Imperial College London
Department of Computing

Coding Against Synchronisation and Related Errors

João Miguel Lourenço Ribeiro

Submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing of Imperial College, February 2021

Copyright declaration

Statement of originality

I declare that the material presented in this thesis is my own work except where otherwise stated and referenced according to the established practices of the field.

João Miguel Lourenço Ribeiro, February 2021

Abstract

In this thesis, we study aspects of coding against synchronisation errors, such as deletions and replications, and related errors. Synchronisation errors are a source of fundamental open problems in information theory, because they introduce correlations between output symbols even when input symbols are independently distributed. We focus on random errors, and consider two complementary problems:

- We study the optimal rate of reliable information transmission through channels with synchronisation and related errors (the *channel capacity*). Unlike simpler error models, the capacity of such channels is unknown. We first consider the geometric sticky channel, which replicates input bits according to a geometric distribution. Previously, bounds on its capacity were known only via numerical methods, which do not aid our conceptual understanding of this quantity. We derive sharp analytical capacity upper bounds which approach, and sometimes surpass, numerical bounds. This opens the door to a mathematical treatment of its capacity. We consider also the geometric deletion channel, combining deletions and geometric replications. We derive analytical capacity upper bounds, and notably prove that the capacity is bounded away from the maximum when the deletion probability is small, meaning that this channel behaves differently than related well-studied channels in this regime. Finally, we adapt techniques developed to handle synchronisation errors to derive improved upper bounds and structural results on the capacity of the discrete-time Poisson channel, a model of optical communication.

- Motivated by portable DNA-based storage and trace reconstruction, we introduce and study the *coded trace reconstruction problem*, where the goal is to design efficiently encodable high-rate codes whose codewords can be efficiently reconstructed from few reads corrupted by deletions. Remarkably, we design such n-bit codes with rate $1-O(1/\log n)$ that require exponentially fewer reads than average-case trace reconstruction algorithms.

Acknowledgements

First and foremost, I would like to thank my amazing advisor, Mahdi. The past three and a half years were all but linear, but he always made sure I got the most out of this experience. Mahdi knows a lot about *a lot* of stuff, and has the creativity to put this knowledge to great use. He always has insightful answers and sound advice, technical and otherwise, and doing research alongside him is a pleasure. I also thank him for hosting me in Ann Arbor, and for getting me home safe on short notice when the pandemic broke out. I am looking forward to visiting again soon, hopefully with nicer weather! Above all, Mahdi is an outstanding person, and I am honoured to call him my advisor.

I am grateful to Divesh, Maciej, and Olgica for their care and mentorship. Visiting them has led to enjoyable and lasting collaborations. Divesh and Maciej hosted me in Singapore and taught me a lot about pseudorandomness. They are strong theorists and good friends: I had a lot of fun at the Holi festival and the many dinners at the Korean BBQ place. I also had a great time in Urbana, hosted by Olgica, who has a knack for extracting (and solving!) meaningful information-theoretic problems from her vast practical knowledge. I learned a lot about DNA-based data storage and associated problems while there. I am also grateful to Abhi, Roberto, Ryan, and Sri for welcoming me and showing me around town.

I thank André, Paulo, and Ueli for their significant influence on my earlier development as a researcher. André and Paulo patiently introduced me to research in theoretical computer science during my undergraduate years in Lisbon. Ueli went above and beyond during my time in Zurich. He advised me closely, and it was fascinating to learn from his unique perspective on information-theoretic cryptography. I also thank Paulo for hosting me in Lisbon several times during my PhD.

I enjoy doing research in collaborative environments, and I was lucky to have had the pleasure of working together with many great people: Abhishek Agarwal, Divesh Aggarwal, Gianluca Brian, Mahdi Cheraghchi, Ivan Damgård, Joseph Downs, Antonio Faonio, Ryan Gabrys, Siyao Guo, Venkatesan Guruswami, Daniel Jost, Paulo Mateus, Ueli Maurer, Olgica Milenkovic, Jesper Buus Nielsen, Maciej Obremski, Srilakshmi Pattabiraman, Erick Purwanto, Vishal Rana, Mark Simkin, Luisa Siniscalchi, Maciej Skórski, André Souto, Noah Stephens-Davidowitz, Alexandra Veliche, Daniele Venturi, and Ivan Visconti. Although I can only fit a small fraction of the results obtained over the past few years in this thesis, all works were equally important to my growth as a theoretical computer scientist. I also thank Huck Bennett, Mario Berta, Francesco Borderi, Pedro Branco, Eshan Chattopadhyay, Salim El Rouayheb, Ricardo Faleiro, Jesse Goodman, Manuel Goulão, Vipul Goyal, Elena Grigorescu, Onur Günlü, Xin Li, Dimitrios Myrisiotis, Cyrus Rashtchian, Ilan Shomorony, and Minshen Zhu for

many enlightening discussions about cryptography, pseudorandomness, and information theory. Salim invited me to give a talk at the Shannon Channel on trace reconstruction, which I really appreciated. Interesting discussions about the deletion channel with Elena were kickstarted by a chance encounter at Caffé Bene in Urbana, and I thank her for her enthusiasm. I am also grateful to Cong Ling and Michael Mitzenmacher for agreeing to examine my thesis, and to Mario Berta for accepting to be my second supervisor. Finally, I thank Iosif Pinelis for suggesting an elegant approach for proving some inequalities in Chapter 3.

A special acknowledgement goes to Dimitrios for the companionship. We started our PhD's at the same time as Mahdi's first students in a new country, which made for a strong bond. Dimitrios is genuine and caring, and I am happy to call him a good friend. He has done great work on complexity theory, which is really cool but also really hard, and I always look forward to discussing research with him. I also thank the other people from ACE 358 and affiliated members, Alexei, Daniel, Dominik, Dragos, Francesco, Katerina, Lewis, Paul, Rami, Sam, and Sirvan for good times and interesting discussions in London, and Manel, Pedro, Rafael, and Ricardo for making me feel welcome as an unofficial group member in Lisbon.

My appreciation goes also to my great friends in Lisbon and Zurich. In particular, Gonçalo and Álvaro, with whom I have been especially close for the past twenty-odd years, David, with whom I got to share many concerts and some time in London, and Erik, Hèctor, Kaitlyn, and Karim, who made Zurich a great experience.

My brothers, parents, and grandparents have always supported me and made me feel loved. My wife, Sara, and I have been at each other's sides for several years, most of them with thousands of kilometers between us. Her support has been unwavering throughout, and I love her very much. To them I dedicate this thesis.

Publications

The material in this thesis is based on the following works. Author ordering is alphabetical (as usual in theoretical computer science), and does not reflect the contribution of each author.

- Mahdi Cheraghchi and João Ribeiro. Improved upper bounds and structural results on the capacity of the discrete-time Poisson channel. *IEEE Transactions on Information Theory*, 65(7):4052-4068, July 2019. A preliminary version was presented at the 2018 IEEE International Symposium on Information Theory.
- Mahdi Cheraghchi and João Ribeiro. Sharp analytical capacity upper bounds for sticky and related channels. *IEEE Transactions on Information Theory*, 65(11):6950-6974, Nov 2019. A preliminary version was presented at the 56th Annual Allerton Conference on Communication, Control, and Computing, 2018.
- Mahdi Cheraghchi, Ryan Gabrys, Olgica Milenkovic, and João Ribeiro. Coded trace reconstruction. *IEEE Transactions on Information Theory*, 66(10):6084-6103, Oct 2020. A preliminary version was presented at the 2019 IEEE Information Theory Workshop.
- Mahdi Cheraghchi and João Ribeiro. An overview of capacity results for synchronization channels. *IEEE Transactions on Information Theory*, to appear. DOI: 10.1109/TIT.2020.2997329.
- Mahdi Cheraghchi and João Ribeiro. Non-asymptotic capacity upper bounds for the discrete-time Poisson channel with positive dark current. Preprint at `https://arxiv.org/abs/2010.14858`.

I am the main author in all publications above, and I was in addition fully responsible for the writeup of all papers. Furthermore, an extension of my results from Section 5.5 was subsequently developed in collaboration with fellow PhD students Joseph Downs and Alexandra Veliche and is not discussed in this thesis:

- Mahdi Cheraghchi, Joseph Downs, João Ribeiro, and Alexandra Veliche. Mean-based trace reconstruction over practically any replication-insertion channel. *Proceedings of the 2021 IEEE International Symposium on Information Theory*, to appear. Preprint at `https://arxiv.org/abs/2102.09490`.

Other research produced during the course of my PhD which has not been included in this thesis is listed below in chronological order (one work marked with (*) does not follow alphabetical author ordering):

- Mahdi Cheraghchi and João Ribeiro. Simple codes and sparse recovery with fast decoding. *Proceedings of the 2019 IEEE International Symposium on Information Theory*, pages 156–160.

- Divesh Aggarwal, Ivan Damgård, Jesper Buus Nielsen, Maciej Obremski, Erick Purwanto, João Ribeiro, and Mark Simkin. Stronger leakage-resilient and non-malleable secret sharing schemes for general access structures. *Advances in Cryptology – CRYPTO 2019*, pages 510–539.

- Divesh Aggarwal, Maciej Obremski, João Ribeiro, Luisa Siniscalchi, and Ivan Visconti. How to extract useful randomness from unreliable sources. *Advances in Cryptology – EUROCRYPT 2020*, pages 343–372.

- Abhishek Agarwal, Olgica Milenkovic, Srilakshmi Pattabiraman, and João Ribeiro. Group testing with runlength constraints for topological molecular storage. *Proceedings of the 2020 IEEE International Symposium on Information Theory*, pages 132–137.

- Divesh Aggarwal, Siyao Guo, Maciej Obremski, João Ribeiro, and Noah Stephens-Davidowitz. Extractor lower bounds, revisited. *Proceedings of RANDOM 2020*, pages 1:1–1:20.

- Gianluca Brian, Antonio Faonio, Maciej Obremski, João Ribeiro, Mark Simkin, Maciej Skórski, and Daniele Venturi. The mother of all leakages: How to simulate noisy leakages via bounded leakage (almost) for free. *Advances in Cryptology – EUROCRYPT 2021*, to appear. Preprint at `https://eprint.iacr.org/2020/1246`.

- Divesh Aggarwal, Maciej Obremski, João Ribeiro, Mark Simkin, and Luisa Siniscalchi. Two-source non-malleable extractors and applications to privacy amplification with tamperable memory. Preprint at `https://eprint.iacr.org/2020/1371`.

- (*) Ryan Gabrys, Srilakshmi Pattabiraman, Vishal Rana, João Ribeiro, Mahdi Cheraghchi, Venkatesan Guruswami, and Olgica Milenkovic. AC-DC: Amplification curve diagnostics for Covid-19 group testing. Preprint at `https://arxiv.org/abs/2011.05223`.

Basic notation

$\mathbb{R}$	the set of real numbers
$\mathbb{R}_0^+$	the set of non-negative real numbers
$\mathbb{C}$	the set of complex numbers
$\mathbb{N}$	the set of natural numbers $\{1, 2, 3, \dots\}$
$\mathbb{N}_0$	the set $\{0\} \cup \mathbb{N}$
$[n]$	the set $\{1, 2, \dots, n\}$
$\mathbb{Z}$	the set of integers
$\log$	the base-2 logarithm
$\ln$	the natural logarithm
$\lceil x \rceil$	the ceiling of a real number x
$\lfloor x \rfloor$	the floor of a real number x
$\lceil x \rfloor$	the closest integer to a real number x, with ties broken by rounding down
$\mathcal{S}$	a set
ε	the empty string (when clear from context)
$\mathcal{S}^n$	the n-fold cartesian product of $\mathcal{S}$, with $\mathcal{S}^0 = \{\varepsilon\}$
$\mathcal{S}^*$	the set $\bigcup_{i=0}^{\infty} \mathcal{S}^i$
X	a random variable
$X(x)$	the probability that a discrete random variable X equals x
$\mathbf{1}_{\{E\}}$	the indicator random variable of event E
F_X	the cumulative distribution function of X (most relevant when X is not discrete)
$\mathbb{E}[X]$	the expected value of X
$\mathrm{Var}[X]$	the variance of X
$\mathrm{Cov}[X, Y]$	the covariance between X and Y
$X \sim P$	random variable X follows distribution P

$x \sim X$	when x is sampled according to X
$x \leftarrow \mathcal{S}$	when x is sampled uniformly at random from the set $\mathcal{S}$
$\Pr[E]$	the probability of an event E
$\Pr[E, E']$	the probability of events E and E' occurring simultaneously
$\Pr[E \mid E']$	the conditional probability of event E given E'
$\mathsf{supp}(X)$	the support of X. In general, this is the smallest closed set such that $\Pr[X \in \mathcal{S}] = 1$
$\mathrm{poly}(n)$	some fixed polynomial of the parameter n
$\lvert x \rvert$	the length of vector x
$\lVert x \rVert_p$	the p-norm of vector x
$x \oplus y$	bitwise XOR between bitstrings x and y
$\mathsf{wgt}(x)$	the Hamming weight of vector x
$x \Vert y$	the concatenation of vectors x and y
x^ℓ	the vector x concatenated ℓ times with itself (when clear from context)
$x[i, j)$	the substring $(x_i, x_{i+1}, \dots, x_{j-1})$
$x[i :]$	the substring $(x_i, x_{i+1}, \dots, x_n)$ if $\lvert x \rvert = n$
$x[: i]$	the substring $(x_1, x_2, \dots, x_i)$
y is subsequence of x	if there exist $i_1 < i_2 < \dots < i_{\lvert y \rvert}$ such that $x_{i_j} = y_j$
a run of x	a substring of x of the form s^ℓ for some symbol s

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Over the past few decades, data has become one of the most valuable resources in the world, and we have experienced an extreme increase in its usage and storage requirements. However, data storage technologies are naturally susceptible to errors that affect the integrity of stored data or the process of reading it. Therefore, it is imperative to encode data so that it can be reliably recovered from such errors.

Synchronisation errors, such as deletions and replications of data symbols or insertions of random symbols, occur when one attempts to read data off many data storage media. As mentioned in [6], examples include digital magnetic and optical storage systems [7, 8, 9, 10, 11], racetrack memories [12, 13, 14], and, most relevant to our motivation, latest generation portable DNA-based data storage systems [15, 16]. This type of errors also has intimate connections to other problems, such as multiple sequence alignment in computational biology [17], file synchronisation [18, 19, 20], and the behaviour of the length of the longest common subsequence between two strings [21, 22, 23].

At a high level, synchronisation errors cause a loss of synchronisation between the sender and receiver of a message, which makes them extremely difficult to analyse. While this is a general statement, a particularly instructive example of this phenomenon can be found in Figure 1.1. Note that the receiver, upon observing the output of the deletion process, is not sure which output bits correspond to which input bits. This is because several different deletion patterns on the same input string may lead to the same output string. In fact, there are three different ways of producing the string 010 by deleting three bits of the string 010101. On the other hand, this is not a problem if bits are *erased* instead of deleted: There is only one way of erasing three bits of the string 010101 to obtain the string ??010?.

In the case of erasures, the receiver knows that the i-th output symbol is always a possibly corrupted version of the i-th input symbol, and so there is no loss of synchronisation.

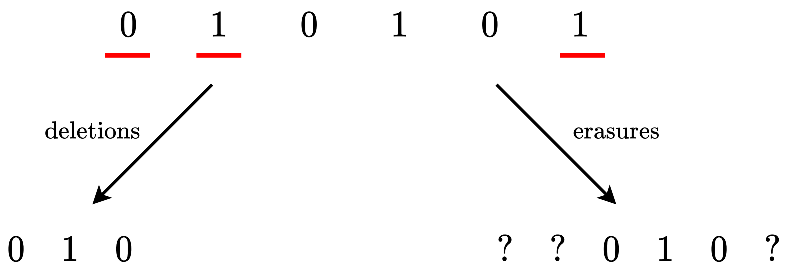

Figure 1.1: Comparison between deletions and erasures of the underlined bits.

A better understanding of the effect of synchronisation errors is key for designing improved data storage systems. With this in mind, it is natural to start by studying a simplified model. Since the errors introduced by the read process are non-adversarial, a good starting point is to study channels which on input an n-bit string $x \in \{0,1\}^n$ output a string Y obtained by independently corrupting each bit x_i according to a relevant probabilistic error model. Writing and reading data on/from a storage system is an instance of communication through an error-prone channel: The person who stores the data communicates it to the person retrieving it, and the data may be corrupted along the way. We are then interested in two fundamental information-theoretic problems in this model:

Question 1.1. How efficiently can we communicate through such a channel? In other words, how much redundancy must be added to a message before sending it through the channel so that every message can be recovered with high probability from its corresponding channel output? This is the same as asking for the *capacity* of the channel, a fundamental quantity in information theory first introduced and studied rigorously by Shannon [24, 25].

Note that Question 1.1 asks only to show whether a given redundancy is achievable, and the usual techniques exploited to derive achievability results yield only computationally intractable coding methods. This leads us to consider the following complementary problem.

Question 1.2. Can we design coding schemes for reliable information transmission under random synchronisation errors with redundancy as low as possible that also support *efficient* data encoding and decoding procedures?

These problems appear much harder to tackle for synchronisation errors than for other more common types of errors considered in information theory, such as erasures and bit-flips. While the analogous problems for erasures and bit-flips are largely understood, the questions above remain open even for simple models of synchronisation errors.

We present a telling example of this dichotomy. A basic channel in information theory is the Binary Erasure Channel (BEC), which independently erases each input bit with probability d. We have seen an example of this channel's behaviour in Figure 1.1. Shannon [24] showed that the capacity of the BEC is $1-d$, which means that it is both necessary and sufficient to encode a message of approximately $(1-d)n$ bits into n bits (i.e., add approximately dn bits of redundancy to the message) so that reliable information transmission is possible. The BEC falls within the general class of *Discrete Memoryless Channels* (DMCs), which on input $x = (x_1, x_2, \dots, x_n)$ output

$$Y = Y_1 \| Y_2 \| \cdots \| Y_n,$$

where $\|$ denotes string concatenation, and each Y_i is composed of a single symbol and is conditionally independent of $(Y_j)_{j \neq i}$ given x_i. The receiver, observing Y, knows that the i-th symbol of Y corresponds to Y_i, and hence to x_i. A large range of techniques have been developed over the past 70 years to study such channels, and we now have a great understanding of a large fraction of DMCs (for a good introduction to this topic, see [26, Section 7]).

Making a parallel with the above, one of the simplest channels with synchronisation errors (in short, a *synchronisation channel*) that is also practically motivated is the deletion channel, which independently deletes each input bit with probability d. Figure 1.1 showcases the behaviour of this channel. Surprisingly, although the deletion channel is simple to describe and is closely related to the BEC, determining its capacity is still a major open problem in information theory. Despite many efforts over the past decades, we still only have a rudimentary understanding of the capacity of the deletion channel and all other non-trivial synchronisation channels (see the surveys [27, 6] for exhaustive accounts of these efforts). As an example, we still do not even know whether the capacity curve of the deletion channel is a convex function of d [28]. One of the reasons for this bleak state of affairs is that most of the techniques in information theory are tailored to DMCs, and do not extend beyond this class of channels. The deletion channel is not a DMC, but falls within the class of what we may call *Discrete Memoryless Synchronisation Channels* (DMSCs). These are channels which on input $x = (x_1, \dots, x_n)$ output

$$Y = Y_1 \| Y_2 \| \cdots \| Y_n,$$

where, as before, each Y_i is conditionally independent of $(Y_j)_{j \neq i}$ given x_i, *but each Y_i may be composed of a different number of symbols.* Therefore, as was the case with the deletion channel, it may happen that someone observing Y output by a DMSC cannot tell which part of Y corresponds to Y_i, because it is not clear how many symbols of Y have been contributed by each Y_j. This is the main source of difficulty when analysing synchronisation channels compared to DMCs.

This thesis is divided into two parts, depending on whether we focus on Question 1.1 or 1.2. In the remainder of this chapter, we proceed to describe the concrete problems we study and our contributions.

1.1 Capacity upper bounds for the geometric sticky and geometric deletion channels

In the first part of this thesis, we study the capacity of a subset of so-called *repeat channels*, which are DMSCs that generalise the deletion channel. More precisely, on input $x \in \{0,1\}^n$, a repeat channel replaces each input bit x_i by R_i copies of it (with 0 copies meaning that x_i is deleted), where $R_1, R_2, \dots, R_n$ are independent and identically distributed (i.i.d.) over the non-negative integers. A deletion channel with deletion probability d can be seen as a repeat channel where the R_i's are independent Bernoulli random variables with success probability $1 - d$.

As a starting point, we consider *sticky channels*, first introduced by Drinea, Kirsch, and Mitzenmacher [29, 30, 31, 32]. These are special repeat channels where $R_i \geq 1$ with probability 1, meaning that no input bit ever gets deleted. Figure 1.2 illustrates the behaviour of such channels. Besides being natural information-theoretic objects, sticky channels have also been used to model nanopore-based DNA sequencing [33] (which is a key part of the read process of portable DNA-based storage systems [15, 16]) and small-sample distribution estimation problems in biology [34], and there has been work on codes correcting a limited number of sticky errors, such as duplications of input bits, with connections to communication chip design [35, 36] (see [37, Section III.F] for a more complete overview of works on coding against the more general notion of *weak synchronisation* errors).

Sticky channels are thought to be more tractable than channels with deletions, and understanding their behaviour may lead to insights about more general types of synchronisation channels [32]. The main reason for this belief is that their capacity is equivalent to the capacity *per unit cost* [38] of certain DMCs, which, as we have discussed before, are significantly more approachable in principle.

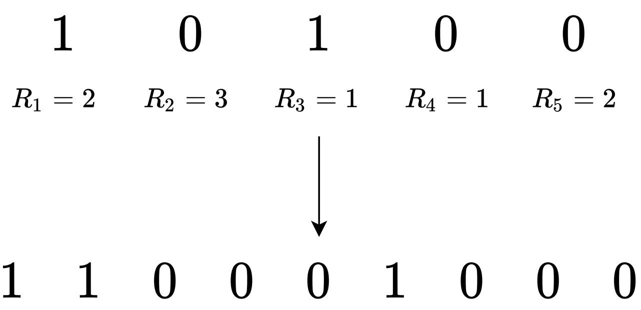

Figure 1.2: Behaviour of a sticky channel. Note that no input bits are deleted.

This connection between sticky channels and DMCs has been exploited in a series of works [32, 1, 39], and has led to tight *numerical* bounds on the capacity of two staple sticky channels: The elementary duplication channel, which independently replaces each input bit with two copies with probability p, and the geometric sticky channel, which independently replaces each input bit with a number of copies following a geometric distribution with support $\{1, 2, \dots\}$ and success probability p.

Although the numerical bounds above give an excellent computer-assisted estimate of the capacity of these channels for a given *replication parameter* p, i.e., the optimal redundancy required for reliable coding, they do not improve our conceptual understanding of the capacity curve of these channels as a function of p. Ideally, we would like to undertake a computer-unaided analysis of the capacity curve. Moreover, the numerical upper bounds above are obtained by combining a general convex optimisation-based framework with numerical analysis of a finite version of the channel in question obtained by providing extra information to the receiver, which strictly increases the capacity. Therefore, one cannot obtain an exact characterisation of the channel capacity via this approach. This means we require a different approach to move towards a deeper analysis of the capacity curve without computer assistance.

1.1.1 Contributions of this thesis

In this thesis, we undertake a different approach towards upper bounding the capacity of sticky channels using a different general framework based on convex optimisation originally developed in [40] to study channels with deletions. Surprisingly, we show that techniques which do not work for channels with deletions can be applied to yield sharp *analytical* upper bounds on the capacity of the geometric sticky channel for a large range of the replication parameter p. Here, by *analytical* we mean that our upper bounds are given by the supremum of an analytic function (which depends on p) over $(0, 1)$, and which can be easily approximated to the desired accuracy. These new upper bounds are an important step towards a mathematical, computer-unaided, treatment of these channels. Moreover, they improve upon previous numerical upper bounds for some values of p.

We continue to study the capacity of repeat channels by considering the capacity of a natural extension of the geometric sticky channel, called the *geometric deletion channel*, which independently replicates each input bit according to a geometric distribution with support on $\{0, 1, 2, \dots\}$ and parameter p. We begin by adapting the blueprint of [40] to obtain preliminary analytical capacity upper bounds for this channel. Then, we present a technique for generically improving these upper bounds by exploiting key properties of the geometric deletion channel. We numerically show that, under a plausible conjecture supported by numerical evidence and asymptotic results, our modified analytical bounds improve significantly on the initial bounds over a large range of p, and thus deserve further study. Notably, combining the technique above with surprising connections between the deletion and geometric deletion channels (without assuming any conjecture) also allows us to give a simple proof *without computer assistance* that the capacity of the geometric deletion channel is at most 0.73 bits/channel use when $p \to 1$ (equivalently, when the deletion probability $d = 1 - p \to 0$), which we call the *large replication regime*. This shows that the geometric deletion channel behaves differently than other well-studied repeat channels with deletions, such as the deletion channel and the Poisson-repeat channel [41, 40] (which replicates each bit according to a Poisson distribution with mean λ), whose capacities approach 1 bit/channel use in the large replication regime.

More details can be found in Chapter 3, which is based on [2, 6].

1.2 From synchronisation errors to the discrete-time Poisson channel

The strategy we use for upper bounding the capacity of a repeat channel starts by reducing this task to that of upper bounding the capacity of an associated *mean-limited* DMC (in the case of sticky channels, this reduction is lossless). Roughly speaking, a μ-mean-limited DMC is a DMC which only accepts input random variables X whose associated channel output Y satisfies $\mathbb{E}[Y] = \mu$, where $\mathbb{E}[Y]$ denotes the expected value of Y (we refer the interested reader to Section 2.5.1 for a more detailed discussion of the techniques in question). When studying the Poisson-repeat channel with mean $\lambda = 1$, the associated DMC is the channel which on input an integer $x \geq 0$ outputs a sample from the Poisson distribution with mean x, which we denote by Poi_x, with some output mean constraint μ. It is then natural to consider a continuous analogue of this channel where the input x is now allowed to be any non-negative real number, as opposed to a non-negative integer, since we hope that techniques developed to study the capacity of either of the two channels may be translated back and forth.

Remarkably, the continuous analogue of the DMC in question corresponds to the *Discrete-Time Poisson* (DTP) channel without dark current, a well-studied model of optical communication [42]. This channel can also be obtained from the well-known continuous-time Poisson channel model of optical communication by restricting the sender to more realistic transmission techniques. The input x to the DTP channel can be thought of as controlling the intensity of a photon-emitting source. The receiver then observes a probabilistic photon count induced by the intensity x, modelled by the Poi_x distribution. This model can be extended by considering the role of background interference on the photon count that the receiver observes, which is modelled as an additive *dark current* parameter $\lambda \geq 0$ to the mean photon count. Therefore, the DTP channel with dark current λ outputs a sample of $\mathsf{Poi}_{\lambda+x}$ on input x. A diagram of this model can be found in Figure 1.3. For practical reasons, one imposes constraints on the input X to the DTP channel: An average-power constraint μ, meaning that $\mathbb{E}[X] \leq \mu$, and possibly also a peak-power constraint A, meaning that $\Pr[X \leq A] = 1$. The DTP channel with an average-power constraint can then be seen as a natural continuous analogue of the mean-limited DMC associated with the Poisson-repeat channel.

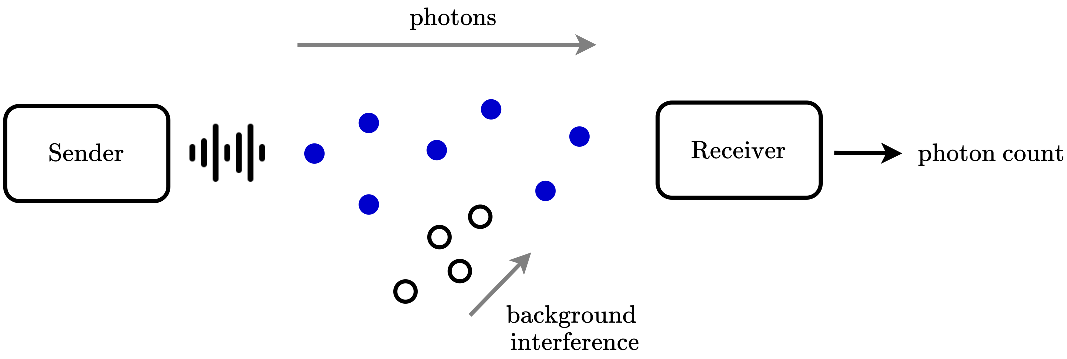

Figure 1.3: The DTP channel model of optical communication. The intensity of the sender's photon-emitting source is controlled by the channel input x, while the background intereference is controlled by the dark current parameter λ. The receiver observes a photon count following a Poisson distribution with mean $\lambda + x$.

Surprisingly, although the capacity of the continuous-time Poisson channel has been well-understood under different constraints for several decades [43, 44, 45, 46], the exact capacity of the DTP channel, both with and without dark current, under an average- and/or peak-power constraint is still unknown.

We only have a good grasp of its behaviour in asymptotic settings where the average-power constraint μ is either small [47, 48, 49] or large [50, 47, 51], with only loose upper bounds known in non-asymptotic regimes [47].

1.2.1 Contributions of this thesis

We focus on the DTP channel with arbitrary dark current $\lambda \geq 0$ and an average-power constraint μ only, and show that techniques used to study the capacity of the Poisson-repeat and geometric deletion channels can be adapted to make progress on our understanding of the capacity of the DTP channel.

First, we study capacity upper bounds for the DTP channel. In a setting without dark current ($\lambda = 0$), we show that techniques from [40], combined with a careful choice of parameters, lead to improved upper bounds on the capacity of the DTP channel with average-power constraint μ whenever μ is not tiny. To handle the case of arbitrary dark current $\lambda > 0$, we additionally employ techniques that we used to improve capacity upper bounds for the geometric deletion channel. Consequently, we are able to obtain improved upper bounds on the capacity of the DTP channel with dark current $\lambda > 0$. In both cases, the bounds have elementary expressions.

Afterwards, we turn to *structural* results on the capacity of the DTP channel. More precisely, we are interested in properties of the capacity-achieving distribution under given dark current and average/peak-power constraints. This is the input distribution satisfying the constraints that maximises the mutual information with its associated channel output distribution (we refer the interested reader to Section 2.4 for an introduction to basic information-theoretic concepts). One of the most basic properties of interest about capacity-achieving distributions is whether their support is discrete (finite or countably infinite). This property has been studied for several different channels, going back to the work of Smith [52] on the Additive White Gaussian Noise (AWGN) channel, and it is well-understood for *noise-additive* channels [53], of which the AWGN channel is an example. Shamai [42] showed that the capacity-achieving distribution of the DTP channel under a *finite* peak-power constraint has finite support, leaving open the case in which no peak-power constraint is present, where it is only known that the support is unbounded [54]. Using a different approach, we show that the support of the capacity-achieving distribution is countably infinite in this case, with only a finite number of mass points in every bounded interval, and we also recover the previous result for finite peak-power constraint with an alternative proof.

More details can be found in Chapter 4, which is based on [4, 55].

1.3 Coded trace reconstruction

In the second part of this thesis, we study Question 1.2 and construct low-redundancy coding schemes with efficient encoding and decoding procedures for a setting with random deletions motivated by the read process of recent portable DNA-based data storage systems [15, 16]. When we attempt to read data off such systems, we can obtain several copies of the data corrupted by deletions, replications, and insertions. Figure 1.4 schematises the writing and reading process of these DNA-based storage systems (for a detailed exposition of DNA-based data storage, see [56, 15, 16]). Therefore, it is of interest to design coding schemes with good tradeoffs between the redundancy that must be added and the number of copies (*traces*) required to reliably reconstruct the encoded data.

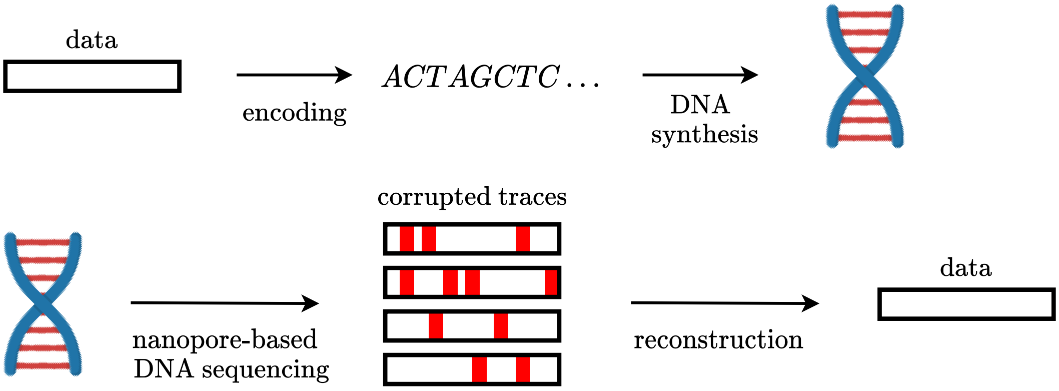

Figure 1.4: A diagram of the data writing and reading process in DNA-based data storage with nanopore-based sequencing.

We introduce the setting of *coded trace reconstruction*, a simplification of the scenario above where an encoded message is sent through several independent deletion channels with some deletion probability d. Figure 1.5 illustrates the trace reconstruction setting.

The goal is to design coding schemes with *efficient* encoding and decoding procedures with a good tradeoff between redundancy and number of traces required for reliable reconstruction. This setting already captures many of the difficulties of the real scenario, and is a first step towards obtaining efficient coding schemes with *provable* guarantees and scalability for such DNA-based storage systems. In contrast, the coding schemes used in [15, 16] are based on heuristics and designed for fixed system

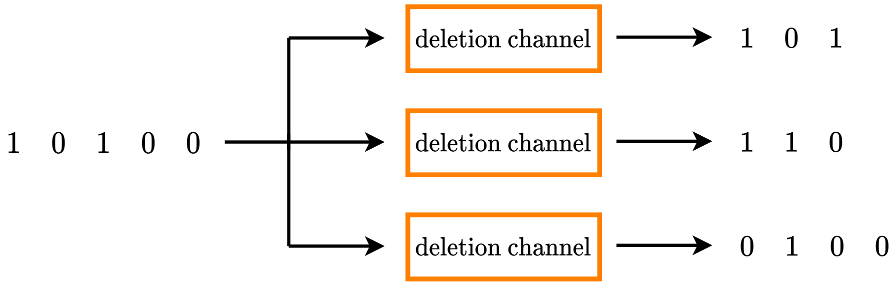

Figure 1.5: The trace reconstruction setting with three traces.

parameters, and have no provable guarantees.

We note that, similarly to previous questions about the channel capacity, the problem of reconstructing an input from several independent channel outputs also appears to be much more challenging for synchronisation channels than for DMCs. The analogous problem to coded trace reconstruction where we replace the deletion channel by a BEC with erasure probability d is trivial, in the sense that t independent BECs with erasure probability d are equivalent to a single BEC with erasure probability d^t.

The coded trace reconstruction problem is closely related to the well-studied *trace reconstruction* problem introduced in [17], originally motivated by applications in computational biology. The difference between the two settings is that in the original trace reconstruction problem the input string is not allowed to be encoded. There, the goal is to design reconstruction algorithms that either recover *all* input strings with high probability (*worst-case* trace reconstruction), or recover the input string with high probability on average over a uniform choice of the input string (*average-case* trace reconstruction). A bound of $t = t(n)$ traces for average-case trace reconstruction of n-bit strings can also be interpreted from the perspective of coded trace reconstruction: It states that there exist n-bit binary codes with 1 bit of redundancy whose codewords can be reconstructed from t traces. However, a key difference with respect to our goal in coded trace reconstruction is that the codes provided by average-case trace reconstruction are not known to be efficiently encodable. Nevertheless, as an intermediate goal towards obtaining the best tradeoff between redundancy and number of traces required for reconstruction, we aim to design efficiently encodable codes with as little redundancy as possible that can be reconstructed with significantly fewer traces than the best average-case trace reconstruction algorithms.

1.3.1 Contributions of this thesis

We present various efficiently encodable binary coding schemes with many desirable properties in the coded trace reconstruction setting with constant deletion probability by showing how to leverage results from *uncoded* trace reconstruction. We begin by combining a marker-based construction with worst-case trace reconstruction. This leads to efficiently encodable coding schemes with $O\left(\frac{n}{\log n}\right)$ bits of redundancy which can be efficiently reconstructed from $\exp(O(\log^{2/3} n))$ traces. Although this construction does not beat the best algorithms for average-case trace reconstruction, which require $\exp(O(\log^{1/3} n))$ traces [57], it is interesting due to its flexibility. Motivated by applications in DNA-based storage, we show how to adapt it to obtain an efficiently encodable code over the alphabet $\{A, C, G, T\}$ (each letter denoting a nucleotide in the DNA strand) which has *balanced GC-content*, an important property for DNA-based storage [58] which enforces that half of the symbols of each codeword must be either G or C, while the redundancy and number of traces required for efficient reconstruction remain the same (up to constant factors).

Subsequently, we leverage *average-case* trace reconstruction results to improve exponentially on the number of traces required by the previous construction, with the same redundancy. More precisely, by combining the average-case trace reconstruction algorithm from [59] with derandomisation techniques and a marker-based construction, we obtain efficiently encodable binary coding schemes still with $O\left(\frac{n}{\log n}\right)$ bits of redundancy which can be efficiently reconstructed from $\mathrm{poly}(\log n)$ traces, provided the deletion probability is smaller than some absolute constant. In particular, our result shows that by increasing the allowed redundancy from 1 bit to $O\left(\frac{n}{\log n}\right)$ bits, we can improve exponentially on the best known upper bound for average-case trace reconstruction *while ensuring efficient encoding and reconstruction.*

Finally, we also extend known techniques for analysing mean-based algorithms for worst-case trace reconstruction [59, 60, 61] to handle errors introduced by general repeat channels. In particular, this result allows us to extend some of our results on coded trace reconstruction over the deletion channel to coded trace reconstruction over other repeat channels.

More details can be found in Chapter 5, which is based on [5].

Chapter 2

General Background

The problem of determining the optimal rate of reliable information transmission in noisy regimes was first studied rigorously in the seminal work of Shannon [24, 25]. Shannon connected this optimal rate, the so-called *channel capacity*, to fundamental quantities in information theory such as the (Shannon) entropy and the mutual information. This connection allowed the computation of the exact capacity for several natural communication channels, and spurred more than 70 years of steady, fundamental developments in information theory (the book of Cover and Thomas [26] provides an excellent broad overview of the discipline). However, knowing the exact channel capacity, or a lower bound on this quantity, generally implies only that there exists some coding scheme with that given rate that can be used to reliably transmit information through the channel, and encoding and decoding messages through this coding scheme may be computationally intractable. Therefore, in parallel with the study of the capacity of several different channels, there has also been tremendous interest (both theoretical and practical) on the design of coding schemes for reliable information transmission through different channels, introducing as little redundancy as possible into the message, and supporting computationally efficient encoding and decoding procedures.

The goal of this chapter is to introduce common basic concepts and results that will be useful throughout the rest of this thesis, and to cover prior work related to the topics considered here. We begin by presenting some basic facts from probability theory, combinatorics, and the theory of special functions. This is followed by an introduction to information theory and related concepts that we will focus on in this thesis. This includes a comparison between synchronisation errors and other types of errors, such as erasures and substitutions. Finally, we present a detailed survey of the historical background of the problems studied in this thesis: Capacity bounds and efficient coding schemes for channels with synchronisation errors.

Mathematical background. Throughout this thesis, we assume the reader is familiar with discrete probability, as it will be the main focus of all remaining chapters. Moreover, some basic familiarity with measure-theoretic probability is required only for the proof of Theorem 4.1 in Appendix B. We note that Theorem 4.1 is a natural extension of previous results proved for other classes of channels, and its proof follows previous standard approaches with minor modifications only. We include an introductory section in Appendix B discussing the required concepts, in order to make the exposition as self-contained as possible.

2.1 Useful concepts from probability theory and combinatorics

In this section, we present some facts from discrete probability and combinatorics that will be useful throughout this thesis. We will often be dealing with a small number of important discrete probability distributions. These distributions and the corresponding notation are defined, along with some useful properties, in Table 2.1. A fundamental concept that can be associated to every discrete distribution[1] X over $\mathbb{N}_0$ is its *probability generating function* g_X, given by

$$g_X(z) = \sum_{i=0}^{\infty} X(i) \cdot z^i$$

for every $z \in \mathbb{C}$ whenever the infinite series on the right-hand side converges. Such probability generating functions will make a key appearance in all later chapters. For a complete treatment of discrete distributions and their properties, the book of Johnson, Kemp, and Kotz [62] is recommended.

We will need the following notion of distance between distributions with strong properties.

Definition 2.1 (Statistical distance). *Given distributions X and Y over a finite set $\mathcal{S}$, the* statistical distance between X and Y*, denoted by $\Delta(X;Y)$, is given by*

$$\Delta(X;Y) = \max_{\mathcal{T} \subseteq \mathcal{S}} |\Pr[X \in \mathcal{T}] - \Pr[Y \in \mathcal{T}]| = \frac{1}{2} \sum_{s \in \mathcal{S}} |X(s) - Y(s)|.$$

[1] We identify random variables with their distributions and let $X(x)$ stand for $\Pr[X = x]$ when X is a discrete random variable.

Distributions	PMF	Expected value	PGF
(Bernoulli) Ber_p	$\mathsf{Ber}_p(y) = p^y(1-p)^{1-y}, \quad y = 0, 1$	p	$1 - p + pz$
(Binomial) $\mathsf{Bin}_{n,p}$	$\mathsf{Bin}_{n,p}(y) = \binom{n}{y} p^y (1-p)^{n-y}, \quad y \geq 0$	np	$(1 - p + pz)^n$
(Geometric) $\mathsf{Geom}_{0,p}$	$\mathsf{Geom}_{0,p}(y) = (1-p)p^y, \quad y \geq 0$	$\frac{p}{1-p}$	$\frac{1-p}{1-pz}$
(Geometric) $\mathsf{Geom}_{1,p}$	$\mathsf{Geom}_{1,p}(y) = (1-p)p^{y-1}, \quad y \geq 1$	$\frac{1}{1-p}$	$\frac{z(1-p)}{1-pz}$
(Negative binomial) $\mathsf{NB}_{r,p}$	$\mathsf{NB}_{r,p}(y) = \binom{y+r-1}{y}(1-p)^r p^y, \quad y \geq 0$	$\frac{rp}{1-p}$	$\left(\frac{1-p}{1-pz}\right)^r$
(Poisson) Poi_λ	$\mathsf{Poi}_\lambda(y) = e^{-\lambda}\frac{\lambda^y}{y!}, \quad y \geq 0$	λ	$e^{\lambda(z-1)}$

Table 2.1: Properties of some distributions over the integers. PMF stands for "Probability Mass Function", PGF stands for "Probability Generating Function", and we assume $y \in \mathbb{Z}$.

The statistical distance Δ is a metric on the space of distributions above. In particular, it satisfies the triangle inequality. Its definition also implies that for any (deterministic or randomised) algorithm A we have $\Delta(\mathsf{A}(X); \mathsf{A}(Y)) \leq \Delta(X; Y)$.

At certain points we will need to lower bound the expectation of a function of a random variable. This is especially approachable if the function is convex via Jensen's inequality. We present a special case here (it holds in significantly more general settings, e.g., [63]).

Lemma 2.1 (Jensen's inequality)**.** *Let $\mathcal{I} \subseteq \mathbb{R}$ be an interval and $f : \mathcal{I} \to \mathbb{R}$ a convex function on $\mathcal{I}$.*[2] *Suppose X satisfies $\mathsf{supp}(X) \subseteq \mathcal{I}$ and $\mathbb{E}[|X|] < \infty$. Then, we have*

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X]).$$

We now present concentration bounds for some of the distributions described in Table 2.1, starting with the Hoeffding bound.

Lemma 2.2 (Hoeffding bound [64, Theorem 2.2.6])**.** *Suppose that $Z = \sum_{i=1}^n X_i$ for $X_i \in [0, 1]$ independent random variables. Letting $\mu = \mathbb{E}[Z]$, for every $\alpha \geq 0$ it holds that*

$$\Pr[Z \geq \mu + \alpha] \leq \exp\left(-\frac{2\alpha^2}{n}\right)$$

and

$$\Pr[Z \leq \mu - \alpha] \leq \exp\left(-\frac{2\alpha^2}{n}\right).$$

[2] A function $f : \mathcal{I} \to \mathbb{R}$ is *convex on* $\mathcal{I} \subseteq \mathbb{R}$ if for any $x, y \in \mathcal{I}$ and $\lambda \in [0, 1]$ we have $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$. A function f is said to be *concave* if $-f$ is convex.

We will need concentration bounds for the negative binomial distribution. We follow the reasoning of Brown [65], who proved a similar bound.

Lemma 2.3. *Suppose $Z \sim \mathsf{NB}_{r,p}$ for $r \in \mathbb{N}$ and $p \in [0,1)$. If $\mu = \mathbb{E}[Z] = \frac{rp}{1-p}$, it holds that*

$$\Pr[Z > \lceil \mu + \alpha \rceil] \le \exp\left(-\frac{2\alpha^2(1-p)^3}{r + (\alpha+1)(1-p)}\right)$$

and

$$\Pr[Z < \lfloor \mu - \alpha \rfloor] \le \exp\left(-\frac{2\alpha^2(1-p)^2}{r}\right)$$

for all $\alpha \ge 0$.

Proof. Letting $W = r + Z$ and $\mu' = \mathbb{E}[W] = \frac{r}{1-p}$, it is enough to prove that

$$\Pr[W > \lceil \mu' + \alpha \rceil] \le \exp\left(-\frac{2\alpha^2(1-p)^3}{r + (\alpha+1)(1-p)}\right) \tag{2.1}$$

and

$$\Pr[W < \lfloor \mu' - \alpha \rfloor] \le \exp\left(-\frac{2\alpha^2(1-p)^2}{r}\right) \tag{2.2}$$

for all $\alpha \ge 0$. We begin by proving (2.1). Since $W > n$ is equivalent to having fewer than r successes in n independent Ber_{1-p} trials, we have

$$\Pr[W > \lceil \mu' + \alpha \rceil] = \Pr[\mathsf{Bin}_{\lceil \mu' + \alpha \rceil, 1-p} < r].$$

Setting $V \sim \mathsf{Bin}_{\lceil \mu' + \alpha \rceil, 1-p}$, we have $\mathbb{E}[V] \ge (1-p)(\mu' + \alpha) = r + \alpha(1-p)$. Therefore, by the Hoeffding bound it follows that

$$\Pr[V < r] \le \Pr[V < \mathbb{E}[V] - \alpha(1-p)] \le \exp\left(-\frac{2\alpha^2(1-p)^2}{\lceil \mu' + \alpha \rceil}\right) \le \exp\left(-\frac{2\alpha^2(1-p)^3}{r + (\alpha+1)(1-p)}\right)$$

for all $\alpha \ge 0$. To prove (2.2), we assume without loss of generality that $\alpha \le \mu = \frac{rp}{1-p}$ and observe that

$$\Pr[W \le \lfloor \mu' - \alpha \rfloor] = 1 - \Pr[W > \lfloor \mu' - \alpha \rfloor] = 1 - \Pr[\mathsf{Bin}_{\lfloor \mu' - \alpha \rfloor, 1-p} < r] = \Pr[\mathsf{Bin}_{\lfloor \mu' - \alpha \rfloor, 1-p} \ge r].$$

Setting $V' \sim \mathsf{Bin}_{\lfloor \mu' - \alpha \rfloor, 1-p}$, we have $\mathbb{E}[V'] \le r - \alpha(1-p)$. Therefore, the Hoeffding bound implies that

$$\Pr[V' \ge r] \le \Pr[V' \ge \mathbb{E}[V'] + \alpha(1-p)] \le \exp\left(-\frac{2\alpha^2(1-p)^2}{\lfloor \mu' - \alpha \rfloor}\right) \le \exp\left(-\frac{2\alpha^2(1-p)^2}{r}\right). \qquad \square$$

Next, we present a concentration bound for the Poisson distribution [66].

Lemma 2.4 ([66, Theorem 1]). *Let $Z \sim \mathsf{Poi}_\lambda$. Then, for every $\alpha > 0$ it holds that*

$$\Pr[|Z - \lambda| \geq \alpha] \leq 2 \cdot \exp\left(-\frac{\alpha^2}{2(\lambda + \alpha)}\right).$$

Moving in another direction, we discuss the concept of *almost k-wise independence*, which is central to the fields of pseudorandomness and derandomisation.

Definition 2.2 (ε-almost k-wise independent random variable). *A random variable X over $\{0,1\}^m$ is said to be* ε-almost k-wise independent *if for all sets of k distinct indices $i_1, i_2, \dots, i_k \in [m]$ we have*

$$|\Pr[X_{i_1} = x_1, \dots, X_{i_k} = x_k] - 2^{-k}| \leq \varepsilon$$

for all $(x_1, \dots, x_k) \in \{0,1\}^k$.

Alon, Goldreich, Håstad, and Peralta [67] gave various elegant and efficient ways of generating ε-almost k-wise independent random variables from few uniformly random bits. We present a particularly clean version of their results that is appropriate for our derandomisation applications in Chapter 5, following their exposition. The starting point is a version of Vazirani's XOR lemma, stating that small bias against linear tests suffices for almost k-wise independence. For a detailed exposition of different versions of the XOR lemma, including the one below, see [68].

Lemma 2.5 ([67, Corollary 1 and Appendix]). *If $X \in \{0,1\}^m$ is δ-biased with respect to linear tests*[3]*, then X is δ-almost k-wise independent for every $k \leq m$.*

Although Lemma 2.5 is stated for random variables uniformly distributed over subsets of $\{0,1\}^m$ in [67, Corollary 1], it applies to arbitrary distributions [68, Section 1.5]. We obtain the desired generator by combining Lemma 2.5 with the following construction from [67].

Lemma 2.6 ([67, Proposition 3]). *For every m and ℓ, there is a function $g : \{0,1\}^{2\ell} \to \{0,1\}^m$ computable in time $m \cdot \mathrm{poly}(\ell)$ such that $g(U_{2\ell})$ is δ-biased with respect to linear tests for $\delta = \frac{m-1}{2^\ell}$, where $U_{2\ell}$ is uniformly distributed over $\{0,1\}^{2\ell}$.*

Combining Lemmas 2.5 and 2.6 leads to the following corollary.

[3] A random variable X over $\{0,1\}^m$ is *δ-biased with respect to linear tests* if for every nonempty $\mathcal{S} \subseteq [m]$ it holds that $\left|\Pr\left[\bigoplus_{i \in \mathcal{S}} X_i = 0\right] - \Pr\left[\bigoplus_{i \in \mathcal{S}} X_i = 1\right]\right| \leq \delta$.

Corollary 2.1. *For every m, $\varepsilon > 0$ there is a function $g : \{0,1\}^t \to \{0,1\}^m$ with $t = 2\lceil \log(1/\varepsilon) + \log m \rceil$ computable in time $m \cdot \mathrm{poly}(t)$ such that $g(U_t)$ is ε-almost k-wise independent for every $k \leq m$.*

The construction from [67] used to prove Lemma 2.6 assumes knowledge of an irreducible polynomial of degree ℓ over the finite field of order two, $\mathbb{F}_2$. Shoup [69] gave a deterministic algorithm running in time $\mathrm{poly}(t)$ for finding irreducible degree-ℓ polynomials over $\mathbb{F}_2$. Therefore, even with this preprocessing step, computing g still takes overall time $m \cdot \mathrm{poly}(\ell)$. As noted in [67, Section 8], such a polynomial could also be naively preprocessed in time $2^{O(\ell)}$ by iterating over all monic polynomials of that degree and discarding those with nontrivial divisors. Since we will choose $\varepsilon = 1/\mathrm{poly}(m)$ in our applications, this naive procedure suffices for our needs.

2.2 Special functions

In this section, we introduce some well-known special functions that will be useful in Chapters 3 and 4. A detailed treatment of special functions and their properties can be found in the book of Abramowitz and Stegun [70].

The first function we discuss is the *gamma function* Γ, defined as

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt$$

for all real numbers $z > 0$. Outside this region, the gamma function is defined by analytic continuation to all real z except the non-positive integers, where it is not defined. Notably, the gamma function satisfies $\Gamma(1) = 1$ and the recurrence relation $\Gamma(1+z) = z\Gamma(z)$. As a result, for every $n \in \mathbb{N}_0$ it holds that $\Gamma(1+n) = n!$.

In view of the properties of the gamma function above, for real numbers x and y we define the binomial coefficient $\binom{x}{y}$ in terms of the gamma function as

$$\binom{x}{y} = \frac{\Gamma(1+x)}{\Gamma(1+x-y)\Gamma(1+y)}$$

whenever the gamma function is well-defined (this being exactly when its argument is not a non-positive integer). When $y \in \mathbb{N}_0$, the expression above simplifies to

$$\binom{x}{y} = \frac{\prod_{i=0}^{y-1}(x-i)}{y!},$$

which can be extended to all $x \in \mathbb{R}$. In particular, we have $\binom{x}{0} = 1$ for all $x \in \mathbb{R}$.

Another special function of interest related to the gamma function is the *log gamma* function $\ln \Gamma$. Similarly to the gamma function, the log gamma function also has an integral representation which will prove useful in Chapter 3.

Lemma 2.7 ([40, Expression (100)]). *We have*

$$\ln \Gamma(1+z) = \int_0^1 \frac{1 - tz - (1-t)^z}{t \ln(1-t)} dt$$

for all $z > -1$.

Moreover, a sharp asymptotic expansion is known for the log gamma function, as detailed in the following lemma.

Lemma 2.8 ([70, Sections 6.1.41 and 6.1.42]). *We have*

$$\ln \Gamma(z) = \left(z - \frac{1}{2}\right) \ln z - z + \frac{1}{2} \ln(2\pi) + r(z)$$

for all $z > 0$, where $r(z)$ satisfies $0 \leq r(z) \leq \frac{1}{12z}$.

The derivative of the log gamma function, called the *digamma* function, is also a well studied special function, and it will make an important appearance in Chapter 4. The digamma function ψ is defined as

$$\psi(z) = \frac{\Gamma'(z)}{\Gamma(z)},$$

where Γ' denotes the first derivative of the gamma function, for all real z except non-positive integers. Notably, when the argument z is a positive integer, the digamma function takes on a simple form in terms of the harmonic numbers,

$$\psi(z) = -\gamma + \sum_{i=1}^{z-1} \frac{1}{i},$$

where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant and $\sum_{i=1}^{z-1} \frac{1}{i}$ is the $(z-1)$-th harmonic number.

Another special function whose properties will be of interest in Chapter 4 is the *exponential integral function* E_1, defined as

$$E_1(z) = \int_1^\infty \frac{e^{-tz}}{t} dt$$

for $z > 0$. This function has two properties that will prove useful to us. First, the derivative of $E_1(z)$ is $E_1'(z) = -\frac{e^{-z}}{z}$. Second, the exponential integral function enjoys sharp bounds in terms of elementary functions, as made precise in the following lemma.

Lemma 2.9 ([70, Section 5.1.20], [71, Theorem 2]). *For every $z > 0$ it holds that*

$$\max\left(\frac{1}{2}\ln(1+2/z), -e^z \ln\left(1 - e^{-ze^\gamma}\right)\right) < e^z E_1(z) < \ln(1+1/z),$$

where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant.

To conclude, we define the *logarithmic integral* $\mathrm{li}(z)$, one more well known special function that will make an appearance in Chapter 3, which is given by

$$\mathrm{li}(z) = \int_0^z \frac{1}{\ln t} dt$$

for all real numbers $z \in [0, 1)$.

We note that the gamma and logarithmic integral functions accept complex arguments. However, in this thesis we will only consider real-valued arguments for such special functions.

2.3 Channels

Before we proceed, we need to define what we mean by a *channel.* In the most general setting, a channel with input alphabet $\mathcal{X}$ and output alphabet $\mathcal{Y}$ receives as input $x \in \mathcal{X}^*$ and outputs a random variable Y_x supported in $\mathcal{Y}^*$. Here, we are interested in the class of *memoryless* channels. Roughly speaking, such channels receive an ordered sequence of symbols as input, and do not keep state when processing the different symbols. This section is a close adaptation of material from [6].

The simplest family of memoryless channels are the *Discrete Memoryless Channels* (DMCs), which were first studied by Shannon [24].

Definition 2.3 (Discrete memoryless channel)**.** *A channel* Ch *is said to be a* discrete memoryless channel *with discrete input alphabet* $\mathcal{X}$ *and discrete output alphabet* $\mathcal{Y}$ *if it acts on the input* $x \in \mathcal{X}^*$ *as follows: If* $x \in \mathcal{X}$*, then* Ch *outputs a discrete random variable* Y_x *supported in* $\mathcal{Y}$*. If* $x = (x_1, \ldots, x_n) \in \mathcal{X}^n$*, then* Ch *outputs* Y_x *satisfying*

$$Y_x = Y_{x_1} \| Y_{x_2} \| \cdots \| Y_{x_n}$$

supported in $\mathcal{Y}^n$*, where* $Y_{x_1}, Y_{x_2}, \ldots, Y_{x_n}$ *are all independent.*

The fact that DMCs map single input symbols to single output symbols means they satisfy the useful property that, for $x \in \mathcal{X}^n$, the product decomposition

$$Y_x(y) = \prod_{i=1}^{n} Y_{x_i}(y_i) \tag{2.3}$$

holds, where $Y_{x_i}(y_i)$ denotes the probability that the channel outputs y_i on input x_i. As we shall see, this implies that in order to study a DMC it is enough to study its behaviour on a single input symbol.

Two well-studied examples of DMCs are the Binary Symmetric Channel (BSC) and the Binary Erasure Channel (BEC), the latter of which we have already discussed in Chapter 1. The BSC receives bits as input, and independently flips each bit with some error probability d. According to Definition 2.3, we can define the channel BSC_d as the DMC with input and output alphabets $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ and Y_b for $b \in \{0, 1\}$ satisfying

$$Y_b(y) = \begin{cases} 1 - d, & \text{if } y = b, \\ d, & \text{if } y = 1 - b. \end{cases}$$

The BEC receives bits as input, and independently erases each bit with probability d. We can define the channel BEC_d as the DMC with input alphabet $\mathcal{X} = \{0, 1\}$ and output alphabet $\mathcal{Y} = \{0, 1, ?\}$ and Y_b for $b \in \{0, 1\}$ satisfying

$$Y_b(y) = \begin{cases} 1 - d, & \text{if } y = b, \\ d, & \text{if } y = ?, \\ 0, & \text{otherwise.} \end{cases}$$

In this thesis, we are interested in studying channels which are not DMCs, but rather belong to the more general class of what we call *Discrete Memoryless Synchronisation Channels* (DMSCs). Nevertheless, we shall see that our analysis of DMSCs will naturally require us to study some associated DMCs. The general definition of DMSCs we present here is due to Dobrushin [72].

Definition 2.4 (Discrete memoryless synchronisation channel)**.** *A channel* Ch *is said to be a* discrete memoryless synchronisation channel *with discrete input alphabet* $\mathcal{X}$ *and discrete output alphabet* $\mathcal{Y}$ *if it acts as follows: If* $x \in \mathcal{X}$*, then* Ch *outputs a random variable* Y_x *supported in* $\mathcal{Y}^*$*. If* $x = (x_1, \dots, x_n) \in \mathcal{X}^n$*, then* Ch *outputs* Y_x *satisfying*

$$Y_x = Y_{x_1} \| Y_{x_2} \| \cdots \| Y_{x_n}$$

supported in $\mathcal{Y}^*$*, where* $Y_{x_1}, Y_{x_2}, \dots, Y_{x_n}$ *are all independent.*

Arguably the most well-known DMSC is the deletion channel, which independently deletes input bits with deletion probability d. According to Definition 2.4, we can define this deletion channel as the DMSC denoted by BDC_d with input and output alphabets $\mathcal{X} = \mathcal{Y} = \{0,1\}$ and Y_b for $b \in \{0,1\}$ satisfying

$$Y_b(y) = \begin{cases} 1-d, & \text{if } y = b, \\ d, & \text{if } y = \varepsilon, \\ 0, & \text{otherwise,} \end{cases}$$

recalling that ε denotes the empty string. We are interested in a natural generalisation of the deletion channel, which we call *repeat channels.* A deletion channel independently replaces each input bit with either 0 or 1 copies in the output. In general, a repeat channel independently replaces each input bit x_i with R_i copies, where the R_i's are i.i.d. according to some replication rule R. Thus, a repeat channel with replication rule R is a DMSC with input and output alphabets $\mathcal{X} = \mathcal{Y} = \{0,1\}$ and Y_b for $b \in \{0,1\}$ satisfying

$$Y_b(y) = \begin{cases} R(r), & \text{if } y = b^r, \\ 0, & \text{otherwise.} \end{cases}$$

In particular, the deletion channel with deletion probability d is a repeat channel with replication rule R satisfying $R(0) = d$ and $R(1) = 1 - d$. As already discussed in Chapter 1, these channels are natural models of errors caused by physical processes corrupting data in modern data storage systems. Throughout this thesis, we focus on repeat channels induced by replication rules R with finite expected value, and assume that this holds always.

The definitions of DMCs and DMSCs above are similar, the only difference being that for DMSCs a single input symbol may lead to the output of a variable number of output symbols. As already discussed informally in Chapter 1, this leads to a loss of synchronisation between the sender and receiver, since the receiver is not sure which parts of the output correspond to each input symbol. Notably, this means that the product decomposition described in (2.3) does not apply in general to DMSCs, although it applies to all DMCs. For this reason, unlike the case for DMCs, we cannot hope to characterise the properties of a DMSC simply by looking at its behaviour on a single input symbol, which makes their analysis significantly more complicated.

2.4 Reliable information transmission

Suppose that we are allowed to send n symbols over a certain DMC or DMSC (in other words, we are allowed n *uses* of the channel). In this section, we are interested in the question of how many different messages we can reliably transmit through the channel as a function of the n uses of the channel, when n goes to infinity. This section is a close adaptation of material from [6].

We wish to transmit an arbitrary message m belonging to a message set $\mathcal{M}$ through n uses of a memoryless channel Ch. First, we encode m into some $x \in \mathcal{X}^n$, which is then transmitted through Ch. The receiver observes Y_x and wishes to recover x, and hence the message m, with small error probability. We formalise the notion of a coding scheme below.

Definition 2.5 (Coding scheme)**.** *A pair of functions* $\mathsf{Enc} : \mathcal{M} \to \mathcal{X}^n$ *and* $\mathsf{Dec} : \mathcal{Y}^n \to \mathcal{M}$ *such that* Enc *is deterministic and injective and* Dec *is deterministic is said to be an* (n, R, λ)-coding scheme for Ch *if* $|\mathcal{M}| = \lceil 2^{Rn} \rceil$ *and for every* $m \in \mathcal{M}$ *and* $x = \mathsf{Enc}(m)$ *it holds that*

$$\Pr[\mathsf{Dec}(Y_x) = m] \geq 1 - \lambda.$$

Interchangeably, we may instead work with the associated code $\mathcal{C} = \mathsf{Enc}(\mathcal{M}) \subseteq \mathcal{X}^n$*, where* n *is called the* blocklength of $\mathcal{C}$*, and its encoding and decoding procedures* $(\mathsf{Enc}, \mathsf{Dec})$*. We say that* R *is the* rate *of the coding scheme* $(\mathsf{Enc}, \mathsf{Dec})$*, or of the code* $\mathcal{C}$*, and* λ *is the* decoding error probability*. Moreover, if* $\mathsf{Enc} : \mathcal{X}^\ell \to \mathcal{X}^n$*, we define the* redundancy of $\mathcal{C}$ *in bits as* $(n - \ell) \log |\mathcal{X}|$*.*

Recall that we are aiming for *reliable* information transmission. We define this as being able to transmit messages through the channel with decoding error probability approaching 0 when the number of channel uses n grows. More precisely, we have the following definition.

Definition 2.6 (Achievable rate). *A non-negative real number R is said to be an* achievable rate for the channel Ch *if there exists a family of (n, R_n, λ_n)-coding schemes for* Ch *such that* $\lim_{n\to\infty} R_n \geq R$ *and* $\lim_{n\to\infty} \lambda_n = 0$.

We are now ready to define the capacity of a channel, which is the optimal transmission rate we can achieve with vanishing decoding error probability as the number of channel uses grows.

Definition 2.7 (Channel capacity). *The* capacity of the channel Ch*, denoted by* $\mathsf{Cap}(\mathsf{Ch})$*, is defined as*

$$\mathsf{Cap}(\mathsf{Ch}) = \sup\{R \geq 0 : R \textit{ is an achievable rate for } \mathsf{Ch}\}.$$

Shannon's *noisy channel coding theorem* [24] is a fundamental early result in information theory which characterises the capacity of a DMC in terms of the *mutual information* between the channel input and output. Before we can state this result, we need to define some information-theoretic concepts.

Definition 2.8 (Entropy). *The* (Shannon) entropy *of a discrete random variable X, denoted by $H(X)$, is defined as*

$$H(X) = -\mathbb{E}_{x\sim X}[\log X(x)] = -\sum_{x\in\mathsf{supp}(X)} X(x)\log X(x).$$

Moreover, for discrete random variables X and Y and event E, we define the conditional entropy of X given Y and E*, denoted by $H(X|Y,E)$, as*

$$H(X|Y,E) = \sum_{y\in\mathsf{supp}(Y|E)} Y_E(y)H(X|Y=y,E),$$

where Y_E denotes Y conditioned on event E.

Definition 2.9 (Kullback-Leibler divergence). *The* Kullback-Leibler divergence *between discrete random variables X and X', denoted by $D_{\mathsf{KL}}(X\|X')$, is defined as*

$$D_{\mathsf{KL}}(X\|X') = \sum_{x\in\mathsf{supp}(X)} X(x)\log\left(\frac{X(x)}{X'(x)}\right)$$

provided $X(x) = 0$ whenever $X'(x) = 0$ (i.e., X is absolutely continuous with respect to X'). If this does not hold, we set $D_{\mathsf{KL}}(X\|X') = \infty$.

In full generality, we define the mutual information between X and Y as the Kullback-Leibler divergence between their joint and product distributions.

Definition 2.10 (Mutual information)**.** *The* mutual information between discrete random variables X and Y, *denoted by $I(X;Y)$, is defined as*

$$\begin{aligned} I(X;Y) &= D_{\mathsf{KL}}(XY \| X \otimes Y) \\ &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X), \end{aligned}$$

where $(XY)(x,y) = \Pr[X = x, Y = y]$ and $(X \otimes Y)(x,y) = \Pr[X = x] \cdot \Pr[Y = y]$, and the last two equalities hold if $H(X)$ and $H(Y)$ are finite, respectively. For discrete random variables X, Y, and Z, the conditional mutual information between X and Y given Z, *denoted by $I(X;Y|Z)$, is defined as*

$$I(X;Y|Z) = \sum_{z \in \mathsf{supp}(Z)} Z(z) \cdot I(X;Y|Z = z),$$

where $I(X;Y|Z = z) = I((X|Z = z);(Y|Z = z))$.

The mutual information enjoys several useful properties via its connection to the Kullback-Leibler divergence. For example, we always have $I(X;Y) \geq 0$ with equality if and only if X and Y are independent. Furthermore, it satisfies the chain rule $I(X;Y,Z) = I(X;Z) + I(X;Y|Z)$, and, if Y is conditionally independent of X given Z, the data processing inequality $I(X;Y) \leq I(Z;Y)$.

We are now ready to state the noisy channel coding theorem.

Theorem 2.1 ([24])**.** *If Ch is a DMC with input alphabet $\mathcal{X}$, then it holds that*

$$\mathsf{Cap}(\mathsf{Ch}) = \lim_{n \to \infty} \frac{1}{n} \sup_{X^{(n)}} I(X^{(n)}; Y_{X^{(n)}}) = \sup_{X} I(X; Y_X), \tag{2.4}$$

where in the middle term the supremum is taken over all distributions $X^{(n)}$ supported in $\mathcal{X}^n$, and in the right-hand term the supremum is taken over all distributions X supported in $\mathcal{X}$. Moreover, $Y_{X^{(n)}}$ and Y_X denote the outputs of Ch on input $X^{(n)}$ and X, respectively.

The rightmost equality in (2.4) simplifies the computation of the channel capacity considerably, and is due to the product decomposition formula in (2.3) which holds for DMCs: It suffices to consider $X^{(n)} = (X_1^{(n)}, X_2^{(n)}, \ldots, X_n^{(n)})$ where the $X_i^{(n)}$ are i.i.d. according to some X over $\mathcal{X}$. If X achieves the supremum on the right-hand side, we call it *capacity-achieving.*

Using Theorem 2.1, Shannon [24] showed that

$$\mathsf{Cap}(\mathsf{BSC}_d) = 1 - h(d),$$

where $h(d) = -d \log d - (1-d)\log(1-d)$ is the *binary entropy function*, and

$$\mathsf{Cap}(\mathsf{BEC}_d) = 1 - d.$$

In both cases, the capacity-achieving distribution is uniform over $\{0,1\}$.

Shannon's noisy channel coding theorem did not apply to DMSCs. The analogous result for DMSCs satisfying a mild assumption was proved later by Dobrushin [72], with subsequent extensions in [73, 74, 75, 76, 77].

Theorem 2.2 ([72]). *If* Ch *is a DMSC with finite input alphabet* $\mathcal{X}$ *and there exist constants* $c_2 > c_1 > 0$ *such that* $c_1 < \mathbb{E}[|Y_x|] < c_2$ *for all* $x \in \mathcal{X}$*, then it holds that*

$$\mathsf{Cap}(\mathsf{Ch}) = \lim_{n\to\infty} \frac{1}{n} \sup_{X^{(n)}} I(X^{(n)}; Y_{X^{(n)}}),$$

where the supremum is taken over all distributions $X^{(n)}$ *supported in* $\mathcal{X}^n$ *and* $Y_{X^{(n)}}$ *denotes the output of* Ch *on input* $X^{(n)}$.

In contrast with Theorem 2.1, there is no clear way of simplifying the computation of the capacity of a DMSC in Theorem 2.2. This is because (2.3) does not hold for DMSCs in general, and in particular it does not hold for any non-trivial repeat channel. As a result, Theorem 2.2 does not seem to provide a useful way of determining the capacity of a DMSC, although it can be used to obtain *bounds* on the capacity of DMSCs. For example, in view of Theorem 2.2, we can define the *achievable rate* of a specific family of input distributions $(X^{(n)})_{n\in\mathbb{N}}$ as

$$\liminf_{n\to\infty} \frac{1}{n} I(X^{(n)}; Y_{X^{(n)}}).$$

Analysing this quantity for well-chosen input distributions yields lower bounds on $\mathsf{Cap}(\mathsf{Ch})$. Moreover, as we shall see in Section 2.5.1.1, Theorem 2.2 can be used to obtain a computationally intractable procedure for deriving capacity upper bounds for DMSCs.

Given the fact that a uniform input distribution achieves the capacity of the BSC and BEC, which are similar to the deletion channel, one could hope that an i.i.d. uniform input $X^{(n)}$ would yield a good lower bound on the capacity of the deletion channel, and maybe even other repeat channels. However, this is only the case for the deletion channel when the deletion probability d is small [78, 79]. As we shall see, input distributions with memory achieve much better rates for repeat channels. This is a telling example of the difficulty of studying the capacity of even simple DMSCs compared to DMCs.

2.5 Historical background

In this section, we first discuss previous work on obtaining improved bounds on the capacity of DMSCs. Such capacity bounds show the existence (in the case of lower bounds) or the inexistence (in the case of upper bounds) of coding schemes with a certain rate for such DMSCs, without regard for whether these schemes support efficient encoding and decoding procedures. Afterwards, we discuss parallel progress made on the design of efficient coding schemes in several relevant models featuring deletions and replications, and on trace reconstruction under synchronisation errors.

