\mathscr{Y}) \leq 1$. Note that this bound is achieved when $\mathscr{Y}$ is the i.u.d.. Again, starting from (3.5), we obtain upper bounds for the SIR by removing conditioning from the entropy of the output.

$$\begin{aligned} C^{\text{iud}}(p) &\leq \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} H(Y_i|Y_{i-1}) - \frac{h_2(p)}{2} \\ &= h_2\left(\frac{1-p+p^2}{2}\right) - \frac{h_2(p)}{2} \triangleq U_1^{\text{iud}}(p). \end{aligned} \tag{3.10}$$

As with the lower bounds, we can find a tighter upper bound for the entropy rate $\mathscr{H}(\mathscr{Y})$ as follows

$$\begin{aligned} C^{\text{iud}}(p) &\leq \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} H(Y_i|Y_{[i-2:i-1]}) - \frac{h_2(p)}{2} \\ &= \frac{1-p+p^2}{2} h_2\left(\frac{1}{2(1-p+p^2)}\right) \\ &\quad + \frac{1+p-p^2}{2} h_2\left(\frac{1}{2(1+p-p^2)}\right) - \frac{h_2(p)}{2} \triangleq U_2^{\text{iud}}(p). \end{aligned} \tag{3.11}$$

Figure 3.6 plots the lower and the upper bounds for SIR discussed above. Note that the tighter lower and upper bounds — $L_2^{\text{iud}}(p)$ in (3.8) and $U_2^{\text{iud}}(p)$ in (3.11) — almost coincide for $p \leq 0.3$ (and from symmetry, for $p \geq 0.7$). In this range, therefore, where the bounds themselves are greater than 0.5, they approximate the SIR fairly accurately. As for the channels in Section 2.4.2, the SIR can be numerically computed for the Bernoulli state channel. This is shown as the "SIR" curve in Figure 3.6. This indicates that the upper bound $U_2^{\text{iud}}(p)$ in (3.11) is a good approximation for the SIR.

3.3.2 First-order Markov Input process

The input symmetry of the Bernoulli state channel (Proposition 3.2) can be shown to be true for stationary ergodic input processes in general. Therefore, we consider a *symmetric* binary Markov process $\mathscr{X}$ with $\mathsf{P}(X_i = 1|X_{i-1} = 0) = \mathsf{P}(X_i = 0|X_{i-1} = 1) = \beta$.

Starting from (3.4), we can arrive at lower and upper bounds for the *Symmetric Markov*-1 *Rate* (M1R), $C^{\mathscr{M}1}(\beta, p)$.The lower bounds analogous to those for the SIR are

$$C^{\mathscr{M}1}(\beta, p) \geq 1 - h_2\left(\frac{p}{2}\right) \triangleq L_0^{\mathscr{M}1}(p),$$

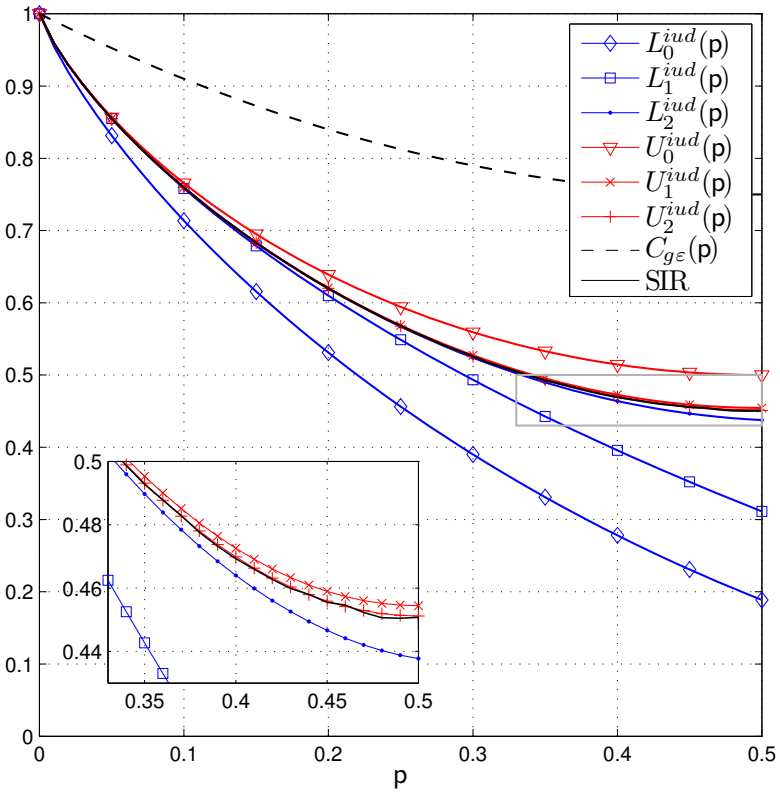

Figure 3.6: Lower and upper bounds for SIR given in (3.6), (3.7), (3.8), (3.9), (3.10) and (3.11), and the upper bound for capacity in (3.3). The SIR from simulations is also shown.

which is the same as $L_0^{\text{iud}}(p)$ in (3.6),

$$C^{\mathscr{M}1}(\beta,p) = \mathscr{H}(\mathscr{Y})\Big|_{\mathscr{X}\sim\mathscr{M}_1^{(2)}(\beta)} - \beta h_2(p) \geq \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n} H(Y_i|Y_{[i-1]},X_{i-1}) - \beta h_2(p)$$
$$= h_2(\beta(1-p)) - \beta h_2(p) \triangleq L_1^{\mathscr{M}1}(\beta,p),$$

and

$$C^{\mathscr{M}1}(\beta,p) \geq \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n} H(Y_i|Y_{[i-1]},X_{i-2}) - \beta h_2(p)$$
$$= (1-\beta(1-p))\, h_2\left(\frac{\beta\left(1-\beta(1-p^2)\right)}{1-\beta(1-p)}\right)$$
$$+\beta(1-p)h_2\left(\beta(1-p)\right) - \beta h_2(p) \triangleq L_2^{\mathscr{M}1}(\beta,p). \tag{3.12}$$

The trivial upper bound analogous to $U_0^{\text{iud}}(p)$ in (3.9) is

$$C^{\mathscr{M}1}(\beta,p) \le 1-\beta h_2(p) \triangleq U_0^{\mathscr{M}1}(\beta,p).$$

The upper bounds corresponding to U_1^{iud} and U_2^{iud} are

$$C^{\mathscr{M}1}(\beta,p) \le h_2\left(1-\beta+2\beta^2 p(1-p)\right) - \beta h_2(p) \triangleq U_1^{\mathscr{M}1}(\beta,p)$$

and

$$\begin{aligned} C^{\mathscr{M}1}(\beta,p) \le{}& \left(\beta^2(1-\beta)(1-p+p^2)+\beta^2(3-\beta)p(1-p)+(1+\beta)(1-\beta)^2\right) \\ &\times h_2\left(\frac{\beta^2(1-\beta)(1-p+p^2)+\beta^3 p(1-p)+\beta(1-\beta)^2}{\beta^2(1-\beta)(1-p+p^2)+\beta^2(3-\beta)p(1-p)+(1+\beta)(1-\beta)^2}\right) \\ &+\left(2\beta^2(1-\beta)(1-p+p^2)+\beta^3(1-2p+2p^2)+\beta(1-\beta)^2\right) \\ &\times h_2\left(\frac{\beta^2(1-\beta)(1-p+p^2)+\beta^3 p(1-p)+\beta(1-\beta)^2}{2\beta^2(1-\beta)(1-p+p^2)+\beta^3(1-2p+2p^2)+\beta(1-\beta)^2}\right) \\ &-\beta h_2(p) \triangleq U_2^{\mathscr{M}1}(\beta,p). \end{aligned} \tag{3.13}$$

Figure 3.7 shows the contours of the bounds for $C^{\mathscr{M}1}$ in (3.12) and (3.13). (Only the tighter bounds, $L_2^{\mathscr{M}1}(\beta,p)$ and $U_2^{\mathscr{M}1}(\beta,p)$ are shown.) As was the case for i.u.d. input, the tighter bounds (3.12) and (3.13) have almost coinciding contours for a wide range of parameters (β,p). Unlike the case of i.i.d. inputs, the rate-maximizing input parameter $\beta^*(p)$ is not easily obtained. A close estimate can be obtained by maximizing the bounds obtained above. These are shown (dashed lines) in Figure 3.7. Since the optimal β values for the tighter lower and upper bounds $L_2^{\mathscr{M}1}(\beta,p)$ and $U_2^{\mathscr{M}1}(\beta,p)$ almost coincide, we can say that $\beta^*(p)$ is monotonically decreasing in p in the interval $[0,\frac{1}{2}]$ with $\beta^*(0)=\frac{1}{2}$ and $\beta^*(1/2)\approx 0.31$.

Figure 3.8 compares the SIR (solid line representing $L_2^{\text{iud}}(p)$ in (3.8), and dashed line representing $U_2^{\text{iud}}(p)$ in (3.11)) and the M1R (solid line for $L_2^{\mathscr{M}1}(\beta_{L_2}^*(p),p)$ in (3.12), dashed line for $U_2^{\mathscr{M}1}(\beta_{U_2}^*(p),p)$ in (3.13)) over the range of p values. It is clear that considerable gains in reliable information transfer rate are possible by using an input with memory. In particular, note that whereas there is a range of p values for which the SIR is smaller than 0.5, the M1R is strictly larger than 0.5 $\forall$ p. It is clear that for a sequence of input Markov processes of increasing orders, the achievable rates are non-decreasing. The algorithm suggested in [57, 116] can be employed to optimize the input Markov process to maximize the rate.

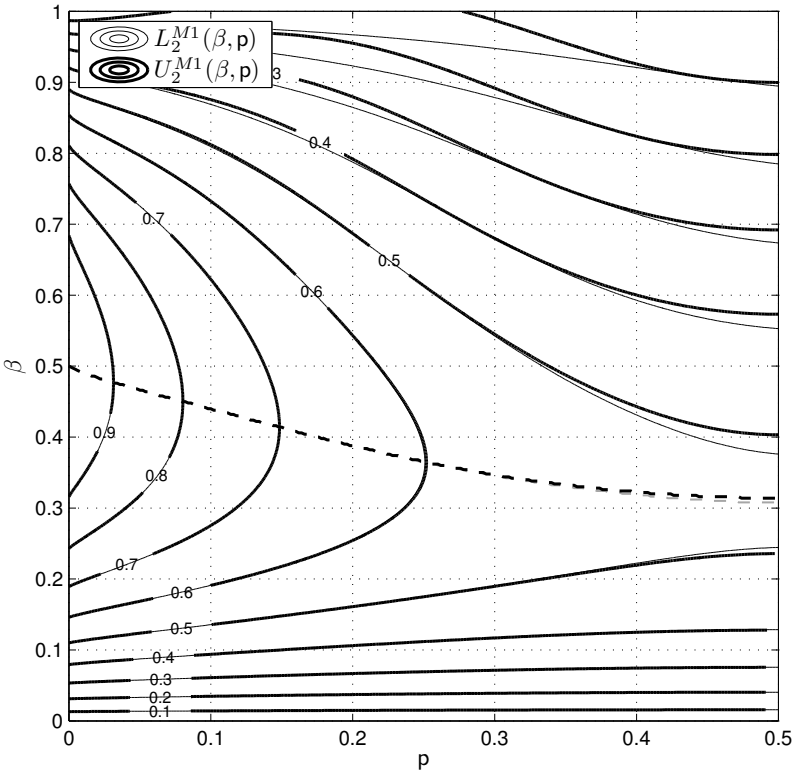

Figure 3.7: Contours of lower (thin curves) and upper (thick curves) bounds for M1R given in (3.12) and (3.13).

3.4 Binary Markov State Channel

The channel space for the binary Markov state channel defined in Section 3.2 is $(p_{\mathsf{d}}, p_{\mathsf{r}}) \in [0,1]^2$. Note that the channel space is ordered, i.e., the first parameter is the $1 \to 0$ transition (deletion) probability and the second the $0 \to 1$ transition (replication) probability. As in the case of the Bernoulli state channel, the capacity of the binary Markov state channel, denoted $C(p_{\mathsf{d}}, p_{\mathsf{r}})$, is given as

$$C(p_{\mathsf{d}}, p_{\mathsf{r}}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{(p_{\mathsf{d}},p_{\mathsf{r}})}(X_{[n]}; Y_{[n]}).$$

When p_{r} (resp., p_{d}) is zero, the channel is the noise-free channel (resp., noise-free channel with a delay) and hence $C(p_{\mathsf{d}}, 0) = 1$ (resp., $C(0, p_{\mathsf{r}}) = 1$). We first establish the following symmetry property of the binary Markov state channel.

Proposition 3.4 (Channel symmetry)**.** $C(p_{\mathsf{r}}, p_{\mathsf{d}}) = C(p_{\mathsf{d}}, p_{\mathsf{r}})$.

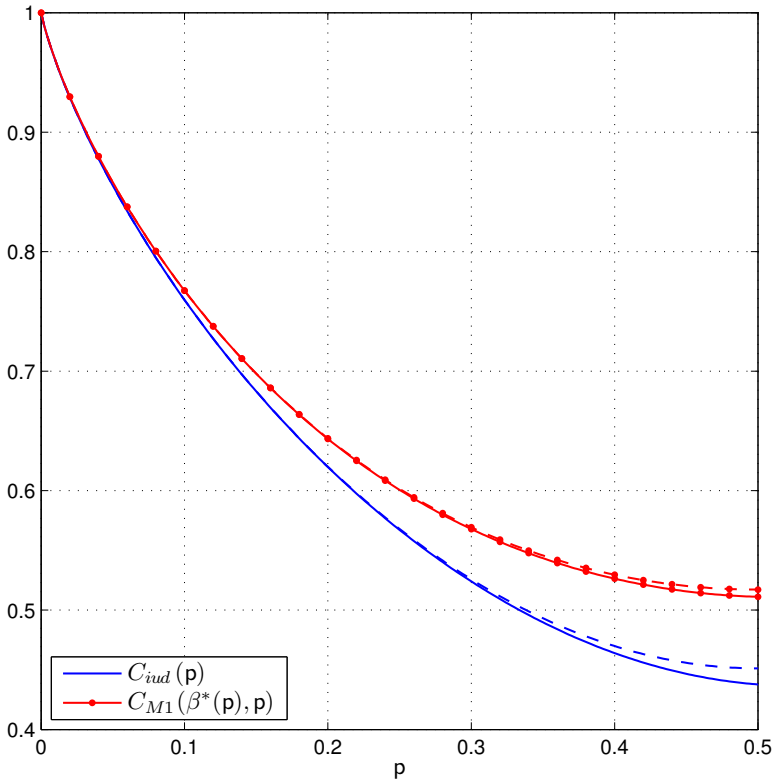

Figure 3.8: Comparison between the tighter lower and upper bounds for SIR and M1R.

Proof. We know that if $Z_{[n]} = (Z_1, Z_2, \cdots, Z_n)$ is a Markov process, then so is the time-reversed vector $(Z_n, Z_{n-1}, \cdots, Z_1)$ [91, §16-4]. Furthermore, since the channel is assumed to have converged to the stationary distribution, the conditional distributions $\mathsf{P}(Z_i|Z_{i-1})$ and $\mathsf{P}(Z_i|Z_{i+1})$ are identical (from the symmetry in the Markov process $\mathscr{Z}$). However, note that whereas a transition $Z_{i-1} = 0 \to Z_i = 1$ is a replication, the transition $Z_{i+1} = 0 \to Z_i = 1$ is a deletion. This implies that $I_{(p_\mathsf{d},p_\mathsf{r})}(X_i; Y_{[n]}|X_{[i-1]}) = I_{(p_\mathsf{r},p_\mathsf{d})}(S_{n-i+1}; Y_{[n]}|S_{[n-i+2:n]})$, where $S_i = X_{i-1}$, for all but a vanishing fraction of indices i, as $n \to \infty$. Therefore,

$$C(p_\mathsf{d}, p_\mathsf{r}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{(p_\mathsf{d},p_\mathsf{r})}(X_{[n]}; Y_{[n]}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} \sum_{i=1}^{n} I_{(p_\mathsf{d},p_\mathsf{r})}(X_i; Y_{[n]}|X_{[i-1]})$$

$$= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} \sum_{i=1}^{n} I_{(p_\mathsf{r},p_\mathsf{d})}(S_{n-i+1}; Y_{[n]}|S_{[n-i+2:n]})$$

$$= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{(p_\mathsf{r},p_\mathsf{d})}(S_{[n]};Y_{[n]}) = \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{(p_\mathsf{r},p_\mathsf{d})}(X_{[n-1]};Y_{[n]})$$
$$= \lim_{n\to\infty} \sup_{\mathsf{P}(X_{[n]})} \frac{1}{n} I_{(p_\mathsf{r},p_\mathsf{d})}(X_{[n]};Y_{[n]}) = C(p_\mathsf{r},p_\mathsf{d}).$$

The channel space can therefore be reduced to the region $p_\mathsf{r} \in [0,1], p_\mathsf{d} \in [0,p_\mathsf{r}]$. As was the case for the $\mathbf{W}_n(p)$ channel, we have the above $p_\mathsf{d}, p_\mathsf{r}$ symmetry for any fixed input distribution for the $\mathbf{W}_n(p_\mathsf{d},p_\mathsf{r})$ channel. Hence, we can assume an unordered pair $\{p_\mathsf{d},p_\mathsf{r}\}$ parameterizing the channel space. □

As for the Bernoulli state channel, we can define the genie-erasure capacity $C_{g\varepsilon}(p_\mathsf{d},p_\mathsf{r})$ of the binary Markov state channel $\mathbf{W}_n(p_\mathsf{d},p_\mathsf{r})$. In this case, the resulting channel is a correlated erasure channel with an average erasure probability $\mathsf{P}(Z_{i-1} = 1, Z_i = 0) = \frac{p_\mathsf{d}p_\mathsf{r}}{p_\mathsf{d}+p_\mathsf{r}}$, so that

$$C(p_\mathsf{d},p_\mathsf{r}) \le 1 - \frac{p_\mathsf{d}p_\mathsf{r}}{p_\mathsf{d}+p_\mathsf{r}} \triangleq C_{g\varepsilon}(p_\mathsf{d},p_\mathsf{r}). \tag{3.14}$$

As a result of the memory in the $\mathscr{Z}$ process, it is considerably harder than it was for the $\mathbf{W}_n(p)$ channel to obtain closed-form expressions for lower bounds on the capacity of $\mathbf{W}_n(p_\mathsf{d},p_\mathsf{r})$ by computing information rates for known input distributions. However, the SIR $C^{\text{iud}}(p,p)$ can still be obtained numerically. Figure 3.9 shows the contours of the SIR for the $\mathbf{W}_n(p_\mathsf{d},p_\mathsf{r})$ channel. Note that the symmetry proved in Proposition 3.4 is evident.

In Figure 3.10, we show the SIR for the case $p_\mathsf{r} = p_\mathsf{d} = p$, which we call the *symmetric* binary Markov state channel $\mathbf{W}_n(p,p)$. When $p = 0$, the channel is noiseless. When $p = 1$, the channel deterministically flips between the identity and the delayed channel so that every odd bit is repeated twice, and every even bit is lost, and the maximum achievable information transfer rate is $\frac{1}{2}$ bit per channel use. Also shown is the genie-erasure capacity in (3.14), which in this case becomes,

$$C(p,p) \le C_{g\varepsilon}(p,p) = 1 - \frac{p}{2}.$$

Interestingly enough, when $p = 1$, the SIR satisfies $C^{\text{iud}}(1,1) = C_{g\varepsilon}(1,1)$, which implies that $C(1,1) = \frac{1}{2}$. We also include in Figure 3.11 the SIR obtained for the channel $\mathbf{W}_n(1,p_\mathsf{r})$, as well as the corresponding genie-erasure capacity, $C_{g\varepsilon}(1,p_\mathsf{r}) = \frac{1}{1+p_\mathsf{r}}$.

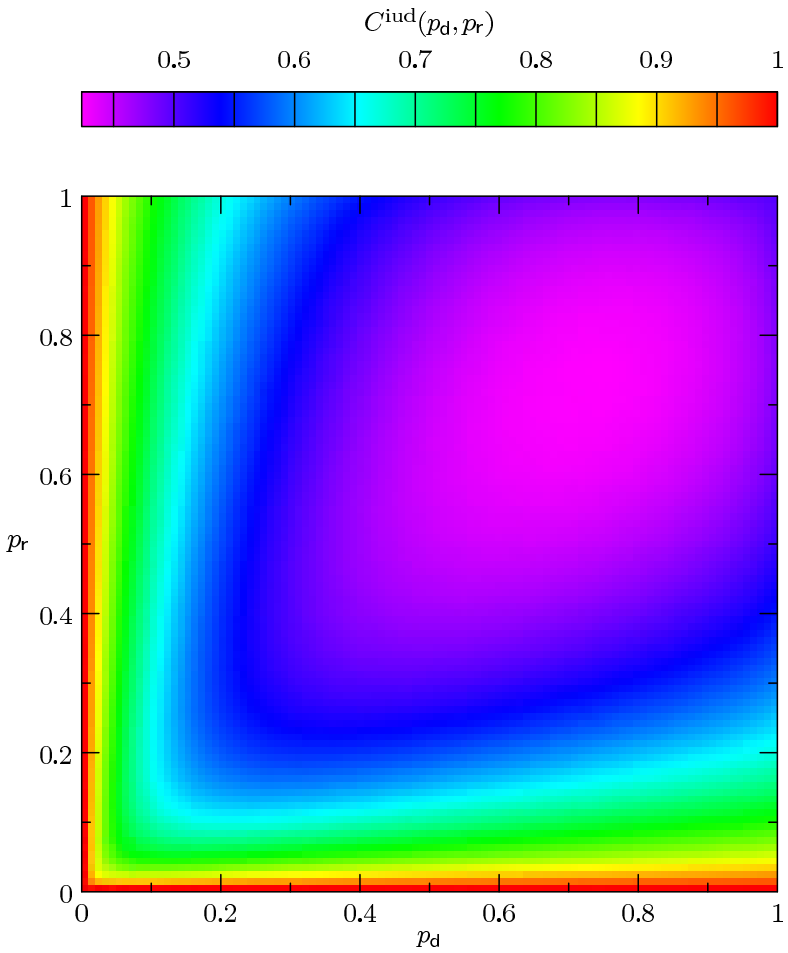

Figure 3.9: The SIR $C^{\mathrm{iud}}(p_{\mathsf{d}}, p_{\mathsf{r}})$ estimates for the binary Markov state channel $\mathbf{W}_n(p_{\mathsf{d}}, p_{\mathsf{r}})$.

3.5 K-ary Markov State Channel

We now consider a generalization of the binary Markov state channel as described in Equation (3.2). Here, we allow $Z_i \in \{0, 1, \cdots, K-1\}$ and let $\mathscr{Z}$ be a first-order Markov process whose transition probabilities satisfy

$$\begin{aligned} \mathsf{P}(Z_i = z | Z_{i-1} = z+1) &= p_{\mathsf{d}}, \\ \mathsf{P}(Z_i = z | Z_{i-1} = z-1) &= p_{\mathsf{r}}, \\ \mathsf{P}(Z_i = z | Z_{i-1} = z) &= 1 - p_{\mathsf{d}} - p_{\mathsf{r}} \end{aligned}$$

for $z \in \{1, 2, \cdots, K-2\}$. Further, we have

$$\mathsf{P}(Z_i = 1 | Z_{i-1} = 0) = p_{\mathsf{r}} = 1 - \mathsf{P}(Z_i = 0 | Z_{i-1} = 0)$$

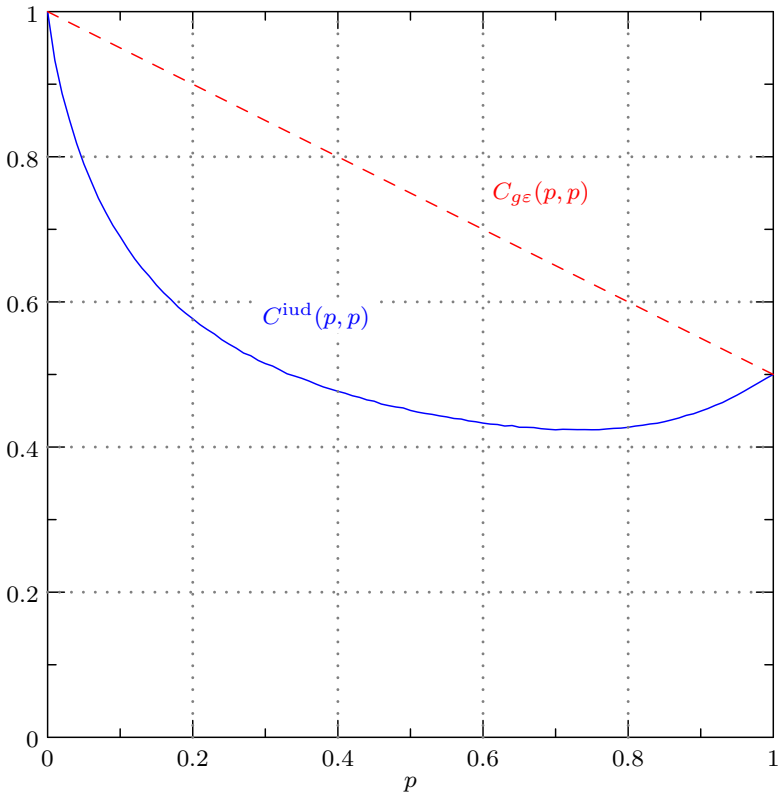

Figure 3.10: The SIR $C^{\mathrm{iud}}(p,p)$ and the genie-erasure capacity $C_{g\varepsilon}(p,p)$ for the symmetric binary Markov state channel $\mathbf{W}_n(p,p)$.

and

$$\mathsf{P}(Z_i = K-2 | Z_{i-1} = K-1) = p_{\mathsf{d}} = 1 - \mathsf{P}(Z_i = K-1 | Z_{i-1} = K-1).$$

Note that when $K = 2$, this model gives the binary Markov state channel considered earlier. We will hence be interested in the K-ary Markov state channel where $K > 2$, which we denote by $\mathbf{W}_n(p_{\mathsf{d}}, p_{\mathsf{r}}, K)$. Note that this channel now generalizes the binary Markov state channel in the sense that it allows up to $(K-1)$ consecutive deletions or replications.

We further assume that the parameters $p_{\mathsf{d}}, p_{\mathsf{r}}$ are chosen such that the process $\mathscr{Z}$ is aperiodic and irreducible so that $\mathscr{Y}$ is ergodic. As was the case for the binary Markov state channel, we have $C(p_{\mathsf{d}}, 0, K) = C(0, p_{\mathsf{r}}, K) = 1 \ \forall \ p_{\mathsf{d}}, p_{\mathsf{r}} \in [0,1]$. The channel space is therefore given by $p_{\mathsf{d}} \in [0,1], p_{\mathsf{r}} \in [0, 1-p_{\mathsf{r}}]$. The channel symmetry argument of

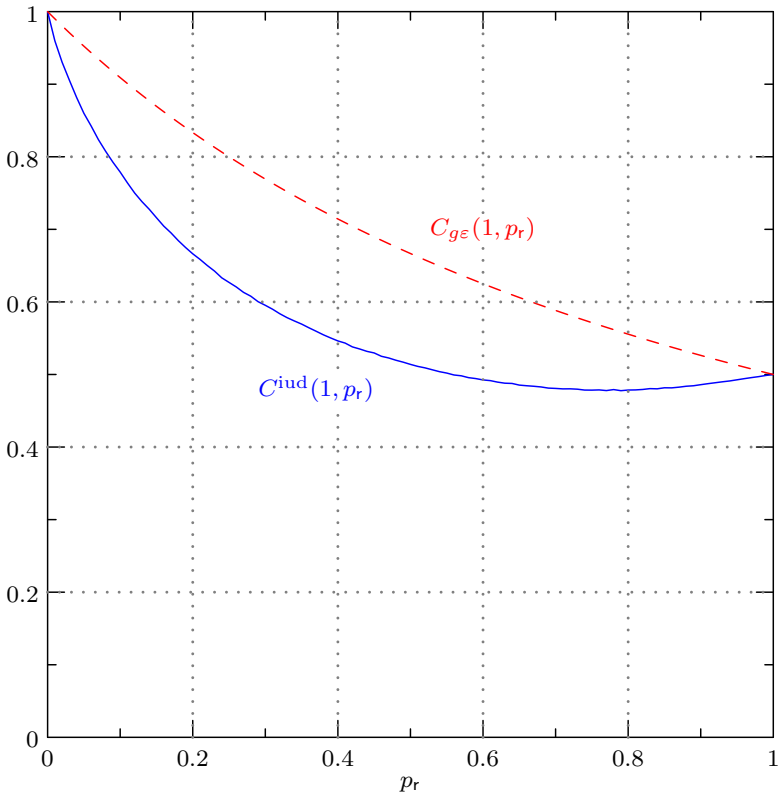

Figure 3.11: The SIR $C^{\mathrm{iud}}(1,p_r)$ and the genie-erasure capacity $C_{g\varepsilon}(1,p_r)$ for the channel $\mathbf{W}_n(1,p_r)$ with the $\mathscr{Z}$ process satisfying the $(1,\infty)$ constraint.

Proposition 3.4 holds in this case also, so that

$$C(p_d,p_r,K) = C(p_r,p_d,K)$$

where $C(p_d,p_r,K)$ is the capacity of K-ary Markov state channel $\mathbf{W}_n(p_d,p_r,K)$. Hence, the channel space can be further reduced to $p_d \in [0,1], p_r \in [0,\min\{1-p_d,p_d\}]$.

The genie-erasure capacity of the $\mathbf{W}_n(p_d,p_r,K)$ channel is given by the capacity of a correlated erasure channel with an average erasure rate

$$\begin{aligned}
\mathsf{P}(Z_i = Z_{i-1}-1) &= \sum_{z=1}^{K-1} \mathsf{P}(Z_{i-1}=z, Z_i = z-1) \\
&= \sum_{z=1}^{K-1} \mathsf{P}(Z_{i-1}=z)\mathsf{P}(Z_i = z-1 | Z_{i-1}=z) = \sum_{z=1}^{K-1} \pi_z p_d \\
&= p_d \left(1 - \left(\frac{p_d}{p_r}\right)^{K-1} \left(\frac{1-\left(\frac{p_d}{p_r}\right)}{1-\left(\frac{p_d}{p_r}\right)^{K}}\right)\right) \triangleq p_\varepsilon
\end{aligned}$$

where $\pi_z = \left(\frac{p_\mathsf{d}}{p_\mathsf{r}}\right)^{K-1-z} \left(\frac{1-\left(\frac{p_\mathsf{d}}{p_\mathsf{r}}\right)}{1-\left(\frac{p_\mathsf{d}}{p_\mathsf{r}}\right)^K}\right)$ is the steady state probability of $Z_i = z$, so that

$$C(p_\mathsf{d}, p_\mathsf{r}, K) \leq 1 - p_\varepsilon \triangleq C_{g\varepsilon}(p_\mathsf{d}, p_\mathsf{r}, K).$$

For the *symmetric* K-ary Markov state channel, we have

$$C(p, p, K) \leq 1 - p\frac{K-1}{K} = C_{g\varepsilon}(p, p, K)$$

for $p \in [0, \frac{1}{2}]$ because in this case, $\pi_z = \frac{1}{K}\ \forall\, z \in \{0, 1, \cdots, K-1\}$. Note that $C_{g\varepsilon}(p, p, K)$ reduces to the capacity of a BEC with erasure probability p, $C_{\text{BEC}}(p) = 1 - p$, when $K \to \infty$.

The SIR of the $\mathbf{W}_n(p_\mathsf{d}, p_\mathsf{r}, K)$ channel can be obtained numerically as in the case of the $\mathbf{W}_n(p_\mathsf{d}, p_\mathsf{r})$ channel. The estimated SIR for $K = 3$ is shown in Figure 3.12.

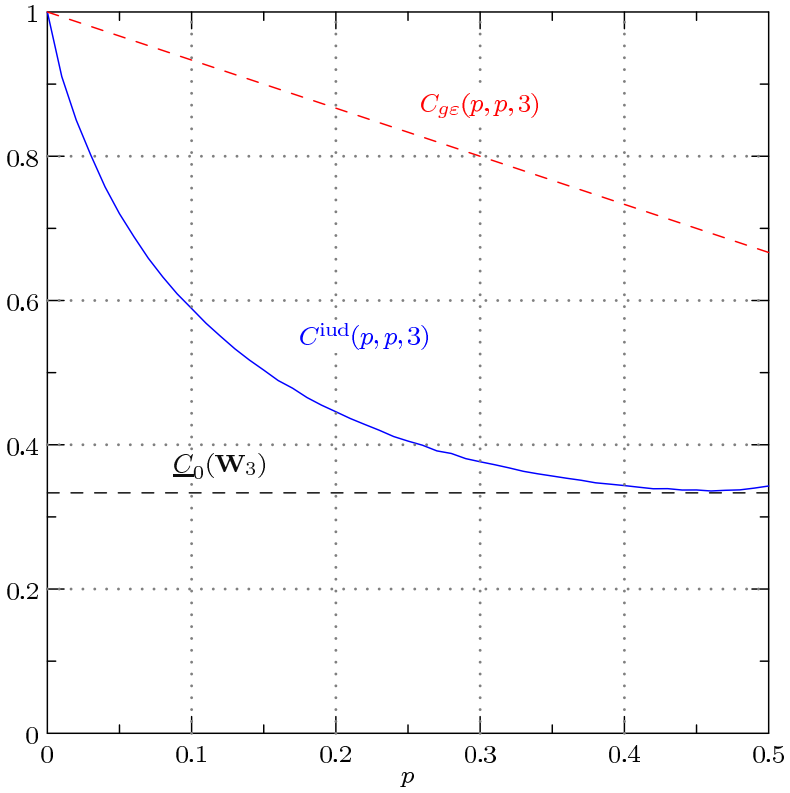

Figure 3.12: The SIR $C^{\text{iud}}(p, p, 3)$ for the symmetric ternary Markov state channel $\mathbf{W}(p, p, 3)$, along with the genie-erasure capacity $C_{g\varepsilon}(p, p, 3)$. Also shown is the lower bound on the zero-error capacity $\underline{C}_0(\mathbf{W}_3)$ from Corollary 3.6.

3.6 Zero-error capacity

The *zero-error capacity* C_0 of a discrete-memoryless channel was introduced by Shannon [102], from which it is readily seen that a noisy discrete-output binary-input memoryless channel has a zero-error capacity of 0, i.e., no information can be transmitted over such a channel with zero error. However, we will now show that the zero-error capacity of the noisy discrete-output binary-input channel in Equation (3.1) is strictly positive. We will denote the generic channel in (3.1) for all binary $\mathscr{Z}$ processes by $\mathbf{W}$.