2.5.1 Capacity bounds for synchronisation channels

This section is a close adaptation of material from [6]. The study of the capacity of DMSCs was ignited by Gallager [80], who showed, using convolutional coding techniques, that the capacity of the deletion channel with deletion probability d, which we denote by $C(d)$, satisfies

$$C(d) \geq 1 - h(d) \tag{2.5}$$

for all $d \leq 1/2$. This result was also independently obtained by Zigangirov [81]. Therefore, curiously, it is possible to transmit at higher rates through a deletion channel than through a BSC. The lower bound in (2.5) can also be proved by studying the rate achievable by codes with i.i.d. uniform codewords under an explicit sub-optimal decoder [82]. Kalai, Mitzenmacher, and Sudan [78] and Kanoria and Montanari [79] concurrently showed that this lower bound is tight in the asymptotic regime $d \to 0$. They proved that

$$C(d) = 1 - h(d) + O(d) = 1 - d \log(1/d) + O(d).$$

It is known that the lower bound in (2.5) is loose whenever d is not small. Therefore, one natural way of improving on (2.5) is to study rates achievable by codes whose codewords are i.i.d. according to some distribution X with memory, in particular $X = (X_1, X_2, \dots, X_n)$ where the X_i are generated according to some Markov chain. Vvedenskaya and Dobrushin [83] were the first to consider this setting with low-order Markov chains, and presented some numerical estimates of the achievable rates for the deletion channel. More recently, Diggavi and Grossglauser [82] obtained the improved lower bound

$$C(d) \geq \frac{1}{\ln 2} \cdot \sup_{\gamma>0, p\in(0,1)} [-(1-d)\ln((1-q)A + qB) - \gamma], \tag{2.6}$$

where $q = 1 - \frac{1-p}{1+d(1-2p)}$, $A = \frac{(1-p)e^{-\gamma}}{1-pe^{-\gamma}}$, and $B = \frac{(1-p)^2 e^{-2\gamma}}{1-pe^{-\gamma}} + pe^{-\gamma}$, which is derived by studying the rate achieved by codes generated by order-1 Markov chains over $\{0,1\}$ with transition probability $1-p$ from 0 to 1 and vice-versa under an explicit decoder. The lower bound in (2.6) was subsequently improved and the reasoning extended to more general repeat channels by Drinea and Mitzenmacher [84, 29] via an explicit improved decoding procedure they termed *jigsaw decoding* for codes generated by X with i.i.d. runs according to some distribution P with a geometrically decreasing tail (if P is geometric, then this coincides with the distributions considered by Diggavi and Grossglauser). Remarkably, such lower bounds on the capacity of the *Poisson-repeat* channel, a repeat channel with replication rule $R \sim \mathsf{Poi}_\lambda$, were used by Mitzenmacher and Drinea [41] to show that

$$C(d) \geq 0.1185(1-d) > \frac{1-d}{9}$$

for all $d \in [0,1]$. This lower bound is still the state-of-the-art for the deletion channel when d is close to 1, and it shows that communicating through the deletion channel is never much worse than communicating through the BEC (recall that the capacity of BEC_d is $1-d$, which is also a trivial upper bound on $C(d)$). Subsequently, a series of works has focused on deriving improved lower bounds on the capacity of the deletion channel and related repeat channels using inputs generated by low-order Markov chains or with i.i.d. runs [85, 31, 1, 86, 87, 39].

In the first part of Chapter 3, we study *sticky* channels, which are a subclass of repeat channels with replication rule R satisfying $R(0) = 0$, meaning that no input bit is ever deleted. Two well-studied examples of sticky channels are the *elementary duplication channel*, which duplicates each input bit with some probability p and hence has replication rule satisfying $R(1) = 1-p$ and $R(2) = p$, and the *geometric sticky channel*, which replicates each input bit according to a geometric distribution with success probability p, meaning that the replication rule R satisfies

$$R(r) = (1-p)p^{r-1}, \quad r = 1, 2, \ldots.$$

We shall focus on the geometric sticky channel, and we call p the *replication parameter*. Lower bounds for such channels (among others) were first studied by Drinea, Kirsch, and Mitzenmacher [29, 30, 31]. Subsequently, Mitzenmacher [32] studied these channels directly and obtained improved numerical lower bounds which are close to the true capacity of these channels for all values of the replication parameter p. Because of their structure, sticky channels are amenable to relatively simple lower bound techniques that do not work for channels with deletions. Moreover, Mercier, Tarokh, and Labeau [1] showed that inputs generated by low-order Markov chains already achieve rates close to the true capacity of the geometric sticky channel, and both Drinea and Mitzenmacher [29] and Iyengar, Siegel, and Wolf [39] derived analytical expressions for the achievable rates of such input distributions as a function of the replication parameter p, for Markov chains of arbitrary order. Iyengar, Siegel, and Wolf [39] then used these analytical lower bounds to show that the capacity of the geometric sticky channel with replication parameter p, which we denote by $\mathsf{Cap}(\mathsf{Geom}_{1,p})$, satisfies

$$\mathsf{Cap}(\mathsf{Geom}_{1,p}) \geq 1 - p\log(1/p) + cp - O(p^2) \tag{2.7}$$

for p small enough, where $c \approx 0.8458$ is an explicit constant, achieved by an i.i.d. uniform input distribution. Ramezani and Ardakani [88] showed that the capacity of the elementary duplication channel with replication parameter p, which we denote by $\mathsf{Cap}(D_p)$, satisfies

$$\mathsf{Cap}(D_p) = 1 - p\log(1/p) + cp + O(p^{3/2-\varepsilon})$$

for any constant $\varepsilon > 0$ as $p \to 0$, where, as in (2.7), we also have $c \approx 0.8458$. Remarkably, both of these results are out of reach of purely numerical methods.

True upper bounds on the capacity of repeat channels appeared much later than the first capacity lower bounds. The first capacity upper bounds were obtained for the deletion channel by Diggavi, Mitzenmacher, and Pfister [89]. Subsequently, other works obtained improved capacity upper bounds for the deletion channel [85, 3, 40]. Regarding sticky channels, Mitzenmacher [32] obtained tight capacity upper bounds for the elementary duplication channel. However, he was unable to achieve the same for the geometric sticky channel, mostly due to the fact that the replication rule for this channel has unbounded support. Later, Mercier, Tarokh, and Labeau [1] were able to obtain tight capacity upper bounds for the geometric sticky channel. Overall, the works mentioned above follow similar high-level approaches. They obtain numerical capacity upper bounds that require significant computer assistance in order to be computed for a given deletion probability d or replication parameter p, and cannot provide an exact characterisation of the capacity.

The only work that does not fit into this group is that of Cheraghchi [40], which focused on deriving *analytical* capacity upper bounds for the deletion and Poisson-repeat channels, given by the supremum over $(0, 1)$ of an analytic function which can be easily approximated to the desired accuracy. The goal of this approach is to enable a better conceptual, and eventually exact, understanding of the capacity of DMSCs with the help of such upper bounds. Thus, not only is one interested in obtaining easier-to-compute and potentially improved capacity upper bounds, but also in deriving results about the capacity curve without computer assistance. For example, using an analytical upper bound, Cheraghchi [40] was able to improve on the previous best capacity upper bounds on $C(d)$ for $d \leq 0.02$ and also to give closed-form capacity upper bounds on $C(d)$, including a proof without computer assistance that $C(1/2) \leq \frac{\log \varphi}{2}$, where $\varphi \approx 1.618$ is the golden ratio.

We discuss frameworks for obtaining the upper bounds mentioned above in more detail in Section 2.5.1.1.

In Chapter 3, we also study capacity upper bounds for a channel closely related to the geometric sticky channel, which we call the *geometric deletion channel.* This is a repeat channel that replicates bits according to a geometric distribution *with support starting at* 0. In other words, the geometric deletion channel with replication parameter p has replication rule $R \sim \mathsf{Geom}_{0,p}$ satisfying

$$R(r) = (1-p)p^r, \quad r = 0, 1, \dots.$$

Therefore, a geometric deletion channel combines geometric replications with deletions. Comparatively to other channels already discussed in this section, much less is known about this channel. The only known general upper bounds on the capacity of the geometric deletion channel with replication parameter p, which we denote by $\mathsf{Cap}(\mathsf{Geom}_{0,p})$, are obtained by observing that

$$\mathsf{Cap}(\mathsf{Geom}_{0,p}) \leq C(1-p)$$

and

$$\mathsf{Cap}(\mathsf{Geom}_{0,p}) \leq \mathsf{Cap}(\mathsf{Geom}_{1,p}),$$

as we show in Chapter 3. We may call $d = 1 - p$ the deletion probability of the geometric deletion channel.

For the special case $p = 1/2$, the geometric deletion channel corresponds to the *binary deletion-duplication channel* studied by Mercier, Tarokh, and Labeau [1] and Iyengar, Siegel, and Wolf [39] with $p_d = 1 - p_t$, for which non-trivial numerical capacity upper bounds are known when $p_d = p_t$ [1], as well as estimates of the rate achievable by low-order Markov chains [39] (in particular, these two regimes coincide when $p_d = p_t = 1/2$). This is a DMSC with input alphabet $\{0, 1\}$ parameterised by $p_d, p_t \in [0, 1]$ which on input a bit x_i behaves as follows:

1. With probability p_d, delete x_i and stop;

2. With probability $1 - p_d - p_t$, append x_i to the output and stop;

3. With probability p_t, append one copy of x_i to the output and return to Step 1.

We may see the channel above as a repeat channel with replication rule satisfying

$$R(r) = \begin{cases} p_d, & \text{if } r = 0, \\ p_d \cdot p_t^r + (1 - p_d - p_t)p_t^{r-1}, & \text{if } r \geq 1. \end{cases}$$

Therefore, if $p = p_t = 1 - p_d$, the binary deletion-duplication channel from [1, 39] corresponds exactly to the geometric deletion channel with replication parameter p.

2.5.1.1 Frameworks for capacity upper bounds

As one of the main goals in this thesis is the derivation of improved capacity upper bounds for repeat channels, we discuss here prior frameworks in more detail, including the general framework of Cheraghchi [40] that will form the basis for our approach in Chapters 3 and 4. The material in this section is based on [2, 6].

The most direct way of obtaining capacity upper bounds for repeat channels is by exploiting Theorem 2.2 jointly with special properties of the sequence

$$\left(\sup_{X^{(n)}} \frac{1}{n} I(X^{(n)}; Y_{X^{(n)}})\right)_{n\in\mathbb{N}},$$

where the supremum is taken over all input distributions $X^{(n)}$ over $\mathcal{X}^n$. Setting

$$\mathsf{fCap}_n(\mathsf{Ch}) = \sup_{X^{(n)}} \frac{1}{n} I(X^{(n)}; Y_{X^{(n)}}),$$

it can be shown that

$$(n+m)\mathsf{fCap}_{n+m}(\mathsf{Ch}) \leq n \cdot \mathsf{fCap}_n(\mathsf{Ch}) + m \cdot \mathsf{fCap}_m(\mathsf{Ch})$$

for all n and m. This means that the sequence $(n \cdot \mathsf{fCap}_n)$ is *subadditive*, and so Fekete's lemma [90] ensures that in this case we have

$$\mathsf{Cap}(\mathsf{Ch}) = \lim_{n\to\infty} \mathsf{fCap}_n(\mathsf{Ch}) = \inf_{n\geq 1} \mathsf{fCap}_n(\mathsf{Ch})$$

for every DMSC Ch satisfying the hypotheses of Theorem 2.2. In particular, we have $\mathsf{Cap}(\mathsf{Ch}) \leq \mathsf{fCap}_n(\mathsf{Ch})$ for all $n \geq 1$, and it was also shown by Dobrushin [72] that

$$\mathsf{fCap}_n(\mathsf{Ch}) - \frac{\log(n+1)}{n} \leq \mathsf{Cap}(\mathsf{Ch}) \leq \mathsf{fCap}_n(\mathsf{Ch}) \tag{2.8}$$

for all such DMSCs, which is tight [74]. If the replication rule R has finite support, computing $\mathsf{fCap}_n(\mathsf{Ch})$ for a fixed n is a finite-dimensional convex optimisation problem which can be solved numerically using well-known procedures such as the Blahut-Arimoto algorithm [91, 92]. Combining this with (2.8), we have a direct strategy for bounding $\mathsf{Cap}(\mathsf{Ch})$. However, there are two main drawbacks: First, numerically computing $\mathsf{fCap}_n(\mathsf{Ch})$ is computationally intractable whenever n is not small. Second, even if we succeed in computing $\mathsf{fCap}_n(\mathsf{Ch})$ for larger n, this approach does not improve our conceptual understanding of the capacity curve of the DMSC. Fertonani and Duman [85] evaluated $\mathsf{fCap}_n(\mathsf{BDC}_d)$ for $n \leq 17$ with the help of the Blahut-Arimoto algorithm, and obtained improved capacity upper bounds for the deletion channel when $d \in [0.1, 0.8]$. The current best capacity upper bounds for all $d \geq 0.1$ are obtained by combining these bounds with a "convexification" result of Rahmati and Duman [3], which in particular states that $C(d) \leq \frac{(1-d)C(d')}{1-d'}$ for any $d \geq d'$. This yields $C(d) \leq 0.4143(1-d)$ for all $d \geq d' = 0.65$ by considering the best numerical upper bound on $C(0.65)$ from [85]. Remarkably, this generalises an approach of Dalai [28] showing that $\lim_{d\to 1}\frac{C(d)}{1-d} = \inf_{d\in(0,1)}\frac{C(d)}{1-d} \leq 0.4143$.

In Chapter 3, we are mainly interested in deriving *analytical* capacity upper bounds for repeat channels as a step towards a deeper, computer-unaided study of their capacity. This means the direct approach detailed above is not viable for us. Instead, the basis for our results, and in fact for most previously known capacity upper bounds for repeat channels, is the following high-level strategy:

1. Reduce upper bounding the capacity of the repeat channel to upper bounding the capacity *per unit cost* of an associated DMC. This is done by carefully modifying the output of the original repeat channel so that it reveals more information about the input (which only increases capacity);

2. Derive upper bounds on the capacity per unit cost of the associated DMC.

The capacity per unit cost of a channel is a generalisation of the channel capacity to account for potential costs associated with transmitting a given symbol through the channel. We present a definition below. For a more detailed discussion on the capacity per unit cost, see [38].

Definition 2.11 (Capacity per unit cost)**.** *Given a DMC* Ch *with input and output alphabets* $\mathcal{X}$ *and* $\mathcal{Y}$*, respectively, its* capacity per unit cost with cost function $c : \mathcal{X} \to \mathbb{R}^+$, *denoted by* $\overline{\mathsf{Cap}}_c(\mathsf{Ch})$*, is given by*

$$\overline{\mathsf{Cap}}_c(\mathsf{Ch}) = \sup_{X:\mathbb{E}[c(X)]<\infty} \frac{I(X;Y_X)}{\mathbb{E}[c(X)]},$$

where the supremum is over all possible input distributions X *supported in* $\mathcal{X}$ *such that* $\mathbb{E}[c(X)] < \infty$*, and* Y_X *denotes the associated channel output.*

If $\mathcal{X} \subseteq \mathbb{R}$ *and* $c(x) = x$ *for all* $x \in \mathcal{X}$*, we simply write* $\overline{\mathsf{Cap}}(\mathsf{Ch})$ *for the capacity per unit cost of* Ch.

Remark 2.1. *Note that the original definition of channel capacity corresponds to the capacity per unit cost with cost function* $c(x) = 1$ *for all* $x \in \mathcal{X}$*, i.e., all symbols cost the same to be transmitted.*

For the purpose of deriving capacity upper bounds, it is useful to represent the input to a repeat channel by its *runlength encoding.* More precisely, suppose our input string for the repeat channel is

$$x = 0^{\ell_1}1^{\ell_2}0^{\ell_3}\ldots,$$

where the different 0^{ℓ_i} and 1^{ℓ_j} are called *runs* of x. Then, the runlength encoding of x is

$$(0, \ell_1, \ell_2, \ell_3, \dots).$$

For the particular application of studying the capacity of repeat channels, we may without loss of generality assume that every input string x starts with a 0. This does not affect the capacity of the channel, and allows us to use the simpler runlength encoding

$$(\ell_1, \ell_2, \ell_3, \dots)$$

for x, with the understanding that odd numbered runs correspond to 0s and even numbered runs correspond to 1s. For example, the runlength encoding of the bitstring 001000110 is $(2, 1, 3, 2, 1)$.

The behaviour of repeat channels under runlength encoding is easy to describe. Each input run of length $\ell \geq 1$ is independently mapped to an output run of length

$$R^{(\ell)} = \sum_{i=1}^{\ell} R_i,$$

where the R_i are i.i.d. according to the replication rule R of the repeat channel. Output runs of length zero are simply omitted from the output.

Using the runlength encoding perspective allows us to see that Step 1 in the high-level approach above does not incur any loss for sticky channels. Suppose Ch is a sticky channel with replication rule R, and let Ch_R denote the DMC with input alphabet $\mathcal{X} = \{1, 2, \dots\}$ such that $Y_\ell \sim R^{(\ell)}$ for all $\ell \geq 1$. It was noted by Mitzenmacher [32] that the capacity of the sticky channel, which we denote by $\mathsf{Cap}(R)$, equals the capacity per unit cost of Ch_R with cost function $c(\ell) = \ell$: Under the runlength encoding, the sticky channel Ch behaves exactly like the DMC Ch_R, with the only difference being that sending ℓ through Ch_R corresponds to sending a run of ℓ bits through Ch, and hence ℓ uses of Ch. More formally, we have the following theorem.

Theorem 2.3 ([32, Theorem 2.1])**.** *For any replication rule R satisfying $R(0) = 0$ it holds that*

$$\mathsf{Cap}(R) = \overline{\mathsf{Cap}}(\mathsf{Ch}_R).$$

Given Theorem 2.3, all that is left to do is to upper bound $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$. This is accomplished in [32, 1] by relying on the following result of Abdel-Ghaffar [93].

Theorem 2.4 ([93])**.** *Consider a DMC* Ch *with input and output alphabets* $\mathcal{X}$ *and* $\mathcal{Y}$*, respectively. Let* Y *be any distribution over* $\mathcal{Y}$*. Then,*

$$\overline{\mathsf{Cap}}_c(\mathsf{Ch}) \leq \sup_{x \in \mathcal{X}} \frac{D_{\mathsf{KL}}(Y_x \| Y)}{c(x)}.$$

Moreover, an input distribution X *is capacity-achieving if and only if there is* $\lambda \in \mathbb{R}$ *such that* $\frac{D_{\mathsf{KL}}(Y_x \| Y_X)}{c(x)} \leq \lambda$ *for all* $x \in \mathcal{X}$ *with equality for* $x \in \mathsf{supp}(X)$*, in which case we have* $\overline{\mathsf{Cap}}_c(\mathsf{Ch}) = \lambda$*.*

Analytically designing good candidate distributions Y to be used in Theorem 2.4, and hence obtaining good analytical capacity upper bounds via this approach, turns out to be complex even for simple cost functions like $c(x) = x$. Instead, previous works, including those on sticky channels [32, 1], use a numerical approach for designing Y. We describe the approach from [1] for the geometric sticky channel in more detail, which is similar to previous approaches for other repeat channels [89, 32]. The first step is to obtain from the DMC Ch_R a modified DMC Ch'_R with *finite* input and output alphabets by truncating the input and output alphabets of Ch_R appropriately to $i_{\max}$ and $o_{\max}$, thus lowering the capacity. The fact that Ch'_R now has finite input and output alphabets allows one to use a Blahut-Arimoto-type algorithm for the capacity per unit cost, such as the Jimbo-Kunisawa algorithm [94], to numerically obtain a strict lower bound C on $\overline{\mathsf{Cap}}(\mathsf{Ch}'_R)$, and hence on $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$, along with an associated input distribution X with achievable rate C. If $i_{\max}$ and $o_{\max}$ are large enough, then we expect X to be also a nearly-optimal input distribution for Ch_R. However, plugging the corresponding output distribution Y_X of X over Ch_R into Theorem 2.4 leads to an unmanageable infinite optimisation problem. In order to avoid this, Mercier, Tarokh, and Labeau replace this infinite optimisation problem by a finite one that can be solved numerically. This is accomplished in two steps: First, one considers another version of Ch_R with larger capacity, which we call Ch''_R, with genie-aided decoding. More precisely, an input x to Ch''_R is revealed to the receiver (using a special symbol to distinguish it from a normal output of the channel) if either $x > i_{\max}$ or x is mapped to some $y > o_{\max}$. Otherwise, the channel Ch''_R behaves exactly like Ch_R. Second, one adds a suitable geometrically decaying tail to X based on the lower bound C given by the Jimbo-Kunisawa algorithm and renormalises the distribution, leading to a new input distribution X'' with output distribution Y'' over Ch''_R. This is done to ensure that, if Y''_x denotes the output distribution of Ch''_R on input x, we have

$$\frac{D_{\mathsf{KL}}(Y''_x \| Y'')}{x} = C < \overline{\mathsf{Cap}}(\mathsf{Ch}_R) < \overline{\mathsf{Cap}}(\mathsf{Ch}''_R).$$

for all $x > i_{\max}$. Given the above and invoking Theorem 2.4 with Y'' means that in order to derive an upper bound on $\overline{\mathsf{Cap}}(\mathsf{Ch}''_R)$, and hence on $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$, it is now enough to compute $\frac{D_{\mathsf{KL}}(Y''_x\|Y'')}{x}$ for all $x \leq i_{\max}$. Due to genie-aided decoding and the fact that the channel is sticky, the terms $Y''(y)$ are given by finite sums and the support of Y''_x is also finite, allowing one to compute $D_{\mathsf{KL}}(Y''_x\|Y'')$ exactly.

Although the approach described in the previous paragraph yields tight capacity upper bounds for the geometric sticky channel, one cannot avoid heavy numerical computations for each fixing of the replication parameter p, which, as discussed before, severely limits our conceptual understanding of the channel behaviour. Moreover, the use of the truncated channel Ch'_R and the modified channel Ch''_R with extra information at the receiver, required for the numerical methods to work in [1], immediately preclude an exact characterisation of $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$ via Theorem 2.4 for any replication parameter p. As a step towards a (potentially exact) analysis of DMSCs without computer assistance, Cheraghchi [40] considered a different general framework for deriving good analytical upper bounds on $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$, which may be seen as a "mean-constrained" variant of Theorem 2.4. Before we proceed, we must define the concept of the mean-limited capacity of a DMC Ch.

Definition 2.12 (Mean-limited capacity). *Given a DMC Ch with input and output alphabets $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}$ and a parameter $\mu > 0$, we define the* μ-limited capacity of Ch, *denoted by $\mathsf{Cap}_\mu(\mathsf{Ch})$, as*

$$\mathsf{Cap}_\mu(\mathsf{Ch}) = \sup_{X:\mathbb{E}[Y_X]=\mu} I(X;Y_X),$$

where the supremum is taken over all input distributions X supported in $\mathcal{X}$.

Cheraghchi [40] proved the following theorem by casting the maximisation problem of $\mathsf{Cap}_\mu(\mathsf{Ch})$ as a convex program and applying the Karush-Kuhn-Tucker conditions to the dual program. This result can be seen as a special case of a more general framework handling a broad class of channels with discrete and continuous input alphabets which we discuss in Appendix B.3.

Theorem 2.5 ([40, Theorem 1], adapted). *Fix a replication rule R, its associated DMC Ch_R, and $\mu > \mathbb{E}[R]$. If there exist constants $a \in \mathbb{R}_0^+$ and $b \in \mathbb{R}$ and a distribution Y over $\mathbb{N}_0$ such that*

$$D_{\mathsf{KL}}(Y_x\|Y) \leq a\mathbb{E}[Y_x] + b$$

for all $x \in \mathbb{N}$, then $\mathsf{Cap}_\mu(\mathsf{Ch}_R) \leq a\mu + b$. Moreover, an input distribution X is capacity-achieving if *and only if* $\mathbb{E}[Y_X] = \mu$ *and there exist constants* $a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$ *such that*

$$D_{\mathsf{KL}}(Y_x \| Y_X) \leq a\mathbb{E}[Y_x] + b$$

for all $x \in \mathbb{N}$, *with equality when* $x \in \mathsf{supp}(X)$. *In this case, we have* $\mathsf{Cap}_\mu(\mathsf{Ch}_R) = a\mu + b$.

The relevance of Theorem 2.5 to sticky channels comes from the fact that, taking into account Theorem 2.3, if we define $\lambda = \mathbb{E}[R]$, then we can write

$$\begin{aligned}\overline{\mathsf{Cap}}(\mathsf{Ch}_R) &= \sup_{L:\mathbb{E}[L]<\infty} \frac{I(L;Y_L)}{\mathbb{E}[L]} \\ &= \sup_{\mu'\geq 1} \sup_{L:\mathbb{E}[L]=\mu'} \frac{I(L;Y_L)}{\mu'} \\ &= \lambda \sup_{\mu\geq\lambda} \sup_{L:\mathbb{E}[Y_L]=\mu} \frac{I(L;Y_L)}{\mu} \\ &= \lambda \sup_{\mu\geq\lambda} \frac{\mathsf{Cap}_\mu(\mathsf{Ch}_R)}{\mu}.\end{aligned}$$

The second equality holds because $\mathbb{E}[L] \geq 1$, since L is supported in $\{1, 2, \dots\}$. The third equality follows from the fact that $\mu = \mathbb{E}[Y_L] = \mathbb{E}[R] \cdot \mathbb{E}[L] = \lambda \cdot \mu'$ under the constraint that $\mathbb{E}[L] = \mu'$, and the fact that we take the supremum over $\mu' \geq 1$, which is equivalent to $\mu/\lambda \geq 1$. The fourth equality follows directly from the definition of μ-limited capacity. Therefore, we have the following result.

Theorem 2.6. *For any replication rule satisfying* $R(0) = 0$ *and* $\lambda = \mathbb{E}[R]$ *it holds that*

$$\mathsf{Cap}(R) = \lambda \sup_{\mu\geq\lambda} \frac{\mathsf{Cap}_\mu(\mathsf{Ch}_R)}{\mu}.$$

Note that both Theorem 2.4 and Theorem 2.5 present conditions to verify the *optimality* of the candidate distribution Y. In the case of Theorem 2.5, if Y is the output distribution of the DMC Ch_R on input some distribution X and $D_{\mathsf{KL}}(Y_x\|Y) \leq a\mathbb{E}[Y_x] + b$ with equality for all $x \in \mathsf{supp}(X)$, then we recover the exact capacity of $\mathsf{Cap}_\mu(\mathsf{Ch}_R)$ for $\mu = \mathbb{E}[Y]$. In general, we call the quantity

$$\Delta(x) = a\mathbb{E}[Y_x] + b - D_{\mathsf{KL}}(Y_x\|Y)$$

the *KL-gap* of Y with respect to the line $a\mathbb{E}[Y_x] + b$ (the line with respect to which we compute the KL-gap will always be clear from context, so we refrain from explicitly adding it to the notation). We may then rewrite one of the optimality conditions above as requiring that $\Delta(x) \geq 0$ for all x with equality for all $x \in \mathsf{supp}(X)$. This is an important quantity because, from experience, it appears that candidate distributions with smaller KL-gaps lead to better capacity upper bounds [40, 2].

As we shall see in Chapter 3, we are able to analytically design for the first time candidate distributions Y satisfying $\Delta(x) = 0$ for all x, i.e., distributions with *zero KL-gap everywhere.* This means that these distributions satisfy one of the optimality conditions of Theorem 2.5, analogous to the optimality conditions of Theorem 2.4. Remarkably, these distributions lead to great analytical capacity upper bounds on $\mathsf{Cap}_\mu(\mathsf{Ch}_R)$, and hence tight upper bounds on the capacity $\mathsf{Cap}(R)$ of the sticky channel via Theorem 2.6, even improving on the previous numerical capacity upper bounds discussed above for some parameters. Therefore, further study of these explicit distributions is a viable approach towards a sharp analysis of the corresponding capacity without computer assistance (we discuss some concrete next steps in Chapter 6).

Although Theorem 2.3 does not apply to general repeat channels, one can still use the frameworks described above to derive capacity upper bounds for such channels. Perhaps the most basic way of doing this is by modifying the repeat channel into an associated sticky-like channel with larger capacity, which can be accomplished by marking deleted runs, and upper bounding the capacity of the latter. This was the approach originally undertaken by Diggavi, Mitzenmacher, and Pfister [89] to derive numerical capacity upper bounds for the deletion channel. From the perspective of the runlength encoding, the deletion channel BDC_d independently maps runs of length $\ell \geq 1$ into runs of binomial length $R^{(\ell)} \sim \mathsf{Bin}_{\ell,1-d}$, outputting the empty string ε when $R^{(\ell)} = 0$. They consider the modified deletion channel which behaves exactly like the deletion channel above, but does not omit deleted runs, i.e., it always outputs $R^{(\ell)}$ even when $R^{(\ell)} = 0$. Figure 2.1 illustrates the differences between the two channels.

The capacity of the modified deletion channel is at least as large as that of the deletion channel, since one can remove the 0s from the runlength encoding of the output of the modified deletion channel and sum adjacent runlengths with the same bit value to obtain the output of the true deletion channel. Moreover, as was the case for sticky channels, the capacity of the modified deletion channel equals $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$, where, as before, Ch_R is the DMC with input alphabet $\mathbb{N}$, output alphabet $\mathbb{N}_0$, and output $Y_\ell \sim R^{(\ell)}$ on input $\ell \in \{1, 2, \dots\}$. In fact, this holds for any repeat channel, and we obtain the general inequality

$$\mathsf{Cap}(R) \leq \overline{\mathsf{Cap}}(\mathsf{Ch}_R)$$

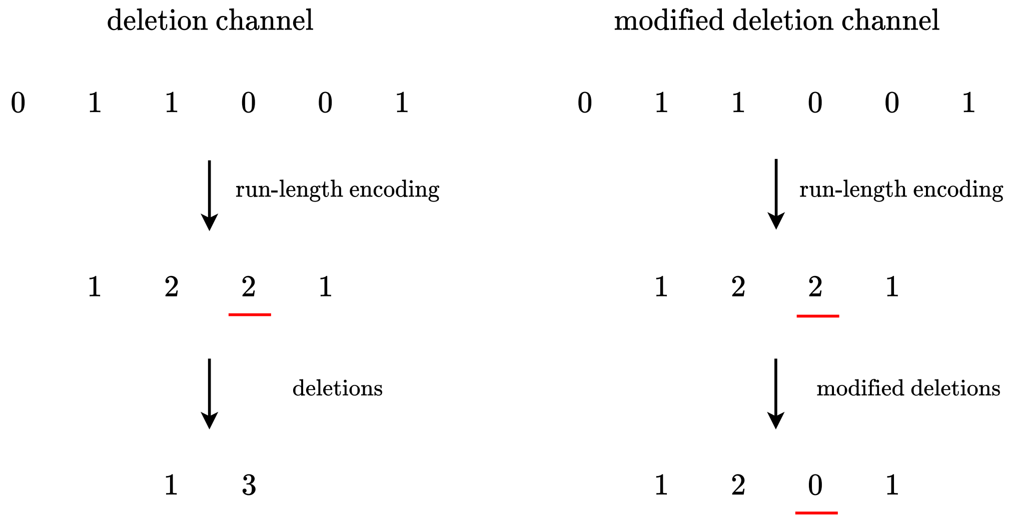

Figure 2.1: Comparison between the deletion channel and the modified deletion channel with marked deleted runs. The underlined run of two 0s is deleted.

for all replication rules R. One could make use of either Theorem 2.4 or Theorem 2.5 to derive upper bounds on $\overline{\mathsf{Cap}}(\mathsf{Ch}_R)$ with the help of the Jimbo-Kunisawa algorithm, similarly to what was described for sticky channels in [32, 1].

The approach above is already good enough to derive non-trivial capacity upper bounds for the deletion channel, and in the meantime other approaches have surfaced which lead to even better capacity upper bounds [85, 3, 40]. In order to analyse the capacity of the geometric deletion channel, we make use of the following tighter connection between the capacity of a repeat channel with replication rule R and the μ-limited capacity of another associated DMC Ch'_R. We then connect Ch'_R to Ch_R and exploit Theorem 2.5 with carefully designed candidate distributions Y to derive good analytical capacity upper bounds on the geometric deletion channel. Notably, we obtain a proof without computer assistance that the capacity of the geometric deletion channel is at most 0.73 bits/channel use when p is close enough to 1.

Theorem 2.7 ([40, Theorem 4]). *Fix a distribution R over $\mathbb{N}_0$, and let $\overline{R}$ denote R conditioned on the event $R \neq 0$. Let Ch'_R denote the channel which on input $1+\ell$ for $\ell \in \{0,1,\dots\}$ outputs $\overline{R}+\sum_{i=1}^{\ell} R_i$, where $\overline{R}$ and the R_i are independent and furthermore the R_i are i.i.d. according to R. Then,*

$$\mathsf{Cap}(R) \leq \sup_{\mu \geq \overline{\lambda}} \frac{\mathsf{Cap}_\mu(\mathsf{Ch}'_R)}{1/\alpha + (\mu - \overline{\lambda})/\lambda}, \tag{2.9}$$

where $\lambda = \mathbb{E}[R]$, $\overline{\lambda} = \mathbb{E}[\overline{R}]$, and $\alpha = 1 - R(0)$.

In particular, we can recover the upper bound in Theorem 2.6 via Theorem 2.7 by noting that when $R(0) = 0$ we have $\overline{R} \sim R$, $\overline{\lambda} = \lambda$, and $\alpha = 1$. We discuss the reduction used to prove Theorem 2.7, following [40, Section 4]. The key to deriving the relevant upper bound is to interpret the behaviour of the repeat channel as a two-stage process. Intuitively, the first stage applies only certain deletions caused by the repeat channel. Its outcome is an intermediate string Z with the special property that the first bit of each run is not deleted by the repeat channel. Under this conditioning, the repeat channel behaves like the DMC Ch'_R on each run of Z, which makes up the second stage. Let X denote the input n-bit string, and $X^{(1)}, X^{(2)}, \dots, X^{(m)}$ denote its runs. We may assume that the repeat channel does not delete the first input bit $X_1^{(1)}$, as this does not change its capacity. Consider the following two stages:

1. First stage: We begin by sending $X^{(2)}$ through a channel which iteratively and independently deletes each bit of $X^{(2)}$ with probability $d = R(0)$. If $X_i^{(2)}$ is the first bit not deleted by the channel, we let $\overline{X}^{(2)} = X^{(2)}[i:]$. Provided $X^{(2)}$ is completely deleted, we send next $X^{(4)}$ (the run with the same bit value) through the channel above and set $\overline{X}^{(3)} = X^{(3)}$. Otherwise, we send $X^{(3)}$ through the channel (the run with the opposite bit value). More generally, if the i-th run $X^{(i)}$ was last sent through the channel above, then we send $X^{(i+2)}$ next and set $\overline{X^{(i+1)}} = X^{(i+1)}$ if $X^{(i)}$ was completely deleted, and send $X^{(i+1)}$ otherwise, provided they exist. After this process is completed, we obtain a string $Z = X^{(1)} \| \overline{X}^{(2)} \| \cdots \| \overline{X}^{(m)}$ with runs $Z^{(1)}, Z^{(2)}, \dots, Z^{(M)}$, where M is a random variable, and corresponding runlength encoding $(L_1, L_2, \dots, L_M)$.

   Crucially, after this stage we may now condition the behaviour of the repeat channel on the event that the first bit of each run $Z^{(i)}$ is not deleted, meaning that the repeat channel now behaves like the DMC Ch'_R on the runlength encoding of Z.

2. In the second stage, we send $(L_1, L_2, \dots, L_M)$ through the DMC Ch'_R. This leads to output $(L'_1, L'_2, \dots, L'_M)$, which is distributed exactly like the runlength encoding of the output Y of the repeat channel on input X.

To derive (2.9), one can employ Theorem 2.2 and, via the reduction above, observe that $I(X;Y) \leq I(Z;Y)$, the latter quantity being easier to upper bound (up to $o(n)$ terms) when $n \to \infty$.

2.5.2 Efficient coding against synchronisation errors

In the previous section, we discussed past work on deriving better bounds on the capacity of DMSCs. Although capacity lower bounds show the existence of a coding scheme with vanishing decoding error probability for a DMSC with a given rate, these results do not give coding schemes with efficient encoding and decoding procedures. To complement this, a parallel line of work has focused on the design of coding schemes with polynomial-time (in the blocklength of the code) encoding and decoding procedures for various communication models with deletions, insertions, and replications, including certain DMSCs. First, we make precise what we mean by an efficiently encodable and decodable family of codes.

Definition 2.13 (Efficiently encodable and decodable code)**.** *We say a family of codes $\mathcal{C}_n \subseteq \mathcal{X}^n$ is* efficiently encodable *if there is a constant $c > 0$ such that for every n the code $\mathcal{C}_n$ has an associated encoding procedure Enc_n such that $\mathsf{Enc}_n(\cdot)$ can be computed in time $O(n^c)$. Moreover, we say the family of codes is* efficiently decodable *if there is a constant $c > 0$ such that every associated decoding procedure $\mathsf{Dec}_n(\cdot)$ can be computed in time $O(n^c)$. If a family of codes is both efficiently encodable and decodable, we say that the family of codes is* efficient*.*

Throughout this thesis, for the sake of convenience, we may ignore the parameterisation in the blocklength n, and simply write $\mathcal{C}$ to implicitly refer to all codes in the family $(\mathcal{C}_n)_{n\in\mathbb{N}}$. The parameterisation will be clear from context.

The study of efficient codes for correcting synchronisation errors was initiated by Sellers [95], who studied some marker-based constructions, and Levenshtein [96], who showed that the Varshamov-Tenengolts (VT) code [97] is able to correct one worst-case deletion or insertion of a bit. Put differently, the binary VT code $\mathcal{C} \subseteq \{0,1\}^n$ with associated encoding/decoding procedures $(\mathsf{Enc}, \mathsf{Dec})$ has the property that if $c \in \mathcal{C}$ with $c = \mathsf{Enc}(m)$ and c' is obtained from c by either deleting one bit of c or inserting an arbitrary bit at an arbitrary position, then it holds that $\mathsf{Dec}(c') = m$. This notion can be generalised to multiple worst-case synchronisation errors. We begin with an auxiliary definition.

Definition 2.14 (Longest common subsequence distance)**.** *Given two strings $x, y \in \mathcal{S}^*$, the* Longest Common Subsequence (LCS) distance between *x* and *y, denoted by $\mathsf{dLCS}(x, y)$, is the minimum number of insertions and deletions required to transform x into y.*

The LCS distance between x and y can be written in terms of the *longest common subsequence* between

x and y, which we denote by $\mathsf{LCS}(x,y)$. In general, we have

$$\mathsf{dLCS}(x,y)=(|x|-|\mathsf{LCS}(x,y)|)+(|y|-|\mathsf{LCS}(x,y)|)=|x|+|y|-2|\mathsf{LCS}(x,y)|.$$

We are now ready to define d-insdel correcting codes, which can correct a combination of up to d worst-case deletions and insertions.

Definition 2.15 (Insdel correcting code). *We say a code $\mathcal{C}\subseteq\mathcal{X}^n$ with encoding/decoding procedures* $(\mathsf{Enc},\mathsf{Dec})$ *is a d-*insdel correcting code *if for every $c\in\mathcal{C}$ with $c=\mathsf{Enc}(m)$ for some m and c' such that $\mathsf{dLCS}(c,c')\leq d$, we have $\mathsf{Dec}(c')=m$.*

By Definition 2.15, VT codes are efficient 1-insdel correcting codes with about $\log n$ bits of redundancy, where n denotes the blocklength of the code, which is nearly optimal [96]. Surprisingly, the generalisation of Levenshtein's result to a larger number of worst-case deletions and insertions with similarly low redundancy remained elusive for several decades. Brakensiek, Guruswami, and Zbarsky [98] were the first to construct efficient binary d-insdel correcting codes for $d\geq 2$ with redundancy $o(n)$. Their codes have redundancy $O(d^2\log d\cdot\log n)$, while it is known that the redundancy must be $\Omega(d\log n)$. This result was subsequently improved for $d=2$ by Gabrys and Sala [99] and Sima, Raviv, and Bruck [100], achieving redundancy $8\log n+O(\log\log n)$ and $7\log n+o(\log n)$, respectively, and for arbitrary constant d by Sima and Bruck [101], achieving redundancy $8d\log n+o(\log n)$, with efficient decoding from worst-case deletions only.[4] Recently, Guruswami and Håstad [102] gave improved efficiently encodable codes with redundancy approximately $4\log n$ when $d=2$, again with efficient decoding from worst-case deletions.

For the general case where d is an arbitrary function of the blocklength n, with particular interest given to the setting where $d=cn$ for some constant $c>0$, there are also some known results. Schulman and Zuckerman [103] constructed efficient binary d-insdel correcting codes with positive rate when $d\leq cn$ for a small enough constant $c>0$. This was later improved by Guruswami and Wang [104], with an efficient decoder for worst-case deletions only. They constructed efficient codes with rate $1-O(\sqrt{c}\cdot\log c)$ correcting $d=cn$ worst-case deletions for a small enough constant $c>0$. In particular, the rate of these codes approaches 1 as the fraction of errors c approaches 0. This result was subsequently extended to the setting with d worst-case deletions *and insertions* by Guruswami and Li [105]. Nearly-optimal

[4] A code whose codewords can be recovered from up to d worst-case deletions is also d-insdel correcting. However, an efficient decoder for up to d worst-case deletions does not immediately translate to an efficient decoder from any combination of up to d worst-case deletions *and insertions*.

efficient binary d-insdel-correcting codes for general d were obtained concurrently by Cheng, Jin, Li, and Wu [19] and Haeupler [20] via connections to deterministic document exchange protocols. We state one of their results below.

Theorem 2.8 ([20, Theorem II.3], adapted)**.** *For every m and $d < m$ there exists an efficient binary* systematic[5] *d-insdel correcting code with message size m and blocklength $n = m + r$ with redundancy $r = \Theta\left(d \log^2(m/d) + d\right)$.*

A related fundamental question is determining the largest constant $c^\star > 0$ such that there exist positive rate binary $(d = c^\star n)$-insdel correcting codes with blocklength n. First, we must have $c^\star < 1/2$, since otherwise we can delete either all 0s or all 1s of a given codeword. On the opposite end, the code of Schulman and Zuckerman [103] implies that $c^\star > 0$. This lower bound was later improved by Kash, Mitzenmacher, Thaler, and Ullman [106] and Bukh, Guruswami, and Håstad [107]. The latter work showed that $\sqrt{2} - 1 < c^\star < 1/2$, and gave an efficient binary code with positive rate that can correct a fraction of worst-case deletions arbitrarily close to $\sqrt{2} - 1$.

Although the focus of this thesis related to this topic lies in efficient *binary* codes, we remark that several interesting results regarding efficient d-insdel correcting codes are also known over larger (both constant and non-constant) alphabet sizes. For example, Guruswami and Wang [104] showed that by increasing the alphabet size to $\mathrm{poly}(1/c)$ it is possible to construct efficient codes correcting $d = (1-c)n$ worst-case deletions with rate $\mathrm{poly}(c)$, and Guruswami and Li [105] extended this result to worst-case deletions and insertions. The state-of-the-art results in the large-alphabet setting were obtained via the introduction of *synchronisation strings* by Haeupler and Shahrasbi [108], along with a series of subsequent works [109, 110, 111, 112] (see also the survey [113]). Besides the results covered here, there is a large amount of literature on several types of efficient codes appropriate for different practically-motivated models with synchronisation errors. The survey of Mercier, Bhargava, and Tarokh [37] contains a detailed account of this line of work.

In Chapter 5, we will be interested in the problem of designing efficient codes that can correct a constant rate of i.i.d. *random* deletions with high probability. Making a bridge between worst-case and random deletions, Guruswami and Li [114] studied the existence of (not necessarily efficient) codes with positive rate correcting large fractions of *oblivious* and *online* deletions. In the model of oblivious deletions, an n-bit codeword is corrupted by dn deletions for some constant $d > 0$, and the positions of

[5] A code $\mathcal{C}$ is said to be *systematic* if its encoding procedure Enc satisfies $\mathsf{Enc}(s) = s \| \mathsf{Enc}'(s)$ for some function Enc'.

the deletions are oblivious to the codeword itself. Online deletions correspond to the setting where an adversary is allowed to decide whether the i-th bit c_i of the codeword c is deleted (up to a budget of dn deletions) based only on the values of $c_1, c_2, \dots, c_i$. With respect to random deletions, Guruswami and Li [115] complemented the result of Mitzenmacher and Drinea [41], which states that $C(d) \geq \frac{1-d}{9}$ for all $d \in [0,1]$, by constructing efficient binary codes for the BDC_d with rate $\frac{1-d}{120}$ and vanishing decoding error probability. This was subsequently improved by Con and Shpilka [116], who designed efficient binary codes for the BDC_d with rate $\frac{1-d}{16}$. Moreover, they also designed the first efficient binary codes for the Poisson-repeat channel, which, as we have seen, is closely connected to the deletion channel.

On a related note, polar codes have been constructed for the deletion channel with constant deletion probability under a Markov input source [117], which have been shown to achieve the capacity of the deletion channel [77]. Furthermore, some works have proposed practical codes for DMSCs whose performance is evaluated empirically (see [37, Section III.J] for a detailed account).

2.5.3 Coding with multiple traces and trace reconstruction

In the previous sections, we have focused almost entirely on settings where a (usually coded) string x is sent through a channel, and the receiver must recover x with high probability from the channel output. On the other hand, recent developments in DNA-based data storage with nanopore-based sequencing [15, 16] provide practical motivation for the rigorous study of an extension of the setting above where the receiver has access to $t \geq 1$ outputs of the channel on the same input x. The process for reading data in such storage systems is illustrated in Figure 1.4, and the goal is to obtain a good tradeoff between the rate of the coding scheme used to encode the data in the system and the number of traces required to reconstruct the input with high probability (with higher rate and fewer traces being more desirable). Some coding schemes were proposed in [15, 16]. However, they are designed for fixed parameters only, and their decoding procedures and the corresponding analysis are heavily based on heuristics and experimental evidence, meaning that these coding schemes have no provable guarantees even in simplified error models (e.g., where the DNA sequencing process corrupts each read with i.i.d. deletions). The decoding procedure from [15] is based on multiple sequence alignment algorithms, which are notoriously difficult to analyse rigorously, and the decoding procedure from [16] uses a heuristic variant of a well-known *trace reconstruction* algorithm (which we shall discuss below) that is only guaranteed to work when the error rate vanishes as the input blocklength increases.

This situation naturally leads us to consider the problem of designing efficient, high-rate coding schemes with provable guarantees regarding reliable decoding with access to multiple, but relatively few, channel outputs corrupted by a constant rate of i.i.d. synchronisation errors, which we introduce and study in Chapter 5. Compared with the single-channel output setting discussed in Sections 2.5.1 and 2.5.2, much less is known about coding schemes in the multi-trace setting. However, several related problems have been studied, and we survey them below. Some results will be discussed in detail, as they will be useful in Chapter 5. This section is an expansion of material from [5].

The problem above fits into the more general framework of what we may call *multi-trace problems.* In a general t-trace model, an input string x is transformed into t traces

$$Y^{(1)}, Y^{(2)}, \ldots, Y^{(t)}$$

according to some combinatorial or probabilistic rule. Several realisations of this general setting have been studied in the literature. For example, the traces may be obtained by sending x through a DMSC, in which case the $Y^{(i)}$ are i.i.d. according to the channel output distribution Y_x. Alternatively, one may take a combinatorial viewpoint, and assume that the t traces are obtained by corrupting x with t different worst-case error patterns (with a bounded number of errors). Additionally, we may distinguish the cases where the number of available traces t is fixed and where t may vary, say with the length of x. These settings were first studied by Levenshtein [118, 119], who was mostly interested in the problem of determining the minimum number of traces $t = t(n)$ required to reconstruct the length n input string x either exactly (in the combinatorial setting), or with a given error probability (when x is sent through a memoryless channel).

In the combinatorial setting, Levenshtein considered the case where the traces $(Y^{(i)})_{i\in[t]}$ are obtained by corrupting x with t arbitrary, but different, patterns of at most d errors, including substitutions, deletions, and insertions. For example, in the case of substitutions it was shown in [118] that every $x \in \{0,1\}^n$ can be recovered exactly from any set of $t(n)$ different error patterns for

$$t(n) = 1 + 2\sum_{i=0}^{d-1} \binom{n-1}{i},$$

while in the case of deletions it was shown that every $x \in \{0,1\}^n$ can be recovered exactly from

$$t(n) = 1 + 2\sum_{i=0}^{d-1} \binom{n-d-1}{i}$$

traces. These results were also generalised for strings over any finite alphabet, and simple reconstruction algorithms using the optimal number of traces were described in [119] for deletions and insertions of arbitrary symbols. In the case of worst-case insertions, the problem above was generalised by Sala, Gabrys, Schoeny, and Dolecek [120]. More precisely, they consider a *coded* version of Levenshtein's problem, where it is now assumed that a length n string x is a codeword of an $(\ell-1)$-insdel-correcting q-ary code, and, as before, the main goal is to determine the minimum number of traces $t(n,\ell)$ required to recover x exactly, where the traces are obtained by corrupting x with arbitrary, but different, patterns of at most d insertions. One can recover Levenshtein's original problem (for the case of insertions) from the formulation above by setting $\ell = 1$. They succeed in giving an upper bound for $t(n,\ell)$ that is tight when the underlying code is arbitrary. Gabrys and Yaakobi [121] studied the analogous problem in the case of deletions, where x is assumed to belong to a 1-insdel-correcting code. Horovitz and Yaakobi [122] focused on a more general setting where some of the traces may be affected by more errors than others. Similarly to previous works, they begin by studying the minimum number of traces required to exactly recover the input string x when it is assumed that x is a codeword of a code with minimum distance α, where the notion of distance depends on the type of errors[6]. Furthermore, they also study a variant of the problem above in the case where the number of traces t is fixed. Namely, for a fixed number of traces t, they are interested in the smallest minimum distance α of the code which ensures that every codeword can be exactly recovered from t traces. Finally, orthogonal generalisations of Levenshtein's model have also been studied: Yaakobi and Bruck [123] consider a generalisation under worst-case substitutions in connection to information retrieval, and the problem of reading data in racetrack memories has been modelled as trace reconstruction under worst-case deletions and insertions with certain correlations between traces [12, 13, 14].

As mentioned above, the probabilistic setting where the t traces $(Y^{(i)})_{i\in[t]}$ of x are i.i.d. according to the output distribution Y_x of some channel Ch on input x was first studied by Levenshtein [118], who focused on the case where Ch is a DMC. There, the goal was to determine, given targets ε and δ, the minimum number of traces $t(n,\varepsilon,\delta)$ required to obtain x' such that $d_H(x,x') \leq \delta$, where d_H denotes the Hamming distance, with probability at least $1-\varepsilon$. This generalises the notion of exact

[6] We say a code $\mathcal{C}$ has *minimum distance* α according to some distance function $\rho(\cdot,\cdot)$ if $\min_{c,c'\in\mathcal{C}:c\neq c'} \rho(c,c') = \alpha$.

(probabilistic) recovery, which corresponds to the case where $\delta = 0$. Later, Batu, Kannan, Khanna, and McGregor [17], motivated by applications in computational biology, considered a similar problem, called *trace reconstruction*, where Ch is the deletion channel. Similarly to the work of Levenshtein, the goal is to design (preferably efficient) reconstruction algorithms which recover the input string x from as few traces as possible with high probability according to some meaningful notion of reconstruction. They considered two main settings: *Worst-case* trace reconstruction, where the error probability of the reconstruction algorithm must be small for *every* input x, and *average-case* trace reconstruction, where the error probability of the reconstruction algorithm must be small *on average* over all inputs x. We formally define these two notions below. Note that although we only provide definitions for the case of binary strings, these can be immediately generalised to strings over arbitrary finite alphabets.

Definition 2.16 (Worst-case trace reconstruction algorithm, [17])**.** *An algorithm* Rec *is said to be a* $(t = t(n), d)$-worst-case trace reconstruction algorithm *if for n large enough and all $x \in \{0,1\}^n$ it holds that*

$$\Pr[\mathsf{Rec}(Y_x^{(1)}, Y_x^{(2)}, \dots, Y_x^{(t)}) = x] \geq 1 - 1/n,$$

where the $(Y_x^{(i)})_{i\in[t]}$ are i.i.d. according to the output distribution of BDC_d on input x.

Definition 2.17 (Average-case trace reconstruction algorithm, [17])**.** *An algorithm* Rec *is said to be a* $(t = t(n), d)$-average-case trace reconstruction algorithm *if for n large enough it holds that*

$$2^{-n} \cdot \sum_{x\in\{0,1\}^n} \Pr[\mathsf{Rec}(Y_x^{(1)}, Y_x^{(2)}, \dots, Y_x^{(t)}) = x] \geq 1 - 1/n,$$

where the $(Y_x^{(i)})_{i\in[t]}$ are i.i.d. according to the output distribution of BDC_d on input x.

Some works have studied a version of trace reconstruction with a *fixed* number of traces. Haeupler and Mitzenmacher [124] considered the capacity of the t-trace deletion channel when the deletion probability $d \to 0$, which corresponds to the optimal rate of reliable information transmission when the receiver observes a fixed number t of independent traces. They extended results of Kalai, Mitzenmacher, and Sudan [78] and Kanoria and Montanari [79] originally obtained for the case $t = 1$, showing that the capacity is at least

$$1 - c_t \cdot d^t \log(1/d) - O(d^t)$$

when $d \to 0$ for fixed t, where $c_t > 0$ is an explicit constant depending only on t. This rate is achievable by a uniform input distribution. In another direction, Srinivasaradhan, Du, Diggavi, and

Fragouli [125, 126] proposed maximum likelihood and maximum a posteriori algorithms for trace reconstruction from a fixed number of traces, and empirically study their performance. This problem was also studied by Sabary, Yaakobi, and Yucovich [127] for the case of two traces.

2.5.3.1 Average-case trace reconstruction

We begin by surveying previous work on average-case trace reconstruction. Batu, Kannan, Khanna, and McGregor [17] were the first to obtain results in this setting by introducing and analysing the *Bitwise Majority Alignment* (BMA) algorithm, a variant of which is used in the heuristic decoder for DNA-based data storage from [16].

The BMA algorithm is simple to describe. We start by looking at the first bit of every trace, and use the majority of all such t bits as the guess for the first bit of the input x. To guess the second bit of x, we look at the second bit of all the traces whose first bits coincided with the first majority, and at the first bit of all the other traces. Here, we are making the bet that in almost all traces which agree with the majority the first bit of x was not deleted, and that in all other traces the first bit of x was deleted (and hence the first bit of these traces is a later bit of x). More generally, suppose we observe t traces $Y^{(1)}, \dots, Y^{(t)}$ and set up a counter $c_i = 1$ for each trace $Y^{(i)}$. Then, the BMA algorithm has n rounds, guesses one bit of x per round, and proceeds as follows in round j:

1. Let m_j denote the majority of $Y^{(i)}_{c_i}$ for $i \in [t]$. Then, set $x'_j = m_j$;
2. For each $i \in [t]$, if $c_i = m_j$, set $c_i \leftarrow c_i + 1$;
3. Go back to Step 1 with $j \leftarrow j + 1$.

It was shown in [17] that the BMA algorithm is an *efficient* (i.e., running in time polynomial in n) (t, d)-average-case trace reconstruction algorithm with $t = c_1 \log n$ and $d \leq \frac{c_2}{\log n}$ for absolute constants $c_1, c_2 > 0$. This result was extended in [128] to a setting where each bit of x may be corrupted not only by i.i.d. deletions, but also by i.i.d. geometric insertions of random bits and substitutions. In particular, they show that $O(\log n)$ traces are also enough in this more general setting whenever the deletion and insertion probabilities are $O(1/\log^2 n)$, and the substitution probability is an arbitrary constant. An improvement of these results, allowing deletion and insertion probabilities $O(1/\log n)$ for average-case trace reconstruction with $t = O(\log n)$ traces, was obtained subsequently by Viswanathan and Swaminathan [129].

Later, a breakthrough result by Holenstein, Mitzenmacher, Panigrahy, and Wieder [59] showed that efficient average-case trace reconstruction is possible for small enough *constant* deletion probability d from $t = \text{poly}(n)$ traces. This was then improved in the "large alphabet setting" by McGregor, Price, and Vorotnikova [130], who showed that $\exp(\sqrt{\log n} \cdot \text{poly}(\log\log n))$ traces are sufficient for average-case trace reconstruction when the input alphabet has size $q = \Theta(\log n)$ and the deletion probability d is constant.[7] These results were improved further, first by work of Peres and Zhai [131], who proved that $t = \exp(O(\sqrt{\log n}))$ traces are sufficient for average-case trace reconstruction of binary strings for any $d < 1/2$. Subsequently, Holden, Pemantle, and Peres [57] improved the result of [131] to $t = \exp(O(\log^{1/3} n))$ traces and extended it to all constant $d \in (0, 1)$. We note that this result holds not only in the case of i.i.d. deletions, but also when i.i.d. deletions are combined with the insertion of a geometric number of uniformly random bits.

Interestingly, there is a different coding-theoretic perspective of results in average-case trace reconstruction that ties with the question of designing coding schemes that can be reliably decoded with access to multiple, but few, traces of a codeword corrupted by i.i.d. deletions. By an averaging argument, every (t, d)-average-case trace reconstruction algorithm implies the existence of a family of codes $\mathcal{C} \subseteq \{0, 1\}^n$ with 1 bit of redundancy satisfying the following property: There exists a reconstruction algorithm Rec such that for every codeword $c \in \mathcal{C}$ we have

$$\Pr[\mathsf{Rec}(Y_c^{(1)}, Y_c^{(2)}, \dots, Y_c^{(t)}) = c] \geq 1 - 2/n,$$

where the $Y_c^{(i)}$ are i.i.d. according to the output distribution of BDC_d on input c. However, we remark that although the code $\mathcal{C}$ may be efficiently decodable from t traces, average-case trace reconstruction results do not guarantee that $\mathcal{C}$ is efficiently encodable. This perspective casts average-case trace reconstruction as a subset of what we will come to call *coded trace reconstruction* in Chapter 5. In contrast with the results above, and as previously mentioned, we will be mostly interested in designing *efficiently encodable and decodable* high-rate coding schemes that can be decoded from few traces. Notably, we will make use of techniques from average-case trace reconstruction to do so, in particular those from [59], which we discuss below and in Chapter 5.

The HMPW trace reconstruction algorithm. The discussion below is based on [59], and is an adaptation of the exposition already found in [5]. We begin by introducing the important concept of

[7] We note that trace reconstruction becomes easier as the alphabet size grows. We discuss this in more detail in Chapter 5.

subsequence-unique strings.[8]

Definition 2.18 (w-subsequence-unique string)**.** *A string $x \in \{0,1\}^n$ is said to be* w-subsequence-unique *if for every $a, b \in [n]$ such that $a + w, b + 1.1w \leq n + 1$ and either $a < b$ or $b + 1.1w < a + w$, we have that the substring $x[a, a + w)$ is not a subsequence of $x[b, b + 1.1w)$.*

We remark that the constant 1.1 in Definition 2.18 is arbitrary, and can be replaced by any other constant close to 1. Intuitively, a string x is w-subsequence-unique if no length w substring of x appears as a *subsequence* of another slightly longer substring of x, except for the trivial case where the longer substring contains the shorter one. Note that these strings have been defined under the name "substring-unique" in [59]. We chose to change this name to avoid confusion with a different definition under the same name from [132]. The following result about subsequence-unique strings was established in [59].

Theorem 2.9 ([59, Theorem 2.2])**.** *For $w = 100 \log n$ and every small enough constant deletion probability d there exists an algorithm that reconstructs every w-subsequence-unique string $x \in \{0,1\}^n$ with probability at least $1 - \exp(-\Omega(n))$ from $\mathrm{poly}(n)$ traces in time $\mathrm{poly}(n)$.*

Since a uniformly random string is w-subsequence-unique for $w = 100 \log n$ with high probability [59], Theorem 2.9 leads to an average-case trace reconstruction algorithm using polynomially many traces for constant deletion probability. The trace reconstruction algorithm that yields this theorem, which we call the HMPW algorithm, will be a key catalyst behind our results on coded trace reconstruction in Chapter 5, and we provide a high-level discussion of the main ideas present in the algorithm below. A more detailed analysis can be found in Section 5.4.

As already mentioned in [59], the HMPW algorithm may be seen as an iterative voting-based trace reconstruction, just like the earlier BMA algorithm from [17]. The intuitive difference between the two methods is that the HMPW algorithm only allows a subset of "good" traces to vote on the value of the next bit, while we saw that the BMA algorithm allows every trace to vote equally.

Fix a w-subsequence-unique string $x \in \{0,1\}^n$. The HMPW algorithm begins with a bootstrapping step which recovers the first $O(\log n)$ bits of x with probability at least $1 - \exp(-\Omega(n))$ using $\mathrm{poly}(n)$ traces and time. This bootstrapping step follows from the more general algorithm in the lemma below.

[8] The definition of w-subsequence unique string presented here differs slightly from the one found in the conference version of [59], but leads only to minor modifications in the analysis of the HMPW trace reconstruction algorithm.

Lemma 2.10 ([59, Theorem 2.1], adapted)**.** *Suppose that $d < 1/3$. Then, there is an algorithm which, for an arbitrary string $x \in \{0,1\}^n$, recovers the first h bits $x_1, x_2, \ldots, x_h$ with probability at least $1 - \varepsilon$ using $O(he^{14dh} \log(1/\varepsilon))$ time and traces of x.*

We can now assume we have already recovered the first $i-1$ bits $x_1, x_2, \ldots, x_{i-1}$ for $i = \Omega(\log n)$, and our goal is to recover x_i with high probability using this knowledge from $\mathrm{poly}(n)$ traces and time, where the success probability, the number of traces, and the runtime are independent of i. Note that we cannot afford to run the bootstrapping algorithm from Lemma 2.10 to recover more than $O(\log n)$ bits of x, since we would need a superpolynomial number of traces. Therefore, we require a different approach going forward.

To recover x_i from the previous bits and additional traces, the HMPW algorithm considers a length-w "anchor" substring $x[i-v-w, i-v)$, for an appropriate parameter v, and retains only the traces which feature this anchor as a substring, i.e., traces that have a *matching* with $x[i-v-w, i-v)$, as formalised in the following definition.

Definition 2.19 (Matching)**.** *Fix a string $x \in \{0,1\}^n$ and let T denote a trace of x. Then, we say that there is a* matching of $x[a,b)$ in T *if there exists some u such that $T[u-(b-a), u) = x[a,b)$.*

After setting w and v appropriately, the two main insights regarding matchings are that (i) matchings of $x[i-v-w, i-v)$ occur in a sizeable fraction of the traces, meaning $\mathrm{poly}(n)$ traces suffice to collect polynomially many "good" traces matching $x[i-v-w, i-v)$, and (ii) by the w-subsequence-uniqueness of x, the trace bits close to the end of the matching must come from positions in x close to $i-v$, unless there are many deletions in the trace T. Put differently, the second property ensures that the first bits of the suffix $\mathsf{Suff} = T[u:]$ come from positions of x close to $i-v$ with high probability. Consequently, it is possible to show that for an appropriate $j \approx v$, the value of the bit Suff_j is markedly influenced by the value of x_i, in the sense that it satisfies a threshold property depending on whether $x_i = 0$ or $x_i = 1$: There are bounds $B_1 > B_0 + \frac{1}{\mathrm{poly}(n)}$ such that

$$\Pr[\mathsf{Suff}_j = 1 | x_i = 1] \geq B_1 > B_0 \geq \Pr[\mathsf{Suff}_j = 1 | x_i = 0].$$

This means that we can recover x_i with high probability by using $\mathrm{poly}(n)$ good traces to estimate $\Pr[\mathsf{Suff}_j = 1]$, B_1, and B_0 to within sufficiently small additive error, and checking whether the estimate is closer to B_1 (in which case we should guess $x_i = 1$) or B_0 (in which case we should guess $x_i = 0$). By repeating this procedure for the remaining $i \leq n$, the HMPW algorithm yields Theorem 2.9.

2.5.3.2 Worst-case trace reconstruction

Recalling Definitions 2.16 and 2.17, it is clear that worst-case trace reconstruction is significantly more demanding than average-case trace reconstruction, because in the former we require that the reconstruction algorithm succeed with high probability for an arbitrary input string. Therefore, it is not surprising that the known results about worst-case trace reconstruction are substantially weaker than the state-of-the-art algorithms for average-case trace reconstruction.

The first result on worst-case trace reconstruction was obtained by Batu, Kannan, Khanna, and McGregor [17], who showed that a modified version of the BMA algorithm described above is a (t, d)-worst-case trace reconstruction algorithm for $d = O(n^{-(1/2+\varepsilon)})$ and $t = O(n \log n)$, where $\varepsilon > 0$ is an arbitrary constant. This result was extended to worst-case trace reconstruction from deletions and insertions of strings *without long runs* by Kannan and McGregor [128]. Recently, Chen, De, Lee, Servedio, and Sinha [133] improved the result from [17] by showing that $\mathrm{poly}(n)$ traces suffice for worst-case trace reconstruction when $d = O(n^{-(1/3+\varepsilon)})$ for an arbitrary constant $\varepsilon > 0$. The first non-trivial reconstruction algorithm for worst-case trace reconstruction with any *constant* deletion probability $d < 1$ was obtained by Holenstein, Mitzenmacher, Panigrahy, and Wieder [59], showing that $\exp(\sqrt{n} \cdot \mathrm{poly}(\log n))$ traces are enough by analysing a restricted class of reconstruction algorithms, called *mean-based algorithms*, that only use single-bit statistics of the trace distribution. We discuss such algorithms in more detail below. Their analysis was subsequently improved in an elegant way concurrently by De, O'Donnell, and Servedio [60] and Nazarov and Peres [61], who showed that $\exp(O(n^{1/3}))$ traces are both sufficient *and necessary* for mean-based algorithms. Moreover, they showed that $\exp(O(n^{1/3}))$ traces are also sufficient for mean-based trace reconstruction in a more general channel model combining deletions, geometric insertions of random bits, and substitutions. The result above in the case of i.i.d. deletions only was subsequently extended to some settings where different input coordinates or symbols may be deleted with different probabilities by Hartung, Holden, and Peres [134]. While this thesis was being written, Chase [135] improved the best upper bound on worst-case trace reconstruction to $\exp(n^{1/5}\mathrm{poly}(\log n))$ traces by going beyond mean-based reconstruction algorithms, and Grigorescu, Sudan, and Zhu [136] studied the performance of mean-based algorithms when distinguishing strings at small Hamming or edit distance from each other.

Mean-based trace reconstruction. We proceed to discuss the approach of [60, 61] towards worst-case trace reconstruction in more detail, as we will extend it to handle more general replication errors in Chapter 5. We will focus on the simpler case where the goal is to distinguish between two distinct arbitrary strings $x, x' \in \{-1,1\}^n$. In other words, we observe traces from a string a that is either x or x', and we wish to correctly guess with high probability whether $a = x$ or $a = x'$. Observe that we have changed the alphabet from $\{0,1\}$ to $\{-1,1\}$. This is without loss of generality, and will make for clearer exposition. An upper bound on the number of traces required to distinguish between arbitrary $x \neq x'$ can then be translated into a similar upper bound on the number of traces for worst-case trace reconstruction.

The traces of an input string $x \in \{-1,1\}^n$ are i.i.d. according to the output Y_x of the deletion channel BDC_d on input x. Noting that $|Y_x| \leq n$ always, we can consider a padded version of Y_x, which we denoted by Y'_x, that is obtained by padding Y_x with $n - |Y_x|$ zeros. We can then define the *mean trace* of x as the vector $\mu_x \in \mathbb{R}^n$ given by

$$\mu_x = \left(\mathbb{E}[(Y'_x)_1], \mathbb{E}[(Y'_x)_2], \ldots, \mathbb{E}[(Y'_x)_n]\right),$$

where $(Y'_x)_i$ denotes the i-th coordinate of Y'_x. Then, mean-based reconstruction algorithms attempt to distinguish between $a = x$ and $a = x'$ by first estimating μ_a from the t traces, and then making some decision based on this estimate of μ_a only. By standard arguments, obtaining upper and lower bounds on the number of traces required for such mean-based algorithms is equivalent to obtaining upper and lower bounds on the quantity $\|\mu_x - \mu_{x'}\|_1$. Both De, O'Donnell, and Servedio [60] and Nazarov and Peres [61] show that

$$\|\mu_x - \mu_{x'}\|_1 \geq \exp(-Cn^{1/3}) \tag{2.10}$$

for every $x \neq x'$, where C is an absolute constant independent of x, x', and n, and that this is tight, leading to the following result.

Theorem 2.10 ([60, 61])**.** *For every n and constant deletion probability $d < 1$ there exists a (t,d)-worst-case trace reconstruction algorithm for $t = \exp(O(n^{1/3}))$ with error probability $\exp(-n)$ running in time $\exp(O(n^{1/3}))$. Moreover, every mean-based trace reconstruction algorithm requires $\exp(\Omega(n^{1/3}))$ traces to be successful with probability at least $3/4$.*

We describe how the lower bound in (2.10) is obtained in [60, 61], which leads to an upper bound on the number of traces required by mean-based algorithms. Deriving the desired time complexity requires a slightly more general, but very similar, argument. We defer this discussion to Chapter 5.

First, to every $x \in \{-1,1\}^n$ we can associate the polynomial P_x over the complex numbers satisfying

$$P_x(z) = \sum_{i=1}^{n} x_i z^{i-1}$$

for $z \in \mathbb{C}$. In the same way, we can associate to x its *mean trace* polynomial $\overline{P}_x$ defined as

$$\overline{P}_x(z) = \sum_{i=1}^{n} (\mu_x)_i z^{i-1}.$$

The key result proved in [60, 61] to bound $\|\mu_x - \mu_{x'}\|_1$ via this approach is that P_x and $\overline{P}_x$ are related through a change of variable. Namely, for every $z \in \mathbb{C}$ it holds that

$$P_x(z) = (1-d) \cdot \overline{P}_x(d + (1-d)z). \tag{2.11}$$

Then, with (2.11) in mind, one hopes to obtain a suitable lower bound on $\|\mu_x - \mu_{x'}\|_1$ by lower bounding $|P_x(z) - P_{x'}(z)|$ for a suitable choice of z. An important property of $P_x(z) - P_{x'}(z)$ is that, due to linearity, it satisfies $P_x(z) - P_{x'}(z) = 2p(z)$, where p is a *Littlewood polynomial*: Its coefficients lie in $\{-1,0,1\}$. The following result of Borwein and Erdélyi provides such a lower bound on small subarcs of the unit circle for general Littlewood polynomials.

Lemma 2.11 ([137]). *There is an absolute constant $c > 0$ such that for any nonzero Littlewood polynomial p and every $L \geq 1$ it holds that*

$$\max_{z = e^{i\varphi} : |\varphi| \leq \frac{\pi}{L}} |p(z)| \geq \exp(-cL).$$

By Lemma 2.11, there exists z_L such that $|P_x(z_L) - P_{x'}(z_L)| \geq \exp(-cL)$ and $z_L = e^{i\varphi}$ for $|\varphi| \leq \frac{\pi}{L}$. Using the fact that $|d + (1-d)z_L| = 1 + O(1/L^2)$ and (2.11), we conclude that

$$\begin{aligned}
\exp(-cL) &\leq |P_x(z_L) - P_{x'}(z_L)| \\
&= (1-d)\left|\overline{P}_x(d + (1-d)z_L) - \overline{P}_{x'}(d + (1-d)z_L)\right| \\
&\leq \|\mu_x - \mu_{x'}\|_1 \cdot \exp(O(n/L^2)),
\end{aligned}$$

and the desired lower bound on $\|\mu_x - \mu_{x'}\|_1$ follows by setting $L = n^{1/3}$. Finally, relying on results from [138], it can be seen that the analysis above is tight up to absolute constants in the exponent.

2.5.3.3 Lower bounds for trace reconstruction

In the previous sections, we surveyed the design of reconstruction algorithms for trace reconstruction. However, there has also been some work on general lower bounds on the number of traces required by these algorithms.

The first lower bound for worst-case trace reconstruction was proved by Batu, Kannan, Khanna, and McGregor [17], who gave a simple argument that $\Omega(nd(1-d))$ traces are required for worst-case trace reconstruction for any deletion probability d. This lower bound can be derived by studying the number of traces required to distinguish between the strings $1^{n/2-1}0^{n/2+1}$ and $1^{n/2+1}0^{n/2-1}$. The first lower bound for average-case trace reconstruction was proved by McGregor, Price, and Vorotnikova [130], who showed that $\Omega(\log^2 n)$ traces are required for any constant deletion probability $d > 0$. There is a general way of transforming a lower bound for worst-case trace reconstruction into a lower bound for average-case trace reconstruction, which so far has been the only way previous lower bounds in the latter setting. The two lower bounds above were improved by Holden and Lyons [139], who gave the first superlinear lower bound for worst-case trace reconstruction of $\Omega\left(\frac{n^{5/4}}{\sqrt{\log n}}\right)$ traces by analysing a more complex pair of strings. This lower bound then immediately implies an improved lower bound of $\Omega\left(\frac{\log^{9/4} n}{\sqrt{\log\log n}}\right)$ traces for average-case trace reconstruction. The analysis of Holden and Lyons was subsequently refined by Chase [140], leading to improved lower bounds of $\Omega\left(\frac{n^{3/2}}{\log^7 n}\right)$ traces for worst-case trace reconstruction and $\Omega\left(\frac{\log^{5/2} n}{(\log\log n)^7}\right)$ traces for average-case trace reconstruction. Note that there is currently a large gap between upper and lower bounds for trace reconstruction in both the worst-case and average-case settings.

2.5.3.4 Related models

Besides the results discussed above, there has also been some work on other models related to both trace reconstruction over channels with deletions and DNA-based data storage. We briefly discuss them here.

Some works have focused on studying other theoretical models which attempt to capture different aspects of DNA-based data storage. Magner, Duda, Szpankowski, and Grama [33] model nanopore-based sequencing as trace reconstruction over general sticky channels (which is more approachable than trace reconstruction over channels with deletions), and obtained some results regarding the number of traces required for reconstruction. Mao, Diggavi, and Kannan [141] study the capacity of nanopore-based sequencing, but focus on more low-level aspects of this technology and consider an incomparable model for reconstruction. A series of works have been motivated by DNA-based data storage systems that employ different sequencing techniques, both in the setting of worst-case errors [142, 143, 144, 145] and random errors [146, 147, 148, 149, 150, 151, 152, 153]. Roughly speaking, in the model considered by these works the input string is broken up into small blocks, each of which is corrupted by a bounded number of worst-case errors (deletions, insertions, or substitutions). The output of the channel is a random permutation of all, or a subset of, the corrupted blocks, meaning in particular that the receiver does not have any information about which corrupted block corresponds to the i-th block of the input string. Another reconstruction problem motivated by DNA-based data storage which has received significant attention recently is that of reconstructing a string given some information about its set of substrings [154, 155, 132, 156, 157, 158]. These models are incomparable to trace reconstruction and the coded trace reconstruction problem we study in Chapter 5.

Another set of works has focused on modifications or extensions of the trace reconstruction problem. Ban, Chen, Freilich, Servedio, and Sinha [159] introduced the problem of population recovery over the deletion channel, a generalisation of trace reconstruction where the input string is now sampled independently from a fixed distribution for each trace, and the goal is to approximate this distribution in statistical distance from as few traces as possible. Their results were subsequently improved by Ban, Chen, Servedio, and Sinha [160] and Narayanan [161]. Other orthogonal generalisations of trace reconstruction have been studied. Davies, Racz, and Rashtchian [162] studied trace reconstruction over trees (in this setting, the original trace reconstruction problem corresponds to reconstruction over paths). Krishnamurthy, Mazumdar, McGregor, and Pal [163] studied trace reconstruction of matrices, and also trace reconstruction of sparse vectors. Chen, De, Lee, Servedio, and Sinha [164] studied a smoothed version of worst-case trace reconstruction where a worst-case string is corrupted by an arbitrary constant rate of i.i.d. substitutions. They show that in this case poly(n) traces and time are enough, in contrast with exponentially many traces in the worst-case setting. Narayanan and Ren [165] studied trace reconstruction combined with random cyclic shifts. Sabary, Yucovich, Shapira, and Yaakobi [166] studied the problem of recovering a good approximation in edit distance of the input string with few traces. Bhardwaj, Pevzner, Rashchtian, and Safonova [167] survey trace reconstruction variants relevant to computational biology. Some approximate notions of trace reconstruction have been studied by Davies, Racz, Rashtchian, and Schiffer [168].

Chapter 3

Capacity bounds for synchronisation channels

In this chapter, we derive analytical capacity upper bounds for the geometric sticky channel and the geometric deletion channel. The sharp analytical bounds we obtain for the geometric sticky channel are a by-product of the explicit candidate distributions we design for Theorem 2.5 with zero KL-gap everywhere, which are a first step towards a computer-unaided treatment of this channel. They also improve upon previous bounds for some range of the replication parameter. With respect to the geometric deletion channel, we modify the original approach of [40] for designing candidate distributions, leading to generally improved analytical capacity upper bounds. Under a plausible conjecture, we show numerically that these improvements are significant over a large range of parameters, and thus deserve further study. These ideas will also prove useful in the derivation of improved capacity upper bounds for the discrete-time Poisson channel in Chapter 4. We also use these techniques to give a proof without computer assistance that the capacity of the geometric deletion channel is at most 0.73 bits/channel use in the large replication regime $p \to 1$.

We present our sharp analytical capacity upper bounds for the geometric stick channel in Section 3.1. Then, we study the capacity of the geometric deletion channel in Section 3.2.

The material presented in this chapter is based on [2] with minor modifications to improve exposition and consistency with the rest of this thesis.

In order to avoid dealing with several leading constants in intermediate computations and results, we will be working with information-theoretic quantities taken with respect to the natural logarithm ln instead of the base-2 logarithm log as introduced in Section 2.4. For each information-theoretic quantity, such as the Kullback-Leibler divergence D_{KL}, the mutual information I, the Shannon entropy H, and the binary entropy function h, we denote its corresponding version under the natural logarithm by $D^{(e)}_{\mathsf{KL}}$, $I^{(e)}$, $H^{(e)}$, and $h^{(e)}$, respectively. In general, going from the base-e quantity to the corresponding base-2 quantity is done by dividing the former by $\ln 2$.

3.1 Analytical capacity upper bounds for the geometric sticky channel

Given that excellent numerical capacity upper bounds exist for sticky channels due to their special structure [32, 1], it is natural to wonder whether an analytical approach to the capacity of these channels would yield equally sharp analytical capacity upper bounds, or even succeed in determining the exact capacity of such channels.

In this section, we make significant progress towards a complete conceptual understanding of the capacity of the geometric sticky channel. Based on Theorems 2.6 and 2.5, we design for the first time a family of candidate distributions for Theorem 2.5 *with zero KL-gap everywhere.* This means that every distribution Y in this family satisfies

$$D^{(e)}_{\mathsf{KL}}(Y_x \| Y) = a\mathbb{E}[Y_x] + b$$

for all $x \in \mathbb{N}$ and some $a, b \in \mathbb{R}$. In other words, we have KL-gap $\Delta(x) = 0$ for all x. Therefore, these distributions provably satisfy one of the optimality conditions from Theorem 2.5. Remarkably, previous attempts towards achieving this for channels with deletions in [40] failed, suggesting one more fundamental dichotomy between repeat channels with and without deletions. We note also that previous works on numerical capacity upper bounds for sticky channels [32, 1], which make use of the (equivalent) framework from Theorem 2.4, never explicitly considered the slackness with which the analogous constraints in Theorem 2.4 are satisfied by the distributions they design.

Our result suggests a promising approach for determining the *exact* capacity of the geometric sticky channel, and leaves open the exciting possibility that an adaptation of our techniques leads to distributions which are also realisable as channel output distributions. Regardless, from experience, the fact that the candidate distributions have zero KL-gap suggests that they should yield sharp capacity upper bounds. We will see that this is indeed the case, and we also manage to improve upon the current state-of-the-art numerical upper bounds for some reported parameters.

As introduced before, the geometric sticky channel independently replicates each bit according to a geometric distribution supported in $\mathbb{N}$. In other words, the geometric sticky channel with replication parameter $p \in (0,1)$ is a repeat channel with replication rule R satisfying

$$R(r) = (1-p)p^{r-1}, \quad r = 1, 2, \dots.$$

Taking into account Theorem 2.6, in order to understand the capacity $\mathsf{Cap}(R)$ of the geometric sticky channel it suffices to understand the mean-limited capacity of the DMC Ch_R which on input $x \in \mathbb{N}$ outputs

$$Y_x = \sum_{i=1}^{x} R_i$$

for R_i i.i.d. according to R. Due to the properties of the geometric distribution, the output distribution Y_x has a nice form. We have $Y_x \sim x + \mathsf{NB}_{x,p}$, where we recall $\mathsf{NB}_{x,p}$ is a negative binomial distribution satisfying

$$\mathsf{NB}_{x,p}(y) = \binom{y+x-1}{y}(1-p)^x p^y, \quad y = 0, 1, 2, \dots.$$

This holds because $\mathsf{NB}_{x,p} = \sum_{i=1}^{x} R_{0i}$ for R_{0i} i.i.d. according to $R_0 \sim \mathsf{Geom}_{0,p}$. Then, it suffices to note that $R \sim 1 + R_0$. As a result, we have that

$$Y_x(y) = \mathsf{NB}_{x,p}(y-x) = \binom{y-1}{x-1}(1-p)^x p^{y-x}, \quad y = x, x+1, \dots \tag{3.1}$$

for all $x \in \mathbb{N}$. In the following section, we design zero-KL gap candidate distributions to be used in Theorem 2.5 with Ch_R.

The remaining material in this section is a reproduction of technical material from [2, Section III], with modifications to improve quality of exposition and consistency with the rest of the thesis.

3.1.1 Distributions with zero KL-gap everywhere for the geometric sticky channel

In this section, we show how to design distributions for Theorem 2.5 with zero KL-gap everywhere for the geometric sticky channel. At a high-level, the design of such a class of distributions follows the blueprint from [40], with some key differences:

1. We consider a general form for a distribution Y that leads to a simple expression for $D_{\mathsf{KL}}^{(e)}(Y_x||Y)$ with respect to some function g to be defined;
2. We show that Y satisfies $D_{\mathsf{KL}}^{(e)}(Y_x||Y) = a\mathbb{E}[Y_x]+b$ for every x if and only if g satisfies a functional equation of the form
$$\mathbb{E}[g(Y_x)] = f(x), \quad x = 1, 2, \ldots \tag{3.2}$$
for a specific function f;
3. We show that the functional equation above has a solution g, and we instantiate Y with it. This yields the desired result *if* we can show Y is a valid distribution, i.e., it can be normalised;
4. We show that Y with this choice of g can be normalised. This yields a valid distribution with zero KL-gap everywhere.

Cheraghchi [40] attempted to apply the approach above to the deletion and Poisson-repeat channels. However, he could not realise Steps 3 and 4 above in conjunction for either of these channels. As a result, Cheraghchi turned to alternative techniques that lead to candidate distributions with positive KL-gap for all $x \in \mathbb{N}$. Remarkably, in this section we show that the ideal approach above *can* be made to work successfully for the geometric sticky channel, leading in particular to sharp analytical capacity upper bounds for this channel. We believe that the situation above highlights an important difference between repeat channels with and without deletions. We have yet to find an example of a repeat channel with deletions for which the approach above works, while we conjecture that we can design candidate distributions with zero KL-gap everywhere (and hence potentially obtain sharp analytical capacity upper bounds) for all sticky channels.

We begin by noting that $D_{\mathsf{KL}}^{(e)}(Y_x||Y)$ can be rewritten as

$$D_{\mathsf{KL}}^{(e)}(Y_x||Y) = -H^{(e)}(Y_x) - \sum_{y=x}^{\infty} Y_x(y) \ln Y(y).$$

Then, recalling (3.1) and noting that $\mathbb{E}[Y_x] = \frac{x}{1-p}$, we have

$$\begin{aligned} -H^{(e)}(Y_x) &= \mathbb{E}\left[\ln \binom{Y_x - 1}{x - 1}\right] + x\ln(1-p) + (\mathbb{E}[Y_x] - x)\ln p \\ &= \mathbb{E}\left[\ln \binom{Y_x - 1}{x - 1}\right] - \mathbb{E}[Y_x]h^{(e)}(p) \end{aligned}$$

$$= \mathbb{E}[\ln[(Y_x - 1)!]] - \mathbb{E}[\ln[(Y_x - x)!]] - \ln[(x-1)!] - \mathbb{E}[Y_x]h^{(e)}(p). \tag{3.3}$$

Our goal is to design a family of distributions Y such that $D^{(e)}_{\mathsf{KL}}(Y_x||Y) = a\mathbb{E}[Y_x] + b$ for all $x = 1, 2, \dots$ and some $a, b \in \mathbb{R}$. Given $q \in (0,1)$, consider the distribution $Y^{(q)}$ with general form

$$Y^{(q)}(y) = y_0 q^y \exp(g(y) - yh^{(e)}(p)), \quad y = 1, 2, \dots \tag{3.4}$$

where y_0 is the normalising factor and g is a function to be defined. Using (3.3), we have

$$\begin{aligned} D^{(e)}_{\mathsf{KL}}(Y_x||Y^{(q)}) &= -H^{(e)}(Y_x) - \sum_{y=x}^{\infty} Y_x(y) \ln Y^{(q)}(y) \\ &= \mathbb{E}\left[\ln \binom{Y_x - 1}{x - 1}\right] - \mathbb{E}[Y_x]h^{(e)}(p) - \ln y_0 - \mathbb{E}[Y_x]\ln q - \mathbb{E}[g(Y_x)] + \mathbb{E}[Y_x]h^{(e)}(p) \\ &= -\ln y_0 - \mathbb{E}[Y_x]\ln q + \mathbb{E}[\ln[(Y_x - 1)]!] - \mathbb{E}[\ln[(Y_x - x)!]] - \ln[(x-1)!] - \mathbb{E}[g(Y_x)]. \end{aligned} \tag{3.5}$$

Taking into account (3.5), we would like to have

$$\mathbb{E}[g(Y_x)] = \mathbb{E}[\ln[(Y_x - 1)!]] - \mathbb{E}[\ln[(Y_x - x)!]] - \ln[(x-1)!] \tag{3.6}$$

for all $x \in \mathbb{N}$, so that

$$D^{(e)}_{\mathsf{KL}}(Y_x||Y^{(q)}) = -\ln y_0 - \mathbb{E}[Y_x]\ln q.$$

We will proceed to design such a function g. Before we begin, we first state a version of the Fubini-Tonelli theorem specialised for the counting measure on $\mathbb{N}$ and the Lebesgue measure on $[0,1]$ to exchange an expected value and an integral.

Lemma 3.1 ([169, Theorem 7.8, specialised]). *Let $(f_n)_{n\in\mathbb{N}}$ be a family of continuous functions $f_n : [0,1] \to \mathbb{R}$, and suppose that either*

$$\int_0^1 \sum_{n=1}^{\infty} |f_n(t)| dt < \infty$$

or

$$\sum_{n=1}^{\infty} \int_0^1 |f_n(t)| dt < \infty.$$

Then,

$$\int_0^1 \sum_{n=1}^{\infty} f_n(t)dt = \sum_{n=1}^{\infty} \int_0^1 f_n(t)dt.$$

Making use of Lemmas 2.7 and 3.1, and of the facts that $\mathbb{E}[Y_x] = \frac{x}{1-p}$ and that the probability generating function of Y_x is

$$\left(\frac{z(1-p)}{1-pz}\right)^x \tag{3.7}$$

whenever $|z| < 1/p$, we have

$$\ln[(x-1)!] = \int_0^1 \frac{1+t-tx-(1-t)^{x-1}}{t\ln(1-t)}dt \tag{3.8}$$

and[1]

$$\begin{aligned}\mathbb{E}[\ln[(Y_x-x)!]] &= \mathbb{E}\left[\int_0^1 \frac{1-t(Y_x-x)-(1-t)^{Y_x-x}}{t\ln(1-t)}dt\right] \\ &= \int_0^1 \frac{1-\frac{txp}{1-p}-\left(\frac{1-p}{1-p(1-t)}\right)^x}{t\ln(1-t)}dt. \end{aligned} \tag{3.9}$$

Consider now the functions

$$f_1(y,t) = \frac{1+t-ty(1-p)-\left(\frac{1-t}{1-pt}\right)^y/(1-t)}{t\ln(1-t)} \tag{3.10}$$

$$f_2(y,t) = \frac{1-typ-\left(\frac{1}{1+pt}\right)^y}{t\ln(1-t)}. \tag{3.11}$$

Recalling (3.7), observe that

$$\mathbb{E}[f_1(Y_x,t)] = \frac{1+t-tx-(1-t)^{x-1}}{t\ln(1-t)}, \tag{3.12}$$

$$\mathbb{E}[f_2(Y_x,t)] = \frac{1-\frac{txp}{1-p}-\left(\frac{1-p}{1-p(1-t)}\right)^x}{t\ln(1-t)}. \tag{3.13}$$

With (3.6) in view, we set

$$\begin{aligned}\Lambda_1(y) &= \int_0^1 f_1(y,t)dt, \\ \Lambda_2(y) &= \int_0^1 f_2(y,t)dt \end{aligned} \tag{3.14}$$

[1] We can justify the switching of the integral and expected value in (3.9) via Lemma 3.1 by noting that the function inside the integral in Lemma 2.7 can be extended by continuity to $[0,1]$ and is non-negative for all integers $z \geq 0$.

with f_1 and f_2 defined as in (3.10) and (3.11), respectively. Taking into account (3.13), we are able to show the following.

Lemma 3.2. *We have*

$$\mathbb{E}[\Lambda_1(Y_x)] = \int_0^1 \mathbb{E}[f_1(Y_x,t)]dt = \ln[(x-1)!],$$
$$\mathbb{E}[\Lambda_2(Y_x)] = \int_0^1 \mathbb{E}[f_2(Y_x,t)]dt = \mathbb{E}[\ln[(Y_x-x)!]]$$

for every $x \in \mathbb{N}$.

Proof. See Section 3.1.1.1. □

Consider the distribution $Y^{(q)}$ in (3.4) defined by the choice of g

$$g(y) = \ln[(y-1)!] - \Lambda_1(y) - \Lambda_2(y). \tag{3.15}$$

By Lemma 3.2, it follows that g satisfies (3.6), and so, recalling (3.5), we have

$$D^{(e)}_{\mathsf{KL}}(Y_x||Y^{(q)}) = -\ln y_0 - \mathbb{E}[Y_x]\ln q$$

provided that $Y^{(q)}$ is a valid distribution. In order to conclude the reasoning, it remains to show this fact, i.e., that

$$0 < 1/y_0 = \sum_{y=1}^{\infty} q^y \exp(g(y) - yh^{(e)}(p)) < \infty$$

if $q \in (0,1)$, and thus $Y^{(q)}$ can be normalised so that $\sum_{y=1}^{\infty} Y^{(q)}(y) = 1$. The following lemma shows that $Y^{(q)}(y)/y_0 = \Theta(q^y/\sqrt{y})$. If this holds, then we have the desired result whenever $q \in (0,1)$.

Lemma 3.3. *We have*

$$\left| yh^{(e)}(p) - g(y) - \frac{1}{2}\ln y \right| = O(1)$$

when $y \to \infty$ for every $p \in (0,1)$.

Proof. See Section 3.1.1.2. □

From the results of this section, it follows that $Y^{(q)}$ is a valid distribution and that

$$D^{(e)}_{\mathsf{KL}}(Y_x||Y^{(q)}) = -\ln y_0 - \mathbb{E}[Y_x]\ln q$$

for all $x \in \mathbb{N}$. Therefore, $Y^{(q)}$ achieves zero KL-gap for all $x \in \mathbb{N}$ and $q \in (0,1)$. Using Theorem 2.5, we conclude that

$$\mathsf{Cap}_\mu(\mathsf{Ch}_R) \leq \inf_{q\in(0,1)}(-\ln y_0 - \mu \ln q) \tag{3.16}$$

for all $\mu \geq 1$.

Finally, we point out that Lemma 3.3 implies that, given any $\mu > 1$, there is $q \in (0,1)$ such that $\mathbb{E}[Y^{(q)}] = \mu$ (see Section 3.1.3 for a proof). This will lead to easier to compute, but still tight, capacity upper bounds in Section 3.1.2.

3.1.1.1 Proof of Lemma 3.2

In this section, we prove Lemma 3.2, namely that

$$\begin{aligned}\mathbb{E}[\Lambda_1(Y_x)] &= \int_0^1 \mathbb{E}[f_1(Y_x,t)]dt = \ln(x-1)!,\\ \mathbb{E}[\Lambda_2(Y_x)] &= \int_0^1 \mathbb{E}[f_2(Y_x,t)]dt = \mathbb{E}[\ln(Y_x-x)!].\end{aligned}$$

The only problem lies with the first equality in each line (the second equality follows directly from (3.8) and (3.9) combined with (3.12) and (3.13)). We start by showing that this equality holds for Λ_1. This follows if the conditions of Lemma 3.1 are satisfied.

First, note that $f_1(y,\cdot)$ is continuous on $(0,1)$, and that

$$\lim_{t\to 1} f_1(y,t) = 0$$

and

$$\lim_{t\to 0} f_1(y,t) = 1 + \frac{y(1-p)(y(1-p)-3-p)}{2}$$

for all $y \geq 1$. This means that $f_1(y,\cdot)$ can be extended by continuity to $[0,1]$ (this does not change the integral). From here onwards we work with this extension. By Lemma 3.1, we only need to show that

$$\int_0^1 \mathbb{E}[|f_1(Y_x,t)|]dt < \infty.$$

We begin by showing that $f_1(y,t) \geq 0$ for all $t \in [0,1]$ if y is large enough. Recalling (3.10), the numerator of $f_1(y,t)$ is

$$h_1(y,t) = 1 + t - ty(1-p) - \left(\frac{1}{1-pt}\right)^y (1-t)^{y-1}.$$

We show that $h_1(y,t) \leq 0$ for all $t \in [0,1]$ if y is large enough. This gives the desired result since the denominator of $f_1(y,t)$ is $t\ln(1-t)$, which is negative for all $t \in (0,1)$. The first and second derivatives with respect to t of $h_1(y,t)$ are

$$\frac{\partial h_1}{\partial t}(y,t) = 1 - y(1-p) + (y(1-p) - (1-pt))\left(\frac{1}{1-pt}\right)^{y+1} (1-t)^{y-2}$$

and

$$\frac{\partial^2 h_1}{\partial t^2}(y,t) = \left((1-p)(3+p(1-4t))y - (1-p)^2y^2 - 2(1-pt)^2\right)\left(\frac{1}{1-pt}\right)^{y+2} (1-t)^{y-3}.$$

For fixed p, we can set $y^* = 1 + \frac{4}{(1-p)^2}$ so that

$$(1-p)(3+p(1-4t))y - (1-p)^2y^2 - 2(1-pt)^2 \leq 4y - (1-p)^2y^2 < 0$$

for all $t \in [0,1]$ when $y \geq y^*$, which implies that $\frac{\partial^2 h_1}{\partial t^2}(y,t) < 0$ for all $t \in (0,1)$ since all other terms in the expression are positive. As a consequence, it follows that $\frac{\partial h_1}{\partial t}(y,t)$ is decreasing in t for $y \geq y^*$. Combining this with the fact that $\frac{\partial h_1}{\partial t}(y,0) = 0$, we conclude that $\frac{\partial h_1}{\partial t}(y,t) \leq 0$ for all $t \in (0,1)$, provided that $y \geq y^*$. Finally, this implies that $h_1(y,t) \leq 0$ holds for all $t \in (0,1)$ when $y \geq y^*$, since $h_1(y,0) = 0$.

Consequently, we have

$$\int_0^1 \mathbb{E}[|f_1(Y_x,t)|]dt \leq \int_0^1 \left(\mathbb{E}[f_1(Y_x,t)] + 2\sum_{y=1}^{y^*} Y_x(y)|f_1(y,t)|\right) dt = \ln[(x-1)!] + C_{x,p} < \infty,$$

where

$$C_{x,p} = 2 \int_0^1 \sum_{y=1}^{y^*} Y_x(y) |f_1(y,t)| dt$$

is a finite constant depending only on x and p, since $f_1(y, \cdot)$ is continuous on $[0,1]$ for all $y \geq 1$, and therefore bounded as well. This means that Lemma 3.1 can be applied, which leads to the desired equality.

The argument for Λ_2 follows in an analogous, but simpler, way. In fact, recalling (3.11), the numerator of $f_2(y,t)$ is

$$h_2(y,t) = 1 - typ - \left(\frac{1}{1+pt}\right)^y,$$

and its derivative with respect to t is

$$\frac{\partial h_2}{\partial t}(y,t) = yp \left(\left(\frac{1}{1+pt}\right)^{y+1} - 1 \right).$$

Observe that $\frac{\partial h_2}{\partial t}(y,t) < 0$ for $t \in (0,1)$ and $y \geq 1$, which implies that $h_2(y,t)$ is decreasing in t for fixed $p \in (0,1)$ and $y \geq 1$. Combining this with the fact that $h_2(y,0) = 0$ for all $y \geq 1$ yields that $f_2(y,t) \geq 0$ for all $t \in (0,1)$ and $y \geq 1$, since its denominator is $t \ln(1-t)$, which is negative. As before, note that $f_2(y,\cdot)$ can be extended by continuity to $[0,1]$. This means we can apply Lemma 3.1 and obtain the desired result.

3.1.1.2 Proof of Lemma 3.3

In this section, we prove Lemma 3.3, namely that

$$\left| yh^{(e)}(p) - g(y) - \frac{1}{2}\ln y \right| = O(1)$$

when $y \to \infty$ for every $p \in (0,1)$.

In order to show this lemma, we first prove two intermediate results.

Lemma 3.4. *We have*

$$|\Lambda_1(y) - \ln\Gamma(y(1-p))| = O(1).$$

Proof. Using Lemma 2.7, we have

$$\ln\Gamma(y(1-p)) = \int_0^1 \frac{1+t-ty(1-p)-(1-t)^{y(1-p)-1}}{t\ln(1-t)}dt.$$

Recalling the definition of $\Lambda_1(y)$ in (3.14), it follows that

$$\Lambda_1(y) - \ln\Gamma(y(1-p)) = \int_0^1 \frac{(1-t)^{y(1-p)} - \left(\frac{1-t}{1-pt}\right)^y}{t(1-t)\ln(1-t)}dt < 0.$$

First, we observe that

$$\left|\int_{1/2}^1 \frac{(1-t)^{y(1-p)} - \left(\frac{1-t}{1-pt}\right)^y}{t(1-t)\ln(1-t)}dt\right| \to 0$$

when $y \to \infty$. As a result, it suffices to show that

$$\left|\int_0^{1/2} \frac{(1-t)^{y(1-p)} - \left(\frac{1-t}{1-pt}\right)^y}{t\ln(1-t)}dt\right| = \int_0^{1/2} \frac{(1-t)^{y(1-p)} - \left(\frac{1-t}{1-pt}\right)^y}{-t\ln(1-t)}dt = O(1) \tag{3.17}$$

since $1/2 \leq 1-t \leq 1$ for $t \leq 1/2$.

We follow an approach suggested by Pinelis [170]. Define $a_1(t) = \ln\left(\frac{1-t}{1-pt}\right)$, $a_2(t) = (1-p)\ln(1-t)$, and $\alpha(t) = a_2(t) - a_1(t) > 0$. Observe that we can rewrite the left-hand side of (3.17) as

$$\int_0^{1/2} \frac{e^{a_2(t)y} - e^{a_1(t)y}}{-t\ln(1-t)}dt.$$

Then, we have

$$\begin{aligned}
\int_0^{1/2} \frac{e^{a_2(t)y} - e^{a_1(t)y}}{-t\ln(1-t)}dt &\leq \int_0^{1/2} \frac{\alpha(t)ye^{a_2(t)y}}{-t\ln(1-t)}dt \\
&\leq 2(1-p)\int_0^{1/2} ye^{a_2(t)y}dt \\
&\leq 2(1-p)\int_0^{1/2} ye^{-(1-p)ty}dt \\
&\leq 2(1-p)\int_0^{\infty} ye^{-(1-p)ty}dt \\
&= 2.
\end{aligned}$$

The first inequality follows from the fact that

$$e^{by} - e^{ay} < (b-a)ye^{by} \tag{3.18}$$

if $y \geq 0$ and $b > a$ (recall that $a_2(t) > a_1(t)$ for all $t > 0$). The second inequality follows because

$$\begin{aligned}\alpha(t) &= (1-p)\ln(1-t) + \ln\left(\frac{1-pt}{1-t}\right) \\ &= (1-p)\ln(1-t) + \ln\left(1 + \frac{(1-p)t}{1-t}\right) \\ &\leq -(1-p)t + \frac{(1-p)t}{1-t} \\ &= \frac{(1-p)t^2}{1-t} \\ &\leq 2(1-p)t^2\end{aligned}$$

for all $t \in [0, 1/2]$ and $p \in (0,1)$ using $\ln(1+z) \leq z$ valid for $z > -1$. The third inequality stems also from the fact that $\ln(1+z) \leq z$. The fourth inequality holds because the function inside the integral is positive. It follows that (3.17) holds, as desired. □

Lemma 3.5. *We have*

$$|\Lambda_2(y) - \ln\Gamma(1+yp)| = O(1).$$

Proof. The reasoning we use is similar to the proof of Lemma 3.4. Using Lemma 2.7, we have

$$\Lambda_2(y) - \ln\Gamma(1+yp) = \int_0^1 \frac{(1-t)^{yp} - \left(\frac{1}{1+pt}\right)^y}{t\ln(1-t)} > 0.$$

Observe that

$$\left|\int_{1/2}^1 \frac{(1-t)^{yp} - \left(\frac{1}{1+pt}\right)^y}{t\ln(1-t)}\right| \to 0$$

when $y \to \infty$. Therefore, it remains to show that

$$\left|\int_0^{1/2} \frac{(1-t)^{yp} - \left(\frac{1}{1+pt}\right)^y}{t\ln(1-t)}\right| = \int_0^{1/2} \frac{(1-t)^{yp} - \left(\frac{1}{1+pt}\right)^y}{t\ln(1-t)} = O(1).$$

Defining $b_1(t) = p\ln(1-t)$, $b_2(t) = -\ln(1+pt)$ and $\beta(t) = b_2(t) - b_1(t) > 0$, we must show that

$$\int_0^{1/2} \frac{e^{b_2(t)y} - e^{b_1(t)y}}{-t\ln(1-t)} = O(1).$$

Proceeding analogously to the proof of the previous claim following Pinelis [170], we have

$$\int_0^{1/2} \frac{e^{b_2(t)y} - e^{b_1(t)y}}{-t\ln(1-t)}dt \leq \int_0^{1/2} \frac{\beta(t)ye^{b_2(t)y}}{-t\ln(1-t)}dt$$

$$
\begin{aligned}
&\leq p(2+p)\int_0^{1/2} ye^{b_2(t)y}dt\\
&\leq p(2+p)\int_0^{1/2} ye^{-\frac{2pt}{2+pt}y}dt\\
&\leq p(2+p)\int_0^{\infty} ye^{-\frac{2pt}{3}y}dt\\
&=\frac{3(2+p)}{2},
\end{aligned}
$$

where the first inequality follows from (3.18), the second inequality holds because $\beta(t) \leq \frac{pt}{1-t} - \frac{2pt}{2+pt} = \frac{p(2+p)t^2}{(1-t)(2+pt)} \leq p(2+p)t^2$ when $t \in [0, 1/2]$ using $-p\ln(1-t) \leq \frac{pt}{1-t}$ and $-\ln(1+pt) \leq -\frac{2pt}{2+pt}$, and the third inequality follows again from the fact that $b_2(t) = -\ln(1+pt) \leq -\frac{2pt}{2+pt}$. □

We are now ready to prove Lemma 3.3.

Proof of Lemma 3.3. We make use of the asymptotic expansion of the log-gamma function from Lemma 2.8. Taking into account (3.15), we can apply Lemma 2.8 to $\ln[(y-1)!] = \ln\Gamma(y)$, $\Lambda_1(y)$, and $\Lambda_2(y)$ and invoke Lemmas 3.4 and 3.5 to obtain

$$
\begin{aligned}
\ln\Gamma(y) &= y\ln y - y - \frac{1}{2}\ln y \pm O(1),\\
\Lambda_1(y) &= y(1-p)\ln y + y(1-p)\ln(1-p) - y(1-p) - \frac{1}{2}\ln y \pm O(1),\\
\Lambda_2(y) &= (1+yp)\ln(1+yp) - (1+yp) - \frac{1}{2}\ln(1+yp) \pm O(1)\\
&= yp\ln y + yp\ln p - yp + \frac{1}{2}\ln y \pm O(1),
\end{aligned}
$$

where in the last equality we have used the fact that $\ln(1+yp) = \ln(yp) + O\left(\frac{1}{yp}\right)$ when $y \to \infty$. As a result, we have

$$
\begin{aligned}
g(y) &= -yp\ln p - y(1-p)\ln(1-p) - \frac{1}{2}\ln y \pm O(1)\\
&= yh^{(e)}(p) - \frac{1}{2}\ln y \pm O(1).
\end{aligned}
$$

□

3.1.2 Bounds for the geometric sticky channel

In this section, we derive an analytical capacity upper bound for the geometric sticky channel by combining the family of distributions $Y^{(q)}$ designed in Section 3.1.1 with Theorems 2.5 and 2.6, and compare it to the known numerical bounds from [32, 1]. In order to apply Theorem 2.6, note that $\lambda = \mathbb{E}[R] = 1/(1-p)$. Then, the bound follows from (3.16).

Corollary 3.1. *For every $p \in (0,1)$, we have*

$$\begin{aligned}\mathsf{Cap}(R) &\leq \sup_{\mu \geq 1/(1-p)} \frac{\inf_{q\in(0,1)}(-\ln y_0 - \mu \ln q)}{\mu(1-p)} && (3.19)\\ &\leq \sup_{\substack{q\in(0,1):\\ \mathbb{E}[Y^{(q)}]\geq 1/(1-p)}} \frac{-\ln y_0 - \mathbb{E}[Y^{(q)}]\ln q}{\mathbb{E}[Y^{(q)}](1-p)}, && (3.20)\end{aligned}$$

where

$$\begin{aligned}1/y_0 &= \sum_{y=1}^{\infty} q^y e^{\ln[(y-1)!]-\Lambda_1(y)-\Lambda_2(y)-yh^{(e)}(p)} < \infty, && (3.21)\\ \mathbb{E}[Y^{(q)}] &= \sum_{y=1}^{\infty} y \cdot y_0 \cdot q^y e^{\ln[(y-1)!]-\Lambda_1(y)-\Lambda_2(y)-yh^{(e)}(p)}, && (3.22)\end{aligned}$$

with Λ_1 and Λ_2 defined as in (3.14)*.*

We remark that (3.20) is obtained by choosing, for each $\mu \geq 1/(1-p) > 1$, the value of $q \in (0,1)$ such that $\mathbb{E}[Y^{(q)}] = \mu$. Lemma 3.3 ensures that such q always exists for every $\mu > 1$. A proof of this fact can be found in Section 3.1.3.

Table 3.1 compares the results obtained via the analytical capacity upper bound (3.20) with the numerical bounds from [1]. As discussed in Section 2.5.1.1, the upper bound from [1] is obtained from an application of the Jimbo-Kunisawa algorithm coupled with numerically solving a finite optimisation problem. In contrast, our bound is obtained from explicit candidate distributions $Y^{(q)}$, and consists in maximising an analytic function over $(0,1)$ which can be easily approximated to the desired level of accuracy. Notably, this function appears to be concave for all values of p considered, and we envision that such a bound will lead to a better computer-unaided understanding of the capacity of the geometric sticky channel via further study of the normalising factor and expected value of $Y^{(q)}$.

Approximating the values of $1/y_0$ and $\mathbb{E}[Y^{(q)}]$ to the desired accuracy can be done by computing the sums in (3.21) and (3.22) for a large enough number of terms depending on q. Lemma 3.3 justifies this by ensuring that the terms in these sums decay exponentially fast. This also means the number of terms we must consider is not a large function of q. Each term in the sum requires approximating $\Lambda_1(y)$ and $\Lambda_2(y)$, but this is accomplished by standard numerical integration procedures. Combining

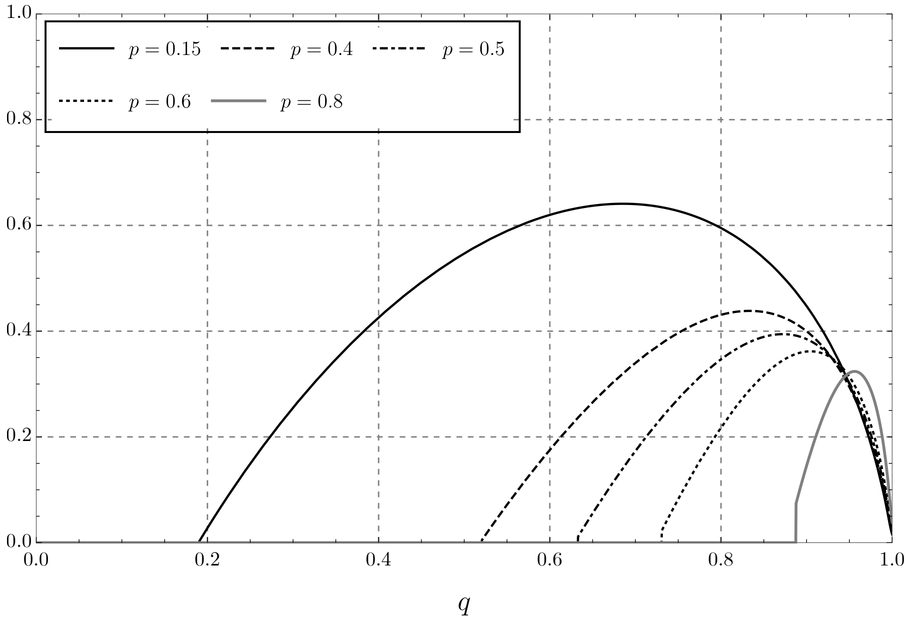

Figure 3.1: Function inside the supremum in (3.20) for some values of p, normalised by $\ln 2$. The region where the function is zero corresponds to the cases where $\mathbb{E}[Y^{(q)}] < \frac{1}{1-p}$. Adapted from [2]. ©2019 IEEE.

these observations with the experimentally observed concavity of the function inside the supremum in its positive region (see Figure 3.1) ensures that numerically computing the maximum over $(0, 1)$ for a given p is tractable.

Figure 3.2 plots the numerical capacity upper bound from [1] and the analytical upper bound (3.20). Table 3.1 and Figure 3.2 present capacity bounds in bits/channel use. We see that for $p \leq 0.4$ we are off the numerical upper bound by less than 10^{-6}. The error for $p \leq 0.5$ is still less than 10^{-5}. This shows that our analytical bound is tight whenever $p \leq 0.5$. We also improve over the numerical upper bound for $p = 0.15$. However, the bound degrades when p is large; When $p = 0.85$, the difference between the analytical and numerical bound is approximately 0.0117. For $p \geq 0.9$, the bound increases.

As an aside, it is interesting to note that the function inside the supremum in (3.20), call it $f(p, q)$, approaches 0 as $q \to 1$ for any $p \in (0, 1)$. To see this, observe that

$$f(p,q) = \frac{-\ln y_0 - \mathbb{E}[Y^{(q)}] \ln q}{\mathbb{E}[Y^{(q)}](1-p)} = \frac{-\ln y_0}{\mathbb{E}[Y^{(q)}](1-p)} - \frac{\ln q}{1-p}.$$

First, we have $\ln q \to 0$ when $q \to 1$, and we deal with the remaining term on the right-hand side. Recalling the definition of $Y^{(q)}$ and Lemma 3.3, which states that

$$\left| g(y) - y h^{(e)}(p) - \frac{1}{2} \log y \right| \leq c_{0p}$$

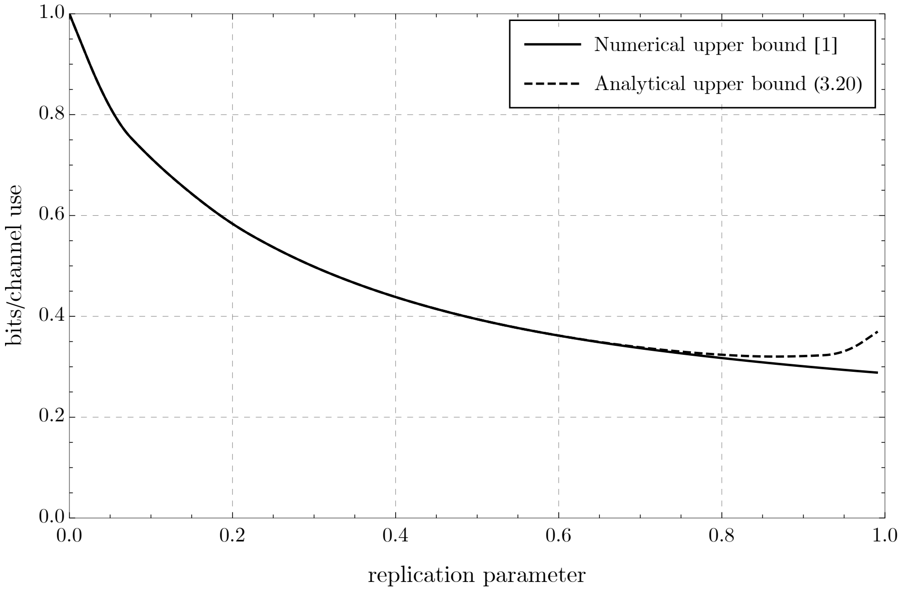

Figure 3.2: Plot of the numerical capacity upper bound from [1] for the geometric sticky channel and the analytical capacity upper bound (3.20). Adapted from [2]. ©2019 IEEE

for all $y \geq 1$ and some constant c_{0p}, we have

$$\frac{-\ln y_0}{\mathbb{E}[Y^{(q)}](1-p)} \leq \frac{c_{1p} \cdot F(q)(\ln(F(q)) + c_{1p})}{G(q)} \tag{3.23}$$

for some constant $c_{1p} > 0$, where $F(q) = \sum_{k=1}^{\infty} \frac{q^k}{\sqrt{k}}$ and $G(q) = \sum_{k=1}^{\infty} q^k\sqrt{k}$. To see that the right-hand side of (3.23) approaches 0 as $q \to 1$, observe that $\frac{F(q)}{G(q)} \to 0$ in this case, and that $F(q) \leq \frac{1}{1-q}$ and

$$G(q) \geq q^{N_q}\sqrt{N_q} \cdot \sum_{k=0}^{\infty} q^k = \frac{q^{N_q}\sqrt{N_q}}{1-q}$$

for all $q \in (0,1)$, where we set $N_q = \lceil |\ln(1-q)^3| \rceil$. Therefore, we have

$$\frac{F(q)\ln F(q)}{G(q)} \leq \frac{1}{q^{N_q}|\ln(1-q)|^{1/2}} \to 0$$

when $q \to 1$, since $q^{N_q} \to 1$ by the choice of N_q.

3.1.3 Proof of (3.20)

In this section, we show that for any $\mu > 1$ there exists $q \in (0,1)$ such that $\mathbb{E}[Y^{(q)}] = \mu$ for $Y^{(q)}$ of the form

$$Y^{(q)}(y) = y_0 q^y \exp(g(y) - yh^{(e)}(p)), \quad y = 1, 2, \ldots$$

p	Lower bound [1]	Upper bound [1]	Upper bound (3.20)
0.05	0.814457	0.814464	0.814464
0.10	0.714096	0.714114	0.714115
0.15	0.640901	0.643267	**0.640930**
0.20	0.583575	0.583611	0.583611
0.25	0.537038	0.537076	0.537076
0.30	0.498427	0.498463	0.498463
0.35	0.465925	0.465957	0.465957
0.40	0.438291	0.438318	0.438318
0.45	0.414637	0.414659	0.414660
0.50	0.394311	0.394331	0.394333
0.55	0.376821	0.376849	0.376855
0.60	0.361775	0.361794	0.361875
0.65	0.348491	0.348575	0.349152
0.70	0.336593	0.336946	0.338551
0.75	0.325900	0.326678	0.330062
0.80	0.316257	0.317317	0.323856
0.85	0.307560	0.308767	0.320448
0.90	0.299601	0.300952	0.321210
0.95	0.292373	0.293788	0.330824
0.99	0.287036	0.288476	0.368459

Table 3.1: Comparison between the numerical capacity bounds for the geometric sticky channel from [1] and the upper bound (3.20) in bits/channel use. Reproduced from [2]. ©2019 IEEE

with $g(y)$ defined as in (3.15). This suffices to justify (3.20), which is obtained from (3.19) by choosing, for each $\mu \geq \frac{1}{1-p} > 1$, the value of $q \in (0,1)$ such that $\mathbb{E}[Y^{(q)}] = \mu$.

We begin by recalling that Lemma 3.3 implies that

$$\exp(g(y) - yh^{(e)}(p)) = \Theta(1/\sqrt{y})$$

when $y \to \infty$, where the hidden constants depend only on p. Note also that $g(y)$ is finite for all integers $y \geq 1$. This can be seen by observing that

$$g(y) = \ln[(y-1)!] - \Lambda_1(y) - \Lambda_2(y)$$

where $\Lambda_1(y)$ and $\Lambda_2(y)$ are integrals over $(0,1)$ of functions which are continuous on $(0,1)$ and can be extended by continuity to $[0,1]$, and hence the integrals are finite. Combining these two observations implies that there exist constants $C_L, C_U > 0$ depending only on p such that

$$C_L/\sqrt{y} \leq \exp(g(y) - yh^{(e)}(p)) \leq C_U/\sqrt{y} \tag{3.24}$$

for all integers $y \geq 1$.

We claim that it is enough to show that

$$\lim_{q \to 0^+} \mathbb{E}[Y^{(q)}] = 1 \tag{3.25}$$

and

$$\lim_{q \to 1^-} \mathbb{E}[Y^{(q)}] = \infty. \tag{3.26}$$

This is the case because $\mathbb{E}[Y^{(q)}]$ is a continuous function of $q \in (0,1)$, and the main result then follows from the intermediate value theorem. To see the continuity of $\mathbb{E}[Y^{(q)}]$, it suffices to note that both $1/y_0$ and $\mathbb{E}[Y^{(q)}]/y_0$ are power series of q which converge to a positive value when $q \in (0,1)$, and hence are continuous on $(0,1)$.

We now argue that (3.25) holds. First, note that $\mathbb{E}[Y^{(q)}] \geq 1$ since $Y^{(q)}$ only takes values in $\mathbb{N}$. As a result, it suffices to observe that

$$\begin{aligned}
\mathbb{E}[Y^{(q)}] &= \frac{q \exp(g(1) - h^{(e)}(p))}{\sum_{y=1}^{\infty} q^y \exp(g(y) - yh^{(e)}(p))} + \frac{\sum_{y=2}^{\infty} yq^y \exp(g(y) - yh^{(e)}(p)}{\sum_{y=1}^{\infty} q^y \exp(g(y) - yh^{(e)}(p))} \\
&\leq 1 + \frac{\sum_{y=2}^{\infty} yq^y \exp(g(y) - yh^{(e)}(p))}{q \exp(g(1) - h^{(e)}(p))} \\
&\leq 1 + C_U \frac{\sum_{y=2}^{\infty} \sqrt{y} q^y}{q \exp(g(1) - h^{(e)}(p))} \\
&\leq 1 + C_U \frac{\sum_{y=2}^{\infty} yq^y}{q \exp(g(1) - h^{(e)}(p))} \\
&= 1 + C_U \frac{q(2-q)}{(1-q)^2 \exp(g(1) - h^{(e)}(p))} \to 1
\end{aligned}$$

when $q \to 0^+$, where the second inequality follows from (3.24).

It remains to show (3.26). Fix an integer $k \geq 1$. Then, we have

$$\begin{aligned}
\mathbb{E}[Y^{(q)}] &\geq \frac{C_L \sum_{y=1}^{\infty} \sqrt{y} q^y}{C_U \sum_{y=1}^{\infty} q^y / \sqrt{y}} \\
&\geq \frac{C_L \sqrt{k} \sum_{y=k}^{\infty} q^y}{C_U \sum_{y=1}^{\infty} q^y} \\
&= \frac{C_L \sqrt{k} q^{k-1}}{C_U},
\end{aligned}$$

where the first inequality follows from (3.24). It follows that for q sufficiently close to 1 we have

$$\mathbb{E}[Y^{(q)}] \geq \frac{C_L\sqrt{k}}{2C_U}.$$

Since $k \geq 1$ is an arbitrary integer and C_L, C_U are constants, this implies (3.26), which concludes the proof.

3.2 Analytical capacity upper bounds for the geometric deletion channel

In the previous section, we studied the capacity of the geometric sticky channel, which replicates each input bit according to a $\mathsf{Geom}_{1,p}$ distribution, and obtained sharp analytical capacity upper bounds for this channel from distributions with zero KL-gap for Theorem 2.5. It is then natural to consider the effect of combining deletions with geometric replications of input bits. This leads us to consider what we call the *geometric deletion channel*, which independently replicates each input bit according to a $\mathsf{Geom}_{0,p}$ distribution. More precisely, the geometric deletion channel with replication parameter p is the repeat channel with replication rule R_0 satisfying

$$R_0(y) = (1-p)p^y, \quad y = 0, 1, 2, \dots$$

Given the above, we may also call $d = 1 - p = R_0(0)$ the deletion probability of the geometric deletion channel.

Our goal in this section is to study the capacity of the geometric deletion channel via Theorems 2.7 and 2.5. Due to the memoryless property of the geometric distribution R_0, we have $\overline{R_0} \sim 1 + R_0$, where $\overline{R_0}$ corresponds to $(R_0 | R_0 \neq 0)$. This means that the DMC Ch'_{R_0} which on input $x \in \mathbb{N}$ outputs

$$Y'_x = \overline{R_0} + \sum_{i=1}^{x-1} R_{0i},$$

where $\overline{R_0}$ and the R_{0i} are independent and the R_{0i} are i.i.d. according to R_0 can be equivalently cast as the DMC that on input $x \in \mathbb{N}$ outputs

$$Y'_x \sim 1 + \mathsf{NB}_{x,p},$$

since $\mathsf{NB}_{x,p}$ is the sum of x i.i.d. $\mathsf{Geom}_{0,p}$ random variables.

We are now ready to apply Theorem 2.7 to the geometric deletion channel. Observing that we have $\lambda = \mathbb{E}[R_0] = \frac{p}{1-p}$ and $\overline{\lambda} = \mathbb{E}[\overline{R_0}] = 1 + \mathbb{E}[R_0] = \frac{1}{1-p}$, it holds that

$$\mathsf{Cap}(R_0) \leq \sup_{\mu \geq \frac{1}{1-p}} \frac{\mathsf{Cap}_\mu(\mathsf{Ch}'_{R_0})}{1/p + (\mu - \overline{\lambda})/\lambda} = \frac{p}{1-p} \cdot \sup_{\mu \geq \frac{1}{1-p}} \frac{\mathsf{Cap}_\mu(\mathsf{Ch}'_{R_0})}{\mu}.$$

We can make one further simplification to the expression above. Recalling that Ch'_{R_0} outputs $Y'_x = 1 + \mathsf{NB}_{x,p}$, the μ-limited capacity of Ch'_{R_0} is equal to the $(\mu-1)$-limited capacity of the channel Ch_{R_0} which on input $x \in \mathbb{N}$ outputs $\mathsf{NB}_{x,p}$. In symbols, we have

$$\mathsf{Cap}_\mu(\mathsf{Ch}'_{R_0}) = \mathsf{Cap}_{\mu-1}(\mathsf{Ch}_{R_0}).$$

This follows from the fact that the decoder can perfectly simulate the output of Ch_{R_0} from the output of Ch'_{R_0} and vice-versa (by subtracting/adding 1). As a result, we have

$$\mathsf{Cap}(R_0) \leq \frac{p}{1-p} \cdot \sup_{\mu \geq \frac{1}{1-p}} \frac{\mathsf{Cap}_{\mu-1}(\mathsf{Ch}_{R_0})}{\mu}. \tag{3.27}$$

With the upper bound from (3.27) in mind, our goal is now to study $\mathsf{Cap}_\mu(\mathsf{Ch}_{R_0})$ via Theorem 2.5. In this section, we obtain several upper bounds on this mean-limited capacity which lead to good capacity upper bounds for the geometric deletion channel in different regimes, and we also uncover surprising connections between candidate distributions originally designed to obtain capacity upper bounds for the deletion channel and new distributions designed for the geometric deletion channel. Our bounds are obtained by developing and applying techniques that significantly improve upon the bounds given by the approach of [40]. We proceed in steps:

1. We begin by showing in Sections 3.2.1 and 3.2.2 that approaches analogous to the ones used in [40] to obtain capacity upper bounds for the deletion and Poisson-repeat channels can also be used to design candidate distributions Y for Theorem 2.5 applied to Ch_{R_0} satisfying

$$D^{(e)}_{\mathsf{KL}}(Y_x \| Y) = a\mathbb{E}[Y_x] + b - \Delta(x) \tag{3.28}$$

for all $x \in \mathbb{N}$, where $\Delta(x) \geq 0$ is the KL-gap to the line $a\mathbb{E}[Y_x] + b$ at x (and generally $\Delta(x) > 0$). We remark that such approaches were developed in [40] specifically because one was unable to design distributions Y with zero KL-gap for channels with deletions. We contrast this with Section 3.1, where we were able to successfully design such zero KL-gap distributions for the geometric sticky channel.

2. We observe that, unlike [40], here we only need to consider $D^{(e)}_{\mathsf{KL}}(Y_x\|Y)$ for $x \geq 1$. Because of the structure of the deletion and Poisson-repeat channels, Cheraghchi [40] was forced to use a generally weaker result than Theorem 2.7 where one must upper bound the mean-limited capacity of the DMC extending Ch_R to $x \in \{0, 1, 2, \dots\}$ with $Y_0 = 0$ (namely [40, Corollary 5]). Although this appears to be an innocent modification, it has deep consequences. Taking into account (3.28), for the distributions designed in [40] one has

$$D^{(e)}_{\mathsf{KL}}(Y_0\|Y) = b,$$

and hence $\Delta(0) = 0$. As a result, it is not clear how to improve the upper bound $a\mu + b$ for the related DMC with input $x \in \{0, 1, 2, \dots\}$. However, in our case, we do not care about satisfying the inequality at $x = 0$. Moreover, we experimentally observe that $\Delta(x) \geq \Delta$ with Δ significantly larger than 0 for all $x \in \mathbb{N}$, which leads to the significantly improved upper bound $a\mu + b - \Delta$ on the mean-limited capacity of Ch_{R_0}. However, we do not stop here, and show that we can do much better.

3. Taking again into account that we only need to satisfy the inequality

$$D^{(e)}_{\mathsf{KL}}(Y_x\|Y) \leq a\mathbb{E}[Y_x] + b$$

for $x \geq 1$, we develop a general technique in Section 3.2.3 that can be used to transform distributions Y designed in Step 1 above into new distributions Y_δ, parameterised by $\delta \in (0, 1)$, which yield even better upper bounds on $\mathsf{Cap}_\mu(\mathsf{Ch}_{R_0})$ than simply applying the observation from Step 2 above to Y. We showcase explicit values of δ which appear to lead to significantly improved upper bounds. Surprisingly, this general technique also shows that a family of distributions originally designed to obtain capacity upper bounds for the deletion channel in [40], called *inverse binomial distributions*, are feasible candidates for the geometric deletion channel too.

4. Finally, in Section 3.2.5 we focus on what we call the *large replication regime*, which corresponds to the asymptotic regime where $p \to 1$. Equivalently, this is the regime where the deletion probability d approaches 0. For the deletion channel and the Poisson-repeat channel, two well-

studied repeat channels, it can be shown that their capacity approaches 1 as the probability of deletion approaches 0.[2] On the other hand, numerical evidence suggests that the capacity of the geometric deletion channel is much smaller than 1 when $p \to 1$. Given this, we would like to derive non-trivial properties of $\mathsf{Cap}(R_0)$ in this regime *without computer assistance.* We use our techniques from Step 3 to give a proof, without computer assistance, that $\mathsf{Cap}(R_0) < 0.73$ bits/channel use when $p \to 1$.

The remaining material in this section is a reproduction of [2, Section V], with some modifications to improve exposition and consistency with the theme of the thesis.

3.2.1 A bound via convexity

In this section, we obtain a capacity upper bound for the negative binomial channel by following a reasoning similar to the one used to derive capacity upper bounds for the deletion channel in [40]. We will later show how this bound can be improved. For convenience, we define $d = 1 - p$.

As previously observed, we can write

$$D_{\mathsf{KL}}^{(e)}(Y_x||Y) = \sum_{y=0}^{\infty} Y_x(y) \ln\left(\frac{Y_x(y)}{Y(y)}\right) = -H^{(e)}(Y_x) - \sum_{y=0}^{\infty} Y_x(y) \ln Y(y).$$

Furthermore, recalling that $Y_x \sim \mathsf{NB}_{x,p}$ and $\mathbb{E}[Y_x] = \frac{xp}{1-p}$, we have

$$\begin{aligned} -H^{(e)}(Y_x) &= \sum_{y=0}^{\infty} Y_x(y) \ln\left(\binom{y+x-1}{y} d^x p^y\right) \\ &= \mathbb{E}\left[\ln\binom{Y_x+x-1}{Y_x}\right] + x\ln d + \mathbb{E}[Y_x]\ln p \\ &= \mathbb{E}\left[\ln\binom{Y_x+x-1}{Y_x}\right] - \mathbb{E}[Y_x]\frac{h^{(e)}(p)}{p}. \end{aligned} \tag{3.29}$$

We consider a family of distributions $Y^{(q)}$ for $q \in (0,1)$ of the form

$$Y^{(q)}(y) = y_0 \binom{g(y)}{y} q^y \exp(-y h^{(e)}(p)/p), \quad y = 0, 1, 2, \ldots$$

[2] This is trivial for the deletion channel, and can be easily shown for the Poisson-repeat channel by noting that the Poisson distribution concentrates strongly around the mean [2].

for a function g to be defined, where

$$y_0 = \left(\sum_{y=0}^{\infty} Y^{(q)}(y)/y_0 \right)^{-1}$$

is the normalising factor. Instantiating Y with $Y^{(q)}$ leads to

$$\begin{aligned} D_{\mathsf{KL}}^{(e)}(Y_x||Y^{(q)}) &= \mathbb{E}\left[\ln \binom{Y_x + x - 1}{Y_x}\right] + \mathbb{E}[Y_x]\frac{h^{(e)}(p)}{p} - \sum_{y=0}^{\infty} Y_x(y) \ln \left(y_0 \binom{g(y)}{y} q^y \exp(-yh^{(e)}(p)/p) \right) \\ &= \mathbb{E}\left[\ln \frac{\binom{Y_x+x-1}{Y_x}}{\binom{g(Y_x)}{Y_x}}\right] - \ln y_0 - \mathbb{E}[Y_x]\ln q. \end{aligned} \tag{3.30}$$

Equipped with some insight, we want to choose g such that

$$g(\mathbb{E}[Y_x]) = \mathbb{E}[Y_x] + x - 1,$$

which can be accomplished by setting $g(y) = y/p - 1$. This leads to the expression

$$Y^{(q)}(y) = y_0 \binom{y/p - 1}{y} q^y \exp(-yh^{(e)}(p)/p). \tag{3.31}$$

The fact that $Y^{(q)}$ is a valid distribution for all $q \in (0,1)$, i.e., $1/y_0 < \infty$, follow from the asymptotic expression for $\binom{y/p-1}{y}$ obtained via the asymptotic expansion for the log gamma function from Lemma 2.8.

Combining (3.30) and (3.31), we obtain

$$D_{\mathsf{KL}}^{(e)}(Y_x||Y^{(q)}) \leq -\varepsilon(p) - \ln y_0 - \mathbb{E}[Y_x]\ln q \tag{3.32}$$

for all $x \in \mathbb{N}$, where

$$\varepsilon(p) = \inf_{x\in\mathbb{N}} \mathbb{E}\left[\ln \frac{\binom{Y_x/p-1}{Y_x}}{\binom{Y_x+x-1}{Y_x}}\right].$$

The next lemma shows we can always replace $\varepsilon(p)$ by 0 in (3.32) to obtain a valid upper bound.

Lemma 3.6. *We have $\varepsilon(p) \geq 0$ for all $p \in (0,1)$.*

Proof. See Section 3.2.1.1. □

While Lemma 3.6 implies that we can replace $\varepsilon(p)$ by 0 in (3.32), it turns out that $\varepsilon(p)$ is actually significantly larger than zero for most values of p, and so keeping it in (3.32) leads to improved capacity upper bounds for the negative binomial channel. We are now in a position to apply Theorem 2.5 using (3.32).

Theorem 3.1. *We have*

$$\begin{aligned}\mathsf{Cap}_\mu(R_0) &\leq -\varepsilon(p) + \inf_{q\in(0,1)}(-\ln y_0 - \mu \ln q)\\ &\leq \inf_{q\in(0,1)}(-\ln y_0 - \mu \ln q).\end{aligned}$$

Interestingly, $Y^{(q)}$ is closely related to the *inverse binomial distribution* defined in [40] to obtain capacity upper bounds for the deletion channel. For given $p, q \in (0,1)$, we denote the inverse binomial distribution by $\mathsf{InvBin}_{p,q}$. It satisfies

$$\mathsf{InvBin}_{p,q}(y) = y_{\mathsf{IB}}\binom{y/p}{y} q^y \exp(-yh^{(e)}(p)/p), \quad y = 0, 1, \dots,$$

where y_{IB} is the normalising factor. Using the equality

$$\binom{y/p-1}{y} = d\binom{y/p}{y}$$

valid for all $y > 0$ and $p \in (0,1)$ and recalling (3.31), we conclude that

$$\frac{Y^{(q)}(y)}{y_0} = d \cdot \frac{\mathsf{InvBin}_{p,q}(y)}{y_{\mathsf{IB}}} \tag{3.33}$$

for all $y \geq 1$. This property of $Y^{(q)}$ will prove useful in the following sections, as there exist sharp bounds for the normalising factor and expected value of $\mathsf{InvBin}_{p,q}$ in terms of both special and elementary functions [40, Sections 6.1.1 and 6.1.2]. In particular, we use such a bound in the proof that the capacity of the geometric deletion channel is bounded well away from 1 when $p \to 1$ in Section 3.2.5.

It is also relevant to study the behaviour of the KL-gap

$$\Delta(x) = \mathbb{E}\left[\ln \frac{\binom{Y_x/p-1}{Y_x}}{\binom{Y_x+x-1}{Y_x}}\right].$$

The following lemma characterises the asymptotic behaviour of $\Delta(x)$ for large x. In particular, it shows that $\Delta(x) \to 1/2$, a fact which will be useful in Section 3.2.3.

Lemma 3.7. *We have*

$$\Delta(x) \geq 1/2 - \frac{4}{3p(x-1)} - \frac{(2-p)^2}{12(1-p)x} - O(1/x^2)$$

for $x \geq 2$, and

$$\Delta(x) \leq 1/2 + O(x^{-1/2+\beta})$$

for all $p \in (0,1)$ and $\beta > 0$.

Proof. See Section 3.2.1.2. □

3.2.1.1 Proof of Lemma 3.6

In this section, we prove Lemma 3.6. We first present an auxiliary result from [171] that will be useful.

Lemma 3.8 ([171, Lemma 1, specialised])**.** *Consider the function $f : (0,\infty) \to \mathbb{R}$ satisfying*

$$f(y) = \ln\left(\frac{\prod_{i=1}^{k_1}\Gamma(A_i y + a_i)}{\prod_{j=1}^{k_2}\Gamma(B_j y + b_j)}\right),$$

where $A_i, B_j > 0$ and $a_i, b_j \geq 0$ for all i and j. Then, f is convex on $(0,\infty)$ provided that

$$\sum_{i=1}^{k_1} \frac{\exp(-a_i u/A_i)}{1-\exp(-u/A_i)} - \sum_{j=1}^{k_2} \frac{\exp(-b_j u/B_j)}{1-\exp(-u/B_j)} \geq 0$$

for all $u > 0$.

We are now ready to prove that $\varepsilon(p) \geq 0$ for all $p \in (0,1)$. We show that $f_x(y) = \ln\left[\binom{y/p-1}{y}/\binom{y+x-1}{y}\right]$ is convex on $[0,\infty)$ for all $x \in \mathbb{N}$. This implies the desired result via Jensen's inequality, since then we have

$$\mathbb{E}[f_x(Y_x)] \geq f_x(\mathbb{E}[Y_x]) = \ln 1 = 0$$

for all $x \in \mathbb{N}$, and so $\varepsilon(p) = \inf_{x\in\mathbb{N}} \mathbb{E}[f_x(Y_x)] \geq 0$.

For any $x \geq 1$ and $y > 0$ we have

$$f_x(y) = \ln\Gamma(x) + \ln\left[\frac{\Gamma(y/p)}{\Gamma(y(1/p-1))\Gamma(y+x)}\right].$$

Then, by Lemma 3.8, f_x is convex on $(0, \infty)$ if

$$P_x(u) = \frac{1}{1 - e^{-up}} - \frac{1}{1 - e^{-up/(1-p)}} - \frac{e^{-ux}}{1 - e^{-u}} \geq 0$$

for all $x \geq 1$ and $u > 0$. Note that $P_x(u) \geq P_1(u)$ for $x \geq 1$ and that $P_1(u)$ can be rewritten as

$$P_1(u) = \frac{1}{e^{up} - 1} - \frac{1}{e^{up/(1-p)} - 1} - \frac{1}{e^u - 1}$$

using the fact that $\frac{1}{1-e^{-a}} = 1 + \frac{1}{e^a - 1}$ for every $a \neq 0$. Therefore, it suffices to show that

$$\frac{1-p}{e^{up} - 1} - \frac{1}{e^{up/(1-p)} - 1} \geq 0 \tag{3.34}$$

and

$$\frac{p}{e^{up} - 1} - \frac{1}{e^u - 1} \geq 0. \tag{3.35}$$

We show (3.35) and observe that (3.34) follows in an analogous manner. Rearranging terms, we want to show that

$$p(e^u - 1) - (e^{up} - 1) \geq 0. \tag{3.36}$$

Note that the left-hand side of (3.36) is 0 at $u = 0$, and that its derivative with respect to u is $p(e^u - e^{up})$, which is positive for all $u > 0$. This yields the desired inequality.

It remains to see that f_x is convex on $[0, \infty)$. Note that $f_x(0) = 0$, since $\binom{-1}{0} = 1$ by the expression for the binomial coefficients in Section 2.2. Furthermore, using the continuity of $\Gamma(z)$ for $z > 0$ we have

$$\lim_{y \to 0^+} \ln \binom{y/p - 1}{y} = \lim_{y \to 0^+} \ln \left(d \binom{y/p}{y} \right) = \ln d$$

and

$$\lim_{y \to 0^+} \ln \binom{y + x - 1}{y} = 0.$$

This implies that $\lim_{y \to 0^+} f_x(y) = \ln d < 0$ for all $x \geq 1$. We then have $f_x(0) = 0 \geq \lim_{y \to 0^+} f_x(y)$, which shows that f_x is convex on $[0, \infty)$ (recall we had already shown it was convex on $(0, \infty)$).

3.2.1.2 Proof of Lemma 3.7

In this section, we prove Lemma 3.7, namely that

$$\Delta(x) \geq 1/2 - \frac{4}{3p(x-1)} - \frac{(1+p)^2}{12px} - O(1/x^2)$$

for $x \geq 2$, and

$$\Delta(x) \leq 1/2 + O(x^{-1/2+\beta})$$

for all $p \in (0,1)$ and $\beta > 0$.

First, we have

$$\ln \binom{y/p-1}{y} \geq y\frac{h^{(e)}(p)}{p} - \frac{1}{2}\ln y + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) - \frac{1}{12(y+1)} - \frac{p}{12(1-p)y} - \frac{1}{2y}$$

and

$$\ln \binom{y/p-1}{y} \leq y\frac{h^{(e)}(p)}{p} - \frac{1}{2}\ln y + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) + \frac{p}{12y}$$

for all $y \geq 1$. This follows from the asymptotic expansion of the log gamma function from Lemma 2.8 and the inequalities $\frac{2}{1+2y} \leq \ln(1+1/y) \leq 1/y$ valid for all $y > 0$. Therefore, it holds that

$$\ln Y^{(q)}(y) \geq \ln y_0 + y\ln q - \frac{1}{2}\ln y + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) - \frac{2}{3(1-p)y} \tag{3.37}$$

and

$$\ln Y^{(q)}(y) \leq \ln y_0 + y\ln q - \frac{1}{2}\ln y + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) + \frac{p}{12y} \tag{3.38}$$

for $y \geq 1$, and $\ln Y^{(q)}(0) = \ln y_0$.

Furthermore, sharp asymptotic expansions are also known for $H^{(e)}(Y_x)$ when $x \to \infty$. To be precise, according to [172, Section 5] we have[3]

$$H^{(e)}(Y_x) = \frac{1}{2}\ln\left(\frac{2\pi exp}{(1-p)^2}\right) - \frac{(1+p)^2}{12px} - O(1/x^2). \tag{3.39}$$

[3] General results about the asymptotic expansion of the entropy of sums of i.i.d. random variables can also be found in [173, 174].

We begin by proving the first inequality in the lemma statement. For $x \geq 2$, we have

$$\begin{aligned}
\Delta(x) &= -\ln y_0 - \mathbb{E}[Y_x]\ln q - D^{(e)}_{\mathsf{KL}}(Y_x || Y^{(q)}) \\
&= -\ln y_0 - \mathbb{E}[Y_x]\ln q + H^{(e)}(Y_x) + \sum_{y=0}^{\infty} Y_x(y)\ln Y^{(q)}(y) \\
&\geq H^{(e)}(Y_x) + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) - \frac{1}{2}\sum_{y=1}^{\infty} Y_x(y)\ln y - \frac{2}{3(1-p)}\sum_{y=1}^{\infty} Y_x(y)/y \qquad (3.40) \\
&\geq H^{(e)}(Y_x) + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) - \frac{1}{2}\ln\mathbb{E}[Y_x] - \frac{4}{3p(x-1)} - O(1/x^2) \qquad (3.41) \\
&= \frac{1}{2}\left(\ln\left(\frac{2\pi exp}{(1-p)^2}\right) - \ln\left(\frac{xp}{1-p}\right) + \ln\left(\frac{1-p}{2\pi}\right)\right) - \frac{4}{3p(x-1)} - \frac{(1+p)^2}{12px} - O(1/x^2) \qquad (3.42) \\
&= \frac{1}{2} - \frac{4}{3p(x-1)} - \frac{(1+p)^2}{12px} - O(1/x^2),
\end{aligned}$$

as desired. The inequality in (3.40) follows from (3.37). To justify (3.41), note that

$$\begin{aligned}
-\sum_{y=1}^{\infty} Y_x(y)\ln y &\geq -(1-Y_x(0))\ln\left(\frac{\mathbb{E}[Y_x]}{1-Y_x(0)}\right) \\
&\geq -\ln\mathbb{E}[Y_x] + (1-p)^x\ln\mathbb{E}[Y_x] + \ln(1-Y_x(0)) \\
&\geq -\ln\mathbb{E}[Y_x] + (1-p)^x\left(\ln\left(\frac{p}{1-p}\right) - 1/p\right)
\end{aligned}$$

by Jensen's inequality, the convexity of $-\ln y$, and the fact that $\ln(1-Y_x(0)) \geq -\frac{Y_x(0)}{1-Y_x(0)} \geq -\frac{(1-p)^x}{p}$. Moreover,

$$\begin{aligned}
\sum_{y=1}^{\infty} Y_x(y)/y &\leq 2\sum_{y=0}^{\infty}\frac{Y_x(y)}{y+1} \\
&= \frac{2(1-p)}{p(x-1)}\sum_{y=0}^{\infty}\binom{y+x-1}{y+1}(1-p)^{x-1}p^{y+1} \\
&= \frac{2(1-p)}{p(x-1)}(1-Y_{x-1}(0)) \\
&\leq \frac{2(1-p)}{p(x-1)} \qquad (3.43)
\end{aligned}$$

for $x \geq 2$. The equality in (3.42) follows from (3.39).