Proposition 3.5. $C_0(\mathbf{W}) = \frac{1}{2}$.

Proof. Consider the 2-repetition code. The channel input process $\mathscr{X}$ and the message process $\mathscr{M}$ satisfy the relationship $X_{2i-1} = M_i, X_{2i} = M_i$ for $i = 1, 2, \cdots$. Since $X_{2i-1} = X_{2i}$, $Y_{2i} = X_{2i} = M_i$ so that discarding the Y_{2i-1}s gives us $\mathscr{M}$ exactly, thereby achieving zero error at a rate $\frac{1}{2}$. Thus, $C_0(W) \geq \frac{1}{2}$.

It is obvious that the capacity of a channel is an upper bound on the zero-error capacity. From (3.14), we can see that $C(1,1) \leq \frac{1}{2}$, and the coding scheme above achieves a rate $\frac{1}{2}$ for any realization of $\mathscr{Z}$. Thus, $C(1,1) = \frac{1}{2}$. Therefore, as long as the channel admits the infinitely long alternating sequence $\mathscr{Z} = 010101\cdots$, we have $C_0(\mathbf{W}) = \frac{1}{2}$.

Note that both the Bernoulli state channel $\mathbf{W}_n(p)$ and the binary Markov state channel $\mathbf{W}_n(p_{\mathsf{d}}, p_{\mathsf{r}})$ can generate the infinite sequence $\mathscr{Z} = 010101\cdots$, albeit with vanishing probability. It is also worth noting here that this infinite sequence satisfies the $(1,\infty)$ constraint (See Section 3.2.1). □

Thus, unlike binary-input discrete memoryless channels, the binary-input discrete channel in (3.1) allows a non-zero information rate even under the severe requirement of zero-error. From this result and Figure 3.8, it is clear that even under the milder condition of asymptotically vanishing error probability, random linear coset coding achieves a lower rate than the 2-repetition code, which guarantees zero errors, for the Bernoulli state channel over a range of p values. Referring to Figure 3.9, the same can be said for the binary Markov state channels. However, by using a first-order Markov input (cf. Figure 3.8), higher rates than that of the zero-error achieving scheme

are achievable for all Bernoulli state channels, although only with asymptotically vanishing error probability.

The zero-error capacity of the $\mathbf{W}_n(p_\mathsf{d}, p_\mathsf{r}, K)$ channel, denoted by $C_0(\mathbf{W}_K)$, satisfies the following bounds.

Corollary 3.6. $\frac{1}{K} \leq C_0(\mathbf{W}_K) \leq \frac{K+1}{2K}$.

Proof. Repeating every bit K times achieves zero error, since every K^{th} bit is always correct for any realization of the $\mathscr{Z}$ process. Hence, $C_0(\mathbf{W}_K) \geq \frac{1}{K} = \underline{C}_0(\mathbf{W}_K)$.

The smallest upper bound for the capacity is also an upper bound for the zero-error capacity. Thus,

$$C_0(\mathbf{W}_K) \leq \min_{p_\mathsf{d}, p_\mathsf{r}} C(p_\mathsf{d}, p_\mathsf{r}, K) \leq \min_{p_\mathsf{d}, p_\mathsf{r}} C_{g\varepsilon}(p_\mathsf{d}, p_\mathsf{r}, K) = C_{g\varepsilon}\left(\frac{1}{2}, \frac{1}{2}, K\right) = \frac{K+1}{2K}.$$

Observe that when $K = 2$, the minimum genie-erasure capacity occurs at $p_\mathsf{d} = p_\mathsf{r} = 1$ and at this point the upper bound $C_{g\varepsilon}(1,1,2)$ is same as the lower bound $\underline{C}_0(\mathbf{W}_2)$ which was used in the proof of Proposition 3.5. □

Note that the bounds proposed above are loose asymptotically, i.e., $\frac{1}{K} \to 0$ and $\frac{K+1}{2K} \to \frac{1}{2}$ as $K \to \infty$.

3.7 Conclusions

We proposed a new write channel model for bit-patterned media recording that reflects the data dependence of write synchronization errors. For Bernoulli and Markov channel state models, we computed bounds and numerical estimates for the maximum achievable information rate under different assumptions on the channel input statistics. We then generalized the Markov channel state model to allow a channel state space of size $K > 2$ and computed the SIR numerically for the case $K = 3$. Finally, we showed that the rate-$\frac{1}{2}$ 2-repetition code achieves the zero-error capacity over the new write channel when the channel state space is binary. Bounds on the zero-error capacity of the general K-ary Markov channel state model were also presented.

Acknowledgement

This chapter contains material from the paper “Write Channel Model for Bit-Patterned Media Recording”, A.R. Iyengar, P. H. Siegel, J. K. Wolf; published in the IEEE Transactions on Magnetics, vol. 47, no. 1, Jan. 2011. The dissertation author was the primary investigator and author of this paper.

Chapter 4

Windowed Decoding of Spatially Coupled Codes

Sparse graph codes have been of great interest in the coding community for close to two decades, after it was shown that statistical inference techniques on graphical models representing these codes had decoding performance that surpassed that of the best known codes. One class of such codes are low-density parity-check (LDPC) codes, which although introduced by Gallager in the 60's [38] were rediscovered in the 90's after the advent of Turbo Codes [13] and iterative decoding. Luby et al. showed [78, 79] that a decoder based on *belief propagation* (BP) [92] had very good performance for these codes over the binary erasure channel (BEC). This superior performance of LDPC codes was shown by Richardson and Urbanke [99] to be true over a broader class of binary-input, memoryless, symmetric-output (BMS) channels. Furthermore these codes were optimized to approach capacity on many of these BMS channels [2, 3].

The convolutional counterparts of LDPC block codes were first introduced by Felstrom and Zigangirov in [33], although a similar construction was suggested in [110]. There is considerable literature on the constructions and analysis of these ensembles [30, 31, 69, 111]. The BP thresholds for these ensembles were reported in [108]. In [67] the authors construct regular LDPC convolutional codes based on *protographs* [112] that have BP thresholds close to capacity. In [63], Kudekar et al. considered convolutional-like codes which they called *spatially coupled codes* and showed that the BP thresholds of these codes approached the *maximum a-posteriori* (MAP) thresholds of the underly-

ing unstructured ensembles over the BEC. This observation was made for protograph-based generalized LDPC codes in [71]. Evidence for similar results over general BMS channels was given in [62], and proven recently in [64]. Moreover this phenomenon, termed *threshold saturation*, was shown to be a more generic effect of coupling by showing an improvement in performance of systems based on other graphical models : the random K-SAT, Q-COL problems from computation theory, Curie-Weiss model from statistical mechanics [42], and LDGM & rateless code ensembles [5]. Non-binary LDPC codes obtained through coupling have also recently been investigated [114].

The good performance of spatially coupled codes is apparent when both the blocklength of individual codes and the coupling length becomes large. However, as either of these parameters becomes large, BP decoding becomes complex. We therefore consider a *windowed decoder* that exploits the structure of the coupled codes to reduce the decoding complexity while maintaining the advantages of the BP decoder in terms of performance. An additional advantage of the windowed decoder is the reduced latency of decoding. The windowed decoding scheme studied here is the one used to evaluate the performance of protograph-based codes over erasure channels with and without memory [47, 48, 90], which will be the topic of the next chapter. The main result in this chapter is that the windowed decoding thresholds over the BEC approach the BP thresholds exponentially in the size of the window W. Since the BP thresholds are themselves close to the MAP thresholds for spatially coupled codes, windowed decoding thus gives us a way to achieve close to ML performance with complexity reduced further beyond that of the BP decoder. Although the results presented here are proved only for the BEC, similar asymptotic behavior is expected on general BMS channels.

This chapter is organized as follows. Section 4.1 gives a brief introduction to spatially coupled codes. In Section 4.2 we discuss the windowed decoding scheme. We state here the main result which we prove in Section 4.3. We give some numerical results in Section 4.4 and conclude in Section 4.5. Much of the terminology and notation used in here is reminiscent of the definitions in [63].

4.1 Spatially Coupled Codes

We describe the (d_l, d_r) spatially coupled ensemble that was introduced in [63] in terms of its Tanner graph. There are M variable nodes at each position in $[L] \triangleq \{1, 2, \cdots, L\}$. We will assume that there are $M\frac{d_l}{d_r}$ check nodes at every integer position, but only some of these interact with the variable nodes. The variable (check) nodes at position i constitute the i^{th} *section* of variable (check, respectively) nodes in the code. The L sections of variables are together referred to as the *chain* and L is called the *chain length*. For each of the d_l edges incident on a variable at position $i \in [L]$, we first choose a section uniformly at random from the set $\{i, i+1, \cdots, i+\gamma-1\}$, then choose a check uniformly at random from the $M\frac{d_l}{d_r}$ checks in the chosen section, and connect the variable to this check. We refer to the parameter γ as the *coupling length*. It can be shown (see, e.g., [63]) that this procedure amounts roughly to choosing each of the d_r connections of a check node at position i uniformly and independently from the set $\{i-\gamma+1, i-\gamma+2, \cdots, i\}$. Observe that when $\gamma = 1$ this procedure gives us L copies of the (d_l, d_r)-regular uncoupled ensemble. Since we are interested in coupled ensembles, we will henceforth assume that $\gamma > 1$. Further, we will typically be concerned with this ensemble when $L \gg \gamma$, in which case the *design rate* given by

$$R(d_l, d_r, \gamma, L) = 1 - \frac{d_l}{d_r}\left(1 + O(\frac{\gamma}{L})\right) \tag{4.1}$$

is close to $1 - \frac{d_l}{d_r}$.

BP Performance

In the following we will briefly state known results that are relevant to this work. See [63] for detailed analysis of the BP performance of spatially coupled codes. The BP performance of the (d_l, d_r, γ, L) spatially coupled ensemble when $M \to \infty$ can be evaluated using *density evolution*. Denote the average erasure probability of a message from a variable node at position i as x_i. We refer to the vector $\underline{x} = (x_1, x_2, \cdots, x_L)$ as the *constellation*.

Definition 4.1 (BP Forward Density Evolution)**.** *Consider the BP decoding of a* (d_l, d_r, γ, L) *spatially coupled code over a BEC with channel erasure rate* ε*. We can write the*

forward density evolution (DE) equation as follows. Set the initial constellation to be $\underline{x}^{(0)} = (1, 1, \cdots, 1)$ *and evaluate the constellations* $\{\underline{x}^{(\ell)}\}_{\ell=1}^{\infty}$ *according to*

$$x_i^{(\ell)} = \begin{cases} 0, & \text{if } i \notin [L] \ \forall \ \ell, \\ \varepsilon\Big(1 - \frac{1}{\gamma}\sum_{j=0}^{\gamma-1}(1 - \frac{1}{\gamma}\sum_{k=0}^{\gamma-1} x_{i+j-k}^{(\ell-1)})^{d_r-1}\Big)^{d_l-1} & \text{, otherwise.} \end{cases} \tag{4.2}$$

This is called the parallel schedule *of the BP forward density evolution.* □

For ease of notation, we will write

$$x_i = \varepsilon\Big(1 - \frac{1}{\gamma}\sum_{j=0}^{\gamma-1}(1 - \frac{1}{\gamma}\sum_{k=0}^{\gamma-1} x_{i+j-k})^{d_r-1}\Big)^{d_l-1} = \varepsilon g(x_{i-\gamma+1}, \cdots, x_{i+\gamma-1}). \tag{4.3}$$

It is clear that the function $g(\cdot)$ is monotonic in each of its arguments.

Definition 4.2 (FP of BP Forward DE)**.** *Consider the parallel schedule of the BP forward DE for the* (d_l, d_r, γ, L) *spatially coupled code over a BEC with erasure rate* ε*. It can be easily seen from the monotonicity of* $g(\cdot)$ *in Equation* (4.3) *that the sequence of constellations* $\{\underline{x}^{(\ell)}\}_{\ell=0}^{\infty}$ *are ordered as* $\underline{x}^{(\ell)} \succeq \underline{x}^{(\ell+1)} \ \forall \ \ell \geq 0$*, i.e.,* $x_i^{(\ell)} \geq x_i^{(\ell+1)} \ \forall \ \ell \geq 0, i \in [L]$ *(the ordering is pointwise). Since the constellations are all lower bounded by the all-zero constellation* $\underline{0}$*, the sequence converges pointwise to a limiting constellation* $\underline{x}^{(\infty)}$*, called the fixed point (FP) of the forward DE.* □

It is clear that the FP of forward DE $\underline{x}^{(\infty)}$ satisfies

$$x_i^{(\infty)} = \begin{cases} 0, & i \notin [L] \\ \varepsilon g(x_{i-\gamma+1}^{(\infty)}, \cdots, x_{i+\gamma-1}^{(\infty)}), & i \in [L]. \end{cases}$$

Definition 4.3 (BP Threshold)**.** *Consider the parallel schedule of the BP forward DE for the* (d_l, d_r, γ, L) *spatially coupled code over a BEC with erasure rate* ε*. The* BP threshold $\varepsilon^{\text{BP}}(d_l, d_r, \gamma, L)$ *is defined as the supremum of the channel erasure rates* $\varepsilon \in [0, 1]$ *for which the FP of forward DE is the all-zero constellation, i.e.,* $\underline{x}^{(\infty)} = \underline{0}$*.* □

Table 4.1 gives the BP thresholds evaluated from BP forward DE for the $(d_l = 3, d_r = 6)$ coupled ensemble for a few values of γ and L rounded to the sixth decimal place. The MAP threshold of the underlying (d_l, d_r)-regular ensemble is $\varepsilon^{\text{MAP}}(d_l = 3, d_r = 6) \approx 0.488151$. We see from the table that the BP thresholds for (d_l, d_r) spatially

Table 4.1: BP Thresholds $\varepsilon^{\mathrm{BP}}$ for the $(d_l = 3, d_r = 6, \gamma, L)$ spatially coupled code ensemble.

$L\backslash\gamma$	2	3	4
16	0.487983	0.488207	0.489805
32	0.487656	0.487923	0.488044
64	0.487014	0.487514	0.487733

coupled codes are close to the MAP threshold of the (d_l, d_r)-regular unstructured code ensemble for all γ when L is large enough. Note that some of the thresholds in Table 4.1 are larger than the corresponding MAP threshold of the underlying ensemble. This is because, for small values of L, the rate of the spatially coupled ensemble (cf. Equation (4.1)) is much smaller than that of the underlying regular ensemble.

It was shown in [63] that the BP thresholds satisfy

$$\lim_{\gamma\to\infty}\lim_{L\to\infty}\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma,L) = \lim_{\gamma\to\infty}\lim_{L\to\infty}\varepsilon^{\mathrm{MAP}}(d_l,d_r,\gamma,L) = \varepsilon^{\mathrm{MAP}}(d_l,d_r).$$

This means that the BP threshold *saturates* to the MAP threshold under these limits, and we can obtain MAP performance with the reduced complexity of the BP decoder. Later when we analyze the windowed decoder, we will want to keep the coupling length γ finite and hence will be concerned with the quantity

$$\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma) \triangleq \lim_{L\to\infty}\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma,L) \tag{4.4}$$

as a measure of the performance of the BP decoder. It immediately follows from [63, Theorem 10] that

$$\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma) \leq \varepsilon^{\mathrm{MAP}}(d_l,d_r).$$

4.2 Windowed Decoding

The *windowed decoder* (WD) exploits the structure of the spatially coupled codes to break down the BP decoding scheme into a series of sub-optimal decoding steps—we trade-off the performance of the decoder for reduced complexity and decoding latency. When decoding with a window of size W, the WD performs BP over the subcode consisting of the first W sections of the variable nodes and their neighboring check nodes and attempts to decode a subset of symbols (those in the first section) within

the window. These symbols that we attempt to decode within a window are referred to as the *targeted symbols*. Upon successful decoding of the targeted symbols (or when a maximum number of iterations have been performed) the window slides over one section and performs BP, attempting to decode the targeted symbols in the window in the new position.

More formally, let $\underline{x}$ be the constellation representing the average erasure probability of messages from variables in each of the sections 1 through L. Initially, the window consists only of the first W sections in the chain. We will refer to this as the *first window configuration*, and as the window slides to the right, we will increment the window configuration. In other words, when the window has slid through $(c-1)$ sections to the right (when it consists of sections $c, c+1, \cdots, c+W-1$), it is said to be in the c^{th} window configuration. The c^{th} *window constellation*, denoted $\underline{y}_{\{c\}}$, is the average erasure probability of the variables in the c^{th} window configuration. Thus,

$$\underline{y}_{\{c\}} = (y_{1,\{c\}}, y_{2,\{c\}}, \cdots, y_{W,\{c\}}) = (x_c, x_{c+1}, \cdots, x_{c+W-1})$$

for $c \in [L]$, where we assume that $x_c = 0 \; \forall \; c > L$. Thus the c^{th} window constellation, $\underline{y}_{\{c\}}$, represents the "active" sections within the constellation $\underline{x}$. While referring to the entire constellation after the action of the c^{th} window, we will write $\underline{x}_{\{c\}}$. When the window configuration being considered is clear from the context, with some abuse of notation, we drop the $\{c\}$ from the notation and write $\underline{y} = (y_1, \cdots, y_W)$ to denote the window constellation.

Remark 4.1 (Note on notation). When we wish to emphasize the size of the window when we write the constellation, we write $\underline{y}_{\langle W \rangle} = (y_{i,\langle W \rangle}, y_{2,\langle W \rangle}, \cdots, y_{W,\langle W \rangle})$. Note that the window configuration and the window size are specified as subscripts within curly brackets $\{\cdot\}$ and angle brackets $\langle \cdot \rangle$, respectively. Finally, when the constellation after a particular number of iterations ℓ of DE is to be specified, we write $\underline{y}^{(\ell)} = (y_1^{(\ell)}, y_2^{(\ell)}, \cdots, y_W^{(\ell)})$, where the iteration number appears as a superscript within parentheses $(\cdot)$. Although $\underline{y}^{(\ell)}_{\{c\},\langle W \rangle}$ would be the most general way of specifying the window constellation for the c^{th} window configuration with a window of size W after ℓ iterations of DE, for notational convenience we will write as few of these parameters as possible based on the relevance to the discussion. $\square$

4.2.1 Complexity and Latency

For the BP decoder, the number of iterations required to decode all the symbols in a (d_l, d_r, γ, L) spatially coupled code depends on the channel erasure rate ε. Whereas when $\varepsilon \in [0, \varepsilon^{\text{BP}}(d_l, d_r)]$ this required number of iterations can be fixed to a constant number, when $\varepsilon \in (\varepsilon^{\text{BP}}(d_l, d_r), \varepsilon^{\text{BP}}(d_l, d_r, \gamma, L)]$ the number of iterations scales as $O(L)$ [87]. Therefore, in the waterfall region, the complexity of the BP decoder scales as $O(ML^2)$. For the WD of size W, if we let the number of iterations performed scale as $O(W)$, the overall complexity is of the order $O(MW^2L)$. Thus, for small window sizes $W < \sqrt{L}$, we see that the complexity of the decoder can be reduced. A larger reduction in the complexity is possible if we fix the number of iterations performed within each window.

Another advantage of using the WD is that the decoder only needs to know the symbols in the first W sections of the code to be able to decode the targeted symbols. Therefore, in latency-constrained applications, the decoder can work on-the-fly, resulting in a latency which is a fraction $\frac{W}{L}$ that of the BP decoder.

4.2.2 Asymptotic Performance

The asymptotic performance of the (d_l, d_r, γ, L) spatially coupled ensemble with WD can be analyzed using density evolution as was done for the BP decoder. We will consider the performance of the ensemble with $M \to \infty$ when the transmission happens over a BEC with channel erasure rate $\varepsilon \in [0, 1]$. Further we will assume that for each window configuration, infinite rounds of message passing are performed.

Definition 4.4 (WD Forward Density Evolution)**.** *Consider the WD of a (d_l, d_r, γ, L) spatially coupled code over a BEC with channel erasure rate ε with a window of size W. We can write the forward DE equation as follows. Set the initial constellation $\underline{x}_{\{0\}}$ according to*

$$x_{i,\{0\}} = \begin{cases} 1, & i \in [L] \\ 0, & i \notin [L]. \end{cases}$$

For every window configuration $c = 1, 2, \cdots, L$, let

$$\underline{y}^{(0)}_{\{c\}} = (x_{c,\{c-1\}}, x_{c+1,\{c-1\}}, \cdots, x_{c+W-1,\{c-1\}})$$

and evaluate the sequence of window constellations $\{\underline{y}^{(\ell)}_{\{c\}}\}_{\ell=1}^{\infty}$ *using the update rule*

$$y^{(\ell)}_{i,\{c\}} = \varepsilon g(z^{(\ell-1)}_{i-\gamma+1,\{c\}}, \cdots, z^{(\ell-1)}_{i+\gamma-1,\{c\}}), i \in [W],$$

where for every ℓ*,*

$$z^{(\ell)}_i = \begin{cases} x_{c+i-1,\{c-1\}}, & i \notin [W] \\ y^{(\ell)}_i, & i \in [W], \end{cases}$$

and set $\underline{x}_{\{c\}}$ *as*

$$x_{i,\{c\}} = \begin{cases} x_{i,\{c-1\}}, & i \neq c \\ y^{(\infty)}_{1,\{c\}}, & i = c. \end{cases}$$

□

Discussion : Note that the constellation $\underline{x}_{\{c\}}$ keeps track of the erasure probabilities of targeted symbols of all window configurations up to the c^{th}, followed by erasure probability of 1 for the variables in sections $c+1$ through W, and zeros for sections outside this range. As defined, $\underline{x}_{\{c\}}$ discards all information obtained by running the WD in its c^{th} configuration apart from the values corresponding to the targeted symbols. In practice, it is more efficient to define

$$x_{i,\{c\}} = \begin{cases} x_{i,\{c-1\}}, & i \notin \{c, c+1, \cdots, c+W-1\} \\ y^{(\infty)}_{i-c+1,\{c\}}, & \text{otherwise.} \end{cases}$$

In the sequel, we will stick to Definition 4.4. We do this for two reasons: first, discarding some information between two window configurations can only perform worse than retaining all the information; and second, this assumption makes the analysis simpler since we then have $\underline{y}^{(0)}_{\{c\}} = \underline{1}\ \forall\ c \in [L]$. □

Definition 4.4 implicitly assumes that the limiting window constellations $\underline{y}^{(\infty)}_{\{c\}}$ exist. The following guarantees that the updates for $x_{i,\{c\}}$ are well-defined.

Definition 4.5 (c^{th} Window Configuration FP of FDE)**.** *Consider the WD forward DE (FDE) of a* (d_l, d_r, γ, L) *spatially coupled code over a BEC with erasure rate* ε *with a window of size* W*. Then the limiting window constellation* $\underline{y}^{(\infty)}_{\{c\}}$ *exists for each* $c \in [L]$*. We refer to this constellation as the* c^{th} *window configuration FP of forward DE.* □

Discussion : As noted earlier, $\underline{y}^{(0)}_{\{c\}} = \underline{1}\ \forall\ c \in [L]$, and $\underline{y}^{(0)}_{\{c\}} = \underline{1} \succeq \underline{\varepsilon} \succeq \underline{y}^{(1)}_{\{c\}}$. By induction, from the monotonicity of $g(\cdot)$, this implies that $\underline{y}^{(\ell)}_{\{c\}} \succeq \underline{y}^{(\ell+1)}_{\{c\}}\ \forall\ \ell \geq 0$. Since these constellations are lower bounded by $\underline{0}$, the c^{th} window configuration FP of FDE $\underline{y}^{(\infty)}_{\{c\}}$ exists for every $c \in [L]$. □

The c^{th} window configuration FP of forward DE therefore satisfies

$$y^{(\infty)}_{i,\{c\}} = \begin{cases} x_{c+i-1,\{c-1\}}, & i \notin [W] \\ \varepsilon g(y^{(\infty)}_{i-\gamma+1,\{c\}}, \cdots, y^{(\infty)}_{i+\gamma-1,\{c\}}), & i \in [W] \end{cases} \tag{4.5}$$

for every $c \in [L]$. Since the $\underline{x}_{\{0\}}$ vector has non-zero values by definition, from the continuity of the WD FDE equations, so do the vectors $\underline{x}_{\{c\}}\ \forall\ c$. Hence $\underline{0}$ cannot satisfy Equation (4.5), i.e., $\underline{0}$ cannot be the c^{th} window configuration FP of forward DE. Therefore, $\underline{y}^{(\infty)}_{\{c\}} \succ \underline{0}\ \forall\ c \in [L]$. This means that WD can never reduce the erasure probability of the symbols of a spatially coupled code to zero, although it can be made arbitrarily small by using a large enough window. Therefore, an acceptable *target erasure rate* δ forms a part of the description of the WD. We say that the WD is successful when $\underline{x}_{\{L\}} \preceq \underline{\delta}$.

Lemma 4.1 (Maximum of $\underline{x}_{\{L\}}$)**.** *The vector $\underline{x}_{\{L\}}$ obtained at the end of WD forward DE satisfies $x_{i-1,\{L\}} \leq x_{i,\{L\}}\ \forall\ i \in [L-\gamma+1]$. Moreover, $\exists\ \hat{x} \in [0,1]$ independent of L such that $x_{i,\{L\}} \leq \hat{x}\ \forall\ i$.*

Proof. By definition, $x_{i,\{L\}} = y^{(\infty)}_{1,\{i\}}\ \forall\ i \in [L]$. The claim is true for $i = 1$ since $x_{1,\{L\}} = y^{(\infty)}_{1,\{1\}} \geq 0 = x_{0,\{L\}}$. For the i^{th} window configuration, it is clear from Definition 4.4 that $\underline{y}^{(0)}_{\{i-1\}} \preceq \underline{y}^{(0)}_{\{i\}}, i \in [L-\gamma+1]$. By induction, from the monotonicity of $g(\cdot)$, it follows that $y^{(\infty)}_{1,\{i-1\}} \leq y^{(\infty)}_{1,\{i\}}$ for i in this range.

For $i \in \{L-\gamma+2, \cdots, L\}$, the above claim is not valid because we defined $x_{j,\{c\}} = 0$ for $j > L$ and we cannot make use of the monotonicity of $g(\cdot)$ since some arguments (corresponding to sections up to the L^{th} section) are increasing and others (corresponding to the sections beyond the L^{th} section) decreasing. Nevertheless, we can still claim that $x_{i,\{L\}} \leq x_{i,\{\infty\}}\ \forall\ i \in \mathbb{N}$ where $\underline{x}_{\{\infty\}}$ is the vector of erasure probabilities obtained after WD for a spatially coupled code with an infinite chain length, i.e., $L = \infty$. For $L = \infty$, the sequence $\{x_{i,\{\infty\}}\}$ is non-decreasing and since the $x_{i,\{\infty\}}$ are probabilities, they are in the bounded, closed interval $[0,1]$. Consequently, the limit $\lim_{i\to\infty} x_{i,\{\infty\}}$ exists in the interval $[0,1]$, and $\lim_{i\to\infty} x_{i,\{\infty\}} = \sup_i x_{i,\{\infty\}} \triangleq \hat{x}$. □

As a consequence of Lemma 4.1, we can say that the WD is successful when $\hat{x} \leq \delta$. This definition of the success of WD is independent of the chain length L and allows us to compare the performance of WD to that of the BP decoder through the thresholds defined in Equation (4.4). Note that although the upper bound for $\hat{x}$ in Lemma 4.1 is a trivial bound, we will in the following give conditions when $\hat{x}$ can be made smaller than an arbitrarily chosen δ, thereby characterizing the WD thresholds.

Definition 4.6 (WD Thresholds)**.** *Consider the WD of a (d_l, d_r, γ, L) spatially coupled code over a BEC of erasure rate ε with a window of size W. The WD threshold $\varepsilon^{\mathrm{WD}}(d_l, d_r, \gamma, W, \delta)$ is defined as the supremum of channel erasure rates ε for which $\hat{x} \leq \delta$.* □

Discussion : Since we defined the WD threshold based on $\hat{x}$, it is clear that this is independent of the chain length L. On the other hand, if we used $\max_{i\in[L]} x_{i,\{L\}} \leq \delta$ as the condition for success of the WD in our definition, we would obtain an L-dependent threshold. But $\hat{x}$ denotes the "worst-case" remnant erasure probability after WD, and imposing constraints on $\hat{x}$ therefore guarantees good performance for codes with any L.

Note that keeping $\hat{x} \leq \delta$ is sufficient to guarantee an *a-posteriori* erasure probability p_e smaller than δ because

$$p_e = \varepsilon\left(\frac{\hat{x}}{\varepsilon}\right)^{\frac{d_l}{d_l-1}} = \hat{x}\left(\frac{\hat{x}}{\varepsilon}\right)^{\frac{1}{d_l-1}} \leq \hat{x} \leq \delta. \qquad \square$$

We will now state the main result and prove it in the following section.

Theorem 4.2 (WD Threshold Bound)**.** *Consider windowed decoding of the (d_l, d_r, γ, L) spatially coupled ensemble over the BEC. Then for a target erasure rate $\delta < \delta_*$, there exists a positive integer $W_{\min}(\delta)$ such that when the window size $W \geq W_{\min}(\delta)$ the WD threshold satisfies*

$$\varepsilon^{\mathrm{WD}}(d_l, d_r, \gamma, W, \delta) \geq \left(1 - \frac{d_l d_r}{2}\delta^{\frac{d_l-2}{d_l-1}}\right)\left(\varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma) - e^{-\frac{1}{\mathsf{B}}\left(\frac{W}{\gamma-1} - \mathsf{A}\ln\ln\frac{\mathsf{D}}{\delta} - \mathsf{C}\right)}\right). \quad (4.6)$$

Here $\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}$ and δ_* are strictly positive constants that depend only on the ensemble parameters d_l, d_r and γ. □

Theorem 4.2 says that the WD thresholds approach the BP threshold $\varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma)$ defined in Equation (4.4) at least exponentially fast in the ratio of the size of the window

W to the coupling length γ for a fixed target erasure probability $\delta < \delta_*$. Moreover, the sensitivity of the bound to changes in δ is small in the exponent in (4.6) owing to the $\ln\ln\frac{1}{\delta}$ factor, but larger in the first term in the product on the right hand side of (4.6) where it is roughly linear in δ. However, since we intend to set δ to be very small, e.g. 10^{-15}, the first term does not influence the bound heavily. The requirement that $W \geq W_{\min}(\delta)$ is necessary to keep the term within parentheses in the exponent non-negative. Therefore the minimum window size required, $W_{\min}(\delta)$, also depends on the constants A, C and D and, in turn, on the ensemble parameters d_l, d_r and γ.

The bound guaranteed by Theorem 4.2 is actually fairly loose. Numerical results suggest that the minimum window size $W_{\min}(\delta)$ is actually much smaller than the bound obtained from analysis (cf. Section 4.3). Density evolution also reveals that for a fixed window size, the WD thresholds are much closer to the BP threshold than the bound obtained from Theorem 4.2. We note here that the gap between analytical results and numerical experiments is mainly due to the reliance on bounding the density evolution function in Equation (4.3) using the counterpart for regular unstructured LDPC ensembles, which proves to be easier to handle than the multivariate Equation (4.3). However, the scaling of the WD thresholds with the window size and the target erasure probability seem to be as dictated by the bound in (4.6), suggesting that Theorem 4.2 captures the essence of the WD algorithm.

Table 4.2 gives the WD thresholds obtained through forward DE for the $(d_l = 3, d_r = 6, \gamma = 3, L)$ spatially coupled ensemble for different target erasure rates δ and different window sizes W. These thresholds have been rounded to the sixth decimal point. A few comments are in order. As can be seen from the table, the thresholds are close

Table 4.2: WD Thresholds $\varepsilon^{\mathrm{WD}}$ for the $(d_l = 3, d_r = 6, \gamma = 3)$ spatially coupled code ensemble with window size W and target erasure rate δ.

$W \backslash \delta$	10^{-6}	10^{-12}	10^{-18}
4	0.068403	0.000772	0.000008
8	0.472992	0.390749	0.254339
16	0.487504	0.487504	0.487504

to $\varepsilon^{\mathrm{BP}}(d_l = 3, d_r = 6, \gamma = 3) \approx 0.487514$ even for window sizes that are much smaller than the $W_{\min}(\delta)$ obtained analytically, e.g., $W = 16$. Moreover, the WD thresholds are more sensitive to changes in δ for small window sizes where the bound in Theorem 4.2

is not valid. It is obvious that the thresholds decrease as δ is decreased. Also note that for a fixed target erasure rate, the window size can be made large enough to make the WD thresholds close to the BP threshold.

4.3 Performance Analysis

In this section, we prove Theorem 4.2 in steps. First, we analyze the performance of the first window configuration. We will characterize the first window configuration FP of forward DE. We will establish that for the variables in the first section of the window, the FP erasure probability can be made small at least double-exponentially in the size of the window. We will show that this is possible for all channel erasure rates smaller than a certain ε, which we will call the *first window threshold* $\varepsilon^{\mathrm{FW}}(d_l, d_r, \gamma, W, \delta)$, provided the window size is larger than a certain minimum size.

Once we have this, we consider the performance of the c^{th} window configuration for $1 < c \leq L$. In this case also, we will show that the FP erasure probability of the first section within the window is guaranteed to decay double-exponentially in the window size. As for the first window configuration, this result holds provided the window size is larger than a certain minimal size and this time the minimal size is slightly larger than the minimal size required for the first window configuration. Moreover, such a result is true for channel erasure rates smaller than a value which is itself smaller than the first window threshold, and this value will be our lower bound for the WD threshold.

4.3.1 First Window Configuration

From Definition 4.4, forward DE for the first window configuration amounts to the following. Set $\underline{y}_{\{1\}}^{(0)} = \underline{1}$ and evaluate the sequence of window constellations $\{\underline{y}_{\{1\}}^{(\ell)}\}_{\ell=1}^{\infty}$ according to

$$y_{i,\{1\}}^{(\ell)} = \begin{cases} 0, & i \leq 0 \\ \varepsilon g(y_{i-\gamma+1,\{1\}}^{(\ell-1)}, \cdots, y_{i+\gamma-1,\{1\}}^{(\ell-1)}), & i \in [W] \\ 1, & i > W. \end{cases} \tag{4.7}$$

Since $\underline{y}^{(0)}_{\{1\}}$ is non-decreasing, i.e., $y^{(0)}_{i,\{1\}} \leq y^{(0)}_{i+1,\{1\}}\ \forall\, i$, so is the first window configuration FP, $\underline{y}^{(\infty)}_{\{1\}}$, by induction and monotonicity of $g(\cdot)$.