For the second inequality, note that

$$\Delta(x) = -\ln y_0 - \mathbb{E}[Y_x]\ln q + H^{(e)}(Y_x) + \sum_{y=0}^{\infty} Y_x(y)\ln Y^{(q)}(y)$$

$$\leq H^{(e)}(Y_x) + \frac{1-Y_x(0)}{2} \cdot \ln\left(\frac{1-p}{2\pi}\right) - \frac{1}{2}\sum_{y=1}^{\infty} Y_x(y)\ln y + \frac{p}{12}\sum_{y=1}^{\infty} Y_x(y)/y \tag{3.44}$$

$$\leq H^{(e)}(Y_x) + \frac{1}{2}\ln\left(\frac{1-p}{2\pi}\right) - \frac{1}{2}\ln \mathbb{E}[Y_x] + O(x^{-1/2+\beta}) \tag{3.45}$$

$$= \frac{1}{2} + O(x^{-1/2+\beta}). \tag{3.46}$$

The inequality in (3.44) follows from (3.38). The inequality in (3.45) holds because of (3.43) and the fact that

$$\sum_{y=1}^{\infty} Y_x(y)\ln y \geq \ln \mathbb{E}[Y_x] - O(x^{-1/2+\beta}),$$

which follows from the concentration bound for the negative binomial from Lemma 2.3. Finally, (3.46) follows from (3.39).

3.2.2 A bound via truncation

In this section, we design a distribution whose KL-gap converges to 0 exponentially fast as x increases. The process will be similar to that of Section 3.1.1, and we will reutilise some arguments. As was the case for the deletion and Poisson-repeat channels in [40, Sections 5 and 6], in this case we cannot ensure that the KL-gap is zero everywhere (in fact, it will be positive everywhere).

We consider a family of distributions $\overline{Y}^{(q)}$ for $q \in (0,1)$ of the form

$$\overline{Y}^{(q)}(y) = \overline{y_0} q^y \exp(g(y) - yh^{(e)}(p)/p), \quad y = 0, 1, 2, \ldots \tag{3.47}$$

for some function g to be determined, where $\overline{y_0}$ is the normalising factor. Recalling that $Y_x \sim \mathsf{NB}_{x,p}$ and (3.29), we want g to satisfy

$$\mathbb{E}[g(Y_x)] = \mathbb{E}[\ln[(Y_x + x - 1)!]] - \ln[(x-1)!] - \mathbb{E}[\ln(Y_x!)] + R_p(x),$$

where $R_p(x) \geq 0$ is an error term which vanishes exponentially fast with x. Furthermore, we want g to have moderate growth so that $\overline{Y}^{(q)}$ is a valid probability distribution.

Recalling Lemma 2.7, we have

$$\ln[(x-1)!] = \int_0^1 \frac{1 + t - tx - (1-t)^{x-1}}{t\ln(1-t)} dt$$

and[4]

$$\begin{aligned}\mathbb{E}[\ln[(Y_x+x-1)!]] &= \mathbb{E}\left[\int_0^1 \frac{1+t-t(y+x)-(1-t)^{y+x-1}}{t\ln(1-t)}dt\right]\\ &= \int_0^1 \frac{1+t-\frac{tx}{1-p}-(1-t)^{x-1}\left(\frac{1-p}{1-p(1-t)}\right)^x}{t\ln(1-t)}dt. \end{aligned} \tag{3.48}$$

Consider the functions

$$\begin{aligned} f_1(y,t) &= \frac{1+t-ty(1-p)/p-\left(\frac{p-t}{p(1-t)}\right)^y/(1-t)}{t\ln(1-t)},\\ f_2(y,t) &= \frac{1+t-ty/p-\left(\frac{p-t(1+p)}{p(1-t)}\right)^y/(1-t)}{t\ln(1-t)}. \end{aligned}$$

We would hope that $\mathbb{E}\left[\int_0^1 f_1(Y_x,t)dt\right] = \ln[(x-1)!]$ and $\mathbb{E}\left[\int_0^1 f_2(Y_x,t)dt\right] = \mathbb{E}[\ln[(Y_x+x-1)!]]$. However, this does not hold as $f_1(y,t)$ and $f_2(y,t)$ grow exponentially fast for large t. We contrast this with the geometric sticky channel in Section 3.1, where we derive a solution to the corresponding functional equation via this method and prove that it is well-defined and leads to a valid distribution.

In order to overcome the above, we truncate the integration bounds. To determine the point at which to truncate, note that $\frac{p-t(1+p)}{p(1-t)} \geq -1$ whenever $t \leq \frac{2p}{1+2p}$. Truncating at this point ensures that the exponential terms in y in the two integrals are controlled, and so the resulting function g has the desired growth. Consider the truncated integrals

$$\Lambda_1(y) = \int_0^{\frac{2p}{1+2p}} f_1(y,t)dt, \tag{3.49}$$

$$\Lambda_2(y) = \int_0^{\frac{2p}{1+2p}} f_2(y,t)dt. \tag{3.50}$$

Then, we have

$$\begin{aligned}\mathbb{E}[\Lambda_1(Y_x)] &= \int_0^{\frac{2p}{1+2p}} \mathbb{E}[f_1(Y_x,t)]dt \qquad (3.51)\\ &= \ln[(x-1)!] - \int_{\frac{2p}{1+2p}}^1 \frac{1+t-tx-(1-t)^{x-1}}{t\ln(1-t)}dt\\ &= \ln[(x-1)!] - \eta\left(\frac{1}{1+2p}\right) + (x-1)\mathrm{li}\left(\frac{1}{1+2p}\right) + \int_{\frac{2p}{1+2p}}^1 \frac{(1-t)^{x-1}}{t\ln(1-t)}dt, \end{aligned}$$

[4] Once again, switching the integral and expected value in (3.48) is allowed via Lemma 3.1, since the function inside the integral is continuous on $[0,1]$ and positive for all $y \geq 0$ and $x \geq 1$.

and

$$\begin{aligned}\mathbb{E}[\Lambda_2(Y_x)] &= \int_0^{\frac{2p}{1+2p}} \mathbb{E}[f_2(Y_x,t)]dt \\ &= \mathbb{E}[\ln[(Y_x+x-1)!]] - \int_{\frac{2p}{1+2p}}^{1} \frac{1+t-\frac{tx}{1-p}-(1-t)^{x-1}\left(\frac{1-p}{1-p(1-t)}\right)^x}{t\ln(1-t)}dt \\ &= \mathbb{E}[\ln[(Y_x+x-1)!]] - \eta\left(\frac{1}{1+2p}\right) + \left(\frac{x}{1-p}-1\right)\operatorname{li}\left(\frac{1}{1+2p}\right) \\ &+ \int_{\frac{2p}{1+2p}}^{1} \frac{(1-t)^{x-1}\left(\frac{1-p}{1-p(1-t)}\right)^x}{t\ln(1-t)}dt,\end{aligned} \tag{3.52}$$

where we recall from Section 2.2 that $\operatorname{li}(z)=\int_0^z \frac{dt}{\ln t}$ for $z<1$ is the logarithmic integral, and we define $\eta(z)=\int_0^z \frac{dt}{(1-t)\ln t}$. An analogous argument to that used in the proof of Lemma 3.2 shows that (3.51) and (3.52) hold. For the sake of completeness, we present proofs of (3.51) and (3.52) in Appendix A.

We set

$$g(y)=\Lambda_2(y)-\Lambda_1(y)-\ln(y!)-y\cdot\operatorname{li}\left(\frac{1}{1+2p}\right).$$

Note that g satisfies

$$\mathbb{E}[g(Y_x)]=\mathbb{E}[\ln[(Y_x+x-1)!]]-\ln[(x-1)!]-\mathbb{E}[\ln(Y_x!)]+R_p(x), \tag{3.53}$$

where

$$R_p(x)=-\int_{\frac{2p}{1+2p}}^{1}\frac{(1-t)^{x-1}\left(1-\left(\frac{1-p}{1-p(1-t)}\right)^x\right)}{t\ln(1-t)}dt\geq 0. \tag{3.54}$$

Observe that $R_p(x)$ vanishes exponentially fast in x.

It now remains to show that g has the correct asymptotic growth. The proof of the following result is analogous to the proof of Lemma 3.3. For the sake of completeness, we present a proof in Appendix A.3.

Lemma 3.9. *We have*

$$\begin{aligned}\Lambda_1(y) &= \ln\Gamma\left(\frac{y(1-p)}{p}\right)+\frac{y(1-p)}{p}\cdot\operatorname{li}\left(\frac{1}{1+2p}\right)-\eta\left(\frac{1}{1+2p}\right)+O(1), \\ \Lambda_2(y) &= \ln\Gamma\left(\frac{y}{p}\right)+\frac{y}{p}\cdot\operatorname{li}\left(\frac{1}{1+2p}\right)-\eta\left(\frac{1}{1+2p}\right)+O(1).\end{aligned}$$

In particular, it holds that

$$\left|y\frac{h^{(e)}(p)}{p}-g(y)-\frac{1}{2}\ln y\right|=O(1).$$

Proof. See Appendix A.3. □

Lemma 3.9 implies that $\overline{Y}^{(q)}$ is a valid distribution if $q \in (0,1)$, since then $\overline{Y}^{(q)}/\overline{y_0} = \Theta(q^y/\sqrt{y})$. It remains to upper bound $D_{\mathsf{KL}}^{(e)}(Y_x||\overline{Y}^{(q)})$. We have

$$\begin{aligned} D_{\mathsf{KL}}^{(e)}(Y_x||\overline{Y}^{(q)}) &= -H^{(e)}(Y_x) - \sum_{y=0}^{\infty} Y_x(y) \ln \overline{Y}^{(q)}(y) \\ &= -H^{(e)}(Y_x) - \ln \overline{y_0} - \mathbb{E}[Y_x] \ln q - \mathbb{E}[g(Y_x)] + \mathbb{E}[Y_x] h^{(e)}(p)/p \\ &= \mathbb{E}\left[\ln \binom{Y_x + x - 1}{Y_x}\right] - \ln \overline{y_0} - \mathbb{E}[Y_x] \ln q - \mathbb{E}[g(Y_x)] \\ &= -R_p(x) - \ln \overline{y_0} - \mathbb{E}[Y_x] \ln q \\ &\leq -\ln \overline{y_0} - \mathbb{E}[Y_x] \ln q. \end{aligned} \tag{3.55}$$

The second equality follows from (3.47), the third equality follows from (3.29), the fourth equality holds because of (3.53), and the inequality follows from the fact that $R_p(x) \geq 0$ for all $x \geq 1$. Combining (3.55) with Theorem 2.5, we immediately obtain the capacity upper bound

$$\mathsf{Cap}_\mu(R_0) \leq \inf_{q \in (0,1)} (-\ln \overline{y_0} - \mu \ln q)$$

for all $\mu \geq 0$. There are two important comments regarding this bound: First, as shown in (3.55), the gap between $D_{\mathsf{KL}}^{(e)}(Y_x||\overline{Y}^{(q)})$ and the line $-\ln \overline{y_0} - \mathbb{E}[Y_x] \ln q$ is exactly $R_p(x)$, which converges to 0 exponentially fast as x increases. Second, we have $R_p(1) \gg 0$.

3.2.3 Improving the bounds by fixing the mass at $y = 0$

In this section, we showcase a technique that can be used to significantly improve the bounds we obtain from the distributions designed in Sections 3.2.1 and 3.2.2. We will also use this technique to give a proof without computer assistance that the capacity of the geometric deletion channel is at most 0.73 bits/channel use when $p \to 1$ in Section 3.2.5. This technique will be useful in Chapter 4 as well.

The technique we present below consists in optimising the mass at $y = 0$ of any given family of distributions suitable for Theorem 2.5. This leads to an upper bound which is at least as good as the original, and, when applied to the distributions from Section 3.2.1, we see significant improvements for a large range of the replication parameter p. At a high-level, we proceed as follows:

1. We study how the Kullback-Leibler divergence $D^{(e)}_{\mathsf{KL}}(Y_x||Y)$ behaves when we change the value of $Y(0)$ to some value δ and renormalise Y. Calling the new distribution obtained in this way Y_δ, we show that there is a simple relationship between $D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta)$ and the original KL divergence $D^{(e)}_{\mathsf{KL}}(Y_x||Y)$;

2. From experience, we know that a small KL-gap leads to better capacity upper bounds. Naively, one could numerically optimise over δ to obtain a better upper bound than the one given by Y. However, adding a new layer of numerical optimisation is cumbersome. Instead, we analytically derive explicit choices of δ that significantly reduce the KL-gap under some mild assumptions, and lead to much better upper bounds. This derivation is possible because there is a simple relationship between $D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta)$ and $D^{(e)}_{\mathsf{KL}}(Y_x||Y)$;

3. We instantiate this reasoning with the distributions designed in Sections 3.2.1 and 3.2.2, and give evidence that both distributions satisfy the assumptions required for the effectiveness of our choices of δ for several values of p.

A special case of this approach was used by Martinez [47] to derive an improved capacity upper bound for the discrete-time Poisson channel. Here, we provide a general treatment of the technique for the geometric deletion channel, show that it can be used to significantly improve the capacity upper bounds originally given by the distributions designed in Sections 3.2.1 and 3.2.2, and give further applications of this technique to the derivation of a non-trivial capacity upper bound in the large replication regime.

Consider a distribution Y with support on $\{0, 1, 2, \dots\}$ and probability mass function $Y(y) = y_0 a(y)$ for some function $a(y)$ with $a(0) = 1$ and normalising factor y_0. For $\delta \in (0, 1]$, we define the modified distribution Y_δ satisfying

$$Y_\delta(y) = \begin{cases} \alpha\delta, & \text{if } y = 0 \\ \alpha a(y), & \text{if } y > 0, \end{cases}$$

where α is the new normalising factor, satisfying $1/\alpha = \delta + 1/y_0 - 1$. Intuitively, Y_δ is obtained from Y by modifying the mass of Y at $y = 0$, and setting $\delta = 1$ yields the original distribution Y.

3.2.3.1 KL-divergence and KL-gap of Y_δ

In this section, we study how $D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta)$ behaves with respect to $D^{(e)}_{\mathsf{KL}}(Y_x||Y)$, where Y_δ denotes the modified version of Y. A key point that will be useful in later sections is that $D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta)$ has a simple expression in terms of $D^{(e)}_{\mathsf{KL}}(Y_x||Y)$ for all x. In fact, letting $d = 1 - p$ and recalling that $Y_x \sim \mathsf{NB}_{x,p}$ yields

$$\begin{aligned}
D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta) &= -H^{(e)}(Y_x) - \ln\alpha - \sum_{y=1}^{\infty} Y_x(y)\ln a(y) - d^x \ln\delta \\
&= -H^{(e)}(Y_x) - \ln\alpha - \sum_{y=0}^{\infty} Y_x(y)\ln a(y) - d^x \ln\delta \\
&= -H^{(e)}(Y_x) - \ln y_0 - \sum_{y=0}^{\infty} Y_x(y)\ln a(y) + \ln y_0 - \ln\alpha - d^x \ln\delta \\
&= D^{(e)}_{\mathsf{KL}}(Y_x||Y) + \ln y_0 - \ln\alpha - d^x\ln\delta \qquad (3.56) \\
&\leq D^{(e)}_{\mathsf{KL}}(Y_x||Y) + \ln y_0 - \ln\alpha - d\ln\delta. \qquad (3.57)
\end{aligned}$$

In the first equality we used the fact that $Y_x(0) = d^x$ for all $x \geq 1$. The second equality holds because $\ln a(0) = 0$ since $a(0) = 1$. In the last equality we used that $\delta \leq 1$, and so $-d^x \ln\delta \leq -d\ln\delta$ for $x \geq 1$.

Since the KL-gap of a distribution is a good indicator for the quality of the capacity upper bound induced by that distribution via Theorem 2.5, it is instructive to study how the KL-gap changes with δ. Given the simple form of (3.56) and (3.57), we have good control of how the KL-gap changes when we transform Y into Y_δ. Suppose Y satisfies

$$D^{(e)}_{\mathsf{KL}}(Y_x||Y) \leq a\mathbb{E}[Y_x] + b \qquad (3.58)$$

for some $a, b \in \mathbb{R}$ and all $x \in \mathbb{N}$. Then, the KL-gap associated to Y and the line $a\mathbb{E}[Y_x] + b$ is

$$\Delta(x) = a\mathbb{E}[Y_x] + b - D^{(e)}_{\mathsf{KL}}(Y_x||Y) \geq 0. \qquad (3.59)$$

Combining (3.57) and (3.58), we have

$$D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta) \leq a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha - d\ln\delta. \qquad (3.60)$$

As a result, we may compute the KL-gap associated to Y_δ and the line $a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha - d\ln\delta$, denoted $\Delta_\delta(x)$, as a function of the original KL-gap $\Delta(x)$:

$$\begin{aligned}
\Delta_\delta(x) &= a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha - d\ln\delta - D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta) \\
&= \Delta(x) - d\ln\delta + d^x\ln\delta
\end{aligned}$$

$$\geq 0, \tag{3.61}$$

where the second equality follows from (3.56) and (3.59). In particular, we have $\Delta_\delta(1) = \Delta(1)$ and $\Delta_\delta(x) \geq \Delta(x)$ for all $x \in \mathbb{N}$.

Since we may have $\Delta_\delta(x) \gg 0$ simultaneously for all x, we can refine (3.60) considerably by shifting the line on the right hand side of (3.60) down by the smallest value the KL-gap $\Delta_\delta(x)$ attains, i.e., we can shift the line down by $\inf_{x\in\mathbb{N}} \Delta_\delta(x)$. Based on this, we have the refined bound

$$D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta) \leq a\mathbb{E}[Y_x] + b + \ln y_0 - \ln \alpha - d \ln \delta - \varepsilon_\delta(p), \tag{3.62}$$

where we have defined

$$\varepsilon_\delta(p) = \inf_{x\in\mathbb{N}} \Delta_\delta(x) \geq \varepsilon_1(p). \tag{3.63}$$

Taking into account (3.61), the KL-gap associated to Y_δ and the refined line on the right hand side of (3.62), which we denote by Δ'_δ, satisfies

$$\begin{aligned}\Delta'_\delta(x) &= a\mathbb{E}[Y_x] + b + \ln y_0 - \ln \alpha - d \ln \delta - \varepsilon_\delta(p) - D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta)\\ &= \Delta_\delta(x) - \varepsilon_\delta(p).\end{aligned} \tag{3.64}$$

Put differently, $\Delta'_\delta(x)$ is a shift of $\Delta_\delta(x)$ designed so that $\inf_{x\in\mathbb{N}} \Delta'_\delta(x) = 0$. Recall that we could have $\Delta_\delta(x) \gg 0$ for all $x \in \mathbb{N}$, which would imply some slackness in the capacity upper bound induced by the first inequality (3.60) via Theorem 2.5. The refinement in (3.62) removes this slackness.

Finally, combining the previous discussion with Theorem 2.5 leads to the capacity upper bound

$$\mathsf{Cap}_\mu(R_0) \leq \inf_{q\in(0,1),\delta\in(0,1]} (a\mu + b + \ln y_0 - \ln \alpha - d \ln \delta - \varepsilon_\delta(p)). \tag{3.65}$$

In the following section, we show that the optimisation over δ is not required, in the sense that we are able to analytically derive good explicit choices of δ under mild assumptions (which are satisfied by the distributions designed in Sections 3.2.1 and 3.2.2 over a large range of p).

3.2.3.2 Analytical derivation of good choices of δ

Optimising the right hand side of (3.65) over two parameters q and δ is cumbersome. In this section, we argue that a specific choice of δ works well over a large range of p for the distributions we designed, thus obtaining a much simpler bound than (3.65) which still gives great results. As discussed before, as a rule of thumb, a smaller KL-gap leads to improved upper bounds. The distributions we designed in Sections 3.2.1 and 3.2.2 have associated KL-gaps which converge to 1/2 and 0 when $x \to \infty$, respectively. In the case of the truncation-based distribution from Section 3.2.2, the speed of convergence is exponential in x. However, the KL-gap at small x does not behave as well. In general, it is significantly bounded away from 0 when $x = 1$. From experience, the KL-gap at small x appears to have significant influence on the sharpness of the upper bounds obtained. As a result, it is natural to wonder how one can obtain a small gap for small x without affecting the behaviour of the gap for large x.

Let $\Delta(x)$ be the original KL-gap of some distribution Y. Throughout this section, we make the following assumptions:

- $\Delta(x) \to L$ when $x \to \infty$;
- $\Delta_\delta(x) \geq \Delta(1) \geq L$ for all $x \in \mathbb{N}$.

We now describe how we can modify the mass at $y = 0$ to derive a new upper bound on $D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta)$ with a new KL-gap $\Delta'_\delta(x)$ such that $\Delta'_\delta(1) = 0$ and $\Delta'_\delta(x) \to 0$ when $x \to \infty$ with similar speed of convergence to the original KL-gap Δ. In other words, the new KL-gap $\Delta'_\delta(x)$ is as small as possible at $x = 1$, and also small when x is large. This leads to improved capacity upper bounds, as we shall see in Section 3.2.4. Figure 3.3 illustrates how the KL-gap changes with our explicit choice of δ.

We pick $\delta = \exp(-(\Delta(1) - L)/d)$, and proceed to justify this choice. Recalling the definition of $\Delta_\delta(x)$ in (3.61) and our assumptions, it holds that $\Delta_\delta(1) = \Delta(1)$ and that

$$\begin{aligned}\Delta_\delta(x) &= \Delta(x) - d\ln\delta + d^x \ln\delta \\ &= \Delta(x) + \frac{(d - d^x)(\Delta(1) - L)}{d} \\ &= \Delta(x) + (1 - d^{x-1})(\Delta(1) - L) \to \Delta(1) \qquad (3.66)\end{aligned}$$

when $x \to \infty$. Note that we pay only an exponentially small penalty in the speed of convergence compared to Δ. As we shall see, we have $\Delta_\delta(x) \geq \Delta(1)$ for all $x \in \mathbb{N}$ often for the distributions designed in Sections 3.2.1 and 3.2.2. In that case, we have $\varepsilon_\delta(p) = \inf_{x\in\mathbb{N}} \Delta_\delta(x) = \Delta(1)$. Recalling (3.62) and the choice of δ, it holds that

$$\begin{aligned} D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta) &\leq a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha - d\ln\delta - \varepsilon_\delta(p) \\ &= a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha + (\Delta(1) - L) - \Delta(1) \\ &= a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha - L, \end{aligned}$$

with corresponding KL-gap (recall (3.64))

$$\begin{aligned} \Delta'_\delta(x) &= a\mathbb{E}[Y_x] + b + \ln y_0 - \ln\alpha - L - D^{(e)}_{\mathsf{KL}}(Y_x||Y_\delta) \\ &= \Delta_\delta(x) - \Delta(1) \\ &\geq 0. \end{aligned}$$

As desired, the new KL-gap $\Delta'_\delta(x)$ satisfies

$$\Delta'_\delta(1) = \Delta_\delta(1) - \Delta(1) = \Delta(1) - \Delta(1) = 0$$

and, using (3.66),

$$\Delta'_\delta(x) = \Delta_\delta(x) - \Delta(1) \to \Delta(1) - \Delta(1) = 0$$

when $x \to \infty$.

3.2.3.3 Instantiation with concrete distributions

In this section, we instantiate the techniques developed in Sections 3.2.3.1 and 3.2.3.2 with the distributions designed in Sections 3.2.1 and 3.2.2.

We begin by considering the distribution $\overline{Y}^{(q)}$ from Section 3.2.2. We will use overlines over the relevant quantities associated to $\overline{Y}^{(q)}$ to distinguish from the same quantities associated to $Y^{(q)}$ from Section 3.2.1. Recalling (3.55), the KL-gap for the original distribution $\overline{Y}^{(q)}$, denoted by $\overline{\Delta}(x)$, satisfies

$$\overline{\Delta}(x) = R_p(x)$$

with $R_p(x)$ defined as in (3.54). We now compute the quantity $\overline{\Delta}_{\overline{\delta}}(x)$ associated with $\overline{Y}_{\overline{\delta}}^{(q)}$ for some value $\overline{\delta}$. According to (3.61), we have

$$\begin{aligned}\overline{\Delta}_{\overline{\delta}}(x) &= \overline{\Delta}(x) - d\ln\overline{\delta} + d^x \ln\overline{\delta} \\ &= R_p(x) - d\ln\overline{\delta} + d^x \ln\overline{\delta}. \qquad (3.67)\end{aligned}$$

Observe that we have $\overline{\Delta}(1) = R_p(1) > 0$ and $\overline{\Delta}(x) \to 0$ exponentially fast when $x \to \infty$. As a result, in accordance with Section 3.2.3.2, we set $\overline{\delta} = \exp(-R_p(1)/d)$. Recalling (3.55) and (3.62), this choice leads to the upper bound

$$D_{\mathsf{KL}}^{(e)}(Y_x||\overline{Y}_{\overline{\delta}}^{(q)}) \leq -\ln\overline{\alpha} - \mathbb{E}[Y_x]\ln q + R_p(1) - \overline{\varepsilon}_{\overline{\delta}}(p), \qquad (3.68)$$

where $\overline{\varepsilon}_{\overline{\delta}}(p) = \inf_{x\in\mathbb{N}} \overline{\Delta}_{\overline{\delta}}(x)$ and $1/\overline{\alpha} = \overline{\delta} + 1/\overline{y_0} - 1$.

Numerical evidence suggests that for $p \geq 0.6$ we have $\overline{\Delta}_{\delta}(x) \geq R_p(1)$ for all $x \in \mathbb{N}$. As was the case in Section 3.2.3.2, this implies that $\overline{\varepsilon}_{\overline{\delta}}(p) = R_p(1)$. Consequently, combining this fact with (3.68) leads to the bound

$$D_{\mathsf{KL}}^{(e)}(Y_x||\overline{Y}_{\overline{\delta}}^{(q)}) \leq -\ln\overline{\alpha} - \mathbb{E}[Y_x]\ln q$$

for $p \geq 0.6$, with respective KL-gap

$$\overline{\Delta}'_{\overline{\delta}}(x) = -\ln\overline{\alpha} - \mathbb{E}[Y_x]\ln q - D_{\mathsf{KL}}^{(e)}(Y_x||\overline{Y}_{\overline{\delta}}^{(q)}) = \Delta_{\overline{\delta}}(x) - R_p(1) \geq 0.$$

In particular, for this particular choice of $\overline{\delta}$ and $p \geq 0.6$ we now have

$$\overline{\Delta}'_{\overline{\delta}}(1) = 0 \quad \text{and} \quad \lim_{x\to\infty} \overline{\Delta}'_{\overline{\delta}}(x) = 0, \qquad (3.69)$$

and the convergence for large x is exponentially fast, as desired. These KL-gaps are plotted for some $p \geq 0.6$ in Figure 3.11.

Concluding, from (3.68) and Theorem 2.5 we obtain the following upper bound for general p.

Theorem 3.2. *We have*

$$\mathsf{Cap}_\mu(R_0) \leq \inf_{q\in(0,1)} (-\ln\overline{\alpha} - \mu\ln q) + R_p(1) - \overline{\varepsilon}_{\overline{\delta}}(p),$$

where $\overline{\delta} = \exp(-R_p(1)/d)$, $1/\overline{\alpha} = \overline{\delta} + 1/\overline{y_0} - 1$, *and* $\overline{\varepsilon}_{\overline{\delta}}(p) = \inf_{x \in \mathbb{N}} \overline{\Delta}_{\overline{\delta}}(x)$.

We now consider the distribution $Y^{(q)}$ defined in (3.31). The reasoning is analogous to the previous case, so we skip some parts of the instantiation. Recalling (3.30), we have

$$\Delta(x) = -\ln y_0 - \mathbb{E}[Y_x] \ln q - D^{(e)}_{\mathsf{KL}}(Y_x || Y^{(q)}) = \mathbb{E}\left[\ln \frac{\binom{Y_x/p-1}{Y_x}}{\binom{Y_x+x-1}{Y_x}} \right]. \tag{3.70}$$

By Lemma 3.7, we have that $\Delta(x) \to 1/2$ when $x \to \infty$. In the cases where $\Delta(1) \geq 1/2$, we can follow the general reasoning previously described in Section 3.2.3.2 and set $\delta = \exp(-(\Delta(1) - 1/2)/d)$. However, when $\Delta(1) < 1/2$, we simply set $\delta = 1$, i.e., we use the original distribution $Y^{(q)}$. Therefore, in general we set $\delta = \min(\exp(-(\Delta(1) - 1/2)/d), 1)$.

Recalling (3.61), we have

$$\Delta_\delta(x) = -\ln \alpha - \mathbb{E}[Y_x] \ln q - d \ln \delta - D^{(e)}_{\mathsf{KL}}(Y_x || Y^{(q)}_\delta) = \Delta(x) - d \ln \delta + d^x \ln \delta \geq 0, \tag{3.71}$$

where $1/\alpha = \delta + 1/y_0 - 1$. If $\Delta(1) \geq 1/2$, this leads to the bound

$$D^{(e)}_{\mathsf{KL}}(Y_x || Y^{(q)}_\delta) \leq -\ln \alpha - \mathbb{E}[Y_x] \ln q + \Delta(1) - 1/2 - \varepsilon_\delta(p), \tag{3.72}$$

where $\varepsilon_\delta(p) = \inf_{x \in \mathbb{N}} \Delta_\delta(x)$. Furthermore, in this case we have $\Delta_\delta(1) = \Delta(1)$ and $\Delta_\delta(x) \to \Delta(1)$ when $x \to \infty$, as before.

Numerical evidence suggests that for $p \in [0.35, 0.5]$ we have $\Delta_\delta(1) = \Delta(1) > 1/2$ and $\Delta_\delta(x) \geq \Delta(1)$ for all $x \in \mathbb{N}$. This means that $\varepsilon_\delta(p) = \inf_{x \in \mathbb{N}} \Delta_\delta(x) = \Delta(1)$ in this case, and so

$$D^{(e)}_{\mathsf{KL}}(Y_x || Y^{(q)}_\delta) \leq -\ln \alpha - \mathbb{E}[Y_x] \ln q - 1/2$$

for $p \in [0.35, 0.5]$, with associated KL-gap

$$\Delta'_\delta(x) = -\ln \alpha - \mathbb{E}[Y_x] \ln q - 1/2 - D^{(e)}_{\mathsf{KL}}(Y_x || Y^{(q)}) = \Delta_\delta(x) - \Delta(1) \geq 0.$$

Observe that, similarly to previous cases, we have

$$\Delta'_\delta(1) = 0 \quad \text{and} \quad \lim_{x \to \infty} \Delta'_\delta(x) = 0, \tag{3.73}$$

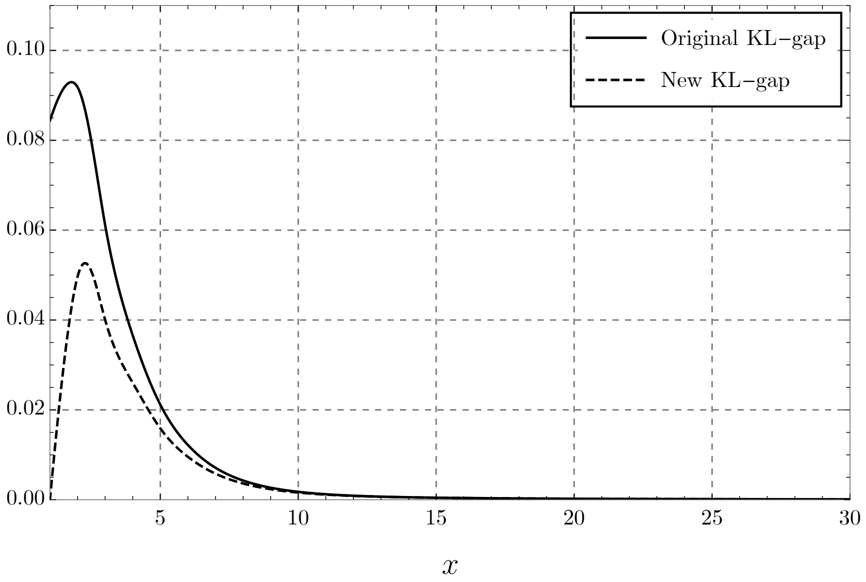

Figure 3.3: How the gap changes when the mass at $y = 0$ is modified, as a function of x for $p = 1/2$ and $Y^{(q)}$ defined in Section 3.2.1. Black curve: The original KL-gap $\Delta(x) - 1/2$. Dashed curve: The new KL-gap $\Delta'_\delta(x)$ after fixing the mass at $y = 0$ appropriately. Reproduced from [2]. ©2019 IEEE

as desired. Figure 3.3 showcases how the KL-gap changes for $p = 1/2$ when we modify $Y^{(q)}$ at $y = 0$ with our choice of δ. As can be observed, the KL-gap improves substantially for small x. The resulting KL-gaps are plotted for some $p \in [0.35, 0.5]$ in Figure 3.8.

From (3.72) and Theorem 2.5 we obtain the following upper bound for general p.

Theorem 3.3. *We have*

$$\mathsf{Cap}_\mu(R_0) \leq \inf_{q\in(0,1)} (-\ln \alpha - \mu \ln q) + \max(\Delta(1) - 1/2, 0) - \varepsilon_\delta(p),$$

where $\delta = \min(\exp(-(\Delta(1) - 1/2)/d), 1)$*,* $1/\alpha = \delta + 1/y_0 - 1$*, and* $\varepsilon_\delta(p) = \inf_{x\in\mathbb{N}} \Delta_\delta(x)$*.*

3.2.3.4 Some comments on the choice $\delta = d$

In this section, we briefly discuss the choice $\delta = d$ for $Y^{(q)}$ defined in Section 3.2.1. This choice corresponds *exactly* to the inverse binomial distribution, which was designed independently for the deletion channel [40], and leads to the result that the capacity of the geometric deletion channel is bounded well away from 1 when $p \to 1$ in Section 3.2.5.

We argue that there is a natural justification behind this choice. First, observe that we can extend the function $Y^{(q)}(\cdot)/y_0$ to $[0, \infty)$ in a natural way. Then, recalling the definition of the binomial coefficients in Section 2.2, we have

$$Y^{(q)}(0)/y_0 = \binom{-1}{0} = 1.$$

However, it is also the case that

$$\lim_{y\to 0^+} Y^{(q)}(y)/y_0 = d < 1.$$

It follows that $Y^{(q)}(\cdot)/y_0$ is not right-continuous at $y = 0$, and the unique choice of δ that makes $Y_\delta^{(q)}(\cdot)/\alpha$ right-continuous at $y = 0$ is $\delta = d$.

3.2.4 Capacity upper bounds for the geometric deletion channel

In this section, we analyse the capacity upper bounds we obtain for the geometric deletion channel by combining (3.27) with the distributions designed in Sections 3.2.1 and 3.2.2 and their modifications described in Section 3.2.3.

Note that the capacity of the geometric deletion channel with replication parameter p is upper bounded by the capacity of the deletion channel with deletion probability $d = 1-p$. In fact, we can simulate the output of a geometric deletion channel via the output of the deletion channel by having the receiver independently replace every output bit by $R = 1 + R_0$ copies of it.

It also holds that the capacity of the geometric deletion channel with replication parameter p is upper bounded by the capacity of the geometric sticky channel with the same replication parameter. For the remainder of this paragraph, we consider a modified geometric deletion channel that always deletes the first input bit. This modification does not change the capacity of the channel, and we can see it as a composition of a deletion channel with deletion probability $d = 1-p$ that always deletes the first input bit and a geometric sticky channel with replication parameter p. Let X be an input distribution for the modified geometric deletion channel supported on $\{0,1\}^n$ with associated output distribution Y, and Z the associated output distribution of the deletion channel. By the data processing inequality, we have $I(X;Y) \leq I(Z;Y)$, and Y can be seen as the output of the geometric sticky channel with replication parameter p on input Z. It always holds that $|Z| < n$, and we can extend Z to n bits by adding an extra run $\widehat{Z}$ (with different bit value) of length $n - |Z|$ at the end of Z. Letting $\widehat{Y}$ denote the output of the geometric sticky channel on input $\widehat{Z}$, using the fact that the channel is sticky yields

$$I(Z;Y) \leq I(Z,\widehat{Z};Y,\widehat{Y}) = I(Z\|\widehat{Z};Y\|\widehat{Y}) \leq \sup_{Z^{(n)}} I(Z^{(n)};Y_{Z^{(n)}}),$$

where the supremum is taken over all distributions $Z^{(n)}$ with support in $\{0,1\}^n$ and associated output $Y_{Z^{(n)}}$ under the geometric sticky channel. By Theorem 2.2, it follows that $\mathsf{Cap}(R_0) \leq \mathsf{Cap}(R)$ for all $p \in (0,1)$.

We will compare the bounds we obtain with the state-of-the-art capacity upper bounds for the deletion channel from [3] which, for most values of p, are still the best bounds found in the literature, and also to the best known capacity upper bounds on the geometric sticky channel from [1] and Section 3.1. However, when $p = 1/2$ the geometric deletion channel corresponds exactly to the binary replication channel which has been studied in depth in [1, 39] with $p_d = p_t = 1/2$. In particular, good numerical upper bounds have been derived for the binary replication channel, and hence for the geometric deletion channel with $p = 1/2$. Therefore, the setting with $p = 1/2$ will serve as a good standard to judge the performance of our upper bounds, and we shall single it out.

For $p \in (0,1)$, our bounds are obtained by combining (3.27) with Theorems 3.2 and 3.3, and choosing, for each $\mu > 0$, the value of q satisfying $\mathbb{E}[Y_\delta^{(q)}] = \mu$. This is possible because both distributions grow like $\Theta(q^y/\sqrt{y})$, and the proof is analogous to the one in Section 3.1.3. For the sake of completeness, we include it in Appendix A.4.

Corollary 3.2. *We have*

$$\mathsf{Cap}(R_0) \leq \sup_{\substack{q\in(0,1):\\ \mu_q \geq p/(1-p)}} \frac{p(-\varepsilon_\delta(p) - d\ln\delta - \ln\alpha - \mu_q \ln q)}{d(1+\mu_q)} \tag{3.74}$$

with $\delta = \min(\exp(-(\Delta(1) - 1/2)/d), 1)$, *and*

$$\mathsf{Cap}(R_0) \leq \sup_{\substack{q\in(0,1):\\ \overline{\mu}_q \geq p/(1-p)}} \frac{p(-\overline{\varepsilon}_{\overline{\delta}}(p) - d\ln\overline{\delta} - \ln\overline{\alpha} - \overline{\mu}_q \ln q)}{d(1+\overline{\mu}_q)} \tag{3.75}$$

with $\overline{\delta} = \exp(-R_p(1)/d)$, *where*

$$\begin{aligned}
1/\alpha &= \delta + \sum_{y=1}^{\infty} \binom{y/p-1}{y} q^y e^{-yh^{(e)}(p)/p},\\
\mu_q &= \sum_{y=1}^{\infty} \alpha y \binom{y/p-1}{y} q^y e^{-yh^{(e)}(p)/p},\\
1/\overline{\alpha} &= \overline{\delta} + \sum_{y=1}^{\infty} \frac{q^y e^{\Lambda_2(y)-\Lambda_1(y)-y\cdot\mathrm{li}(1/(1+2p))-yh^{(e)}(p)/p}}{y!},
\end{aligned}$$

$$\overline{\mu}_q = \sum_{y=1}^{\infty} \frac{\overline{\alpha} y q^y e^{\Lambda_2(y)-\Lambda_1(y)-y\cdot\mathrm{li}(1/(1+2p))-yh^{(e)}(p)/p}}{y!},$$

with Λ_1 *and* Λ_2 *as defined in* (3.49) *and* (3.50)*, respectively,* $\varepsilon_\delta(p) = \inf_{x\in\mathbb{N}} \Delta_\delta(x)$ *for* $\Delta_\delta(x)$ *defined in* (3.71)*, and* $\overline{\varepsilon}_{\overline{\delta}}(p) = \inf_{x\in\mathbb{N}} \overline{\Delta}_{\overline{\delta}}(x)$ *for* $\overline{\Delta}_{\overline{\delta}}(x)$ *defined in* (3.67)*.*

Figure 3.4 compares (3.74), (3.75), the state-of-the-art capacity upper bound for the deletion channel from [3], and the state-of-the-art capacity upper bounds for the geometric sticky channel from [1] and Section 3.1.2. Table 3.2 reports values of these bounds for the values of p analysed. Similarly to Section 3.1.2, one can reliably approximate $1/\alpha$, $1/\overline{\alpha}$, μ_q, and $\overline{\mu}_q$ by computing the sums up to a large enough number of terms depending on q. The asymptotic behaviour of the associated distributions ensures that the terms in the sums decrease exponentially fast, and that the number of terms we must consider is not a large function of q. Plots of the functions inside the suprema in (3.74) and (3.75) can be found in Figures 3.5 and 3.6, respectively. Similarly to the geometric sticky channel, numerical evidence strongly suggests that these functions are concave in q for fixed p. Exploiting these observations, maximising the relevant analytic function over $q \in (0,1)$ is tractable for a given $p \in (0,1)$. As in Section 3.1.2, Table 3.2 and Figure 3.4 present values in bits/channel use.

Figures 3.7 and 3.8 showcase the KL-gap attained by the distribution $Y_\delta^{(q)}$ from Section 3.2.1 with the choice of δ specified in Corollary 3.2 before and after shifting by the minimum KL-gap, respectively, for some values of p analysed. Figures 3.10 and 3.11 showcase the analogous gaps for $\overline{Y}_{\overline{\delta}}^{(q)}$. For the values of $p \in (0, 0.5]$ analysed, numerical evidence suggests that the infimum in $\varepsilon_\delta(p)$ is achieved at $x = 1$ (see Figure 3.7 for examples), and the same holds for $\overline{\varepsilon}_{\overline{\delta}}(p)$ when $p \in [0.6, 1)$ (see Figure 3.10 for examples). Moreover, we have $\Delta(1) \geq 1/2$ for the values of $p \in [0.35, 0.5]$ analysed. This indicates that the choices of δ and $\overline{\delta}$ in Corollary 3.2 (which were derived in Section 3.2.3) for the values of $p \in [0.35, 0.5]$ and $p \in [0.6, 1)$ analysed, respectively, yield distributions $Y_\delta^{(q)}$ and $\overline{Y}_{\overline{\delta}}^{(q)}$ whose KL-gaps are exactly 0 at $x = 1$ and converge to 0 quickly for large x. These gaps are displayed in Figures 3.8 and 3.11, respectively. For the sake of comparison, Figures 3.9 and 3.12 show the original KL-gaps of the distributions $Y^{(q)}$ and $\overline{Y}^{(q)}$ for some values of p.

We remark that the numerical values plotted in Figure 3.4 and reported in Table 3.2 are only approximations of the true bounds, since we instantiated (3.74) and (3.75) with the minimum KL-gaps suggested by numerical evidence and asymptotic convergence results for Δ_δ and $\overline{\Delta}_{\overline{\delta}}$. With respect to (3.75), our approximation of the minimum KL-gap is guaranteed to be correct to within additive error 10^{-10} for $p \geq 0.6$. Therefore, instantiating (3.75) with this approximation guarantees we are within $2 \cdot 10^{-10}$ bits/channel use of the true bound in that range of p. In fact, we can approximate $\varepsilon_\delta(p)$ and $\overline{\varepsilon}_{\overline{\delta}}(p)$ with high accuracy by numerically computing the KL-gap for a small number of values of x. This is guaranteed by Lemma 3.7 and the fact that $R_p(x) \to 0$ exponentially fast in x. In the case of $\overline{\varepsilon}_{\overline{\delta}}(p)$, for a prescribed accuracy $\alpha > 0$ we can explicitly compute the threshold $T(\alpha)$ required so that computing the KL-gap and taking the minimum for all integers $1 \leq x \leq T(\alpha)$ ensures that we obtain an approximation of $\overline{\varepsilon}_{\overline{\delta}}(p)$ to within additive error α: Recall that $\overline{\Delta}_{\overline{\delta}}(x) \to R_p(1)$ and

$$\overline{\Delta}_{\overline{\delta}}(x) = R_p(x) + R_p(1) - d^{x-1}R_p(1) \geq R_p(1) - d^{x-1}R_p(1)$$

by the choice of $\overline{\delta}$. Therefore, in order to approximate $\overline{\varepsilon}_{\overline{\delta}}(p)$ to within error α it suffices to compute $\overline{\Delta}_{\overline{\delta}}(x)$ for x until $d^{x-1}R_p(1) \leq \alpha$, or, equivalently, to compute it for integers $x \leq T(\alpha) = 1 + \frac{\ln \alpha - \ln R_p(1)}{\ln d}$. In particular, for $p \geq 0.6$ it is enough to compute $\overline{\Delta}_{\overline{\delta}}(x)$ for integers $1 \leq x \leq 30$, as shown in Figure 3.10, and take the minimum over these values to approximate $\overline{\varepsilon}_{\overline{\delta}}(p)$ to within additive error $\alpha = 10^{-10}$.

3.2.4.1 The case $p = 1/2$

When $p = 1/2$, the best known capacity upper bound was given in [1]. They report a bound of 0.209092 bits/channel use, obtained by employing a reduction from the original channel to a memoryless channel via the addition of undeletable markers between input runs (this same reduction was used in [89]), coupled with clever numerical methods. Our best analytical upper bound, which in particular employs a tighter reduction via Theorem 2.7 and the technique from Section 3.2.3, yields a bound of 0.168074 bits/channel use. In contrast, the best analytical upper bound we obtain *without* using the technique from Section 3.2.3, but still shifting the KL-gap down by $\varepsilon_1(1/2) = 1/2$ (suggested by numerical evidence), is noticeably worse: 0.199082 bits/channel use.

Remarkably, $Y^{(q)}$ is closely related to a negative binomial distribution $\mathsf{NB}_{1/2,q}$ when $p = 1/2$. This holds because of (3.33) and the fact that $\mathsf{InvBin}_{1/2,q} = \mathsf{NB}_{1/2,q}$ [40, Claim 20 and Expression (85)]. As a result, we can derive closed-form expressions for y_0 and $\mathbb{E}[Y^{(q)}]$. More precisely, since $y_{\mathsf{IB}} = \sqrt{1-q}$ when $p = 1/2$ we have

$$1/y_0 = 1 + \frac{1}{2} \cdot \frac{1 - \mathsf{NB}_{1/2,q}(0)}{y_{\mathsf{IB}}} = \frac{1}{2} + \frac{1}{2\sqrt{1-q}}$$

and

$$\mathbb{E}[Y^{(q)}] = \frac{y_0}{2y_{\mathsf{IB}}} \cdot \mathbb{E}[\mathsf{NB}_{1/2,q}] = \frac{q}{2(1-q)(1+\sqrt{1-q})}$$

for all $q \in (0,1)$. These closed-form expressions can be combined with Corollary 3.2 to yield simpler upper bounds when $p = 1/2$.

p	Upper bound deletion [3]	Upper bound geom. sticky	Analytical upper bound
0.05	0.021	0.814464	0.021244
0.10	0.041	0.714114	0.041351
0.15	0.062	0.640930	0.061242
0.20	0.082	0.583611	0.076981
0.25	0.103	0.537076	0.091134
0.30	0.123	0.498463	0.104846
0.35	0.144	0.465957	0.119552
0.40	0.165	0.438318	0.135271
0.45	0.187	0.414659	0.151342
0.50	0.212	0.394331	0.168074
0.55	0.241	0.376849	0.186588
0.60	0.275	0.361794	0.208075
0.65	0.315	0.348575	0.234480
0.70	0.362	0.336946	0.262103
0.75	0.420	0.326678	0.269490
0.80	0.491	0.317317	0.271810
0.85	0.579	0.308767	0.270561
0.90	0.689	0.300952	0.275251
0.95	0.816	0.293788	0.337581
0.99	0.963	0.288476	0.769426

Table 3.2: Comparison between the numerical capacity bounds for the deletion channel from [3], the best capacity upper bounds for the geometric sticky channel from [1] and Section 3.1.2, and the best analytical upper bound from Corollary 3.2 in bits/channel use. Reproduced from [2] with corrections to the values reported in the rightmost column when $p = 0.10$, $p = 0.60$, $p = 0.90$, and $p = 0.99$. ©2019 IEEE

3.2.5 A non-trivial capacity upper bound when $p \to 1$

Building on results obtained in Sections 3.2.1 and 3.2.3, we prove without computer assistance that the capacity of the geometric deletion channel is at most 0.73 bits/channel use for large replication parameter p, and thus bounded well away from 1 in this regime. This is in contrast with the behaviour of the deletion and Poisson-repeat channels in their analogous regimes.

Theorem 3.4. *We have*

$$\mathsf{Cap}(R_0) \leq \frac{1}{2\ln 2} + o(1) \quad \textit{bits/channel use}$$

when $p \to 1$, and $\frac{1}{2\ln 2} \approx 0.7214$.

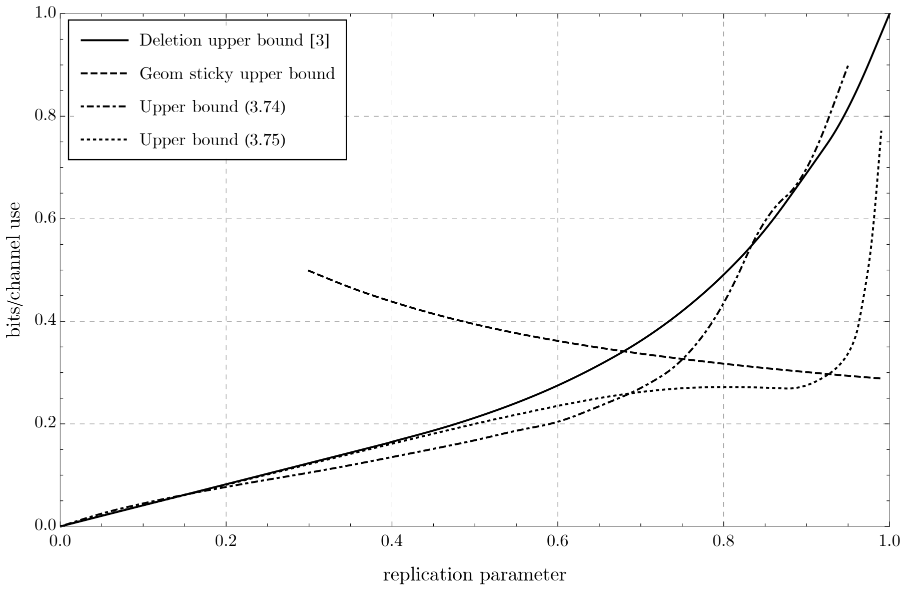

Figure 3.4: Plot of analytical upper bounds (3.74) and (3.75), the state-of-the-art deletion channel capacity upper bound from [3], and the state-of-the-art capacity upper bound for the geometric sticky channel from [1] and Section 3.1.2. Adapted from [2]. ©2019 IEEE

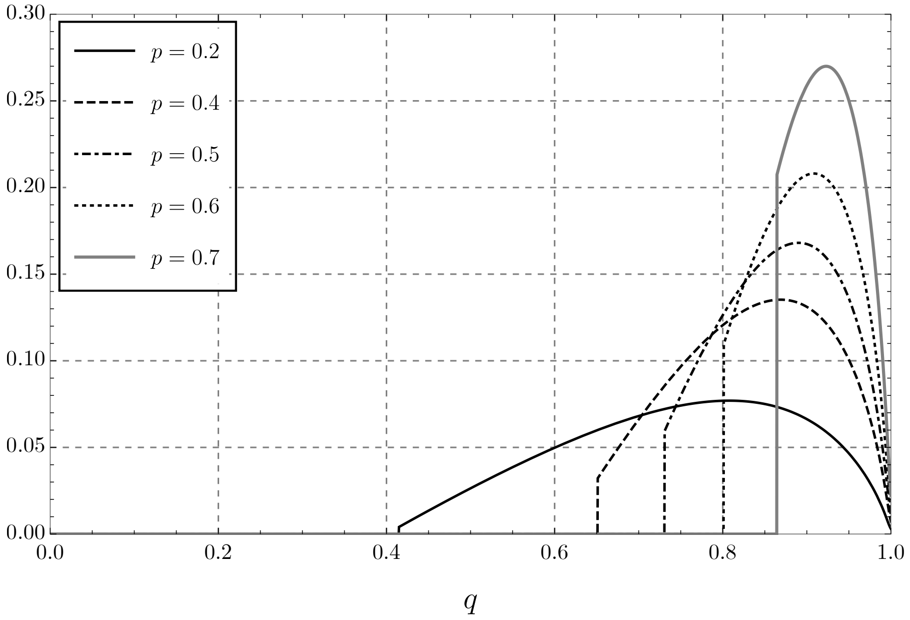

Figure 3.5: Function inside the supremum in (3.74) for some values of p, normalised by $\ln 2$. The region where the function is zero corresponds to the cases where $\mathbb{E}[Y_\delta^{(q)}] < \frac{p}{1-p}$. Adapted from [2]. ©2019 IEEE

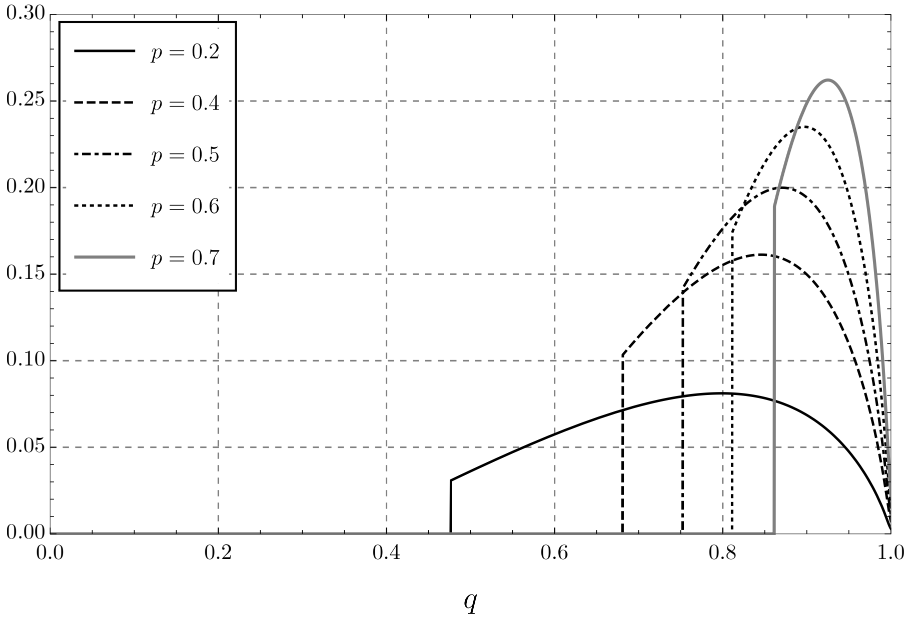

Figure 3.6: Function inside the supremum in (3.75) for some values of p, normalised by $\ln 2$. The region where the function is zero corresponds to the cases where $\mathbb{E}[\overline{Y}_{\overline{\delta}}^{(q)}] < \frac{p}{1-p}$. Adapted from [2]. ©2019 IEEE

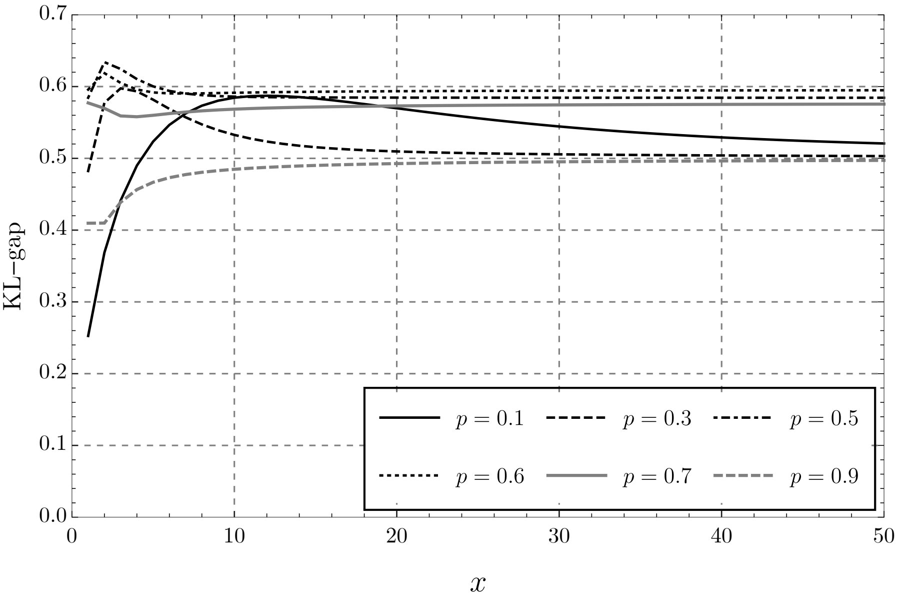

Figure 3.7: KL-gap $\Delta_\delta(x)$ (defined in (3.71)) of the distribution $Y_\delta^{(q)}$ from Section 3.2.1 with the choice of δ in Corollary 3.2 plotted for $1 \leq x \leq 50$ for some values of the replication parameter p.

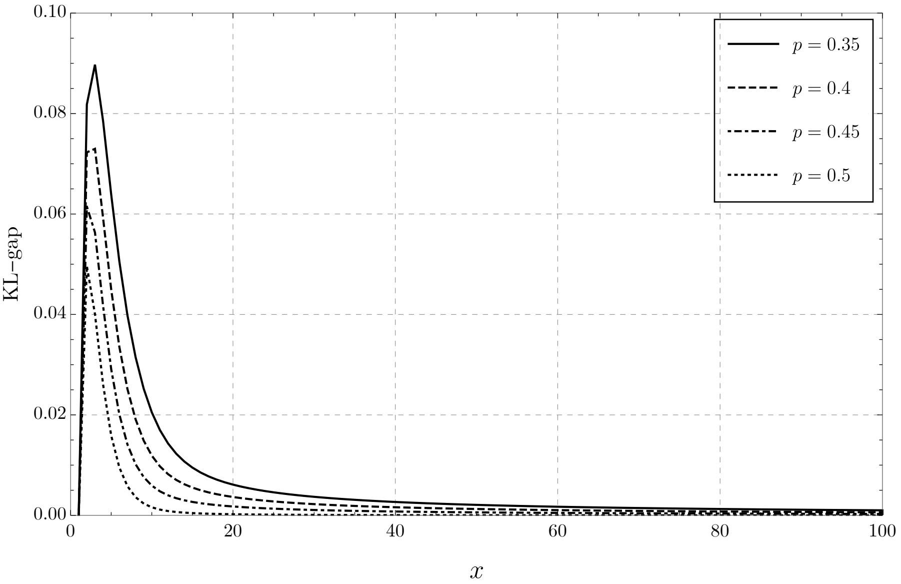

Figure 3.8: KL-gap $\Delta'_\delta(x)$ (defined in (3.64)) of the distribution $Y_\delta^{(q)}$ from Section 3.2.1 with the choice of δ in Corollary 3.2 plotted for $1 \le x \le 100$ for some values of the replication parameter p.

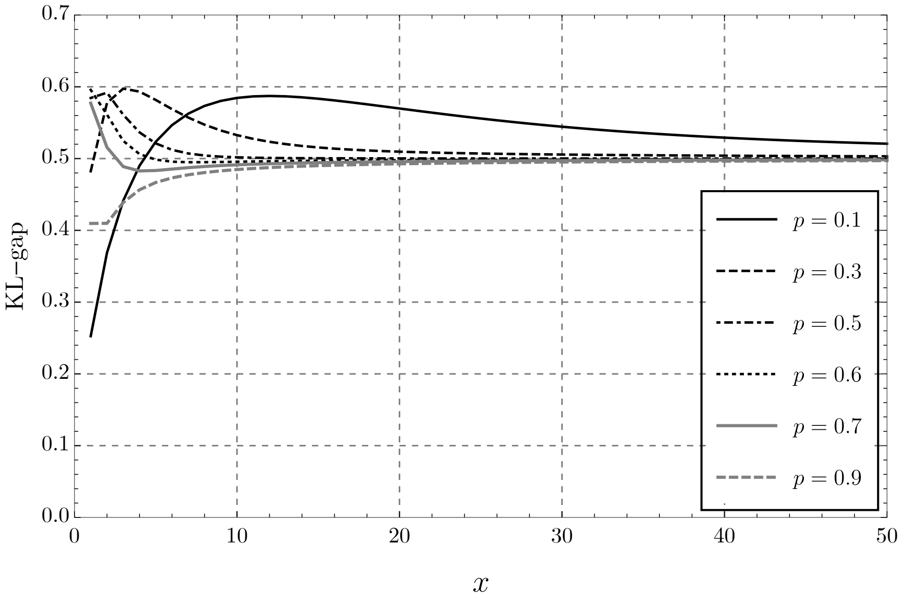

Figure 3.9: KL-gap $\Delta(x)$ (defined in (3.70)) of the distribution $Y^{(q)}$ from Section 3.2.1 plotted for $1 \le x \le 50$ for some values of the replication parameter p.

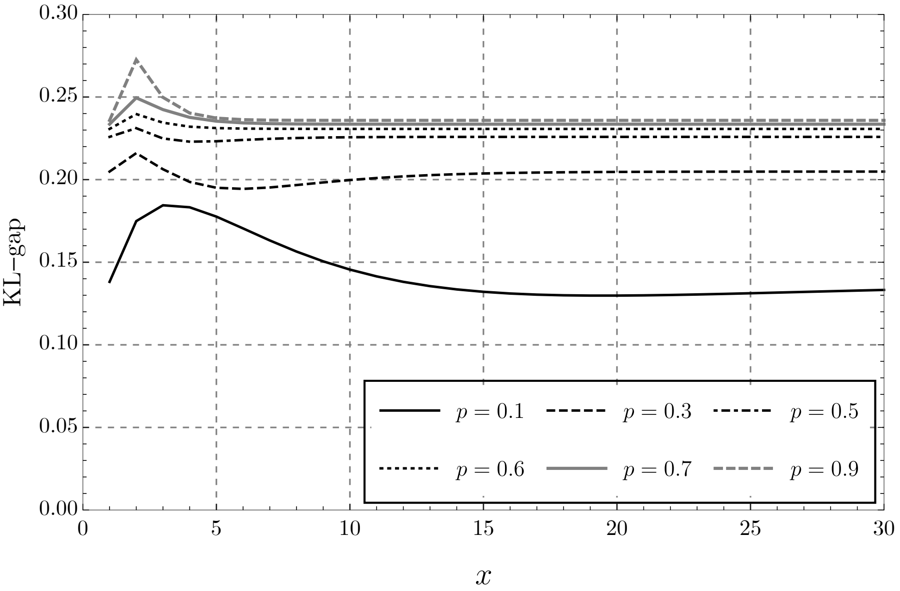

Figure 3.10: KL-gap $\overline{\Delta}_{\overline{\delta}}(x)$ (defined in (3.67)) of the distribution $\overline{Y}_{\overline{\delta}}^{(q)}$ from Section 3.2.2 with the choice of $\overline{\delta}$ in Corollary 3.2 plotted for $1 \leq x \leq 30$ for some values of the replication parameter p.

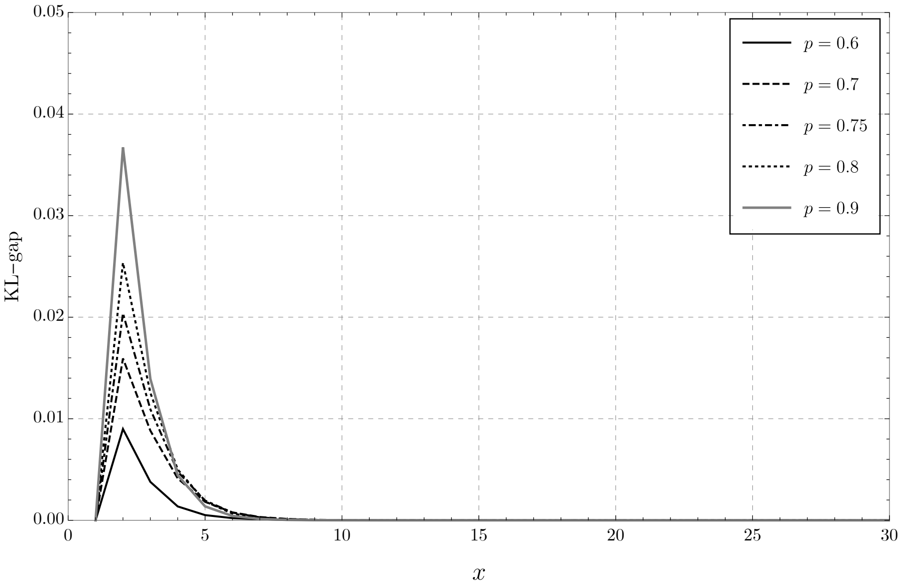

Figure 3.11: KL-gap $\overline{\Delta}'_{\overline{\delta}}(x)$ (defined in (3.64)) of the distribution $\overline{Y}_{\overline{\delta}}^{(q)}$ from Section 3.2.2 with the choice of $\overline{\delta}$ in Corollary 3.2 plotted for $1 \leq x \leq 30$ for some values of the replication parameter p.

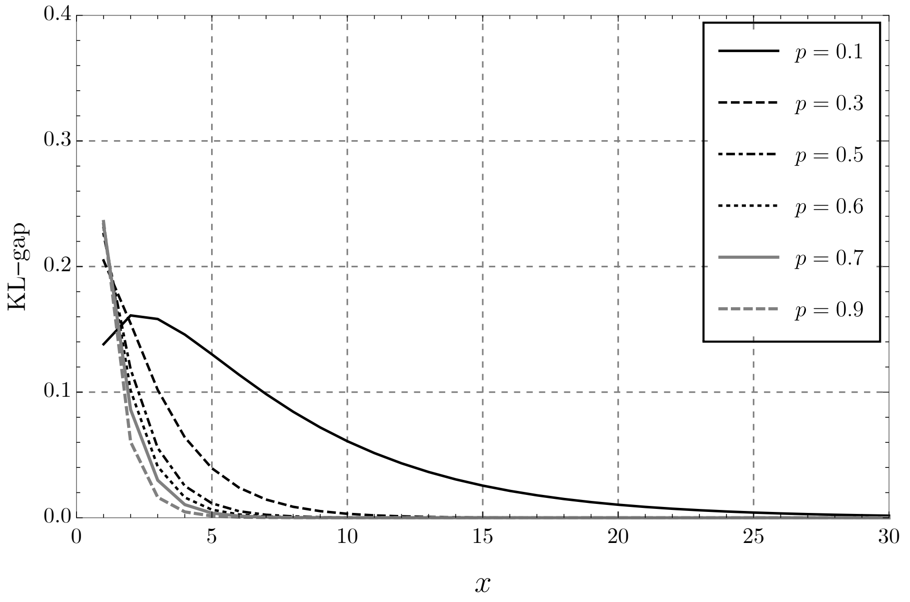

Figure 3.12: KL-gap $R_p(x)$ (defined in (3.54)) of the distribution $\overline{Y}^{(q)}$ from Section 3.2.2 plotted for $1 \leq x \leq 30$ for some values of the replication parameter p.

Proof. Define $d = 1 - p$. Combining (3.27) and (3.65) instantiated with $Y^{(q)}$ defined in Section 3.2.1, we conclude that

$$\mathsf{Cap}(R_0) \leq \frac{p}{d} \sup_{\mu \geq 1/d} \frac{1}{\mu} \inf_{q \in (0,1), \delta \in (0,1]} (-\varepsilon_\delta(p) - d \ln \delta - \ln \alpha - (\mu - 1) \ln q).$$

Moreover, recalling (3.63) and Lemma 3.6, we have

$$\varepsilon_\delta(p) \geq \varepsilon(p) \geq 0$$

for all $\delta \in (0,1]$ and $p \in (0,1)$. Therefore,

$$\mathsf{Cap}(R_0) \leq \frac{p}{d} \sup_{\mu \geq 1/d} \frac{1}{\mu} \inf_{q \in (0,1), \delta \in (0,1]} (-d \ln \delta - \ln \alpha - (\mu - 1) \ln q). \tag{3.76}$$

We set $\delta = d$, and begin by estimating $-\ln \alpha$. Recall that $1/\alpha = \delta + 1/y_0 - 1$. Then,

$$1/\alpha = \delta + d(1/y_{\mathsf{IB}} - 1) = d + d(1/y_{\mathsf{IB}} - 1) = d/y_{\mathsf{IB}}.$$

We can bound $1/y_{\mathsf{IB}}$ according to [40, Corollary 22] for $p \geq 1/2$ as

$$1/y_{\mathsf{IB}} \leq 1 + \frac{1}{\sqrt{2d}} \left(\frac{1}{\sqrt{1-q}} - 1 \right),$$

and so

$$1/\alpha \leq d + \sqrt{d/2}\left(\frac{1}{\sqrt{1-q}} - 1\right).$$

Setting $q = 1 - d/2$ yields

$$1/\alpha \leq d + \sqrt{d/2}\left(\frac{1}{\sqrt{d/2}} - 1\right) = 1 + d - \sqrt{d/2} < 1$$

for $d < 1/2$, which implies that $-\ln \alpha < 0$. Taking into account (3.76) and setting $\delta = d$, $q = 1 - d/2$, we obtain the bound

$$\begin{aligned}\mathsf{Cap}(R_0) &\leq \frac{p}{d} \sup_{\mu \geq 1/d} \frac{-d \ln d - \ln \alpha - (\mu - 1)\ln q}{\mu} \\ &\leq \sup_{\mu \geq 1/d} \frac{-d \ln d - \ln \alpha - (\mu - 1)\ln q}{\mu d} \\ &\leq -d \ln d - \frac{\ln q}{d},\end{aligned}$$

where in the second inequality we used the fact that $p < 1$, and in the third inequality we used the fact that $\mu d \geq 1$, $-\ln \alpha < 0$, and $-\frac{(\mu-1)\ln q}{\mu d} \leq -\frac{\ln q}{d}$ since $\mu > 1$.

Recalling that $q = 1 - d/2$, we have $-\frac{\ln q}{d} = \frac{1}{2} + o(1)$, where $o(1) \to 0$ when $d \to 0$. Finally, observe that $-d \ln d = o(1)$ as well. This gives the desired bound in nats/channel use, and dividing it by $\ln 2$ concludes the proof. □

Chapter 4

From synchronisation errors to the discrete-time Poisson channel

When one wishes to derive capacity upper bounds for the Poisson-repeat channel via Theorems 2.7 and 2.5, as was done in [40], one is led to study the mean-limited capacity of the DMC which on input $x \in \{0, 1, 2, \dots\}$ outputs

$$Y_x = \sum_{i=1}^{x} R_i,$$

where the R_i are i.i.d. according to Poi_λ. Because of the properties of the Poisson distribution, we have $Y_x \sim \mathsf{Poi}_{\lambda x}$. When $x = 0$, we have $Y_0 \sim \mathsf{Poi}_0$, which is the degenerate distribution satisfying $\mathsf{Poi}_0(0) = 1$. Throughout this chapter, we adopt the convention that $0 \ln 0 = 0$.

Interestingly, the channel above is a discrete-input analogue of a well-studied continuous-input channel with applications to optical communication, called the *Discrete-Time Poisson (DTP) Channel* with dark current λ [42]. This is a memoryless channel which on input $x \in \mathbb{R}_0^+$ outputs a sample from $\mathsf{Poi}_{\lambda+x}$. In particular, when $\lambda = 1$ for the channel above and $\lambda = 0$ for the DTP channel, these two channels behave in exactly the same way, except that the former is restricted to non-negative *integer* inputs. Without any constraints on the input, the capacity of the DTP channel is infinite. However, motivated by applications, one is interested in the capacity of the DTP channel under an *average-power constraint* μ, in which case the DTP channel only accepts input distributions X satisfying $\mathbb{E}[X] \leq \mu$. Orthogonally, sometimes it is also practical to impose a *peak-power constraint* A on the input distribution. In this case, the DTP channel only accepts input distributions X satisfying $\Pr[X \leq A] = 1$, which may additionally have to satisfy an average-power constraint. Given the similarity between the two problems above, a natural question arises: *Can we exploit the frameworks and results used to study the mean-limited capacity of the discrete-input DMC above to improve on state-of-the-art results for the DTP channel under an average-power constraint?*

In this chapter, we explore this question in two complementary directions. We begin by giving some historical background on the DTP channel in Section 4.1. Then, we show that capacity upper bounds from [40] on the Poisson-repeat channel can be easily adapted to give significantly improved capacity upper bounds for the DTP channel without dark current (i.e., $\lambda = 0$) in Section 4.2. Following that, in Section 4.3 we combine the results from Section 4.2 with ideas already developed in Section 3.2.3 for the geometric deletion channel to derive improved non-asymptotic capacity upper bounds for the DTP channel with dark current $\lambda > 0$. In Section 4.4, we complement our capacity upper bounds above by using an analogue of Theorem 2.5 to uncover novel properties of the capacity-achieving distributions for the DTP channel.

The material in this chapter is a close adaptation of material found in [4, 55], with minor modifications to improve exposition and consistency with the rest of the thesis.

4.1 Historical background

The DTP channel was first explicitly introduced and studied by Shamai [42], and is a discretised version of the continuous-time Poisson (CTP) channel used to model optical communication. Roughly speaking, as described in [42], the input to the CTP channel is a non-negative function $x(t)$ which modulates the intensity of a photon-emitting source on the sender's side. Then, the receiver observes a photon count following a Poisson process induced by $x(t)$, which may additionally be corrupted by some background interference. The DTP channel is derived in a natural way from the CTP channel by imposing a realistic *bandwidth* constraint on the channel input $x(t)$, meaning that we divide the timeline into equally sized sub-intervals, and require the input function x to be constant over each of these sub-intervals. As shown in [42], the capacity of this restricted channel is achieved by setting the value of $x(t)$ in each sub-interval in an i.i.d. fashion. This is captured by the DTP channel without dark current, which on input $x \in \mathbb{R}_0^+$ outputs a sample from Poi_x. Additionally, as mentioned above, this process may be corrupted by some background interference, and this is modelled in terms of an additive *dark current* term λ to x. Therefore, on input x, the DTP channel with dark current λ outputs $Y_x \sim \mathsf{Poi}_{\lambda+x}$. As mentioned above, the input distribution X may be constrained by imposing an average-power constraint $\mathbb{E}[X] \leq \mu$ and/or a peak-power constraint $X \leq A$. Interestingly, while the capacity of the CTP channel has been well-understood in several regimes under the constraints above for several decades [43, 44, 45, 46, 175], along with several other properties (e.g., see the recent works [176, 177] and references therein), no exact expression is known for the capacity of the DTP channel (which can be seen as a CTP channel restricted to more realistic coding techniques) under average- and/or peak-power constraints, and only loose bounds are known on this quantity outside the asymptotic regimes when $\mu \to 0$ or $\mu \to \infty$, especially when no finite peak-power constraint is imposed.

Throughout this chapter we will be mostly interested in the capacity of the DTP channel with an average-power constraint only (both with and without dark current), which is the continuous analogue of the DMC associated with the Poisson-repeat channel. We let $C(\lambda, \mu, A)$ denote the capacity of the DTP channel with dark current λ, average-power constraint μ, and peak-power constraint A, defined as

$$C(\lambda, \mu, A) = \sup_{X:\mathbb{E}[X]\leq\mu, X\leq A} I^{(e)}(X; Y_X), \tag{4.1}$$

where the supremum is taken over all input distributions X with support in $\mathbb{R}_0^+$ satisfying the average- and peak-power constraints, and Y_X denotes the output distribution of the DTP channel with dark current λ on input X. Observe that X in (4.1) is allowed to be *any* input distribution satisfying the constraints. In particular, it need not be discrete nor continuous and may not have an associated probability density function, although it always has an associated cumulative distribution function (cdf) F. While we only considered the case where X is discrete in Section 2, the mutual information is extended to the setting of this chapter as

$$I^{(e)}(X; Y_X) = \int D_{\mathsf{KL}}^{(e)}(Y_x \| Y_X) dF(x) = H^{(e)}(Y_X) - \int H^{(e)}(Y_x) dF(x),$$

where $Y_X(y) = \int Y_x(y) dF(x)$, $\int D_{\mathsf{KL}}^{(e)}(Y_x \| Y_X) dF(x)$, and $\int H^{(e)}(Y_x) dF(x)$ are Lebesgue integrals with respect to the probability measure induced by F (see [178, Section 2.3] or [179, Expression (2.10)]). As mentioned before, we will only deal with measure-theoretic probability in the proof of Theorem 4.1, which is discussed in Appendix B and requires only basic familiarity with this topic. To make the exposition mostly self-contained, we include an introductory section discussing the required concepts in Appendix B.1. Applying Theorem 4.1, which is a natural extension of previous results for other channels obtained via standard techniques, the maximisation problem in (4.1) is immediately replaced by an equivalent formulation that reduces to the analysis of information-theoretic quantities associated with discrete distributions only, as discussed in Section 2.4 and following the theme of the remainder of this thesis.

Shannon's noisy channel coding theorem (Theorem 2.1) can be extended to stationary memoryless channels with an average power constraint [178, Section 19], of which the DTP channel is an example. This means that, similarly to previous cases, the quantity $C(\lambda, \mu, A)$ is the supremum of rates achievable by length-n codes $\mathcal{C} \subseteq [0, A]^n$ with vanishing decoding error probability over the DTP channel as $n \to \infty$ whose codewords $c \in \mathcal{C}$ satisfy the average-power constraint

$$\frac{1}{n}\sum_{i=1}^{n} c_i \leq \mu.$$

When $A = \infty$, which is our main setting of interest, we use $C(\lambda, \mu)$ to denote the capacity. Moreover, when $\lambda = 0$ we denote the capacity of the DTP channel with zero dark current and under an average-power constraint only by $C(\mu)$. For every λ, μ, and A, we have the chain of inequalities

$$C(\mu) \geq C(\lambda, \mu) \geq C(\lambda, \mu, A),$$

which we will make use of later.

4.1.1 The capacity of the constrained DTP channel

The capacity of the DTP channel under average- and/or peak-power constraints, with or without dark current λ, has been studied in several different regimes. However, as discussed above, an exact expression for this capacity is still unknown, and in general we do not have sharp bounds on this quantity. As a result, most works have focused on asymptotic regimes where $\mu \to \infty$ or $\mu \to 0$.

Brady and Verdú [50] (see also the PhD thesis of Brady [180, Section 4]) were the first to study the capacity $C(\lambda, \mu)$ in a regime where the ratio μ/λ is fixed and $\mu \to \infty$. In particular, they obtained a class of asymptotic upper bounds (see [180, Section 4, Proof of Theorem 4])

$$C(\lambda, \mu) \leq \ln(1 + \mu + \lambda) + (\mu + \lambda)\ln\left(1 + \frac{1}{\mu + \lambda}\right) - \frac{1}{2}\ln(2\pi(\mu + \lambda)) + \ln(3/2) + \varepsilon \tag{4.2}$$

valid for all $\mu > C_\varepsilon$, where C_ε is a large constant depending on $\varepsilon > 0$. We note that, even disregarding the asymptotic term ε, the bound in (4.2) is only good when μ is large. Later, Lapidoth and Moser [51] studied $C(\lambda, \mu)$ when λ is a fixed arbitrary constant and $\mu \to \infty$. In particular, they showed that

$$\lim_{\mu \to \infty} \frac{C(\lambda, \mu)}{\ln \mu} = \frac{1}{2} \tag{4.3}$$

for arbitrary constant dark current $\lambda \geq 0$. They also determined the asymptotic behaviour of $C(\lambda, \mu, A)$ when the ratio μ/A is held fixed and $\mu \to \infty$.

In the regime where $\mu \to 0$, Lapidoth, Shapiro, Venkatesan, and Wang [48] determined the first-order asymptotics of $C(\lambda, \mu, A)$. Namely, in the case where there is no dark current, they showed that

$$\lim_{\mu \to 0} \frac{C(\mu)}{\mu \ln(1/\mu)} = 1.$$

Moreover, they gave the following general upper bound matching the asymptotic behaviour [48, Expression (86)],

$$C(\mu) \leq -\mu \ln p - \ln(1-p) + \frac{\mu}{\beta} + \mu \cdot \max\left(0, \frac{1}{2}\ln \beta + \ln\left(\frac{\overline{\Gamma}(1/2, 1/\beta)}{\sqrt{\pi}}\right) + \frac{1}{2\beta}\right), \tag{4.4}$$

where $p \in (0,1)$ and $\beta > 0$ are free constants, and $\overline{\Gamma}$ is the upper incomplete gamma function[1]. The optimal choice for p in (4.4) is $p = \frac{\mu}{1+\mu}$. When $\lambda > 0$, the capacity $C(\lambda, \mu)$ behaves quite differently from $C(\mu)$ when μ is small. It was shown in [48] that

$$\frac{1}{2} \leq \liminf_{\mu \to 0} \frac{C(\lambda, \mu)}{\mu \ln \ln(1/\mu)} \leq \limsup_{\mu \to 0} \frac{C(\lambda, \mu)}{\mu \ln \ln(1/\mu)} \leq 2 \tag{4.5}$$

for every $\lambda > 0$. In order to prove (4.5), the authors [48, Expression (114)] derive a non-asymptotic upper bound on $C(\lambda, \mu)$ given by

$$C(\lambda, \mu) \leq F_1(\lambda, \mu) + F_2(\lambda, \mu) + F_3(\lambda, \mu) \tag{4.6}$$

with F_1, F_2, and F_3 given by

$$F_1(\lambda, \mu) = \left(\eta \ln \eta + \frac{1}{12\eta} + \frac{1}{2}\ln(2\pi\eta) + \lambda - \eta \ln \lambda - \ln(1-p)\right) e^{\eta + \eta \ln \lambda - \eta \ln \eta + \frac{\mu}{\eta - \sqrt{\eta} - \lambda}},$$
$$F_2(\lambda, \mu) = \max\left(0, (1 + \ln(1/p) + \ln \lambda)\left(\mu + \frac{\lambda\mu}{\eta - \sqrt{\eta} - \lambda} + \lambda e^{\eta - 1 - \lambda + (\eta-1)\ln \lambda - (\eta - 1)\ln(\eta - 1)}\right)\right),$$

[1] The upper incomplete gamma function $\overline{\Gamma}$ is defined as $\overline{\Gamma}(s, \alpha) = \int_{\alpha}^{\infty} t^{s-1} e^{-t} dt$. Note that $\overline{\Gamma}(s, 0) = \Gamma(s)$.

$$F_3(\lambda,\mu) = \mu\left(1+\frac{\lambda}{\eta-\lambda}\right)\max\left(0,\ln(1/\lambda)\right) + \mu\frac{\eta\ln(\eta/\lambda)}{\eta-\lambda},$$

where η is a free integer parameter that must be larger than some non-explicit constant $C_\lambda \geq 0$ depending on λ and $p \in (0,1)$ is a free parameter. In [48, Section IV-B], it is explictly stated that the derivation is carried out assuming that η is large compared to λ, and inspection shows that we must at least have $\eta - \sqrt{\eta} > \lambda$. Therefore, the upper bound in (4.6) is significantly larger than

$$\mu(1+\max(0,1+\ln\lambda)+\max(0,\ln(1/\lambda))). \tag{4.7}$$

Later, Wang and Wornell [49] determined the higher-order asymptotic behaviour of $C(\lambda,\mu)$ when $\mu \to 0$ under the assumption that the dark current λ decreases linearly with μ, i.e., one has $\lambda = c\mu$ for some constant $c \geq 0$. In this case, it was shown that

$$C(\lambda = c\mu, \mu) = \mu\ln(1/\mu) - \mu\ln\ln(1/\mu) + O_c(\mu)$$

when $\mu \to 0$, where $O_c(\cdot)$ hides constants which depend on c. A version of this result was previously noted by Chung, Guha, and Zheng [181], although, as mentioned in [49], they only proved it for a more restricted set of input distributions. Wang and Wornell gave an upper bound on $C(\lambda,\mu)$ matching this asymptotic behaviour which holds whenever μ and λ are small enough. Namely, according to [49, Expression (180)], for μ small enough and $\lambda = c\mu$ it holds that

$$\begin{aligned} C(\lambda,\mu) \leq{}& \mu\ln\ln(1/\mu) + \mu - \ln(1-\mu-\lambda) - \lambda + \frac{\lambda^2}{2}\ln\ln(1/\mu) \\ &- (\mu+\lambda)\ln\left(1-\frac{1}{\ln(1/\mu)}\right) + \mu e^{-\lambda}\sup_{x\geq 0}\phi_{\mu,\lambda}(x), \end{aligned} \tag{4.8}$$

where $\phi_{\mu,\lambda}(x) = \frac{1-e^{-x}}{x}\ln\left(\frac{x+\lambda}{(\mu+\lambda)\ln(1/\mu)}\right)$. In the special case where $c = 0$ (i.e., there is no dark current), we have

$$C(\mu) \leq \mu\ln\ln(1/\mu) + \mu - \ln(1-\mu) - \mu\ln\left(1-\frac{1}{\ln(1/\mu)}\right) + \mu\cdot\sup_{x\geq 0}\phi_{\mu,0}(x). \tag{4.9}$$

Regarding general non-asymptotic upper bounds on $C(\mu)$, the best result for every μ except in the limiting regime $\mu \to 0$ is due to Martinez [47, Expression (10)], and is given by

$$C(\mu) \leq \left(\mu+\frac{1}{2}\right)\ln\left(\mu+\frac{1}{2}\right) - \mu\ln\mu - \frac{1}{2} + \ln\left(1+\frac{\sqrt{2e}-1}{\sqrt{1+2\mu}}\right). \tag{4.10}$$

We have that (4.10) attains the first-order asymptotic behaviour of $C(\mu)$ both when $\mu \to 0$ and when $\mu \to \infty$, and is strictly better than (4.4) for all $\mu > 0$. However, as noted in [51], the proof in [47] is not considered to be completely rigorous as it contains a gap (an intermediate step is only verified numerically). As we shall see in Section 4.3, the bound in (4.10) is also the best known upper bound on $C(\lambda, \mu)$ for reasonable choices of the dark current λ and the average-power constraint μ (namely, when λ is not significantly larger than μ). Remarkably, this means that there are no non-trivial non-asymptotic upper bounds on $C(\lambda, \mu)$ for reasonable choices of parameters, in the sense that the best upper bound we have in that case is an upper bound on $C(\mu)$, and the inequality $C(\mu) \geq C(\lambda, \mu)$ holds for every $\lambda \geq 0$.

Aminian, Arjmandi, Gohari, Nasiri-Kenari, and Mitra [182, Example 2] also derived a non-asymptotic upper bound on $C(\lambda, \mu, A)$ for finite peak-power constraint A given by

$$C(\lambda, \mu, A) \leq \sup_{X:\mathbb{E}[X]\leq\mu, X\leq A} \operatorname{Cov}(X+\lambda, \ln(X+\lambda)) = \begin{cases} \frac{\mu}{A}(A-\mu)\ln\left(1+\frac{A}{\lambda}\right), & \text{if } \mu < A/2, \\ \frac{A}{4}\ln\left(1+\frac{A}{\lambda}\right), & \text{otherwise,} \end{cases} \tag{4.11}$$

where Cov denotes the covariance. This upper bound is most useful when λ is large compared to the constraints μ and A. Note that when $A \to \infty$, the upper bound in (4.11) becomes arbitrarily large. Therefore, it does not imply any non-trivial upper bound on $C(\lambda, \mu)$. However, this bound may be used to recover some known asymptotic results on $C(\lambda, \mu, A)$ when $\mu \to 0$ from [48].

We note that an analytical lower bound was also given in [47]. This lower bound was obtained by considering gamma distributions as the input to the DTP channel (and thus negative binomial channel output). We have

$$\begin{aligned} C(\mu) \geq (\mu+\nu)\ln\left(\frac{\mu+\nu}{\nu}\right) &+ \mu(\psi(v+1)-1) \\ &- \int_0^1 \left[\left(1 - \left(\frac{\nu}{\nu+\mu(1-t)}\right)^{\nu}\right)\frac{t^{\nu-1}}{(1-t)\ln t} - \frac{\mu}{\ln t}\right] dt \end{aligned} \tag{4.12}$$

for all $\nu > 0$, where $\psi(y)$ is the digamma function. Martinez [183] also obtained the elementary lower bound $C(\mu) \geq \frac{1}{2}\ln(1+\mu)$. This bound behaves well when μ is large. In fact, as already mentioned, the capacity is known to behave like $\frac{1}{2}\ln\mu$ when $\mu \to \infty$. The lower bound in (4.12) was extended to dark current $\lambda > 0$ by Cao, Hranilovic, and Chen [184]. Other (asymptotic and non-asymptotic) capacity lower bounds for several settings were already implictly present in early works [185, 186, 187, 188], and can also be found in more recent works [51, 48, 49, 189]. Moreover, numerical approximations of the capacity of the DTP channel under a finite peak-power constraint were studied in [190, 54, 191].

4.1.2 Structure of capacity-achieving distributions

The problem of determining properties of interest of capacity-achieving distributions has been considered before for many different classes of channels, mostly those with continuous input alphabets. Usually, one is interested in determining whether the support of the capacity-achieving distribution is finite or discrete. Showing that the capacity-achieving distribution has these properties is useful in practice because it reduces the complexity of the problem of approximating this distribution, and allows the application of a wider range of numerical methods.

The landscape of this problem is well-understood for quite general classes of noise-additive channels under several input constraints. This line of work started with Smith [52], who showed that the capacity-achieving distributions for amplitude-constrained Gaussian channels have finite support. This result and its underlying technique were then extended to a more general set of amplitude- and average-power-constrained noise-additive channels by Das [192] and Tchamkerten [193], and later by Fahs and Abou-Faycal [53] to more general input-cost constraints. Other works have focused on studying such properties for certain noise-additive and closely related channels under peak and/or average-power constraints, such as noise-additive channels with piecewise-constant noise density functions [194], quadrature Gaussian channels [195], Rayleigh-fading channels [196], non-coherent and partially-coherent Gaussian channels [197], and non-coherent Rician fading channels [198]. A brief account of the application of Smith's technique from [52] to more general channels can be found in [199].

The structure of capacity-achieving distributions for the DTP channel was first studied by Shamai [42], who showed that capacity-achieving distributions for the DTP channel under a finite peak-power constraint must have finite support, and conjectured that capacity-achieving distributions for the DTP channel under an average-power constraint only have discrete support. He also gave conditions which ensure that distributions with two mass points are optimal. These results were extended by Cao, Hranilovic, and Chen [54, 200]. In particular, they showed that the support of a capacity-achieving distribution for the DTP channel under an average-power constraint only must be an unbounded set. Moreover, they also proved that such a distribution must have some mass at $x = 0$. Additionally, if there is only an active peak-power constraint A, they show there must also be some mass at $x = A$, and that this may not be the case otherwise. Unlike noise-additive channels, not much is known about the capacity-achieving distributions of the DTP channel when there is only an average-power constraint present. We will discuss previous approaches to deriving properties of the support of capacity-achieving distributions for the DTP channel in Section 4.4, along with our new result that capacity-achieving distributions for the DTP channel with an average-power constraint have a finite number of mass points in every bounded interval.

4.2 Improved capacity upper bounds for the DTP channel

In this section, we are interested in upper bounding the quantity

$$C(\mu) = \max_{X:\mathbb{E}[X]\leq\mu} I^{(e)}(X;Y_X),$$

where the maximum is taken over all distributions X supported in $\mathbb{R}_0^+$ satisfying $\mathbb{E}[X] \leq \mu$, and Y_X is the corresponding channel output distribution with $Y_x \sim \mathsf{Poi}_x$, i.e., such that

$$Y_x(y) = \frac{e^{-x}x^y}{y!}, \quad y = 0, 1, 2, \ldots,$$

with $Y_0(0) = 1$. We denote the DTP channel with dark current λ and average-power constraint μ by $\mathsf{DTP}_{\lambda,\mu}$, and omit λ when it is 0. We are interested in obtaining improved upper bounds on $C(\lambda, \mu)$ by transferring techniques used to upper bound the capacity of the Poisson-repeat channel to the average-power constrained DTP channel. In order to do this, we must first be able to apply an analogue of Theorem 2.5 in a continuous setting. Although Cheraghchi [40] only proved Theorem 2.5 for DMCs, it can be extended to stationary memoryless channels with continuous input alphabets as well.

Theorem 4.1. *Fix $\lambda \geq 0$ and suppose there exist constants $a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$ and a distribution Y over $\mathbb{N}_0$ such that*

$$D^{(e)}_{\mathsf{KL}}(Y_x \| Y) \leq ax + b$$

for every $x \in \mathbb{R}_0^+$, where $Y_x \sim \mathsf{Poi}_{\lambda+x}$ is the output of $\mathsf{DTP}_{\lambda,\mu}$ on input x. Then, we have

$$C(\lambda, \mu) \leq a\mu + b.$$

Moreover, an input X is capacity-achieving for $\mathsf{DTP}_{\lambda,\mu}$ if and only if $\mathbb{E}[X] = \mu$ and there exist constants

$a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$ such that

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y_X) \leq ax + b$$

for every $x \in \mathbb{R}_0^+$, with equality for all $x \in \mathsf{supp}(X)$. In this case, we have $C(\lambda, \mu) = a\mu + b$.

The first part of Theorem 4.1 can be obtained as a special case of a result of Lapidoth and Moser [201], which generalises an earlier convex duality result [202, Chapter 2, Theorem 3.4] to arbitrary alphabets.

Lemma 4.1 ([201, Theorem 5.1, specialised])**.** *Fix a stationary memoryless channel* Ch *with input alphabet $\mathbb{R}_0^+$ and output alphabet $\mathbb{N}_0$. Suppose that for every set $\mathcal{S} \subseteq \mathbb{N}_0$ the map $x \mapsto Y_x(\mathcal{S}) = \sum_{y \in \mathcal{S}} Y_x(y)$ is continuous on $\mathbb{R}_0^+$. Let X be any distribution on $\mathbb{R}_0^+$ with cdf F and Y any distribution on $\mathbb{N}_0$. Then, we have*

$$I^{(e)}(X; Y_X) \leq \int D_{\mathsf{KL}}^{(e)}(Y_x || Y) dF(x),$$

where Y_X denotes the output distribution of Ch *on input X. If instead the input alphabet is $\mathbb{N}$, then no assumptions on Y_x are necessary for the result to hold.*

In order to derive the first part of Theorem 4.1 from Lemma 4.1, suppose that Y is such that $D_{\mathsf{KL}}^{(e)}(Y_x \| Y) \leq ax + b$ for every $x \in \mathbb{R}_0^+$ and some $a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$. Then, for every X with cdf F and $\mathsf{supp}(X) \subseteq \mathbb{R}_0^+$ satisfying the average-power constraint we have

$$\begin{aligned} I^{(e)}(X; Y_X) &\leq \int D_{\mathsf{KL}}^{(e)}(Y_x || Y) dF(x) \\ &\leq \int (ax + b) dF(x) \\ &\leq a\mu + b, \end{aligned}$$

where the third inequality follows from the fact that $\int x dF(x) = \mathbb{E}[X] \leq \mu$. An analogous reasoning can be used to recover the capacity upper bounds in Theorem 2.5 from Lemma 4.1.

The optimality conditions in Theorem 4.1, which imply in particular that we can consider this more specialised version of Lemma 4.1 without any loss in optimality, and the existence of a capacity-achieving distribution for the DTP channel can be proved by following the approach from [196, Appendices I and II] for the Rayleigh-fading channel with an average-power constraint. For the sake of completeness, we provide a proof of these results in Appendix B, which is a close adaptation of [4, Appendices A and B].

4.2.1 The digamma distribution

The starting point for deriving improved upper bounds on $C(\mu)$ is the family of *digamma distributions*, which were already used in [40] to obtain analytical capacity upper bounds for the Poisson-repeat channel. This family of distributions $Y^{(q)}$ is parameterised by $q \in (0,1)$, and the distribution is given by

$$Y^{(q)}(y) = y_0 q^y \exp(g(y) - y - \ln(y!)), \quad y = 0, 1, \dots, \tag{4.13}$$

with g defined as

$$g(y) = \begin{cases} y\psi(y), & \text{if } y > 0, \\ 0, & \text{if } y = 0, \end{cases}$$

where ψ is the digamma function and y_0 is the normalising factor. In particular, for integer y we have $\psi(y) = -\gamma + \sum_{i=1}^{y-1} 1/i$, where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant.

We begin by showing that the digamma distribution $Y^{(q)}$ satisfies

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y^{(q)}) \leq -\ln y_0 - x \ln q \tag{4.14}$$

for every $x \geq 0$ and $q \in (0,1)$ via well-established results from the theory of special functions. First, similarly to other computations from Chapter 3, we have

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y^{(q)}) = D_{\mathsf{KL}}^{(e)}(\mathsf{Poi}_x \| Y^{(q)}) = -\ln y_0 - x \ln q + x \ln x - \mathbb{E}[g(Y_x)] \tag{4.15}$$

for every $x \geq 0$. Therefore, in order to show (4.14) it now suffices to prove that

$$\mathbb{E}[g(Y_x)] = \sum_{y=0}^{\infty} \frac{e^{-x} x^y g(y)}{y!} \geq x \ln x$$

for all $x \geq 0$. This holds when $x = 0$, and so it remains to consider the case $x > 0$. From the theory of special functions (by instantiating the Tricomi confluent hypergeometric function $U(a, n+1, x)$ with approriate parameters: [70, 13.1.6, p. 504 with $a = n + 1 = 1$ and $z = x$] combined with [70, 13.6.12, p. 509] and [70, 13.6.30, p. 510]), we have the identity

$$e^x E_1(x) = \sum_{y=0}^{\infty} \frac{\psi(1+y)}{y!} x^y - e^x \ln x, \tag{4.16}$$

where we recall the exponential integral E_1 from Section 2.2. Multiplying both sides of (4.16) by xe^{-x} and shifting the index in the sum appropriately leads to

$$\sum_{y=0}^{\infty} \frac{e^{-x}x^y g(y)}{y!} = x\ln x + xE_1(x) \geq x\ln x.$$

As a result, using the convention that $0E_1(0) = 0$, we obtain (4.14) for every $x \geq 0$ and $q \in (0,1)$ with associated KL-gap

$$\Delta(x) = xE_1(x). \tag{4.17}$$

Consequently, by Theorem 4.1 it follows that

$$C(\mu) \leq \inf_{q\in(0,1)} (-\ln y_0 - \mu \ln q). \tag{4.18}$$

4.2.2 Improved closed-form upper bounds

In this section, we discuss how to obtain good closed-form upper bounds on $C(\mu)$ from (4.18). Our starting point is a result of Cheraghchi [40] that bounds $-\ln y_0$ as a function of q via well-known properties of the gamma and digamma functions.

Lemma 4.2 ([40, Corollary 16])**.** *For all $y \geq 1$ and $q \in (0,1)$, we have*

$$\frac{2}{e^{1+\gamma}}\mathsf{NB}_{1/2,q}(y) \leq \frac{\sqrt{1-q}\cdot Y^{(q)}(y)}{y_0} \leq \frac{1}{\sqrt{2e}}\mathsf{NB}_{1/2,q}(y).$$

In particular, this implies that

$$\ln\left(1+\frac{2}{e^{1+\gamma}}\left(\frac{1}{\sqrt{1-q}}-1\right)\right) \leq -\ln y_0 \leq \ln\left(1+\frac{1}{\sqrt{2e}}\left(\frac{1}{\sqrt{1-q}}-1\right)\right)$$

for all $q \in (0,1)$.

Combined with (4.18), this immediately leads to the upper bound

$$C(\mu) \leq \inf_{q\in(0,1)} f(\mu,q), \tag{4.19}$$

where $f(\mu,q) = -\mu\ln q + \ln\left(1+\frac{1}{\sqrt{2e}}\left(\frac{1}{\sqrt{1-q}}-1\right)\right)$. In order to obtain good closed-form upper bounds on $C(\mu)$, it suffices now to instantiate $f(\mu,q)$ with good choices $q = q(\mu)$, and we briefly discuss a heuristic for deriving such choices.

As a starting point, since $C(\mu)$ is a concave unbounded function of μ, we know that the capacity-achieving distribution X for the DTP channel under average-power constraint μ satisfies $\mathbb{E}[X] = \mathbb{E}[Y_X] = \mu$, where Y_X denotes the associated channel output distribution. With this in mind, we want to derive choices $q = q(\mu)$ such that $\mathbb{E}[Y^{(q)}]$ is close to μ for all μ. Numerical evidence suggests that the optimal q satisfies this property, and moreover that the optimal q approaches 1 as $\mu \to \infty$. For q close to 1, Lemma 4.2 implies that $Y^{(q)}$ is well-approximated by $\mathsf{NB}_{1/2,q}$. To ensure that $\mathbb{E}[\mathsf{NB}_{1/2,q}] = \mu$, it suffices to set $q = \frac{2\mu}{1+2\mu}$, and thus we expect the choice $q_1(\mu) = \frac{2\mu}{1+2\mu}$ to yield a good upper bound on $C(\mu)$. Interestingly, we have

$$C(\mu) \leq f(\mu, q_1(\mu)) = \left(\mu + \frac{1}{2}\right) \ln\left(\mu + \frac{1}{2}\right) - \mu \ln \mu - \frac{1}{2} + \ln\left(1 + \frac{\sqrt{2e} - 1}{\sqrt{1 + 2\mu}}\right),$$

thus providing a rigorous proof of Martinez's bound in (4.10). This is not entirely surprising, as Martinez reaches (4.10) by considering a similar duality-based approach and a negative binomial distribution as the candidate, instead of the digamma distribution which we use here. However, we can refine the choice $q_1(\mu)$ above and improve significantly on Martinez's bound by setting $q_2(\mu)$ so that, similarly to $q_1(\mu)$, we have $q_2(\mu) = \frac{2\mu}{1+2\mu} + o(1/\mu)$ when $\mu \to \infty$ and also $\mathbb{E}[Y^{(q_2(\mu))}] = (1 + o(1))\mu$ when $\mu \to 0$.

We consider the choice $q_2(\mu)$ satisfying

$$\frac{1}{1 - q_2(\mu)} = 1 + \alpha\mu + \frac{\beta\mu^2}{1 + \mu}$$

for some constants α and β. We have that $\frac{1}{1-q_2(\mu)}$ behaves as $1 + \alpha\mu + o(\mu)$ when $\mu \to 0$ and as $1 + (\alpha + \beta)\mu + o(\mu)$ when $\mu \to \infty$, which means we can set its asymptotic behaviour in both the small and large μ regimes independently of each other. Moreover, setting $\alpha + \beta = 2$ leads to the desired behaviour $q_2(\mu) = \frac{2\mu}{1+2\mu} + o\left(\frac{1}{\mu}\right)$ when $\mu \to \infty$. We now proceed to choose α. Given our previous discussion, we determine the choice of α which ensures that $\mathbb{E}\left[Y^{(q_2(\mu))}\right] = \mu + o(\mu)$ when $\mu \to 0$. By construction, $q_2(\mu) = \alpha\mu + o(\mu)$ when $\mu \to 0$. We will need the following result.

Lemma 4.3. *We have* $\mathbb{E}\left[Y^{(q)}\right] = e^{-(1+\gamma)}q + o(q)$ *as* $q \to 0$.

Proof. Using the fact that $g(1)-1=\psi(1)-1=-(1+\gamma)$, we have

$$\frac{\mathbb{E}\left[Y^{(q)}\right]}{q}=y_0e^{-(1+\gamma)}+y_0\sum_{y=2}^{\infty}y\cdot\frac{e^{g(y)-y}q^{y-1}}{y!}. \tag{4.20}$$

From Lemma 4.2 it follows that y_0 approaches 1, and the rightmost term on the right-hand side of (4.20) vanishes when $q\to 0$. □

The remarks above, combined with Lemma 4.3, imply that $\mathbb{E}\left[Y^{(q_2(\mu))}\right]=e^{-(1+\gamma)}\alpha\mu+o(\mu)$ when $\mu\to 0$. Therefore, it suffices to set $\alpha=e^{1+\gamma}$ to have $\mathbb{E}\left[Y^{(q_2(\mu))}\right]=\mu+o(\mu)$ when $\mu\to 0$, as desired. Based on this, we choose $q_2(\mu)$ satisfying

$$\frac{1}{1-q_2(\mu)}=1+e^{1+\gamma}\mu+\frac{(2-e^{1+\gamma})\mu^2}{1+\mu}.$$

With this choice of $q_2(\mu)$, we obtain the upper bound

$$C(\mu)\ \leq\ \mu\ln\left(\frac{1+\left(1+e^{1+\gamma}\right)\mu+2\mu^2}{e^{1+\gamma}\mu+2\mu^2}\right)+\ln\left(1+\frac{1}{\sqrt{2e}}\left(\sqrt{\frac{1+(1+e^{1+\gamma})\mu+2\mu^2}{1+\mu}}-1\right)\right), \tag{4.21}$$

which improves on (4.10) for all $\mu>0$. Figure 4.1 compares the bounds derived in this section to previously known bounds. The curve corresponding to the bound of Lapidoth et al. (4.4) is the plot of $\mu\ln\left(\frac{1+\mu}{\mu}\right)+\ln(1+\mu)$, which lower bounds the right-hand side of (4.4). There is a noticeable improvement over Martinez's bound (4.10) when μ is not small, and one can see that (4.21) is close to (4.19) and (4.18), which confirms that the choice $q_2(\mu)$ is close to optimal. Due to the fact that our bounds are tighter than Martinez's bound, both of them satisfy the first-order asymptotic behaviour of $C(\mu)$ when $\mu\to 0$ and when $\mu\to\infty$. However, they do not exhibit the correct second order asymptotic term when $\mu\to 0$. In fact, the second-order asymptotic term of our bounds when $\mu\to 0$ is $-O(\mu)$, while the correct term is $-\mu\ln\ln(1/\mu)$. For this reason, our bounds do not improve on the Wang-Wornell bound (4.9) when μ is sufficiently small (numerically, when $\mu<10^{-6}$), while they noticeably improve on every previous bound when μ is not too small.

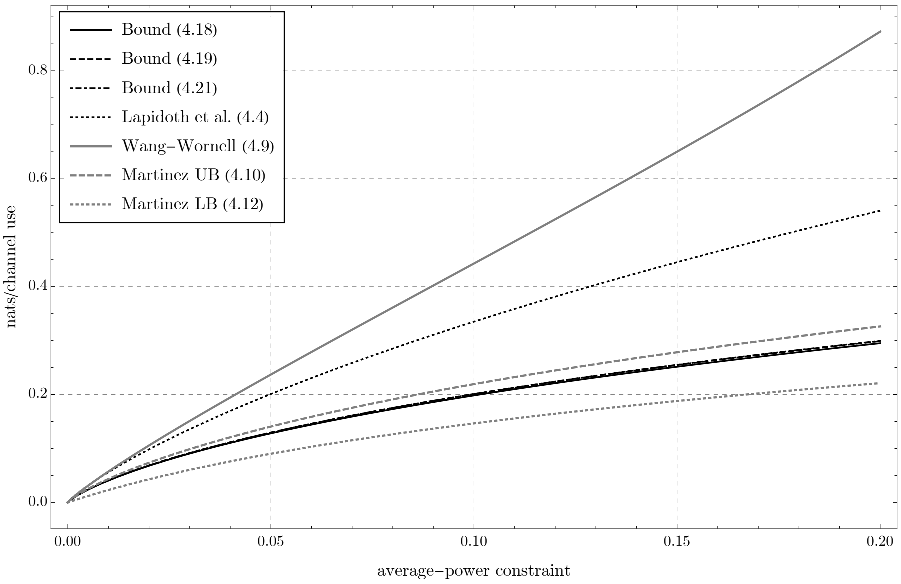

Figure 4.1: Comparison of upper bounds and the analytical lower bound (4.12) with $\nu = 0.05$ for $\mu \in [0, 0.2]$. Adapted from [4]. ©2019 IEEE

4.3 Capacity upper bounds for the DTP channel with positive dark current

In this section, we are interested in obtaining improved upper bounds on the capacity of the DTP channel with arbitrary dark current $\lambda > 0$ and average-power constraint μ, which is given by the expression

$$C(\lambda, \mu) = \sup_{X:\mathbb{E}[X]\leq\mu} I^{(e)}(X; Y_X),$$

where Y_X is the output distribution of $\mathsf{DTP}_{\lambda,\mu}$ induced by the input distribution X. As discussed before, every upper bound on $C(\mu)$ is also an upper bound $C(\lambda, \mu)$ for every $\lambda \geq 0$, which means that our improved upper bounds derived in Section 4.2 are also upper bounds on $C(\lambda, \mu)$. Our goal in this section is to improve on known upper bounds when $\lambda > 0$, and we do this by combining the digamma distribution described in Section 4.2.1 with techniques from Section 3.2.3.

4.3.1 The modified digamma distribution

The starting point in the derivation of our upper bounds is what we call the *modified digamma distribution* $Y_\delta^{(q)}$, where $\delta \in (0, 1]$ is a free parameter to be determined. Analogously to Section 3.2.3, our modification consists in changing the value of the digamma distribution $Y^{(q)}$ at $y = 0$ and renormalising the distribution. More precisely, for every such δ we define

$$Y_\delta^{(q)}(y) = \begin{cases} \alpha\delta, & \text{if } y = 0, \\ \alpha Y^{(q)}(y)/y_0, & \text{if } y > 0, \end{cases} \tag{4.22}$$

where $Y^{(q)}$ is the digamma distribution defined in (4.13) and α is the new normalising factor satisfying

$$1/\alpha = 1/y_0 + \delta - 1,$$

which follows from the fact that $Y^{(q)}(0)/y_0 = 1$ for the digamma distribution $Y^{(q)}$. We note again that a similar approach was employed by Martinez [47] in the special case where $\lambda = 0$ to improve the upper bound given by his different candidate distribution and obtain (4.10). However, no rigorous proof is given in [47] to show that this approach works in that special case, with only numerical evidence being presented. In this section, we use the approach above for general $\lambda \geq 0$ to analytically derive improved upper bounds on $C(\lambda, \mu)$.

Based on Section 3.2.3, we know that the quantity $D_{\mathsf{KL}}^{(e)}(Y_x \| Y_\delta^{(q)})$ has a simple expression in terms of $D_{\mathsf{KL}}^{(e)}(Y_x \| Y^{(q)})$. We compute it here explicitly. Recalling that $Y_x \sim \mathsf{Poi}_{\lambda+x}$ and setting $z = \lambda + x$, we have

$$\begin{aligned}
D_{\mathsf{KL}}^{(e)}(Y_x \| Y_\delta^{(q)}) &= D_{\mathsf{KL}}^{(e)}(\mathsf{Poi}_z \| Y_\delta^{(q)}) \\
&= -H^{(e)}(\mathsf{Poi}_z) - \sum_{y=0}^{\infty} \mathsf{Poi}_z(y) \ln Y_\delta^{(q)}(y) \\
&= -\ln\alpha - \mathsf{Poi}_z(0)\ln\delta - H^{(e)}(\mathsf{Poi}_z) + \sum_{y=1}^{\infty} \mathsf{Poi}_z(y)(\ln(y!) + y - g(y)) \\
&= -\ln\alpha - \mathsf{Poi}_z(0)\ln\delta - H^{(e)}(\mathsf{Poi}_z) + \mathbb{E}_{y\sim\mathsf{Poi}_z}[\ln(y!) + y - g(y) - y\ln q] && (4.23) \\
&= -\ln\alpha - z\ln q - e^{-z}\ln\delta - H^{(e)}(\mathsf{Poi}_z) + \mathbb{E}_{y\sim\mathsf{Poi}_z}[\ln(y!) + y - g(y)] && (4.24) \\
&= -\ln\alpha - z\ln q - e^{-z}\ln\delta + z\ln z - \mathbb{E}_{y\sim\mathsf{Poi}_z}[g(y)] && (4.25) \\
&= -\ln\alpha - z\ln q - e^{-z}\ln\delta - zE_1(z). && (4.26)
\end{aligned}$$

The equality (4.23) holds because the term inside the sum is 0 at $y = 0$. The equality (4.24) is true since $\mathbb{E}[\mathsf{Poi}_z] = z$ and $\mathsf{Poi}_z(0) = e^{-z}$. The equality (4.25) follows from the fact that

$$H^{(e)}(\mathsf{Poi}_z) = z + \mathbb{E}_{y\sim\mathsf{Poi}_z}[\ln(y!)] - z\ln z$$

for all $z \geq 0$. Finally, the equality (4.26) follows from (4.15) and (4.17).

We continue following the line of reasoning from Section 3.2.3. Given $\lambda \geq 0$, consider the choice

$$\delta_\lambda = \exp(-\lambda e^\lambda E_1(\lambda)). \tag{4.27}$$

Then, recalling that $z = \lambda + x$ we have

$$-e^{-z} \ln \delta_\lambda = e^{-x} \lambda E_1(\lambda).$$

Consequently, by defining $Y_\lambda^{(q)} = Y_{\delta_\lambda}^{(q)}$ and using (4.26) we have

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y_\lambda^{(q)}) = -\ln \alpha - (\lambda + x) \ln q + e^{-x} \lambda E_1(\lambda) - (\lambda + x) E_1(\lambda + x) \tag{4.28}$$

for every $x \geq 0$. We now claim that the following result holds.

Theorem 4.2. *For every $x \geq 0$, $q \in (0,1)$, and $\lambda > 0$ we have*

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y_\lambda^{(q)}) \leq -\ln \alpha - (\lambda + x) \ln q,$$

with KL-gap Δ_λ satisfying

$$\Delta_\lambda(x) = -\ln \alpha - (\lambda + x) \ln q - D_{\mathsf{KL}}^{(e)}(Y_x \| Y_\lambda^{(q)}) = (\lambda + x) E_1(\lambda + x) - e^{-x} \lambda E_1(\lambda) < \Delta(\lambda + x).$$

Analogously to Section 3.2.3, the choice of δ_λ in (4.27) ensures that we have $\Delta_\lambda(0) = 0$ and $\Delta_\lambda(x) \to 0$ exponentially fast when $x \to \infty$ (in general, $\Delta_\lambda(x)$ is always smaller than the KL-gap $\Delta(\lambda + x)$ of the original digamma distribution). Theorem 4.2 and the observations above justify our choice of δ_λ in (4.27); With this choice, we obtain a new family of modified digamma distributions $Y_\lambda^{(q)}$ with KL-gap Δ_λ that is always smaller than the original KL-gap Δ of the digamma distributions. Moreover, the KL-gap Δ_λ equals 0 at $x = 0$ and is significantly smaller than $\Delta(\lambda + x)$ around $x = 0$. Figure 4.2 compares the original KL-gap Δ with the new KL-gap Δ_λ for $\lambda = 1/2$. Given the above, intuitively we expect to obtain a sharper upper bound on $C(\lambda, \mu)$ using the family of modified digamma distributions.

Theorem 4.2 is an immediate consequence of (4.28) and the following lemma.

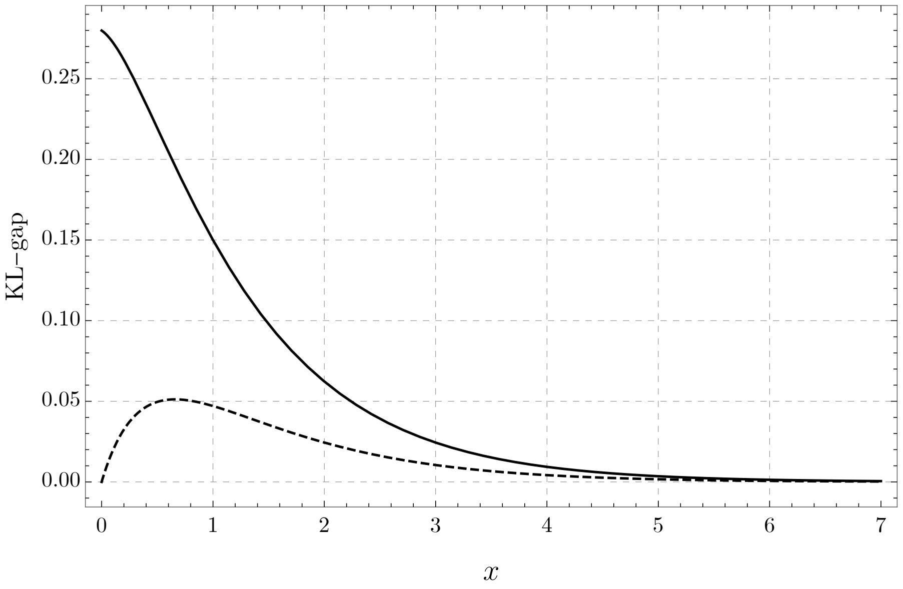

Figure 4.2: Comparison between $\Delta_\lambda(x)$ (dashed line) and $\Delta(\lambda + x)$ (full line) when $\lambda = 1/2$ for $x \in [0, 7]$.

Lemma 4.4. *For every $\lambda, x \geq 0$ we have*

$$\Delta_\lambda(x) = (\lambda + x)E_1(\lambda + x) - e^{-x}\lambda E_1(\lambda) \geq 0.$$

Proof. Multiplying both sides of the inequality above by $e^{\lambda+x}$, we conclude that the desired inequality holds provided we can show that

$$(\lambda + x)e^{\lambda+x}E_1(\lambda + x) \geq \lambda e^{\lambda}E_1(\lambda)$$

for all $\lambda, x \geq 0$. Since equality is achieved for every $\lambda \geq 0$ when $x = 0$, it is enough to show that the function $f(z) = ze^zE_1(z)$ is non-decreasing when $z > 0$. Note that we have

$$f'(z) = (1 + z)e^zE_1(z) - 1$$

for every $z > 0$, where we used the fact that $E_1'(z) = -e^{-z}/z$. We proceed to show that $f'(z) \geq 0$ for all $z > 0$, which implies the desired result. Recalling Lemma 2.9, we have the lower bound

$$e^zE_1(z) \geq \frac{1}{2}\ln(1 + 2/z)$$

for all $z > 0$. Therefore, it is enough to argue that

$$\frac{1+z}{2} \cdot \ln(1+2/z) \geq 1$$

for all $z > 0$. This follows from the fact that $\ln(1+x) \geq \frac{2x}{2+x}$ for all $x \geq 0$, and thus

$$\frac{1+z}{2} \cdot \ln(1+2/z) \geq \frac{1+z}{2} \cdot \frac{4/z}{2+2/z} = 1. \qquad \square$$

4.3.2 Derivation of the capacity upper bounds

In this section, we derive our capacity upper bounds with the help of Theorems 4.1 and 4.2. First, by combining these two results we have

$$C(\lambda,\mu) \leq \inf_{q\in(0,1)} [-\ln\alpha - (\lambda+\mu)\ln q], \tag{4.29}$$

where we recall that α is the normalising factor of the modified digamma distribution $Y_\lambda^{(q)}$ defined in (4.22) with $\delta = \delta_\lambda$ defined in (4.27). Then, similarly to Section 4.2, we can upper bound $-\ln\alpha$ in terms of upper bounds on $-\ln y_0$, where y_0 denotes the normalising factor of the original digamma distribution $Y^{(q)}$, by recalling Lemma 4.2 and the fact that $1/\alpha = 1/y_0 - 1 + \delta_\lambda$. Exploiting these observations, we conclude that

$$-\ln\alpha \leq \ln\left(\delta_\lambda + \frac{1}{\sqrt{2e}}\left(\frac{1}{\sqrt{1-q}} - 1\right)\right) \tag{4.30}$$

for every $q \in (0,1)$ and $\lambda \geq 0$.

It remains now to choose $q = q(\lambda,\mu)$ appropriately. Recalling Section 4.2.2, when $\lambda = 0$ the choice

$$q_2(\mu) = 1 - \frac{1}{1 + e^{1+\gamma}\mu + \frac{2-e^{1+\gamma}}{1+\mu}\mu^2}$$

is close to optimal for all $\mu \geq 0$, and leads to a significantly improved upper bound on $C(\mu)$. For the case where $\lambda > 0$, we consider the direct extension $q(\lambda,\mu)$ defined as

$$q(\lambda,\mu) = 1 - \frac{1}{1 + e^{1+\gamma}(\mu+\lambda) + \frac{2-e^{1+\gamma}}{1+\mu+\lambda}(\mu+\lambda)^2}, \tag{4.31}$$

where γ is the Euler-Mascheroni constant. Combining (4.29), (4.30), and (4.31) leads to the following upper bound on $C(\lambda, \mu)$.

Theorem 4.3. *For every $\mu, \lambda \geq 0$ we have*

$$C(\lambda, \mu) \leq \ln\left(\delta_\lambda + \frac{1}{\sqrt{2e}}\left(\frac{1}{\sqrt{1-q(\lambda,\mu)}} - 1\right)\right) - (\mu+\lambda)\ln q(\lambda,\mu), \tag{4.32}$$

where $\delta_\lambda = \exp(-\lambda e^\lambda E_1(\lambda))$, with the convention that $0E_1(0) = 0$, and $q(\lambda, \mu)$ defined in (4.31).

Remark 4.1. Although the upper bound in (4.32) does not have a closed-form expression due to the use of the exponential integral (which is nevertheless easy to compute numerically), we can use Lemma 2.9 to upper bound δ_λ as

$$\delta_\lambda \leq \min\left((1+2/\lambda)^{-\lambda/2}, (1-e^{-\lambda e^\gamma})^{\lambda e^\lambda}\right) \tag{4.33}$$

for all $\lambda > 0$. Replacing δ_λ in (4.32) by the upper bound in (4.33) leads to an improved upper bound on $C(\lambda, \mu)$ with a closed-form and elementary expression which sharply approaches (4.32) when λ is small and overall improves on previously known bounds.

Figures 4.3 and 4.4 compare our new upper bound on $C(\lambda, \mu)$ from (4.32) and the closed-form, elementary upper bound obtained by combining (4.32) and (4.33) to previously known upper bounds on $C(\lambda, \mu)$ and $C(\lambda, \mu, A)$. In Figure 4.4, the bound (4.6) is replaced by the underestimate (4.7), the upper bound (4.2) is plotted without the positive asymptotic term ε, and the upper bound (4.11), which is only valid when there is a peak-power constraint $A < \infty$, is plotted for the case $A = 1$. The upper bound (4.8) is plotted by ignoring the additive term $\mu e^{-\lambda} \sup_{x\geq 0} \phi_{\mu,\lambda}(x)$, which is always positive when it is well-defined, and the fact that this bound only holds when μ is small enough and $\lambda \to 0$ when $\mu \to 0$. In both figures, we see that the new upper bound (4.32) significantly improves on previously known upper bounds on $C(\lambda, \mu)$ whenever μ is not tiny, including the upper bound (4.11) on $C(\lambda, \mu, A = 1)$. Moreover, it is sharply approached by the elementary upper bound.

4.4 Structure of capacity-achieving distributions for the DTP channel

In this section, we show that capacity-achieving distributions for the DTP channel with arbitrary dark current $\lambda \geq 0$ under an average-power constraint and/or a peak-power constraint must be discrete.

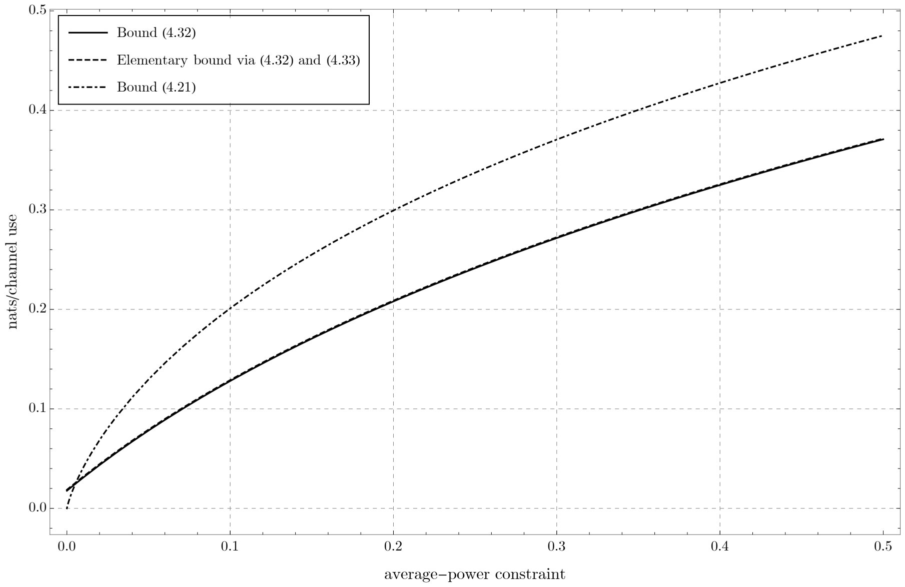

Figure 4.3: Comparison between the upper bound (4.32), the elementary upper bound combining (4.32) and (4.33), and the upper bound (4.21) when $\lambda = 1/10$.

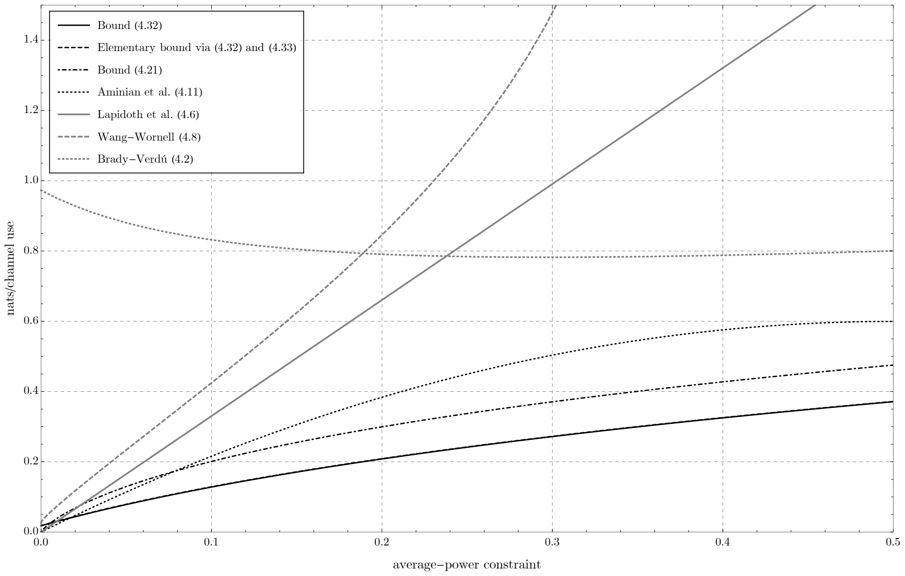

Figure 4.4: Comparison between the upper bound (4.32), the elementary upper bound combining (4.32) and (4.33), and previous upper bounds when $\lambda = 1/10$.

This proves a conjecture of Shamai [42]. As discussed in Section 4.1.2, previously it was only known that capacity-achieving distributions have finite support under a finite peak-power constraint [42], and that their support is an unbounded set when there is only an average-power constraint present [54]. We show the stronger result that the support of capacity-achieving distributions for the DTP channel under an average-power constraint and/or a peak-power constraint must have finite intersection with every bounded interval. Our techniques are general, and we recover Shamai's original result [42] for the DTP channel under a finite peak-power constraint with an alternative proof. For completeness, we show that there exist capacity-achieving distributions for the DTP channel under an average-power constraint in Appendix B.2.

It is interesting to point out connections between the study of properties of capacity-achieving distributions and the derivation of channel capacity upper bounds. First, techniques for tackling both problems make use of the convex duality-based approach to channel capacity. Second, an in-depth study of capacity-achieving distributions may lead to more refined capacity upper bounds. In fact, as we have seen in the previous sections, one obtains capacity upper bounds by designing candidate distributions Y to be used in the framework of Theorem 4.1. Since the capacity-achieving output distribution $Y^\star$ is the optimal candidate distribution for Theorem 4.1, one expects to obtain improved upper bounds by using candidate distributions Y that are more similar to $Y^\star$. As we have seen in Chapter 2, this approach has already been used with great success to obtain tight numerical capacity upper bounds for sticky channels in [32, 1]. Therefore, we expect that knowing more about the properties of the capacity-achieving input distribution will lead to a more informed design of candidate distributions for Theorem 4.1, and hence to better capacity upper bounds for the DTP channel.

4.4.1 The original argument for the DTP channel under a peak-power constraint

Before we present our proof, it is illustrative to describe the technique of Shamai [42], which in turn is an adaptation of the technique that Smith [52] used to study the analogous problem for the amplitude- and average-power-constrained Gaussian channel. The same technique was used to prove that capacity-achieving distributions for the DTP channel under an average-power constraint only have unbounded support [54], and was also recently used to prove discreteness of the support of capacity-achieving distributions for a general class of noise-additive channels under many different input-cost constraints [53].

Consider the DTP channel with average-power constraint μ and peak-power constraint $A < \infty$, and let

X be the capacity-achieving distribution with associated output distribution Y. Then, the optimality conditions from Theorem 4.1 state that there exist $a, b \in \mathbb{R}$ such that

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y) \leq ax + b$$

for all $x \in [0, A]$, with equality for $x \in \mathsf{supp}(X) \subseteq [0, A]$. Shamai [42] begins by noting that $D_{\mathsf{KL}}^{(e)}(Y_x \| Y)$ is a real analytic function[2] of x on $(0, \infty)$. If we assume that $\mathsf{supp}(X)$ is bounded but infinite, then the following property of real analytic functions allows us to conclude (ignoring some details) that $D_{\mathsf{KL}}^{(e)}(Y_x \| Y) = ax + b$ for all $x \in (0, \infty)$.

Theorem 4.4 (Identity theorem for real analytic functions [203, Corollary 1.2.6]). *If f and g are real analytic functions on an open interval $\mathcal{U} \subseteq \mathbb{R}$ and there exists a set $\mathcal{W} \subseteq \mathcal{U}$ with a limit point[3] in $\mathcal{U}$ such that $f(x) = g(x)$ for all $x \in \mathcal{W}$, then $f(x) = g(x)$ for all $x \in \mathcal{U}$.*

In order to reach a contradiction, it is now enough to argue that we must have $D_{\mathsf{KL}}^{(e)}(Y_x \| Y) \neq ax + b$ for some $x \geq 0$. To show that this is the case, Shamai [42] used the fact that $X \in [0, A]$ with probability 1 to argue that $D_{\mathsf{KL}}^{(e)}(Y_x \| Y) = \Omega_A(x \ln x)$ as $x \to \infty$, where the hidden constant depends on A. This implies the desired result, which in turn shows that $\mathsf{supp}(X)$ must be finite.

To prove our new result below, we also use Theorem 4.4 to conclude (again, ignoring some details) that we must have $D_{\mathsf{KL}}^{(e)}(Y_x \| Y) = ax + b$ for all $x \geq 0$. However, in our case we cannot assume that $\mathsf{supp}(X)$ is bounded, and therefore it is not clear whether $D_{\mathsf{KL}}^{(e)}(Y_x \| Y) = \Omega(x \ln x)$ still holds as x grows. We overcome this by instead analysing the behaviour of $D_{\mathsf{KL}}^{(e)}(Y_x \| Y)$ around $x = 0$ and deriving a contradiction.

4.4.2 Capacity-achieving distributions for the DTP channel have discrete support

Consider a discrete probability distribution Y supported on $\mathbb{N}_0$. For our results, it suffices to consider Y with full support. This is because all optimal output distributions of the DTP channel have full support, as the only input distribution which does not induce an output distribution with full support is the distribution that assigns probability 1 to $x = 0$, which is not optimal under average-power

[2] A function $f : \mathbb{R} \to \mathbb{R}$ is said to be *real analytic* on an open set $U \subseteq \mathbb{R}$ if for every $\alpha \in U$ there is an interval $\mathcal{I} = (\alpha - \varepsilon, \alpha + \varepsilon)$ and real numbers $(a_i)_{i=0,1,\dots}$ such that $f(x) = \sum_{i=0}^{\infty} a_i (x - \alpha)^i$ for all $x \in \mathcal{I}$ (see [203] for an extensive treatment of such functions).

[3] A real number x is a *limit point* of a set $\mathcal{W} \subseteq \mathbb{R}$ if there exists a sequence $(x_i)_{i \in \mathbb{N}}$ such that $x_i \in \mathcal{W}$, $x_i \to x$, and $x_i \neq x$ for all i.

constraint $\mu > 0$. For the sake of simplicity, in this section we will work under the assumption that $\lambda = 0$. Nevertheless, the approach extends easily to the case of arbitrary $\lambda \geq 0$. The following result gives a characterisation of optimal output distributions for the DTP channel (which we might also call *capacity-achieving*) that is obtained by writing a general distribution in the form $y_0 q^y \exp(f(y))$ for some function f and invoking Theorem 4.1. As seen before, writing distributions in this form simplifies the inequality $D^{(e)}_{\mathsf{KL}}(Y_x \| Y) \leq a\mathbb{E}[Y_x] + b$ considerably, and will allow us to instead focus on studying an inequality with a nice form. Before we proceed, we need the following definition.

Definition 4.1 (Exponential generating function). *Given a function $g : \mathbb{N}_0 \to \mathbb{R}$, its (real-valued)* exponential generating function*, which we denote by G_g, is defined as*

$$G_g(z) = \sum_{i=0}^{\infty} \frac{g(i)}{i!} z^i.$$

When the context is clear, we may denote G_g by G.

The key property of the distribution $Y_x \sim \mathsf{Poi}_x$ which we will exploit is that $\mathbb{E}[g(Y_x)] = e^{-x} \cdot G(x)$ for all $x \geq 0$.

Lemma 4.5. *Suppose X is capacity-achieving for the DTP channel under an average-power constraint $\mu > 0$ and peak-power constraint A (we may have $A = \infty$) with corresponding output distribution Y. Then, we can write Y as*

$$Y(y) = \exp(-ay - b + g(y) - y - \ln(y!)), \quad y \in \mathbb{N}_0, \tag{4.34}$$

for a function g such that $|g(y)| = O(y \ln y)$ and

$$G(x) \geq x e^x \ln x, \quad \forall x \in [0, A]$$

with equality for all $x \in \mathsf{supp}(X)$, where G is the exponential generating function of g with infinite radius of convergence, and we follow the convention that $0 \ln 0 = 0$.

Proof. By the assumptions on X and Y and Theorem 4.1 there exist $a, b \in \mathbb{R}$ such that

$$D^{(e)}_{\mathsf{KL}}(Y_x || Y) \leq a\mathbb{E}[Y_x] + b \tag{4.35}$$

for all $x \in [0, A]$, with equality if $x \in \mathsf{supp}(X)$. Consider the function g defined as

$$g(y) = ay + b + \ln Y(y) + y + \ln(y!), \quad y \in \mathbb{N}_0.$$

Noting that $\mathsf{supp}(X) \neq \{0\}$ (because otherwise X is not capacity-achieving under average-power constraint $\mu > 0$), it follows that Y must have full support over $\mathbb{N}_0$. Therefore, $g(y)$ is well-defined for every $y \in \mathbb{N}_0$, and (4.34) is satisfied by construction. To see that $|g(y)| = O(y \ln y)$, observe that there is an interval $\mathcal{I} = [c_1, c_2]$ with $0 < c_1 < c_2 < \infty$ such that $\Pr[X \in \mathcal{I}] > 0$ since $\Pr[X > 0] > 0$. Therefore, for $y > c_2$ we have

$$Y(y) \geq \Pr[X \in \mathcal{I}] \cdot Y_{c_1}(y) = \Pr[X \in \mathcal{I}] \cdot e^{-c_1} \cdot \frac{c_1^y}{y!}$$

since the map $x \mapsto Y_x(y)$ is increasing for fixed y when $x < y$, and so

$$0 \leq -\ln Y(y) \leq \ln(y!) - y \ln c_1 + c_1 - \ln \Pr[X \in \mathcal{I}] \leq y \ln y + O(y).$$

Therefore, we have

$$|g(y)| \leq ay + |b| - \ln Y(y) + y + \ln(y!) = ay + |b| + O(y \ln y) + y + \ln(y!) = O(y \ln y).$$

In particular, this implies that the exponential generating function $G(z)$ of g is finite for every $z \in \mathbb{R}$.

Finally, we may write

$$\begin{aligned}
D^{(e)}_{\mathsf{KL}}(Y_x||Y) &= -H^{(e)}(Y_x) - \mathbb{E}[\ln Y(Y_x)] \\
&= x \ln x - x - \mathbb{E}[\ln(Y_x!)] - \mathbb{E}[aY_x + b + g(Y_x) - Y_x - \ln(Y_x!)] \\
&= a\mathbb{E}[Y_x] + b + x \ln x - \mathbb{E}[g(Y_x)] \\
&= a\mathbb{E}[Y_x] + b + x \ln x - e^{-x} \cdot G(x),
\end{aligned}$$

where the last equality follows by observing that $\mathbb{E}[g(Y_x)] = e^{-x} \cdot G(x)$. Recalling (4.35), we conclude that

$$G(x) \geq xe^x \ln x$$

for all $x \in [0, A]$, with equality for all $x \in \mathsf{supp}(X)$. □

We also need the already mentioned result from [54] stating that optimal input distributions for the DTP_μ channel have unbounded support.

Theorem 4.5 ([54])**.** *Suppose X is a capacity-achieving distribution for the DTP channel under an average-power constraint $\mu > 0$ and no peak-power constraint. Then, $\mathsf{supp}(X)$ is an unbounded set.*

We now show that capacity-achieving input distributions for the DTP channel under an average-power constraint and/or a peak-power constraint must be discrete.

Theorem 4.6. *Suppose X is a capacity-achieving distribution for the DTP channel under an average-power constraint $\mu > 0$ and/or a peak-power constraint A (we may have $A = \infty$). Then, $\mathsf{supp}(X) \cap \mathcal{I}$ is finite for every bounded interval $\mathcal{I}$. In particular, $\mathsf{supp}(X)$ is countably infinite when $A = \infty$ and finite when $A < \infty$.*

Proof. Fix X as in the theorem statement and let Y be the corresponding output distribution. Then, Lemma 4.5 guarantees the existence of a function g such that its exponential generating function G converges everywhere and satisfies

$$G(x) \geq xe^x \ln x, \quad \forall x \in [0, A]$$

with equality for $x \in \mathsf{supp}(X)$.

Without loss of generality, we can focus on intervals $\mathcal{I} \subseteq (0, \infty)$, since $\mathsf{supp}(X) \subseteq [0, \infty)$. Suppose that there exists a bounded interval $\mathcal{I} \subseteq (0, \infty)$ such that $\mathcal{S} = \mathsf{supp}(X) \cap \mathcal{I}$ is infinite. Since G is a power series with infinite radius of convergence, it is real analytic on $\mathbb{R}$. Moreover, we have that $f(x) = xe^x \ln x$ is real analytic on $(0, \infty)$ and can be extended by continuity to $x = 0$ by setting $f(0) = 0$. We work with this extension from here onwards. Since G and f are both real analytic on $(0, \infty)$ and agree on the set $\mathcal{S}$ in this interval, it follows by Theorem 4.4 that $G(x) = f(x)$ for all $x \in (0, \infty)$ provided that $\mathcal{S}$ has a limit point in $(0, \infty)$.

Assume that indeed $\mathcal{S}$ has a limit point in $(0, \infty)$. Then, it follows that $G(x) = f(x)$ for all $x \in (0, \infty)$. We show that this leads to a contradiction. First, by right-continuity of G and f at 0, the above implies that $G(0) = f(0) = 0$. Then, we have $G'(0) = \lim_{x \to 0^+} G(x)/x = g(1) \in \mathbb{R}$. On the other hand, we have $\lim_{x \to 0^+} f(x)/x = \lim_{x \to 0^+} e^x \ln x = -\infty$. This implies that we cannot have $G(x) = f(x)$ for $x > 0$ small enough. As a result, we conclude that $\mathsf{supp}(X) \cap \mathcal{I}$ must be finite, as desired.

We now prove that $\mathcal{S}$ must have a limit point in $(0, \infty)$. Suppose that $\mathcal{S}$ has no limit points in $(0, \infty)$. Then, since $\mathcal{S}$ is a bounded infinite set in $(0, \infty)$ by hypothesis, it must be the case that 0 is a limit point of $\mathcal{S}$ (bounded infinite sets have at least one limit point by the Bolzano-Weierstrass theorem, and this limit point must lie in $[0, \infty)$ since $\mathcal{S} \subseteq (0, \infty)$). Therefore, there exists a sequence $(x_i)_{i \in \mathbb{N}}$ such that $x_i \in \mathcal{S}$ and $x_i \to 0$. By definition of $\mathcal{S}$, we must have $G(x_i) = f(x_i)$ for all i, and the argument from the previous paragraph shows that $G(x_i) \neq f(x_i)$ for i large enough, a contradiction.

Combining the above with Theorem 4.5 yields the desired result. $\square$

Chapter 5

Coded trace reconstruction

In this chapter, we introduce and study the *coded trace reconstruction* problem, where the goal is to design high-rate *efficient* coding schemes that can be decoded with high probability from few traces obtained by corrupting a codeword with a constant rate of i.i.d. deletions. The motivation for studying this problem is twofold, and is already apparent from our discussion in Section 2.5.3: On the one hand, it can be seen as a first step towards the design of efficient coding schemes for DNA-based data storage systems with nanopore-based sequencing [15, 16]. As previously discussed, current coding schemes for such storage systems are based on heuristics, and have no rigorous decoding properties, even in a simplified model with i.i.d. deletions. Overall, not much is known about the design of coding schemes in the trace reconstruction setting in general. On the other hand, coded trace reconstruction can also be seen as a natural extension of the average-case trace reconstruction problem. As we remarked before, average-case trace reconstruction algorithms imply the existence of high-rate coding schemes with an associated reconstruction algorithm that can reconstruct all codewords from few traces. However, efficient encoding procedures are not known for such codes. In contrast, our goal in this chapter is to design coding schemes with efficient encoding and decoding procedures.

As we shall see, coded trace reconstruction has deep connections to the original trace reconstruction problem. Notably, we will leverage results obtained in both the worst-case and average-case trace reconstructions already discussed in Section 2.5.3 to design our coding schemes.

We begin by rigorously introducing the coded trace reconstruction setting in Section 5.1. Then, we discuss a first efficient coding scheme combining a marker-based approach with worst-case trace reconstruction algorithms in Section 5.2. The flexibility of this construction allows us to enforce additional desirable properties, and in Section 5.3 we show that the coding scheme from Section 5.2 can be modified so as to have *balanced GC-content*, an important property for codes used in DNA-based data storage systems. In Section 5.4, we show that by leveraging results on average-case trace reconstruction we can obtain an efficient coding scheme with the same rate as that of Section 5.2, but requiring significantly fewer traces for reliable reconstruction. Then, we study extensions of the mean-based worst-case trace reconstruction algorithm from [60, 61] discussed in Section 2.5.3 that handle a more general set of replication errors, and argue how such reconstruction algorithms can be coupled with the approach from Section 5.2 to obtain results on coded trace reconstruction from a large class of repeat channels. Finally, we briefly discuss related work that appeared in parallel or subsequently to the first release of the work presented in this chapter.