Figure 4.1 shows the first window configuration FP of forward DE for the $(d_l = 3, d_r = 6, \gamma = 3, L)$ ensemble with a window of size $W = 16$ for a channel erasure rate $\varepsilon = 0.48812$.

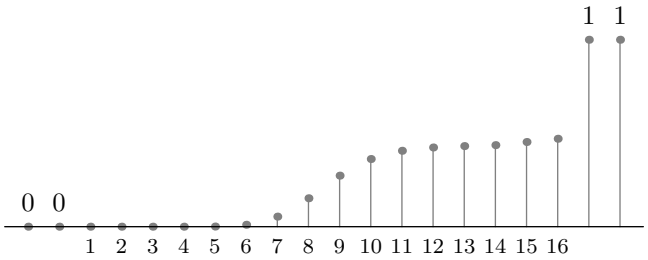

Figure 4.1: The first window configuration FP of forward DE for the $(d_l = 3, d_r = 6, \gamma = 3, L)$ ensemble with a window of size $W = 16$ for $\varepsilon = 0.48812$. The left and the right boundaries are fixed at 0 and 1 respectively. The sections within the window are indexed from 1 to $W = 16$. The first section has a FP erasure probability $y^{(\infty)}_{1,\{1\}} \approx 2 \times 10^{-15}$.

The scheduling scheme used in the definition of the window configuration FPs is what is called the *parallel* schedule. In general, we can consider a scheduling scheme where, in each step, a subset of the sections within the window are updated. We say that such an arbitrary scheduling scheme is admissible if every section is updated infinitely often with the correct boundary conditions, i.e., with the correct values set at the left and the right ends of the window. It is easy to from the standard argument of nested computation trees (see, e.g., [63]) that the FP is independent of the scheduling scheme.

We know that the first window configuration FP of forward DE, $\underline{y}^{(\infty)}_{\{1\}}$, is non-decreasing, i.e., $y^{(\infty)}_{i,\{1\}} \leq y^{(\infty)}_{i+1,\{1\}}\ \forall\, i$. The following shows the ordering of the FP values of individual sections in windows of different sizes. With the understanding that we are considering only the first window configuration in this subsection, we will drop the window configuration number from the notation for window constellations throughout this subsection for convenience.

Lemma 4.3 (FPs and Window Size)**.** *Let $\underline{y}_{\langle W \rangle}$ and $\underline{y}_{\langle W+1 \rangle}$ denote the first window configuration FPs of forward DE with windows of sizes W and $W+1$ respectively for $\varepsilon \in [0,1]$.*

Then,

$$y_{i,\langle W\rangle} \geq y_{i,\langle W+1\rangle} \geq y_{i-1,\langle W\rangle}$$

where $y_{i,\langle W\rangle}$ denotes the FP erasure probability of the i^{th} section in a window of size W.

Proof. Consider the following schedule. Set $\underline{y}^{(0)}_{\langle W+1\rangle} = (\underline{y}_{\langle W\rangle}, 1)$ and evaluate the sequence of window constellations $\{\underline{y}^{(\ell)}_{\langle W+1\rangle}\}$ according to Equation (4.7). Clearly, we have

$$\underline{y}^{(1)}_{\langle W+1\rangle} \preceq (\underline{y}_{\langle W\rangle}, \varepsilon) \preceq (\underline{y}_{\langle W\rangle}, 1) = \underline{y}^{(0)}_{\langle W+1\rangle}$$

so that the sequence $\{\underline{y}^{(\ell)}_{\langle W+1\rangle}\}$ is pointwise non-increasing by induction. We claim that this schedule is admissible. This is true because the DE updates are first performed infinitely many times over the first W sections to obtain $\underline{y}^{(0)}_{\langle W+1\rangle}$, and then over all the $W+1$ sections infinitely many times again. Therefore the updates are performed over all sections infinitely often with the correct boundary conditions. The limiting FP must hence be exactly $\underline{y}_{\langle W+1\rangle}$ and the first inequality in the statement of the lemma holds. Intuitively, this is true because in going from W to $W+1$ and checking the i^{th} section, we have moved further away from the right end of the window (where $y_i = 1$) while remaining at the same distance from the left end (where $y_i = 0$).

To prove the second inequality, consider the following schedule. Set $\underline{y}^{(0)}_{i,\langle W\rangle} = \underline{y}_{i+1,\langle W+1\rangle}, i = 1, \cdots, W$ and evaluate the sequence of constellations $\{\underline{y}^{(\ell)}_{\langle W\rangle}\}$ according to Equation (4.7). Since $\underline{y}^{(0)}_{0,\langle W\rangle} = 0 \leq \underline{y}_{1,\langle W+1\rangle}$, we must have $\underline{y}^{(1)}_{\langle W\rangle} \preceq \underline{y}^{(0)}_{\langle W\rangle}$ and by induction the sequence of constellations thus obtained is also pointwise non-increasing. Again we claim that the above mentioned schedule is admissible. This is true because we first update all W sections within the window and also the zeroth section infinitely often, and then set the boundary condition that the zeroth section also has all variables completely known. In all, every section within the window gets updated infinitely often with the correct boundary conditions. The limiting FP must hence be exactly $\underline{y}_{\langle W\rangle}$ and the second inequality claimed in the statement of the lemma follows. As in the previous case, this is intuitively true because in going from the $(i+1)^{\text{th}}$ section with window size $W+1$ to the i^{th} section with window size W, we have moved closer to the left end of the window while maintaining the distance from the right end. □

We now give some bounds on the FP erasure probabilities of individual sections within a window.

Lemma 4.4 (Bounds on FP)**.** *Consider the WD of the (d_l, d_r, γ, L) ensemble with a window of size W over a channel with erasure rate ε. The first window configuration FP $\underline{y}$ satisfies*

$$y_i \geq \left(\varepsilon(\frac{\gamma-1}{2\gamma})^{d_l-1}\right)^{\frac{(d_l-1)^j-1}{d_l-2}} y_{i+j}^{(d_l-1)^j}$$
$$y_i \leq \varepsilon\left(1-\alpha_k(1-y_{i+k})^{d_r-1}\right)^{d_l-1}$$

for $i \in [1,W], j \in [0,W+1-i], k \in [0,\gamma-1]$, where $\alpha_k = (1-\frac{(\gamma-k-1)(\gamma-k)}{2\gamma^2})^{d_r-1}$.

Proof. For the lower bound, we have

$$y_i = \varepsilon g(y_{i-\gamma+1}, \cdots, y_{i+\gamma-1}) \geq \varepsilon g(\underbrace{0, \cdots, 0}_{\gamma}, \underbrace{y_{i+1}, \cdots, y_{i+1}}_{\gamma-1}) \overset{(a)}{\geq} \varepsilon\left(\frac{\gamma-1}{2\gamma} y_{i+1}\right)^{d_l-1}$$

where (a) follows from the fact [63, Lem. 24(iii)] that

$$g(y_{i-\gamma+1}, \cdots, y_{i+\gamma-1}) \geq \bar{y}_i^{d_l-1}$$

where $\bar{y}_i = \frac{1}{\gamma^2}\sum_{j,k=0}^{\gamma-1} y_{i+j-k}$. Applying this bound recursively for $y_i, y_{i+1}, \cdots, y_{i+j-1}$, we get

$$y_i \geq \left(\varepsilon(\frac{\gamma-1}{2\gamma})^{d_l-1}\right)^{\frac{(d_l-1)^j-1}{d_l-2}} y_{i+j}^{(d_l-1)^j} \triangleq \Phi e^{-\phi_j(d_l-1)^j}$$

where $\Phi = \left(\varepsilon(\frac{\gamma-1}{2\gamma})^{d_l-1}\right)^{\frac{-1}{d_l-2}} \geq 1$ and $\phi_j = \ln\left(\frac{\Phi}{y_{i+j}}\right) \geq 0$. When $i+j = W+1$, since $y_{W+1} = 1$,

$$y_i \geq \Phi e^{-\phi(d_l-1)^{W+1-i}},\ i = 1, 2, \cdots, W$$

with $\phi = \phi_{W+1-i} = \ln\Phi$.

For the upper bound,

$$y_i \leq \varepsilon g(\underbrace{y_{i+k}, \cdots, y_{i+k}}_{\gamma+k}, \underbrace{1, \cdots, 1}_{\gamma-k-1}) \overset{(b)}{\leq} \varepsilon\left(1-\alpha_k(1-y_{i+k})^{d_r-1}\right)^{d_l-1} \triangleq f_k(\varepsilon, y_{i+k})$$

for $k \in [0, \gamma-1]$, where $\alpha_k = (1 - \frac{(\gamma-k-1)(\gamma-k)}{2\gamma^2})^{d_r-1}$. Here (b) follows from [63, Lem. 24(i)]

$$g(y_{i-\gamma+1}, \cdots, y_{i+\gamma-1}) \leq (1-(1-\bar{y}_i)^{d_r-1})^{d_l-1}.$$

Note that for $k = \gamma-1$, $f_{\gamma-1}(\varepsilon, x) = f(\varepsilon, x)$, the forward DE update equation for the (d_l, d_r)-regular ensemble. This proves the Lemma. We now discuss the utility and limitations of the upper bounds derived here.

Figure 4.2 plots the bounds $f_k(\varepsilon, y_{i+k})$ for the $(d_l = 3, d_r = 6, \gamma = 3, L)$ ensemble for two values of ε, one below and the other above the BP threshold $\varepsilon^{\text{BP}}(d_l, d_r)$. As is

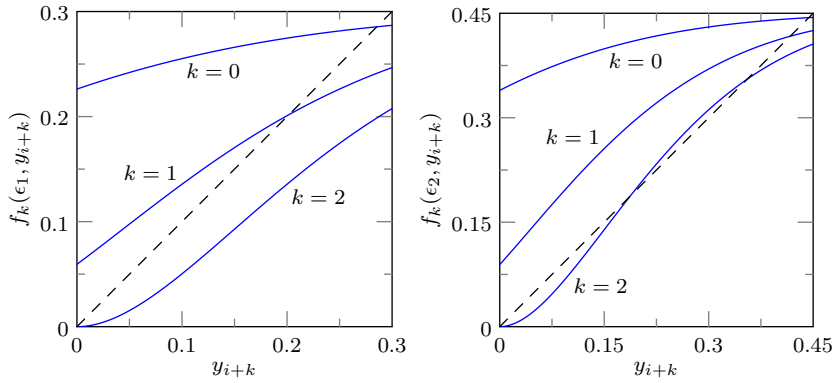

Figure 4.2: The upper bounds $f_k(\varepsilon, y_{i+k})$ for two values of ε : $\varepsilon_1 = 0.3 < \varepsilon^{\text{BP}}(d_l, d_r) \approx 0.4294 < \varepsilon_2 = 0.45$ for the $(d_l = 3, d_r = 6, \gamma = 3, L)$ ensemble.

clear from the figure, the tightest bounds are obtained for $k = \gamma-1$. Note that the bound when $k = 0$ can be recursively computed to obtain a universal upper bound y_{ub} on all the window constellation points y_i for a given (d_l, d_r, γ, L) ensemble, given by the fixed point of the equation

$$y = f_0(\varepsilon, y) = \varepsilon\Big(1 - (\frac{\gamma+1}{2\gamma})^{d_r-1}(1-y)^{d_r-1}\Big)^{d_l-1}$$

which is plotted in Figure 4.3. As can be seen from the plot, these upper bounds are only marginally tighter than the trivial upper bound of ε. In general, we can write $y_W \leq y_{\text{ub}}$ and use the other upper bounds $f_k(\cdot,\cdot)$ to obtain better bounds for other sections as follows. In the sequel, we shall write $f_{k_1,k_2,\cdots,k_c}(\varepsilon, y)$ to denote $f_{k_1}(\varepsilon, f_{k_2}(\varepsilon, \cdots f_{k_c}(\varepsilon, y)))$ and similarly define

$$f_{k^c}(\varepsilon, y) \triangleq \underbrace{f_k(\varepsilon, f_k(\varepsilon, \cdots, f_k(\varepsilon, y)))}_{c}.$$

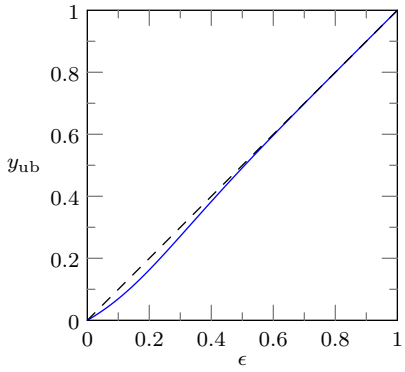

Figure 4.3: Universal upper bounds y_{ub} on the constellation points y_i as a function of ε for the $(d_l = 3, d_r = 6, \gamma = 2, L)$ ensemble. These bounds are only marginally tighter than the straightforward upper bound ε. Also, the bounds are non-decreasing in γ.

Thus, for $j = c(\gamma-1)+d, 0 \le c, 0 \le d < \gamma-1$, we can write $y_i \le f_{d,(\gamma-1)^c}(\varepsilon, y_{i+j})$. The FP value of the erasure probability of a variable node in the first section, y_1, can therefore be bounded in terms of the window size W as $y_1 \le f_{d,(\gamma-1)^c}(\varepsilon, y_{\text{ub}})$, where $c = \left\lfloor \frac{W-1}{\gamma-1} \right\rfloor, d = W-1-c(\gamma-1)$. This bounding is particularly useful when $\varepsilon \le \varepsilon^{\text{BP}}(d_l, d_r)$ when the fixed point of the $f_{\gamma-1}(\cdot,\cdot)$ upper bound is zero. It is sometimes possible that $f_{(\gamma-1)^c}(\varepsilon, y_{\text{ub}}) \le f_{d,(\gamma-1)^c}(\varepsilon, y_{\text{ub}})$, in which case we can retain the tighter upper bound $f_{(\gamma-1)^c}(\varepsilon, y_{\text{ub}})$. Figure 4.4 shows an example of the upper bound on y_1 graphically. As

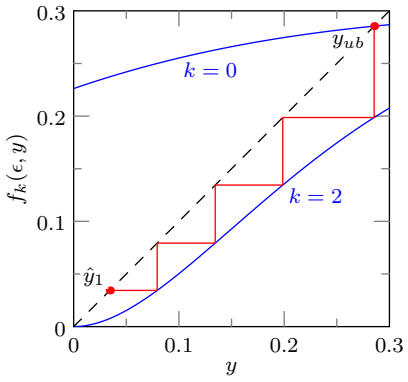

Figure 4.4: Upper bound, $\hat{y}_1 \ge y_1$, for the $(d_l = 3, d_r = 6, \gamma = 3, L)$ ensemble with a window of size $W = 9$. The channel erasure rate $\varepsilon = 0.3$.

a consequence of this upper bound, as $W \to \infty$, we have that $y_1 \to 0$ for $\varepsilon \le \varepsilon^{\text{BP}}(d_l, d_r)$. However, for $\varepsilon > \varepsilon^{\text{BP}}(d_l, d_r)$, these upper bounds are not very useful since the FP of the $f_{\gamma-1}(\cdot,\cdot)$ upper bound is non-zero (cf. Figure 4.2). □

The following shows that once the FP erasure probability of a section within the window is smaller than a certain value, it decays very quickly as we move further to the left in the window.

Lemma 4.5 (Doubly-Exponential Tail of the FP)**.** *Consider WD of the (d_l, d_r, γ, L) ensemble with a window of size W over a channel with erasure rate $\varepsilon \in (0,1)$. Let $d_l \geq 3$ and let $\underline{y}$ be the first window configuration FP of forward DE. If there exists an $i \in [W]$ such that $y_i < \delta_0 \triangleq \left((d_r-1)^{\frac{d_l-1}{d_l-2}}\right)^{-1}$, then*

$$y_{i-j(\gamma-1)} \leq \Psi e^{-\psi(d_l-1)^j}$$

where $\Psi = \delta_0 \varepsilon^{\frac{-1}{d_l-2}}$ and $\psi = \ln(\frac{\Psi}{\delta_0}) = \frac{1}{d_l-2}\ln\frac{1}{\varepsilon} > 0$.

Proof. Since the FP is non-decreasing, we have

$$\begin{aligned} y_{i-(\gamma-1)} &= \varepsilon g(y_{i-2(\gamma-1)}, \cdots, y_i) \leq \varepsilon g(y_i, y_i, \cdots, y_i) = \varepsilon(1-(1-y_i)^{d_r-1})^{d_l-1} \\ &\leq \varepsilon((d_r-1)y_i)^{d_l-1} \end{aligned} \tag{4.8}$$

which can be applied recursively to obtain

$$\begin{aligned} y_{i-j(\gamma-1)} &\leq \varepsilon^{\frac{(d_l-1)^j-1}{d_l-2}}(d_r-1)^{\frac{d_l-1}{d_l-2}((d_l-1)^j-1)} y_i^{(d_l-1)^j} \\ &< \varepsilon^{\frac{(d_l-1)^j-1}{d_l-2}}(d_r-1)^{\frac{d_l-1}{d_l-2}((d_l-1)^j-1)} \delta_0^{(d_l-1)^j} \triangleq \Psi e^{-\psi(d_l-1)^j} \end{aligned} \tag{4.9}$$

where Ψ and ψ are as defined in the statement. It is worthwhile to note that δ_0 is a lower bound on the *breakout value* for the (d_l, d_r)-regular ensemble [70]. The emergence of the breakout value in this context is not entirely unexpected since it is known that for the (d_l, d_r)-regular ensemble, the erasure probability decays double-exponentially in the number of iterations below the breakout value, and in case of spatially coupled ensembles, the counterpart for the number of iterations is the number of sections (cf. Equation (4.8)). □

We now show that the FP erasure probability of a message from a variable node in the first section, y_1, can be made small by increasing the window size W for any $\varepsilon < \varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma)$. Assuming that the window size is "large enough," we will count the number of sections, starting from the right, that have a FP erasure probability larger than a small δ for a channel erasure rate $\varepsilon = \varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma) - \Delta\varepsilon$.

Definition 4.7 (Transition Width)**.** *Consider WD of a (d_l, d_r, γ, L) spatially coupled code over a BEC of erasure rate ε. Let $\underline{y}$ be the c^{th} window configuration FP of forward DE. Then we define the transition width $\tau(\varepsilon, \delta)$ of $\underline{y}$ as*

$$\tau(\varepsilon, \delta) = |\{i \in [W] : \delta < y_i \leq 1\}|. \qquad \square$$

We first upper bound $\tau(\varepsilon, \delta) \leq \hat{\tau}(\varepsilon, \delta)$ and then claim from Lemma 4.3 that by employing a window whose size is larger than $\hat{\tau}(\varepsilon, \delta)$, we can guarantee $y_1 \leq \delta$.

Definition 4.8 (First Window Threshold)**.** *Consider WD of the (d_l, d_r, γ, L) spatially coupled ensemble with a window of size W over a BEC with erasure rate ε. The first window threshold $\varepsilon^{\mathrm{FW}}(d_l, d_r, \gamma, W, \delta)$ is defined as the supremum of channel erasure rates for which the first window configuration FP of forward DE $\underline{y}$ satisfies $y_1 \leq \delta$.* $\square$

From Definitions 4.7 and 4.8, we can see that by ensuring that $W \geq \hat{\tau}(\varepsilon, \delta)$, we can bound $\varepsilon^{\mathrm{FW}}(d_l, d_r, \gamma, W, \delta) \geq \varepsilon$.

Proposition 4.6 (Maximum Transition Width)**.** *Consider the first window configuration FP of forward DE $\underline{y}$ for the (d_l, d_r, γ, L) spatially coupled ensemble with a window of size $W < L$ for $\varepsilon \in [\frac{\varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma) + \varepsilon^{\mathrm{BP}}(d_l, d_r)}{2}, \varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma)) = \mathscr{E}$. Then,*

$$\tau(\varepsilon, \delta) \leq (\gamma - 1)\left(\mathsf{A} \ln \ln \frac{\mathsf{D}}{\delta} + \mathsf{B} \ln \frac{1}{\Delta\varepsilon} + \hat{\mathsf{C}}\right) \triangleq \hat{\tau}(\varepsilon, \delta)$$

provided $\delta \leq \delta_0$. Here $\Delta\varepsilon = \varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma) - \varepsilon$, and $\mathsf{A}, \mathsf{B}, \hat{\mathsf{C}}, \mathsf{D}$ and δ_0 are strictly positive constants that depend only on the ensemble parameters d_l, d_r and γ. $\square$

Remark 4.2. To prove the above proposition, we will use some results from [63] summarized below. We define $h(y) \triangleq f(\varepsilon, y) - y$ where

$$f(\varepsilon, y) = \varepsilon(1 - (1 - y)^{d_r - 1})^{d_l - 1},$$

the DE update equation for randomized (d_l, d_r)-regular ensembles. For ε in the interval $(\varepsilon^{\mathrm{BP}}(d_l, d_r), 1)$, the equation $h(y) = 0$ has exactly three roots in the interval $[0, 1]$, given by $0, y_u(\varepsilon)$ and $y_s(\varepsilon)$. Between 0 and $y_u(\varepsilon)$, $h(y)$ is negative, attaining a unique minimum at $y_{\min}(\varepsilon)$. Between $y_u(\varepsilon)$ and $y_s(\varepsilon)$, $h(y)$ is positive, attaining a unique maximum

at $y_{\max}(\varepsilon)$. Beyond $y_s(\varepsilon)$, $h(y)$ is negative again. Between 0 and $y_{\min}(\varepsilon)$, $h(y)$ is upper bounded by a line through the origin with slope

$$-\mu_1(\varepsilon) \triangleq \frac{h(y_{\min}(\varepsilon))}{y_{\min}(\varepsilon)},$$

i.e., the line $l(y) = -\mu_1(\varepsilon)y$. Between $y_{\min}(\varepsilon)$ and $y_u(\varepsilon)$, $h(y)$ is upper bounded by a line passing through $(y_u(\varepsilon),0)$ with a slope

$$\mu_2(\varepsilon) \triangleq \min\{\frac{-h(y_{\min}(\varepsilon))}{y_u(\varepsilon)-y_{\min}(\varepsilon)}, h'(y_u(\varepsilon))\}.$$

Between $y_u(\varepsilon)$ and $y_{\max}(\varepsilon)$, $h(y)$ is lower bounded by a line through $(y_u(\varepsilon),0)$ with a slope

$$\mu_3(\varepsilon) \triangleq \min\{\frac{h(y_{\max}(\varepsilon))}{y_{\max}(\varepsilon)-y_u(\varepsilon)}, h'(y_u(\varepsilon))\}.$$

Between $y_{\max}(\varepsilon)$ and $y_s(\varepsilon)$, $h(y)$ is lower bounded by the line through $(y_s(\varepsilon),0)$ with slope

$$-\mu_4(\varepsilon) \triangleq \max\{\frac{-h(y_{\max}(\varepsilon))}{y_s(\varepsilon)-y_{\max}(\varepsilon)}, h'(y_s(\varepsilon))\}.$$

Beyond $y_s(\varepsilon)$, $h(y)$ is upper bounded by the line through $(y_s(\varepsilon),0)$ with slope

$$-\mu_5(\varepsilon) \triangleq h'(y_s(\varepsilon)).$$

Each of the $\mu_i(\varepsilon)$'s, $i = 1,\cdots,5$, defined above is strictly positive for ε in the specified range. For a general ε, we will drop the dependence of each of these parameters on ε from the notation. When $\varepsilon = \varepsilon^* \triangleq \varepsilon^{\text{MAP}}(d_l,d_r)$, the corresponding parameters are themselves shown with $*$'s. These properties of $h(y)$ are illustrated in Figure 4.5. We can lower bound $y_{\min}$ as $y_{\min} \geq \frac{1}{d_l^2 d_r^2}$, and the slope μ_1 as $\mu_1 \geq = \frac{1}{8d_r^2} \triangleq \tilde{\mu}_1 \; \forall \; \varepsilon \in (\varepsilon^{\text{BP}}(d_l,d_r),1)$.

Further, $|h'(y)| \leq d_l d_r \; \forall \; y \in [0,1]$. We have $h'(0) = h'(1) = -1$, $h'(y_{\min}) = h'(y_{\max}) = 0$. $h''(0) = h''(\hat{y}) = h''(1) = 0$, where

$$\hat{y} = 1 - \Big(\frac{d_r - 2}{d_l d_r - d_l - d_r}\Big)^{\frac{1}{d_r-1}},$$

and

$$h''(y) \begin{cases} > 0, & y \in (0,\hat{y}) \\ < 0, & y \in (\hat{y},1). \end{cases}$$

From Rolle's Theorem, $y_{\min} \leq \hat{y} \leq y_{\max}$. We first give some simple bounds for the μ_i's defined earlier which will be useful in the proof.

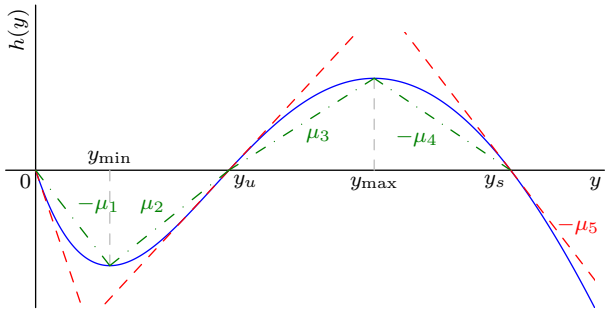

Figure 4.5: Plot of $h(y)$ (in solid blue) for the $(d_l = 3, d_r = 6)$ ensemble for $\varepsilon = 0.47$ illustrating the properties stated above. We have dropped the dependence of all the parameters on ε from the notation. The tangents at $0, y_u$ and y_s are shown as dashed red lines. The other lines used in bounding $h(y)$ are shown as dash-dotted green lines. The μ_i's, $i = 1, \cdots, 5$, are shown in the same color as the lines, whose absolute values of slopes they represent, that bound $h(y)$ in various regions.

Lemma 4.7 (μ_4, μ_5 bounds). *For* $\varepsilon \in (\varepsilon^{\text{BP}}(d_l, d_r), 1)$, *we have* $0 < \mu_4 \leq \mu_5 < 1$.

Proof. Since $h''(y) < 0$ for $y \in (y_{\max}, 1)$, $h'(y)$ monotonically decreases in this interval. Thus, $0 < \mu_5 = -h'(y_s) < 1$. From the mean value theorem, we have $h'(\xi) = -\frac{h(y_{\max})}{y_s - y_{\max}} \geq h'(y_s)$ for some $\xi \in [y_{\max}, y_s]$ so that $0 < \mu_4 = \frac{h(y_{\max})}{y_s - y_{\max}} \leq -h'(y_s) = \mu_5 < 1$. □

The values y_u and y_s are referred to as the *unstable* and *stable* fixed points (FPs) of DE for the (d_l, d_r)-regular ensemble, respectively. This is because both these values satisfy $h(y) = 0$ or $y = f(\varepsilon, y) = \varepsilon(1 - (1-y)^{d_r - 1})^{d_l - 1}$. The ε for which the FP is $y \in [0, 1]$ is given by

$$\varepsilon(y) = \frac{y}{(1 - (1-y)^{d_r - 1})^{d_l - 1}}.$$

The BP threshold is hence the smallest value of $\varepsilon(y)$, i.e., $\varepsilon^{\text{BP}}(d_l, d_r) = \min\{\varepsilon(y), y \in [0, 1]\}$. The value of y that achieves this minimum is denoted y^{BP}. Then, $\forall\ \varepsilon \in [\varepsilon^{\text{BP}}(d_l, d_r), 1]$, the unstable and stable FPs are given by

$$\begin{aligned} y_u(\varepsilon) &= y \in [0, y^{\text{BP}}] : \varepsilon(y) = \varepsilon, \\ y_s(\varepsilon) &= y \in [y^{\text{BP}}, 1] : \varepsilon(y) = \varepsilon. \end{aligned} \tag{4.10}$$

Figure 4.6 plots these stable and unstable FPs. The reason why y_s is called the stable FP (and y_u the unstable FP) can be explained through Figure 4.6. For $\varepsilon \in (\varepsilon^{\text{BP}}(d_l, d_r), 1]$,

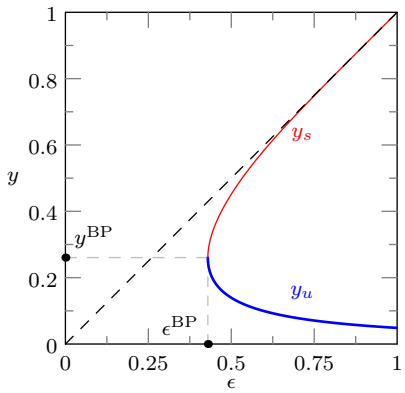

Figure 4.6: The unstable and stable FPs of DE for the $(d_l = 3, d_r = 6)$-regular ensemble as given by Equation (4.10). $y_u(\varepsilon)$ is shown as the thick blue curve and $y_s(\varepsilon)$ as the thin red curve. By definition, $y_s(\varepsilon) \geq y^{\mathrm{BP}}$ and $y_u(\varepsilon) \leq y^{\mathrm{BP}}$. y^{BP} and $\varepsilon^{\mathrm{BP}} \equiv \varepsilon^{\mathrm{BP}}(d_l = 3, d_r = 6)$ are also shown. Note that $y_s(\varepsilon^{\mathrm{BP}}) = y^{\mathrm{BP}} = y_u(\varepsilon^{\mathrm{BP}})$.

when the forward DE updates are performed, the value y monotonically decreases from 1 and converges to the first solution of the equation $h(y) = 0$, which happens to be $y_s(\varepsilon)$ for ε in this range. Therefore, performing BP always results in the FP y_s and hence the adjective "stable". Similarly for y_u, which is a solution never reached through BP, it can be shown that a small perturbation from the value of y_u will result in convergence to either y_s or 0. Therefore, y_u's are "unstable" FPs.

We can define the derivatives y_s' and y_u' of y_s and y_u, respectively, with respect to ε for $\varepsilon \in (\varepsilon^{\mathrm{BP}}(d_l, d_r), 1)$. It is easy to see that y_s' is monotonically decreasing and y_u' is monotonically increasing in ε. For details and proofs of the aforementioned properties, see [63, Appendix II], [98]. □

Proof of Proposition 4.6. Note that when W is smaller than the claimed upper bound on the transition width, the claim is trivially true; i.e., the transition width cannot be longer than the window size. However, in this case, we cannot guarantee $y_1 \leq \delta$. Hence, we will assume that W is larger than the bound. In the following, we will often use the bound

$$f(\varepsilon, y_{i-\gamma+1}) \leq y_i \leq f(\varepsilon, y_{i+\gamma-1}).$$

We now define a schedule that results in a FP window constellation that dominates the FP of the parallel schedule, $\underline{y}$, for a channel erasure rate $\varepsilon \in \mathscr{E}$. We then

upper bound the actual transition width by the transition width of the dominating FP. We generate the dominating FP in steps.

i) Set $\underline{y}^{(0)} = \underline{1}$ and evaluate the sequence of window constellations $\{\underline{y}^{(\ell)}\}$ according to Equation (4.7), but with the boundary conditions

$$y_i^{(\ell)} = \begin{cases} y_1^{(\ell)}, & i \leq 0 \\ 1, & i > W. \end{cases}$$

We have the FP in this case, $\underline{y}^A$, satisfying $\underline{y}^A \succeq \underline{y}$ by induction. Further,

$$\begin{aligned} y_1^A &= \varepsilon g(\underbrace{y_1^A, \cdots, y_1^A}_{\gamma}, y_2^A, \cdots, y_\gamma^A) \\ &\geq \varepsilon g(\underbrace{y_1^A, \cdots, y_1^A}_{2\gamma-1}) = f(\varepsilon, y_1^A), \end{aligned} \tag{4.11}$$

so that $y_1^A \geq y_s$. Note that $y_1^A \leq y_u$ cannot happen since, starting from 1, y_1^A will equal the first solution of (4.11), which from the continuity of the DE equations is guaranteed to have a solution no smaller than y_s. Starting from the right end, we now count the number of sections until $y_i^A \leq y_s + \Upsilon_A$ where we choose $\Upsilon_A = \frac{y_s^* - y_s}{2}$. Recall that the $*$-ed values correspond to $\varepsilon^* = \varepsilon^{\text{MAP}}(d_l, d_r)$. We first observe that

$$y_{W-\gamma+1}^A \leq \varepsilon g(y_W^A, \cdots, y_W^A) = f(\varepsilon, y_W^A).$$

Hence,

$$y_W^A - y_{W-\gamma+1}^A \geq y_W^A - f(\varepsilon, y_W^A) = -h(y_W^A) \geq \mu_5(y_W^A - y_s),$$

which implies that $y_{W-\gamma+1}^A - y_s \leq (1-\mu_5)(y_W^A - y_s)$. From similar reasoning, we can show that

$$\begin{aligned} y_{W-m(\gamma-1)}^A - y_s &\leq (1-\mu_5)(y_{W-(m-1)(\gamma-1)}^A - y_s) \leq (1-\mu_5)^m (y_W^A - y_s) \\ &\leq (1-\mu_5)^m (y_{\text{ub}} - y_s). \end{aligned}$$

Since $\mu_5 < 1$ from Lemma 4.7, the above difference is decreasing in m. From the definition of y_{ub} (note that this upper bound is valid even for the boundary

conditions specified here) in the proof of Lemma 4.4, it is easy to see that $y_{\text{ub}} \geq y_s$ so that the right hand side of the above chain of inequalities is non-negative. Thus, $y^A_{W-m(\gamma-1)} \leq y_s + \Upsilon_A$ if

$$m \geq \left\lceil \frac{\ln \frac{y_{\text{ub}} - y_s}{\Upsilon_A}}{\ln \frac{1}{1-\mu_5}} \right\rceil.$$

Let $\tilde{\mu}_5 = \min\{\mu_5, \varepsilon \in \mathscr{E}\}$. Then, we can write from the mean value theorem

$$\Upsilon_A = \frac{y^*_s - y_s}{2} = \frac{\varepsilon^* - \varepsilon}{2} y'_s(\hat{\varepsilon})$$

for some $\hat{\varepsilon} \in [\varepsilon, \varepsilon^*]$. We can lower bound this as

$$\Upsilon_A \geq \frac{\varepsilon^* - \varepsilon}{2} y'_s(\varepsilon^*) \geq \frac{\Delta\varepsilon}{2} y'_s(\varepsilon^*)$$

where the first inequality follows from the fact that y'_s is decreasing in ε in the interval $(\varepsilon^{\text{BP}}(d_l, d_r), 1)$ and the second from $\varepsilon^* \geq \varepsilon^{\text{BP}}(d_l, d_r, \gamma)$. Therefore this width is no more than

$$(\gamma - 1)\left(\frac{\ln \frac{2(y_{\text{ub}} - y^{\text{BP}})}{y'_s(\varepsilon^*)\Delta\varepsilon}}{\ln \frac{1}{1-\tilde{\mu}_5}} + 1 \right)$$

sections, since $y^{\text{BP}} \leq y_s$.

ii) From the definition of Υ_A, we have $y_s + \Upsilon_A = y^*_s - \Upsilon_A$. Let i_A be the largest index for which $y^A_{i_A} \leq y^*_s - \Upsilon_A$. Set $\underline{y}^{(0)} = \underline{y}^A$ and evaluate the sequence of window constellations $\{\underline{y}^{(\ell)}\}$ according to Equation (4.7) performing the updates only for those sections with indices $i < i_A$. Further, perform the updates for the channel erasure rate ε^* since we only require an upper bound on the transition width. We set the left end of the window to perform these updates to 0, i.e., $y_i = 0 \; \forall \; i \leq 0$. Let $\underline{y}^B$ denote the FP window constellation at the end of this procedure. By induction, we have $y^B_i \leq y^*_s - \Upsilon_A \; \forall \; i \leq i_A$. Also, $y^B_{i_A - 1} \geq f(\varepsilon^*, y^B_{i_A - \gamma})$ so that

$$y^B_{i_A-1} - y^B_{i_A-\gamma} \geq h(y^B_{i_A-\gamma}) \geq \mu^*_4 (y^*_s - y^B_{i_A-\gamma})$$

which implies that

$$y^*_s - y^B_{i_A-\gamma} \geq \frac{y^*_s - y^B_{i_A-1}}{1 - \mu^*_4}.$$

Similarly, it can be shown that as long as $y^B_{i_A-1-m(\gamma-1)} \geq y^*_{\max}$,

$$y^*_s - y^B_{i_A-1-m(\gamma-1)} \geq \frac{y^*_s - y^B_{i_A-1-(m-1)(\gamma-1)}}{1-\mu^*_4}$$

and by induction

$$y^*_s - y^B_{i_A-1-m(\gamma-1)} \geq \frac{y^*_s - y^B_{i_A-1}}{(1-\mu^*_4)^m} \geq \frac{\Upsilon_A}{(1-\mu^*_4)^m}.$$

Note that since $\mu^*_4 < 1$ from Lemma 4.7, the above difference is increasing in m. Thus, there are no more than

$$(\gamma-1)\Big(\frac{\ln \frac{2(y^*_s-y^*_{\max})}{y'_s(\varepsilon^*)\Delta\varepsilon}}{\ln \frac{1}{1-\mu^*_4}}+1\Big)$$

sections with $y^B_i \in [y^*_{\max}, y^*_s]$.

iii) Let $i_{\max}$ be the largest index i such that $y^B_i \leq y^*_{\max}$. We define $\Upsilon_B = \frac{y_u-y^*_u}{2}$ and count the number of sections with FP values y^B_i between $y^*_u+\Upsilon_B$ and $y^*_{\max}$. Since $y^B_{i_{\max}} \geq f(\varepsilon^*, y^B_{i_{\max}-(\gamma-1)})$ we have

$$y^B_{i_{\max}} - y^B_{i_{\max}-(\gamma-1)} \geq h(y^B_{i_{\max}-(\gamma-1)}) \geq \mu^*_3(y^B_{i_{\max}-(\gamma-1)} - y^*_u)$$

which implies that

$$y^B_{i_{\max}-(\gamma-1)} - y^*_u \leq \frac{y^B_{i_{\max}} - y^*_u}{1+\mu^*_3}.$$

Again by induction,

$$y^B_{i_{\max}-m(\gamma-1)} - y^*_u \leq \frac{y^B_{i_{\max}} - y^*_u}{(1+\mu^*_3)^m}$$

as long as $y^B_{i_{\max}-m(\gamma-1)} \geq y^*_u$. Since $\mu^*_3 > 0$, the above difference is decreasing in m, and consequently, $y^B_{i_{\max}-m(\gamma-1)} \leq y^*_u + \Upsilon_B$ if

$$m \geq \left\lceil \frac{\ln \frac{y^*_{\max}-y^*_u}{\Upsilon_B}}{\ln(1+\mu^*_3)} \right\rceil.$$

Writing

$$\Upsilon_B = \frac{y_u - y^*_u}{2} = -\frac{\varepsilon^* - \varepsilon}{2} y'_u(\check{\varepsilon})$$

from the mean value theorem for some $\breve{\varepsilon} \in [\varepsilon, \varepsilon^*]$, we can bound this as

$$\Upsilon_B \geq -\frac{\varepsilon^* - \varepsilon}{2} y_u'(\varepsilon^*) \geq -\frac{\Delta\varepsilon}{2} y_u'(\varepsilon^*)$$

where the first inequality follows because $-y_u'$ is decreasing in ε in the interval $(\varepsilon^{\mathrm{BP}}(d_l, d_r), 1)$ and the second because $\varepsilon^* \geq \varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma)$. This implies that there are no more than

$$(\gamma - 1)\Big(\frac{\ln \frac{2(y^*_{\max} - y^*_u)}{-y_u'(\varepsilon^*)\Delta\varepsilon}}{\ln(1+\mu_3^*)} + 1\Big)$$

sections with FP values between $y_u^* + \Upsilon_B$ and $y^*_{\max}$.

iv) From the definition of Υ_B, we have $y_u^* + \Upsilon_B = y_u - \Upsilon_B$. Let i_B be the largest index i such that $y_i^B \leq y_u - \Upsilon_B$. Set $\underline{y}^{(0)} = \underline{y}^B$ and evaluate $\{\underline{y}^{(\ell)}\}$ according to Equation (4.7) performing the updates only for sections with indices $i \leq i_B$ with channel erasure rate ε. Again we set the left end of the window to 0 while performing the updates. Denote the FP obtained at the end of this procedure as $\underline{y}^C$. Clearly, $y_i^C \leq y_u - \Upsilon_B \; \forall \, i \leq i_B$. Since $y^C_{i_B-\gamma+1} \leq f(\varepsilon, y^C_{i_B})$, we have

$$y^C_{i_B} - y^C_{i_B-\gamma+1} \geq -h(y^C_{i_B}) \geq \mu_2(y_u - y^C_{i_B})$$

so that

$$y_u - y^C_{i_B-\gamma+1} \geq (1+\mu_2)(y_u - y^C_{i_B}).$$

From similar reasoning, as long as $y^C_{i_B-m(\gamma-1)} \geq y_{\min}$,

$$y_u - y^C_{i_B-m(\gamma-1)} \geq (1+\mu_2)(y_u - y^C_{i_B-(m-1)(\gamma-1)})$$

and by induction

$$y_u - y^C_{i_B-m(\gamma-1)} \geq (1+\mu_2)^m (y_u - y^C_{i_B}) \geq (1+\mu_2)^m \Upsilon_B.$$

Since $\mu_2 > 0$, the above difference is increasing in m. By letting $\tilde{\mu}_2 = \min\{\mu_2, \varepsilon \in \mathscr{E}\}$ and noting that

$$y_u - y_{\min} \leq y^{\mathrm{BP}} - y_{\min} \leq y^{\mathrm{BP}} - \frac{1}{d_l^2 d_r^2},$$

we have that there are no more than

$$(\gamma - 1)\Big(\frac{\ln \frac{2(y^{\mathrm{BP}} - \frac{1}{d_l^2 d_r^2})}{-y_u'(\varepsilon^*)\Delta\varepsilon}}{\ln(1+\tilde{\mu}_2)} + 1\Big)$$

sections with FP values in the interval $[y_{\min}, y_u]$.

v) Let i_C be the largest index i such that $y_i^C \leq y_{\min}$. Proceeding as above, we have

$$y_{i_C}^C - y_{i_C-\gamma+1}^C \geq -h(y_{i_C}^C) \geq \mu_1 y_{i_C}^C$$

and by induction,

$$y_{i_C-m(\gamma-1)}^C \leq (1-\mu_1)^m y_{i_C}^C.$$

Thus, between δ_0 and $y_{\min}$, there are no more than

$$(\gamma-1)\Big(\frac{\ln \frac{y^{\mathrm{BP}}}{\delta_0}}{\ln \frac{1}{1-\tilde{\mu}_1}}+1\Big)$$

sections with FP values in the interval $[\delta_0, y_{\min}]$, since $y_{\min} \leq y^{\mathrm{BP}}$ and $\mu_1 \geq \tilde{\mu}_1$.

vi) Let i_D be the largest index i such that $y_i^C \leq \delta_0$. From Lemma 4.5, we know that the tail decays doubly-exponentially for $i \leq i_D$. From (4.9), we have $y_{i_D-m(\gamma-1)}^C \leq \Psi e^{-\psi(d_l-1)^m} \leq \tilde{\Psi} e^{-\tilde{\psi}(d_l-1)^m}$, where

$$\Psi = \delta_0 \varepsilon^{-\frac{1}{d_l-2}} \leq \delta_0 \big(\frac{\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma)+\varepsilon^{\mathrm{BP}}(d_l,d_r)}{2}\big)^{-\frac{1}{d_l-2}} \triangleq \tilde{\Psi}$$

and

$$\psi = \frac{1}{d_l-2}\ln\frac{1}{\varepsilon} \geq \frac{1}{d_l-2}\ln\frac{1}{\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma)} \triangleq \tilde{\psi}.$$

Thus there are no more than

$$(\gamma-1)\Big(\frac{1}{\ln(d_l-1)}\ln\ln\frac{\tilde{\Psi}}{\delta}+\frac{\ln\frac{1}{\tilde{\psi}}}{\ln(d_l-1)}+1\Big)$$

sections with $y_i^C \in [\delta, \delta_0]$.

Finally, collecting all these terms, we conclude that the transition width of the FP obtained from the procedure highlighted in the steps i) through vi) is upper bounded by

$$\hat{\tau}(\varepsilon,\delta) = (\gamma-1)\Big(\mathsf{A}\ln\ln\frac{\mathsf{D}}{\delta}+\mathsf{B}\ln\frac{1}{\Delta\varepsilon}+\hat{\mathsf{C}}\Big)$$

where the constants A, B and D are as follows

$$\mathsf{A} = \frac{1}{\ln(d_l - 1)},$$

$$\mathsf{B} = \frac{1}{\ln(1+\tilde{\mu}_2)} + \frac{1}{\ln(1+\mu_3^*)} + \frac{1}{\ln(\frac{1}{1-\mu_4^*})} + \frac{1}{\ln(\frac{1}{1-\tilde{\mu}_5})},$$

$$\hat{\mathsf{C}} = \frac{\ln \frac{1}{\tilde{\Psi}}}{\ln(d_l-1)} + \frac{\ln \frac{y^{\mathrm{BP}}}{\delta_0}}{\ln \frac{1}{1-\tilde{\mu}_1}} + \frac{\ln \frac{2(y^{\mathrm{BP}} - \frac{1}{d_l^2 d_r^2})}{-y_u'(\varepsilon^*)}}{\ln(1+\tilde{\mu}_2)} + \frac{\ln \frac{2(y^*_{\max} - y^*_u)}{-y_u'(\varepsilon^*)}}{\ln(1+\mu_3^*)} + \frac{\ln \frac{2(y_s^* - y^*_{\max})}{y_s'(\varepsilon^*)}}{\ln \frac{1}{1-\mu_4^*}} + \frac{\ln \frac{2(y_{\mathrm{ub}} - y^{\mathrm{BP}})}{y_s'(\varepsilon^*)}}{\ln \frac{1}{1-\tilde{\mu}_5}} + 6,$$

and $\mathsf{D} = \tilde{\Psi}$. Note that these constants depend only on the ensemble parameters d_l, d_r and γ. Since it is clear that the FP obtained through the procedure in steps i) through vi) above dominates pointwise the first window configuration FP of forward DE with a window of size W for channel erasure rate ε, we can guarantee that the transition width is upper bounded by the above expression. This completes the proof. □

Note that the above result means that the smallest window size that guarantees $y_1 \leq \delta$ for a channel erasure rate $\frac{\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma)+\varepsilon^{\mathrm{BP}}(d_l,d_r)}{2}$ is

$$\begin{aligned}\hat{W}_{\min}(\delta) &= (\gamma-1)\Big(\mathsf{A}\ln\ln\frac{\mathsf{D}}{\delta} + \mathsf{B}\ln\frac{1}{\Delta\varepsilon_{\max}} + \hat{\mathsf{C}}\Big) \\ &= \hat{\tau}\Big(\frac{\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma)+\varepsilon^{\mathrm{BP}}(d_l,d_r)}{2}, \delta\Big)\end{aligned}$$

where $\Delta\varepsilon_{\max} = \frac{\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma)-\varepsilon^{\mathrm{BP}}(d_l,d_r)}{2}$. When $W \geq \hat{W}_{\min}(\delta)$, we have

$$\varepsilon^{\mathrm{FW}}(d_l,d_r,\gamma,W,\delta) \geq \varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma) - e^{-\frac{1}{\mathsf{B}}(\frac{W}{\gamma-1} - \mathsf{A}\ln\ln\frac{\mathsf{D}}{\delta} - \hat{\mathsf{C}})}. \tag{4.12}$$

Discussion : We restricted $\varepsilon \in \mathscr{E}$ in Proposition 4.6 to obtain constants that are independent of ε. As can be seen from the proof of the proposition, these constants are dependent on ε, unless each is optimized in the range $\mathscr{E}$. As we let the minimum ε in $\mathscr{E}$ approach $\varepsilon^{\mathrm{BP}}(d_l,d_r)$, the constants in the expression for $\hat{\tau}(\varepsilon,\delta)$ blow up and the upper bound will be useless. It is therefore necessary to keep the minimum of $\varepsilon \in \mathscr{E}$ strictly larger than $\varepsilon^{\mathrm{BP}}(d_l,d_r)$ and the value chosen in the above was motivated by our intent to ensure that the first window threshold was closer to $\varepsilon^{\mathrm{BP}}(d_l,d_r,\gamma)$ than to $\varepsilon^{\mathrm{BP}}(d_l,d_r)$.

Note that the increase in the upper bound for $\tau(\varepsilon,\delta)$ with decrease in ε is purely an artifact of the upper bounding technique we have employed; i.e., it is obvious that as we decrease ε, $\tau(\varepsilon,\delta)$ also decreases. □

4.3.2 c^{th} Window Configuration, $1 < c \leq L$

We now evaluate the performance of the windowed decoding scheme when the window has slid certain number of sections from the left end of the code. We arrive at conditions under which $\hat{x}$ is guaranteed to be smaller than δ while operating with a window of size W. We start by establishing a property of $\hat{x}$.

Lemma 4.8 (FP Equation Involving $\hat{x}$)**.** *Consider the function $\Omega(\underline{y})$ where*

$$\Omega(y_i) = \begin{cases} \pi(y_i), & i < 1 \\ \varepsilon g(\pi(y_{i-\gamma+1}), \cdots, \pi(y_{i+\gamma-1})), & i \in [W] \\ 1, & i > W \end{cases}$$

where

$$\pi(y_i) = \begin{cases} y_1, & i < 1 \\ y_i, & i \in [W] \\ 1, & i > W. \end{cases}$$

Then there exists a solution $\underline{\omega}$ to the equation $\underline{y} = \Omega(\underline{y})$ such that $\omega_1 = \hat{x}$. Moreover, $\underline{\omega}$ is the smallest such constellation, i.e., if $\underline{\hat{\omega}} = \Omega(\underline{\hat{\omega}})$, then $\underline{\hat{\omega}} \succeq \underline{\omega}$.

Proof. We have

$$\begin{aligned} \hat{x} = x_{\infty,\{\infty\}} = y^{(\infty)}_{1,\{\infty\}} &\overset{(4.5)}{=} \varepsilon g(y^{(\infty)}_{-\gamma+2,\{\infty\}}, \cdots, y^{(\infty)}_{0,\{\infty\}}, y^{(\infty)}_{1,\{\infty\}}, \cdots, y^{(\infty)}_{\gamma,\{\infty\}}) \\ &= \varepsilon g(\underbrace{x_{\infty,\{\infty\}}, \cdots, x_{\infty,\{\infty\}}}_{\gamma}, y^{(\infty)}_{2,\{\infty\}}, \cdots, y^{(\infty)}_{\gamma,\{\infty\}}) \\ &= \varepsilon g(\underbrace{\hat{x}, \cdots, \hat{x}}_{\gamma}, y^{(\infty)}_{2,\{\infty\}}, \cdots, y^{(\infty)}_{\gamma,\{\infty\}}). \end{aligned}$$

Hence, if we define $\underline{\omega}$ as

$$\omega_i = \begin{cases} \hat{x} = y^{(\infty)}_{1,\{\infty\}}, & i \leq 1 \\ y^{(\infty)}_{i,\{\infty\}}, & i > 1, \end{cases}$$

then it is clear that $\underline{\omega} = \pi(\underline{\omega}) = \Omega(\underline{\omega})$.

Note that any fixed point $\underline{\hat{\omega}}$ of the function $\Omega(\cdot)$ has to satisfy $\underline{\hat{\omega}} \succeq \underline{y}^{(\infty)}_{\{1\}}$ for the same channel erasure rate $\varepsilon \in [0,1]$ from the monotonicity of $g(\cdot)$. In particular, $\underline{\omega} \succeq \underline{y}^{(\infty)}_{\{1\}}$. From the continuity of the DE equations in Definition 4.4, it follows that $\underline{\omega}$ is the least solution to the equation $\underline{y} = \Omega(\underline{y})$, since it is the limiting constellation of the sequence of non-decreasing constellations $\{\underline{y}^{(\infty)}_{\{n\}}\}_{n=1}^{\infty}$. □

The following proposition forms the central argument in the proof of Theorem 4.2. Using the bound on the maximum transition width from Proposition 4.6, we obtain an upper bound on $\hat{x}$ for a given window size W and erasure rate $\varepsilon \in \mathscr{E}$. From this, we arrive at a lower bound for ε that guarantees $\hat{x} \leq \delta$ when δ is an arbitrarily chosen value smaller than δ_* (which depends only on d_l, d_r) and the window size is larger than $W_{\min}(\delta)$ (which depends only on the code parameters d_l, d_r, γ and the decoder parameter δ). This gives us our lower bound on the WD threshold.

Proposition 4.9 (WD & FW Thresholds)**.** *Consider WD of the (d_l, d_r, γ, L) spatially coupled ensemble with a window of size $W \geq W_{\min}(\delta) = \hat{W}_{\min}(\delta) + \gamma - 1$ over a BEC with erasure rate ε. Then, we have*

$$\varepsilon^{\mathrm{WD}}(d_l, d_r, \gamma, W, \delta) \geq \left(1 - \frac{d_l d_r}{2}\delta^{\frac{d_l-2}{d_l-1}}\right)\varepsilon^{\mathrm{FW}}(d_l, d_r, \gamma, W-\gamma+1, \delta)$$

provided $\delta < \delta_* = \left(\frac{2}{d_l d_r}\right)^{\frac{d_l-1}{d_l-2}}$, where $\varepsilon^{\mathrm{FW}}(d_l, d_r, \gamma, W, \delta)$ is the first window threshold.

Proof. We start with the first window configuration FP of forward DE when the channel erasure rate is ε and show that this FP dominates the c^{th} window configuration FP of forward DE for every c for a smaller channel erasure rate $\upsilon \leq \varepsilon$. To prove this, it suffices to show that the FP $\underline{\omega}$ defined in Lemma 4.8 for channel erasure rate υ is dominated pointwise by the first window configuration FP for channel erasure rate ε. This establishes υ as being a lower bound on the WD threshold $\varepsilon^{\mathrm{WD}}$.

Set $\underline{y}^{(0)} = \underline{y}^{(\infty)}_{\{1\}}$, the first window configuration FP of forward DE for channel erasure rate ε. Evaluate $\{\underline{y}^{(\ell)}\}_{\ell=1}^{\infty}$ according to $\underline{y}^{(\ell)} = \Omega(\underline{y}^{(\ell-1)})$ where $\Omega(\cdot)$ is as defined

in Lemma 4.8, but for channel erasure rate $\upsilon = \varepsilon - \Delta\varepsilon$. Then, the following are true:

$$y_i^{(0)} = \begin{cases} \varepsilon g(\underbrace{0,\cdots,0}_{\gamma-i}, y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)}), & 1 \le i < \gamma \\ \varepsilon g(y_{i-\gamma+1}^{(0)},\cdots,y_{i+\gamma-1}^{(0)}), & \gamma \le i \le W \end{cases}$$

and

$$y_i^{(1)} = \begin{cases} \upsilon g(\underbrace{y_1^{(0)},\cdots,y_1^{(0)}}_{\gamma-i}, y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)}), & 1 \le i < \gamma \\ \upsilon g(y_{i-\gamma+1}^{(0)},\cdots,y_{i+\gamma-1}^{(0)}), & \gamma \le i \le W. \end{cases}$$

For $\gamma \le i \le W$,

$$y_i^{(1)} = \frac{\upsilon}{\varepsilon} y_i^{(0)} = \frac{\varepsilon - \Delta\varepsilon}{\varepsilon} y_i^{(0)} \le y_i^{(0)}. \tag{4.13}$$

Let us write

$$g_i(\sigma, y_1,\cdots,y_{i+\gamma-1}) \triangleq g(\underbrace{\sigma,\cdots,\sigma}_{\gamma-i}, y_1,\cdots,y_{i+\gamma-1})$$

and

$$G_i(\varepsilon,\sigma,y_1,\cdots,y_{i+\gamma-1}) \triangleq \varepsilon g_i(\sigma,y_1,\cdots,y_{i+\gamma-1})$$

for $1 \le i < \gamma$. For i in this range, consider

$$\begin{aligned} y_i^{(0)} - y_i^{(1)} &= G_i(\varepsilon,0,y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)}) - G_i(\upsilon,y_1^{(0)},y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)}) \\ &= \Big[G_i(\varepsilon,0,y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)}) - G_i(\upsilon,0,y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)})\Big] \\ &\quad - \Big[G_i(\upsilon,y_1^{(0)},y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)}) - G_i(\upsilon,0,y_1^{(0)},\cdots,y_{i+\gamma-1}^{(0)})\Big] \\ &\overset{(a)}{=} \Delta\varepsilon \frac{\partial G_i}{\partial \varepsilon}\Big|_{\xi,\sigma=0,\underline{y}^{(0)}} - y_1^{(0)} \frac{\partial G_i}{\partial \sigma}\Big|_{\upsilon,\zeta,\underline{y}^{(0)}} \end{aligned} \tag{4.14}$$

where $\xi \in [\upsilon,\varepsilon]$ and $\zeta \in [0, y_1^{(0)}]$. Here, (a) follows from the mean value theorem. We have

$$\frac{\partial G_i}{\partial \varepsilon}\Big|_{\xi,\sigma=0,\underline{y}^{(0)}} = g_i(\sigma,y_1,\cdots,y_{i+\gamma-1})\Big|_{\xi,\sigma=0,\underline{y}^{(0)}} = \frac{y_i^{(0)}}{\varepsilon}.$$

Since $\frac{\partial G_i}{\partial \sigma} = \varepsilon \frac{\partial g_i}{\partial \sigma}$, we focus on g_i. Expanding out the expression for g_i, it can be written as

$$\begin{aligned} g_i = \Big[1 - \frac{1}{\gamma}\Big((\alpha_{i,1} - \frac{\gamma-i}{\gamma}\sigma)^{d_r-1} + \cdots \\ + (\alpha_{i,\gamma-i} - \frac{1}{\gamma}\sigma)^{d_r-1} + \alpha_{i,\gamma-i+1}^{d_r-1} + \cdots + \alpha_{i,\gamma}^{d_r-1}\Big)\Big]^{d_l-1} \end{aligned}$$

where

$$\alpha_{i,j+1} = \begin{cases} 1 - \frac{1}{\gamma}\sum_{c=1}^{i+j} y_c, & 0 \le j \le \gamma - i - 1 \\ 1 - \frac{1}{\gamma}\sum_{c=i+j-\gamma+1}^{i+j} y_c, & \gamma - i \le j \le \gamma - 1. \end{cases}$$

Clearly, $0 \le \alpha_{i,j+1} \le 1 \; \forall \; 1 \le i < \gamma, 0 \le j \le \gamma - 1$. Therefore,

$$\begin{aligned} \frac{\partial g_i}{\partial \sigma} &= \frac{(d_l-1)(d_r-1)}{\gamma} g_i^{\frac{d_l-2}{d_l-1}} \Big[(\alpha_{i,1} - \frac{\gamma-i}{\gamma}\sigma)^{d_r-2} \frac{\gamma-i}{\gamma} + \cdots + (\alpha_{i,\gamma-i} - \frac{1}{\gamma}\sigma)^{d_r-2} \frac{1}{\gamma} \Big] \\ &\overset{(a)}{\le} \frac{d_l d_r}{\gamma^2} g_i^{\frac{d_l-2}{d_l-1}} \frac{(\gamma-i)(\gamma-i+1)}{2} \le \frac{d_l d_r}{2} g_i^{\frac{d_l-2}{d_l-1}}. \end{aligned}$$

Here, (a) holds because $0 \le (\alpha_{i,j+1} - \frac{\gamma-i-j}{\gamma}\sigma) \le 1 \; \forall \; 0 \le j \le \gamma - i - 1$. This implies that

$$\begin{aligned} \frac{\partial G_i}{\partial \sigma}\Big|_{\upsilon,\zeta,\underline{y}^{(0)}} &\le \frac{d_l d_r}{2} g_i^{\frac{d_l-2}{d_l-1}} \varepsilon \Big|_{\upsilon,\zeta,\underline{y}^{(0)}} = \frac{d_l d_r}{2} \upsilon^{\frac{1}{d_l-1}} \Big(\upsilon g(\underbrace{\zeta,\cdots,\zeta}_{\gamma-i}, y_1^{(0)}, \cdots, y_{i+\gamma-1}^{(0)}) \Big)^{\frac{d_l-2}{d_l-1}} \\ &\overset{(b)}{\le} \frac{d_l d_r}{2} \upsilon^{\frac{1}{d_l-1}} \Big(\upsilon g(\underbrace{y_1^{(0)},\cdots,y_1^{(0)}}_{\gamma-i}, y_1^{(0)}, \cdots, y_{i+\gamma-1}^{(0)}) \Big)^{\frac{d_l-2}{d_l-1}} \\ &= \frac{d_l d_r}{2} \upsilon^{\frac{1}{d_l-1}} (y_i^{(1)})^{\frac{d_l-2}{d_l-1}} \\ &\overset{(c)}{\le} \frac{d_l d_r}{2} \upsilon^{\frac{1}{d_l-1}} (y_\gamma^{(1)})^{\frac{d_l-2}{d_l-1}} \le \frac{d_l d_r}{2} (y_\gamma^{(1)})^{\frac{d_l-2}{d_l-1}} \overset{(4.13)}{\le} \frac{d_l d_r}{2} y_\gamma^{(0)^{\frac{d_l-2}{d_l-1}}}. \end{aligned}$$

Here, the inequality labeled (b) is true because $\zeta \le y_1^{(0)}$, (c) follows from the observation that $y_i^{(1)} \le y_{i+1}^{(1)}, i \ge 1$, which is in turn true since $\underline{y}^{(0)}$ and $\pi(\underline{y}^{(0)})$ were non-decreasing. Substituting back in (4.14), we have for $1 \le i < \gamma$

$$y_i^{(0)} - y_i^{(1)} \ge \Delta\varepsilon \frac{y_1^{(0)}}{\varepsilon} - y_1^{(0)} \frac{d_l d_r}{2} y_\gamma^{(0)^{\frac{d_l-2}{d_l-1}}}.$$

Thus if $\frac{\Delta\varepsilon}{\varepsilon} \ge \frac{d_l d_r}{2} y_\gamma^{(0)^{\frac{d_l-2}{d_l-1}}}$, $y_i^{(0)} \ge y_i^{(1)} \; \forall \; i \ge 1$, and hence $\pi(\underline{y}^{(0)})$ dominates $\underline{y}^{(1)}$ pointwise. Recall that

$$\pi(y_i) = \begin{cases} y_1, & i < 1 \\ y_i, & i \ge 1. \end{cases}$$

It therefore follows by induction that the limiting constellation $\underline{y}^{(\infty)}$ exists, and is also dominated by $\pi(\underline{y}^{(0)})$. It is clear that $\underline{y}^{(\infty)}$ satisfies $\underline{y}^{(\infty)} = \Omega(\underline{y}^{(\infty)})$. From Lemma 4.8, $\underline{y}^{(\infty)} \succeq \underline{\omega}$ and hence $y_1^{(\infty)} \ge \hat{x}$.

If the window size is chosen to be $W \geq \hat{W}_{\min}(\delta) + \gamma - 1 \triangleq W_{\min}(\delta)$, then for the first window, we can guarantee $y_{\gamma}^{(0)} \leq \delta$ for some $\delta < \delta_0$ for all channel erasure rates smaller than $\varepsilon^{\mathrm{FW}} \equiv \varepsilon^{\mathrm{FW}}(d_l, d_r, \gamma, W - \gamma + 1, \delta)$. From the above argument, it follows that we can ensure $\hat{x} \leq \delta$ for all erasure rates smaller than $\varepsilon^{\mathrm{FW}}\left(1 - \frac{d_l d_r}{2}\delta^{\frac{d_l-2}{d_l-1}}\right)$. As long as

$$\delta < \delta_* \triangleq \left(\frac{2}{d_l d_r}\right)^{\frac{d_l-1}{d_l-2}} < \left(\frac{1}{d_r - 1}\right)^{\frac{d_l-1}{d_l-2}} = \delta_0,$$

this erasure rate is a non-trivial lower bound on the WD threshold $\varepsilon^{\mathrm{WD}}$. □

From Proposition 4.9 and Equation (4.12), we immediately have that

$$\varepsilon^{\mathrm{WD}}(d_l, d_r, \gamma, W, \delta) \geq \left(1 - \frac{d_l d_r}{2}\delta^{\frac{d_l-2}{d_l-1}}\right)\left(\varepsilon^{\mathrm{BP}}(d_l, d_r, \gamma) - e^{-\frac{1}{\mathsf{B}}\left(\frac{W-\gamma+1}{\gamma-1} - \mathsf{A}\ln\ln\frac{\mathsf{D}}{\delta} - \hat{\mathsf{C}}\right)}\right)$$

provided $W \geq W_{\min}(\delta)$. By making the substitution $\mathsf{C} = \hat{\mathsf{C}} + 1$, we see that this proves Theorem 4.2.

4.4 Experimental Results

In this section, we give results obtained by simulating windowed decoding of finite-length spatially coupled codes over the binary erasure channel. The code used for simulation was generated randomly by fixing the parameters $M = 1024$, $d_l = 3, d_r = 6$, with coupling length $\gamma = 3$ and chain length $L = 64$. The blocklength of the code was hence $n = ML = 65,536$ and the rate was $R \approx 0.484375$. From Table 4.1, the BP threshold for the ensemble to which this code belongs is $\varepsilon^{\mathrm{BP}}(d_l = 3, d_r = 6, \gamma = 3, L = 64) \approx 0.487514$.

Figure 4.7 shows the bit erasure rates achieved by using windows of length $W = 4, 6, 8$, i.e., the number of bits within each window was $WM = 4096, 6144$ and 8192 respectively. From the figure, it is clear that good performance can be obtained for a wide range of channel erasure rates even for small window lengths, e.g., $W = 6, 8$. In performing the simulations above, we let the decoders (BP and WD) run for as many iterations as possible, until the decoder could solve for no further bits. For the windowed decoder, this meant that within each window configuration, the decoder was allowed to run until it could solve no further bits within the window. Figure 4.8 plots the average

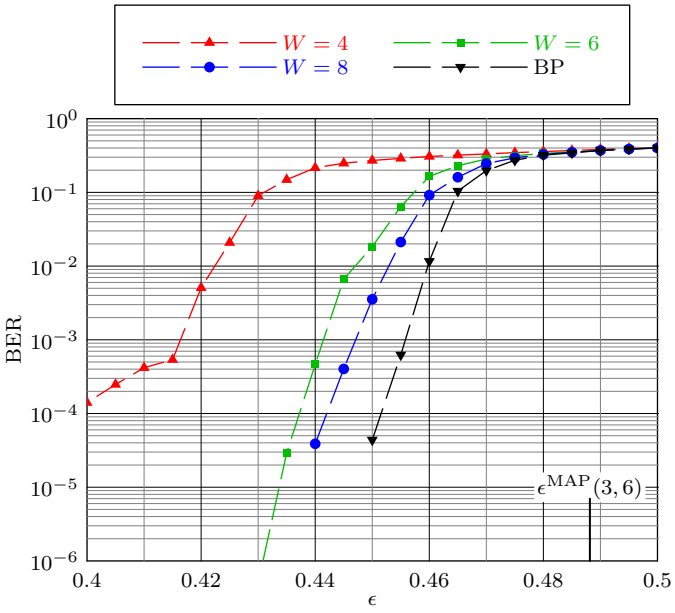

Figure 4.7: Bit erasure probability of the $(d_l = 3, d_r = 6, \gamma = 3, L = 64)$ spatially coupled code with $M = 1024$ achieved with a windowed decoder of window sizes $W = 4, 6$ and 8.

number of iterations for the BP decoder and the average number of iterations within each window configuration times the chain length (which corresponds to the average number of iterations) for the WD. We can see that for randomly chosen spatially coupled codes, a modest reduction in complexity is possible by using the windowed decoder in the waterfall region. Interestingly, the average number of iterations required per window configuration is independent of the chain length below certain channel erasure rates. The number of iterations required decreases beyond a certain value of ε because for these higher erasure rates, the decoder is no longer able to decode and gets stuck quickly. Although the smaller window sizes have a large reduction in complexity and a decent BER performance, the block erasure rate performance can be fairly bad, e.g., for the window of size 4, the block erasure rate was 1 in the range of erasure rates considered in Figure 4.7. However, the block erasure rate improves drastically with increasing window size—for the window of size 8, the block erasure rate at $\varepsilon = 0.44$ was $\approx 6.3 \times 10^{-4}$.