The material presented in this chapter excepting Section 5.5 is based on [5] with minor modifications to improve the exposition and consistency with the rest of the thesis.

5.1 The coded trace reconstruction problem

The channel model of coded trace reconstruction may be formalised as follows. For a given input string $x \in \{0,1\}^n$, a deletion probability d, and an integer $t(n)$, the channel returns $t(n)$ *traces* of x. Each trace of x is obtained by sending x through the BDC_d channel, i.e., a deletion channel with deletion probability d. Thus, the $t(n)$ traces are i.i.d. according to the output distribution of BDC_d on input x.

Given a code $\mathcal{C} \subseteq \{0,1\}^n$, we say that $\mathcal{C}$ can be *efficiently reconstructed from* $t(n)$ *traces* if there exists a reconstruction algorithm Rec running in time $\mathrm{poly}(n)$ and a constant $\alpha > 0$ such that for every $c \in \mathcal{C}$ it holds that

$$\Pr\left[\mathsf{Rec}(T_1, \dots, T_t) = c\right] \geq 1 - \alpha/n,$$

where the T_i are i.i.d. according to the output distribution of BDC_d on input c for $i \in [t]$. In words, the reconstruction algorithm Rec recovers every codeword $c \in \mathcal{C}$ from $t(n)$ i.i.d. traces of c with probability at least $1 - \alpha/n$ over the randomness of the traces and the reconstruction algorithm. We remark that this definition corresponds to worst-case trace reconstruction restricted to codewords of $\mathcal{C}$. The goal of coded trace reconstruction is to design *efficiently encodable* codes $\mathcal{C}$ that can be efficiently reconstructed from $t(n)$ traces for $t(n)$ as small as possible. We note also that although this definition is stated specifically for the deletion channel, as that is our main focus, it can be extended in a straightforward manner to the setting where the traces are obtained by sending c through a different repeat channel. We study this setting in Section 5.5.

Remark 5.1. The reconstruction algorithms we study in Sections 5.2 and 5.3 are deterministic. On the other hand, the reconstruction algorithms from Section 5.4 are randomised as presented, since they require access to independent samples from Ber_d. Although this is already reasonable in practice, we can completely derandomise these algorithms by adding an efficient preprocessing step which extracts the necessary samples (to within small statistical distance) from extra traces, while keeping the total number of traces required and the final error probability of the same order. We discuss this in more detail in Remark 5.5.

Remark 5.2. We have decided to require reconstruction success probability $1-\alpha/n$ both for concreteness and for consistency when comparing our results to previous works on trace reconstruction (which generally also require similar success probability). The general tradeoff between code rate, number of traces, *and* success probability is not a focus of this chapter. However, it is natural to consider other settings (say, requiring only success probability $1-o(1)$), and we leave this as an interesting direction for future research. In view of this, we note that the difference (in terms of number of traces required) by algorithms with some constant success probability $\eta > 1/2$ and algorithms with success probability $1 - 1/\text{poly}(n)$ is relatively small. Given any algorithm Rec with success probability $\eta > 1/2$ using t traces, we can obtain an algorithm Rec' with success probability at least $1-1/n$ using $O(t \log n)$ traces. This is achieved by repeating Rec $O(\log n)$ times, each on a new batch of t traces, and choosing the most common output of Rec. By a direct application of the Hoeffding bound, we obtain the desired result. Of course, there is still much room for improvement if one aims for trace reconstruction with a sublogarithmic number of traces.

Remark 5.3. For simplicity, our exposition focuses mostly on constructions of *binary* codes, although it provides some guidelines and simple coding procedures for quaternary codes. It is also important to note that designing a code with a given rate R for coded trace reconstruction is inherently harder for smaller alphabets. Indeed, the existence of a binary code $\mathcal{C} \subseteq \{0,1\}^n$ with rate R that can be efficiently encoded and reconstructed from t traces with error probability ε implies the existence of a Q-ary code $\mathcal{C}'$ (for $Q = 2^q$) with the same rate R that can be efficiently encoded and reconstructed from t traces with error probability at most $q\varepsilon$. To see this, consider

$$\mathcal{C}' = \{(c^{(1)}, c^{(2)}, \dots, c^{(q)}) : c^{(i)} \in \mathcal{C}, i \in [q]\} \subseteq \{0,1\}^{q \cdot n},$$

which is a Q-ary code of length n and rate R. Moreover, let T denote a trace of some $c' = (c^{(1)}, c^{(2)}, \dots, c^{(q)}) \in \mathcal{C}'$. Observe that the trace $T^{(i)}$ obtained by replacing each Q-ary symbol in T by the i-th bit of its binary expansion has the same distribution as a trace of c^i. As a result, applying the transformation $T \mapsto T^{(i)}$ to each of the t traces of c' and running the reconstruction algorithm associated with $\mathcal{C}$ allows us to recover $c^{(i)}$ with error probability at most ε. Since this holds for every $i = 1, \dots, q$, a union bound over all i shows that we can simultaneously recover $c^{(1)}, c^{(2)}, \dots, c^{(q)}$ from t traces of c' with error probability at most $q\varepsilon$.

5.2 A simple marker-based construction

We start with a simple construction of high-rate codes that can be efficiently reconstructed from few traces. The idea behind the approach is the following: Each codeword contains markers, consisting of sufficiently long runs of 0s and 1s. Between two consecutive markers, we add a short block containing a codeword from an inner code satisfying a mild constraint.

Intuitively, the runs in the markers will still be long in the trace, and so we hope to be able to correctly identify the positions of all markers in a trace with high probability. After this is done, we can effectively split the trace into many shorter, independent sub-traces corresponding to a block (and possibly some bits from the two markers delimiting it). Then, we can apply worst-case trace reconstruction algorithms to the sub-traces. The savings in the number of traces required for reconstruction stem from the fact that sub-traces are short, and that each trace can be utilised simultaneously (and independently) by all blocks. This idea for reconstruction almost works as is, except that the process of identifying the markers in a trace may be affected by long runs of 0s originating from a block between two markers. However, this can be solved by requiring that all runs of 0s in each block are short enough. There exist low-redundancy codes satisfying the desired property, and hence we have good candidates for the inner code. This approach leads to the following result.

Theorem 5.1. *For n large enough there is an efficiently encodable code $\mathcal{C} \subseteq \{0,1\}^{n+r}$ with redundancy $r = O\left(\frac{n}{\log n}\right)$ which can be efficiently reconstructed from $\exp(O(\log^{2/3} n))$ traces for any constant deletion probability $d < 1$. Moreover, encoding can be perfomed in nearly-linear time $n \cdot \mathrm{poly}(\log n)$ and reconstruction can be performed in nearly-linear time $n \cdot \exp(O(\log^{2/3} n))$.*

In Section 5.3, we extend these ideas to the $\{A, C, G, T\}$ alphabet in order to obtain high-rate codes with desireable properties for DNA-based storage. Namely, these codes have balanced GC-content and can be reconstructed from few traces. Such codes are designed by exploiting the fact that the marker-based constructions can be instantiated with a large range of inner codes, and we can make the inner code satisfy yet stronger constraints.

We provide a precise description of the encoder Enc for our code $\mathcal{C}$ and prove Theorem 5.1. For simplicity, we consider $d \leq 1/2$ throughout. This choice of d is arbitrary, and the construction and analysis can be easily generalised to all constant $d \in [0, 1)$ by modifying leading constants appropriately.

Let $\ell = 50\lceil \log n \rceil$. Then, a marker M is a string of length 2ℓ of the form $M = 0^\ell \| 1^\ell$. We also require an efficiently encodable and decodable inner code $\mathcal{C}' \subseteq \{0,1\}^{m+r}$ with encoder $\mathsf{Enc}' : \{0,1\}^m \to \{0,1\}^{m+r}$, where $m = \lceil \log^2 n \rceil$ and r is the redundancy, satisfying the following property.

Property 5.1. *For all $c \in \mathcal{C}'$ and substrings s of c with $|s| = \lceil \sqrt{m} \rceil$, it holds that $\mathsf{wgt}(s) \geq |s|/3$, where $\mathsf{wgt}(s)$ denotes the Hamming weight of s.*

In other words, every codeword of $\mathcal{C}'$ has many 1s in all long enough substrings. Such efficient codes exist with redundancy $r = O(\log m) = O(\log \log n)$, which is enough for our needs. We provide a simple construction in Section 5.2.1.

Suppose we wish to encode an n-bit message $x \in \{0,1\}^n$. The encoder Enc on input x proceeds as follows:

1. Split x into $B = \lceil n/m \rceil$ blocks, each of length m (the last block may have to be padded with up to m bits)
$$x = x^{(1)} \| x^{(2)} \| \cdots \| x^{(B)};$$
2. Encode each block $x^{(i)}$ under the inner code $\mathcal{C}'$ to obtain $\overline{x}^{(i)} = \mathsf{Enc}'(x^{(i)}) \in \{0,1\}^{m+r}$;
3. Set the encoding of x, denoted by $\mathsf{Enc}(x)$, to be
$$\mathsf{Enc}(x) = 1^\ell \| \overline{x}^{(1)} \| M \| \overline{x}^{(2)} \| M \| \cdots \| M \| \overline{x}^{(B)} \| 0^\ell.$$

We remark that the first run 1^ℓ and the last run 0^ℓ are superfluous, and are added only to make the analysis simpler. Computing $\mathsf{Enc}(x)$ from x and decoding x from $\mathsf{Enc}(x)$ can both be done efficiently if the inner code $\mathcal{C}'$ is efficiently encodable and decodable. Figure 5.1 illustrates the encoding procedure for $\mathcal{C}$ with a general inner code $\mathcal{C}'$ satisfying Property 5.1 detailed above.

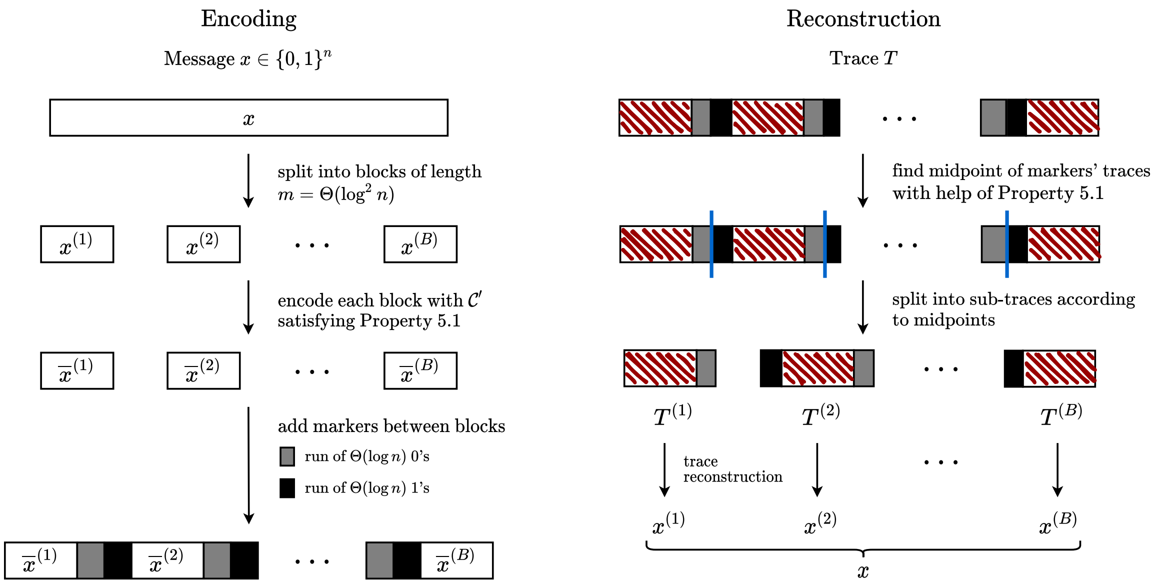
Encoding
Message $x \in \{0,1\}^n$
x
split into blocks of length $m = \Theta(\log^2 n)$
$x^{(1)}$ $x^{(2)}$ ⋯ $x^{(B)}$
encode each block with $\mathcal{C}'$ satisfying Property 5.1
$\overline{x}^{(1)}$ $\overline{x}^{(2)}$ ⋯ $\overline{x}^{(B)}$
add markers between blocks
run of $\Theta(\log n)$ 0's
run of $\Theta(\log n)$ 1's
Reconstruction
Trace T
find midpoint of markers' traces with help of Property 5.1
split into sub-traces according to midpoints
$T^{(1)}$ $T^{(2)}$ $T^{(B)}$
trace reconstruction
$x^{(1)}$ $x^{(2)}$ $x^{(B)}$
x

Figure 5.1: The general encoding and reconstruction procedures for $\mathcal{C}$. By considering different instantiations of the inner code $\mathcal{C}'$ and the trace reconstruction algorithm, we obtain Theorem 5.1, Theorem 5.2 (with slightly different markers and a 4-ary alphabet), and Theorem 5.4. Adapted from [5]. ©2020 IEEE

We now compute the redundancy of $\mathcal{C}$. It holds that

$$|\mathsf{Enc}(x)| \leq B(|M| + |\overline{x}^{(1)}|) = n + O\left(\frac{n}{\log n}\right) + Br. \tag{5.1}$$

Since we have $r = O(\log m) = O(\log \log n)$ and $B = O\left(\frac{n}{\log^2 n}\right)$, it follows that $\mathcal{C}$ has redundancy $O\left(\frac{n}{\log n}\right)$.

In the remainder of this section, we prove Theorem 5.1 using $\mathcal{C}$ via a sequence of lemmas. The reconstruction procedure works as follows: First, we show that the markers M still contain long enough runs after they are sent through the deletion channel. Then, we show that no long runs of 0s originate from the sub-traces associated with each block. This implies that we can correctly identify the position of the “01” part of each marker in the trace. As a result, we can split the trace into smaller “sub-traces”, each one associated to a different block $\overline{x}^{(i)}$. Then, we apply a reconstruction algorithm to the set of sub-traces associated to each block in order to reconstruct the blocks, and thus the whole codeword. This general reconstruction procedure is illustrated in Figure 5.1.

To obtain Theorem 5.1, we show that we can apply the worst-case trace reconstruction algorithm from Theorem 2.10 to recover each block with high probability and with the desired number of traces. As we will see in Section 5.4, a more careful instantiation of the inner code $\mathcal{C}'$ will allow us to use even more efficient trace reconstruction algorithms.

We start by proving that the markers M still contain long runs after they are sent through the deletion channel.

Lemma 5.1. *Let $0^{L_0}1^{L_1}$ be the output of the deletion channel on input M. Then,*

$$\Pr[L_0 > 10\log n, L_1 > 0] \geq 1 - n^{-4}$$

if n is large enough.

Proof. The result follows by a direct application of the Hoeffding bound. More precisely, we have $\mathbb{E}[L_0] \geq 25\log n$ since $\ell = 50\lceil \log n\rceil$ and $d \leq 1/2$, and thus

$$\begin{aligned}\Pr[L_0 \leq 10\log n] &\leq \Pr[\mathsf{Bin}_{\ell,1/2} \leq \mathbb{E}[\mathsf{Bin}_{\ell,1/2}] - 15\log n] \\ &\leq \exp\left(-\frac{2\cdot 15^2\log^2 n}{\ell}\right) \\ &\leq \exp\left(-\frac{2\cdot 15^2\log^2 n}{50(1+\log n)}\right) \\ &\leq n^{-5}\end{aligned}$$

if n is large enough. To conclude the proof, we note that $\Pr[L_1 = 0] \leq 2^{-\ell} \leq n^{-50}$. A union bound over the two events yields the desired result. □

We now show that no long runs of 0s originate from the sub-traces associated with each block.

Lemma 5.2. *Let $c \in \mathcal{C}'$. Then, a trace of c does not contain a run of* 0*s of length at least* $10\log n$ *with probability at least* $1 - n^{-3}$ *if n is large enough.*

Proof. Suppose that the run of 0s in question is obtained by deleting bits from a substring z of c with at most $4\log n$ 1s. This implies that for n large enough there is a substring s of z with $|s| = \lceil\sqrt{m}\rceil$ and $\mathsf{wgt}(s) < |s|/3$, violating Property 5.1. Therefore, for n large enough there must exist a substring of c with Hamming weight at least $4\log n$ such that all its 1s are deleted in the trace. The probability that all 1s of a fixed substring of c with Hamming weight at least $4\log n$ are deleted is at most n^{-4}. Since there are at most $m = \lceil\log^2 n\rceil$ choices for the substring, a union bound shows that the desired probability is at most $m\cdot n^{-4} < n^{-3}$ for n large enough. □

The next lemma follows immediately by combining Lemmas 5.1 and 5.2 with a union bound over the $B \leq \lceil \frac{n}{\log^2 n} \rceil$ blocks.

Lemma 5.3. *Consider the following event E with respect to a trace of $\mathsf{Enc}(x)$: Set $y^{(i)} = 1^\ell \| \overline{x}^{(i)} \| 0^\ell$ for each $i \in [B]$. Then, for every i the first run 1^ℓ of $y^{(i)}$ is not completely deleted, the last run 0^ℓ has length at least $10 \log n$ in the trace, and there is no run of 0s of length at least $10 \log n$ in the trace of $\overline{x}^{(i)}$.*

Then, E happens with probability at least $1 - n^{-2}$ over the randomness of the trace if n is large enough. In particular, if E happens we correctly identify the separation between the traces of 0^ℓ and 1^ℓ from every marker in the trace of $\mathsf{Enc}(x)$ by looking for all 1s that appear immediately after a run of at least $10 \log n$ 0s.

We are now ready to prove Theorem 5.1. Let E denote the event described in Lemma 5.3. Then, Lemma 5.3 implies that, conditioned on E, we can split a trace T of $\mathsf{Enc}(x)$ into B *sub-traces* $T^{(1)}, \dots, T^{(B)}$ satisfying the following:

- The sub-traces $T^{(i)}$ are independent for $i = 1, 2, \dots, B$;
- Each sub-trace $T^{(i)}$ is distributed like a trace of $1^\ell \| \overline{x}^{(i)} \| 0^\ell$ conditioned on the first run 1^ℓ not being completely deleted, the trace of the last run 0^ℓ having length at least $10 \log n$, and the trace of $\overline{x}^{(i)}$ not containing any run of 0s of length at least $10 \log n$.

As mentioned above, each sub-trace $T^{(i)}$ can be identified by looking for the $(i-1)$-th and i-th runs of 0 of length at least $10 \log n$ in the trace T, and picking every bit in T immediately after the $(i-1)$-th run up to and including the i-th such run.

Observe that $1^\ell \| \overline{x}^{(i)} \| 0^\ell$ has length $O(\log^2 n)$. Suppose that we have $t = \exp(O(\log n)^{2/3})$ independent traces $T_1, \dots, T_t$ of $\mathsf{Enc}(x)$. Let E_{all} denote the event that E holds for all T_i simultaneously. Combining Lemma 5.3 with a union bound over the t traces yields

$$\Pr[E_{\mathsf{all}}] \geq 1 - t/n^2 > 1 - 1/n \tag{5.2}$$

for n large enough. Fix some trace reconstruction algorithm $\mathcal{A}$, and let $E^{(i)}_{\mathsf{indFail}}$ denote the event that $\mathcal{A}$ fails to recover a fixed string $y^{(i)} = 1^\ell \| \overline{x}^{(i)} \| 0^\ell$ from t independent traces of $y^{(i)}$. Assuming that E_{all} holds, the sub-traces $T_1^{(i)}, \dots, T_t^{(i)}$ (note that $T_j^{(i)}$ denotes the i-th sub-trace of the j-th trace) are distributed as t independent traces of $y^{(i)}$, each also satisfying the conditions that the first run 1^ℓ is not completely deleted, the last run 0^ℓ has length at least $10 \log n$ in the trace, and there is no run of 0s of length at least $10 \log n$ in the trace of $\overline{x}^{(i)}$. We denote the event that these conditions hold for all of the t independent traces of $y^{(i)}$ by $E^{(i)}_{\mathsf{split}}$, meaning that we may write $E_{\mathsf{all}} = (\forall_i : E^{(i)}_{\mathsf{split}})$. Finally, we let E_{fail} denote the event that we fail to recover $\mathsf{Enc}(x)$ from the t i.i.d. traces $T_1, \dots, T_t$. Then, we have

$$\begin{aligned}\Pr[E_{\mathsf{fail}}] &\leq \Pr[E_{\mathsf{fail}}, E_{\mathsf{all}}] + \Pr[\neg E_{\mathsf{all}}]\\ &= \Pr[(\exists i : E^{(i)}_{\mathsf{indFail}}), (\forall i : E^{(i)}_{\mathsf{split}})] + \Pr[\neg E_{\mathsf{all}}]\\ &\leq \Pr[\exists i : E^{(i)}_{\mathsf{indFail}}] + 1/n\\ &\leq \sum_{i=1}^{B} \Pr[E^{(i)}_{\mathsf{indFail}}] + 1/n. \qquad (5.3)\end{aligned}$$

The first equality follows from the discussion in the previous paragraph, the second inequality follows from (5.2), and the third inequality follows by a union bound. Instantiating $\mathcal{A}$ with the worst-case trace reconstruction algorithm from Theorem 2.10, we conclude from (5.3) that

$$\Pr[E_{\mathsf{fail}}] \leq n \cdot \exp(-2 \log^2 n) + 1/n < 2/n$$

for n large enough.

As a result, we can successfully recover x from $\exp(O(\log n)^{2/3})$ traces of $\mathsf{Enc}(x)$ with probability at least $1 - 2/n$. Since recovering each $\overline{x}^{(i)}$ from the associated traces takes time $\exp(O(\log^{2/3} n))$, the inner code $\mathcal{C}'$ has an efficient decoder, and we can efficiently recover x from $\mathsf{Enc}(x)$, the whole procedure is efficient.

Remark 5.4. By modifying the inner block length m from $\lceil \log^2 n \rceil$ to $\lceil \log^{1+\gamma} n \rceil$ for an arbitrary constant $\gamma \in (0,1)$, the reasoning above can be adapted to yield efficiently encodable codes with redundancy $O(n/\log^\gamma n)$ which are efficiently reconstructible from $\exp\left(O\left(\log^{\frac{1+\gamma}{3}} n\right)\right)$ traces.

5.2.1 Instantiating the inner code

It remains to instantiate the inner code $\mathcal{C}'$ with the appropriate parameters and properties. To this end, we present a simple construction of an efficiently encodable and decodable inner code $\mathcal{C}'$ with encoder $\mathsf{Enc}' : \{0,1\}^m \to \{0,1\}^{m+r}$ and redundancy $r = O(\log m)$. We can then obtain the desired code by setting $m = \lceil \log^2 n \rceil$. Our starting point is the following result.

Lemma 5.4. *Let $g : \{0,1\}^t \to \{0,1\}^m$ be the function whose existence is guaranteed by Corollary 2.1 with $\varepsilon = 2^{-10w}$ for $w = 100\lceil \log m \rceil$ (hence $t = O(\log m)$). Fix some $x \in \{0,1\}^m$ and consider the random variable $Y = x \oplus g(U_t)$. Then, with probability at least $1 - 2/m$, we have that Y satisfies the following property:*

Property 5.2. *It holds that $\mathsf{wgt}(Y[a, a+w)) \geq 0.4w$ simultaneously for all $a \in [m - w + 1]$.*

Proof. Fix some $a \in [m - w + 1]$. Then, we have

$$\begin{aligned} \Pr[\mathsf{wgt}(Y[a, a+w)) < 0.4w] &= \sum_{y:\mathsf{wgt}(y)<0.4w} \Pr[Y[a, a+w) = y] \\ &\leq \sum_{y:\mathsf{wgt}(y)<0.4w} (2^{-w} + \varepsilon) \\ &\leq 2^{wh(0.4)} \cdot 2^{-w+1} \\ &\leq \frac{2}{m^2}. \end{aligned}$$

The first inequality holds because Y is ε-almost k-wise independent for every $k \leq m$, and the second inequality follows from a standard upper bound on the volume of the Hamming ball[1] and the choice of ε. Since there are at most m choices for a, a union bound implies that Y fails to satisfy the desired property with probability at most $m \cdot 2/m^2 = 2/m$, as desired. □

Given $x \in \{0,1\}^m$, we compute $\mathsf{Enc}'(x)$ as follows: First, iterate over all $z \in \{0,1\}^t$ until we find a z such that $y = x \oplus g(z)$ satisfies $\mathsf{wgt}(s[a, a+w)) \geq 0.4w$. Such a string z is known to exist by Lemma 5.4 and can be found in time $\text{poly}(m)$ since $t = O(\log m)$ and $g(z)$ is computable in time $\text{poly}(m)$. Then, set $\mathsf{Enc}'(x) = z \| [x \oplus g(z)]$.

Observe that the redundancy of $\mathcal{C}'$ is exactly $|z| = t = O(\log m)$, and that we have encoders and decoders for $\mathcal{C}'$ running in time $\text{poly}(m)$ since $t = O(\log m)$ and $g(z)$ is computable in time $\text{poly}(m)$. To see that $\mathcal{C}'$ satisfies the property required in this section, fix some substring s of $\mathsf{Enc}'(x)$ such that $|s| = \lceil \sqrt{m} \rceil$. Then, $\mathsf{wgt}(s) \geq 0.4w \cdot \lfloor |s|/w \rfloor - t \geq 0.39|s|$ provided that m is large enough.

[1] A standard inequality on the volume of the radius-r N-dimensional Hamming ball $\mathcal{B}_r = \{x \in \{0,1\}^N : \mathsf{wgt}(x) \leq r\}$ states that $|\mathcal{B}_r| \leq 2^{N \cdot h(r/N)}$ when $r \leq N/2$ [204, Proposition 3.3.1].

5.3 Codes with balanced GC-content

We describe next how to adapt the ideas from Section 5.2 and combine them with techniques from [58] in order to construct codes over the alphabet $\{A, C, G, T\}$ that have balanced GC-content and provably require few traces for reconstruction. A string $c \in \{A, C, G, T\}^n$, for n even, has balanced GC-content if $c_i \in \{C, G\}$ for exactly $n/2$ indices i. Strings with balanced GC-content are significantly easier to be synthesised as DNA strands than their non-balanced counterparts [58]. Therefore, constructions accommodating this constraint are well-suited for use in codes for DNA-based data storage. We prove the following result.

Theorem 5.2. *For n large enough there exists an efficiently encodable code $\mathcal{C} \subseteq \{A, C, G, T\}^{n+r}$ with redundancy $r = O\left(\frac{n}{\log n}\right)$ and balanced GC-content which can be efficiently reconstructed from $\exp(O(\log^{2/3} n))$ traces for any constant deletion probability $d < 1$. Moreover, encoding can be performed in nearly-linear time $n \cdot \mathrm{poly}(\log n)$ and reconstruction can be performed in nearly-linear time $n \cdot \exp(O(\log^{2/3} n))$.*

The construction follows the approach outlined in Sections 5.2 (see also Figure 5.1). The only modifications are the choice of markers and the definition of the inner code. We focus on discussing these changes and their properties within the setting of Section 5.2.

We first describe the modified markers. The marker M used throughout the section is of the form $M = (AC)^\ell \| (TG)^\ell$, where $\ell = 25\lceil \log n \rceil$ and n is the message length. Observe that this marker has the same length as the original marker in Section 5.2 and has balanced GC-content.

In order to proceed as in Section 5.2 we need to design an efficiently encodable and decodable inner code $\mathcal{C}' \subseteq \{A, C, T, G\}^{m'}$ with balanced GC-content which satisfies a property analogous to Property 5.1. Suppose that $\mathcal{C}'$ has encoder $\mathsf{Enc}' : \{0, 1\}^m \to \{A, C, T, G\}^{m'}$ and that $m' = m/2 + r$, where $m = \lceil \log^2 n \rceil$ as in Section 5.2 and r denotes the redundancy to be determined (if m is odd, we may pad each block with a 0 while keeping the redundancy $O\left(\frac{n}{\log n}\right)$). Given the composition of M, the property we wish $\mathcal{C}'$ to satisfy is the following:

Property 5.3. *For all $c \in \mathcal{C}'$ and substrings s of c with $|s| = \lceil \sqrt{m} \rceil$, it holds that at least $|s|/3$ symbols of s are T or G.*

Similarly to Lemma 5.2, it can be shown that if $\mathcal{C}'$ satisfies Property 5.3, then with high probability a trace of $c \in \mathcal{C}'$ will not contain long runs consisting only of symbols A and C. As a result, with high probability we can split a trace into multiple sub-traces associated with different blocks as in Section 5.2 (see the high-level reconstruction procedure in Figure 5.1). This is accomplished by looking for all long substrings of the trace consisting only of A's and C's in the trace. The reason is that, with high probability, each such substring consists of the trace of an $(AC)^\ell$ substring from a marker M possibly with some extra symbols prepended. In that case we can correctly identify the separation between the traces of $(AC)^\ell$ and $(TG)^\ell$ in all markers by looking for the first T or G after every sufficiently long substring containing A's and C's only.

We proceed to describe the encoder Enc' of the inner code $\mathcal{C}'$ with redundancy $r = O(\log m)$. We combine a technique from [58] with the code from Section 5.2.1. As an additional ingredient in the construction, we require an efficiently encodable and decodable binary balanced code[2] $\mathcal{C}_1$ with encoder $\mathsf{Enc}_1 : \{0,1\}^{m/2} \to \{0,1\}^{m/2+r_1}$. Nearly-optimal efficient constructions of such codes are known with redundancy $r_1 = O(\log m)$ [205, 206, 207]. Let $\mathcal{C}_2 \subseteq \{0,1\}^{m/2+r_2}$ denote the code from Section 5.2.1 with encoder $\mathsf{Enc}_1 : \{0,1\}^{m/2} \to \{0,1\}^{m/2+r_2}$ and redundancy $r_2 = O(\log m)$. By padding $\mathcal{C}_1$ and/or $\mathcal{C}_2$ appropriately, we may assume that $m/2+r$ is even and $r_1 = r_2 = r$, i.e., that both codes have the same even block length $m/2+r$. To see this, note that we can pad every $c \in \mathcal{C}_1$ with a fixed balanced string of even length $r - r_1$ and every $c \in \mathcal{C}_2$ with the string 1^{r-r_2}. The properties of the two codes are still satisfied after padding. Similarly to [58], we define the bijection $\Psi : \{0,1\}^\ell \times \{0,1\}^\ell \to \{A,C,G,T\}^\ell$ as

$$\Psi(a,b)_i = \begin{cases} A, \text{ if } (a_i,b_i) = (0,0), \\ T, \text{ if } (a_i,b_i) = (0,1), \\ C, \text{ if } (a_i,b_i) = (1,0), \\ G, \text{ if } (a_i,b_i) = (1,1), \end{cases}$$

for $i \in [\ell]$. The code $\mathcal{C}'$ is defined via an encoding $\mathsf{Enc}' : \{0,1\}^m \to \{A,C,G,T\}^{m/2+r}$ satisfying

$$\mathsf{Enc}'(x) = \Psi(\mathsf{Enc}_1(x^{(1)}), \mathsf{Enc}_2(x^{(2)})),$$

where $x = x^{(1)} \| x^{(2)} \in \{0,1\}^{m/2} \times \{0,1\}^{m/2}$. Decoding x from $\mathsf{Enc}'(x)$ can be performed efficiently since we can efficiently decode $x^{(i)}$ from $\mathsf{Enc}_i(x^{(i)})$, $i = 1,2$, and Ψ is a bijection. We have the following lemma.

Lemma 5.5. *The inner code $\mathcal{C}'$ has balanced GC-content and satisfies Property 5.3.*

[2] A code $\mathcal{C} \subseteq \{0,1\}^\ell$ with ℓ even is said to be *balanced* if every $c \in \mathcal{C}$ has Hamming weight $\ell/2$.

Proof. Suppose that $c = \Psi(c_1, c_2)$, where $c_1 \in \mathcal{C}_1$ and $c_2 \in \mathcal{C}_2$. To see that c has balanced GC-content, note that the number of C's and G's in c equals $\mathsf{wgt}(c_1)$. We have $\mathsf{wgt}(c_1) = |c_1|/2$ since $\mathcal{C}_1$ is a balanced code, and so c has balanced GC-content. To verify that $\mathcal{C}$ satisfies Property 5.3, note that the number of T's and G's within a substring $c[i, j)$ equals $\mathsf{wgt}(c_2[i, j))$. Since $\mathcal{C}_2$ satisfies Property 5.1, the desired statement follows. □

Given Lemma 5.5, we can now proceed along the steps described in Section 5.2 by splitting a trace of $\mathcal{C}$ into many short sub-traces associated with different blocks, and then applying a worst-case trace reconstruction algorithm on each block. We remark that although the algorithm from Theorem 2.10 works for worst-case trace reconstruction over binary strings, it can be easily adapted to recover quaternary strings with the same number of traces and twice the error probability as discussed in Remark 5.3. Taking into account the previous discussion, applying the reasoning from Section 5.2 to the marker M and inner code $\mathcal{C}'$ defined in this section leads to Theorem 5.2.

5.4 Coded trace reconstruction from average-case trace reconstruction

In Section 5.2, we gave a construction of efficient marker-based codes. A simple property of the inner code ensured that we could correctly identify all markers with high probability, effectively dividing the global trace into many independent, shorter traces (see Figure 5.1). Subsequently, we applied the state-of-the-art worst-case trace reconstruction algorithm from Theorem 2.10 on each short trace in order to obtain the desired codes. It seems plausible, however, that one could design the inner code more carefully so that significantly fewer traces are needed to recover the short codewords contained between the markers. This is the main problem we address in this section. We design a code that, when used as the inner code in the construction from Section 5.2, leads to an almost exponential reduction of the number of traces required for reconstruction without hurting the rate, provided that the deletion probability is a sufficiently small constant. The trace reconstruction algorithm we use is a variation of the HMPW *average-case* trace reconstruction algorithm from [59] presented in Section 2.5.3.1.

Our starting point is a low redundancy code with the property that it can be reconstructed from poly(n) traces. We discuss this construction in Section 5.4.1, and it leads to the following result.

Theorem 5.3. *There is an absolute constant $d^\star \in (0, 1)$ such that the following holds for every constant*

$d \leq d^\star$: For n large enough there is an efficiently encodable code $\mathcal{C} \subseteq \{0,1\}^{n+r}$ with redundancy $r = O(\log n)$ which can be efficiently reconstructed with probability $1 - \exp(-\Omega(n))$ from $\mathrm{poly}(n)$ traces with deletion probability d.

In Section 5.4.2, we show how to adapt this code so that it can be successfully used as an inner code in the marker-based construction introduced in Section 5.2. Consequently, we obtain the following result.

Theorem 5.4. *There is an absolute constant $d^\star \in (0,1)$ such that the following holds for every constant $d \leq d^\star$: For n large enough there is an efficiently encodable code $\mathcal{C} \subseteq \{0,1\}^{n+r}$ with redundancy $r = O\left(\frac{n}{\log n}\right)$ which can be efficiently reconstructed from $\mathrm{poly}(\log n)$ traces with deletion probability d. Moreover, encoding and reconstruction can be performed in nearly-linear time $n \cdot \mathrm{poly}(\log n)$.*

5.4.1 Low redundancy codes reconstructible from polynomially many traces

We prove Theorem 5.3 in this section. Our code encodes n-bit messages into codewords that are *almost* w-subsequence-unique for $w = O(\log n)$, in the sense that all but the first $O(\log n)$ bits of the codeword comprise a w-subsequence-unique string (recall Definition 2.18). This is possible because an ε-almost k-wise independent random variable over $\{0,1\}^n$ with the appropriate parameters is w-subsequence-unique with high probability. We make this statement rigorous in the following lemma, which uses a standard derandomisation technique. For example, this technique has also been used in [19] to generate strings satisfying related but different properties with high probability.

Lemma 5.6. *Let $g : \{0,1\}^t \to \{0,1\}^m$ be the function guaranteed by Corollary 2.1 with $\varepsilon = 2^{-10w}$ for $w = 100\lceil \log m \rceil$ (hence $t = O(\log m)$). Fix some $x \in \{0,1\}^m$ and define the random variable $Y = x \oplus g(U_t)$. Then, for m large enough it holds that Y is w-subsequence-unique with probability at least $1 - m^{-45}$.*

Proof. First, note that Y is ε-almost k-wise independent for every $k \leq m$. This proof follows along the lines of the proof that a uniformly random string is w-subsequence-unique with high probability [59, Lemma 2.4].

Without loss of generality we may fix a and b such that $a < b$ (otherwise reverse Y) and indices $b \leq i_1 < i_2 < \cdots < i_w \leq b + 1.1w - 1$. Moreover, let $\mathcal{S} = \{i_1, \ldots, i_w\}$ and $\alpha = |\mathcal{S} \cap [a, a+w)|$. Note that

$$\Pr[Y_{\mathcal{S}} = Y[a, a+w)] = \sum_{u \in \{0,1\}^w} \Pr[Y_{\mathcal{S}} = u, Y[a, a+w) = u]. \tag{5.4}$$

Call a string u *consistent* if $\Pr[Y_{\mathcal{S}} = u, Y[a, a+w) = u] > 0$. We claim that the number of consistent strings is at most $2^{w-\alpha}$. To see this, suppose that $y_{\mathcal{S}} = y[a, a+w)$. Note that if $i_j \in \mathcal{S} \cap [a, a+w)$, then $i_j = \ell$ for some $\ell \in [a, a+w)$, and so $y_\ell = y_{i_j} = y_{a+j-1}$. Moreover, we have $\ell > a+j-1$ since $\ell = i_j \geq b+j-1 > a+j-1$. Consequently, all α bits y_ℓ for $\ell \in \mathcal{S} \cap [a, a+w)$ are deterministic functions of bits of $y[a, a+w)$ outside $\mathcal{S} \cap [a, a+w)$. Observe also that

$$\Pr[Y_{\mathcal{S}} = u, Y[a, a+w) = u] \leq 2^{-2w+\alpha} + \varepsilon,$$

since Y is ε-almost k-wise independent for every $k \leq m$ and $2w - \alpha$ bits of Y are fixed in this event. Combining (5.4) with the observations above yields

$$\begin{aligned} \Pr[Y_{\mathcal{S}} = Y[a, a+w)] &\leq 2^{w-\alpha}(2^{-2w+\alpha} + \varepsilon) \\ &\leq 2^{-w} + 2^{w}\varepsilon \\ &\leq 2^{-w+1}. \end{aligned}$$

Since there are $\binom{1.1w}{w}$ choices for $\mathcal{S}$ for each pair (a, b) and fewer than m^2 possible pairs (a, b), the probability that Y is not w-subsequence-unique is at most

$$\begin{aligned} m^2 \binom{1.1w}{w} 2^{-w+1} &\leq m^2 (11e)^{0.1w} 2^{-w+1} \\ &\leq 2m^2 (1.415)^{-w} \\ &\leq m^{-45}, \end{aligned}$$

where the first inequality uses the fact that $\binom{n}{k} \leq \left(\frac{en}{k}\right)^k$. □

Lemma 5.6 leads to a simple, efficient candidate construction of the encoder Enc. Given $x \in \{0,1\}^n$, we first iterate over all $z \in \{0,1\}^t$ until we find z such that $x \oplus g(z)$ is w-subsequence-unique. Most strings z satisfy this, according to Lemma 5.6. By the choice of parameters and Corollary 2.1, we have $t = O(\log n)$ and $g(z)$ computable in time $\mathrm{poly}(n)$. Therefore, we can iterate over all such z, compute $g(z)$ in time $\mathrm{poly}(n)$, and verify whether $x \oplus g(z)$ is w-subsequence-unique for each z in time $\mathrm{poly}(n)$ too. The latter can be accomplished by exhaustive search over all valid pairs of substrings $[a, a+w)$ and $[b, b+1.1w)$, which are fewer than n^2, and over all $\binom{1.1w}{w} = \mathrm{poly}(n)$ length-w subsequences of $[b, b+1.1w)$, since $w = O(\log n)$. Then, the encoder Enc for $\mathcal{C}$ maps a message $x \in \{0,1\}^n$ to the codeword

$$\mathsf{Enc}(x) = z\|[x \oplus g(z)] \in \{0,1\}^{t+n},$$

where z is the first string (in lexicographic order) such that $x \oplus g(z)$ is w-subsequence-unique. Observe that the redundancy of $\mathcal{C}$ is $t = O(\log n)$.

5.4.1.1 The trace reconstruction algorithm

We describe next an efficient trace reconstruction algorithm for $\mathcal{C}$ that works whenever the deletion probability is a small enough constant, thus proving Theorem 5.3. This algorithm is a slight modification of the HMPW algorithm from [59] discussed at a high level in Section 2.5.3.1. Before proceeding, we introduce a definition and basic related results from [59]. Given integers i and j and a deletion probability d, we denote the probability that the i-th bit of a string appears as the j-th bit of its trace by $P(i,j)$. Then, we have

$$P(i,j) = \binom{i-1}{j-1}(1-d)^j d^{i-j}.$$

The following lemma states some useful properties of $P(i,j)$.

Lemma 5.7 ([59, Lemma 2.1])**.** *If $d < 1/3$ and $j \leq (1-3d)i$, then $\sum_{i'>i} P(i',j) \leq \frac{1}{2}P(i,j)$. Furthermore, if $(1-4d)i \leq j \leq (1-3d)i$ we have $P(i,j) \geq \exp(-7di)$.*

Set $w = 100\lceil \log n \rceil$, $v = \lceil w/d \rceil = \Theta(\log n)$ and $j = \lfloor (v - 0.1w)(1-3d) \rfloor = \Theta(\log n)$, where d is a small enough constant. Given a codeword $c = \mathsf{Enc}(x) = z\|[x \oplus g(z)]$, we begin by using the same bootstrapping method as the HMPW algorithm: We apply the algorithm from Lemma 2.10 to recover z and the first $\ell' = 2v + 1.1w + |z| = O(\log n)$ bits of $y = x \oplus g(z)$ with probability at least $1 - \exp(-\Omega(n))$ using $\mathrm{poly}(n)$ traces and time. If $n \leq \ell'$, we are done, so assume from here onwards that n is large enough so that $n > \ell' \geq 2$. We note that when we later attempt to use this code as the inner code of a marker-based construction, we will need to, among other things, modify both the code and the bootstrapping method so that trace reconstruction is still possible with few traces.

Now, suppose that we know z and $y_1, \dots, y_{i-1}$ for $i - 1 \geq \ell'$. We show how to recover y_i with probability at least $1 - 2n\exp(-n)$ using $2^{1000w+1} = \mathrm{poly}(n)$ traces and runtime upper bounded by some $\mathrm{poly}(n)$ independent of i. Let T denote a trace of c. As in [59], we will look for a matching of $y[i-v-w, i-v)$ within T. However, we shall discard matchings that occur too early in T. More precisely, suppose that $y[i-v-w, i-v)$ is matched with $T[u-w, u)$. We call such a matching *good* if $u - w > |z|$. This definition ensures that all trace bits in a good matching come from y. If T does not contain a good matching of $y[i-v-w, i-v)$, we discard it. Otherwise, if the first good matching occurs at $T[u-w, u)$, we let $\mathsf{Suff} = T[u:]$ and discard the remaining bits of T. The following two lemmas establish a result analogous to [59, Lemma 2.2] via the same approach. We denote the event that a good matching of $y[i-v-w, i-v)$ occurs in T by E_{good} and by Last the position in y of the last bit appearing in the first good matching within T (with $\mathsf{Last} = \perp$ if no good matching exists).

Lemma 5.8. *Fix constant $d < 1/2$ and assume n is large enough. Then, we have* $\Pr[E_{\mathsf{good}}] \geq \Pr[E_{\mathsf{good}}, \mathsf{Last} < i] \geq 2^{-(w+1)}$ *for every $\ell' < i \leq n$.*

Proof. First, observe that the probability that no bit in $y[i-v-w, i-v)$ is deleted is exactly $(1-d)^w \geq 2^{-w}$ if $d < 1/2$. Given this, suppose that y_{i-v-w} appears in position T_P for some random variable $P \in [|z|+n]$. Then, the probability that the given matching is good equals $\Pr[P > |z|]$. We have $i - v - w \geq |z|$ since we have already learned the first $\ell' = 2v + 1.1w + |z|$ bits of y. Therefore, we have

$$\Pr[P \leq |z|] \leq \Pr[\mathsf{Bin}_{2|z|,1-d} \leq |z|] \leq 1/2,$$

where the last inequality holds if $d < 1/2$. Concluding, E_{good} and $\mathsf{Last} < i$ hold with probability at least $1/2 \cdot 2^{-w} = 2^{-(w+1)}$. □

Lemma 5.9. *Fix constant $d < \frac{2^{-10^4}}{11e}$ and assume n is large enough. Then, we have* $\Pr[\mathsf{Last} \notin [i-v-1, i-v-1+0.1w] | E_{\mathsf{good}}] \leq 2^{-100w}$ *for every $\ell' < i \leq n$.*

Proof. Let E denote the event that $\mathsf{Last} \notin [i-v-1, i-v-1+0.1w]$. Then, we wish to upper bound $\Pr[E|E_{\mathsf{good}}]$. By Lemma 5.8, we have $\Pr[E|E_{\mathsf{good}}] = \frac{\Pr[E, E_{\mathsf{good}}]}{\Pr[E_{\mathsf{good}}]} \leq 2^{w+1} \cdot \Pr[E, E_{\mathsf{good}}]$. We now show that $\Pr[E, E_{\mathsf{good}}] \leq 2^{-200w}$, which concludes the argument.

The probability that E and E_{good} hold simultaneously is at most the probability that more than $0.1w$ bits are deleted from some substring $y[b, b+1.1w)$. To see this, first note that the bits in a good matching must come from y. If at most $0.1w$ bits are deleted from every substring $y[b, b+1.1w)$, then the w bits of the good matching in T of $y[i-v-w, i-v)$ must be a subsequence of $y[b, b+1.1w)$ for some b, which means $y[i-v-w, i-v)$ appears as a subsequence of $y[b, b+1.1w)$. Since y is w-subsequence-unique, for this to happen we must have $b \leq i-v-w$ and $b+1.1w \geq i-v$. Now, suppose that $\mathsf{Last} \notin [i-v-1, i-v-1+0.1w]$. In particular, this implies that $\mathsf{Last} \notin [i-v-1, b+1.1w-1]$ because $i-v-1+0.1w \geq b+1.1w-1 \geq i-v-1$. Then, it must be the case that $y[i-v-w, i-v)$ is a subsequence of $y[b, i-v-1)$. Hence, $y[i-v-w, i-v)$ is a subsequence of $y[i-v-1-1.1w, i-v-1)$ too since we know that $b \geq i-v-1.1w$ (note that $i-v-1-1.1w \geq v \geq 1$ since $i \geq 2v+1.1w+|z|+1$, so this choice is well-defined). However, this violates the w-subsequence-uniqueness of y.

The probability that at least $0.1w$ bits are deleted from $y[b, b+1.1w)$ for a fixed b is at most $\binom{1.1w}{w} d^{0.1w} \leq (11ed)^{0.1w} \leq 2^{-1000w}$ by the upper bound on d. Combining this with a union bound over the fewer than $n \leq 2^w$ choices for b, we conclude that $\Pr[E, E_{\mathsf{good}}] \leq n2^{-1000w} \leq 2^{-200w}$. □

We proceed analogously to [59]. Fix a constant deletion probability $d < \frac{2^{-10^4}}{11e}$ and assume n is large enough. Then, for every $\ell' < i \leq n$ we may write

$$\begin{aligned}\Pr[\mathsf{Suff}_j = 1|E_{\mathsf{good}}] &= \sum_{r=1}^{n} \Pr[\mathsf{Last} = r|E_{\mathsf{good}}] \Pr[\mathsf{Suff}_j = 1|\mathsf{Last} = r, E_{\mathsf{good}}] \\ &= \varepsilon_i(c) + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r|E_{\mathsf{good}}] \Pr[\mathsf{Suff}_j = 1|\mathsf{Last} = r, E_{\mathsf{good}}] \qquad (5.5)\end{aligned}$$

for some $\varepsilon_i(c) \in [0, 2^{-100w}]$ by Lemma 5.9. Note that once $\mathsf{Last} = r$ is fixed, the random variable Suff corresponds to the trace of $y[r+1:]$, which is unaffected by E_{good}. As a result, we have

$$\Pr[\mathsf{Suff}_j = 1|\mathsf{Last} = r, E_{\mathsf{good}}] = \Pr[\mathsf{Suff}_j = 1|\mathsf{Last} = r] = \sum_{\ell=r+1}^{n} P(\ell - r, j) \cdot y_\ell, \qquad (5.6)$$

where we recall that $P(\ell, j)$ denotes the probability that the ℓ-th bit ends up in the j-th position of the trace. Combining (5.5) and (5.6) yields

$$\begin{aligned}\Pr[\mathsf{Suff}_j = 1|E_{\mathsf{good}}] &= \varepsilon_i(c) + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r|E_{\mathsf{good}}] \sum_{\ell=r+1}^{n} P(\ell - r, j) y_\ell \\ &= \varepsilon_i(c) + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r|E_{\mathsf{good}}] \sum_{\ell=r+1}^{i-1} P(\ell - r, j) y_\ell \\ &+ \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r|E_{\mathsf{good}}] \cdot P(i-r, j) y_i \\ &+ \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r|E_{\mathsf{good}}] \cdot \sum_{\ell=i+1}^{n} P(\ell - r, j) y_\ell.\end{aligned}$$

Since $(1-3d)(i-r) > (1-3d)(v-0.1w) \geq j$ when $i-v-1 \leq r \leq i-v-1+0.1w$, the first part of Lemma 5.7 implies that $\sum_{\ell=i+1}^{n} P(\ell - r, j) \leq \frac{1}{2} P(i-r, j)$. Therefore, we have the threshold property

$$y_i = 0 \implies \Pr[\mathsf{Suff}_j = 1 | E_{\mathsf{good}}] \leq \varepsilon_i(c) + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r | E_{\mathsf{good}}] \sum_{\ell=r+1}^{i-1} P(\ell - r, j) y_\ell + \frac{1}{2} \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r | E_{\mathsf{good}}] P(i - r, j) \quad (5.7)$$

and

$$y_i = 1 \implies \Pr[\mathsf{Suff}_j = 1 | E_{\mathsf{good}}] \geq \varepsilon_i(c) + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r | E_{\mathsf{good}}] \sum_{\ell=r+1}^{i-1} P(\ell - r, j) y_\ell + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r | E_{\mathsf{good}}] P(i - r, j). \quad (5.8)$$

By the choice $j = \lfloor (1 - 3d)(v - 0.1w) \rfloor$, we have

$$(1 - 4d)(i - r) < j \leq (1 - 3d)(i - r)$$

for every $\ell' < i \leq n$ and $i - v - 1 \leq r \leq i - v - 1 + 0.1w$. Therefore, the second part of Lemma 5.7 implies that

$$P(i - r, j) \geq e^{-7d(i-r)} \geq e^{-7d(v+1)} \geq e^{-8w} \geq 2^{-12w}$$

for every such i and r, since $i - r \leq v + 1 \leq w/d + 2$. Since $\sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r | E_{\mathsf{good}}] \geq 1 - 2^{-100w} > 1/2$ by Lemma 5.9, this discussion shows that the gap between the right hand sides of (5.7) and (5.8) is at least $2^{-12w} \cdot \frac{1}{2} \cdot \frac{1}{2} = 2^{-(12w+2)}$.

Finally, we argue that we can produce good estimates of the terms $\Pr[\mathsf{Suff}_j = 1 | E_{\mathsf{good}}]$ and $\Pr[\mathsf{Last} = r | E_{\mathsf{good}}]$ with high probability using $\mathrm{poly}(n)$ traces and time. This allows us to decide whether $y_i = 0$ or $y_i = 1$ based on (5.7) and (5.8).

Lemma 5.10. *Fix constant $d < 1/2$ and assume n is large enough. Then, for every $\ell' < i \leq n$, with knowledge of z and $y_1, \ldots, y_{i-1}$ we can output an estimate $\widehat{p}$ of $p = \Pr[\mathsf{Suff}_j = 1 | E_{\mathsf{good}}]$ satisfying $|\widehat{p} - p| \leq 2^{-100w}$ with probability at least $1 - 2\exp(-n)$ using $2^{1000w+1} = \mathrm{poly}(n)$ traces in time at most $\mathrm{poly}(n)$ independent of i.*

Proof. If $d < 1/2$ is constant and n is large enough, Lemma 5.8 and the Hoeffding bound ensure that for every $\ell' < i \leq n$ the corresponding event E_{good} holds for at least 2^{900w} out of $2^{1000w+1}$ i.i.d. traces with probability at least $1 - \exp(-n)$. Assuming knowledge of z and $y_1, \ldots, y_{i-1}$, we can check whether E_{good} occurs for a given i in a given trace T in time at most $\mathrm{poly}(n)$ independent of i by looking for a matching of $y[i-v-w, i-v)$ starting at some position $u > |z|$ of T. Suppose we run this procedure for a given i over all $2^{1000w+1}$ traces and identify at least $N = 2^{900w}$ traces for which E_{good} holds; call the first N such traces $T^{(1)}, \dots, T^{(N)}$. Then, we compute $\widehat{p} = \frac{1}{N}\sum_{\ell=1}^{N} \mathsf{Suff}_j^{(\ell)}$, where $\mathsf{Suff}_j^{(\ell)}$ denotes the value of Suff_j in $T^{(\ell)}$. We have $\mathbb{E}[\widehat{p}] = p$, and so the Hoeffding bound yields

$$\Pr[|\widehat{p} - p| > 2^{-100w}] \leq 2\exp(-2 \cdot 2^{-200w} N) \leq 2\exp(-2^{700w}) \leq \exp(-n).$$

A union bound over the two events above shows that the procedure succeeds for a given i with probability at least $1 - 2\exp(-n)$, and its runtime is upper bounded by a polynomial $\mathrm{poly}(n)$ independent of i. □

Lemma 5.11. *Fix constant $d < \frac{2^{-10^4}}{11e}$ and assume n is large enough. Then, for every $\ell' < i \leq n$ and $1 \leq r < i$, with knowledge of z and $y_1, \dots, y_{i-1}$ we can output an estimate $\widehat{p_r}$ of $p_r = \Pr[\mathsf{Last} = r | E_{\mathsf{good}}]$ satisfying $|\widehat{p_r} - p_r| \leq 2^{-90w}$ with probability at least $1 - 2\exp(-n)$ in time at most $\mathrm{poly}(n)$ independent of i and r.*

Proof. We begin by simulating $2^{1000w+1}$ independent traces of the substring $z \| (y_1, y_2, \dots, y_{i-1})$ in time $\mathrm{poly}(n)$ assuming access to at most $2n \cdot 2^{1000w+1} = \mathrm{poly}(n)$ i.i.d. samples of Ber_d, and record the value of Last for each such trace featuring a good matching. Then, we select in time $\mathrm{poly}(n)$ a subset of $N = 2^{900w}$ traces $T^{(1)}, \dots, T^{(N)}$ for which E_{good} and $\mathsf{Last} < i$ hold simultaneously. If $d < 1/2$ and n is large enough, such a subset exists with probability at least $1 - \exp(-n)$ for every $\ell' < i \leq n$ via Lemma 5.8 and the Hoeffding bound. For any such i, we can check whether E_{good} and $\mathsf{Last} < i$ hold for a given trace in time at most $\mathrm{poly}(n)$ independent of i using $z \| (y_1, y_2, \dots, y_{i-1})$.

If this subset exists, we compute $\widehat{p_r} = \frac{1}{N}\sum_{\ell=1}^{N} \mathbf{1}_{\{\mathsf{Last}^{(\ell)} = r\}}$, where $\mathsf{Last}^{(\ell)}$ denotes the value of Last for $T^{(\ell)}$. Observe that $\mathbb{E}[\widehat{p_r}] = p'_r = \Pr[\mathsf{Last} = r | E_{\mathsf{good}}, \mathsf{Last} < i]$. Therefore, the Hoeffding bound ensures that $|\widehat{p_r} - p'_r| \leq 2^{-100w}$ with probability at least $1 - \exp(-n)$. By Lemma 5.9 and the upper bound on d, if n is large enough we have $\Pr[\mathsf{Last} < i | E_{\mathsf{good}}] \geq 1 - 2^{-100w}$ for every $\ell' < i \leq n$. As a result, it holds that $|p_r - p'_r| \leq 2 \cdot 2^{-100w}$ for n large enough, which combined with the above implies that $|\widehat{p_r} - p_r| \leq 3 \cdot 2^{-100w} \leq 2^{-90w}$. A union bound over the two events above ensures that the procedure succeeds for a given i with probability at least $1 - 2\exp(-n)$, and its runtime is upper bounded by a polynomial $\mathrm{poly}(n)$ independent of i and r. □

Fix constant $d < \frac{2^{-10^4}}{11e}$ and suppose n is large enough independent of i (but depending on d). Then, Lemmas 5.10 and 5.11 along with a union bound imply that we can, for every $\ell' < i \leq n$, approximate the left and right hand sides of (5.7) and (5.8) to within additive error 2^{-100w} and $2^{-100w}+2n\cdot 2^{-90w} \leq 2^{-80w}$, respectively, using z and $y_1,\dots,y_{i-1}$ with probability at least $1-2n\exp(-n)$ from $2^{1000w+1}$ traces of c in time at most $\mathrm{poly}(n)$ independent of i. Recalling that the gap between the right hand sides of (5.7) and (5.8) is at least $2^{-(12w+2)}$ for every $\ell' < i \leq n$, this means we can efficiently recover each y_i iteratively from $2^{1000w+1}$ traces with probability at least $1-2n\exp(-n)$. Applying a union bound over all $\ell' < i \leq n$ implies that we correctly recover all remaining bits with probability at least $1-\exp(-\Omega(n))$ in overall time $\mathrm{poly}(n)$ from $n2^{1000w+1} = \mathrm{poly}(n)$ traces, which concludes the proof.

Remark 5.5. The procedure in Lemma 5.11 requires access to at most $2n\cdot 2^{1000w+1}$ i.i.d. samples from Ber_d. Therefore, in total the reconstruction algorithm presented above must have access to at most $2n^3\cdot 2^{1000w+1}$ such i.i.d. samples. However, we can make the algorithm deterministic by extracting this randomness efficiently from an extra $\mathrm{poly}(n)$ traces while adding an $\exp(-\Omega(n))$ term to the reconstruction error probability.

First, we can sample B satisfying $\Delta(B;\mathsf{Ber}_d) \leq 2^{-n}$ (recall that Δ denotes statistical distance) from $n' = n+1$ i.i.d. samples of $\mathsf{Ber}_{1/2}$ in a standard manner: Let d' be d truncated to the first n' bits of its binary expansion, i.e., $d' = \sum_{i=1}^{n'} b_i 2^{-i}$ if $d = \sum_{i=1}^{\infty} b_i 2^{-i}$. Note that $\Delta(\mathsf{Ber}_{d'};\mathsf{Ber}_d) = |d-d'| \leq 2^{-n'}$. Now, let $D_1, D_2, \dots, D_{n'}$ be i.i.d. according to $\mathsf{Ber}_{1/2}$, and set $B=1$ if $\sum_{i=1}^{n'} D_i 2^{-i} \leq \sum_{i=1}^{n'} b_i 2^{-i} = d'$ and $B=0$ otherwise. Then, we have $\Delta(B;\mathsf{Ber}_{d'}) = \Pr[\forall i \in [n'] : D_i = b_i] = 2^{-n'}$, and so $\Delta(B;\mathsf{Ber}_d) \leq 2^{-n'} + 2^{-n'} = 2^{-n}$ by the triangle inequality. Second, we can sample B' satisfying $\Delta(B';\mathsf{Ber}_{1/2}) \leq \frac{|1-2d|^n}{2}$ from one trace T of any n-bit string: It suffices to let $B'=1$ if $|T|$ is even and $B'=0$ otherwise. Since $|T| \sim \mathsf{Bin}_{n,1-d}$, we have that $\Pr[B'=1] = \frac{1}{2} + \frac{(2d-1)^n}{2}$, which yields the desired result.

To conclude, suppose the algorithm uses $\tau \leq 2n^3\cdot 2^{1000w} = \mathrm{poly}(n)$ i.i.d. samples $S_1, S_2, \dots, S_\tau$ of Ber_d in total. From the discussion above, for each $i \in [\tau]$ we can use n' extra traces to sample i.i.d. random variables $B'_{i,1},\dots,B'_{i,n'}$ satisfying $\Delta((B'_{i,1},\dots,B'_{i,n'});U_n) \leq \frac{n'|1-2d|^n}{2}$ by the triangle inequality, where U_n is uniformly distributed over $\{0,1\}^n$. Therefore, by the discussion above we can use $B'_{i,1},\dots,B'_{i,n'}$ instead of n' i.i.d. samples of $\mathsf{Ber}_{1/2}$ to sample B_i satisfying $\Delta(B_i;S_i) \leq \frac{n'|1-2d|^n}{2} + 2^{-n}$ by the triangle inequality. In total, we use $n'\cdot\tau = \mathrm{poly}(n)$ extra traces to sample i.i.d. bits $B_1, B_2, \dots, B_\tau$ satisfying $\Delta((B_1,\dots,B_\tau);(S_1,\dots,S_\tau)) \leq \tau\left(\frac{n'|1-2d|^n}{2} + 2^{-n}\right) = \exp(-n)$ when d is constant. Finally, consider the modified deterministic reconstruction algorithm that first samples $\tau\cdot n'$ traces, computes $B_1, B_2, \dots, B_\tau$ as above, and then runs the reconstruction algorithm described in this section using the B_i's instead of the S_i's. This modified algorithm still uses $\mathrm{poly}(n)$ traces and runs in time $\mathrm{poly}(n)$. Moreover, its reconstruction error probability is within $\Delta((B_1, \dots, B_\tau); (S_1, \dots, S_\tau)) \leq \exp(-\Omega(n))$ of that of the original reconstruction algorithm, meaning it is still $\exp(-\Omega(n))$.

5.4.2 Using the code within a marker-based construction

In this section, we combine the constructions from Sections 5.2 and 5.4.1 with some additional modifications in order to prove Theorem 5.4. The basic idea is that we would like to use the code designed in Section 5.4.1 as the instantiation of the inner code $\mathcal{C}'$ in the construction of $\mathcal{C}$ in Section 5.2 (see Figure 5.1). Then, we could combine the high-level trace reconstruction procedure illustrated there with the trace reconstruction algorithm from Section 5.4.1.1 on each sub-trace and mitigate the use of worst-case trace reconstruction algorithms. This idea does not work as is, but some modifications to the code from Section 5.4.1 will allow the argument to go through.

The first issue we must address is that the inner code $\mathcal{C}'$ must satisfy Property 5.1. If this property holds, then the reasoning of Section 5.2 implies that we can focus on the trace reconstruction problem for strings of the form $1^\ell \| c \| 0^\ell$, where $c \in \mathcal{C}'$ has length $\Theta(\log^2 n)$ and $\ell = \Theta(\log n)$, as long as we use fewer than n traces (recall Lemma 5.3 and (5.3)). If we were to directly apply the trace reconstruction algorithm from Section 5.4.1.1, we would run into a problem. For the aforementioned algorithm to work, we need to bootstrap it by recovering the first few bits of c using the procedure from Lemma 2.10. However, in this case c only appears after a run of length $\ell = \Theta(\log n)$. Even though we know the previous bits, we would still require $\mathrm{poly}(n)$ traces to recover the first bit of c in this way, which is not acceptable as we want to use $\mathrm{poly}(\log n)$ traces. Consequently, we need an alternative bootstrapping method. Another issue we need to resolve is that the reconstruction algorithm from Section 5.4.1.1 assumed that all but the first few bits of c lead to a subsequence-unique string. However, this is not the case here, as we must deal with a suffix of the form $c \| 0^\ell$.

Before we proceed to describe a modified version of our code from Section 5.4.1 that avoids the issues raised above, we prove the following lemma.

Lemma 5.12. *Let $g : \{0,1\}^t \to \{0,1\}^m$ be the function guaranteed by Corollary 2.1 with $\varepsilon = 2^{-10w}$ for $w = 100\lceil \log m \rceil$ (hence $t = O(\log m)$). For any $\ell \leq m$ and $x \in \{0,1\}^m$, define the random variable $Y = [x \oplus g(U_t)] \| 0^\ell$. Then, for m large enough we have that Y satisfies the following property with probability at least $1 - m^{-45}$.*

Property 5.4. *For any a and b such that $a+w \leq \min(m+1,b)$ and $b+1.1w \leq m+\ell+1$, we have that $Y[a,a+w)$ is not a subsequence of $Y[b,b+1.1w)$.*

Proof. Fix a pair (a,b) satisfying $a+w \leq \min(m+1,b)$ and $b+1.1w \leq m+\ell+1$, and let $\mathcal{S} \subseteq \{b,\dots,b+1.1w-1\}$ be the set of w distinct indices $i_1 < \cdots < i_w$. Moreover, let $\beta = |\mathcal{S} \cap [m+1,m+\ell]|$. Observe that

$$\Pr[Y_{\mathcal{S}} = Y[a,a+w)] = \sum_{u\in\{0,1\}^w} \Pr[Y_{\mathcal{S}} = u, Y[a,a+w) = u]. \tag{5.9}$$

Call $u \in \{0,1\}^w$ *consistent* if $\Pr[Y_{\mathcal{S}} = u, Y[a,a+w) = u] > 0$. Then, there are at most $2^{w-\beta}$ consistent strings, since we have $y_j = 0$ whenever $i_j \geq m+1$ if $y[m+1,m+\ell] = 0^\ell$ and $y_{\mathcal{S}} = y[a,a+w)$. If u is consistent, and setting $\mathcal{S}' = \mathcal{S} \setminus [m+1,m+\ell]$ and $u' = (u_j)_{i_j \leq m}$, it holds that

$$\Pr[Y_{\mathcal{S}} = u, Y[a,a+w) = u] = \Pr[Y_{\mathcal{S}'} = u', Y[a,a+w) = u] \leq 2^{-2w+\beta} + \varepsilon,$$

since $x \oplus g(U_t)$ is ε-almost k-wise independent for every $k \leq m$ and $2w-\beta$ bits of $x \oplus g(U_t)$ are fixed in the event of the middle expression above. Combining (5.9) with the observations above yields

$$\begin{aligned}\Pr[Y_{\mathcal{S}} = Y[a,a+w)] &\leq 2^{w-\beta}(2^{-2w+\beta} + \varepsilon)\\ &\leq 2^{-w} + 2^w\varepsilon\\ &\leq 2^{-w+1}.\end{aligned}$$

Since there are fewer than $2m^2$ possible pairs (a,b) and $\binom{1.1w}{w}$ choices of $\mathcal{S}$ for each pair, a union bound shows that the probability that the desired event does not occur is at most $2m^2\binom{1.1w}{w}2^{-w+1} \leq m^{-45}$, provided m is large enough. □

Intuitively, Lemma 5.12 guarantees that $x \oplus g(U_t)$ satisfies a stronger form of subsequence-uniqueness with high probability. In fact, not only is $x \oplus g(U_t)$ w-subsequence-unique with high probability based on Lemma 5.6, but it is also impossible to find a substring of $x \oplus g(U_t)$ that is a subsequence of $[x \oplus g(U_t)]\|0^\ell$ elsewhere.

We are now ready to describe our modified inner code $\mathcal{C}'$ with encoder $\mathsf{Enc}' : \{0,1\}^m \to \{0,1\}^{m+r'}$. On input a message $x \in \{0,1\}^m$ for m large enough, Enc' operates as follows:

1. Set $x' = 0^{\ell'}\|x$ for $\ell' = 10\ell = O(\sqrt{m})$. Let $m' = |x'|$ and set $w = 100\lceil \log m' \rceil$;

2. Iterate over all $z \in \{0,1\}^t$ for $t = O(\log m') = O(\log m)$ until a z such that $x' \oplus g(z)$ is w-subsequence-unique and also satisfies Properties 5.2 and 5.4 is found. Such a string z is guaranteed to exist because all such properties hold for $x' \oplus g(U_t)$ with probability $1 - o(1)$ (see Lemmas 5.4, 5.6, and 5.12). Moreover, computing $g(z)$ and checking whether $x' \oplus g(z)$ satisfies all three properties can be done in time $\mathrm{poly}(m)$;

3. Obtain z' from z by setting $z' = \mathsf{Enc}_{\mathsf{edit}}(0\|z)$, where $\mathsf{Enc}_{\mathsf{edit}}$ is the efficient encoder of the systematic code $\mathcal{C}_{\mathsf{edit}}$ from Theorem 2.8 correcting $|z|/10$ edit errors with redundancy at most $\beta_{\mathsf{edit}}|z| = O(\log m)$ for some constant $\beta_{\mathsf{edit}} > 0$. Here, d is assumed to be a constant satisfying $5d(1+\beta_{\mathsf{edit}}) < 1/10$, so that $\mathcal{C}_{\mathsf{edit}}$ corrects a $5d$-fraction of edit errors in z';

4. Define $\mathsf{Enc}'(x) = z'\|[x' \oplus g(z)] = z'\|y'$.

For a given message $x \in \{0,1\}^m$, we can compute $\mathsf{Enc}'(x)$ in time $\mathrm{poly}(m)$. Furthermore, recalling that $m = \lceil \log^2 n \rceil$ in the construction of Section 5.2, the redundancy of $\mathcal{C}'$ is

$$r' = |z'| + \ell' = O(\log m + \sqrt{m}) = O(\sqrt{m}) = O(\log n).$$

If we use $\mathcal{C}'$ as the inner code in the construction of $\mathcal{C}$ from Section 5.2, then according to (5.1) we obtain overall redundancy $r = O\left(\frac{n}{\log n}\right)$ for $\mathcal{C}$, as desired. Moreover, $\mathcal{C}'$ satisfies Property 5.1. By the choice of z, we have $\mathsf{wgt}(y'[a, a+w)) \geq 0.4w$ for every a and $w = 100\lceil \log m' \rceil$. Therefore, for any substring s such that $|s| = \lceil \sqrt{m} \rceil$ we have

$$\mathsf{wgt}(s) \geq 0.4w\lfloor |s|/w \rfloor - |z'| \geq 0.39|s|$$

provided that m is large enough, since $|z'|$ and w are $O(\log m)$. As a result, the reasoning used in Section 5.2 applies to this choice of $\mathcal{C}'$. To prove Theorem 5.4, it remains to give a trace reconstruction algorithm to recover strings of the form $1^\ell\|\mathsf{Enc}'(x)\|0^\ell$ from $\mathrm{poly}(m) = \mathrm{poly}(\log n)$ traces with probability at least $1 - n^{-10}$. Suppose we already have such an algorithm, call it $\mathcal{A}$. Recall (5.3) and the definition of the event $E^{(i)}_{\mathsf{indFail}}$ from Section 5.2. Instantiating $E^{(i)}_{\mathsf{indFail}}$ with algorithm $\mathcal{A}$ leads to the bound $\Pr[E^{(i)}_{\mathsf{indFail}}] \leq n^{-10}$, for all i. Combining this observation with (5.3) allows us to conclude that the probability that we successfully recover $c \in \mathcal{C}$ from $\mathrm{poly}(\log n)$ i.i.d. traces of c is at least $1 - 2/n$ for n large enough.

5.4.2.1 The trace reconstruction algorithm

In this section, we analyse an algorithm for recovering strings of the form $1^\ell \| \mathsf{Enc}'(x) \| 0^\ell$ from $\mathrm{poly}(m) = \mathrm{poly}(\log n)$ traces with probability $1 - 1/\mathrm{poly}(n)$. As discussed before, we proceed by adapting the algorithm from Section 5.4.1.1, which in turn is a slightly modified version of the algorithm from [59] discussed in Section 2.5.3.1.

The first difference between the current and the previous setting is that the original bootstrapping technique cannot be applied, as $\mathsf{Enc}'(x)$ is enclosed by two long runs. We show that the structure of Enc' allows for an alternative bootstrapping method. Recall that $c = \mathsf{Enc}'(x) = z' \| y'$, where $y' = x' \oplus g(z)$ and the first $\ell' = O(\sqrt{m})$ bits of x' are zero. Therefore, if we can recover z from few traces of $1^\ell \| c \| 0^\ell$, then we can recover the first $O(\sqrt{m})$ bits of y', which suffice for bootstrapping, by computing $g(z)$. The following lemma states that we can recover z with high probability from $O(\log n)$ traces.