The above illustration suggests that for good performance with reduced complexity via windowed decoding, careful code design is necessary. For a certain variety of spatially coupled codes—protograph-based LDPC convolutional codes—certain design

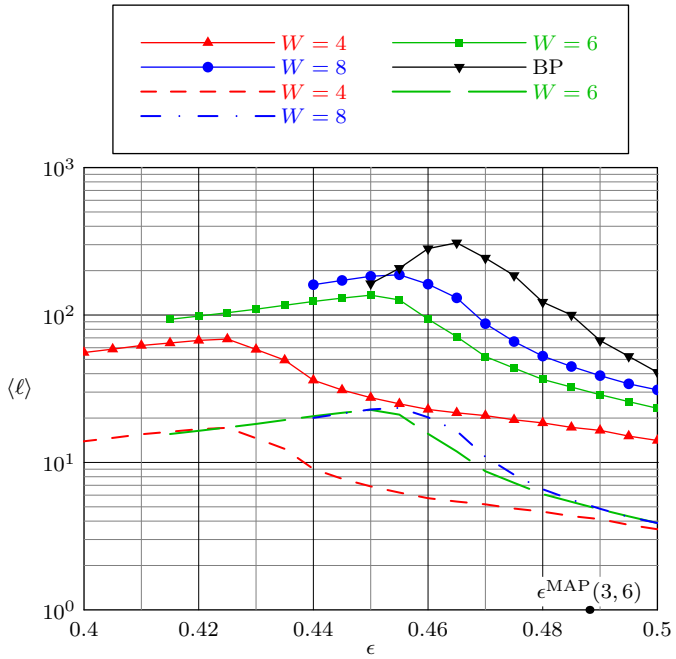

Figure 4.8: Average number of iterations $\langle \ell \rangle$ for BP and WD as a function of the channel erasure rate is shown for each window size in solid lines. For the WD, we show in dashed lines, the average number of iterations required within each window configuration.

rules for good performance with windowed decoding were given in [48] (See Chapter 5), and ensembles with good performance for a wide range of window sizes (including window sizes as small as γ) over erasure channels with and without memory were constructed. For these codes constructed using PEG [45] and ACE [113] techniques, not only can the error floor be lowered but also the performance of a medium-sized windowed decoder with fixed number of iterations can be made to be very close to that of the BP decoder [48]. It is for such codes that the windowed decoder is able to attain very good performance with significant reduction in complexity and decoding latency.

4.5 Conclusions

We considered a windowed decoding (WD) scheme for decoding spatially coupled codes that has smaller complexity and latency than the BP decoder. We analyzed the asymptotic performance limits of such a scheme by defining WD thresholds for meeting target erasure rates. We gave a lower bound on the WD thresholds and showed

that these thresholds are guaranteed to approach the BP threshold for the spatially coupled code at least exponentially in the window size. Through density evolution, we showed that, in fact, the WD thresholds approach the BP threshold much faster than is guaranteed by the lower bound proved analytically. Since the BP thresholds for spatially coupled codes are themselves close to the MAP threshold, WD gives us an efficient way to trade off complexity and latency for decoding performance approaching the optimal MAP performance. Since the MAP decoder is capacity-achieving as the degrees of variables and checks are increased, similar performance is achievable through a WD scheme for a target erasure floor.

Through simulations, we showed that WD is a viable scheme for decoding finite-length spatially coupled codes and that even for small window sizes, good performance is attainable for a wide range of channel erasure rates. However, the complexity reduction for randomly constructed spatially coupled codes is not as significant as that obtained for protograph-based LDPC convolutional codes with a large girth. Thus, characterizing good spatially coupled codes within the ensemble of randomly coupled codes is a question that remains.

The WD scheme was analyzed here for the BEC and, therefore, the superior performance of these codes and the low complexity and latency of the WD scheme make these attractive for applications in coding over upper layers of the internet protocol. Furthermore, the same scheme can be employed for decoding spatially coupled codes over any channel. However, for channels that introduce errors apart from erasures, the WD scheme can suffer from error propagation. Analysis of the WD scheme and providing performance guarantees over such channels will play a key role in making spatially coupled codes and the WD scheme practical.

Acknowledgement

This chapter is a reprint of the paper “Windowed Decoding of Spatially Coupled Codes”, A.R. Iyengar, P. H. Siegel, R. L. Urbanke, J. K. Wolf; which has been submitted for publication in the IEEE Transactions on Information Theory. The dissertation author was the primary investigator and author of this paper.

Chapter 5

Windowed Decoding of Protograph-based LDPC Convolutional Codes

In this chapter, we will consider an application of the windowed decoder introduced in the previous chapter to decode a class of LDPC codes. We will consider decoding LDPC Convolutional Codes (LDPC-CC) that are constructed from protographs. As noted in the previous chapter, whereas irregular LDPC block codes have also been shown to have BP thresholds close to capacity [98], the advantage with convolutional counterparts is that good performance is achieved by relatively simple regular ensembles. Also, the construction of finite-length codes from LDPC-CC ensembles can be readily optimized to ensure desirable properties, e.g., large girths and fewer cycles, using well-known techniques of LDPC code design. Similar decoding schemes have been proposed in [68, 73]. However, the aim in these papers was not to reduce the decoding latency or complexity. When used to decode terminated (block) LDPC-CC, the windowed decoder provides a simple, yet efficient way to trade-off decoding performance for reduced latency. Moreover, the proposed scheme provides the flexibility to set and change the decoding latency on the fly. This proves to be an extremely useful feature when the scheme is used to decode codes over upper layers of the internet protocol.

Our contributions here are to study the requirements of LDPC-CC ensembles for good performance over erasure channels with windowed decoding (WD). We are

interested in identifying characteristics of ensembles that present a good performance-latency trade-off. Further we seek to find such ensembles that are able to withstand not just random erasures but also long bursts of erasures. We reiterate that we will be interested in designing ensembles that have the aforementioned properties, rather than designing codes themselves.

The rest of the chapter is organized as follows. Section 5.1 introduces LDPC convolutional codes . In Section 5.2 we briefly describe the application of the windowed decoder for LDPC-CCs constructed from protographs. Possible variants of the WD scheme will also be discussed. Section 5.3 deals with the performance of LDPC-CC on the binary erasure channel. Starting with a short recapitulation of known results for BP decoding, we will discuss the asymptotic analysis of the WD scheme in detail. Finite-length analysis will include performance evaluation using simulations that reinforce the observations made in the analysis. For erasure channels with memory, we analyze LDPC-CC ensembles both in the asymptotic setting and for finite lengths in Section 5.4. We also include simulations illustrating the good performance of codes derived from the designed protographs over the Gilbert-Elliott channel. Finally, we summarize our findings in Section 5.5.

5.1 LDPC Convolutional Codes

In the following, we will define LDPC-CC, give a construction starting from *protographs*, and discuss various ways of specifying ensembles of these codes.

5.1.1 Definition

A rate $R = b/c$ binary, time-varying LDPC-CC is defined as the set of semi-infinite binary row vectors $\mathbf{v}_{[\infty]}$, satisfying $\mathbf{H}_{[\infty]}\mathbf{v}_{[\infty]}^{\mathrm{T}} = \mathbf{0}_{[\infty]}^{\mathrm{T}}$, where $\mathbf{H}_{[\infty]}$ is the *parity-check*

matrix

$$\mathbf{H}_{[\infty]} = \begin{pmatrix} \mathbf{H}_0(1) & & & & \\ \mathbf{H}_1(1) & \mathbf{H}_0(2) & & & \\ \vdots & \mathbf{H}_1(2) & \ddots & & \\ \mathbf{H}_{m_s}(1) & \vdots & \ddots & \mathbf{H}_0(t) & \\ & \mathbf{H}_{m_s}(2) & \ddots & \mathbf{H}_1(t) & \ddots \\ & & \ddots & \vdots & \ddots \\ & & & \mathbf{H}_{m_s}(t) & \ddots \\ & & & & \ddots \end{pmatrix} \tag{5.1}$$

and $\mathbf{0}_{[\infty]}$ is the semi-infinite all-zero row vector. The elements $\mathbf{H}_i(t)$, $i = 0, 1, \cdots, m_s$ in (5.1) are binary matrices of size $(c-b) \times c$ that satisfy [95]

- $\mathbf{H}_i(t) = \mathbf{0}$, for $i < 0$ and $i > m_s$, $\forall\, t \geq 1$
- $\exists\, t > 0$ such that $\mathbf{H}_{m_s}(t) \neq \mathbf{0}$
- $\mathbf{H}_0(t)$ has full rank $\forall\, t \geq 1$.

The parameter m_s is called the *memory* of the code and $\nu_s = (m_s + 1)c$ is referred to as the *constraint length*. The first two conditions above guarantee that the code has memory m_s and the third condition ensures that the parity-check matrix is full-rank. In order to get sparse graph codes, the Hamming weight of each column $\mathbf{h}$ of $\mathbf{H}_{[\infty]}$ must be very low, i.e., $w_H(\mathbf{h}) \ll \nu_s$. Based on the matrices $\mathbf{H}_i(t)$, LDPC-CC can be classified as follows [33]. An LDPC-CC is said to be *periodic* if $\mathbf{H}_i(t) = \mathbf{H}_i(t+\tau)\ \forall\, i = 0, 1, \cdots, m_s$, $\forall\, t$ and for some $\tau > 1$. When $\tau = 1$, the LDPC-CC is said to be *time-invariant*, in which case the time dependence can be dropped from the notation, i.e., $\mathbf{H}_i(t) = \mathbf{H}_i\ \forall\, i = 0, 1, \cdots, m_s$, $\forall\, t$. If neither of these conditions holds, it is said to be *time-variant*.

Terminated LDPC-CC have a finite parity-check matrix

$$\mathbf{H}_{[L]} = \begin{pmatrix} \mathbf{H}_0(1) & & & \\ \mathbf{H}_1(1) & \mathbf{H}_0(2) & & \\ \vdots & \mathbf{H}_1(2) & \ddots & \\ \mathbf{H}_{m_s}(1) & \vdots & \ddots & \mathbf{H}_0(L) \\ & \mathbf{H}_{m_s}(2) & \ddots & \mathbf{H}_1(L) \\ & & \ddots & \vdots \\ & & & \mathbf{H}_{m_s}(L) \end{pmatrix}$$

where we say that the convolutional code has been terminated after L *instants*. Such a code is said to be (J,K) regular if $\mathbf{H}_{[L]}$ has exactly J 1's in every column and K 1's in every row excluding the first and the last $m_s(c-b)$ rows, i.e., ignoring the terminated portion of the code. It follows that for a given J, the parity-check matrix can be made sparse by increasing c or m_s or both, leading to different code constructions [106]. In this paper, we will consider LDPC-CC characterized by large c and small m_s. As in [67], we will focus on regular LDPC-CC which can be constructed from a protograph.

5.1.2 Protograph-based LDPC-CC

A protograph [112] is a relatively small bipartite graph from which a larger graph can be obtained by a copy-and-permute procedure—the protograph is copied M times, and then the edges of the individual replicas are permuted among the M replicas to obtain a single, large bipartite graph referred to as the *derived graph*. We will refer to M as the *expansion factor*. M is also referred to as the *lifting factor* in literature [98]. Suppose the protograph possesses N_P variable nodes (VNs) and M_P check nodes (CNs), with degrees $J_j, j = 1,\cdots,N_P$, and $K_i, i = 1,\cdots,M_P$, respectively. Then the derived graph will consist of $n = N_P M$ VNs and $m = M_P M$ CNs. The nodes of the protograph are labeled so that if the VN V_j is connected to the CN C_i in the protograph, then V_j in a replica can only connect to one of the M replicated C_i's.

Protographs can be represented by means of an $M_P \times N_P$ bi-adjacency matrix $\mathbf{B}$, called the *base matrix* of the protograph where the entry $\mathbf{B}_{i,j}$ represents the number of edges between CN C_i and VN V_j (a non-negative integer, since parallel edges

are permitted). The degrees of the VNs (CNs respectively) of the protograph are then equal to the sum of the corresponding column (row, respectively) of $\mathbf{B}$. A (J,K) regular protograph-based code is then one with a base matrix where all VNs have degree J and all CNs, excluding those in the terminated portion of the code, have degree K.

In terms of the base matrix, the copy-and-permute operation is equivalent to replacing each entry $\mathbf{B}_{i,j}$ in the base matrix with the sum of $\mathbf{B}_{i,j}$ distinct size-M permutation matrices. This replacement is done ensuring that the degrees are maintained, e.g., a 2 in the matrix $\mathbf{B}$ is replaced by a matrix $\mathbf{H}_2^{(M)} = \mathbf{P}_1^{(M)} \oplus \mathbf{P}_2^{(M)}$ where $\mathbf{P}_1^{(M)}$ and $\mathbf{P}_2^{(M)}$ are two permutation matrices of size M chosen to ensure that each row and column of $\mathbf{H}_2^{(M)}$ has two ones. The resulting matrix after the above transformation for each element of $\mathbf{B}$, which is the bi-adjacency matrix of the derived graph, corresponds to the parity-check matrix $\mathbf{H}$ of the code. The derived graph therefore is nothing but the *Tanner graph* corresponding to the parity-check matrix $\mathbf{H}$ of the code.

For different values of the expansion factor M, different blocklengths of the derived Tanner graph can be achieved, keeping the original graph structure imposed by the protograph. We can hence think of protographs as defining code ensembles that are themselves subsets of random LDPC code ensembles. We will henceforth refer to a protograph $\mathbf{B}$ and the ensemble $\mathscr{C}$ it represents interchangeably. This means that the density evolution analysis for the ensemble of codes represented by the protograph can be performed within the protograph. Furthermore, the structure imposed by a protograph on the derived graph can be exploited to design fast decoders and efficient encoders. Protographs give the code designer a refined control on the derived graph edge connections, facilitating good code design.

Analogous to LDPC block codes, LDPC-CC can also be derived by a protograph expansion. As for block codes, the parity-check matrices of these convolutional codes are composed of blocks of size-M square matrices. We now give two constructions of (J,K) regular LDPC-CC ensembles.

Classical construction

We briefly describe the construction introduced in [107]. For convenience, we will refer to this construction as the *classical* construction of (J,K) regular LDPC-CC

ensembles. Let a be the greatest common divisor (gcd) of J and K. Then there exist positive integers J' and K' such that $J = aJ'$, $K = aK'$, and $\gcd(J', K') = 1$. Assuming we terminate the convolutional code after L instants, we obtain a block code, described by the base matrix

$$\mathbf{B}_{[L]} = \overbrace{\begin{pmatrix} \mathbf{B}_0 & & & \\ \mathbf{B}_1 & \mathbf{B}_0 & & \\ \vdots & \mathbf{B}_1 & \ddots & \\ \mathbf{B}_{m_s} & \vdots & \ddots & \mathbf{B}_0 \\ & \mathbf{B}_{m_s} & \ddots & \mathbf{B}_1 \\ & & \ddots & \vdots \\ & & & \mathbf{B}_{m_s} \end{pmatrix}}^{L}$$

where $m_s = a - 1$ is the *memory* of the LDPC-CC and $\mathbf{B}_i, i = 0, \cdots, m_s$ are $J' \times K'$ submatrices that are all identical and have all entries equal to 1. Note that an LDPC-CC constructed from the protograph with base matrix $\mathbf{B}_{[L]}$ could be time-varying or not depending on the expansion of the protograph into the parity-check matrix.

The protograph of the terminated code has $N_P = LK'$ VNs and $M_P = (L + m_s)J'$ CNs. The rate of the LDPC-CC is therefore

$$R_L = 1 - \left(\frac{L + m_s}{L}\right)\frac{J'}{K'} = 1 - \left(1 + \frac{m_s}{L}\right)(1 - R) \tag{5.2}$$

where $R = 1 - \frac{J'}{K'}$ is the rate of the non-terminated code. Note that $R_L \to R$ and the LDPC-CC has a regular degree distribution [67] when $L \to \infty$. We will assume that the parameters satisfy $K' > J'$ and $L \geq \frac{1-R}{R} m_s$ so that the rates R and R_L of the non-terminated and terminated codes, respectively, are in the proper range.

The classical construction was proposed in [107] and it produces protographs for some (J, K) regular LDPC-CC ensembles. However, not all (J, K) regular LDPC-CC can be constructed, e.g. m_s becomes zero if J and K are relatively prime and consequently the resulting code has no memory. In [67], the authors addressed this problem by proposing a construction rule based on *edge spreading*. We denote an ensemble of (J, K) regular LDPC-CC constructed as described here as $\mathscr{C}_c(J, K)$ with the subscript c for "classical" construction.

Modified construction

We propose a modified construction that is similar to the classical construction except that we do not require that $m_s = a - 1$, i.e., the memory of the LDPC-CC is independent of its degree distribution. We further disregard the requirement that the $\mathbf{B}_i$ matrices are identical and have only ones, i.e., parallel edges in the protograph are allowed. However, the sizes of the submatrices $\mathbf{B}_i, i = 0, 1, \cdots, m_s$ will still be $J' \times K'$. We will denote a (J,K) regular LDPC-CC ensemble constructed in this manner as $\mathscr{C}_m(J,K)$, with subscript m for "modified" construction. Note that the rate of the $\mathscr{C}_m(J,K)$ ensemble is still given by Equation (5.2). Further, the independence of the code memory and the degree distribution allows us to construct LDPC-CC even when J and K are co-primes. This is illustrated in the following example.

Example 5.1. Let $J = 3$ and $K = 4$. Clearly, a classical construction of this ensemble is not possible. However, with the modified construction, we can set $m_s = 1$ and define the ensemble $\mathscr{C}_m(J,K)$ given by

$$\mathbf{B}_0 = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}, \mathbf{B}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

with design rate $R_L = 1 - \frac{3}{4}\left(\frac{L+1}{L}\right)$ for a termination length L. Note that these submatrices are by no means the only possible ones. Another set of submatrices satisfying the constraints is

$$\hat{\mathbf{B}}_0 = \begin{pmatrix} 2 & 0 & 0 & 1 \\ 0 & 2 & 0 & 1 \\ 0 & 0 & 2 & 0 \end{pmatrix}, \hat{\mathbf{B}}_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}. \qquad \square$$

The above example brings out the similarity between the proposed modified construction and the technique of edge spreading employed in [67], wherein the edges of the protograph defined by the matrix

$$\mathbf{B}'_0 = \begin{pmatrix} 2 & 0 & 1 & 1 \\ 0 & 2 & 0 & 2 \\ 1 & 1 & 2 & 0 \end{pmatrix}$$

are "spread" between the matrices $\mathbf{B}_0$ and $\mathbf{B}_1$ (or between $\hat{\mathbf{B}}_0$ and $\hat{\mathbf{B}}_1$) to obtain a $(3,4)$ regular LDPC-CC ensemble with memory $m_s = 1$. The advantage of the modified construction is thus clear—it gives us more degrees of freedom to design the protographs in comparison with the classical construction. In particular, the ensemble specified by the classical construction is contained in the set of ensembles allowed by the modified construction, meaning that the best performing $\mathscr{C}_m(J,K)$ ensemble (with memory the same as that of the $\mathscr{C}_c(J,K)$ ensemble) is at least as good as the $\mathscr{C}_c(J,K)$ ensemble. Note that in [67], there was no indication as to how edges are to be spread between matrices. With windowed decoding, we will shortly show that different protographs (edge spreadings) have different performances. We will also identify certain design criteria for efficient modified constructions that suit windowed decoding.

5.1.3 Polynomial representation of LDPC-CC ensembles

We have thus far specified LDPC-CC ensembles by giving the parameter L and the matrices $\mathbf{B}_i, i = 0,1,\cdots,m_s$. An alternative specification of terminated protograph-based LDPC-CC ensembles using polynomials is useful in establishing certain properties of (J,K) regular ensembles and is described below.

Instead of specifying (m_s+1) matrices $\mathbf{B}_i$ of size $J' \times K'$, we can specify the K' columns of the $(m_s+1)J' \times K'$ matrix

$$\mathbf{B}_{[1]} = \begin{pmatrix} \mathbf{B}_0 \\ \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_{m_s} \end{pmatrix}$$

using a polynomial of degree no more than $\mathtt{d} = (m_s+1)J' - 1$ for each column. The polynomial of the j^{th} column

$$p_j(x) = p_j^{(0)} + p_j^{(1)}x + p_j^{(2)}x^2 + \cdots + p_j^{(\mathtt{d})}x^{\mathtt{d}} \tag{5.3}$$

is defined so that the coefficient of x^i, $p_j^{(i)}$, is the $(i+1,j)$ entry of $\mathbf{B}_{[1]}$ for all $i = 0,1,\cdots,\mathtt{d}$ and $j = 1,2,\cdots,K'$. Therefore, an equivalent way of specifying the LDPC-CC ensemble is by giving L and the set of polynomials $\{p_j(x), j = 1,2,\cdots,K'\}$. With

this notation, the l^{th} column of $\mathbf{B}_{[L]}$ is specified by the polynomial $x^{J'i}p_j(x)$ where $l = iK' + j$ for unique $0 \leq i \leq L-1$ and $1 \leq j \leq K'$. We can hence use "the column index" and "the column polynomial" interchangeably. Further, to define (J,K) regular ensembles, we will need the constraints

$$p_j(1) = J \; \forall \; 1 \leq j \leq K'$$

and

$$\sum_{j=1}^{K'} p_j^{[m]}(1) = K \; \forall \; 0 \leq m \leq J'-1,$$

where $p_j^{[m]}(x)$ is the polynomial of degree no larger than m_s obtained from $p_j(x)$ by collecting the coefficients of terms with degrees l where $l = hJ' + m$ for some $0 \leq h \leq m_s$, i.e., $l = m (\text{mod } J')$:

$$p_j^{[m]}(x) = p_j^{(m)} + p_j^{(J'+m)}x + \cdots + p_j^{(m_sJ'+m)}x^{m_s} = \sum_{h=0}^{m_s} p_j^{(hJ'+m)}x^h. \tag{5.4}$$

We will refer to these polynomials as the *modulo polynomials*. Let us denote the set of polynomials defining an LDPC-CC ensemble as $\mathscr{P} = \{p_j(x), j \in [K']\}$, where $[K'] = \{1,2,\cdots,K'\}$, and the modulo polynomials as $\mathscr{P}_l = \{p_j^{[l]}(x), j \in [K']\}$, $l = 0,1,\cdots,J'-1$. Later in the paper, we will say "the summation of polynomials $p_i(x)$ and $p_j(x)$" to mean the collection of the i^{th} and the j^{th} columns of $\mathbf{B}_{[1]}$. The following example illustrates the notation.

Example 5.2. For $(J,2J)$ codes, we have $J' = 1$ and $K' = 2$, the component base matrices $\mathbf{B}_i, i = 0,...,m_s$ are 1×2 matrices. With the first column of the protograph $\mathbf{B}_{[1]}$, we associate a polynomial $p_1(x) = p_1^{(0)} + p_1^{(1)}x + \cdots + p_1^{(m_s)}x^{m_s}$ of degree at most m_s. Similarly, with the second column we associate a polynomial $p_2(x) = p_2^{(0)} + p_2^{(1)}x + \cdots + p_2^{(m_s)}x^{m_s}$, also of degree at most m_s. Then, the $(2i+1)^{\text{th}}$ column of $\mathbf{B}_{[L]}$ can be associated with the polynomial $x^i p_1(x)$, and the $(2i+2)^{\text{th}}$ column with the polynomial $x^i p_2(x)$. As noted earlier, we will use the polynomial of a column and its index interchangeably, e.g. when we say "choosing the polynomial $x^i p_1(x)$," we mean that we choose the $(2i+1)^{\text{th}}$ column of $\mathbf{B}_{[L]}$. Similarly, by "summations of polynomials $p_1(x)$ and $p_2(x)$," we mean the collection of the corresponding columns of $\mathbf{B}_{[L]}$. In order to define $(J,2J)$ regular ensembles, we will further have the constraint $p_1(1) = p_2(1) = J$. In this case,

since $J' = 1$, $p_1^{[0]}(1) + p_2^{[0]}(1) = 2J$ is the same as the previous constraint, because $p_1^{[0]}(1) + p_2^{[0]}(1) = p_1(1) + p_2(1)$. □

We define the *minimum degree* of a polynomial $a(x)$ as the least exponent of x with a positive coefficient and denote it as $\min\deg(a(x))$. Clearly, $0 \leq \min\deg(a(x)) \leq \deg(a(x))$. Let us define a partial ordering of polynomials with non-negative integer coefficients as follows. We write $a(x) \preceq b(x)$ if $\min\deg(a(x)) = \min\deg(b(x))$, $\deg(a(x)) = \deg(b(x))$ and the coefficients of $a(x)$ are no larger than the corresponding ones of $b(x)$. The ordering $\preceq$ satisfies the following properties over polynomials with non-negative integer coefficients: if $a(x) \preceq b(x)$ and $c(x) \preceq d(x)$, then

$$a(x) + c(x) \preceq b(x) + d(x)$$
$$a(x)c(x) \preceq b(x)d(x).$$

We define the *boundary polynomial* $\beta(a(x))$ of a polynomial $a(x)$ to be $\beta(a(x)) = x^i + x^j$ where $i = \min\deg(a(x))$ and $j = \deg(a(x))$. Note that when $i = j$, we define $\beta(a(x)) = x^i$. We have for any polynomial $a(x)$, $\beta(a(x)) \preceq a(x)$.

5.2 Windowed Decoding

LDPC-CC are characterized by a very large constraint length $\nu_s = (m_s + 1)K'M$. Since the Viterbi decoder has a complexity that scales exponentially in the constraint length, it is impractical for this kind of code. For terminated LDPC-CC, decoding can be performed as in the case of an LDPC block code, meaning that each frame carrying a codeword obtained through the termination can be decoded with belief propagation (BP), i.e., the sum-product algorithm (SPA) [61, 121]. Note that since the BP decoder can start decoding only after the entire codeword is received, the total decoding latency Λ_{BP} is given by $\Lambda_{BP} = T_{cw} + T_{dec}$, where T_{cw} is the time taken to receive the entire codeword and T_{dec} is the time needed to decode the codeword. In many practical applications this latency is large and undesirable. Moreover, for non-terminated LDPC-CC, a BP decoder cannot be employed.

However, LDPC-CCs are a variant of the spatially coupled codes introduced in Chapter 4. In particular, we note that L denotes the chain length, and the memory

m_s corresponds to the coupling length γ in the parlance of spatially coupled codes (cf. Section 4.1). We can therefore exploit the convolutional structure of the parity-check matrix representing these codes and define a windowed decoder as was done for spatially coupled codes.

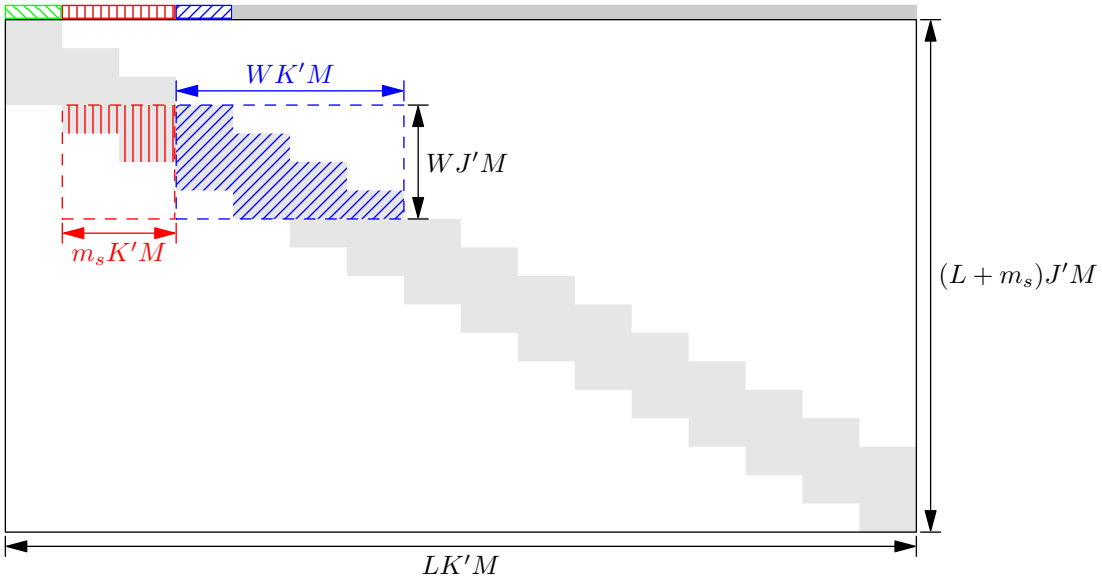

Figure 5.1: Illustration of windowed decoding (WD) with window of size $W = 4$ for a $\mathscr{C}_m(J,2J)$ LDPC-CC with $m_s = 2$ and $L = 16$ at the fourth decoding instant. This window configuration consists of $J_W = WJ'M = 4M$ rows of the parity-check matrix and all the $(W+m_s)K'M = 12M$ columns involved in these equations: this comprises the red (vertically hatched) and the blue (hatched) edges shown within the matrix. Note that the symbols shown in green (back-hatched) above the parity-check matrix have all been processed. The targeted symbols are shown in blue (hatched) above the parity-check matrix and the symbols that are yet to be decoded are shown in gray above the parity-check matrix.

The convolutional structure of the code imposes a constraint on the VNs connected to the same parity-check equations—two VNs of the protograph that are at least $(m_s+1)K'$ columns apart cannot be involved in the same parity-check equation. This characteristic can be exploited in order to perform continuous decoding of the received stream through a "window" that slides along the bit sequence. Moreover, this structure allows for the possibility of parallelizing the iterations of the message passing decoder through several processors working in different regions of the Tanner graph. A pipeline decoder based on this idea was proposed in [33]. In this paper we consider a windowed

decoder to decode terminated codes with reduced latency. Note that whereas a similar sliding window decoder was used to bound the performance of BP decoding in [68], we are interested in evaluating the performance of the windowed decoder from a perspective of reducing the decoding complexity and latency.

Consider a terminated (J,K) regular parity-check matrix $\mathbf{H}$ built from a base matrix $\mathbf{B}$. The windowed decoder works on sub-protographs of the code and the window size W is defined as the number of sets of J' CNs of the protograph $\mathbf{B}$ considered within each window. In the parity-check matrix $\mathbf{H}$, the window thus consists of $J_W = WJ'M = W(c-b)$ rows of $\mathbf{H}$ and all columns that are involved in the check equations corresponding to these rows. We will henceforth refer to the size of the window only in terms of the protograph with the corresponding size in the parity-check matrix implied. The window size W ranges between (m_s+1) and $(L-1)$ because each VN in the protograph is involved in at most $J'(m_s+1)$ check equations; and, although there are a total of $M_P = J'(L+m_s)$ CNs in $\mathbf{B}$, the decoder can perform BP when all the VN symbols are received, i.e., when $L \leq W \leq L+m_s$. As was seen in Chapter 4, apart from the window size, the decoder also has a (typically small) target erasure probability $\delta \geq 0$ as a parameter. The aim of the WD is to reduce the erasure probability of every symbol in the codeword to a value no larger than δ.

At the first decoding instant, or equivalently, in the first window configuration, the decoder performs belief-propagation over the edges within the window with the aim of decoding all of the first K' symbols in the window, called the *targeted symbols* (cf. Chapter 4). The window slides down J' rows and right K' columns in $\mathbf{B}$ after at least a fraction $(1-\delta)$ of the targeted symbols are recovered (or, in general, after a maximum number of belief-propagation iterations have been performed), and continues decoding at the new window configuration at the next decoding time instant.

In the terminated portion of the code, the window configuration will have fewer edges than other configurations within the code. Since the WD aims to recover only the targeted symbols within each window configuration, the entire codeword is recovered in L decoding time instants. Figure 5.1 shows a schematic representation of the WD for $W = 4$.

The decoding latency of the K' targeted symbols with WD is therefore given by

$\Lambda_{WD} = T_W + T_{dec}(W)$, where T_W is the time taken to receive all the symbols required to decode the K' targeted symbols, and $T_{dec}(W)$ is the time taken to decode the targeted symbols. The parameters T_{cw} and T_W are related as

$$T_W = \frac{(W+m_s)K'}{LK'} T_{cw} = \frac{W+m_s}{L} T_{cw},$$

since at most $(W+m_s)K'$ symbols are to be received to process the targeted symbols. The relation between T_{dec} and $T_{dec}(W)$ is given by

$$T_{dec}(W) = \frac{W}{L} T_{dec},$$

since the complexity of BP decoding scales linearly in blocklength and the WD uses BP decoding over WK' symbols in each window configuration. We assume that the number of iterations of message passing performed is fixed to be the same for the BP decoder and the WD. Thus, in latency-limited scenarios, we can use the WD to obtain a latency reduction of

$$\Lambda_{WD} \leq \frac{W+m_s}{L} \Lambda_{BP} \triangleq w\Lambda_{BP}.$$

The smallest latency supported by the code-decoder system is therefore at most a fraction $w_{min} = \frac{2m_s+1}{L}$ that of the BP decoder. As pointed out earlier, the only choice for non-terminated codes is to use some sort of a windowed decoder. For the sequence of ensembles indexed by L, with the choice of the proposed WD with a fixed finite window size W, the decoding latency vanishes as $O(\frac{1}{L})$. We will typically be interested in small values of W where large gains in decoding latencies are achievable. Since the decoding latency increases as W increases, the trade-off between decoding performance and latency can be studied by analyzing the performance of the WD for the entire range of window sizes.

Remark 5.1 (Latency Flexibility). Although reduced latency is an important characteristic of WD, what is perhaps more useful practically is the flexibility to alter the latency with suitable changes in the code performance. The latency can be controlled by varying the parameter W as required. If a large latency can be handled, W can be kept large ensuring good code performance and if a small latency is required, W can be made small while paying a price with the code performance (We will see shortly that the performance of WD is monotonic in the window size).

One possible variant of WD is a decoding scheme which starts with the smallest possible window size and the size is increased whenever targeted symbols cannot be decoded, i.e., the target erasure probability cannot be met within the fixed maximum number of iterations. Other schemes where the window size is either increased or decreased based on the performance of the last few window configurations are also possible.

5.3 Memoryless Erasure Channels

In this section, we confine our attention to the performance of the LDPC-CC when the transmission occurs over a memoryless erasure channel, i.e., a binary erasure channel (BEC) parameterized by the channel erasure rate ε.

5.3.1 Asymptotic analysis

We consider the performance of the LDPC-CC in terms of the average performance of the codes belonging to ensembles defined by protographs in the limit of infinite blocklengths and in the limit of infinite iterations of the decoder. This represents the setting where the results of Chapter 4 are directly applicable, since protograph-based LDPC-CCs are a variant of spatially coupled codes.

For LDPC-CC based on protographs, the BP decoding thresholds can be numerically estimated using the Protograph-EXIT (P-EXIT) analysis [75].The processing at a CN of degree d_C results in an updating of the mutual information on the d_C^{th} edge as

$$I_{\text{out},d_C} = \mathtt{C}\left(I_{\text{in},1},\cdots,I_{\text{in},d_C-1}\right) = \prod_{i=1}^{d_C-1} I_{\text{in},i} \tag{5.5}$$

and the corresponding update at a VN of degree d_V gives

$$I_{\text{out},d_V} = \mathtt{V}\left(I_{ch},I_{\text{in},1},\cdots,I_{\text{in},d_V-1}\right) = 1-\varepsilon\prod_{i=1}^{d_V-1}\left(1-I_{\text{in},i}\right) \tag{5.6}$$

where $I_{ch} = 1-\varepsilon$ is the mutual information obtained from the channel. Note that the edge multiplicities are included in the above check and variable node computations. The

a-posteriori mutual information $\mathscr{I}$ at a VN is found using

$$\mathscr{I} = 1 - \varepsilon \prod_{i=1}^{d_V} (1 - I_{\text{in},i}) = 1 - (1 - I_{\text{out},d_V})(1 - I_{\text{in},d_V})$$

where the second equality follows from (5.6).

Example 5.3. The protograph $\mathbf{B}_{3,6} = (3 \quad 3)$ has a BP threshold of $\varepsilon_{BP}^* \approx 0.4294$. Note that all the CNs in the protograph are of degree 6 while all the VNs are of degree 3. This BP threshold is expected because $\mathbf{B}_{3,6}$ corresponds to the $(3,6)$ regular LDPC block code ensemble. The following protograph $\mathbf{B}'_{3,6}$ has a BP threshold $\varepsilon_{BP}^* \approx 0.4879$ for $L = 40$. Note that, as before, all VNs are of degree 3 and all the CNs except the ones in the terminated portion of the code are of degree 6.

$$\mathbf{B}'_{3,6} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & \cdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 & 1 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 \end{pmatrix}$$

This is the $\mathscr{C}_c(3,6)$ ensemble constructed in [107]. In terms of the notation introduced, this is given as $\mathbf{B}_0 = \mathbf{B}_1 = \mathbf{B}_2 = [1 \quad 1]$; or equivalently as $p_1(x) = p_2(x) = 1 + x + x^2$. □

The above example illustrates the strength of protographs—they allow us to choose structures within an ensemble defined by a pair of degree distributions that may perform better than the ensemble average. As noted in Chapter 4,the BP performance of regular LDPC-CC ensembles has been related to the MAP decoder performance of the corresponding unstructured ensemble [63].

Remark 5.2. In the limit of infinite blocklength, each term in the base protograph $\mathbf{B}$ is replaced by a permutation matrix of infinite size to obtain the parity-check matrix, and therefore the latency of any window size is infinite, apparently defeating the purpose of WD. Our interest in the asymptotic performance, however, is justified as it allows us to establish lower bounds on the probability of failure of the windowed decoder to

recover the symbols of the finite length code. In practice, it is to be expected that the gap between the performance of a finite length code with WD and the asymptotic ensemble performance of the ensemble to which the code belongs increases as the window size reduces due to the reduction in the blocklength of the subcode defined by the window.

We define the threshold $\varepsilon_{(i)}^{\mathrm{WD}}(\mathbf{B}, W, \delta)$ of the i^{th} window configuration to be the supremum of the channel erasure rates for which the WD succeeds in retrieving the targeted symbols of the i^{th} window with a probability at least $(1-\delta)$, given that each of the targeted symbols corresponding to the first $(i-1)$ window configurations is known with probability $1-\delta$. Figure 5.2 illustrates the threshold $\varepsilon_{(i)}^{\mathrm{WD}}(\mathbf{B}, W, \delta)$ of the i^{th} window configuration. The *windowed threshold* $\varepsilon^{\mathrm{WD}}(\mathbf{B}, W, \delta)$ is then defined as the supremum

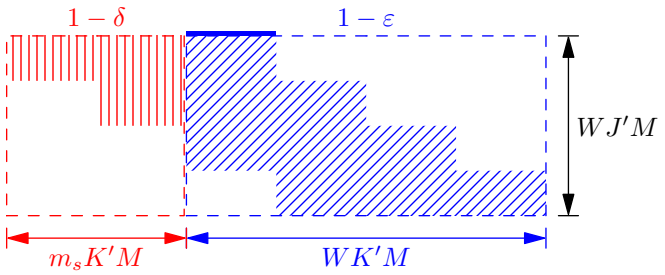

Figure 5.2: Illustration of the threshold of the i^{th} window configuration $\varepsilon_{(i)}^{\mathrm{WD}}(\mathbf{B}, W, \delta)$. The targeted symbols of the previous window configurations are known with probability $1-\delta$. The targeted symbols within the window are highlighted with a solid blue bar on top of the window. The symbols within the blue (hatched) region in the window are initially known with probability $1-\varepsilon$. The task of the decoder is to perform BP within this window until the erasure probability of the targeted symbols is smaller than δ. The window is then slid to the next configuration.

of channel erasure rates for which the windowed decoder can decode each symbol in the codeword with probability at least $1-\delta$.

We assume that between decoding time instants, no information apart from the targeted symbols is carried forward, i.e., when a particular window configuration has been decoded, all the present processing information apart from the decoded targeted symbols themselves is discarded. With this assumption, it is clear that the windowed threshold of a protograph-based LDPC-CC ensemble is given by the minimum of the thresholds of its window configurations. For the classical and modified constructions of LDPC-CC described in Section 5.1.2, all window configurations are similar except

the ones at the terminated portion of the code. Since the window configurations at the terminated portions can only perform better, the windowed threshold is determined by the threshold of a window configuration not in the terminated portion of the code. Note that the performance of WD when the information from processing the previous window configurations is made use of in successive window configurations, e.g., when symbols other than the targeted symbols that were decoded previously are also retained, can only be better than what we obtain here.

We now prove a monotonicity property of the WD for LDPC-CCs. Note that this property is evidently true for spatially coupled codes in general from Theorem 4.2.

Proposition 5.1 (Monotonicity of WD performance in W)**.** *For any $\mathscr{C}_m(J,K)$ ensemble* $\mathbf{B}$*,*

$$\varepsilon^{\mathrm{WD}}(\mathbf{B},W,\delta) \leq \varepsilon^{\mathrm{WD}}(\mathbf{B},W+1,\delta). \qquad \square$$

We now state two lemmas which will be made use of in the proof of the above Proposition. The proofs of these lemmas are straightforward and have been omitted.

Lemma 5.2 (Monotonicity of $\mathtt{C}$)**.** *The CN operation in* (5.5) *is monotonic in its arguments, i.e.,*

$$0 \leq x \leq x' \leq 1 \Rightarrow \mathtt{C}(x,y) \leq \mathtt{C}(x',y)\ \forall\ y \in [0,1],$$

where the two-argument function $\mathtt{C}(x,y) = xy$. $\square$

Lemma 5.3 (Monotonicity of $\mathtt{V}$)**.** *The VN operation in* (5.6) *is monotonic in its arguments, i.e.,*

$$0 \leq x \leq x' \leq 1 \Rightarrow \mathtt{V}(x,y) \leq \mathtt{V}(x',y)\ \forall\ y \in [0,1],$$

where $\mathtt{V}(x,y) = 1-(1-x)(1-y)$. $\square$

The operational significance of the above lemmas is the following: if we can upper (lower) bound the mutual information on some incoming edge of a CN or a VN, and use the bound to compute the outgoing mutual information from that node, we get an upper (lower) bound on the actual outgoing mutual information. Thus, by bounding the mutual information on some edges of a computation tree and repetitively applying Lemmas 5.2 and 5.3, one can obtain bounds for the a-posteriori mutual information at the root of the tree.

Proof of Proposition 5.1. Consider the i^{th} window configuration for window sizes W and $W+1$ shown in Figure 5.3. We are interested in a window configuration that is not at the terminated portion of the code. Call the Tanner graphs of these windows

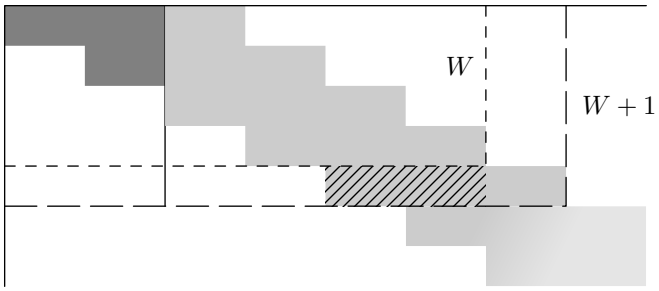

Figure 5.3: Sub-protographs of window sizes W and $W+1$. The edges connected to targeted symbols from previous window configurations are shown in darker shade of gray.

$A = (V_A, C_A, E_A)$ and $B = (V_B, C_B, E_B)$ respectively, where V_A, V_B and C_A, C_B are the sets of VNs and CNs respectively and E_A, E_B are the sets of edges. Clearly, $V_A \subset V_B, C_A \subset C_B$, and $E_A \subset E_B$. Any VN in V_A that is connected to some variable in $V_B \setminus V_A$ has to be connected via some CN in $C_B \setminus C_A$. The edges between these CNs and VNs in V_A are shown hatched in Figure 5.3. Consider the computation trees for the a-posteriori message at a targeted symbol in V_A and that for the same symbol in V_B. Call them $\mathscr{T}_A$ and $\mathscr{T}_B$ respectively. Then we have $\mathscr{T}_A \subset \mathscr{T}_B$.

We start by augmenting $\mathscr{T}_A$, creating another tree $\mathscr{T}_A^+$ that has the same structure as $\mathscr{T}_B$. In particular, $\mathscr{T}_A^+$ includes the additional edges corresponding to the hatched region. In $\mathscr{T}_A^+$ and $\mathscr{T}_B$, we denote the set of these edges by $E_u(\mathscr{T}_A^+)$ and $E_u(\mathscr{T}_B)$ respectively, and we assign 0 mutual information to each edge in $E_u(\mathscr{T}_A^+)$.

Now, let $I^{\mathscr{T}_A}$, $I^{\mathscr{T}_A^+}$ and $I^{\mathscr{T}_B}$ be the a-posteriori mutual information at the roots of the trees $\mathscr{T}_A$, $\mathscr{T}_A^+$ and $\mathscr{T}_B$ respectively. Then it is clear that $I^{\mathscr{T}_A} = I^{\mathscr{T}_A^+}$, since the messages on edges in $E_u(\mathscr{T}_A^+)$ are effectively erasures and zero out the contributions from the checks in $C_A^+ \setminus C_A = C_B \setminus C_A$. On the other hand, if we denote by $I_e(\mathscr{T}_B)$ the mutual information associated with an edge $e \in E_u(\mathscr{T}_B)$, and by $I_e(\mathscr{T}_A^+)$ the mutual information associated with the corresponding edge in $\mathscr{T}_A^+$, we know that $I_e(\mathscr{T}_A^+) = 0$ so that $I_e(\mathscr{T}_A^+) \leq I_e(\mathscr{T}_B)$. Hence, we have from Lemmas 5.2 and 5.3 that $I^{\mathscr{T}_A^+} \leq I^{\mathscr{T}_B}$. Since $I^{\mathscr{T}_A} = I^{\mathscr{T}_A^+}$, it follows that $I^{\mathscr{T}_A} \leq I^{\mathscr{T}_B}$, as desired. □

It follows immediately from the definition of the windowed threshold that

$$\varepsilon^{\mathrm{WD}}(\mathbf{B},W,\delta) \leq \varepsilon^{\mathrm{WD}}(\mathbf{B},W,\delta') \; \forall \; \delta \leq \delta'.$$

Furthermore, from the continuity of the density evolution equations (5.6) and (5.5), we have that when we set $\delta = 0$, we decode not only the targeted symbols within the window but all the remaining symbols also. Since the symbols in the right end of the window are the "worst protected" ones within the window (in the sense that these are the symbols for which the least number of constraints are used to decode), we expect the windowed thresholds $\varepsilon^{\mathrm{WD}}(\mathbf{B},W,\delta = 0)$ to be dictated by the behavior of the submatrix $\mathbf{B}_0$ under BP. In the following, when the base matrix $\mathbf{B}$ of the protograph corresponding to an ensemble $\mathscr{C}$ is unambiguous, we will write $\varepsilon^{\mathrm{WD}}(\mathbf{B},W,\delta)$ and $\varepsilon^{\mathrm{WD}}(\mathscr{C},W,\delta)$ interchangeably.

We next turn to giving some properties of LDPC-CC ensembles with good performance under WD. We start with an example that illustrates the stark change in performance a small difference in the structure of the protograph can produce.

Example 5.4. Consider WD with the ensemble $\mathscr{C}_c(3,6)$ in Example 5.3 with a window of size $W = 3$. The corresponding protograph defining the first window configuration is

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

and we have $\varepsilon^{\mathrm{WD}}(\mathscr{C}_c(3,6),W = 3,\delta = 0) = 0$. This is seen readily by observing that there are VNs of degree 1 that are connected to the same CNs. In fact, from this reasoning, we see that $\varepsilon^{\mathrm{WD}}(\mathscr{C}_c(J,K'J),W,\delta = 0) = 0 \; \forall \; J \leq W \leq L$.

As an alternative, we consider the modified construction of Section 5.1.2 to obtain the $\mathscr{C}_m(J,K)$ ensemble $\mathbf{B}'$ given by $\mathbf{B}_0 = [2 \;\; 2], \mathbf{B}_1 = [1 \;\; 1]$. This ensemble has a BP threshold $\varepsilon^{\mathrm{BP}}(\mathbf{B}') \approx 0.4875$ for $L = 40$ which is quite close to that of the ensemble $\mathscr{C}_c(3,6)$, $\varepsilon^{\mathrm{BP}}(\mathscr{C}_c(3,6)) \approx 0.4879$. WD with a window of size 3 for this ensemble has the first window configuration

$$\begin{pmatrix} 2 & 2 & 0 & 0 & 0 & 0 \\ 1 & 1 & 2 & 2 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 2 \end{pmatrix}$$

which has a threshold $\varepsilon^{\mathrm{WD}}(\mathbf{B}', W = 3, \delta = 0) \approx 0.3331$, i.e., we can theoretically get close to 68.3% of the BP threshold with $< 10\%$ of the latency of the BP decoder. Note that this improvement in threshold has been obtained while also increasing the rate of the ensemble, since $m_s = 1$ for the $\mathbf{B}'$ ensemble in comparison with $m_s = 2$ for $\mathscr{C}_c(3,6)$. □

The above example illustrates the tremendous advantage obtained by using $\mathscr{C}_m$ (J,K) ensembles for WD even under the severe requirement of $\delta = 0$. The following is a good rule of thumb for constructing LDPC-CC ensembles that have good performance with WD.

Design Rule 5.1. For $\mathscr{C}_m(J,K'J)$ ensembles, set $p_j^{(d_j)} \geq 2$ for all $j \in [K']$ where $d_j = \min \deg(p_j(x))$. □

The above design rule says that for $(J,K'J)$ ensembles, it is better to avoid degree-1 VNs within a window. Note that none of the $\mathscr{C}_c(J,K'J)$ ensembles satisfy this design rule. We now illustrate the performance of LDPC-CC ensembles with WD when we allow $\delta > 0$.

Example 5.5. We compare three LDPC-CC ensembles. The first is the classical LDPC-CC ensemble $\mathscr{C}_1 = \mathscr{C}_c(3,6)$. The second and the third are LDPC-CC ensembles constructed as described in Section 5.1.2. The ensemble $\mathscr{C}_2$ is defined by the polynomials $p_1(x) = 2 + x^2, p_2(x) = 2 + x$ and $\mathscr{C}_3$ is defined by $q_1(x) = q_2(x) = 2 + x$.

We first observe that all three ensembles have the same asymptotic degree distribution, i.e., all are $(3,6)$ regular LDPC-CC ensembles when $L \to \infty$. While $\mathscr{C}_1$ and $\mathscr{C}_2$ have a memory $m_s = 2$, $\mathscr{C}_3$ has a memory $m_s = 1$. Therefore, for a fixed L, while $\mathscr{C}_1$ and $\mathscr{C}_2$ have the same rate, $\mathscr{C}_3$ has a higher rate. Another consequence of a smaller m_s is that $\mathscr{C}_3$ can be decoded with a window of size $W_{min}(\mathscr{C}_3) = 2$. Further note that whereas $\mathscr{C}_2$ and $\mathscr{C}_3$ satisfy Design Rule 5.1, $\mathscr{C}_1$ does not. For a window of size 3, the subprotographs for ensembles $\mathscr{C}_1$ and $\mathscr{C}_3$ are as shown in Example 5.4, and that for ensemble $\mathscr{C}_2$ is as shown below

$$\begin{pmatrix} 2 & 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 2 & 0 & 0 \\ 1 & 0 & 0 & 1 & 2 & 2 \end{pmatrix}$$

In Figure 5.4, we show the windowed thresholds plotted against the window size for the three ensembles $\mathscr{C}_1, \mathscr{C}_2$ and $\mathscr{C}_3$ by fixing $L = 100$ for $\delta \in \{10^{-6}, 10^{-12}\}$.

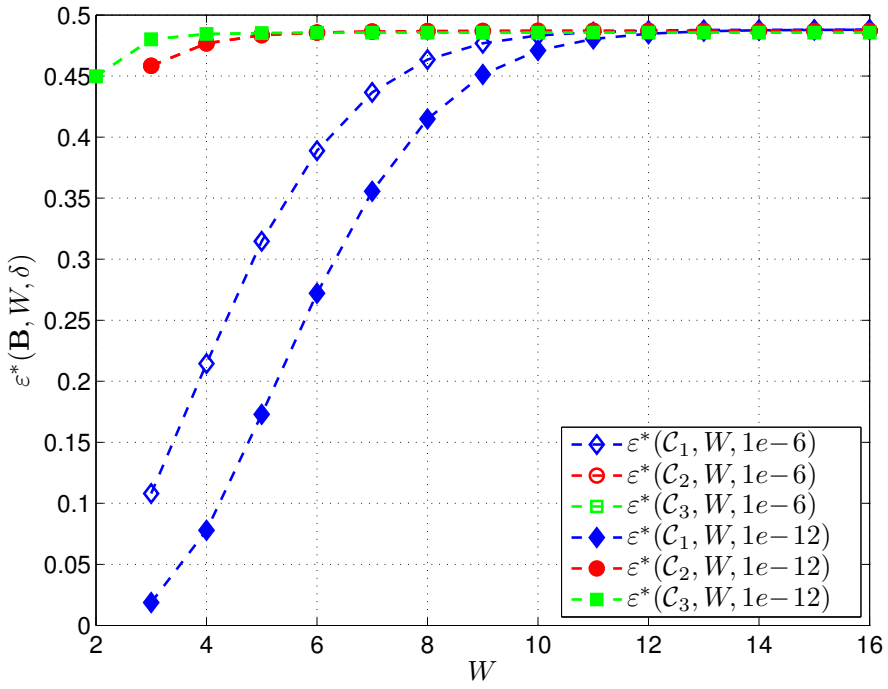

Figure 5.4: Windowed threshold $\varepsilon^* \equiv \varepsilon^{\mathrm{WD}}$ as a function of the window size for the ensembles $\mathscr{C}_i, i = 1,2,3$ with $\delta \in \{10^{-6}, 10^{-12}\}$. The rates of the ensembles $\mathscr{C}_1$ and $\mathscr{C}_2$ are 0.49 whereas that of $\mathscr{C}_3$ is 0.495. The corresponding Shannon limits are therefore 0.51 for $\mathscr{C}_1$ and $\mathscr{C}_2$, and 0.505 for $\mathscr{C}_3$.

A few observations are in order. The monotonicity of $\varepsilon^{\mathrm{WD}}(\mathbf{B}, W, \delta)$ in W as proven in Proposition 5.1 is evident. The windowed thresholds $\varepsilon^{\mathrm{WD}}(\mathbf{B}, W, \delta)$ for $\mathscr{C}_2$ and $\mathscr{C}_3$ are fairly close to the maximum windowed threshold even when $W = W_{min}$. The windowed thresholds for ensembles $\mathscr{C}_2$ and $\mathscr{C}_3$ are robust to changes in δ, i.e., the thresholds are almost the same (the points overlap in the figure) for $\delta = 10^{-6}$ and $\delta = 10^{-12}$. Further, the windowed thresholds $\varepsilon^{\mathrm{WD}}(\mathscr{C}_i, W, \delta)$ are fairly close to the BP thresholds $\varepsilon^{\mathrm{BP}}(\mathscr{C}_i), i = 1,2,3$ for $W \geq 12$. We will see next that this last observation is not always true. □

Remark 5.3 (Effect of termination). The better BP performance of the $\mathscr{C}_c(J,K)$ ensemble in comparison with that of the (J,K)-regular block code ensemble (cf. Example 5.3) is because of the termination of the parity-check matrix of $\mathscr{C}_c(J,K)$ codes. More pre-

cisely, the low-degree CNs at the terminated portion of the protograph are more robust to erasures and their erasure-correcting power is cascaded through the rest of the protograph to give a better threshold for the convolutional ensemble in comparison with that for the corresponding unstructured ensemble [106]. From the definition of the WD, we can see that the sub-protograph within a window does not have the lower-degree checks if previous targeted symbols are not decoded. Therefore, we would expect a deterioration in the performance. Furthermore, the Design Rule 5.1 increases the degrees of the CNs in the terminated portion. Therefore, the effect of different termination on the WD performance is of interest.

Example 5.6. Tables 5.1 and 5.2 illustrate the WD thresholds for $\mathscr{C}_m(J,2J)$ ensembles that satisfy Design Rule 5.1 except when $J = m_s + 1$. These ensembles are defined by the polynomials

$$p_1(x) = p_2(x) = (J - m_s) + x + x^2 + \cdots + x^{m_s}.$$

Note that $J \geq m_s + 1$. The ensembles are terminated so that the rate is $R_L = 0.49$. The worst threshold with WD (corresponding to the least window size $W_{min} = m_s + 1$) is denoted $\varepsilon^{\mathrm{WD}}_{m_s+1}$. The largest threshold with WD is denoted $\varepsilon^{\mathrm{WD}}_{L-m_s}$ and the BP threshold as $\varepsilon^{\mathrm{BP}}$. The increase in the gap between $\varepsilon^{\mathrm{WD}}_{L-m_s}$ and $\varepsilon^{\mathrm{BP}}$ with increasing J illustrates the loss

Table 5.1: WD Thresholds for $m_s = 1$, $R_L = 0.49$-$\mathscr{C}_m(J,2J)$ ensembles with $\delta = 10^{-12}$ and window size $m_s + 1, L - m_s$. BP Thresholds are provided in the last column.

J	$\varepsilon^{\mathrm{WD}}_{m_s+1}$	$\varepsilon^{\mathrm{WD}}_{L-m_s}$	$\varepsilon^{\mathrm{BP}}$
2	0.0008	0.3162	0.3342
3	0.4499	0.4857	0.4872
4	0.4449	0.4469	0.4961
5	0.3915	0.3923	0.4969
6	0.3469	0.3475	0.4959
7	0.3115	0.3118	0.4891
8	0.2829	0.2832	0.4785
9	0.2595	0.2597	0.4666

due to edge multiplicities ("weaker" termination). This is because the terminations at the beginning and at the end of the code are different, i.e., the CN degrees in the terminated portion at the beginning of the code are $2(J - m_s)$ which increases with J; whereas those at the end of the code are 2, a constant. Thus, much of the code performance is

Table 5.2: WD Thresholds for $m_s = 2$, $R_L = 0.49$-$\mathscr{C}_m(J,2J)$ ensembles with $\delta = 10^{-12}$ and window size $m_s+1, L-m_s$. BP Thresholds are provided in the last column.

J	$\varepsilon^{\mathrm{WD}}_{m_s+1}$	$\varepsilon^{\mathrm{WD}}_{L-m_s}$	$\varepsilon^{\mathrm{BP}}$
3	0.0189	0.4882	0.4876
4	0.4875	0.4947	0.4958
5	0.4493	0.4501	0.4971
6	0.3941	0.3945	0.4972
7	0.3489	0.3492	0.4969
8	0.3131	0.3133	0.4967
9	0.2843	0.2845	0.4957
10	0.2607	0.2608	0.4937

determined by the "stronger" (smaller check-degree) termination, the one at the end of the code for $J > m_s + 1$. This is also seen by the fact that the gap between $\varepsilon^{\mathrm{WD}}_{m_s+1}$ and $\varepsilon^{\mathrm{WD}}_{L-m_s}$ decreases as J increases, meaning that the termination at the beginning of the code is weak and increasing the window size helps little. Note that the $\mathscr{C}_m(3,6)$ ensemble in Table 5.2 is in fact the $\mathscr{C}_c(3,6)$ classical ensemble, and that $\varepsilon^{\mathrm{WD}}_{L-m_s}$ is larger than the corresponding BP threshold. This is possible since WD only demands that the erasure probability of the targeted symbols is reduced to δ. In contrast, BP demands that the erasure probability of all the symbols is reduced to 0. □

From the above discussion, we can add the following as another design rule.

Design Rule 5.2. For $\mathscr{C}_m(J,K)$ ensembles, keep the termination at the beginning of the code strong, preferably stronger than the one at the end of the code. That is, use polynomials $\mathscr{P} = \{p_j(x), j \in [K']\}$ such that each of the sums

$$\sum_{j=1}^{K'} p_j^{(0)}, \cdots, \sum_{j=1}^{K'} p_j^{(J'-1)}$$

is kept as small as possible. □

Remark 5.4 (Targeted symbols). We have thus far in this chapter, as well as Chapter 4, considered only the first K' VNs in the sub-protograph contained within the window to be the targeted symbols. However, as an alternative way to trade-off performance for reduced latency, it is possible to consider other VNs also as targeted symbols. In this case, the window would be shifted beyond all the targeted symbols after processing each window configuration. For a window of size W, let us denote by $\varepsilon_i^{\mathrm{WD}}(\mathbf{B}, W, \delta)$ the

windowed threshold when the targeted symbols are the first iK' VNs, $1 \leq i \leq W$. Note that this is different from the i^{th} window configuration threshold $\varepsilon^{\text{WD}}_{(i)}(\mathbf{B}, W, \delta)$. Hence, $\varepsilon^{\text{WD}}(\mathbf{B}, W, \delta) = \varepsilon^{\text{WD}}_{1}(\mathbf{B}, W, \delta)$. By definition, $\varepsilon^{\text{WD}}_{i}(\mathbf{B}, W, \delta) \leq \varepsilon^{\text{WD}}(\mathbf{B}, W, \delta)$.

Example 5.7. Consider the $\mathscr{C}_m(6,12)$ ensemble with $m_s = 1$ defined by $p_1(x) = p_2(x) = 3+3x$, denoted $\mathscr{C}_4$; and the $\mathscr{C}_m(4,8)$ ensemble with $m_s = 1$ defined by $q_1(x) = q_2(x) = 2+2x$, denoted $\mathscr{C}_5$. Also consider ensembles $\mathscr{C}_6$ and $\mathscr{C}_7$ given by $r_1(x) = r_2(x) = 2+4x$ and $s_1(x) = s_2(x) = 2+2x+2x^2$ respectively. Both $\mathscr{C}_6$ and $\mathscr{C}_7$ are $\mathscr{C}_m(6,12)$ ensembles, but with memory $m_s = 1$ and 2 respectively. Table 5.3 gives the windowed thresholds $\varepsilon^{\text{WD}}_{i}(\mathscr{C}_j, W = 4, \delta)$ with iK' targeted symbols for a window of size 4 for $j = 4,5,6,7$.

Table 5.3: Windowed thresholds $\varepsilon^{\text{WD}}_{i}$ for ensembles $\mathscr{C}_j$ with iK' targeted symbols, window size $W = 4$, and $\delta = 10^{-12}$, for $j = 4,5,6,7$.

i	$\mathscr{C}_4$	$\mathscr{C}_5$	$\mathscr{C}_6$	$\mathscr{C}_7$
1	0.4429	0.4912	0.4835	0.4924
2	0.4429	0.4905	0.4835	0.4919
3	0.4427	0.4824	0.4828	0.4824
4	0.4294	0.3331	0.3331	0.3331

One might expect the windowed threshold $\varepsilon^{\text{WD}}(\mathbf{B}, W, \delta)$ to be higher for an ensemble for which $\varepsilon^{\text{WD}}_{W}(\mathbf{B}, W, \delta)$ is higher. This is not quite right: we see that for ensembles $\mathscr{C}_4$ and $\mathscr{C}_5$, $\varepsilon^{\text{WD}}_{4}(\mathscr{C}_4, 4, 10^{-12}) \approx 0.4294 > 0.3331 \approx \varepsilon^{\text{WD}}_{4}(\mathscr{C}_5, 4, 10^{-12})$ whereas $\varepsilon^{\text{WD}}_{i}(\mathscr{C}_5, 4, 10^{-12}) > \varepsilon^{\text{WD}}_{i}(\mathscr{C}_4, 4, 10^{-12}) \; \forall \; i < 4$. This can again be explained as the effect of stronger termination in $\mathscr{C}_5$ in comparison with $\mathscr{C}_4$. This is also evident in the larger thresholds for the $(6,12)$ ensemble $\mathscr{C}_6$ with same memory as $\mathscr{C}_4$, but stronger termination. Also, keeping the same termination and increasing the memory improves the performance, as is exemplified by the larger thresholds of $\mathscr{C}_7$ in comparison with those of $\mathscr{C}_5$. □

The windowed thresholds $\varepsilon^{\text{WD}}_{i}(\mathbf{B}, W, \delta)$ quantify the unequal erasure protection of different VNs in the sub-protograph within the window. Furthermore, it is clear that for good performance, it is advantageous to keep fewer targeted symbols within a window.

5.3.2 Finite length performance evaluation

The finite length performance of LDPC codes under iterative message-passing decoding over the BEC is dependent on the number and the size of *stopping sets* present in the parity-check matrix of the code [22, 100]. Thus, the performance of the codes varies based on the parity-check matrix used to represent the code and, consequently, the performance of iterative decoding can be made to approach that of ML decoding by adding redundant rows to the parity-check matrix (See e.g. [41]). However, since we are exploiting the structure of the parity-check matrix of the convolutional code, we will not be interested in changing the parity-check matrix by adding redundant rows as this destroys the convolutional structure. The *ensemble stopping set size distribution* for some protograph-based LDPC codes was evaluated in [25] where it was shown that a minimum stopping set size that grows linearly in blocklength is important for the good performance of codes with short blocklengths. This analysis is similar to the analysis of the minimum distance growth rate of LDPC-CC ensembles—see [72] and references therein. It is worthwhile to note that although the minimum stopping set size grows linearly for protograph codes expanded using random permutation matrices, the same is not true for codes expanded using circulant permutation matrices [15]. In the following we will evaluate the finite length performance of codes constructed from $\mathscr{C}_m(J,K)$ ensembles with WD through Monte Carlo simulations. Here we considered WD with only the first $K'M$ symbols as the targeted symbols.

In Figures 5.5 and 5.6, the symbol error rate (SER) and the codeword error rate (CER) performance are depicted for codes $C_1 \in \mathscr{C}_1$ and $C_2 \in \mathscr{C}_2$, where the ensembles $\mathscr{C}_1$ and $\mathscr{C}_2$ were defined in Example 5.5. The codes used were those constructed by expanding the protographs using circulant matrices (and sums of circulant matrices) and techniques of progressive edge growth (PEG) [45] and approximate cycle extrinsic message degree (ACE) [113] to avoid small cycles in the Tanner graphs of the codes. The girth of both the codes C_1 and C_2 was 12. The parameters used for the construction were $L = 20$ and $M = 512$ so that the blocklength $n = LK'M = 20480$ and $R_L = 0.45$. The BP thresholds for ensembles $\mathscr{C}_1$ and $\mathscr{C}_2$ with $L = 20$ were 0.4883 and 0.4882 respectively. As is clear from Figure 5.5 and 5.6, code C_2 outperforms code C_1 for small window sizes ($W = 3, 5$), confirming the effectiveness of the proposed design rules for

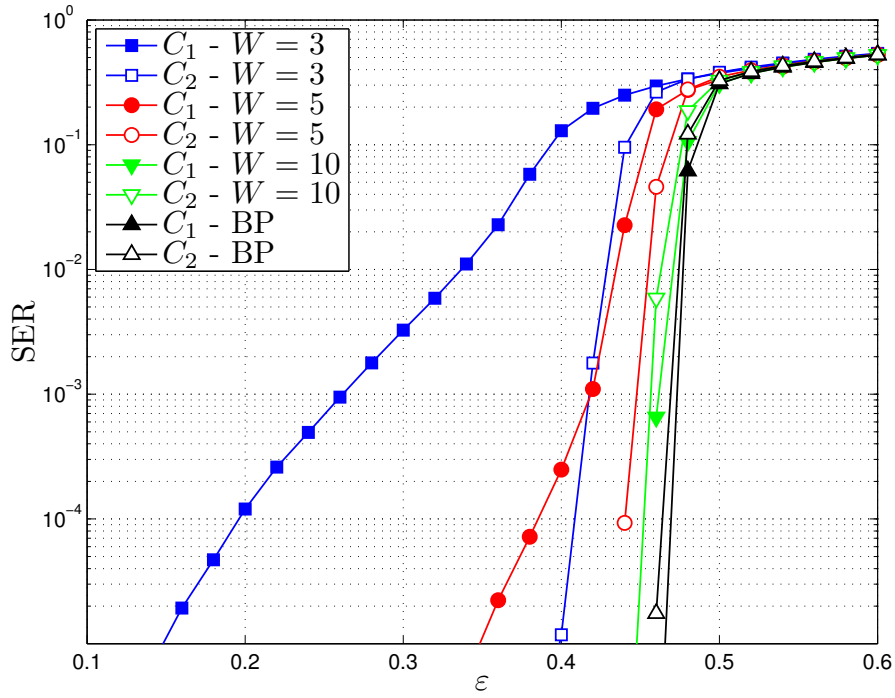

Figure 5.5: SER performance for Belief Propagation and Windowed Decoding over BEC.

windowed decoding. For larger window sizes ($W = 10$), there is no marked difference in the performance of the two codes. It was also observed that for small M values (< 128), the performance of codes constructed through circulant permutation matrices was better than those constructed through random permutation matrices. This difference in performance diminished for larger M values.

We include in Figure 5.6, for comparison, a lower bound on the CER P_{cw}. The Singleton bound, P_{SB}, represents the performance achievable by an idealized (n,k) binary MDS code. This bound for the BEC can be expressed as

$$\mathsf{P}_{cw} \geq \sum_{j=n-k+1}^{n} \binom{n}{j} \varepsilon^j (1-\varepsilon)^{n-j} = \mathsf{P}_{SB}.$$

Note that by the idealized (n,k) binary MDS code, we mean a binary linear code that achieves the Singleton bound $d_{min} \leq n-k+1$ with equality. This code does not exist for all values of k and n.