Lemma 5.13. *Fix constant d such that $5d(1 + \beta_{\mathsf{edit}}) < 1/10$ and assume n is large enough. Then, we can recover z from $400 \log n$ traces of $1^\ell \| c \| 0^\ell$ with probability at least $1 - n^{-30}$ in time* $\mathrm{poly}(\log n)$.

Proof. Recall that $z' = \mathsf{Enc}_{\mathsf{edit}}(0 \| z)$ and that $\mathcal{C}_{\mathsf{edit}}$ is systematic. This means that $z'_1 = 0$, and so with probability $1-d$ the first 0 appearing in the trace will correspond to z'_1. Given a trace T of $1^\ell \| c \| 0^\ell$, we proceed as follows: Let u denote the position of the first 0 in T. Then, we take $\tilde{z} = T[u, u + (1-d)|z'|)$, feed $\tilde{z}$ into $\mathsf{Dec}_{\mathsf{edit}}$, and let the corresponding output be our guess for z. The probability that this procedure fails to yield z is at most the probability that z'_1 was deleted plus the probability that $\tilde{z}$ is too far away in edit distance from z' given that z'_1 was not deleted. We proceed to bound both probabilities. First, the probability that z'_1 is deleted is exactly d. Second, we assume z'_1 is not deleted and let L denote the length of the trace of $z'[2:]$ within T. We have $\mathbb{E}[L] = (1-d)(|z'|-1)$, and the Hoeffding bound gives

$$\Pr[L \leq (1-3d)(|z'|-1)] \leq \exp\left(-8d^2(|z'|-1)\right).$$

Since d is a constant, $|z'| = \Theta(\log m)$, and $L \leq |z'| - 1$ always, we conclude that for m large enough we have

$$\Pr[|L - (1-d)(|z'|-1)| \geq 2d(|z'|-1)] < 1/5.$$

As a result, with probability at least $4/5$ we have that $\tilde{z}$ is within edit distance $5d|z'| < |z|/10$ from

z'. If this condition holds, then $\mathsf{Dec}_{\mathsf{edit}}(\tilde{z}) = z$.

In sum, the procedure fails to return z with probability at most $d+1/5 < 1/4$. Repeating this procedure $400 \log n$ times and taking the most common output ensures, via the Hoeffding bound, that we can recover z from $400 \log n$ traces with success probability at least $1 - n^{-30}$ in time $\mathrm{poly}(\log n)$. □

As discussed before, once z has been recovered the bits of $1^\ell \| c \| 0^\ell = 1^\ell \| z' \| y' \| 0^\ell$ are known up to and including the first $\ell' = \Theta(\sqrt{m})$ bits of y'. Our last task is to recover the remaining bits of y', and given that we have sufficiently many initial bits from y' we follow the approach from Section 5.4.1.1. The differences with respect to Section 5.4.1.1 are the following:

- We use $y'' = y' \| 0^\ell$ in place of y and $1^\ell \| z'$ in place of z. In particular, this implies that the threshold used to declare that a matching is good is now different. In this case, if T is a trace of $1^\ell \| c \| 0^\ell$ and $y''[i-v-w, i-v)$ is matched with $T[u-w, u)$, then the matching is *good* if $u - w > \ell + |z'|$. This ensures that the bits in a good matching always come from y'';
- We are only interested in recovering y''_i for $\ell' < i \leq |y'|$, as we already know the other bits of y''.

The two lemmas below are analogous to Lemmas 5.8 and 5.9 with similar proofs. They show that the modified HMPW trace reconstruction algorithm from Section 5.4.1.1 can also be used in this case to recover the remaining bits of y' with the changes itemised above. From here onwards, we use T to denote a trace of $1^\ell \| c \| 0^\ell$, for every $\ell' < i \leq |y'|$ let E_{good} denote the event that a good matching of $y''[i-v-w, i-v)$ occurs in T, and let Last denote the position in y'' of the last bit appearing in the first such good matching within T (with $\mathsf{Last} = \perp$ if no good matching exists). Moreover, we set $v = \lceil w/d \rceil$ and $j = \lfloor (v - 0.1w)(1-3d) \rfloor$ as before (recall $w = 100 \lceil \log m' \rceil$ and $m' = |y'|$).

Lemma 5.14. *Fix constant $d < 1/2$ and assume m is large enough. Then, we have* $\Pr[E_{\mathsf{good}}] \geq \Pr[E_{\mathsf{good}}, \mathsf{Last} < i] \geq 2^{-(w+1)}$ *for every $\ell' < i \leq |y'|$.*

Proof. The probability that no bit in $y''[i-v-w, i-v)$ is deleted is at least 2^{-w} if $d < 1/2$. Given this, suppose that y''_{i-v-w} appears in position T_P. Then, the matching is good with probability $\Pr[P > \ell + |z'|]$. Since $i > \ell' = 10\ell = \Theta(\sqrt{m})$, we have $i - v - w \geq \ell + |z'|$ for m large enough, because v, w, and $|z'|$ are all $\Theta(\log m)$. In that case, it holds that

$$\Pr[P \leq \ell + |z'|] \leq \Pr[\mathsf{Bin}_{2(\ell+|z'|),1-d} \leq \ell + |z'|] \leq 1/2$$

whenever $d < 1/2$, and so the probability that E_{good} and $\mathsf{Last} < i$ hold is at least $1/2 \cdot 2^{-w} = 2^{-(w+1)}$ for all $\ell' < i \leq |y'|$. □

Lemma 5.15. *Fix constant* $d < \frac{2^{-10^4}}{11e}$ *and assume* m *is large enough. Then, we have* $\Pr[\mathsf{Last} \notin [i-v-1, i-v-1+0.1w] | E_{\mathsf{good}}] \leq 2^{-100w}$ *for every* $\ell' < i \leq |y'|$.

Proof. Let E denote the event that $\mathsf{Last} \notin [i-v-1, i-v-1+0.1w]$. Then, by Lemma 5.14 it suffices to show that $\Pr[E, E_{\mathsf{good}}] \leq 2^{-200w}$. As in the proof of Lemma 5.9, the probability that E and E_{good} hold for any given i is upper bounded by the probability that more than $0.1w$ bits are deleted from some substring $y''[b, b+1.1w)$, and the probability that this happens is at most $2m\binom{1.1w}{w}d^{0.1w} \leq 2^{-200w}$ for m large enough by the upper bound on d and a union bound over all choices of b. We explain why this holds. First, note that the bits in a good matching must come from y''. Suppose that at most $0.1w$ bits are deleted from every substring $y''[b, b+1.1w)$. Then, $y''[i-v-w, i-v)$ must be a subsequence of $y''[b, b+1.1w)$ for some $1 \leq b \leq |y''| - 1.1w + 1$. We distinguish two cases:

- $b + 1.1w > |y'|$:

  Recalling that $v = \lceil w/d \rceil$ and the upper bound on d, we have $i - v \leq |y'| - \lceil w/d \rceil \leq |y'| - 1.1w < \min(|y'|+1, b)$, and so Property 5.4 holds for $y''[i-v-w, i-v)$. Therefore, $y''[i-v-w, i-v)$ cannot be a subsequence of $y''[b, b+1.1w)$ for any such b;

- $b + 1.1w \leq |y'|$:

  Since y' is w-subsequence-unique and $y''[i-v-w, i-v)$, $y''[b, b+1.1w)$ are the substrings $y'[i-v-w, i-v)$, $y'[b, b+1.1w)$, respectively, we must have $b \leq i-v-w$ and $b+1.1w \geq i-v$. The same argument from the proof of Lemma 5.9 (noting that $i > \ell' \geq 2v + 1.1w + 1$ for m large enough) shows that we cannot have $\mathsf{Last} \notin [i-v-1, i-v-1+0.1w]$. □

From here onwards fix a constant deletion probability $d < \min\left(\frac{2^{-10^4}}{11e}, \frac{1}{50(1+\beta_{\mathsf{edit}})}\right)$ so that all relevant lemmas hold for the choice of d and assume m is large enough independent of i (but depending on d). Let T be a trace of $1^\ell \| c \| 0^\ell = 1^\ell \| z' \| y''$ and, if there is a good matching of $y''[i-v-w, i-v)$ at $T[u-w, u)$, define $\mathsf{Suff} = T[u:]$. Repeating the reasoning from Section 5.4.1.1 but replacing Lemma 5.9 by Lemma 5.15 yields

$$y''_i = 0 \implies \Pr[\mathsf{Suff}_j = 1 | E_{\mathsf{good}}] \leq \varepsilon_i + \sum_{r=i-v-1}^{i-v-1+0.1w} \Pr[\mathsf{Last} = r | E_{\mathsf{good}}] \sum_{\ell=r+1}^{i-1} P(\ell - r, j) y_\ell$$

$$+\frac{1}{2}\sum_{r=i-v-1}^{i-v-1+0.1w}\Pr[\mathsf{Last}=r|E_{\mathsf{good}}]P(i-r,j) \quad (5.10)$$

and

$$y''_i=1 \implies \Pr[\mathsf{Suff}_j=1|E_{\mathsf{good}}]\geq \varepsilon_i+\sum_{r=i-v-1}^{i-v-1+0.1w}\Pr[\mathsf{Last}=r|E_{\mathsf{good}}]\sum_{\ell=r+1}^{i-1}P(\ell-r,j)y_\ell$$
$$+\sum_{r=i-v-1}^{i-v-1+0.1w}\Pr[\mathsf{Last}=r|E_{\mathsf{good}}]P(i-r,j) \quad (5.11)$$

for every $\ell'<i\leq|y'|$, where $\varepsilon_i\in[0,2^{-100w}]$, and that the gap between the right hand sides of (5.10) and (5.11) is at least $2^{-(12w+2)}$ for every such i.

The same reasoning used to prove Lemmas 5.10 and 5.11 but replacing Lemmas 5.8 and 5.9 by Lemmas 5.14 and 5.15, respectively, shows that, for every $\ell'<i\leq|y'|$, knowledge of z' and $y''_1,\dots,y''_{i-1}$ allows us to compute estimates of $\Pr[\mathsf{Suff}_j=1|E_{\mathsf{good}}]$ and each $\Pr[\mathsf{Last}=r|E_{\mathsf{good}}]$ for $1\leq r<i$ to within additive error 2^{-100w} and 2^{-90w}, respectively, from $2^{1000w+1}$ traces of $1^\ell\|c\|0^\ell=1^\ell\|z'\|y''$ with probability at least $1-2\exp(-m)$ and runtime upper bounded by $\mathrm{poly}(m)$ independent of i, assuming access to $m^2\cdot 2^{1000w+1}=\mathrm{poly}(m)$ i.i.d. samples of Ber_d. As before, this allows us to accurately estimate both sides of (5.10) and (5.11), and thus recover y''_i correctly, with probability at least $1-2m\exp(-m)$. A union bound over all $\ell'<i\leq|y'|$ shows that we correctly recover all the remaining bits of y' using $|y'|\cdot 2^{1000w+1}=\mathrm{poly}(m)$ traces with probability at least $1-\exp(-\Omega(m))$ in overall time $\mathrm{poly}(m)$ when m is large enough. Recalling Lemma 5.13 and that $m=\lceil\log^2 n\rceil$, we conclude that $\mathrm{poly}(\log n)$ traces and time are sufficient to recover c with probability at least $1-n^{-20}$ when n is large enough, which concludes the proof. Moreover, the reasoning from Remark 5.5 applies as is here, showing that the reconstruction algorithm presented can be made deterministic while still using $\mathrm{poly}(m)$ traces and time with reconstruction error $\exp(-\Omega(m))$ when d is constant.

5.5 Mean-based trace reconstruction over general repeat channels

In this section, we generalise the original analysis of mean-based algorithms for worst-case trace reconstruction over the deletion channel by De, O'Donnell, and Servedio [60] and Nazarov and Peres [61], which we discussed in some detail in Section 2.5.3.2, to trace reconstruction over a more general class of repeat channels (of which the deletion channel is a particular example). In [60], the analysis was also generalised in an orthogonal direction to what is called the *general channel*, which was first studied in [128]. This channel combines deletions, geometric insertions of *random* bits, and bit-flips. More precisely, on input a bit x_i, the general channel is a DMSC that behaves as follows:

1. Append a sequence of $N \sim \mathsf{Geom}_{0,p_{\mathsf{ins}}}$ independent uniformly random bits to the output;
2. Delete x_i with probability p_{del};
3. If x_i is not deleted, append x_i to the output with probability $1 - p_{\mathsf{err}}$, or append $1 - x_i$ with probability p_{err}.

It was shown in [60] that, similarly to trace reconstruction over the deletion channel, $\exp(O(n^{1/3}))$ traces are sufficient for mean-based worst-case trace reconstruction over the general channel with constant probabilities $p_{\mathsf{ins}}, p_{\mathsf{del}}, p_{\mathsf{err}}$. Nazarov and Peres [61] showed an analogous result for a closely related channel: First, a geometric number of random bits is inserted before each input bit. Then, deletions and bit-flips are applied to the resulting string. This was later extended to a setting combining deletions, geometric insertions, and random shifts as an intermediate step in the design of average-case trace reconstruction algorithms in [131, 57]. In yet another direction, the results from [60, 61] for the deletion channel were also extended to the deletion channel with position- and symbol-dependent deletion probabilities satisfying monotonicity and periodicity constraints in [134].

In contrast with the above, here we focus on the setting where each trace of the input x is obtained by sending x through a repeat channel with an *arbitrary* finitely supported replication rule R, which is incomparable to trace reconstruction over the "general channel" above. In this section, we analyse the performance of mean-based trace reconstruction algorithms for trace reconstruction over arbitrary repeat channels.

5.5.1 From the deletion channel to general repeat channels

We begin by extending some concepts from [60, 61] already discussed in Section 2.5.3.2 in a straightforward manner to the setting of general repeat channels. Then, we prove a lemma generalising (2.11), which will allow us to obtain upper bounds on the number of traces required for worst-case trace reconstruction over a large class of repeat channels. We use the convention that $0^0 = 1$.

To every $x \in \mathbb{R}^n$ we can associate a polynomial P_x defined as

$$P_x(z) = \sum_{i=1}^{n} x_i z^{i-1}.$$

Letting Y_x denote the output distribution of the repeat channel on input x,[3] we denote by Y_x' the infinite string obtained by appending zeros to Y_x. Then, we define the mean trace μ_x as

$$\mu_x = (\mathbb{E}[(Y_x')_1], \mathbb{E}[(Y_x')_2], \dots)$$

along with the mean trace power series $\overline{P}_{R,x}$ satisfying

$$\overline{P}_{R,x}(z) = \sum_{i=1}^{\infty} (\mu_x)_i z^{i-1}.$$

Our first lemma relates P_x and $\overline{P}_{R,x}$ through a change of variable, generalising (2.11) for the deletion channel.

Lemma 5.16. *Let g denote the probability generating function of a replication rule R with convergence radius $r > 1$. Then, for all $z \neq 1$ in the disc of convergence of g and $x \in \mathbb{R}^n$ we have*

$$\overline{P}_{R,x}(z) = \frac{1 - g(z)}{1 - z} \cdot P_x(g(z)),$$

and for $z = 1$ it holds that

$$\overline{P}_{R,x}(1) = \mathbb{E}[R] \cdot P_x(1).$$

Proof. For $i \in [n]$, define the random variable J_i as $J_i = \varnothing$ if x_i is deleted and $J_i = [a, a + r - 1]$ if x_i ends up as $(Y_x')_a$ and is replicated r times in the output. Then, for every $j \geq 1$ we have

$$(Y_x')_j = \sum_{i=1}^{n} x_i \cdot \mathbf{1}_{\{j \in J_i\}},$$

and consequently

$$\overline{P}_{R,x}(z) = \sum_{j=1}^{\infty} (\mu_x)_j \cdot z^{j-1} = \sum_{j=1}^{\infty} \sum_{i=1}^{n} x_i \cdot \Pr[j \in J_i] \cdot z^{j-1} = \sum_{i=1}^{n} x_i \sum_{j=1}^{\infty} \Pr[j \in J_i] \cdot z^{j-1},$$

[3] We extend the behaviour of the repeat channel to inputs $x \in \mathbb{R}^n$ in the natural manner: The repeat channel independently outputs R_i copies of x_i for each i.

where the last equality is justified by the finiteness of $\sum_{j=1}^{\infty}\Pr[j\in J_i]\cdot z^{j-1}$ for all i and z in the disc of convergence of g, which we show below. For $R_1,R_2,\dots,R_n$ i.i.d. according to R and $R_0=0$, denote $R^{(i)}=\sum_{j=0}^{i}R_j$. Then, we have

$$\begin{aligned}\sum_{j=1}^{\infty}\Pr[j\in J_i]\cdot z^{j-1}&=\sum_{j=1}^{\infty}\Pr[R^{(i-1)}<j\leq R^{(i)}]\cdot z^{j-1}\\&=\sum_{j=1}^{\infty}\Pr[R^{(i-1)}<j,R_i\geq j-R^{(i-1)}]\cdot z^{j-1}\\&=\sum_{j=1}^{\infty}\sum_{j'=0}^{j-1}\Pr[R^{(i-1)}=j']\cdot\Pr[R\geq j-j']\cdot z^{j-1}\\&=\sum_{j'=0}^{\infty}\Pr[R^{(i-1)}=j']\sum_{j=j'+1}^{\infty}\Pr[R\geq j-j']\cdot z^{j-1}\\&=\sum_{j'=0}^{\infty}\Pr[R^{(i-1)}=j']\cdot z^{j'}\cdot\sum_{j=1}^{\infty}\Pr[R\geq j]\cdot z^{j-1},\end{aligned}$$

where the sum rearrangements are allowed due to absolute convergence, which we show below using the fact that z is in the disc of convergence of g, and the last equality follows by replacing j with $j-j'$. Recalling that g is the probability generating function of R, we have

$$\sum_{j'=0}^{\infty}\Pr[R^{(i-1)}=j']\cdot z^{j'}=g(z)^{i-1}$$

for $i\in[n]$. Moreover, if $z\neq 1$ it holds that

$$\begin{aligned}\sum_{j=1}^{\infty}\Pr[R\geq j]\cdot z^{j-1}&=\sum_{j=1}^{\infty}z^{j-1}\sum_{j'=j}^{\infty}\Pr[R=j']\\&=\sum_{j'=1}^{\infty}\Pr[R=j']\sum_{j=1}^{j'}z^{j-1}\\&=\sum_{j'=1}^{\infty}\Pr[R=j']\cdot\frac{1-z^{j'}}{1-z}\\&=\sum_{j'=0}^{\infty}\Pr[R=j']\cdot\frac{1-z^{j'}}{1-z}\\&=\frac{1-g(z)}{1-z},\end{aligned}$$

while we have $\sum_{j=1}^{\infty} \Pr[R \geq j] \cdot z^{j-1} = \mathbb{E}[R]$ if $z = 1$. As a result, we conclude that

$$\sum_{j=1}^{\infty} \Pr[j \in J_i] \cdot z^{j-1} = \frac{(1 - g(z))g(z)^{i-1}}{1 - z}$$

when $z \neq 1$, and thus

$$\overline{P}_{R,x}(z) = \frac{1 - g(z)}{1 - z} \cdot \sum_{i=1}^{n} x_i \cdot g(z)^{i-1} = \frac{1 - g(z)}{1 - z} \cdot P_x(g(z))$$

for all $z \neq 1$ in the disc of convergence of g, while $\overline{P}_{R,x}(1) = \mathbb{E}[R] \cdot P_x(g(1)) = \mathbb{E}[R] \cdot P_x(1)$. □

Remark 5.6. We can obtain (2.11) from Lemma 5.16 by noting that for the deletion channel we have $g(z) = d + (1-d)z$, which implies that $\frac{1-g(z)}{1-z} = \mathbb{E}[R] = 1 - d$, and so $\overline{P}_x(z) = (1-d) \cdot P_x(d + (1-d)z)$.

5.5.2 Mean-based trace reconstruction over finitely-supported repeat channels

In this section, we use the concepts and results from Section 5.5.1 to obtain an upper bound on the number of traces required for worst-case trace reconstruction over *any* repeat channel with finitely supported replication rule. The main difference with respect to [60, 61, 131, 57] is that while they work directly with concrete and simple probability generating functions (Möbius transformations), our analysis must handle an arbitrary probability generating function. Remarkably, this is accomplished via a simple argument. Before we proceed, we generalise the notion of a worst-case trace reconstruction algorithm to general repeat channels.

Definition 5.1. *An algorithm* Rec *is said to be a* (t, R)-worst-case trace reconstruction algorithm *if for n large enough and all $x \in \{0,1\}^n$ it holds that*

$$\Pr[\mathsf{Rec}(Y_x^{(1)}, Y_x^{(2)}, \ldots, Y_x^{(t)}) = x] \geq 1 - 1/n,$$

where the $Y_x^{(i)}$ are i.i.d. according to the output distribution of the repeat channel with replication rule R on input x.

In this section, we will be focusing on repeat channels whose replication rules have finite support. Put differently, we assume there exists a constant $u > 0$ such that $\Pr[R \leq u] = 1$ and $R(u) > 0$. We prove the following general result.

Theorem 5.5. *For every repeat channel with replication rule R having finite support, there exists a (t, R)-worst-case mean-based trace reconstruction algorithm using $t = \exp(O(n^{1/3}))$ traces and time.*

Remark 5.7. Theorem 5.5 cannot be improved, since we know that $\exp(\Omega(n^{1/3}))$ traces are also required for mean-based trace reconstruction over the deletion channel [60, 61]. Moreover, although we do not expand on it here, this theorem can be generalised to all replication rules R with fast decaying tails, such as the Poisson and geometric distributions, by truncating the mean trace up to an appropriate threshold as done in [60, Appendix A.2] for the case of geometric insertions of random bits.

In order to prove Theorem 5.5, we begin by focusing on distinguishing between traces of two arbitrary distinct strings $x, x' \in \{-1, 1\}^n$. Our first goal is to show that there exists an absolute constant $C > 0$ such that

$$\|\mu_x - \mu_{x'}\|_1 \geq \exp(-Cn^{1/3}) \tag{5.12}$$

for all distinct strings $x, x' \in \{-1, 1\}^n$ when n is large enough via Lemma 5.16.

First, observe that the triangle inequality yields

$$|\overline{P}_{R,x}(z) - \overline{P}_{R,x'}(z)| \leq \sum_{i=1}^{un} |(\mu_x)_i - (\mu_{x'})_i| \cdot |z|^{i-1} \leq \|\mu_x - \mu_{x'}\|_1 \cdot \max_{1\leq i\leq un} |z|^{i-1}. \tag{5.13}$$

On the other hand, by Lemma 5.16 we have

$$|\overline{P}_{R,x}(z) - \overline{P}_{R,x'}(z)| = \left|\frac{1-g(z)}{1-z}\right| \cdot |P_{x-x'}(g(z))| \tag{5.14}$$

for $z \neq 1$. With (5.13) and (5.14) in mind, we will prove (5.12) by choosing z appropriately.

Noting that $P_{x-x'} = 2p$ where p is a Littlewood polynomial, Lemma 2.11 ensures the existence of an absolute constant $c_0 > 0$ such that for any $n, L \geq 1$ and distinct strings $x, x' \in \{-1, 1\}^n$ there exists $w_L = e^{i\varphi_L}$ satisfying $|\varphi_L| \leq \frac{\pi}{L}$ and $|P_{x-x'}(w_L)| \geq \exp(-c_0 L)$. By the continuity of $P_{x-x'}$, we may assume that $w_L \neq 1$ by making c_0 slightly larger. Let $z_L \neq 1$ satisfy $g(z_L) = w_L$. We show below that an appropriate solution z_L is guaranteed to exist for all probability generating functions g, provided L is large enough. Setting $z = z_L$ in (5.13) and (5.14), we obtain

$$\|\mu_x - \mu_{x'}\|_1 \cdot \max_{1\leq i\leq un} |z_L|^{i-1} \geq \left|\frac{1-w_L}{1-z_L}\right| \cdot |P_{x-x'}(w_L)| \geq \left|\frac{1-w_L}{1-z_L}\right| \cdot \exp(-c_0 L). \tag{5.15}$$

In order to obtain the desired lower bound on $\|\mu_x - \mu_{x'}\|_1$, we need to control $\left|\frac{1-w_L}{1-z_L}\right|$ and $|z_L|$. We show there is a choice of z_L with good properties.

Lemma 5.17. *There exist absolute constants $c_1, c_2 > 0$ such that for L large enough and any $\varphi \in \left[-\frac{\pi}{L}, \frac{\pi}{L}\right] \setminus \{0\}$ we may choose z_φ satisfying $g(z_\varphi) = e^{i\varphi}$,*

$$0 < |1 - z_\varphi| \leq c_1 |1 - e^{i\varphi}|,$$

and

$$|z_\varphi| \leq 1 + c_2\varphi^2 \leq \exp\left(c_2\pi^2/L^2\right).$$

Instantiating (5.15) with $z_L = z_{\varphi_L}$ guaranteed by Lemma 5.17 with $\varphi = \varphi_L$ implies that there is an absolute constant $L^\star$ such that for all $L \geq L^\star$, n, and distinct $x, x' \in \{-1, 1\}^n$ we have

$$\|\mu_x - \mu_{x'}\|_1 \geq \frac{1}{c_1} \cdot \exp\left(-c_0 L - c_2\pi^2 un/L^2\right).$$

Setting $L = n^{1/3}$ yields (5.12), as desired.

It remains to prove Lemma 5.17. We do so by invoking the inverse function theorem for analytic functions. Below, we denote the open radius-r disc around z by $\mathcal{D}_r(z) = \{z' \in \mathbb{C} : |z' - z| < r\}$.

Lemma 5.18 ([208, Section VIII.4], adapted)**.** *Suppose that a non-constant function $g : \Omega \to \mathbb{C}$ is analytic on a connected open set $\Omega \subseteq \mathbb{C}$ and $g'(z) \neq 0$ for a given $z \in \Omega$. Then, there exist radii $\rho, \delta > 0$ such that for every $w \in \mathcal{D}_\delta(g(z))$ there exists a unique $z_w \in \mathcal{D}_\rho(z)$ satisfying $g(z_w) = w$. Moreover, the inverse function $f : \mathcal{D}_\delta(g(z)) \to \mathcal{D}_\rho(z)$ defined as $f(w) = z_w$ is analytic on $\mathcal{D}_\delta(g(z))$.*

Proof of Lemma 5.17. Note that g is a non-constant analytic function on $\mathbb{C}$ which satisfies $g(1) = 1$ and $g'(1) = \mathbb{E}[R] > 0$. As a result, we can apply Lemma 5.18 to g with $z = 1$. Let $\rho, \delta > 0$ and $f : \mathcal{D}_\delta(1) \to \mathcal{D}_\rho(1)$ be the analytic inverse function guaranteed to exist by Lemma 5.18. Since f is analytic on $\mathcal{D}_\delta(1)$, there is $\gamma < \delta$ such that we may write

$$f(y) = f(1) + f'(1)(y-1) + \sum_{j=2}^{\infty} \frac{f^{(j)}(1)}{j!}(y-1)^j = 1 + f'(1)(y-1) + \sum_{j=2}^{\infty} \frac{f^{(j)}(1)}{j!}(y-1)^j.$$

for all $y \in \mathcal{D}_\gamma(1)$, where the last equality follows from the fact that $f(1) = 1$, since $g(1) = 1$.

Recall that we must choose z_φ such that $g(z_\varphi) = e^{i\varphi} \neq 1$. Suppose L is large enough so that $\frac{\pi}{L} \leq 1$

and $|1-e^{i\varphi}|<\gamma/2$ whenever $|\varphi|\leq\frac{\pi}{L}$. Then, we set

$$z_\varphi=f(e^{i\varphi})=1+f'(1)(e^{i\varphi}-1)+\sum_{j=2}^{\infty}\frac{f^{(j)}(1)}{j!}(e^{i\varphi}-1)^j. \tag{5.16}$$

This implies that $g(z_\varphi)=e^{i\varphi}$ by the definition of f. Moreover, we have $z_\varphi\neq 1$ and

$$0<|z_\varphi-1|\leq|f'(1)|\cdot|e^{i\varphi}-1|+c_3|e^{i\varphi}-1|^2\leq c_1|e^{i\varphi}-1|$$

for all such φ, where $c_1,c_3>0$ are absolute constants, which implies the first statement of the lemma. This follows from (5.16), the triangle inequality, and a standard upper bound on the remainder of the Taylor series: Since $|f(w)|\leq 1+\rho$ for all $w\in\mathcal{D}_\delta(1)$, we have $\left|\frac{f^{(j)}(1)}{j!}\right|\leq\frac{1+\rho}{\gamma^j}$ for all j (e.g., see [208, Section V.4]), and so

$$\sum_{j=2}^{\infty}\left|\frac{f^{(j)}(1)}{j!}\right|\cdot|e^{i\varphi}-1|^j\leq\frac{2(1+\rho)}{\gamma^2}\cdot|e^{i\varphi}-1|^2=c_3|e^{i\varphi}-1|^2 \tag{5.17}$$

for all $w\in\mathcal{D}_{\gamma/2}(1)$ and $c_3=\frac{2(1+\rho)}{\gamma^2}$. Moreover, since both g and f are analytic on a neighbourhood of 1, the chain rule yields

$$f'(1)=\frac{1}{g'(f(1))}=\frac{1}{g'(1)}=\frac{1}{\mathbb{E}[R]}\in\mathbb{R}. \tag{5.18}$$

Therefore, by (5.16), (5.17), (5.18), and the triangle inequality we have

$$|z_\varphi|\leq\left|1+\frac{e^{i\varphi}-1}{\mathbb{E}[R]}\right|+c_3|e^{i\varphi}-1|^2.$$

Then, observing that

$$\left|1+\frac{e^{i\varphi}-1}{\mathbb{E}[R]}\right|=\sqrt{\left(1+\frac{\cos(\varphi)-1}{\mathbb{E}[R]}\right)^2+\left(\frac{\sin(\varphi)}{\mathbb{E}[R]}\right)^2}\leq\sqrt{1+\frac{\varphi^2}{\mathbb{E}[R]^2}}\leq 1+\frac{\varphi^2}{\mathbb{E}[R]^2}$$

and

$$|e^{i\varphi}-1|^2\leq\varphi^2,$$

we conclude that

$$|z_\varphi|\leq 1+c_2\varphi^2\leq\exp(c_2\pi^2/L^2)$$

for some absolute constant $c_2>0$ and L large enough, since $|\varphi|\leq\frac{\pi}{L}$. □

Concluding the proof and time complexity of mean-based trace reconstruction. In order to conclude the proof of Theorem 5.5, we must argue from the above that $\exp(O(n^{1/3}))$ traces and time are enough to reconstruct the unknown input string x with high probability. That all pairs of distinct strings $x, x' \in \{-1,1\}^n$ satisfy (5.12) is already enough to show that $\exp(O(n^{1/3}))$ traces suffice: If $t = \exp(Cn^{1/3})$ for a sufficiently large constant C and we have access to $N = \lceil (un \cdot t)^3 \rceil$ traces $T^{(1)}, \dots, T^{(N)}$, we can estimate $(\mu_x)_j$ by $\widehat{\mu}_j = \frac{1}{N}\sum_{i=1}^N Y_j'^{(i)}$. Then, the Hoeffding bound combined with a union bound over all entries $j = 1, 2, \dots, un$ shows that $\Pr\left[\|\widehat{\mu} - \mu_x\|_1 \geq \frac{1}{4t}\right] \leq e^{-\Omega(n)}$. Coupling this fact with (5.12) shows that $\|\widehat{\mu} - \mu_{x'}\|_1 \geq \frac{3}{4t}$ simultaneously for all $x' \neq x$ with probability at least $1 - e^{-\Omega(n)}$. We can then recover the unknown input x by computing $\widehat{\mu}$ alongside $\mu_{x'}$ for all $x' \in \{-1,1\}^n$, and returning the unique x' such that $\|\widehat{\mu} - \mu_{x'}\|_1 \leq \frac{1}{4t}$, if it exists.

We now show that $\mathrm{poly}(t)$ *time* is also sufficient to recover x, following the discussion in [60, Section 3.2]. First, note that the definitions of the mean trace μ_x and the power series P_x and $\overline{P}_{R,x}$ can be extended by linearity to all $x \in [-1,1]^n$, and that Lemma 5.16 holds for all $x \in [-1,1]^n$. In order to show that $\mathrm{poly}(t)$ traces suffice, it is enough to prove that

$$\|\mu_x\|_1 \geq \frac{1}{t} \tag{5.19}$$

for all $x \in \{0\}^{i-1} \times \{2\} \times [-2,2]^{n-i}$ and $i = 1, 2, \dots, n$. Then, provided we have an estimate $\widehat{\mu}$ satisfying $\|\widehat{\mu} - \mu_x\|_1 < \frac{1}{4t}$ for the true input x (which can be computed with probability $1 - \exp(-\Omega(n))$ using $\mathrm{poly}(t)$ traces and time as described above) and the ability to compute $\mu_{x'}$ for any $x' \in [-1,1]^n$ (which can be done in time $\mathrm{poly}(n)$ to $\mathrm{poly}(n)$ significant digits)[4], a sequence of n linear programs recovers x in time $\mathrm{poly}(n)$. Assuming that we have already recovered $x_1, x_2, \dots, x_{i-1}$, consider the program

$$\begin{aligned}
&\min_{x', \mu_{x'}} \|\mu_{x'} - \widehat{\mu}\|_1 \\
&\text{s.t.} \quad (\mu_{x'})_j = \sum_{\ell=1}^n x'_\ell \cdot \Pr[j \in J_\ell], \quad j = 1, 2, \dots, un, \\
&\qquad\;\; x'_\ell = x_\ell, \quad 1 \leq \ell < i, \\
&\qquad\;\; x'_i = 1, \\
&\qquad\;\; x'_\ell \in [-1,1], \quad i < \ell \leq n,
\end{aligned}$$

[4] Note that computing $\mu_{x'}$ efficiently reduces to the task of computing $\Pr[j \in J_\ell]$ efficiently for every $\ell \leq n$ and $j \leq un$. Since $\Pr[j \in J_\ell] = \sum_{j' < j} \Pr[R^{(\ell-1)} = j'] \cdot \Pr[R \geq j - j']$, the desired statement follows if we can compute $\Pr[R^{(\ell)} = j']$ efficiently for every $\ell \leq n$ and $j' \leq un$. This can be accomplished in time $O(n^3)$ via dynamic programming by exploiting the fact that $\Pr[R^{(\ell)} = j'] = \sum_{j'' \leq j'} \Pr[R^{(\ell-1)} = j''] \cdot \Pr[R = j' - j'']$.

which is a direct generalisation of [59, Expression (3.6)]. By (5.19), the minimum of the program above is smaller than $\frac{1}{4t}$ if $x_i = 1$ and larger than $\frac{3}{4t}$ if $x_i = -1$. Therefore, solving this program allows us to decide whether $x_i = 1$ or $x_i = -1$. Analogously to [59], we can equivalently cast the program above as a linear program in a standard manner by adding new variables $a_1, a_2, \dots, a_{un} \in \mathbb{R}$ along with the constraints $a_j \geq (\mu_{x'})_j - \widehat{\mu}_j$ and $a_j \geq -((\mu_{x'})_j - \widehat{\mu}_j)$ for each $j = 1, 2, \dots, un$, and minimising instead the linear objective function $\sum_{j=1}^{un} a_j$ over $a_1, a_2, \dots, a_{un}$, x', and $\mu_{x'}$. This allows us to recover x_i in time poly(n) (using, e.g., Karmarkar's algorithm [209]), assuming we have previously computed the appropriate estimate $\widehat{\mu}$.

We can prove (5.19) by following the same reasoning as above, but replacing Lemma 2.11 with the following alternative lemma, also due to Borwein and Erdélyi [137].

Lemma 5.19 ([137]). *There is an absolute constant $c > 0$ such that for any polynomial p with constant coefficient 1 and all other coefficients bounded by 1 in modulus and every $L \geq 1$ it holds that*

$$\max_{z = e^{i\varphi} : |\varphi| \leq \frac{\pi}{L}} |p(z)| \geq \exp(-cL).$$

Following the reasoning above we know that (recall (5.15))

$$\|\mu_x\|_1 \geq |z|^{-un} \left| \frac{1 - g(z)}{1 - z} \right| |P_x(g(z))|$$

for every $z \neq 1$ such that $|z| \geq 1$, where $x \in \{0\}^{i-1} \times \{2\} \times [-2, 2]^{n-i}$. Noting that $P_x(w) = 2w^i \cdot p(w)$ for p satisfying the hypotheses of Lemma 5.19, we have

$$\max_{w = e^{i\varphi} : |\varphi| \leq \frac{\pi}{L}} |P_x(w)| = 2 \cdot \max_{w = e^{i\varphi} : |\varphi| \leq \frac{\pi}{L}} |p(w)| \geq \exp(-cL).$$

As before, combining this observation with Lemma 5.17 guarantees that there are absolute constants $c_0, c_1, c_2, L^\star > 0$ such that for any $L \geq L^\star$ and $x \in \{0\}^{i-1} \times \{2\} \times [-2, 2]^{n-i}$ for any n and $i \in [n]$ we can choose $z_L \neq 1$ satisfying

$$\|\mu_x\|_1 \geq |z_L|^{-un} \left| \frac{1 - g(z_L)}{1 - z_L} \right| |P_x(g(z_L))| \geq \frac{1}{c_1} \exp(-c_0 L - c_2 \pi^2 un/L^2) \geq \exp(-C n^{1/3}),$$

where $C > 0$ is an absolute constant. Setting $L = n^{1/3}$ yields the complete statement of Theorem 5.5.

5.5.3 Coded trace reconstruction over general repeat channels

In this section, we briefly discuss some implications of our results on mean-based trace reconstruction over general repeat channels to coded trace reconstruction. Although our coded trace reconstruction model described in Section 5.1 is specified for the deletion channel, it can be generalised in a straightforward way to the case where traces are obtained by sending the codeword through an arbitrary repeat channel (or DMSC).

Consider the coded trace reconstruction problem over a given repeat channel with finitely supported replication rule R. Using Theorem 5.5, we can modify the flexible marker-based code construction from Section 5.2 so that it works for coded trace reconstruction over this repeat channel. By setting the length of the markers to be $C\lceil \log n \rceil$ for some sufficiently large constant $C > 0$ (depending on the "deletion probability" $d = R(0)$, the largest element of $\mathsf{supp}(R)$, and the hidden constant in Theorem 5.5) and by replacing the worst-case trace reconstruction algorithm from Theorem 2.10 with that from Theorem 5.5 for the repeat channel in question, we can follow the reasoning of Section 5.2 exactly to obtain the result below.

Theorem 5.6. *For every non-trivial replication rule R with finite support there exists an efficiently encodable code $\mathcal{C} \subseteq \{0,1\}^{n+r}$ with redundancy $r = O(n/\log n)$ that can be efficiently reconstructed from $\exp(O(\log^{2/3} n))$ traces over the repeat channel with replication rule R.*

5.6 Parallel and follow-up work

In this section, we briefly discuss some work on coded trace reconstruction that appeared either concurrently or subsequently to our work [5].

In a parallel work, Abroshan, Venkataramanan, Dolecek, and Guillén i Fàbregas [210] studied a related setting for coded trace reconstruction, but with key differences to the model considered here. First, they consider a setting where the number of traces is fixed. In contrast, in our setting we allow the number of traces to increase with the blocklength of the code. Moreover, they consider a constant upper bound k on the number of deletions, whose positions are uniformly distributed. In contrast, we consider a constant *rate* of i.i.d. deletions, meaning in particular that the expected number of deletions grows linearly with the blocklength of the code. The authors study a code obtained by concatenating several blocks, with each block being a codeword of a Varshamov-Tenengolts code correcting one worst-case deletion, and the results obtained are incomparable to those presented in this chapter.

Subsequently to the publication of our work, Brakensiek, Li, and Spang [211] built upon our model and techniques, obtaining more results on coded trace reconstruction in several directions. Notably, they present a connection between arbitrary average-case trace reconstruction algorithms and codes in the coded trace reconstruction model, generalising our connection between codes for coded trace reconstruction and the HMPW average-case trace reconstruction algorithm from Section 5.4. More precisely, they show that if there exists an average-case trace reconstruction algorithm using $t(n)$ traces to recover n-bit strings, then, for a large range of $\varepsilon > 0$, there exist binary codes of rate $1 - \varepsilon$ that can be reconstructed from $t\left(O\left(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}\right)\right)$ traces with high probability. Plugging in the state-of-the-art algorithm for average-case trace reconstruction from [57] (as opposed to the HMPW algorithm [59], a modification of which we used in Section 5.4), the result above implies the existence of codes with rate $1 - \varepsilon$ that can be reconstructed from $\exp(O(\log^{1/3}(1/\varepsilon)))$ traces for any constant deletion probability $d < 1$. When $\varepsilon = O(1/\log n)$, which corresponds to the setting considered in this chapter, their result implies the existence of a code with rate $1 - O(1/\log n)$ that requires only $\exp(O(\log^{1/3}(\log n)))$ traces, as opposed to $\text{poly}(\log n) = \exp(O(\log\log n))$ traces as in Section 5.4. However, we remark that the codes obtained in [211] are *not* efficiently encodable when $\varepsilon = o\left(\frac{\log\log n}{\log n}\right)$, and hence when $\varepsilon = O(1/\log n)$, since their code construction requires superpolynomial preprocessing in that case. As a result, the codes we constructed in Section 5.4 remain the best *efficiently encodable* codes for coded trace reconstruction in that regime.

There have also been some subsequent works on coded trace reconstruction from a fixed number of traces with worst-case deletions, insertions, and substitutions. Kiah, Thahn Nguyen, and Yaakobi [212] designed codes that can be decoded with access to a fixed number of traces, each corrupted by a different worst-case deletion, insertion, or substitution. Later, Chrisnata, Kiah, and Yaakobi [213] complemented the previous result by showing that, in the case of a single worst-case deletion, the codes considered have optimal redundancy (up to a constant additive factor).

Chapter 6

Conclusions

6.1 Summary of the thesis

In this thesis, we have made progress on our understanding of the fundamental limits of coding against synchronisation and related errors and on the design of efficient coding schemes for practically motivated models with synchronisation errors.

We began by studying the capacity of the geometric sticky channel, which independently replicates bits according to a $\mathsf{Geom}_{1,p}$ distribution. This is an example of a sticky channel, a special class of practically relevant channels which, due to their structure, are seen as a gateway towards understanding more complex repeat channels. Prior to this thesis, the study of the capacity of sticky channels had focused solely on devising purely numerical methods (based on genie-aided decoding and variants of the Blahut-Arimoto algorithm) to produce sharp capacity bounds [32, 1]. However, this approach does not aid our conceptual understanding of the channel, and, due to genie-aided decoding, cannot provide an exact characterisation of the capacity. As a first step towards such an exact characterisation without computer assistance, we undertook a different approach and applied a general framework from [40] based on convex duality originally exploited to derive *analytical* upper bounds (given by the supremum of an analytic function over $(0, 1)$ which can be easily approximated to the desired accuracy) on the capacity of the deletion and Poisson-repeat channels. This framework reduces the problem of deriving capacity upper bounds for the geometric sticky channel to the problem of analytically designing candidate distributions Y satisfying some relevant constraints. From experience, the tightness with which the constraints are satisfied is directly related to the quality of the resulting analytical upper bound, and, in fact, satisfying an appropriate subset of the constraints *with equality* is a necessary condition for determining the exact capacity of the channel. Candidate distributions designed for the deletion and Poisson-repeat channels in [40] failed to satisfy all but one constraint with equality. For the first time, we designed candidate distributions that satisfy *all* such constraints with equality for some repeat channel. As a result, we obtained sharp analytical capacity upper bounds for the geometric sticky channel which are not only close to the previously known numerical bounds, but actually surpass them for some choices of the parameters. We believe that further study of the explicit distributions we designed will lead to improved structural results for sticky channels.

Subsequently, we studied the geometric deletion channel, a natural extension of the geometric sticky channel which also deletes input bits, again with the goal of moving towards a deeper analysis of the channel without computer assistance. We began by adapting techniques from [40] to derive analytical capacity upper bounds for the geometric deletion channel. Then, we showed how these upper bounds can be significantly improved by combining them with the technique of modifying the mass of the candidate distribution Y at $y = 0$ along with key properties of the geometric deletion channel. Surprisingly, this approach uncovered connections between candidate distributions for the deletion and geometric deletion channels. Using these connections, we were able to give a proof without computer assistance that the capacity of the geometric deletion channel is at most 0.73 bits/channel use in the *large replication regime*, and thus bounded well away from 1. In particular, this result shows that the geometric deletion channel with replication parameter close to 1 (or, equivalently, deletion probability close to 0) behaves radically different than the deletion and Poisson-repeat channels in analogous regimes, which had only been suggested by numerical evidence.

Next, motivated by the study of the capacity of the Poisson-repeat channel, we were naturally led to consider the continuous analogue of (the mean-limited DMC associated with) this channel. This turns out to be the discrete-time Poisson (DTP) channel, a model of optical communication that has received significant attention from the information theory community under different input constraints. However, despite several prior efforts, we still do not know the exact capacity of the DTP channel, and only loose upper bounds were known outside asymptotic regimes. Given this, we were interested in investigating whether techniques originally developed to bound the capacity of the Poisson-repeat channel could be adapted and combined with other techniques to yield improved and easy-to-compute capacity upper bounds for the DTP channel in non-asymptotic regimes. We succeeded in deriving significantly improved upper bounds for the DTP channel. For the case of nonzero dark current, we made key use of the technique of modifying the mass at $y = 0$, which we originally considered for the geometric deletion channel. To complement our upper bounds, we also studied the structure of capacity-achieving distributions for the DTP channel. More precisely, we were interested in showing that, regardless of the peak-power constraint, the support of the capacity-achieving distribution is discrete. This was proved for the case of a finite peak-power constraint in [42], but the case where the peak-power constraint is infinite remained open, and it was only known that the support is an unbounded set [54]. We succeeded in showing that the support is always a discrete set with a finite number of mass points in every bounded interval, regardless of the peak-power constraint. This not only settles the problem completely, but our argument also recovers the result for finite peak-power constraint in an alternative way.

After studying capacity upper bounds for channels with synchronisation and related errors, we turned to the complementary goal of designing *efficient* coding schemes with good parameters that protect against random synchronisation errors. Motivated by prior work on DNA-based data storage systems with nanopore-based sequencing and uncoded trace reconstruction, we introduced and studied the problem of *coded trace reconstruction*. Here, the goal is to design low-redundancy, efficiently encodable coding schemes which can be efficiently reconstructed provided access to multiple, but few, traces of the codeword corrupted by i.i.d. deletions. In particular, one aims for the best possible tradeoff between redundancy and number of traces required for reconstruction, and, as a stepping stone, one wishes to utilise significantly fewer traces than state-of-the-art results on average-case trace reconstruction. As a starting point, we analysed a low-redundancy marker-based construction combined with the best worst-case trace reconstruction algorithms. Although the number of traces required for reconstruction in this case is larger than the best results on average-case trace reconstruction, we showed that the construction above is flexible and allows us to, for example, obtain a quaternary code with the additional property that all codewords have *balanced GC-content*, an important constraint motivated by DNA-based data storage applications, without affecting the redundancy and reconstruction properties. With the goal of significantly reducing the number of traces required for reconstruction, we then showed how to modify the construction above and leverage algorithms for average-case trace reconstruction to obtain an efficiently encodable coding scheme with essentially the same rate as before, but whose efficient reconstruction now requires exponentially fewer traces than the best average-case trace reconstruction algorithms. Finally, motivated by extending coded trace reconstruction to more general types of synchronisation errors beyond i.i.d. deletions (which is significant because errors introduced by nanopore sequencers are more complex than i.i.d. deletions), we investigated whether mean-based worst-case trace reconstruction algorithms [59, 60, 61] can be extended to trace reconstruction over other repeat channels. In particular, we proved that $\exp(O(n^{1/3}))$ traces are also sufficient for worst-case trace reconstruction of n-bit strings over *any* repeat channel induced by a finitely-supported replication distribution, not just the deletion channel. As a bonus, we can also use this result to design, for any such repeat channel, low-redundancy efficiently encodable codes that can be efficiently reconstructed from subpolynomially many traces over that repeat channel.

6.2 Directions for future research

This thesis leaves open several interesting directions for future research. We discuss some of them below.

6.2.1 Capacity of repeat channels

Exact capacity of sticky channels, or sharp elementary upper bounds. In Section 3.1, we showed that it is possible to analytically design candidate distributions with zero KL-gap everywhere for the geometric sticky channel to be used in conjunction with Theorems 2.3 and 2.5. Notably, these distributions satisfy one of the two optimality conditions required to derive the exact capacity of the geometric sticky channel which had not been achieved by previous works on capacity upper bounds for both sticky and non-sticky repeat channels, and lead to sharp analytical upper bounds on the capacity of the geometric sticky channel. However, although we do not have a formal proof of this fact, numerical evidence suggests that the zero KL-gap distributions we designed cannot be realised as output distributions of the DMC associated with the geometric sticky channel (we discuss this in more detail below, along with other consequences). This means that the second optimality condition of Theorem 2.5 is not satisfied, and thus we are not able to determine the exact capacity of this channel.

The state-of-affairs described in the paragraph above leaves open exciting possibilities for improving upon our results from Section 3.1. For example, can we modify the zero KL-gap distributions we designed in order to obtain distributions satisfying both conditions of optimality in Theorem 2.5, thus deriving the exact capacity of the geometric sticky channel? Alternatively, as a more modest goal, can we exploit the structure our capacity upper bounds to derive sharp *elementary* upper bounds on the capacity of the geometric sticky channel? Such bounds would, for example, allow us to learn more about the behavior of the geometric sticky channel in asymptotic regimes where $p \to 0$ or $p \to 1$ (the so-called "large replication regime," which we also discuss below). In an orthogonal direction, it would also be interesting to find an example of a non-trivial sticky channel for which our techniques do yield the exact channel capacity.

Properties of capacity-achieving runlength distributions for sticky channels. Mitzenmacher [32] presented numerical evidence suggesting that capacity-achieving runlength distributions for the geometric sticky channel have sparse support. It would be interesting to rigorously prove a result of this type, in line with the structural results we obtained in Chapter 4 for the DTP channel. For example: *Can we show that capacity-achieving runlength distributions for the geometric sticky channel do not have full support over* $\mathbb{N}$?

We briefly discuss a viable approach to this problem exploiting the distributions we designed in Chapter 3. For every $p < 1/2$ and output mean constraint $\mu \geq \frac{1}{1-p}$, the distribution $Y^{(q)}$ given in (3.4) with $\mathbb{E}[Y^{(q)}] = \mu$ is the unique valid distribution satisfying the mean constraint with KL-gap $\Delta(x) = 0$ for all $x \in \mathbb{N}$. Moreover, if $Y^{(q)}$ is realisable as an output runlength distribution of the geometric sticky channel, then the corresponding input runlength distribution $X^{(q)}$ is uniquely given by

$$X^{(q)}(x) = \sum_{i=0}^{x-1} \frac{(-p)^i \binom{x-1}{i}}{(1-p)^x} Y^{(q)}(x-i), \quad x \in \mathbb{N}.$$

Combining these observations with the optimality conditions from Theorem 2.5 implies that one of two possibilities must hold:

1. Either $X^{(q)}$ is a valid probability distribution and is optimal among all input runlength distributions satisfying the output mean constraint, or;
2. No runlength distribution satisfying the output mean constraint with full support over $\mathbb{N}$ is optimal.

This suggests the following approach towards showing the inexistence of a capacity-achieving runlength distribution with full support for the geometric sticky channel: Prove that for every $q \in (0,1)$ such that $\mathbb{E}[Y^{(q)}] \geq \frac{1}{1-p}$ there is x such that $X^{(q)}(x) < 0$, and so $X^{(q)}$ is not a probability distribution. We currently do not have a proof of this fact, but we can derive partial results from Chapter 3. Figure 6.1 shows that $X^{(q)}(22) < 0$ for all $q \leq 0.6$ when $p = 1/3$, or, equivalently, that for $p = 1/3$

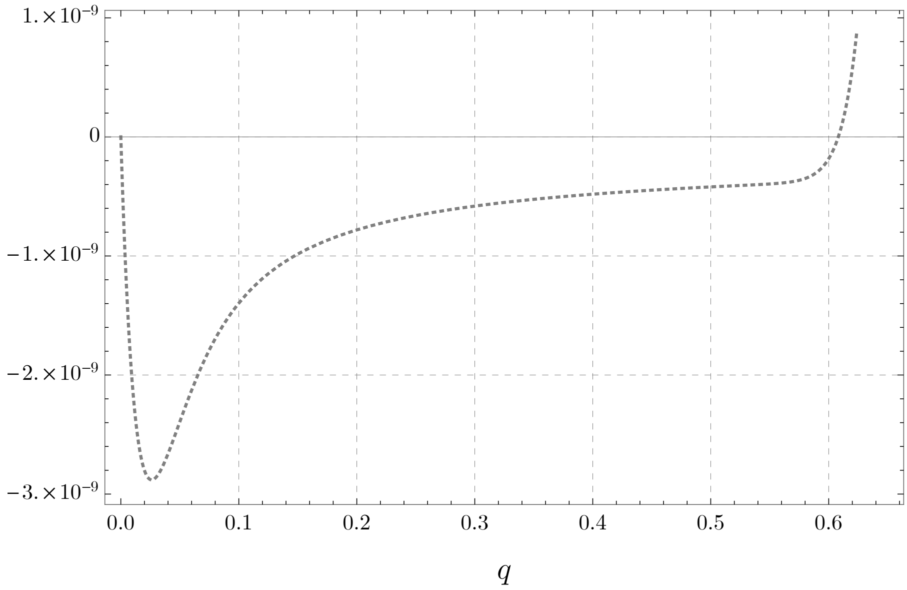

Figure 6.1: Plot of $X^{(q)}(22)$ for $p = 1/3$.

there are no full-support capacity-achieving runlength distributions under any output mean constraint $\mu \leq \mathbb{E}[Y^{(0.6)}] \approx 1.92$.

Zero KL-gap distributions for all sticky channels. As previously discussed in Section 3.1, we were able to design distributions with zero KL-gap everywhere for the DMC associated with the geometric sticky channel. It would be interesting to investigate whether this is an example of a more general phenomenon. Namely, can we show that it is possible to design distributions with zero KL-gap everywhere for every sticky channel? Although this does not guarantee an exact characterisation of the capacity, we expect such a general method would lead to sharp analytical capacity upper bounds for many sticky channels and eventually a general computer-unaided analysis of their capacity. Following the reasoning of Section 3.1, this question can be answered affirmatively by showing that for every DMC with transition rule $Y_x = \sum_{i=1}^{x} R_i$ associated with a sticky channel with replication distribution R (where the R_i are i.i.d. according to R) there exists a function f with appropriate growth satisfying

$$\mathbb{E}[f(Y_x)] = H(Y_x), \quad \forall x \in \mathbb{N}.$$

A better understanding of the KL-gaps of candidate distributions for the geometric deletion channel. In Section 3.2.3, we saw how to modify candidate distributions for the geometric deletion channel in order to obtain significantly improved capacity upper bounds. However, in order to undertake a deeper mathematical treatment of these bounds, it is imperative to have a good understanding of the KL-gap of the original candidate distributions. More precisely, can we derive more amenable expressions or sharp bounds for the quantities $\overline{\Delta}_{\overline{\delta}}(x)$ and $\Delta_{\delta}(x)$ defined in (3.67) and (3.71), respectively, similarly to what was done for the DTP channel in Section 4.3? Or, as an intermediate goal, can we derive sharp explicit bounds on the entropy of the negative binomial distribution, as was done for the Poisson distribution in [214]?

The geometric sticky and deletion channels in the large replication regime. In Section 3.2.5, we showed how previous techniques we developed for the geometric deletion channel lead us to a proof without computer assistance that the capacity of this channel is at most 0.73 bits/channel use when the replication parameter p is close to 1, or, equivalently, when the deletion probability $d = 1 - p$ is close to 0. We believe that the large replication regime is interesting and merits further study, which could lead to the development of new techniques for analysing other repeat channels. The first natural step would be to improve our upper bound; numerical evidence presented in Section 3.2 suggests that the true limit of the capacity when $p \to 1$ is at most 0.3 bits/channel use. The second step would be not only to determine this limit, but to determine also higher-order terms of the asymptotic expansion of the capacity in the large replication regime, making a parallel with results of Kalai, Mitzenmacher, and Sudan [78] and Kanoria and Montanari [79] for the deletion channel with small deletion probability. However, the large replication regime for the geometric sticky and deletion channels appears to be more difficult to tackle. Indeed, the first-order term of the asymptotic expansion of $C(d)$, the capacity of the deletion channel, when $d \to 0$ can already be derived by noting that a uniform input distribution is nearly optimal in this regime (and higher-order terms can be determined by considering perturbations of the uniform distribution). On the other hand, the structure of the optimal input distributions for the geometric sticky and deletion channels in the large replication regime is not clear.

Capacity bounds for multi-trace repeat channels. Motivated by research on trace reconstruction, it would be interesting to extend the capacity upper bound techniques presented in this thesis to a setting where the input is sent through a fixed number $t > 1$ of independent repeat channels (i.e., the t-trace version of a repeat channel). If one knows an upper bound U on the capacity of some repeat channel, then tU is a trivial upper bound on the capacity of the t-trace version of this repeat channel. Therefore, the goal would be to derive capacity upper bounds beating this trivial bound. As discussed in Chapter 2, results of this type are only known for the deletion channel, and even then they only hold in the regime where the deletion probability approaches 0 [124].

It is possible to generalise the discussion of Section 2.5.1.1, and in particular Theorems 2.3 and 2.5, to the t-trace setting. We omit details, and briefly discuss the scenario for t-trace sticky channels. In order to derive capacity upper bounds in this case, it suffices to consider the DMC which on input $x \in \mathbb{N}$ outputs

$$Y_{x,t} = (Y_x^{(1)}, Y_x^{(2)}, \dots, Y_x^{(t)}),$$

where the $Y_x^{(i)}$ are i.i.d. according to Y_x, the output distribution of the DMC associated to the original (1-trace) sticky channel. The output mean constraint in the t-trace setting then corresponds to only allowing input distributions X such that the corresponding output distribution $Y_t = (Y^{(1)}, Y^{(2)}, \dots, Y^{(t)})$ satisfies $\mathbb{E}[Y^{(1)}] = \mu$.[1] Analogously to Theorem 2.5, upper bounding the capacity of the mean-limited DMC boils down to designing a candidate distribution Y over $\mathcal{Y}^t$ such that

$$D_{\mathsf{KL}}(Y_{x,t} \| Y) \leq a\mathbb{E}[Y_x] + b$$

for all $x \in \mathbb{N}$. In turn, this inequality leads to the capacity upper bound $a\mu + b$ for the μ-limited DMC. The first thing one may consider is to set $Y = (\widehat{Y}^{(1)}, \widehat{Y}^{(2)}, \dots, \widehat{Y}^{(t)})$, where the $\widehat{Y}^{(i)}$ are i.i.d. according to a distribution $\widehat{Y}$ over $\mathcal{Y}$ that yields a good capacity upper bound for the 1-trace version of the mean-limited DMC. However, this only leads to a trivial capacity upper bound, because in this case we have

$$D_{\mathsf{KL}}(Y_{x,t} \| Y) = t \cdot D_{\mathsf{KL}}(Y_x \| \widehat{Y})$$

for all $x \in \mathbb{N}$. Therefore, designing $Y = (Y^{(1)}, Y^{(2)}, \dots, Y^{(t)})$ in order to obtain non-trivial analytical capacity upper bounds on this channel requires that the $Y^{(i)}$ be carefully correlated with each other. Given this, it would already be interesting to consider the case $t = 2$ and derive non-trivial analytical capacity upper bounds in this setting. One possible approach would be to start with $Y = (\widehat{Y}^{(1)}, \widehat{Y}^{(2)})$ as above, where $\widehat{Y}^{(1)}$ and $\widehat{Y}^{(2)}$ are i.i.d. according to a distribution $\widehat{Y}$ that led to a good upper bound in the case $t = 1$, and then modify the value of $Y(0,0)$ and renormalise Y. The new distribution would now have two correlated coordinates (a prerequisite for obtaining non-trivial upper bounds), and an appropriate choice of $Y(0,0)$ could lead to non-trivial upper bounds.

[1] In this case we actually have $\mathbb{E}[Y^{(i)}] = \mu$ for all $i \in [t]$.

6.2.2 Capacity of the discrete-time Poisson channel

Improved capacity upper bounds under a finite peak-power constraint. In Chapter 4, we saw how the digamma distribution from [40] combined with techniques initially considered in Section 3.2.3 for the geometric deletion channel can be used to derive improved non-asymptotic upper bounds on the capacity $C(\lambda, \mu)$ of the DTP channel with dark current λ and average-power constraint μ. Notably, these new bounds actually improve on known bounds on the capacity $C(\lambda, \mu, A)$ with a finite peak-power constraint A for certain choices of λ and μ. It would be interesting to make use of the finite peak-power constraint to derive even better upper bounds on $C(\lambda, \mu, A)$. Modifying the mass of the digamma distribution at $y = 0$, as was done in Section 4.3, does not seem to yield useful results in this setting. Instead, we believe that modifying the *tail* of the digamma distribution appropriately as a function of the peak-power constraint A would be a viable approach towards achieving the goal above.

Sparsity of the capacity-achieving distribution. In Section 4.4, we showed that the capacity-achieving distribution for the DTP channel with dark current λ, average-power constraint μ, and peak-power constraint A (where we may have $A = \infty$) has discrete support, with a finite number of mass points in every bounded interval. However, our intuition suggests that an even stronger structural result about the capacity-achieving distribution should hold. Since a Poisson distribution Poi_λ has variance λ, we conjecture that mass points $x < x'$ of the discrete capacity-achieving distribution should satisfy $|x - x'| = \Omega(\sqrt{x})$, provided x is large enough. We believe that a more careful analysis of the optimality conditions from Theorem 4.1 could be used to prove this conjecture. This result would belong to a new type of *refined* structural results about capacity-achieving distributions of stationary memoryless channels. As discussed in Section 4.1.2, previous works have mostly focused only on showing the finiteness or discreteness of the support of the capacity-achieving distribution, with some works also deriving conditions for the optimality of certain binary input distributions (which are never optimal for the DTP channel without a peak-power constraint), or that specific input symbols must be in the support.

6.2.3 Coded and uncoded trace reconstruction

Further derandomising average-case trace reconstruction. In Section 5.4, we showed that an ε-almost k-wise independent string, for appropriate choices of ε and k, satisfies the assumptions required by the HMPW average-case trace reconstruction. Coupling the fact that such strings can be generated from a logarithmic number of uniformly random bits with a modified version of the HMPW algorithm allowed us to obtain an efficiently encodable length n binary code with redundancy $O(\log n)$ that can be efficiently reconstructed from $\mathrm{poly}(n)$ traces, effectively derandomising the HMPW algorithm. Combining this code with our marker-based construction then led to an efficiently encodable code with redundancy $O(n/\log n)$ that can be efficiently reconstructed from $\mathrm{poly}(\log n)$ traces. Alternatively, we can combine the code above with the subsequent construction of Brakensiek, Li, and Spang [211] to obtain the same result.

The situation in the previous paragraph leads us to wonder whether we can "derandomise" improved average-case trace reconstruction algorithms, in particular those due to Peres and Zhai [131] and Holden, Pemantle, and Peres [57]. In other words, starting with such an algorithm, can we design an efficiently encodable length n binary code with redundancy $O(\log n)$ that can be efficiently reconstructed from $\exp(O(\log^{1/3} n))$ traces? Besides being a natural coding-theoretic question by itself, given that we achieved this with respect to the HMPW algorithm, combining such a code with the construction from [211] would lead to *efficiently encodable* codes with rate $1-\varepsilon$ that can be efficiently reconstructed from $\exp(O(\log^{1/3}(1/\varepsilon)))$ traces for ε much smaller than $\frac{\log\log n}{\log n}$. This is significant because the codes from [211] are not efficiently encodable when $\varepsilon = o\left(\frac{\log\log n}{\log n}\right)$.

Trace reconstruction of k-wise independent strings. We can take another perspective on our results from Section 5.4 by considering trace reconstruction of k-wise independent n-bit strings, which interpolates between average-case ($k = n$) and worst-case ($k = 0$) trace reconstruction. In the two extremes, we have sublinear upper bounds for average-case reconstruction, while we only have exponential upper bounds in the worst-case setting. We have complemented this by showing that ε-almost k-wise independent strings, with $k = O(\log n)$, can be reconstructed from $\mathrm{poly}(n)$ traces. Two natural questions remain: First, what is the smallest $k^\star$ for which trace reconstruction of $k^\star$-wise independent strings is possible with $\mathrm{poly}(n)$ traces? We know that $k^\star = O(\log n)$, but even $k^\star = 0$ is possible. Second, what is the smallest $k^\star$ for which a *sublinear* number of traces suffices? Here, we know that $k^\star > 0$ via known lower bounds for worst-case trace reconstruction [140].

Coded trace reconstruction beyond repeat channels. In Chapter 5, we focused mostly on coded trace reconstruction from i.i.d. deletions, and also briefly considered this problem over more general repeat channels. As previously discussed, this setting is motivated by nanopore-based sequencing in portable DNA-based data storage, and is a simplified model of the actual biological process. In fact, the real scenario introduces other errors not considered in Chapter 5, such as insertions of random symbols and substitutions. Moreover, the errors are not necessarily independent of each other.

Given the above, it would be interesting to extend our work on coded trace reconstruction to handle different or more general types of (not necessarily i.i.d.) synchronisation errors. In particular, it would be interesting to design new codes that handle i.i.d. insertions of random bits coupled with i.i.d. deletions, as it is not clear how to extend our marker-based constructions to this setting. Finally, we note that there has been no work on trace reconstruction under non-i.i.d. random errors. It would be interesting to define and study such non-i.i.d. settings capturing key properties of the nanopore-based sequencing process.

Narrowing the gap between bounds for coded and uncoded trace reconstruction. As discussed in Chapter 2, even disregarding efficient encoding and reconstruction, the gaps between known upper and lower bounds for both coded and uncoded (worst-case and average-case) trace reconstruction of binary strings are still exponentially large. Of these, we believe that the lower bounds are generally loose, and can be improved significantly, especially in the case of coded and worst-case trace reconstruction.

Bibliography

[1] H. Mercier, V. Tarokh, and F. Labeau, "Bounds on the capacity of discrete memoryless channels corrupted by synchronization and substitution errors," *IEEE Transactions on Information Theory*, vol. 58, no. 7, pp. 4306–4330, 2012.

[2] M. Cheraghchi and J. Ribeiro, "Sharp analytical capacity upper bounds for sticky and related channels," *IEEE Transactions on Information Theory*, vol. 65, no. 11, pp. 6950–6974, Nov 2019.

[3] M. Rahmati and T. M. Duman, "Upper bounds on the capacity of deletion channels using channel fragmentation," *IEEE Transactions on Information Theory*, vol. 61, no. 1, pp. 146–156, 2015.

[4] M. Cheraghchi and J. Ribeiro, "Improved upper bounds and structural results on the capacity of the discrete-time Poisson channel," *IEEE Transactions on Information Theory*, vol. 65, no. 7, pp. 4052–4068, July 2019.

[5] M. Cheraghchi, R. Gabrys, O. Milenkovic, and J. Ribeiro, "Coded trace reconstruction," *IEEE Transactions on Information Theory*, vol. 66, no. 10, pp. 6084–6103, 2020.

[6] M. Cheraghchi and J. Ribeiro, "An overview of capacity results for synchronization channels," *IEEE Transactions on Information Theory*, 2020, to appear. DOI: 10.1109/TIT.2020.2997329.

[7] A. V. Kuznetsov and A. J. H. Vinck, "A coding scheme for single peak-shift correction in (d, k)-constrained channels," *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1444–1450, 1993.

[8] P. A. H. Bours, "Construction of fixed-length insertion/deletion correcting runlength-limited codes," *IEEE Transactions on Information Theory*, vol. 40, no. 6, pp. 1841–1856, 1994.

[9] R. M. Roth and P. H. Siegel, “Lee-metric BCH codes and their application to constrained and partial-response channels,” *IEEE Transactions on Information Theory*, vol. 40, no. 4, pp. 1083–1096, 1994.

[10] Y. Ng, B. V. K. Vijaya Kumar, K. Cai, S. Nabavi, and T. C. Chong, “Picket-shift codes for bit-patterned media recording with insertion/deletion errors,” *IEEE Transactions on Magnetics*, vol. 46, no. 6, pp. 2268–2271, 2010.

[11] A. R. Krishnan and B. Vasic, “Coding for correcting insertions and deletions in bit-patterned media recording,” in *2011 IEEE Global Telecommunications Conference (GLOBECOM)*, 2011, pp. 1–5.

[12] Y. M. Chee, H. M. Kiah, A. Vardy, V. K. Vu, and E. Yaakobi, “Coding for racetrack memories,” *IEEE Transactions on Information Theory*, vol. 64, no. 11, pp. 7094–7112, Nov 2018.

[13] Y. M. Chee, R. Gabrys, A. Vardy, V. K. Vu, and E. Yaakobi, “Reconstruction from deletions in racetrack memories,” in *2018 IEEE Information Theory Workshop (ITW)*, Nov 2018, pp. 1–5.

[14] J. Sima and J. Bruck, “Correcting deletions in multiple-heads racetrack memories,” in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 1367–1371.

[15] S. M. H. T. Yazdi, R. Gabrys, and O. Milenkovic, “Portable and error-free DNA-based data storage,” *Scientific reports*, vol. 7, no. 1, p. 5011, 2017.

[16] L. Organick, S. D. Ang, Y.-J. Chen, R. Lopez, S. Yekhanin, K. Makarychev, M. Z. Racz, G. Kamath, P. Gopalan, B. Nguyen *et al.*, “Random access in large-scale DNA data storage,” *Nature biotechnology*, vol. 36, no. 3, p. 242, 2018.

[17] T. Batu, S. Kannan, S. Khanna, and A. McGregor, “Reconstructing strings from random traces,” in *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2004, pp. 910–918.

[18] A. Orlitsky, “Interactive communication: balanced distributions, correlated files, and average-case complexity,” in *1991 IEEE 32nd Annual Symposium on Foundations of Computer Science (FOCS)*, 1991, pp. 228–238.

[19] K. Cheng, Z. Jin, X. Li, and K. Wu, “Deterministic document exchange protocols, and almost optimal binary codes for edit errors,” in *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, Oct 2018, pp. 200–211.

[20] B. Haeupler, “Optimal document exchange and new codes for insertions and deletions,” in *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, 2019, pp. 334–347.

[21] M. Kiwi, M. Loebl, and J. Matoušek, “Expected length of the longest common subsequence for large alphabets,” *Advances in Mathematics*, vol. 197, no. 2, pp. 480–498, 2005.

[22] H. Mercier, M. Khabbazian, and V. K. Bhargava, “On the number of subsequences when deleting symbols from a string,” *IEEE Transactions on Information Theory*, vol. 54, no. 7, pp. 3279–3285, July 2008.

[23] Y. Liron and M. Langberg, “A characterization of the number of subsequences obtained via the deletion channel,” *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2300–2312, May 2015.

[24] C. E. Shannon, “A mathematical theory of communication,” *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[25] ——, “A mathematical theory of communication,” *The Bell System Technical Journal*, vol. 27, no. 4, pp. 623–656, 1948.

[26] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. John Wiley & Sons, 2006.

[27] M. Mitzenmacher, “A survey of results for deletion channels and related synchronization channels,” *Probability Surveys*, vol. 6, pp. 1–33, 2009.

[28] M. Dalai, “A new bound on the capacity of the binary deletion channel with high deletion probabilities,” in *2011 IEEE International Symposium on Information Theory (ISIT)*, 2011, pp. 499–502.