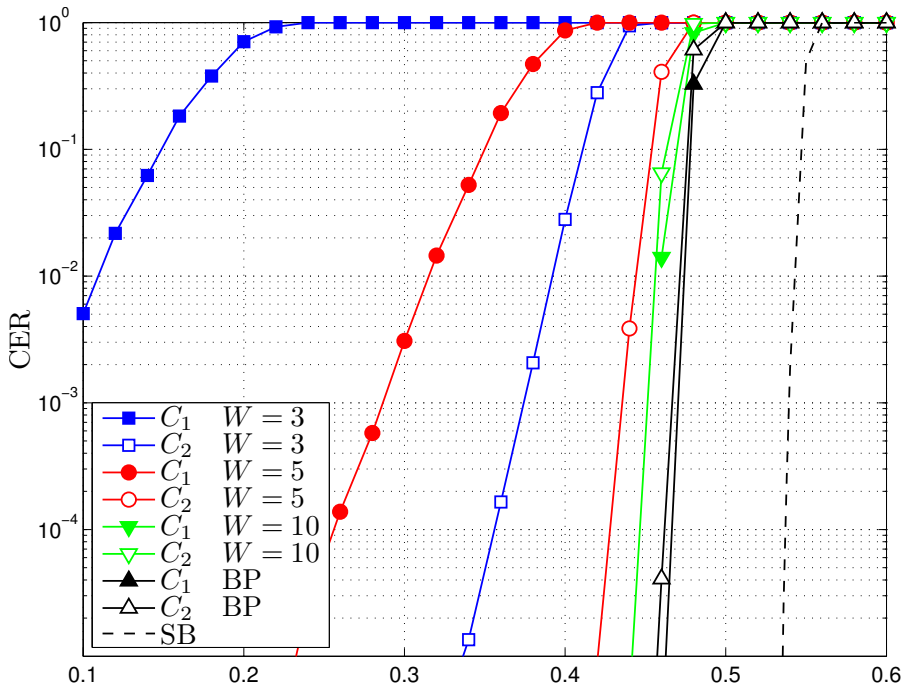

Figure 5.6: CER performance for BP and Windowed Decoding over BEC. Also shown is the (Singleton) lower bound P_{SB} as SB.

5.4 Erasure Channels with Memory

We now consider the performance of LDPC-CC ensembles and codes over erasure channels with memory. We consider the familiar two-state Gilbert-Elliott channel (GEC) [29, 40] as a model of an erasure channel with memory. In this model, the channel is either in a "good" state G, where we assume the erasure probability is 0, or in an "erasure" state E, in which the erasure probability is 1. The state process of the channel is a first-order Markov process with the transition probabilities $\mathsf{P}\{E \to G\} = g$ and $\mathsf{P}\{G \to E\} = b$. With these parameters, we can easily deduce [122] that the *average erasure rate* ε and the *average burst length* Δ are given by

$$\varepsilon = \mathsf{P}\{E\} = \frac{b}{b+g}, \qquad \Delta = \frac{1}{g}.$$

We will consider the GEC to be parameterized by the pair (ε, Δ). Note that there is a one-to-one correspondence between the two pairs (b, g) and (ε, Δ).

Discussion : The channel capacity of a correlated binary erasure channel with an average erasure rate of ε is given as $(1-\varepsilon)$, which is the same as that of the memoryless

channel, provided the channel is ergodic. Therefore, one can obtain good performance on a correlated erasure channel through the use of a capacity-achieving code for the memoryless channel with an interleaver to randomize the erasures [25, 125]. This is equivalent to permuting the columns of the parity-check matrix of the original code. We are not interested in this approach since such permutations destroy the convolutional structure of the code and as a result, we are unable to use the WD for such a scheme. □

Construction of LDPC block codes for bursty erasure channels has been well studied. The performance metric of a code over a bursty erasure channel is related to the *maximum resolvable erasure burst length* (MBL) denoted Δ_{max} [125], which, as the name suggests, is the maximal length of a single solid erasure burst that can be decoded by a BP decoder. Methods of optimizing codes for such channels therefore focus on permuting columns of parity-check matrices to maximize Δ_{max}, e.g. [51, 52, 76, 88, 89, 109]. Instead of permuting columns of the parity-check matrix, in order to maintain the convolutional structure of the code, we will consider designing $\mathscr{C}_m(J,K)$ ensembles that maximize Δ_{max}.

5.4.1 Asymptotic Analysis

As noted earlier, the performance of LDPC-CC ensembles depends on stopping sets. The structure of protographs imposes constraints on the code that limit the stopping set sizes and locations, as will be shown shortly. We will initially consider the performance with BP decoding, and later specialize to the case of WD. Since the base matrix with WD is a submatrix of that of the base matrix of the underlying protograph, the results for WD will be easily obtained once the results with BP are known.

Let us define a *protograph stopping set* to be a subset $\mathbb{S}(\mathbf{B})$ of the VNs of the protograph $\mathbf{B}$ whose neighboring CNs are connected at least twice to $\mathbb{S}(\mathbf{B})$. These are also denoted as $\mathbb{S}(\mathscr{P})$, in terms of the set of polynomials defining the protograph. We define the *size* of the stopping set as the cardinality of $\mathbb{S}(\mathbf{B})$, denoted $|\mathbb{S}(\mathbf{B})|$. We call the least number of consecutive columns of $\mathbf{B}$ that contain the stopping set $\mathbb{S}(\mathbf{B})$ the *span* of the stopping set, denoted $\langle\mathbb{S}(\mathbf{B})\rangle$. Let us denote the size of the smallest protograph stopping set of the protograph $\mathbf{B}$ by $|\mathbb{S}(\mathbf{B})|^*$, and the minimum number of consecutive

columns of the protograph $\mathbf{B}$ that contain a protograph stopping set by $\langle \mathbb{S}(\mathbf{B}) \rangle^*$. When the protograph under consideration is clear from the context, we will drop it from the notation and use $|\mathbb{S}|^*$ and $\langle \mathbb{S} \rangle^*$. The minimum span of a stopping set is of interest because we can give simple bounds for Δ_{max} based on $\langle \mathbb{S}(\mathbf{B}) \rangle^*$. Note that the stopping set of minimal size and the stopping set of minimal span are not necessarily the same set of VNs. However, we always have

$$|\mathbb{S}(\mathbf{B})|^* \leq \langle \mathbb{S}(\mathbf{B}) \rangle^*.$$

The following example clarifies the notation.

Example 5.8. Let us denote the base matrix corresponding to the protograph of the ensembles $\mathscr{C}_i$ of Example 5.5 as $\mathbf{B}^{(i)}, i = 1,2,3$. For ensembles $\mathscr{C}_1$ and $\mathscr{C}_3$, the first two columns of $\mathbf{B}^{(i)}, i = 1,3$ form a protograph stopping set, i.e., $\mathbb{S}(\mathbf{B}^{(i)}) = \{V_1, V_2\}, i = 1,3$ is a stopping set. This is clear from the highlighted columns below

$$\mathbf{B}^{(1)} = \begin{pmatrix} \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 & \cdots \\ \mathbf{1} & \mathbf{1} & 1 & 1 & 0 & 0 & \cdots \\ \mathbf{1} & \mathbf{1} & 1 & 1 & 1 & 1 & \cdots \\ \mathbf{0} & \mathbf{0} & 1 & 1 & 1 & 1 & \cdots \\ \mathbf{0} & \mathbf{0} & 0 & 0 & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix},$$

$$\mathbf{B}^{(3)} = \begin{pmatrix} \mathbf{2} & \mathbf{2} & 0 & 0 & 0 & 0 & \cdots \\ \mathbf{1} & \mathbf{1} & 2 & 2 & 0 & 0 & \cdots \\ \mathbf{0} & \mathbf{0} & 1 & 1 & 2 & 2 & \cdots \\ \mathbf{0} & \mathbf{0} & 0 & 0 & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Therefore, $|\mathbb{S}(\mathbf{B}^{(i)})|^* \leq 2$ and $\langle \mathbb{S}(\mathbf{B}^{(i)}) \rangle^* \leq 2$. Since no single column forms a protograph stopping set, $|\mathbb{S}(\mathbf{B}^{(i)})|^* \geq 2$ and $\langle \mathbb{S}(\mathbf{B}^{(i)}) \rangle^* \geq 2$, implying $|\mathbb{S}(\mathbf{B}^{(i)})|^* = \langle \mathbb{S}(\mathbf{B}^{(i)}) \rangle^* = 2, i = 1,3$.

For ensemble $\mathscr{C}_2$, the highlighted columns of $\mathbf{B}^{(2)}$ in the following matrix form

a protograph stopping set, i.e., $\mathbb{S}(\mathbf{B}^{(2)}) = \{V_1, V_4\}$ is a stopping set.

$$\mathbf{B}^{(2)} = \begin{pmatrix} \mathbf{2} & 2 & 0 & \mathbf{0} & 0 & 0 & \cdots \\ \mathbf{0} & 1 & 2 & \mathbf{2} & 0 & 0 & \cdots \\ \mathbf{1} & 0 & 0 & \mathbf{1} & 2 & 2 & \cdots \\ \mathbf{0} & 0 & 1 & \mathbf{0} & 0 & 1 & \cdots \\ \mathbf{0} & 0 & 0 & \mathbf{0} & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}.$$

Thus, $|\mathbb{S}(\mathbf{B}^{(2)})|^* \leq 2$ and $\langle \mathbb{S}(\mathbf{B}^{(2)}) \rangle^* \leq 4$. As no single column of $\mathbf{B}^{(2)}$ is a protograph stopping set and no three consecutive columns of $\mathbf{B}^{(2)}$ contain a protograph stopping set, it is clear that $|\mathbb{S}(\mathbf{B}^{(2)})|^* \geq 2$ and $\langle \mathbb{S}(\mathbf{B}^{(2)}) \rangle^* \geq 4$, so that

$$2 = |\mathbb{S}(\mathbf{B}^{(2)})|^* \leq \langle \mathbb{S}(\mathbf{B}^{(2)}) \rangle^* = 4.$$

In these cases, it so happened that the stopping set with the minimal size and the stopping set with the minimal span were the same. □

Our aim in the following will be to obtain bounds for the maximal $\langle \mathbb{S}(\mathbf{B}) \rangle^*$ over $\mathscr{C}_m(J,K)$ ensembles with memory m_s, which we denote $\langle \mathbb{S}(J,K,m_s) \rangle^*$, and design protographs that achieve minimal spans close to this optimal value.

The analysis of the minimal span of stopping sets for unstructured LDPC ensembles was performed in [118]. However, the structure of the protograph-based LDPC-CC allows us to obtain $\langle \mathbb{S}(J,K,m_s) \rangle^*$ much more easily for some $\mathscr{C}_m(J,K)$ ensembles.

We start by observing that if one of the VNs in the protograph is connected multiple times to all its neighboring CNs, then it forms a protograph stopping set by itself. In order to obtain a larger minimum span of stopping sets, it is desirable to avoid this case, and we include this as one of our design criteria.

Design Rule 5.3. For a $\mathscr{C}_m(J,K)$ ensemble, choose the polynomials $p_j(x)$ such that for every $j \in [K']$, there exists $0 \leq i_j \leq (m_s+1)J' - 1$ such that $p_j^{(i_j)} = 1$. □

Using the polynomial representation of LDPC-CC ensembles is helpful in this case since we can easily track stopping sets as those subsets that have polynomials whose coefficients are all larger than 1. From this fact, we can prove the following.

Proposition 5.4 ($\langle\mathbb{S}\rangle^*$ for $\mathscr{C}_m(J,2J)$ protographs)**.** *For $\mathscr{C}_m(J,2J)$ protographs of memory m_s defined by polynomials $p_1(x)$ and $p_2(x)$, $\langle\mathbb{S}\rangle^*$ can be upper bounded as*

$$\langle\mathbb{S}\rangle^* \leq \begin{cases} 2m_s, & 0 = i_2 \leq i_1, j_2 \leq j_1 = m_s \\ 2m_s - 1, & 0 = i_2 \leq i_1, j_1 < j_2 = m_s \\ 2m_s - 1, & 0 = i_1 < i_2, j_2 \leq j_1 = m_s \\ 2m_s - 2, & 0 = i_1 < i_2, j_1 < j_2 = m_s \end{cases}$$

where $i_l = \min\deg(p_l(x))$ and $j_l = \deg(p_l(x)), l = 1,2$.

Proof. From the definitions made in the statement of Proposition 5.4, we have $0 \leq i_l < j_l \leq m_s, l = 1,2$. We assume $i_l < j_l$ in order to satisfy Design Rule 5.3. Since the code has memory m_s, we have $i = \min\{i_1, i_2\} = 0$ and $j = \max\{j_1, j_2\} = m_s$. Consider the subset of columns of $\mathbf{B}$ corresponding to the polynomial $r(x) = p_1(x)b_1(x) + p_2(x)b_2(x)$ where

$$b_1(x) = \begin{cases} x^{i_2}, & i_2 = j_2 - 1 \\ x^{i_2} + x^{i_2+1} + \cdots + x^{j_2-1}, & i_2 < j_2 - 1 \end{cases}$$

and

$$b_2(x) = \begin{cases} x^{i_1}, & i_1 = j_1 - 1 \\ x^{i_1} + x^{i_1+1} + \cdots + x^{j_1-1}, & i_1 < j_1 - 1. \end{cases}$$

We claim that this is a protograph stopping set. To see this, consider the columns corresponding to the above subset with $\beta(p_1(x))$ and $\beta(p_2(x))$ as the column polynomials defining $\mathbf{B}$. We have

$$\begin{aligned} \hat{r}(x) &= \beta(p_1(x))b_1(x) + \beta(p_2(x))b_2(x) \\ &= (x^{i_1} + x^{j_1})(x^{i_2} + x^{i_2+1} + \cdots + x^{j_2-1}) \\ &\quad + (x^{i_2} + x^{j_2})(x^{i_1} + x^{i_1+1} + \cdots + x^{j_1-1}) \\ &= x^{i_1+i_2} + x^{i_1+i_2+1} + \cdots + x^{i_1+j_2-1} \\ &\quad + x^{j_1+i_2} + x^{j_1+i_2+1} + \cdots + x^{j_1+j_2-1} \\ &\quad + x^{i_1+i_2} + x^{i_1+i_2+1} + \cdots + x^{j_1+i_2-1} \\ &\quad + x^{i_1+j_2} + x^{i_1+j_2+1} + \cdots + x^{j_1+j_2-1} \\ &= 2x^{i_1+i_2} + \cdots + 2x^{i_1+j_2-1} + 2x^{i_1+j_2} + \cdots + 2x^{j_1+j_2-1} \end{aligned}$$

when $j_l > i_l + 1, l = 1,2$. Similarly, it can be verified that $\hat{r}(x)$ has all coefficients equal to 2 in all other cases also. Clearly, $\hat{r}(x) \preceq r(x)$ and thus $r(x)$ can only differ from $\hat{r}(x)$ in having larger coefficients. Therefore, $r(x)$ also has all coefficients greater than 1. This shows that the chosen subset of columns form a protograph stopping set. Based on the parameters $i_l, j_l, l = 1,2$, we can count the number of columns included in the span of this stopping set and therefore give upper bounds on $\langle \mathbb{S} \rangle^*$ as claimed :

$$\langle \mathbb{S} \rangle^* \leq \begin{cases} 2(j_1 - i_2), & 0 = i_2 \leq i_1, j_2 \leq j_1 = m_s \\ 2(j_2 - i_2) - 1, & 0 = i_2 \leq i_1, j_1 < j_2 = m_s \\ 2(j_1 - i_1) - 1, & 0 = i_1 < i_2, j_2 \leq j_1 = m_s \\ 2(j_2 - i_1 - 1), & 0 = i_1 < i_2, j_1 < j_2 = m_s. \end{cases}$$

□

We see from the above that $\langle \mathbb{S}(J, 2J, m_s) \rangle^* \leq 2m_s$ and a necessary condition for achieving this span is the first of four possible cases listed above, which we include as another design criterion.

Design Rule 5.4. For $\mathscr{C}_m(J, 2J)$ ensembles with memory m_s, set

$$\min \deg(p_2(x)) = 0 \text{ and } \deg(p_1(x)) = m_s. \qquad \square$$

Corollary 5.5 (Optimal $\mathscr{C}_m(J, 2J)$ protographs)**.** *For $\mathscr{C}_m(J, 2J)$ protographs with memory m_s and $J > 2$, $\langle \mathbb{S}(J, 2J, m_s) \rangle^* = 2m_s$.*

Proof. Consider the protograph of the ensemble given by $p_1(x) = (J-1) + x^{m_s}$ and $p_2(x) = (J-1) + x$. Let the polynomial $r(x) = p_1(x)a_1(x) + p_2(x)a_2(x)$ represent an arbitrary subset (chosen from the $2^{2m_s - 1} - 1$ non-empty subsets) of the first $(2m_s - 1)$ columns of $\mathbf{B}$, for any choice of polynomials $a_1(x)$ and $a_2(x)$ with coefficients in $\{0,1\}$ and maximal degrees $(m_s - 1)$ and $(m_s - 2)$ respectively:

$$a_i(x) = \sum_{j=0}^{d_i} a_i^{(j)} x^j, i = 1,2,\ d_1 = m_s - 1, d_2 = m_s - 2$$

where $a_i^{(j)} \in \{0,1\}$ and not all $a_i^{(j)}$s are zeros. When $a_1(x) \neq 0$, let $i_1 = \deg(a_1(x))$. Clearly, $r(x)$ is a monic polynomial of degree $(m_s + i_1)$. When $a_1(x) = 0$ and $a_2(x) \neq 0$,

let $i_2 = \deg(a_2(x))$. Then, $r(x)$ is a monic polynomial of degree $(1+i_2)$. Since in both these cases $r(x)$ is a monic polynomial, there is at least one coefficient equaling 1. Thus, $\langle \mathbb{S} \rangle^* > 2m_s - 1$. Finally, notice that

$$\begin{aligned} p_1(x) + x^{m_s-1} p_2(x) &= (J-1) + x^{m_s} + (J-1)x^{m_s-1} + x^{m_s} \\ &= (J-1) + (J-1)x^{m_s-1} + 2x^{m_s}, \end{aligned}$$

with all coefficients strictly larger than 1. Note that $p_1(x)$ corresponds to the first column of the protograph and $x^{m_s-1}p_2(x)$ to the $2m_s^{\text{th}}$ column. Thus, we have $\langle \mathbb{S} \rangle^* = 2m_s$. Since we have $\langle \mathbb{S}(J, 2J, m_s) \rangle^* \leq 2m_s$ from Proposition 5.4, we conclude that $\langle \mathbb{S}(J, 2J, m_s) \rangle^* = 2m_s$. □

Note that ensemble $\mathscr{C}_2$ in Example 5.5 achieves $\langle \mathbb{S}(J, 2J, m_s) \rangle^*$, as was observed in Example 5.8. It also satisfies Design Rules 5.1, 5.3 and 5.4. We bring to the reader's attention here that constructions other than the one given in the proof of the above corollary that achieve $\langle \mathbb{S} \rangle^* = 2m_s$ are also possible. These constructions allow us to design $\mathscr{C}_m(J, 2J)$ ensembles for a wide range of required $\langle \mathbb{S} \rangle^*$. We quickly see that a drawback of the convolutional structure is that if m_s is increased to obtain a larger $\langle \mathbb{S} \rangle^*$, the code rate R_L decreases linearly for a fixed L.

We give without proof the following upper bound for $\langle \mathbb{S} \rangle^*$ for $\mathscr{C}_m(J, K'J)$ ensembles, as it follows from Proposition 5.4.

Proposition 5.6 ($\langle \mathbb{S} \rangle^*$ for $\mathscr{C}_m(J, K'J)$ protographs)**.** *For $\mathscr{C}_m(J, K'J)$ protographs defined by polynomials $\mathscr{P} = \{p_j(x), j \in [K']\}$, we have*

$$\langle \mathbb{S} \rangle^* \leq \min_{(l_1, l_2) \in [K']^2, l_1 < l_2} \{ \overline{\langle \mathbb{S}_{l_1, l_2} \rangle} \}$$

where $\overline{\langle \mathbb{S}_{l_1, l_2} \rangle}$ is the upper bound for the minimal span $\langle \mathbb{S}_{l_1, l_2} \rangle$ of stopping sets $\mathbb{S}_{l_1, l_2}$ confined within subsets of the form $r_{l_1, l_2}(x) = a_1(x)p_{l_1}(x) + a_2(x)p_{l_2}(x)$ given by

$$\overline{\langle \mathbb{S}_{l_1, l_2} \rangle} = \begin{cases} K'(m_s(l_1, l_2) - 1) + (l_2 - l_1 + 1), & i(l_1, l_2) = i_{l_2} \leq i_{l_1}, j_{l_2} \leq j_{l_1} = j(l_1, l_2) \\ K'(m_s(l_1, l_2) - 1) + 1, & i(l_1, l_2) = i_{l_2} \leq i_{l_1}, j_{l_1} < j_{l_2} = j(l_1, l_2) \\ K'(m_s(l_1, l_2) - 1) + 1, & i(l_1, l_2) = i_{l_1} < i_{l_2}, j_{l_2} \leq j_{l_1} = j(l_1, l_2) \\ K'(m_s(l_1, l_2) - 1) - (l_2 - l_1 - 1), & i(l_1, l_2) = i_{l_1} < i_{l_2}, j_{l_1} < j_{l_2} = j(l_1, l_2). \end{cases} \tag{5.7}$$

where we have used the notation $i_{l_u} = \min\deg(p_{l_u}(x))$, $j_{l_u} = \deg(p_{l_u}(x))$, $u = 1,2$, $i(l_1,l_2) = \min\{i_{l_1}, i_{l_2}\}$, $j(l_1,l_2) = \max\{j_{l_1}, j_{l_2}\}$ *and* $m_s(l_1,l_2) = j(l_1,l_2) - i(l_1,l_2)$. □

Discussion : By looking at the stopping sets confined within columns corresponding to two polynomials only, we can use Proposition 5.4 to upper bound the span of these stopping sets. The minimal such span over all possible choices of the two columns therefore gives an upper bound on the minimal span of the $(J,K'J)$ protograph. Since $\overline{\langle \mathbb{S}_{l_1,l_2} \rangle} \leq K'm_s \; \forall \; l_1,l_2$ from Equation (5.7), we have $\langle \mathbb{S}(J,K'J,m_s) \rangle^* \leq K'm_s$, which is similar to the result in Proposition 5.4. This bound is, however, loose in general.

For terminated codes, we can give an upper bound for $\langle \mathbb{S} \rangle^*$ that is tighter in some cases.

Corollary 5.7 ($\langle \mathbb{S} \rangle^*$ for $\mathscr{C}_m(J,K)$ protographs)**.** *For* $\mathscr{C}_m(J,K)$ *protographs terminated after L instants,* $\langle \mathbb{S} \rangle^* \leq K'L$.

Proof. From the Singleton bound for the protograph, we have $\langle \mathbb{S} \rangle^* \leq J'(L+m_s)$. Since we need $m_s \leq \frac{R}{1-R}L$ for a positive code rate in (5.2), $\langle \mathbb{S} \rangle^* \leq \frac{J'L}{1-R} = K'L$.

Note that for $\mathscr{C}_m(J,K'J)$ protographs, this is tighter than the bound $\langle \mathbb{S} \rangle^* \leq K'm_s \leq K'(K'-1)L$, which, in the worst case, is a factor $(K'-1)$ times larger. However, since we are interested mainly in ensembles for which $m_s \ll L$, this bound might be looser than the one in Proposition 5.6 for $\mathscr{C}_m(J,K'J)$ ensembles. □

Example 5.9. Consider the $\mathscr{C}_m(J,K'J)$ ensemble with memory $m_s = u(K'-1)+1, m_s \leq (K'-1)L$ defined by the polynomials

$$p_l(x) = (J-1) + x^{j_l}, l \in [K'],$$

$j_l = m_s - u(l-1)$. It can be shown by an argument similar to the one used to prove Corollary 5.5 that for the protograph of this ensemble, $\langle \mathbb{S} \rangle^* = K'u+2$. This is exactly the bound in Proposition 5.6 since

$$\min_{l_1<l_2} \{ \overline{\langle \mathbb{S}_{l_1,l_2} \rangle} \} = \langle \mathbb{S}_{K'-1,K'} \rangle = K'u+2.$$

Thus, in this case

$$\langle \mathbb{S} \rangle^* = \frac{K'}{K'-1}(m_s-1)+2 = \left\lceil \frac{K'}{K'-1} m_s \right\rceil$$

which is roughly only a fraction of the (loose) upper bound for $\langle \mathbb{S}(J,K'J,m_s)\rangle^*$ suggested in the discussion of Proposition 5.6. The constructed $\mathscr{C}_m(J,K'J)$ protographs are thus optimal in the sense of maximizing the minimal span of stopping sets, i.e., $\langle \mathbb{S}(J,K'J,u(K'-1)+1)\rangle^* = K'u+2 \;\forall\; u \in [L-1]$. They also satisfy Design Rules 5.1 and 5.3 for $J > 2$. Although Proposition 5.6 gave a tight bound for $\langle \mathbb{S}\rangle^*$ in this case, it is loose in general. □

We can show that the $\mathscr{C}_m(J,K)$ protographs have minimal spans at least as large as the corresponding spans of $\mathscr{C}_m(a,K)$ protographs.

Proposition 5.8. $\langle \mathbb{S}(J,K,m_s)\rangle^* \geq \langle \mathbb{S}(a,K,m_s)\rangle^*$ *where* $a = \gcd(J,K) \geq 2$.

Proof. The equality is trivial when $a = J$. When $2 \leq a < J = aJ'$, one way of constructing the $\mathscr{C}_m(J,K)$ ensembles with memory m_s is to let each set of modulo polynomials $\mathscr{P}_l$ themselves define $\mathscr{C}_m(a,K)$ ensembles with memory m_s. The result then follows by noting that a stopping set for the polynomials $\mathscr{P}$ has to be a stopping set for every set of polynomials $\mathscr{P}_l$, $l = 0,1,\cdots,J'-1$. □

The construction proposed above often allows us to strictly increase the minimal span of the $\mathscr{C}_m(J,K)$ ensemble in comparison with the $\mathscr{C}_m(a,K)$ ensemble, as illustrated by the following example.

Example 5.10. Consider the construction of a $\mathscr{C}_m(4,6)$ ensemble with memory 3. Let us call it $\mathscr{C}_8$. The different parameters in this case are $J = 4$, $K = 6$, $a = 2$, $J' = 2$, $K' = 3$ and $m_s = 3$. Since $m_s = u(K'-1)+1$ with $u = 1$, we have for $\mathscr{C}_m(2,6)$ protographs, $\langle \mathbb{S}(2,6,3)\rangle^* = 5$ from Example 5.9 and we will define the modulo polynomials $\mathscr{P}_0$ to be the optimal construction that achieves this minimal span, i.e., $\mathscr{P}_0 = \{1+x^3, 1+x^2, 1+x\}$. Then, by defining $\mathscr{P}_1 = \{1+x^3, 1+x^3, 1+x^3\}$, we can show that $\langle \mathbb{S}(\mathscr{P})\rangle^* = 6$ and hence $\langle \mathbb{S}(4,6,3)\rangle^* \geq 6 > 5 = \langle \mathbb{S}(2,6,3)\rangle^*$. Note that the protograph defined by $\mathscr{P}$ has no degree-1 VNs associated with the component matrix $\mathbf{B}_0$. In fact, the constructed $\mathscr{C}_m(4,6)$ ensemble has $\varepsilon^{\mathrm{WD}}(\mathscr{C}_8, m_s+1, 10^{-12}) \approx 0.6469$, fairly close to the Shannon limit of $\varepsilon^{Sh} = \frac{2}{3}$, even with the smallest possible window size. Table 5.4 lists the windowed thresholds of this ensemble with different numbers of targeted symbols within the smallest window for $\delta = 10^{-12}$. □

Table 5.4: WD thresholds $\varepsilon_i^{\mathrm{WD}}$ for the ensemble $\mathscr{C}_8$, with iK' targeted symbols, window size m_s+1, and $\delta = 10^{-12}$.

i	1	2	3	4
$\varepsilon_i^{\mathrm{WD}}$	0.6469	0.6184	0.5803	0.4997

The asymptotic analysis for WD is essentially the same as that for BP. We will consider WD with only the first K' symbols within each window as the targeted symbols. We are now interested in the sub-protograph stopping sets, denoted $\mathbb{S}(\mathbf{B},W)$, that include one or more of the targeted symbols within a window. Let us denote the minimal span of such stopping sets as $\langle \mathbb{S}(\mathbf{B},W)\rangle^*$. Since stopping sets of the protograph of the LDPC-CC are also stopping sets of the sub-protograph within a window, and since such stopping sets can be chosen to include some targeted symbols within the window, we have $\langle \mathbb{S}(\mathbf{B},W)\rangle^* \leq \langle \mathbb{S}(\mathbf{B})\rangle^*$. In fact, $\langle \mathbb{S}(\mathbf{B},W)\rangle^* = \langle \mathbb{S}(\mathbf{B})\rangle^*$ when

$$W \geq \left\lceil \frac{\langle \mathbb{S}\rangle^*}{K'} \right\rceil + m_s$$

since in this case the first $K' \left\lceil \frac{\langle \mathbb{S}\rangle^*}{K'} \right\rceil$ columns are completely contained in the window. Further, we have

$$\langle \mathbb{S}(\mathbf{B},W)\rangle^* \leq \langle \mathbb{S}(\mathbf{B},W+1)\rangle^*.$$

This is true because a stopping set for window size W involving targeted symbols is not necessarily a stopping set for window size $W+1$, whereas a stopping set for window size $W+1$ is definitely a stopping set for window size W.

Remark 5.5. When the first iK' symbols within a window are the targeted symbols, we have for $i \leq W - m_s$

$$\langle \mathbb{S}_i(\mathbf{B},W)\rangle^* = \langle \mathbb{S}(\mathbf{B},W-i+1)\rangle^*$$

where $\langle \mathbb{S}_i(\mathbf{B},W)\rangle^*$ denotes the minimal span of stopping sets of the sub-protograph within the window of size W involving at least one of the iK' targeted symbols, and $\langle \mathbb{S}_1(\mathbf{B},W)\rangle^* = \langle \mathbb{S}(\mathbf{B},W)\rangle^*$. Consequently, we have

$$\langle \mathbb{S}_i(\mathbf{B},W)\rangle^* \leq \langle \mathbb{S}(\mathbf{B},W)\rangle^*.$$

The definition of $\langle \mathbb{S}_i(\mathbf{B},W)\rangle^*$ can be extended to accommodate $W - m_s + 1 \leq i \leq W$, as in the case of windowed thresholds. In particular, we have $\langle \mathbb{S}_W(\mathbf{B},W)\rangle^* = \langle \mathbb{S}(\mathbf{B}_0)\rangle^* \leq J'$, where the last inequality is from the Singleton bound. □

Example 5.11. Consider the ensemble $\mathscr{C}_2$ defined in Example 5.5. With a window of size $W = m_s + 1 = 3$, we have $\langle \mathbb{S}(\mathscr{C}_2, 3) \rangle^* = 2$ with the corresponding stopping set $\mathbb{S}_3 = \{V_2, V_3\}$ highlighted below

$$\begin{pmatrix} 2 & \mathbf{2} & \mathbf{0} & 0 & 0 & 0 \\ 0 & \mathbf{1} & \mathbf{2} & 2 & 0 & 0 \\ 1 & \mathbf{0} & \mathbf{0} & 1 & 2 & 2 \end{pmatrix}$$

and with a window size $W = 4$, we have $\langle \mathbb{S}(\mathscr{C}_2, 4) \rangle^* = \langle \mathbb{S}(\mathscr{C}_2) \rangle^* = 4$, and the corresponding stopping sets $\mathbb{S}_4 = \{V_1, V_4\}$ and $\mathbb{S}'_4 = \{V_1, V_2, V_4\}$ are as follows

$$\begin{pmatrix} \mathbf{2} & 2 & 0 & \mathbf{0} & 0 & 0 & 0 & 0 \\ \mathbf{0} & 1 & 2 & \mathbf{2} & 0 & 0 & 0 & 0 \\ \mathbf{1} & 0 & 0 & \mathbf{1} & 2 & 2 & 0 & 0 \\ \mathbf{0} & 0 & 1 & \mathbf{0} & 0 & 1 & 2 & 2 \end{pmatrix},$$

$$\begin{pmatrix} \mathbf{2} & \mathbf{2} & 0 & \mathbf{0} & 0 & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{1} & 2 & \mathbf{2} & 0 & 0 & 0 & 0 \\ \mathbf{1} & \mathbf{0} & 0 & \mathbf{1} & 2 & 2 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & 1 & \mathbf{0} & 0 & 1 & 2 & 2 \end{pmatrix}.$$

Note that for window size 3, whereas the minimal span of a stopping set involving VN V_2 is 2, that of a stopping set involving V_1 is 4. However, for window size 4, the stopping set involving V_1 with minimal span, denoted $\mathbb{S}_4$, and that involving V_2, $\mathbb{S}'_4$, each have a span of 4, although their cardinalities are 2 and 3 respectively. We have in this case, $\mathbb{S}_4 \subset \mathbb{S}'_4$. Notice that $\langle \mathbb{S}_2(\mathscr{C}_2, 4) \rangle^* = \langle \mathbb{S}(\mathscr{C}_2, 3) \rangle^* = 2$. □

5.4.2 Finite length analysis

We now show the relation between the parameters Δ_{max} and $\langle \mathbb{S}(\mathbf{B}) \rangle^*$. We shall assume in the following that $\langle \mathbb{S} \rangle^* \geq 2$, i.e., every column of the protograph has at least one of the entries equal to 1. We will consider the expansion of the protographs by a factor M to obtain codes.

Proposition 5.9. *For any (J,K) regular LDPC-CC, $\Delta_{max} \leq M \langle \mathbb{S} \rangle^* - 1$.*

Proof. Clearly, the set of the $M\langle\mathbb{S}\rangle^*$ columns of the parity-check matrix corresponding to the $\langle\mathbb{S}\rangle^*$ consecutive columns of $\mathbf{B}$ that contain the protograph stopping set with minimal span must contain a stopping set of the parity-check matrix. Therefore, if all symbols corresponding to these columns are erased, they cannot be retrieved. □

Corollary 5.10. *A terminated $\mathscr{C}_m(J,K'J)$ LDPC-CC with $m_s = u(K'-1)+1, u \in [L-1]$ can never achieve the MBL of an MDS code.*

Proof. From the Singleton bound, we have $\Delta_{max} \leq n-k = (L+m_s)M$, assuming that the parity-check matrix is full-rank. From Proposition 5.9 we have,

$$\Delta_{max} \leq M\langle\mathbb{S}(J,K'J,u(K'-1)+1)\rangle^* - 1 = \left\lceil \frac{K'}{K'-1} m_s \right\rceil M - 1$$

where the second equality follows from the discussion in Example 5.9. Since we require $m_s \leq \frac{R}{1-R}L = (K'-1)L$ for a non-negative code rate in (5.2),

$$\Delta_{max} \leq \left\lceil \frac{(K'-1)m_s + (K'-1)L}{K'-1} \right\rceil M - 1 < (L+m_s)M$$

which shows that the MBL of an MDS code can never be achieved. □

Remark 5.6. Although the idealized binary (n,k) MDS code does not exist, there are codes that achieve MDS performance when used over a channel that introduces a single burst of erasures in a codeword. For example, the $(2n,n)$ code with a parity-check matrix $\mathbf{H} = [\mathbf{I}_n\ \mathbf{I}_n]$ has an MBL of $\Delta_{max} = n$. □

Despite the discouraging result from Corollary 5.10, we can guarantee an MBL that linearly increases with $\langle\mathbb{S}\rangle^*$ as follows.

Proposition 5.11. *For any (J,K) regular LDPC-CC, $\Delta_{max} \geq M(\langle\mathbb{S}\rangle^* - 2) + 1$.*

Proof. From the definition of $\langle\mathbb{S}\rangle^*$, it is clear that if one of the two extreme columns is completely known, all other symbols can be recovered, for otherwise the remaining columns within the span of the stopping set S will have to contain another protograph stopping set, violating the minimality of the stopping set span $\langle\mathbb{S}\rangle^*$ (The two extreme columns are *pivots* of the stopping set [88].) The largest solid burst that is guaranteed to have at least one of the extreme columns completely known is of length $M(\langle\mathbb{S}\rangle^* - 2) + 1$. Therefore, $\Delta_{max} \geq M(\langle\mathbb{S}\rangle^* - 2) + 1$. □

Example 5.12. For the $\mathscr{C}_m(J,K'J)$ ensemble with memory $m_s = u(K'-1)+1, u \in [L-1]$ in Example 5.9, we have

$$MK'\left(\frac{m_s-1}{K'-1}\right)+1 \leq \Delta_{max} \leq MK'\left(\frac{m_s-1}{K'-1}\right)+2M-1$$

from Propositions 5.9 and 5.11. Thus, we can construct codes with MBL proportional to m_s. □

The MBL for WD $\Delta_{max}(W)$ can be bounded as in the case of BP based on $\langle \mathbb{S}(\mathbf{B},W)\rangle^*$. Assuming that the window size is $W \geq m_s+1$, the targeted symbols are the first K' symbols within the window, and the polynomials defining the ensemble are chosen to satisfy Design Rule 5.3, we have $\langle \mathbb{S}(\mathbf{B},W)\rangle^* \geq 2$. Propositions 5.9 and 5.11 in this case imply that

$$M(\langle \mathbb{S}(\mathbf{B},W)\rangle^*-2)+1 \leq \Delta_{max}(W) \leq M\langle \mathbb{S}(\mathbf{B},W)\rangle^*-1.$$

5.4.3 Numerical results

The MBL for codes C_1 and C_2 (the same codes used in Section 5.3.2) was computed using an exhaustive search algorithm, by feeding the decoder with a solid burst of erasures and testing all the possible locations of the burst. The MBL for the codes we considered was 1023 and 1751 for codes C_1 and C_2, respectively. Note that for code C_1, the MBL $\Delta_{max} = 1023 = 2M-1$, i.e., code C_1 achieves the upper bound from Proposition 5.9. More importantly, the maximum possible Δ_{max} was achievable while maintaining good performance over the BEC with the BP decoder. However, the MBL for code C_2, $\Delta_{max} = 1751 < 2047 = 4M-1$, is much smaller than the corresponding bound from Proposition 5.9. In this case, although other code constructions with Δ_{max} up to 2045 were possible, a trade-off between the BEC performance and MBL was observed, i.e., the code that achieved $\Delta_{max} = 2045$ was found to be much worse over the BEC than both codes C_1 and C_2 considered here. Such a trade-off has also been observed by others, e.g. [51]. This could be because the codes that achieve large Δ_{max} are often those that have a very regular structure in their parity-check matrices. Nevertheless, our code design does give a large increase in MBL ($> 70\%$) when compared with the corresponding codes constructed from $\mathscr{C}_c$ ensembles, without any decrease in

code rate (same m_s). The MBL achieved as a fraction of the maximum possible MBL $\Delta_{max}/(n-k)$ was roughly 9.1% and 15.5% for codes C_1 and C_2, respectively.

In Figure 5.7, 5.8 and 5.9, we show the CER performance obtained for codes C_1 and C_2 over GEC channels with $\Delta = 10, 50$ and 100 respectively, and $\varepsilon \in [0.1, 0.6]$. As

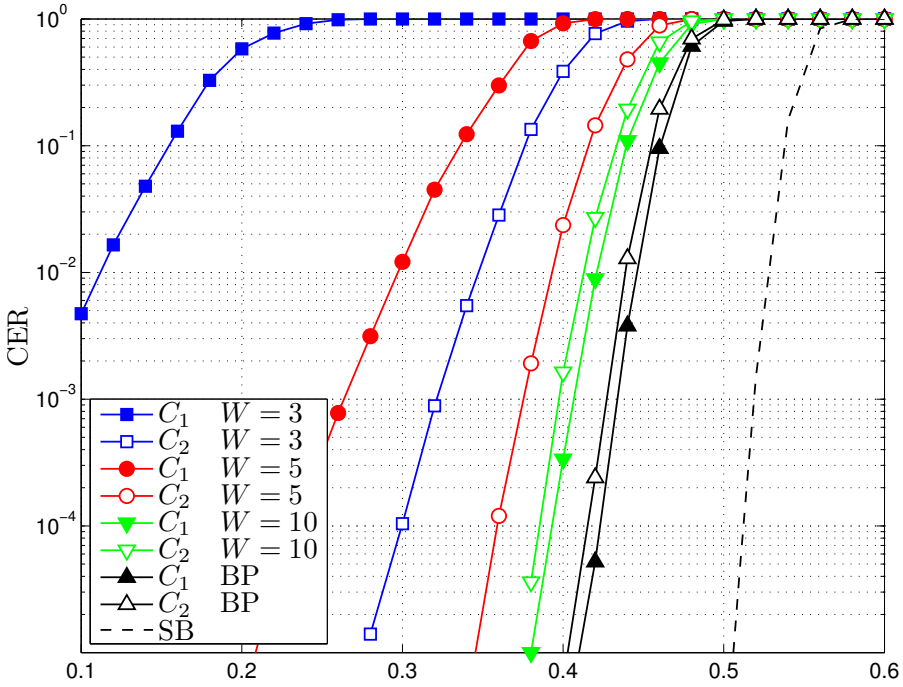

Figure 5.7: CER Performance on the Gilbert-Elliott Channel with $\Delta = 10$ with Singleton bound (SB).

can be seen from the figures, for $W = 3$, code C_2 always outperforms code C_1, while for $W = 5$ there is no such gain when $\Delta = 100$. However, for $W = 10$ and for BP decoding, code C_1 slightly outperforms C_2.

Note that the code C_2 outperforms C_1 for small ε when the average burst length $\Delta = 100$ for large window sizes and for BP decoding. This can be explained because in this regime, the probability of a burst is small but the average burst length is large. Therefore, when a burst occurs, it is likely to resemble a single burst in a codeword, and in this case we know that the code C_2 is stronger than C_1. Also note the significant gap between the BP decoder performance and the Singleton bound, suggesting that unlike some moderate length LDPC block codes with ML decoding [76], LDPC-CC are far from achieving MDS performance with BP or windowed decoding.

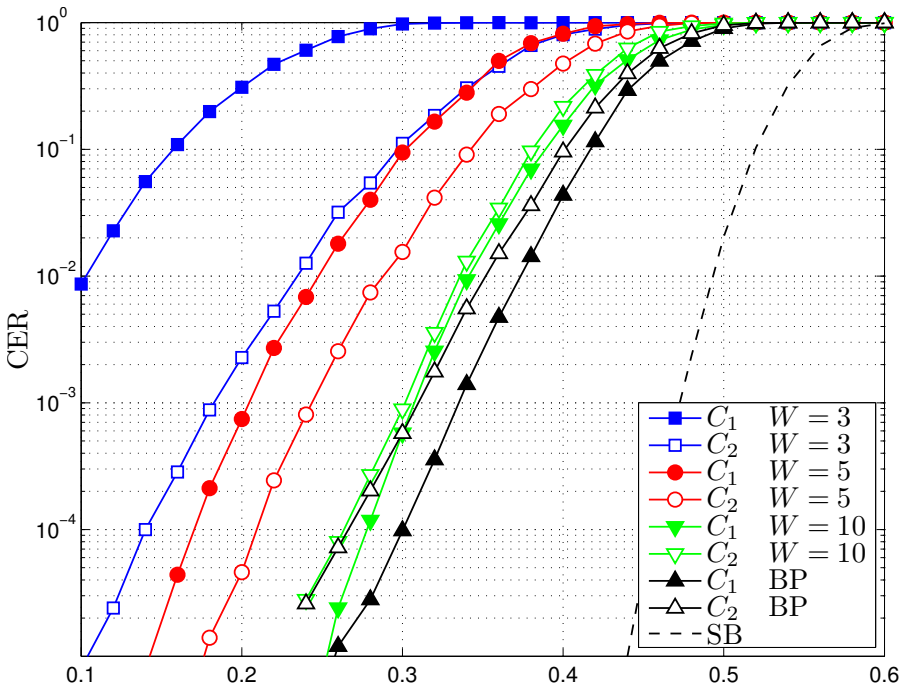

Figure 5.8: CER Performance on the Gilbert-Elliott Channel with $\Delta = 50$ with Singleton bound (SB).

5.5 Conclusions

We studied the performance of a windowed decoding scheme for LDPC convolutional codes over erasure channels. We showed that this scheme, when used to decode terminated LDPC-CC, provides an efficient way to trade-off decoding performance for reduced latency. Through asymptotic performance analysis, several design rules were suggested to avoid bad structures within protographs and, in turn, to ensure good thresholds. For erasure channels with memory, the asymptotic performance analysis led to design rules for protographs that ensure large stopping set spans. Examples of LDPC-CC ensembles that satisfy design rules for the BEC as well as erasure channels with memory were provided. Finite length codes belonging to the constructed ensembles were simulated and the validity of the design rules as markers of good performance was verified. The windowed decoding scheme can be used to decode LDPC-CC over other channels that introduce errors and erasures, although in this case error propagation due to wrong decoding within a window will have to be carefully dealt with.

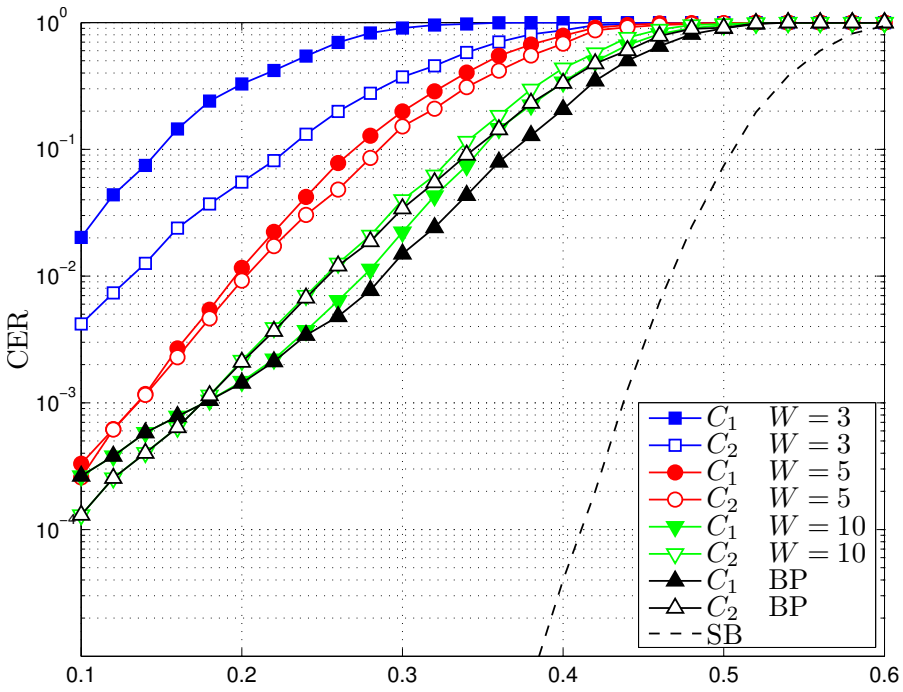

Figure 5.9: CER Performance on the Gilbert-Elliott Channel with $\Delta = 100$ with Singleton bound (SB).

For erasure channels, while close-to-optimal performance (in the sense of approaching capacity) was achievable for the BEC, we showed that the structure of LDPC-CC imposed constraints that bounded the performance over erasure channels with memory strictly away from the optimal performance (in the sense of approaching MDS performance). Nevertheless, the simple structure and good performance of these codes, as well as the latency flexibility and low complexity of the decoding algorithm, are attractive characteristics for practical systems.

Acknowledgement

This chapter contains material from the paper “Windowed Decoding of Protograph - based LDPC Convolutional Codes over Erasure Channels”, A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, G. E. Corazza; published in the IEEE Transactions on Information Theory, vol. 58, no. 4, Apr. 2012. The dissertation author was the primary investigator and author of this paper.

Bibliography

[1] R. Ahlswede and A. Kaspi, "Optimal coding strategies for certain permuting channels," *IEEE Trans. Inf. Theory*, vol. 33, no. 3, pp. 310–314, May 1987.

[2] A. Amraoui, "Asymptotic and finite-length optimization of LDPC codes," Ph.D. dissertation, EPFL, Switzerland, 2006.

[3] A. Amraoui and R. Urbanke, "LDPCOpt," Accessed Jan. 15, 2012, `http://ipgdemos.epfl.ch/ldpcopt/`.

[4] V. Anantharam and S. Verdu, "Bits through queues," *IEEE Trans. Inf. Theory*, vol. 42, no. 1, pp. 4–18, Jan. 1996.

[5] V. Aref and R. Urbanke, "Universal rateless codes from coupled LT codes," in *Information Theory Workshop (ITW), 2011 IEEE*, Oct. 2011, pp. 277 –281.

[6] D. Arnold, A. Kavcic, R. Kotter, H.-A. Loeliger, and P. Vontobel, "The binary jitter channel: a new model for magnetic recording," in *Proc. IEEE Int. Symp. Inf. Theory*, Sorrento, Italy, 25-30 Jun. 2000, p. 433.

[7] D. Arnold, H.-A. Loeliger, P. Vontobel, A. Kavcic, and W. Zeng, "Simulation-based computation of information rates for channels with memory," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3498 –3508, aug. 2006.

[8] R. B. Ash, *Information Theory*. New York : Wiley, 1965.

[9] C. P. M. J. Baggen, "An information theoretic approach to timing jitter," Ph.D. dissertation, University of California San Diego, La Jolla, CA, 1993.

[10] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (corresp.)," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[11] A. R. Barron, "Limits of information, Markov chains, and projection," in *Proc. IEEE Int. Symp. Inf. Theory*, Sorrento, Italy, Jun. 25-30, 2000, p. 25.

[12] T. W. Benjamin, "Coding for a noisy channel with permutation errors," Ph.D. dissertation, Cornell University, Ithaca, NY, 1975.

[13] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proc. IEEE Int. Conf. Comm.*, vol. 2, Geneva, Switzerland, May 23-26 1993, pp. 1064–1070.

[14] D. Blackwell, L. Breiman, and A. J. Thomasian, "Proof of Shannon's transmission theorem for finite-state indecomposable channels," *The Annals of Mathematical Statistics*, vol. 29, no. 4, pp. pp. 1209–1220, 1958. [Online]. Available: http://www.jstor.org/stable/2236957

[15] B. K. Butler and P. H. Siegel, "On distance properties of quasi-cyclic protograph-based LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, TX, USA, Jun. 13-18, 2010, pp. 809–813.

[16] J. Chen and P. Siegel, "Markov processes asymptotically achieve the capacity of finite-state intersymbol interference channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1295 –1303, march 2008.

[17] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. 2nd ed. Cambridge, Massachusetts: The MIT Press, 2001.

[18] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. 2nd ed. New York: John Wiley and Sons, 2006.

[19] L. Cui and A. Eckford, "The delay selector channel: Definition and capacity bounds," in *12th Canadian Workshop on Information Theory (CWIT)*, May 2011, pp. 15–18.

[20] M. Dalai, "A new bound on the capacity of the binary deletion channel with high deletion probabilities," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, 31 Jul.-5 Aug. 2011, pp. 410–413.

[21] C. A. Desoer, "Communication through channels in cascade," Ph.D. dissertation, MIT, Cambridge, Massachusetts, 1953.

[22] C. Di, D. Proietti, I. Telatar, T. Richardson, and R. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570 –1579, Jun. 2002.

[23] S. Diggavi and M. Grossglauser, "On information transmission over a finite buffer channel," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1226–1237, Mar. 2006.

[24] S. Diggavi, M. Mitzenmacher, and H. D. Pfister, "Capacity upper bounds for the deletion channel," in *Proc. IEEE Int. Symp. Inf. Theory*, Nice, France, Jun. 24-29, 2007, pp. 1716–1720.

[25] D. Divsalar, S. Dolinar, and C. Jones, "Protograph LDPC codes over burst erasure channels," JPL INP, Tech. Rep., Tech. Rep., Oct. 2006.

[26] R. L. Dobrushin, "Passage to the limit under the information and entropy signs," *Theory of Probability and its Applications*, vol. 5, no. 1, pp. 25–32, 1960.

[27] ——, "Shannon's theorems for channels with synchronization errors," *Problems Inform. Transmission*, vol. 3, no. 4, pp. 11–26, 1967.

[28] E. Drinea and M. Mitzenmacher, "Improved lower bounds for the capacity of i.i.d. deletion and duplication channels," *IEEE Trans. Inf. Theory*, vol. 53, no. 8, pp. 2693 –2714, Aug. 2007.

[29] E. Elliott, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J*, vol. 42, pp. 1977–1997, Sep. 1963.

[30] K. Engdahl and K. S. Zigangirov, "On the theory of low-density convolutional codes i," *Problemy Peredachi Informatsii*, vol. 35, pp. 12–27, 1999.

[31] K. Engdahl, M. Lentmaier, and K. Zigangirov, "On the theory of low-density convolutional codes," in *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 1999, vol. 1719, pp. 77–86.

[32] A. Feinstein, "On the coding theorem and its converse for finite-memory channels," *Information and Control*, vol. 2, no. 1, pp. 25–44, 1959.

[33] A. J. Felstrom and K. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 2181–2191, Sep. 1999.

[34] D. Fertonani and T. Duman, "Novel bounds on the capacity of binary channels with deletions and substitutions," in *Proc. IEEE Int. Symp. Inf. Theory*, 28 Jun.-3 Jul. 2009, pp. 2552 –2556.

[35] ——, "Novel bounds on the capacity of the binary deletion channel," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2753 –2765, Jun. 2010.

[36] D. Fertonani, T. Duman, and M. Erden, "Bounds on the capacity of channels with insertions, deletions and substitutions," *IEEE Trans. Commun.*, vol. 59, no. 1, pp. 2–6, Jan. 2011.

[37] R. G. Gallager, "Sequential decoding for binary channels with noise and synchronization errors," *Lincoln Lab. Group Report*, 1961.

[38] ——, *Low Density Parity Check Codes*. Cambridge, Massachusetts: MIT Press, 1963.

[39] ——, *Information Theory and Reliable Communication*. New York: John Wiley and Sons, 1968.

[40] E. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J*, vol. 39, pp. 1253–1265, Sep. 1960.

[41] J. Han and P. H. Siegel, "Improved upper bounds on stopping redundancy," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 90 –104, Jan. 2007.

[42] H. Hasani, N. Macris, and R. Urbanke, "Coupled graphical models and their thresholds," in *2010 IEEE Information Theory Workshop*, Dublin, Ireland, Aug. 30-Sep. 3, 2010.

[43] J. Hu, T. Duman, M. Erden, and A. Kavcic, "Achievable information rates for channels with insertions, deletions, and intersymbol interference with i.i.d. inputs," *IEEE Trans. Commun.*, vol. 58, no. 4, pp. 1102 –1111, Apr. 2010.

[44] J. Hu, T. Duman, E. Kurtas, and M. Erden, "Bit-patterned media with written-in errors: modeling, detection, and theoretical limits," *IEEE Trans. Magn.*, vol. 43, no. 8, pp. 3517–3524, Aug. 2007.

[45] X.-Y. Hu, E. Eleftheriou, and D. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.

[46] K. A. S. Immink, *Codes for Mass Data Storage Systems*. Shannon Foundation Publishers, The Netherlands, 1999.

[47] A. R. Iyengar, M. Papaleo, G. Liva, P. H. Siegel, J. K. Wolf, and G. E. Corazza, "Protograph-based LDPC convolutional codes for correlated erasure channels," in *Proc. IEEE Int. Conf. Comm.*, Cape Town, South Africa, May 2010, pp. 1–6.

[48] A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, and G. E. Corazza, "Windowed decoding of protograph-based LDPC convolutional codes over erasure channels," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2303 –2320, Apr. 2012.

[49] A. R. Iyengar, P. H. Siegel, and J. K. Wolf, "LDPC codes for the cascaded BSC-BAWGN channel," in *Proc. 47th Annual Allerton Conf. on Communication, Control and Computing*, Sep. 30 - Oct. 2, 2009, pp. 620–627.

[50] ——, "Write channel model for bit-patterned media recording," *IEEE Trans. Magn.*, vol. 47, no. 1, pp. 35 –45, Jan. 2011.

[51] S. J. Johnson, "Burst erasure correcting LDPC codes," *IEEE Trans. Commun.*, vol. 57, no. 3, pp. 641–652, Mar. 2009.

[52] S. J. Johnson and T. Pollock, "LDPC codes for the classic bursty channel," in *Proc. IEEE Int. Symp. Inf. Theory*, Aug. 2004, pp. 184–189.

[53] P. Kabal and S. Pasupathy, "Partial-response signaling," *IEEE Trans. Commun.*, vol. 23, no. 9, pp. 921–934, Sep. 1975.

[54] A. Kalai, M. Mitzenmacher, and M. Sudan, "Tight asymptotic bounds for the deletion channel with small deletion probability," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, TX, USA, Jun. 13-18, 2010, pp. 997–1001.

[55] Y. Kanoria and A. Montanari, "On the deletion channel with small deletion probability," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, TX, USA, Jun. 13-18, 2010, pp. 1002–1006.

[56] ——, "Optimal coding for the deletion channel with small deletion probability," *CoRR*, vol. abs/1104.5546, 2011.

[57] A. Kavcic, "On the capacity of Markov sources over noisy channels," in *Proc. IEEE Globecom*, vol. 5, San Antonio, TX, USA, Nov. 25-29, 2001, pp. 2997–3001.

[58] A. Kirsch and E. Drinea, "Directly lower bounding the information capacity for channels with i.i.d. deletions and duplications," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 86 –102, Jan. 2010.

[59] D. E. Knuth, "Big omicron and big omega and big theta," *SIGACT News*, vol. 8, pp. 18–24, April 1976. [Online]. Available: http://doi.acm.org/10.1145/1008328.1008329

[60] H. Kobayashi, "A survey of coding schemes for transmission or recording of digital data," *IEEE Trans. on Comm. Techn.*, vol. 19, no. 6, pp. 1087–1100, Dec. 1971.

[61] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[62] S. Kudekar, C. Measson, T. J. Richardson, and R. L. Urbanke, "Threshold saturation on BMS channels via spatial coupling," *CoRR*, vol. abs/1004.3742, 2010.

[63] S. Kudekar, T. Richardson, and R. L. Urbanke, "Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803–834, Feb. 2011.

[64] ——, "Spatially coupled ensembles universally achieve capacity under belief propagation," *CoRR*, vol. abs/1201.2999, 2012.

[65] B. Kurkoski, P. Siegel, and J. Wolf, "Joint message-passing decoding of LDPC codes and partial-response channels," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1410 –1422, Jun. 2002.

[66] G. F. Lawler, *Introduction to Stochastic Processes*. Boca Raton, Florida: 2nd Ed. : Chapman & Hall/CRC, 2006.

[67] M. Lentmaier, G. P. Fettweis, K. S. Zigangirov, and D. J. Costello, "Approaching capacity with asymptotically regular LDPC codes," in *Proc. Inf. Theory and Applications*, San Diego, California, 2009.

[68] M. Lentmaier, A. Sridharan, D. J. Costello, and K. S. Zigangirov, "Iterative decoding threshold analysis for LDPC convolutional codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5274–5289, Oct. 2010.

[69] M. Lentmaier, D. V. Truhachev, and K. S. Zigangirov, "To the theory of low-density convolutional codes. II," *Problems of Information Transmission*, vol. 37, pp. 288–306, 2001.

[70] M. Lentmaier, D. Truhachev, K. Zigangirov, and D. Costello, "An analysis of the block error probability performance of iterative decoding," *IEEE Trans. Inf. Theory*, vol. 51, no. 11, pp. 3834–3855, Nov. 2005.

[71] M. Lentmaier and G. Fettweis, "On the thresholds of generalized LDPC convolutional codes based on protographs," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, TX, USA, Jun. 13-18, 2010, pp. 709–713.

[72] M. Lentmaier, D. G. M. Mitchell, G. P. Fettweis, and D. J. Costello, "Asymptotically regular LDPC codes with linear distance growth and thresholds close to capacity," in *Proc. Inf. Theory and Applications 2010*, San Diego, California, 2010.

[73] M. Lentmaier, A. Sridharan, K. S. Zigangirov, and D. J. Costello, "Terminated LDPC convolutional codes with thresholds close to capacity," in *Proc. IEEE Int. Symp. Inf. Theory*, Sep. 4-9, 2005, pp. 1372–1376.

[74] Z. Liu and M. Mitzenmacher, "Codes for deletion and insertion channels with segmented errors," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 224–232, Jan. 2010.

[75] G. Liva and M. Chiani, "Protograph LDPC codes design based on EXIT analysis," in *Proc. IEEE Globecom*, Washington, D.C., Nov. 27-30, 2007, pp. 3250–3254.

[76] G. Liva, B. Matuz, Z. Katona, E. Paolini, and M. Chiani, "On construction of moderate-length LDPC codes over correlated erasure channels," in *Proc. IEEE Int. Conf. Comm.*, Dresden, Jun. 2009, pp. 1–5.

[77] B. Livshitz, A. Inomata, H. Bertram, and V. Lomakin, "Semi-analytical approach for analysis of BER in conventional and staggered bit patterned media," *IEEE Trans. Magn.*, vol. 45, no. 10, pp. 3519–3522, Oct. 2009.

[78] M. Luby, M. Mitzenmacher, M. Shokrollahi, and D. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.

[79] ——, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 585 –598, Feb. 2001.

[80] D. MacKay and R. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 33, no. 6, pp. 457–458, Mar 1997.

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

[82] A. Mazumdar, A. Barg, and N. Kashyap, "Coding for high-density recording on a 1-d granular magnetic medium," *IEEE Trans. Inf. Theory*, vol. 57, no. 11, pp. 7403–7417, Nov. 2011.

[83] ——, "Coding for high-density magnetic recording," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, TX, USA, Jun. 13-18, 2010, pp. 978–982.

[84] M. Mitzenmacher, "Capacity bounds for sticky channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 1, pp. 72–77, Jan. 2008.

[85] ——, "A survey of results for deletion channels and related synchronization channels," *Probability Surveys*, no. 6, pp. 1–33, 2009.

[86] M. Mitzenmacher and E. Drinea, "A simple lower bound for the capacity of the deletion channel," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4657–4660, Oct. 2006.

[87] P. Olmos and R. Urbanke, "Scaling behavior of convolutional LDPC ensembles over the BEC," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, 31 Jul.-5 Aug. 2011, pp. 1816–1820.

[88] E. Paolini and M. Chiani, "Construction of near-optimum burst erasure correcting low-density parity-check codes," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1320–1328, May 2009.

[89] ——, "Improved low-density parity-check codes for burst erasure channels," in *Proc. IEEE Int. Conf. Comm.*, Istanbul, Jun. 2006, pp. 1183–1188.

[90] M. Papaleo, A. R. Iyengar, P. H. Siegel, J. K. Wolf, and G. Corazza, "Windowed erasure decoding of LDPC convolutional codes," in *2010 IEEE Information Theory Workshop*, Cairo, Egypt, Jan. 2010, pp. 78–82.

[91] A. Papoulis, *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill Inc., 1991.

[92] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Francisco, 1988.

[93] H. Permuter, P. Cuff, B. Van Roy, and T. Weissman, "Capacity of the trapdoor channel with feedback," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 3150 –3165, Jul. 2008.

[94] H. Pfister, J. Soriaga, and P. Siegel, "On the achievable information rates of finite state isi channels," in *Proc. IEEE Globecom 2001*, vol. 5, San Antonio, TX, USA, 25-29 Nov. 2001, pp. 2992–2996 vol.5.

[95] A. Pusane, A. J. Felstrom, A. Sridharan, M. Lentmaier, K. Zigangirov, and D. J. Costello, "Implementation aspects of LDPC convolutional codes," *IEEE Trans. Commun.*, vol. 56, no. 7, pp. 1060–1069, Jul. 2008.

[96] M. Rahmati and T. M. Duman, "Analytical lower bounds on the capacity of insertion and deletion channels," *CoRR*, vol. abs/1101.1310, 2011.

[97] S. I. Resnick, *A Probability Path*. Birkhauser Boston, 2005.

[98] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, New York, 2008.

[99] ——, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb 2001.

[100] ——, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 638 –656, Feb 2001.

[101] S. Shamai and E. Zehavi, "Bounds on the capacity of the bit-shift magnetic recording channel," *IEEE Trans. Inf. Theory*, vol. 37, no. 3, pp. 863 –872, May 1991.

[102] C. Shannon, "The zero error capacity of a noisy channel," *IRE Trans. Inf. Theory*, vol. 2, no. 3, pp. 8–19, Sep. 1956.

[103] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J*, vol. 27, pp. 379–423, 623–656, Jul., Oct. 1948.

[104] R. Silverman, "On binary channels and their cascades," *IRE Trans. Inf. Theory*, vol. 1, no. 3, pp. 19–27, December 1955.

[105] M. Simon, "On the capacity of a cascade of identical discrete memoryless nonsingular channels (corresp.)," *IEEE Trans. Inf. Theory*, vol. 16, no. 1, pp. 100–102, Jan 1970.

[106] A. Sridharan, "Design and analysis of LDPC convolutional codes," Ph.D. dissertation, University of Notre Dame, Notre Dame, Indiana, 2005.

[107] A. Sridharan, D. V. Truhachev, M. Lentmaier, D. J. Costello, and K. S. Zigangirov, "Distance bounds for an ensemble of LDPC convolutional codes," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4537–4555, Dec. 2007.

[108] A. Sridharan, M. Lentmaier, D. J. Costello, and K. S. Zigangirov, "Convergence analysis of a class of LDPC convolutional codes for the erasure channel," in *Proc. 42nd Annual Allerton Conf. on Communication, Control and Computing*, Monticello, IL, USA, Sep. 29-Oct. 1, 2004, pp. 953–962.

[109] G. Sridharan, A. Kumarasubramanian, A. Thangaraj, and S. Bhashyam, "Optimizing burst erasure correction of LDPC codes by interleaving," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 1143–1147.

[110] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[111] R. Tanner, D. Sridhara, A. Sridharan, T. Fuja, and D. Costello, "LDPC block and convolutional codes based on circulant matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 2966 – 2984, Dec. 2004.

[112] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," JPL INP, Tech. Rep., Tech. Rep., Aug. 2003.

[113] T. Tian, C. Jones, J. Villasenor, and R. Wesel, "Selective avoidance of cycles in irregular LDPC code construction," *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1242–1247, Aug. 2004.

[114] H. Uchikawa, K. Kasai, and K. Sakaniwa, "Terminated LDPC convolutional codes over $GF(2^p)$," *CoRR*, vol. abs/1010.0060, 2010.

[115] R. Venkataramanan, S. Tatikonda, and K. Ramchandran, "Achievable rates for channels with deletions and insertions," *CoRR*, vol. abs/1102.5112, 2011.

[116] P. Vontobel, A. Kavcic, D. Arnold, and H.-A. Loeliger, "A generalization of the Blahut-Arimoto algorithm to finite-state channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 1887 –1918, May 2008.

[117] N. D. Vvedenskaya and R. L. Dobrushin, "The computation on a computer of the channel capacity of a line with symbol drop-out," *Problemy Peredachi Informatsii*, vol. 4, no. 3, pp. 92–95, 1968.

[118] T. Wadayama, "Ensemble analysis on minimum span of stopping sets," in *Proc. Inf. Theory and Applications 2006*, San Diego, California, 2006.

[119] F. Wang, D. Aktas, and T. M. Duman, "On capacity and coding for segmented deletion channels," in *Proc. 49th Annual Allerton Conf. on Communication, Control and Computing*, 28-30 Sep., 2011, pp. 1408–1413.

[120] R. White, R. Newt, and R. Pease, "Patterned media: a viable route to 50 Gbit/in^2 and up for magnetic recording?" *IEEE Trans. Magn.*, vol. 33, no. 1, pp. 990–995, Jan 1997.

[121] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linkoping University, Linkoping, Sweden, 1996.

[122] L. Wilhelmsson and L. B. Milstein, "On the effect of imperfect interleaving for the Gilbert-Elliott channel," *IEEE Trans. Commun.*, vol. 47, no. 5, pp. 681–688, May 1999.

[123] D. Williams, *Probability with Martingales*. Cambridge University Press, 1991.

[124] R. Wood, M. Williams, A. Kavcic, and J. Miles, "The feasibility of magnetic recording at 10 Terabits per square inch on conventional media," *IEEE Trans. Magn.*, vol. 45, no. 2, pp. 917–923, Feb. 2009.

[125] M. Yang and W. E. Ryan, "Design of LDPC codes for two-state fading channel models," in *The 5th International Symposium on Wireless Personal Multimedia Communications*, vol. 3, Oct. 2002, pp. 986–990.

[126] S. Yang, A. Kavcic, and S. Tatikonda, "Feedback capacity of finite-state machine channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 799–810, Mar. 2005.