[29] E. Drinea and M. Mitzenmacher, “Improved lower bounds for the capacity of iid deletion and duplication channels,” *IEEE Transactions on Information Theory*, vol. 53, no. 8, pp. 2693–2714, 2007.

[30] E. Drinea and A. Kirsch, “Directly lower bounding the information capacity for channels with i.i.d. deletions and duplications,” in *2007 IEEE International Symposium on Information Theory (ISIT)*, 2007, pp. 1731–1735.

[31] A. Kirsch and E. Drinea, "Directly lower bounding the information capacity for channels with i.i.d. deletions and duplications," *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 86–102, Jan 2010.

[32] M. Mitzenmacher, "Capacity bounds for sticky channels," *IEEE Transactions on Information Theory*, vol. 54, no. 1, pp. 72–77, 2008.

[33] A. Magner, J. Duda, W. Szpankowski, and A. Grama, "Fundamental bounds for sequence reconstruction from nanopore sequencers," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 2, no. 1, pp. 92–106, June 2016.

[34] F. Farnoud, O. Milenkovic, and N. P. Santhanam, "Small-sample distribution estimation over sticky channels," in *2009 IEEE International Symposium on Information Theory (ISIT)*, 2009, pp. 1125–1129.

[35] L. Dolecek and V. Anantharam, "On subsets of binary strings immune to multiple repetition errors," in *2007 IEEE International Symposium on Information Theory (ISIT)*, 2007, pp. 1691–1695.

[36] ——, "Prefixing method for correcting repetition errors," in *2008 IEEE International Symposium on Information Theory (ISIT)*, 2008, pp. 965–969.

[37] H. Mercier, V. K. Bhargava, and V. Tarokh, "A survey of error-correcting codes for channels with symbol synchronization errors," *IEEE Communications Surveys Tutorials*, vol. 12, no. 1, pp. 87–96, First Quarter 2010.

[38] S. Verdú, "On channel capacity per unit cost," *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 1019–1030, 1990.

[39] A. R. Iyengar, P. H. Siegel, and J. K. Wolf, "On the capacity of channels with timing synchronization errors," *IEEE Transactions on Information Theory*, vol. 62, no. 2, pp. 793–810, 2016.

[40] M. Cheraghchi, "Capacity upper bounds for deletion-type channels," *J. ACM*, vol. 66, no. 2, pp. 9:1–9:79, Mar. 2019.

[41] M. Mitzenmacher and E. Drinea, "A simple lower bound for the capacity of the deletion channel," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4657–4660, 2006.

[42] S. Shamai (Shitz), "Capacity of a pulse amplitude modulated direct detection photon channel," *IEE Proceedings I (Communications, Speech and Vision)*, vol. 137, no. 6, pp. 424–430, 1990.

[43] Y. M. Kabanov, "The capacity of a channel of the Poisson type," *Theory of Probability & Its Applications*, vol. 23, no. 1, pp. 143–147, 1978.

[44] M. Davis, "Capacity and cutoff rate for Poisson-type channels," *IEEE Transactions on Information Theory*, vol. 26, no. 6, pp. 710–715, 1980.

[45] A. D. Wyner, "Capacity and error exponent for the direct detection photon channel. I," *IEEE Transactions on Information Theory*, vol. 34, no. 6, pp. 1449–1461, 1988.

[46] ——, "Capacity and error exponent for the direct detection photon channel. II," *IEEE Transactions on Information Theory*, vol. 34, no. 6, pp. 1462–1471, 1988.

[47] A. Martinez, "Spectral efficiency of optical direct detection," *JOSA B*, vol. 24, no. 4, pp. 739–749, 2007.

[48] A. Lapidoth, J. H. Shapiro, V. Venkatesan, and L. Wang, "The discrete-time Poisson channel at low input powers," *IEEE Transactions on Information Theory*, vol. 57, no. 6, pp. 3260–3272, 2011.

[49] L. Wang and G. W. Wornell, "A refined analysis of the Poisson channel in the high-photon-efficiency regime," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4299–4311, 2014.

[50] D. Brady and S. Verdú, "The asymptotic capacity of the direct detection photon channel with a bandwidth constraint," in *28th Allerton Conference on Communication, Control and Computing*, 1990, pp. 691–700.

[51] A. Lapidoth and S. M. Moser, "On the capacity of the discrete-time Poisson channel," *IEEE Transactions on Information Theory*, vol. 55, no. 1, pp. 303–322, 2009.

[52] J. G. Smith, "The information capacity of amplitude- and variance-constrained scalar Gaussian channels," *Information and Control*, vol. 18, no. 3, pp. 203–219, 1971.

[53] J. Fahs and I. Abou-Faycal, "On properties of the support of capacity-achieving distributions for additive noise channel models with input cost constraints," *IEEE Transactions on Information Theory*, vol. 64, no. 2, pp. 1178–1198, Feb 2018.

[54] J. Cao, S. Hranilovic, and J. Chen, "Capacity-achieving distributions for the discrete-time Poisson channel - Part I: General properties and numerical techniques," *IEEE Transactions on Communications*, vol. 62, no. 1, pp. 194–202, 2014.

[55] M. Cheraghchi and J. Ribeiro, "Non-asymptotic capacity upper bounds for the discrete-time Poisson channel with positive dark current," *arXiv e-prints*, p. arXiv:2010.14858, Oct 2020.

[56] S. M. H. T. Yazdi, H. M. Kiah, E. Garcia-Ruiz, J. Ma, H. Zhao, and O. Milenkovic, "DNA-based storage: Trends and methods," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 1, no. 3, pp. 230–248, 2015.

[57] N. Holden, R. Pemantle, and Y. Peres, "Subpolynomial trace reconstruction for random strings and arbitrary deletion probability," in *Proceedings of the 31st Conference On Learning Theory (COLT)*, 2018, pp. 1799–1840.

[58] S. M. H. Tabatabaei Yazdi, H. M. Kiah, R. Gabrys, and O. Milenkovic, "Mutually uncorrelated primers for DNA-based data storage," *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6283–6296, Sep. 2018.

[59] T. Holenstein, M. Mitzenmacher, R. Panigrahy, and U. Wieder, "Trace reconstruction with constant deletion probability and related results," in *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008, pp. 389–398.

[60] A. De, R. O'Donnell, and R. A. Servedio, "Optimal mean-based algorithms for trace reconstruction," *Annals of Applied Probability*, vol. 29, no. 2, pp. 851–874, Apr 2019.

[61] F. Nazarov and Y. Peres, "Trace reconstruction with $\exp(O(n^{1/3}))$ samples," in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, 2017, pp. 1042–1046.

[62] N. L. Johnson, A. W. Kemp, and S. Kotz, *Univariate Discrete Distributions*, 2nd ed., ser. Wiley Series in Probability and Statistics. John Wiley & Sons, 1992.

[63] M. D. Perlman, "Jensen's inequality for a convex vector-valued function on an infinite-dimensional space," *Journal of Multivariate Analysis*, vol. 4, no. 1, pp. 52–65, 1974.

[64] R. Vershynin, *High-Dimensional Probability: An Introduction with Applications in Data Science*, ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[65] D. G. Brown, "How I wasted too long finding a concentration inequality for sums of geometric variables," 2011, available at https://cs.uwaterloo.ca/~browndg/negbin.pdf, last accessed February 7, 2021.

[66] C. Canonne, "A short note on Poisson tail bounds," 2019, available at www.cs.columbia.edu/~ccanonne/files/misc/2017-poissonconcentration.pdf, last accessed February 7, 2021.

[67] N. Alon, O. Goldreich, J. Håstad, and R. Peralta, "Simple constructions of almost k-wise independent random variables," *Random Structures & Algorithms*, vol. 3, no. 3, pp. 289–304, 1992.

[68] O. Goldreich, "Three XOR-lemmas — An exposition," in *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, O. Goldreich, Ed. Springer Berlin Heidelberg, 2011, pp. 248–272.

[69] V. Shoup, "New algorithms for finding irreducible polynomials over finite fields," *Mathematics of Computation*, vol. 54, no. 189, pp. 435–447, 1990.

[70] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*, 10th ed., ser. Applied Mathematics Series. National Bureau of Standards, 1972, vol. 55.

[71] H. Alzer, "On some inequalities for the incomplete gamma function," *Mathematics of Computation*, vol. 66, no. 218, pp. 771–778, 1997.

[72] R. L. Dobrushin, "Shannon's theorems for channels with synchronization errors," *Problemy Peredachi Informatsii*, vol. 3, no. 4, pp. 18–36, 1967.

[73] S. Z. Stambler, "Memoryless channels with synchronization errors: The general case," *Problemy Peredachi Informatsii*, vol. 6, no. 3, pp. 43–49, 1970.

[74] R. Ahlswede and J. Wolfowitz, "Channels without synchronization," *Advances in Applied Probability*, vol. 3, no. 2, pp. 383–403, 1971.

[75] O. K. Kozlov, "A strong converse of Shannon's theorem for memoryless channels with synchronization errors," *Problemy Peredachi Informatsii*, vol. 7, no. 1, pp. 102–105, 1971.

[76] W. Zeng, P. Mitran, and A. Kavčić, "On the information stability of channels with timing errors," in *2006 IEEE International Symposium on Information Theory (ISIT)*, July 2006, pp. 1885–1889.

[77] Y. Li and V. Y. F. Tan, "On the capacity of deletion channels with states," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 2114–2119.

[78] A. Kalai, M. Mitzenmacher, and M. Sudan, "Tight asymptotic bounds for the deletion channel with small deletion probabilities," in *2010 IEEE International Symposium on Information Theory (ISIT)*, 2010, pp. 997–1001.

[79] Y. Kanoria and A. Montanari, "Optimal coding for the binary deletion channel with small deletion probability," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6192–6219, 2013.

[80] R. G. Gallager, "Sequential decoding for binary channels with noise and synchronization errors," MIT Lexington Lincoln Laboratory, Tech. Rep., 1961.

[81] K. Zigangirov, "Sequential decoding for a binary channel with drop-outs and insertions," *Problemy Peredachi Informatsii*, vol. 5, no. 2, pp. 23–30, 1969.

[82] S. Diggavi and M. Grossglauser, "On information transmission over a finite buffer channel," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1226–1237, March 2006.

[83] 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.

[84] E. Drinea and M. Mitzenmacher, "On lower bounds for the capacity of deletion channels," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4648–4657, Oct 2006.

[85] D. Fertonani and T. M. Duman, "Novel bounds on the capacity of the binary deletion channel," *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2753–2765, 2010.

[86] R. Venkataramanan, S. Tatikonda, and K. Ramchandran, "Achievable rates for channels with deletions and insertions," *IEEE Transactions on Information Theory*, vol. 59, no. 11, pp. 6990–7013, 2013.

[87] J. Castiglione and A. Kavčić, "Trellis based lower bounds on capacities of channels with synchronization errors," in *2015 IEEE Information Theory Workshop - Fall (ITW)*, Oct 2015, pp. 24–28.

[88] M. Ramezani and M. Ardakani, "On the capacity of duplication channels," *IEEE Transactions on Communications*, vol. 61, no. 3, pp. 1020–1027, 2013.

[89] S. Diggavi, M. Mitzenmacher, and H. D. Pfister, "Capacity upper bounds for the deletion channel," in *2007 IEEE International Symposium on Information Theory (ISIT)*, 2007, pp. 1716–1720.

[90] M. Fekete, "Über die Verteilung der Wurzeln bei gewissen algebraischen Gleichungen mit ganzzahligen Koeffizienten," *Mathematische Zeitschrift*, vol. 17, no. 1, pp. 228–249, Dec 1923.

[91] R. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 18, no. 4, pp. 460–473, July 1972.

[92] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 18, no. 1, pp. 14–20, January 1972.

[93] K. A. S. Abdel-Ghaffar, "Capacity per unit cost of a discrete memoryless channel," *Electronics Letters*, vol. 29, no. 2, pp. 142–144, 1993.

[94] M. Jimbo and K. Kunisawa, "An iteration method for calculating the relative capacity," *Information and Control*, vol. 43, no. 2, pp. 216–223, 1979.

[95] F. F. Sellers, Jr., "Bit loss and gain correction code," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 35–38, 1962.

[96] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Doklady Akademii Nauk*, vol. 163, no. 4, pp. 845–848, 1965.

[97] R. R. Varshamov and G. M. Tenengolts, "Codes which correct single asymmetric errors," *Avtomatika i Telemekhanika*, vol. 26, no. 2, pp. 288–292, 1965.

[98] J. Brakensiek, V. Guruswami, and S. Zbarsky, "Efficient low-redundancy codes for correcting multiple deletions," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3403–3410, 2018.

[99] R. Gabrys and F. Sala, "Codes correcting two deletions," *IEEE Transactions on Information Theory*, vol. 65, no. 2, pp. 965–974, 2019.

[100] J. Sima, N. Raviv, and J. Bruck, "Two deletion correcting codes from indicator vectors," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 421–425.

[101] J. Sima and J. Bruck, "Optimal k-deletion correcting codes," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 847–851.

[102] V. Guruswami and J. Håstad, “Explicit two-deletion codes with redundancy matching the existential bound,” in *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2021, pp. 21–32.

[103] L. J. Schulman and D. Zuckerman, “Asymptotically good codes correcting insertions, deletions, and transpositions,” *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 2552–2557, 1999.

[104] V. Guruswami and C. Wang, “Deletion codes in the high-noise and high-rate regimes,” *IEEE Transactions on Information Theory*, vol. 63, no. 4, pp. 1961–1970, 2017.

[105] V. Guruswami and R. Li, “Efficiently decodable insertion/deletion codes for high-noise and high-rate regimes,” in *2016 IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 620–624.

[106] I. A. Kash, M. Mitzenmacher, J. Thaler, and J. Ullman, “On the zero-error capacity threshold for deletion channels,” in *2011 Information Theory and Applications Workshop*, 2011, pp. 1–5.

[107] B. Bukh, V. Guruswami, and J. Håstad, “An improved bound on the fraction of correctable deletions,” *IEEE Transactions on Information Theory*, vol. 63, no. 1, pp. 93–103, 2017.

[108] B. Haeupler and A. Shahrasbi, “Synchronization strings: Codes for insertions and deletions approaching the singleton bound,” in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, 2017, pp. 33–46.

[109] ——, “Synchronization strings: Explicit constructions, local decoding, and applications,” in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, 2018, pp. 841–854.

[110] B. Haeupler, A. Shahrasbi, and M. Sudan, “Synchronization strings: List decoding for insertions and deletions,” in *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP)*, 2018, pp. 76:1–76:14.

[111] B. Haeupler, A. Shahrasbi, and E. Vitercik, “Synchronization strings: Channel simulations and interactive coding for insertions and deletions,” in *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP)*, 2018, pp. 75:1–75:14.

[112] K. Cheng, B. Haeupler, X. Li, A. Shahrasbi, and K. Wu, "Synchronization strings: Highly efficient deterministic constructions over small alphabets," in *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2019, pp. 2185–2204.

[113] B. Haeupler and A. Shahrasbi, "Synchronization strings and codes for insertions and deletions – a survey," *IEEE Transactions on Information Theory*, 2021, to appear. Available at https://arxiv.org/abs/2101.00711.

[114] V. Guruswami and R. Li, "Coding against deletions in oblivious and online models," in *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2018, pp. 625–643.

[115] ——, "Polynomial time decodable codes for the binary deletion channel," *IEEE Transactions on Information Theory*, vol. 65, no. 4, pp. 2171–2178, 2019.

[116] R. Con and A. Shpilka, "Explicit and efficient constructions of coding schemes for the binary deletion channel," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 84–89.

[117] I. Tal, H. D. Pfister, A. Fazeli, and A. Vardy, "Polar codes for the deletion channel: Weak and strong polarization," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 1362–1366.

[118] V. I. Levenshtein, "Efficient reconstruction of sequences," *IEEE Transactions on Information Theory*, vol. 47, no. 1, pp. 2–22, Jan 2001.

[119] ——, "Efficient reconstruction of sequences from their subsequences or supersequences," *Journal of Combinatorial Theory, Series A*, vol. 93, no. 2, pp. 310–332, 2001.

[120] F. Sala, R. Gabrys, C. Schoeny, and L. Dolecek, "Exact reconstruction from insertions in synchronization codes," *IEEE Transactions on Information Theory*, vol. 63, no. 4, pp. 2428–2445, April 2017.

[121] R. Gabrys and E. Yaakobi, "Sequence reconstruction over the deletion channel," *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 2924–2931, April 2018.

[122] M. Horovitz and E. Yaakobi, "Reconstruction of sequences over non-identical channels," *IEEE Transactions on Information Theory*, vol. 65, no. 2, pp. 1267–1286, Feb 2019.

[123] E. Yaakobi and J. Bruck, “On the uncertainty of information retrieval in associative memories,” in *2012 IEEE International Symposium on Information Theory (ISIT)*, 2012, pp. 106–110.

[124] B. Haeupler and M. Mitzenmacher, “Repeated deletion channels,” in *2014 IEEE Information Theory Workshop (ITW)*, Nov 2014, pp. 152–156.

[125] S. R. Srinivasavaradhan, M. Du, S. Diggavi, and C. Fragouli, “On maximum likelihood reconstruction over multiple deletion channels,” in *2018 IEEE International Symposium on Information Theory (ISIT)*, June 2018, pp. 436–440.

[126] ——, “Symbolwise MAP for multiple deletion channels,” in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 181–185.

[127] O. Sabary, E. Yaakobi, and A. Yucovich, “The error probability of maximum-likelihood decoding over two deletion/insertion channels,” in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 763–768.

[128] S. Kannan and A. McGregor, “More on reconstructing strings from random traces: insertions and deletions,” in *2005 IEEE International Symposium on Information Theory (ISIT)*, 2005, pp. 297–301.

[129] K. Viswanathan and R. Swaminathan, “Improved string reconstruction over insertion-deletion channels,” in *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008, pp. 399–408.

[130] A. McGregor, E. Price, and S. Vorotnikova, “Trace reconstruction revisited,” in *22nd Annual European Symposium on Algorithms (ESA)*, 2014, pp. 689–700.

[131] Y. Peres and A. Zhai, “Average-case reconstruction for the deletion channel: Subpolynomially many traces suffice,” in *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, Oct 2017, pp. 228–239.

[132] R. Gabrys and O. Milenkovic, “Unique reconstruction of coded strings from multiset substring spectra,” *IEEE Transactions on Information Theory*, vol. 65, no. 12, pp. 7682–7696, 2019.

[133] X. Chen, A. De, C. H. Lee, R. A. Servedio, and S. Sinha, “Polynomial-time trace reconstruction in the low deletion rate regime,” in *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, 2021, pp. 20:1–20:20.

[134] L. Hartung, N. Holden, and Y. Peres, “Trace reconstruction with varying deletion probabilities,” in *Proceedings of the 15th Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, 2018, pp. 54–61.

[135] Z. Chase, “New upper bounds for trace reconstruction,” *arXiv e-prints*, p. arXiv:2009.03296, Sep. 2020.

[136] E. Grigorescu, M. Sudan, and M. Zhu, “Limitations of mean-based algorithms for trace reconstruction at small distance,” *arXiv e-prints*, p. arXiv:2011.13737, Nov. 2020.

[137] P. Borwein and T. Erdélyi, “Littlewood-type problems on subarcs of the unit circle,” *Indiana University mathematics journal*, pp. 1323–1346, 1997.

[138] P. Borwein, T. Erdélyi, and G. Kós, “Littlewood-type problems on $[0, 1]$,” *Proceedings of the London Mathematical Society*, vol. 79, no. 1, pp. 22–46, 1999.

[139] N. Holden and R. Lyons, “Lower bounds for trace reconstruction,” *Ann. Appl. Probab.*, vol. 30, no. 2, pp. 503–525, Apr. 2020.

[140] Z. Chase, “New lower bounds for trace reconstruction,” *arXiv e-prints*, p. arXiv:1905.03031, May 2019, to appear in Ann. Inst. Henri Poincaré Probab. Stat.

[141] W. Mao, S. N. Diggavi, and S. Kannan, “Models and information-theoretic bounds for nanopore sequencing,” *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 3216–3236, 2018.

[142] M. Kovačević and V. Y. F. Tan, “Codes in the space of multisets–Coding for permutation channels with impairments,” *IEEE Transactions on Information Theory*, vol. 64, no. 7, pp. 5156–5169, July 2018.

[143] J. Sima, N. Raviv, and J. Bruck, “On coding over sliced information,” in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 767–771.

[144] W. Song, K. Cai, and K. A. Schouhamer Immink, “Sequence-subset distance and coding for error control in DNA-based data storage,” *IEEE Transactions on Information Theory*, vol. 66, no. 10, pp. 6048–6065, 2020.

[145] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakobi, “Coding over sets for DNA storage,” *IEEE Transactions on Information Theory*, vol. 66, no. 4, pp. 2331–2351, 2020.

[146] R. Heckel, I. Shomorony, K. Ramchandran, and D. N. C. Tse, “Fundamental limits of DNA storage systems,” in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 3130–3134.

[147] A. Makur, “Information capacity of BSC and BEC permutation channels,” in *56th Annual Allerton Conference on Communication, Control, and Computing*, 2018, pp. 1112–1119.

[148] I. Shomorony and R. Heckel, “Capacity results for the noisy shuffling channel,” in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 762–766.

[149] A. Lenz, P. H. Siegel, A. Wachter-Zeh, and E. Yaakobi, “An upper bound on the capacity of the DNA storage channel,” in *2019 IEEE Information Theory Workshop (ITW)*, 2019, pp. 1–5.

[150] ——, “Achieving the capacity of the DNA storage channel,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8846–8850.

[151] A. Makur, “Coding theorems for noisy permutation channels,” *IEEE Transactions on Information Theory*, vol. 66, no. 11, pp. 6723–6748, 2020.

[152] S. Shin, R. Heckel, and I. Shomorony, “Capacity of the erasure shuffling channel,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8841–8845.

[153] I. Shomorony and A. Vahid, “Communicating over the torn-paper channel,” in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[154] J. Acharya, H. Das, O. Milenkovic, A. Orlitsky, and S. Pan, “String reconstruction from substring compositions,” *SIAM Journal on Discrete Mathematics*, vol. 29, no. 3, pp. 1340–1371, 2015.

[155] R. Gabrys and O. Milenkovic, “The hybrid k-deck problem: Reconstructing sequences from short and long traces,” in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 1306–1310.

[156] S. Pattabiraman, R. Gabrys, and O. Milenkovic, “Reconstruction and error-correction codes for polymer-based data storage,” in *2019 IEEE Information Theory Workshop (ITW)*, 2019, pp. 1–5.

[157] R. Gabrys, S. Pattabiraman, and O. Milenkovic, “Mass error-correction codes for polymer-based data storage,” in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 25–30.

[158] S. Pattabiraman, R. Gabrys, and O. Milenkovic, "Coding for polymer-based data storage," *arXiv e-prints*, p. arXiv:2003.02121, Mar. 2020.

[159] F. Ban, X. Chen, A. Freilich, R. A. Servedio, and S. Sinha, "Beyond trace reconstruction: Population recovery from the deletion channel," in *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, Nov 2019.

[160] F. Ban, X. Chen, R. A. Servedio, and S. Sinha, "Efficient average-case population recovery in the presence of insertions and deletions," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*, 2019, pp. 44:1–44:18.

[161] S. Narayanan, "Improved algorithms for population recovery from the deletion channel," in *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2021, pp. 1259–1278.

[162] S. Davies, M. Z. Racz, and C. Rashtchian, "Reconstructing trees from traces," in *Proceedings of the 32nd Conference on Learning Theory (COLT)*, 2019, pp. 961–978.

[163] A. Krishnamurthy, A. Mazumdar, A. McGregor, and S. Pal, "Trace reconstruction: Generalized and parameterized," in *27th Annual European Symposium on Algorithms (ESA)*, 2019, pp. 68:1–68:25.

[164] X. Chen, A. De, C. H. Lee, R. A. Servedio, and S. Sinha, "Polynomial-time trace reconstruction in the smoothed complexity model," in *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2021, pp. 54–73.

[165] S. Narayanan and M. Ren, "Circular trace reconstruction," *arXiv e-prints*, p. arXiv:2009.01346, Sep. 2020, to appear in ITCS 2021.

[166] O. Sabary, A. Yucovich, G. Shapira, and E. Yaakobi, "Reconstruction algorithms for DNA-storage systems," *bioRxiv*, 2020, DOI: 10.1101/2020.09.16.300186.

[167] V. Bhardwaj, P. A. Pevzner, C. Rashtchian, and Y. Safonova, "Trace reconstruction problems in computational biology," *IEEE Transactions on Information Theory*, 2020, to appear. DOI: 10.1109/TIT.2020.3030569.

[168] S. Davies, M. Z. Racz, C. Rashtchian, and B. G. Schiffer, "Approximate trace reconstruction," *arXiv e-prints*, p. arXiv:2012.06713, Dec. 2020.

[169] W. Rudin, *Real and Complex Analysis, 3rd Ed.* McGraw-Hill, 1987.

[170] I. Pinelis, "https://mathoverflow.net/a/299427," MathOverflow, last accessed February 7, 2021.

[171] D. B. Karp and E. G. Prilepkina, "Completely monotonic gamma ratio and infinitely divisible H-function of Fox," *Computational Methods and Function Theory*, vol. 16, no. 1, pp. 135–153, 2016.

[172] J. Cichoń, Z. Gołębiewski, M. Kardas, and M. Klonowski, "On delta-method of moments and probabilistic sums," in *Proceedings of the 10th Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, 2013, pp. 91–98.

[173] A. G. D'yachkov and P. A. Vilenkin, "Asymptotics of the Shannon and Renyi entropies for sums of independent random variables," in *1998 IEEE International Symposium on Information Theory (ISIT)*, 1998, p. 376.

[174] C. Knessl, "Integral representations and asymptotic expansions for Shannon and Renyi entropies," *Applied Mathematics Letters*, vol. 11, no. 2, pp. 69–74, 1998.

[175] M. R. Frey, "Information capacity of the Poisson channel," *IEEE Transactions on Information Theory*, vol. 37, no. 2, pp. 244–256, 1991.

[176] L. Wang, "The Poisson channel with varying dark current known to the transmitter," *IEEE Transactions on Information Theory*, vol. 65, no. 8, pp. 4966–4978, 2019.

[177] Y. Sakai, V. Y. F. Tan, and M. Kovačević, "Second- and third-order asymptotics of the continuous-time Poisson channel," *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 4742–4760, 2020.

[178] Y. Polyanskiy and Y. Wu, "Lecture notes on information theory," 2019, available at http://people.lids.mit.edu/yp/homepage/data/itlectures_v5.pdf, last accessed February 7, 2021.

[179] I. Csiszár, "Arbitrarily varying channels with general alphabets and states," *IEEE Transactions on Information Theory*, vol. 38, no. 6, pp. 1725–1742, 1992.

[180] D. Brady, "The analysis of optical, direct detection communication systems with point process observations," Ph.D. dissertation, Princeton University, 1990, available at https://search.proquest.com/docview/303849140, last accessed February 7, 2021.

[181] H. W. Chung, S. Guha, and L. Zheng, "On capacity of optical channels with coherent detection," in *49th Annual Allerton Conference on Communication, Control, and Computing*, 2011, pp. 879–885.

[182] G. Aminian, H. Arjmandi, A. Gohari, M. Nasiri-Kenari, and U. Mitra, "Capacity of diffusion-based molecular communication networks over LTI-Poisson channels," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 1, no. 2, pp. 188–201, 2015.

[183] A. Martinez, "A lower bound for the capacity of the discrete-time Poisson channel," in *2009 IEEE International Symposium on Information Theory (ISIT)*, 2009, pp. 2214–2215.

[184] J. Cao, S. Hranilovic, and J. Chen, "Lower bounds on the capacity of discrete-time Poisson channels with dark current," in *25th Biennial Symposium on Communications (QBSC)*, 2010, pp. 357–360.

[185] J. P. Gordon, "Quantum effects in communications systems," *Proceedings of the IRE*, vol. 50, no. 9, pp. 1898–1908, 1962.

[186] B. E. Goodwin and L. P. Bolgiano, "Information capacity of a photoelectric detector," *Proceedings of the IEEE*, vol. 53, no. 11, pp. 1745–1746, 1965.

[187] E. Hisdal, "Information in a photon beam vs modulation-level spacing," *J. Opt. Soc. Am.*, vol. 61, no. 3, pp. 328–332, Mar 1971.

[188] R. Jodoin and L. Mandel, "Information rate in an optical communication channel," *J. Opt. Soc. Am.*, vol. 61, no. 2, pp. 191–198, Feb 1971.

[189] Y. Yu, Z. Zhang, L. Wu, and J. Dang, "Lower bounds on the capacity for Poisson optical channel," in *6th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2014, pp. 1–5.

[190] J. Cao, S. Hranilovic, and J. Chen, "Capacity and nonuniform signaling for discrete-time Poisson channels," *J. Opt. Commun. Netw.*, vol. 5, no. 4, pp. 329–337, Apr 2013.

[191] T. Sutter, D. Sutter, P. M. Esfahani, and J. Lygeros, "Efficient approximation of channel capacities," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1649–1666, 2015.

[192] A. Das, "Capacity-achieving distributions for non-Gaussian additive noise channels," in *2000 IEEE International Symposium on Information Theory (ISIT)*, 2000, p. 432.

[193] A. Tchamkerten, "On the discreteness of capacity-achieving distributions," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2773–2778, Nov 2004.

[194] W. Oettli, "Capacity-achieving input distributions for some amplitude-limited channels with additive noise (corresp.)," *IEEE Transactions on Information Theory*, vol. 20, no. 3, pp. 372–374, 1974.

[195] S. Shamai and I. Bar-David, "The capacity of average and peak-power-limited quadrature Gaussian channels," *IEEE Transactions on Information Theory*, vol. 41, no. 4, pp. 1060–1071, 1995.

[196] I. C. Abou-Faycal, M. D. Trott, and S. Shamai (Shitz), "The capacity of discrete-time memoryless Rayleigh-fading channels," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1290–1301, 2001.

[197] M. Katz and S. Shamai, "On the capacity-achieving distribution of the discrete-time noncoherent and partially coherent AWGN channels," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2257–2270, 2004.

[198] M. C. Gursoy, H. V. Poor, and S. Verdú, "The noncoherent Rician fading channel-part I: structure of the capacity-achieving input," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2193–2206, 2005.

[199] A. Dytso, M. Goldenbaum, H. V. Poor, and S. Shamai (Shitz), "When are discrete channel inputs optimal?– Optimization techniques and some new results," in *52nd Annual Conference on Information Sciences and Systems (CISS)*, 2018, pp. 1–6.

[200] J. Cao, S. Hranilovic, and J. Chen, "Capacity-achieving distributions for the discrete-time Poisson channel - Part II: Binary inputs," *IEEE Transactions on Communications*, vol. 62, no. 1, pp. 203–213, 2014.

[201] A. Lapidoth and S. M. Moser, "Capacity bounds via duality with applications to multiple-antenna systems on flat-fading channels," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2426–2467, 2003.

[202] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*, ser. Probability and Mathematical Statistics. Academic Press, 1981.

[203] S. G. Krantz and H. R. Parks, *A Primer of Real Analytic Functions*, ser. Basler Lehrbücher. Birkhäuser, 1992, vol. 4.

[204] V. Guruswami, A. Rudra, and M. Sudan, *Essential Coding Theory.* Unpublished, 2019, draft available at https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book, last accessed February 7, 2021.

[205] D. Knuth, "Efficient balanced codes," *IEEE Transactions on Information Theory*, vol. 32, no. 1, pp. 51–53, January 1986.

[206] L. G. Tallini, R. M. Capocelli, and B. Bose, "Design of some new efficient balanced codes," *IEEE Transactions on Information Theory*, vol. 42, no. 3, pp. 790–802, 1996.

[207] K. A. Schouhamer Immink and J. H. Weber, "Very efficient balanced codes," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 2, pp. 188–192, February 2010.

[208] T. Gamelin, *Complex Analysis*, ser. Undergraduate Texts in Mathematics. Springer Science+Business Media New York, 2001.

[209] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, vol. 4, no. 4, pp. 373–395, Dec 1984.

[210] M. Abroshan, R. Venkataramanan, L. Dolecek, and A. Guillén i Fàbregas, "Coding for deletion channels with multiple traces," in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 1372–1376.

[211] J. Brakensiek, R. Li, and B. Spang, "Coded trace reconstruction in a constant number of traces," in *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, 2020, pp. 482–493.

[212] H. M. Kiah, T. Thanh Nguyen, and E. Yaakobi, "Coding for sequence reconstruction for single edits," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 676–681.

[213] J. Chrisnata, H. M. Kiah, and E. Yaakobi, "Optimal reconstruction codes for deletion channels," *arXiv e-prints*, p. arXiv:2004.06032, Apr. 2020.

[214] J. A. Adell, A. Lekuona, and Y. Yu, "Sharp bounds on the entropy of the Poisson law and related quantities," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2299–2306, 2010.

[215] D. Williams, *Probability with Martingales.* Cambridge University Press, 1991.

[216] P. Billingsley, *Convergence of Probability Measures*, 2nd ed., ser. Wiley Series in Probability and Statistics. John Wiley & Sons, 1999.

[217] K. L. Chung, *A Course in Probability Theory*, 3rd ed. Academic Press, 2001.

[218] M. Loève, *Probability Theory I*, 4th ed., ser. Graduate Texts in Mathematics. Springer-Verlag New York, 1977, vol. 45.

[219] D. G. Luenberger, *Optimization by Vector Space Methods*, ser. Series in Decision and Control. John Wiley & Sons, 1969.

Appendix A

Proofs from Chapter 3

A.1 Proof of (3.51)

In this section, we show that

$$\mathbb{E}[\Lambda_1(Y_x)] = \int_0^{\frac{2p}{1+2p}} \mathbb{E}[f_1(Y_x, t)]dt$$

for all $x \geq 1$, where $\Lambda_1(y) = \int_0^{\frac{2p}{1+2p}} f_1(y,t)dt$ with

$$f_1(y,t) = \frac{1 + t - ty(1-p)/p - \left(\frac{p-t}{p(1-t)}\right)^y/(1-t)}{t\ln(1-t)}.$$

Note that $f_1(y,\cdot)$ is continuous on $\left(0, \frac{2p}{1+2p}\right]$ and can be extended by continuity to $\left[0, \frac{2p}{1+2p}\right]$ for all $y \geq 0$. As in the proof of Lemma 3.2, the desired result follows by Fubini's theorem (Lemma 3.1) if we show that $f_1(y,t) \geq 0$ for all $t \in \left(0, \frac{2p}{1+2p}\right)$ and y large enough depending only on p.

Defining h_1 as

$$h_1(y,t) = 1 + t - ty(1-p)/p - \left(\frac{p-t}{p(1-t)}\right)^y/(1-t),$$

it suffices to show that $h_1(y,t) \leq 0$ for all t when y is large enough. First, we consider the case $t < p$. Noting that $h_1(y,0) = 0$ and $\frac{\partial h_1}{\partial t}(y,0) = 0$, the desired result follows if we show that $\frac{\partial^2 h_1}{\partial t^2}(y,t) \leq 0$ for all $t < p$ when y is large enough independent of t. When $t < p$, we can write

$$\frac{\partial^2 h_1}{\partial t^2}(y,t) = -\frac{(1-t/p)^{y-2}}{p^2(1-t)^{y+3}} \cdot (2(p-t)^2 - (1-p)(1+3p-4t)y + (1-p)^2y^2).$$

Now, it suffices to observe that

$$2(p-t)^2 - (1-p)(1+3p-4t)y + (1-p)^2y^2 \geq (1-p)^2y^2 - 4y > 0$$

for every $t \in [0, p)$ when $y > \frac{4}{(1-p)^2}$. On the other hand, when $p \leq 1/2$ (otherwise $p > \frac{2p}{1+2p} \geq t$) and $t \in \left[p, \frac{2p}{1+2p}\right]$ we have

$$h_1(y,t) \leq 1 + 1 - y(1-p) + (1+2p) \leq 5 - y(1-p),$$

where we have also used the fact that $\left|\frac{p-t}{p(1-t)}\right| \leq 1$. Therefore, we have $h_1(y,t) \leq 0$ for all such t when $y > \frac{5}{1-p}$, which concludes the argument.

A.2 Proof of (3.52)

In this section, we show that

$$\mathbb{E}[\Lambda_2(Y_x)] = \int_0^{\frac{2p}{1+2p}} \mathbb{E}[f_2(Y_x,t)]dt$$

for all $x \geq 1$, where $\Lambda_2(y) = \int_0^{\frac{2p}{1+2p}} f_2(y,t)dt$ with

$$f_2(y,t) = \frac{1 + t - ty/p - \left(\frac{p-t(1+p)}{p(1-t)}\right)^y/(1-t)}{t\ln(1-t)}.$$

Note that $f_2(y,\cdot)$ is continuous on $\left(0, \frac{2p}{1+2p}\right]$ and can be extended by continuity to $\left[0, \frac{2p}{1+2p}\right]$. As in the proof of Lemma 3.2, the desired result follows by Fubini's theorem (Lemma 3.1) if we show that $f_2(y,t) \geq 0$ for all $t \in \left(0, \frac{2p}{1+2p}\right)$ and y large enough depending only on p.

Defining h_2 as

$$h_2(y,t) = 1 + t - ty/p - \left(\frac{p-t(1+p)}{p(1-t)}\right)^y/(1-t),$$

it suffices to show that $h_2(y,t) \leq 0$ for all t when y is large enough. First, we consider the case $t < \frac{p}{1+p}$. Noting that $h_2(y,0) = 0$ and $\frac{\partial h_2}{\partial t}(y,0) = 0$, the desired result follows if we show that $\frac{\partial^2 h_2}{\partial t^2}(y,t) \leq 0$ for all such t when y is large enough. For $t < \frac{p}{1+p}$, we have

$$\frac{\partial^2 h_2}{\partial t^2}(y,t) = -\frac{\left(1 - \frac{t(1+p)}{p}\right)^{y-2}}{p^2(1-t)^{y+3}} \cdot (2(t-p(1-t))^2 - (1+4p(1-t)-4t)y + y^2).$$

Provided $t < \frac{p}{1+p}$, it suffices to observe that

$$2(t - p(1-t))^2 - (1 + 4p(1-t) - 4t)y + y^2 \geq y^2 - 5y > 0$$

when $y > 5$. On the other hand, when $t \in \left[\frac{p}{1+p}, \frac{2p}{1+2p}\right]$ we have

$$h_2(y,t) \leq 1 + 1 - \frac{y}{1+p} + (1+2p) \leq 5 - \frac{y}{1+p},$$

where we have used the fact that $t \geq \frac{p}{1+p}$ and $\left|\frac{p-t(1+p)}{p(1-t)}\right| \leq 1$ when $t \in \left[\frac{p}{1+p}, \frac{2p}{1+2p}\right]$. We have $h_2(y,t) \leq 0$ for all such t when $y > 5(1+p)$, which concludes the argument.

A.3 Proof of Lemma 3.9

In this section, we prove that

$$\Lambda_1(y) = \ln\Gamma\left(\frac{y(1-p)}{p}\right) + \frac{y(1-p)}{p} \cdot \mathrm{li}\left(\frac{1}{1+2p}\right) - \eta\left(\frac{1}{1+2p}\right) + O(1), \tag{A.1}$$

$$\Lambda_2(y) = \ln\Gamma\left(\frac{y}{p}\right) + \frac{y}{p} \cdot \mathrm{li}\left(\frac{1}{1+2p}\right) - \eta\left(\frac{1}{1+2p}\right) + O(1) \tag{A.2}$$

as $y \to \infty$, where $O(1)$ depends only on p, $\mathrm{li}(z) = \int_0^z \frac{dt}{\ln t}$ is the logarithmic integral and $\eta(z) = \int_0^z \frac{dt}{(1-t)\ln t}$, with Λ_1 and Λ_2 defined as in (3.49) and (3.50). The second part of Lemma 3.9 follows from the above by recalling that

$$g(y) = \Lambda_2(y) - \Lambda_1(y) - \ln(y!) - y \cdot \mathrm{li}\left(\frac{1}{1+2p}\right)$$

and invoking the asymptotic expansion of the log gamma function from Lemma 2.8, analogously to the proof of Lemma 3.3.

A.3.1 Proof of (A.1)

Making use of the integral representation of the log gamma function from Lemma 2.7, we have

$$\Lambda_1(y) - \ln\Gamma\left(\frac{y(1-p)}{p}\right) = \frac{y(1-p)}{p} \cdot \mathrm{li}\left(\frac{1}{1+2p}\right) - \eta\left(\frac{1}{1+2p}\right)$$

$$+\int_0^{\frac{2p}{1+2p}} \frac{(1-t)^{\frac{y(1-p)}{p}} - \left(\frac{p-t}{p(1-t)}\right)^y}{t(1-t)\ln(1-t)} dt + \int_{\frac{2p}{1+2p}}^1 \frac{(1-t)^{\frac{y(1-p)}{p}}}{t(1-t)\log(1-t)} dt.$$

Observe that

$$\left| \int_{\frac{2p}{1+2p}}^1 \frac{(1-t)^{\frac{y(1-p)}{p}}}{t(1-t)\log(1-t)} dt \right| \to 0$$

when $y \to \infty$. Therefore, it suffices to show that

$$0 \leq \int_0^{\frac{2p}{1+2p}} \frac{(1-t)^{\frac{y(1-p)}{p}} - \left(\frac{p-t}{p(1-t)}\right)^y}{-t\ln(1-t)} dt = O(1).$$

We prove this following the lines of the proof of Lemma 3.3. First, we have that

$$\int_{p/2}^{\frac{2p}{1+2p}} \frac{(1-t)^{\frac{y(1-p)}{p}} - \left(\frac{p-t}{p(1-t)}\right)^y}{-t\ln(1-t)} dt \leq \int_{p/2}^{\frac{2p}{1+2p}} \frac{(1-t)^{\frac{y(1-p)}{p}} + \left|\frac{p-t}{p(1-t)}\right|^y}{-t\ln(1-t)} dt = O(1)$$

when $y \to \infty$, since $\left|\frac{p-t}{p(1-t)}\right| \leq 1$ whenever $t \leq \frac{2p}{1+2p}$. As a result, it is now enough to show that

$$\int_0^{p/2} \frac{(1-t)^{\frac{y(1-p)}{p}} - \left(\frac{p-t}{p(1-t)}\right)^y}{-t\ln(1-t)} dt = O(1).$$

As before, we follow the approach of Pinelis [170]. Define $a_1(t) = \ln\left(\frac{p-t}{p(1-t)}\right)$, $a_2(t) = \frac{1-p}{p} \cdot \ln(1-t)$, and $\alpha(t) = a_2(t) - a_1(t)$. Note that $\alpha(t) \geq 0$ for all $t \in [0, p/2]$. Then, we have

$$\begin{aligned}
\int_0^{p/2} \frac{(1-t)^{\frac{y(1-p)}{p}} - \left(\frac{p-t}{p(1-t)}\right)^y}{-t\ln(1-t)} dt &= \int_0^{p/2} \frac{e^{a_2(t)y} - e^{a_1(t)y}}{-t\ln(1-t)} dt \\
&\leq \int_0^{p/2} \frac{\alpha(t) y e^{a_2(t)y}}{-t\ln(1-t)} dt \\
&\leq \frac{2(1-p)}{p^2} \cdot \int_0^{p/2} y e^{a_2(t)y} dt \\
&\leq \frac{2(1-p)}{p^2} \cdot \int_0^{p/2} y e^{-\frac{ty(1-p)}{p}} dt \\
&\leq \frac{2(1-p)}{p^2} \cdot \int_0^{\infty} y e^{-\frac{ty(1-p)}{p}} dt \\
&= 2/p. \qquad \text{(A.3)}
\end{aligned}$$

The first equality holds because $e^b - e^a \leq (b-a)e^b$ when $b \geq a$. The second inequality holds because $\alpha(t) \leq \frac{2(1-p)t^2}{p^2}$ when $t \in [0, p/2]$, and $\ln(1-t) \leq -t$. The third inequality holds also because $\ln(1-t) \leq -t$.

A.3.2 Proof of (A.2)

The proof proceeds analogously to the proof of (A.1). The only step where they differ is that here one must show that

$$\int_0^{p/2} \frac{(1-t)^{y/p} - \left(\frac{p-t(1+p)}{p(1-t)}\right)^y}{-t\ln(1-t)} dt = O(1).$$

We show this holds following Pinelis [170]. Define $b_1(t) = \ln\left(\frac{p-t(1+p)}{p(1-t)}\right)$, $b_2(t) = \frac{1}{p} \cdot \ln(1-t)$, and $\beta(t) = b_2(t) - b_1(t)$. Note that $\beta(t) \geq 0$ for all $t \in [0, \frac{p}{1+p})$ and $\beta(t) \leq \frac{2(1+p)t^2}{p^2(1-p)}$ when $t \leq p/2$. Then, following the reasoning used to obtain (A.3), we have

$$\begin{aligned}
\int_0^{p/2} \frac{(1-t)^{y/p} - \left(\frac{p-t(1+p)}{p(1-t)}\right)^y}{-t\ln(1-t)} dt &= \int_0^{p/2} \frac{e^{b_2(t)y} - e^{b_1(t)y}}{-t\ln(1-t)} dt \\
&\leq \int_0^{p/2} \frac{\beta(t) y e^{b_2(t)y}}{-t\ln(1-t)} dt \\
&\leq \frac{2(1+p)}{p^2(1-p)} \cdot \int_0^{p/2} y e^{b_2(t)y} dt \\
&\leq \frac{2(1+p)}{p^2(1-p)} \cdot \int_0^{p/2} y e^{-ty/p} dt \\
&\leq \frac{2(1+p)}{p^2(1-p)} \cdot \int_0^{\infty} y e^{-ty/p} dt \\
&= \frac{2(1+p)}{p(1-p)}.
\end{aligned}$$

A.4 Proof of (3.74) and (3.75)

In this section, we prove (3.74) and (3.75) following exactly the approach from Section 3.1.3.

Recall that we are dealing with distributions $Y^{(q)}$ of the form

$$Y^{(q)}(y) = y_0 q^y \exp(g(y) - y h^{(e)}(p)), \quad y = 0, 1, 2, \ldots$$

with g satisfying

$$C_L/\sqrt{y} \leq \exp(g(y) - y h^{(e)}(p)) \leq C_U/\sqrt{y}$$

for some constants $C_L, C_U > 0$ and all integers $y \geq 1$. We set $\delta = 1$ here for ease of exposition, but the proof goes through in the same way for any $\delta \in (0, 1]$. Our goal is to show that for every $\mu > 0$

there exists $q \in (0,1)$ such that $\mathbb{E}[Y^{(q)}] = \mu$. As in Section 3.1.3, it suffices to show that

$$\lim_{q \to 0^+} \mathbb{E}[Y^{(q)}] = 0 \tag{A.4}$$

and

$$\lim_{q \to 1^-} \mathbb{E}[Y^{(q)}] = \infty. \tag{A.5}$$

To see (A.4), observe that

$$\begin{aligned} 0 \leq \mathbb{E}[Y^{(q)}] &= \frac{\sum_{y=1}^{\infty} y q^y \exp(g(y) - y h^{(e)}(p))}{\sum_{y=0}^{\infty} q^y \exp(g(y) - y h^{(e)}(p))} \\ &\leq \frac{\sum_{y=1}^{\infty} y q^y \exp(g(y) - y h^{(e)}(p))}{\exp(g(0))} \\ &\leq C_U \frac{\sum_{y=1}^{\infty} \sqrt{y} q^y}{\exp(g(0))} \to 0 \end{aligned}$$

when $q \to 0^+$. It remains to show (A.5). Similarly to Section 3.1.3, this follows from the fact that

$$\begin{aligned} \mathbb{E}[Y^{(q)}] &\geq \frac{C_L \sum_{y=1}^{\infty} \sqrt{y} q^y}{\exp(g(0)) + C_U \sum_{y=1}^{\infty} q^y / \sqrt{y}} \\ &\geq \frac{C_L \sqrt{k} \sum_{y=k}^{\infty} q^y}{\exp(g(0)) + C_U \sum_{y=1}^{\infty} q^y} \\ &\geq \frac{C_L \sqrt{k} \cdot \frac{q^k}{1-q}}{\exp(g(0)) + \frac{C_U}{1-q}} \\ &\geq \frac{C_L \sqrt{k} q^k}{\exp(g(0)) + C_U} \end{aligned}$$

for all $k > 0$ and $q \in (0,1)$.

Appendix B

On capacity-achieving distributions for the DTP channel

B.1 A crash course on weak convergence

In order to prove the existence of capacity-achieving distributions for the DTP channel under an average-power constraint only in Section B.2, some basic concepts related to measure-theoretic probability and weak convergence of distributions are required. We introduce them here so that the exposition is mostly self-contained, requiring only basic familiarity with the concept of integration with respect to a measure. For an enjoyable and complete introduction to measure-theoretic probability, the book of Williams [215] is recommended, especially [215, Chapter 5]. For a deeper and more general treatment of weak convergence, the book of Billingsley [216] is recommended.

We denote by $\mathcal{F}$ the set of all cumulative distribution functions (cdfs) F. More precisely, $\mathcal{F}$ consists of all non-decreasing, right-continuous functions $F : \mathbb{R} \to [0,1]$ satisfying $\lim_{x\to+\infty} F(x) = 1$ and $\lim_{x\to-\infty} F(x) = 0$. To each cdf F we can associate the probability measure μ_F on $\mathbb{R}$ with the Borel σ-algebra uniquely determined by the requirement that $\mu_F((a,b]) = F(b) - F(a)$. Then, we write $\int g(x)dF(x)$ for the Lebesgue integral of the function $g : \mathbb{R} \to \mathbb{R}$ with respect to the measure μ_F induced by the cdf F.

We can now proceed to define the notion of weak convergence for cdfs.

Definition B.1 (Weak convergence)**.** *A sequence of cdfs* $(F_n)_{n\in\mathbb{N}}$ *is said to be* weakly convergent to

F, denoted $F_n \xrightarrow{w} F$, if for all bounded continuous functions $g : \mathbb{R} \to \mathbb{R}$ it holds that

$$\int g(x)dF_n(x) \to \int g(x)dF(x).$$

The following lemma collects some well known properties of weak convergence that we will require in the next section.

Lemma B.1 ([216, Theorem 2.1], [217, Theorem 4.4.4 and following remark], [218, Section 11.4.A])**.** *Suppose that $F_n \xrightarrow{w} F$. Then, the following hold:*

1. *If $\mathcal{I} \subseteq \mathbb{R}$ is an open interval, then*

$$\int_{\mathcal{I}} dF(x) \leq \liminf_{n\to\infty} \int_{\mathcal{I}} dF_n(x);$$

2. *If $g : \mathbb{R} \to \mathbb{R}$ is continuous and bounded below, then*

$$\int g(x)dF(x) \leq \liminf_{n\to\infty} \int g(x)dF_n(x);$$

3. *If $g : \mathbb{R} \to \mathbb{R}$ is continuous and uniformly integrable in $(F_n)_{n\in\mathbb{N}}$, meaning that for b large enough we have*

$$\int_{|x|>b} |g(x)|dF_n(x) \leq \varepsilon(b)$$

for all n with $\lim_{b\to\infty} \varepsilon(b) = 0$, then

$$\int g(x)dF_n(x) \to \int g(x)dF(x).$$

It is known that weak convergence of cdfs is equivalent to convergence in the Lévy-Prokhorov metric, which allows us to define compactness and continuity in a more useful manner. Since we only use this result indirectly, we refrain from defining these concepts and state only the useful consequences (see [216, Section 6] for more details).

Definition B.2 (Weak compactness)**.** *A set $\Omega \subseteq \mathcal{F}$ is said to be* weakly compact *if for every sequence $(F_n)_{n\in\mathbb{N}}$ in Ω there exists a subsequence $(F_{n_j})_{j\in\mathbb{N}}$ such that $F_{n_j} \xrightarrow{w} F$ for some $F \in \Omega$.*

A related and useful definition is that of tightness.

Definition B.3 (Tightness). *A set $\Omega \subseteq \mathcal{F}$ is said to be* tight *if for every $\varepsilon > 0$ there exists a constant $C > 0$ such that $F(-C) + (1 - F(C)) \leq \varepsilon$ for every $F \in \Omega$.*

There is a well known relation between tightness and relative compactness, as made precise in the following lemma.

Lemma B.2 (Prokhorov's theorem [216, Theorem 5.1], adapted). *If $\Omega \subseteq \mathcal{F}$ is tight, then for every sequence $(F_n)_{n\in\mathbb{N}}$ in Ω there exists a subsequence $(F_{n_j})_{j\in\mathbb{N}}$ such that $F_{n_j} \xrightarrow{w} F$ for some $F \in \mathcal{F}$.*

Definition B.4 (Weak continuity). *A functional $J : \Omega \to \mathbb{R}$ is said to be* weakly continuous on Ω *if for every $F \in \Omega$ and sequence $(F_n)_{n\in\mathbb{N}}$ in Ω such that $F_n \xrightarrow{w} F$ we have $J(F_n) \to J(F)$.*

The following lemma generalises Weierstrass' extreme value theorem, and is key for proving the existence of capacity-achieving distributions.

Lemma B.3 ([219, Section 5.10]). *If $J : \Omega \to \mathbb{R}$ is weakly continuous on a weakly compact set $\Omega \subseteq \mathcal{F}$, then J is bounded and achieves its maximum over Ω.*

B.2 Existence of capacity-achieving distributions for the DTP channel

In this section, we argue that capacity-achieving distributions exist for the DTP channel under an average-power constraint. For simplicity, we will assume that $\lambda = 0$. The proof proceeds analogously for all $\lambda \geq 0$. Shamai [42] showed that capacity-achieving distributions exist for the DTP channel under a peak-power constraint (using the same approach as Smith [52]). Proving the existence of such distributions under an average-power constraint only requires a somewhat different approach, such as the one used in [196, Appendix I] for Rayleigh-fading channels. Our approach follows [196, Appendix I] closely with only minor differences. We provide it here for completeness.

Before we proceed, we define some relevant concepts. Let

$$\Omega_\mu = \left\{ F \in \mathcal{F} : \lim_{x\to 0^-} F(x) = 0, \int x dF(x) \leq \mu \right\}.$$

In other words, Ω_μ (seen as a subset of the vector space over $\mathbb{R}$ consisting of all functions $f : \mathbb{R} \to \mathbb{R}$ with pointwise addition and scalar multiplication) is the convex set[1] of cdfs associated to non-negative

[1] A subset Ω of a vector space over $\mathbb{R}$ is *convex* if $\lambda x + (1-\lambda)y \in \Omega$ for all $\lambda \in [0,1]$ whenever $x, y \in \Omega$.

random variables with expected value at most μ. We also define

$$\Omega_\mu^= = \left\{ F \in \mathcal{F} : \lim_{x \to 0^-} F(x) = 0, \int x dF(x) = \mu \right\},$$

which is the convex set of cdfs associated to non-negative random variables with expected value exactly μ. Given some cdf $F \in \Omega_\mu$, we denote by $I(F)$ the functional which maps F to the mutual information $I(X_F; Y_F)$, where X_F has cdf F and is the input to the DTP channel, and Y_F is the corresponding output satisfying

$$Y_F(y) = \int Y_x(y) dF(x), \quad y = 0, 1, \ldots.$$

Then, we can write

$$C(\mu) = \sup_{F \in \Omega_\mu} I(F).$$

Our goal is to use Lemma B.3 with $\Omega = \Omega_\mu$ and $J(F) = I(F)$ in order to conclude that the supremum is achieved by some $F^\star$, the capacity-achieving distribution. We accomplish this via the following two lemmas.

Lemma B.4. *The set Ω_μ is weakly compact.*

Proof. First, we show that Ω_μ is tight. Then, Lemma B.2 ensures that every sequence (F_n) in Ω_μ has a weakly convergent subsequence $F_{n_j} \xrightarrow{w} F$, and it suffices to show that $F \in \Omega_\mu$.

To see that Ω_μ is tight, it is enough to observe that $1 - F(C) \le \mu/C$ for every $F \in \Omega_\mu$ and $C > 0$ by the average-power constraint, as otherwise $\int x dF(x) \ge C(1 - F(C)) > \mu$. Then, we can take $C = \mu/\varepsilon$ for each $\varepsilon > 0$. Therefore, Lemma B.2 ensures that every sequence $(F_n)_{n \in \mathbb{N}}$ in Ω_μ has a weakly convergent subsequence $F_{n_j} \xrightarrow{w} F$ for some $F \in \mathcal{F}$. We show that $F \in \Omega_\mu$. First, by Lemma B.1, for the open interval $\mathcal{I} = (-\infty, 0)$ we have

$$0 \le \lim_{x \to 0^-} F(x) = \int_{\mathcal{I}} dF(x) \le \liminf_{n \to \infty} \int_{\mathcal{I}} dF_n(x) = 0.$$

Second, since the function $g(x) = \max(0, x)$ is continuous and bounded from below over $\mathbb{R}$, again by Lemma B.1 we have

$$\int x dF(x) = \int g(x) dF(x) \le \liminf_{n \to \infty} \int g(x) dF_n(x) \le \mu.$$

These two observations show that $F \in \Omega_\mu$, as desired. □

Lemma B.5. *$I(\cdot)$ is weakly continuous on Ω_μ.*

Proof. For $F \in \Omega_\mu$ we have

$$I(F) = H(Y_F) - \int H(Y_x) dF(x),$$

where Y_F is the output distribution induced by F. We show that the two terms in the right hand side are weakly continuous.

First, we show that $F \mapsto \int H(Y_x)dF(x)$ is weakly continuous on Ω_μ. This follows by Lemma B.1 if we show that $x \mapsto H(Y_x)$ is uniformly integrable, i.e., that for b large enough we have

$$\int_{x>b} H(Y_x)dQ(x) \leq \varepsilon(b)$$

for every $Q \in \Omega_\mu$ with $\lim_{b\to\infty} \varepsilon(b) = 0$, given that $H(Y_x)$ (with $H(Y_x) = 0$ when $x < 0$) is a non-negative and continuous function of x. Since $H(Y_x) = O(\log(1+x)) = O(\sqrt{x})$ [214], it suffices to note that for b large enough we have $H(Y_x) < C\sqrt{x}$ when $x > b$ for some absolute constant $C > 0$, and so

$$\int_{x>b} H(Y_x)dQ(x) \leq C\int_{x>b} \sqrt{x}dQ(x) \leq \frac{C}{\sqrt{b}}\int_{x>b} xdQ(x) \leq \frac{C\mu}{\sqrt{b}}$$

by the average-power constraint on Q.

It remains to show that $F \mapsto H(Y_F)$ is weakly continuous on Ω_μ. Fix a sequence $F_n \xrightarrow{w} F$. Then, we have

$$\begin{aligned}
\lim_{n\to\infty} H(Y_{F_n}) &= -\lim_{n\to\infty} \sum_{y=0}^{\infty} Y_{F_n}(y)\log Y_{F_n}(y) \\
&= -\sum_{y=0}^{\infty} \lim_{n\to\infty} Y_{F_n}(y)\log Y_{F_n}(y) \qquad \text{(B.1)} \\
&= -\sum_{y=0}^{\infty} Y_F(y)\log Y_F(y) \qquad \text{(B.2)} \\
&= H(Y_F).
\end{aligned}$$

The first and last equalities follow by definition. To show (B.2), note that, for fixed y, the function $x \mapsto Y_x(y)$ is a bounded, continuous function of x. Therefore, weak convergence implies that

$$Y_{F_n}(y) = \int Y_x(y)dF_n(x) \to \int Y_x(y)dF(x) = Y_F(y).$$

Since $x \mapsto x \log x$ is continuous on $[0,1]$, we conclude that (B.2) holds. It remains to prove (B.1). It suffices to show that we are in a condition to apply the dominated convergence theorem. More specifically, we need to prove that

$$|Y_Q(y) \log Y_Q(y)| \leq f(y) \tag{B.3}$$

for all $Q \in \Omega_\mu$ and $y \in \mathbb{N}_0$, where f satisfies $\sum_{y=0}^{\infty} f(y) < \infty$. Fix $Q \in \Omega_\mu$, and note that

$$Y_Q(y) = \int e^{-x}\frac{x^y}{y!}dQ(x) = \int_{x<y-y^{0.9}} e^{-x}\frac{x^y}{y!}dQ(x) + \int_{x \geq y-y^{0.9}} e^{-x}\frac{x^y}{y!}dQ(x). \tag{B.4}$$

We analyse the two terms. First, since $x \mapsto Y_x(y)$ is increasing for $x < y$ and decreasing for $x > y$, we have

$$\int_{x<y-y^{0.9}} e^{-x}\frac{x^y}{y!}dQ(x) \leq Y_{y-y^{0.9}}(y) \leq e^{-\Omega(y^{0.8})}, \tag{B.5}$$

where the last inequality follows from the concentration bound for the Poisson distribution from Lemma 2.4. For fixed y, we have that $Y_x(y)$ is maximised when $x = y$. Furthermore, we have $Y_y(y) = O(1/\sqrt{y})$ by Lemma 2.8. Since $Q \in \Omega_\mu$, we have $1 - Q(x) \leq \mu/x$ for all $x > 0$, and so

$$\int_{x \geq y-y^{0.9}} e^{-x}\frac{x^y}{y!}dQ(x) \leq \frac{\mu Y_y(y)}{y - y^{0.9}} = O(y^{-3/2}) \tag{B.6}$$

when $y \geq 2$. Combining (B.4) with (B.5) and (B.6) implies that there exist absolute constants $C_0, C_1 > 0$ such that for all $y \geq C_0$ and $Q \in \Omega_\mu$ we have

$$Y_Q(y) \leq \frac{C_1}{y^{3/2}}. \tag{B.7}$$

Observe that, due to (B.7), for every $\varepsilon > 0$ there is a constant y_ε (possibly depending on μ, but independent of Q) such that $Y_Q(y) \leq \varepsilon$ for all $Q \in \Omega_\mu$ when $y \geq y_\varepsilon$. Therefore, there exist absolute constants $C_2, C_3 > 0$ such that for all $y \geq C_2$ and $Q \in \Omega_\mu$ we have

$$|Y_Q(y) \log Y_Q(y)| \leq Y_Q(y)^{0.7} \leq \frac{C_3}{y^{1.05}},$$

where we used (B.7) in the last inequality. Consequently, (B.3) follows by noting that $\sum_{y=1}^{\infty} y^{-1.05} < \infty$ and setting $f(y) = \frac{C_3}{y^{1.05}}$ when $y \geq C_2$ and $f(y) = 1 \geq \max_{p \in [0,1]} |p \log p|$ when $y < C_2$. □

Combining Lemmas B.3, B.4, and B.5 implies that for every $\mu \geq 0$ there exists a capacity-achieving $F^\star \in \Omega_\mu$ such that $C(\mu) = I(F^\star)$, as desired. However, we can show more: If $F^\star \in \Omega_\mu$ is capacity-achieving, then $\mathbb{E}[X_{F^\star}] = \mu$. This is trivial for $\mu = 0$, so assume that $\mu > 0$. Suppose that $\mathbb{E}[X_{F^\star}] = \mu' < \mu$. Then, we must have $C(\mu'') = I(F^\star)$ for all $\mu'' \in [\mu', \mu]$. Since $C(\mu)$ is concave and non-decreasing in μ, this implies that $C(\mu'') = I(F^\star)$ for all $\mu'' \geq \mu'$. However, we know from (4.3) that $C(\mu)$ is unbounded when $\mu \to \infty$. This is a contradiction, and so $\mathbb{E}[X_{F^\star}] = \mu$ necessarily.

B.3 Proof of Theorem 4.1

In this section, we prove Theorem 4.1 for the DTP channel, although our proof can be easily extended to a broad class of well-behaved channels. As we already saw in Chapter 4, the first part of the theorem (capacity upper bounds) is an immediate consequence of general duality-based results [201]. However, the second part (optimality conditions) requires a more careful discussion. We follow the approach used to obtain analogous results for other constrained channels in [52, 42, 196] with minor modifications only, and we focus on [196, Appendix II] in particular.

Before we proceed with the proof of Theorem B.1, we need some auxiliary definitions and results. Given a functional $J\colon \Omega \to \mathbb{R}$, where Ω is a convex subset of a vector space over $\mathbb{R}$, the *weak derivative of J at $F \in \Omega$ in the direction of $Q \in \Omega$*, denoted by $J'_F(Q)$, is defined as

$$J'_F(Q) = \lim_{\theta \to 0^+} \frac{J((1-\theta)F + \theta Q) - J(F)}{\theta}.$$

The functional J is said to be *weakly differentiable on Ω at F* if $J'_F(Q)$ exists for all $Q \in \Omega$. If J is weakly differentiable on Ω at F for all $F \in \Omega$, then we simply say J is *weakly differentiable on* Ω. We have the following result.

Lemma B.6. *Fix a functional $J\colon \Omega \to \mathbb{R}$ on a convex subset Ω of a vector space over $\mathbb{R}$. If $F^\star \in \Omega$ is a maximiser of J in Ω and $J'_{F^\star}(Q)$ exists, then $J'_{F^\star}(Q) \leq 0$.*

Proof. Fix J satisfying the conditions of the lemma statement, and let $F^\star \in \Omega$ be a maximiser of J over Ω. Therefore,

$$\frac{J((1-\theta)F^\star + \theta Q) - J(F^\star)}{\theta} \leq 0$$

for every $\theta \in (0, 1]$, since $(1-\theta)F^\star + \theta Q \in \Omega$ by the convexity of Ω and $J(F^\star) \geq J(F)$ for every $F \in \Omega$ by hypothesis. As a result, if $J'_{F^\star}(Q)$ exists, then we must have $J'_{F^\star}(Q) \leq 0$. □

The following lemma states a form of convex duality.

Lemma B.7 ([219, Section 8.3, Theorem 1, specialised])**.** *Let $J, G\colon \Omega \to \mathbb{R}$ be convex functionals, where Ω is a convex subset of a vector space. Suppose there exists $F \in \Omega$ such that $G(F) < 0$ and that $\inf\{J(F) : G(F) \leq 0, F \in \Omega\}$ is finite. Then, there is $z \geq 0$ such that*

$$\inf\{J(F) : G(F) \leq 0, F \in \Omega\} = \inf\{J(F) + zG(F) : F \in \Omega\}.$$

Moreover, if the infimum on the left hand side is achieved by some $F^\star$, then $F^\star$ also achieves the infimum on the right hand side and $zG(F^\star) = 0$.

The following lemma characterises the weak derivative of $I(\cdot)$. A proof of this result can be found in [53, Appendix A], but its presentation is specialised for noise-additive channels. For completeness, we present a proof using our notation and following their reasoning with minor modifications. Before we proceed, we define the convex set

$$\Omega_{\mathsf{fin}} = \left\{F \in \mathcal{F} : \lim_{x \to 0^-} F(x) = 0, \int x dF(x) < \infty\right\}$$

of cdfs associated to non-negative random variables with finite expectation.

Lemma B.8 (Implicit in [53, Appendix A])**.** *Fix cdfs $F, Q \in \Omega_{\mathsf{fin}}$ and suppose that*

$$\int D_{\mathsf{KL}}(Y_x || Y_F) dQ(x) < \infty \quad \text{and} \quad \int H(Y_x) dQ(x) < \infty.$$

Then, $I'_F(Q)$ exists and is given by

$$I'_F(Q) = \int D_{\mathsf{KL}}(Y_x \| Y_F) dQ(x) - I(F).$$

Proof. Let $F_\theta = (1-\theta)F + \theta Q$ for $\theta \in [0, 1]$. Denote the discrete channel output associated to F_θ by $Y_\theta = (1-\theta)Y_0 + \theta Y_1$, where Y_0 and Y_1 denote the discrete channel outputs of F and Q, respectively. For $\theta \in (0, 1]$ we have

$$\begin{aligned}\frac{I(F_\theta) - I(F)}{\theta} &= \frac{H(Y_\theta) - H(Y_0)}{\theta} + \int H(Y_x) dF(x) - \int H(Y_x) dQ(x) \\ &= \frac{1-\theta}{\theta}\left(-\sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) \log Y_\theta(y) - H(Y_0)\right) - \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_\theta(y)\end{aligned}$$

$$- \int H(Y_x) dQ(x) - I(F). \tag{B.8}$$

First, we show that

$$- \lim_{\theta \to 0^+} \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_\theta(y) = - \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_0(y). \tag{B.9}$$

This follows by the dominated convergence theorem and the hypothesis of the lemma. Observe that

$$0 \leq -\log Y_\theta(y) = -\log((1-\theta)Y_0(y) + \theta Y_1(y)) \leq -\log((1-\theta)Y_0(y)) \leq -\log Y_0(y) + 2 \tag{B.10}$$

for all y, provided that θ is small enough. It suffices to show that $\sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) f(y) < \infty$ for $f(y) = -\log Y_0(y) + 2$. Note that $\int H(Y_x) dQ(x) < \infty$ by hypothesis, and

$$\begin{aligned} \infty > \int D_{\mathsf{KL}}(Y_x || Y_0) dQ(x) &= - \int H(Y_x) dQ(x) - \int \sum_{y=0}^{\infty} Y_x(y) \log Y_0(y) dQ(x) \\ &= - \int H(Y_x) dQ(x) - \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_0(y). \end{aligned}$$

The first inequality follows by hypothesis, the first equality holds under the convention that $0 \log 0 = 0$, and the second equality follows by applying Fubini's theorem to exchange integral and sum. Consequently, it holds that

$$- \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_0(y) < \infty.$$

In particular, we have $\mathsf{supp}(Y_1) \subseteq \mathsf{supp}(Y_0)$ and $\sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) f(y) < \infty$. As a result, the dominated convergence theorem yields

$$- \lim_{\theta \to 0^+} \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_\theta(y) = - \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \lim_{\theta \to 0^+} \log Y_\theta(y) = - \sum_{y \in \mathsf{supp}(Y_1)} Y_1(y) \log Y_0(y).$$

We now show that

$$\lim_{\theta \to 0^+} \frac{1-\theta}{\theta} \left(- \sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) \log Y_\theta(y) - H(Y_0) \right) = 0. \tag{B.11}$$

Let $h(\theta, y) = -\log Y_\theta(y)$. Then, we have

$$\frac{\partial h}{\partial \theta}(\theta, y) = \frac{Y_0(y) - Y_1(y)}{(1-\theta)Y_0(y) + \theta Y_1(y)}$$

for all $\theta \in (0,1)$ and $y \in \mathsf{supp}(Y_0)$. In particular, $h(\cdot, y)$ is continuous on $[0, 1/2]$ and differentiable on $(0, 1/2)$ for every $y \in \mathsf{supp}(Y_0)$. Moreover, we have

$$\left|\frac{\partial h}{\partial \theta}(\theta, y)\right| \leq \frac{1}{1-\theta}\left|1 - \frac{Y_1(y)}{Y_0(y)}\right| \leq 2\left(1 + \frac{Y_1(y)}{Y_0(y)}\right) \tag{B.12}$$

for all $y \in \mathsf{supp}(Y_0)$, provided that $\theta \in (0, 1/2)$. Let $g(y) = 2\left(1 + \frac{Y_1(y)}{Y_0(y)}\right)$. Observe that

$$\sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) g(y) = 2\left(1 + \sum_{y \in \mathsf{supp}(Y_0)} Y_1(y)\right) = 4 \tag{B.13}$$

since $\mathsf{supp}(Y_1) \subseteq \mathsf{supp}(Y_0)$, and so g is integrable with respect to Y_0. We can write the left hand side of (B.11) as

$$\lim_{\theta \to 0^+} \sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) \cdot \frac{h(\theta, y) - h(0, y)}{\theta}.$$

Since $h(\cdot, y)$ is continuous on $[0, 1/2]$ and differentiable on $(0, 1/2)$ for every $y \in \mathsf{supp}(Y_0)$, from the mean value theorem we have that for every $\theta \in (0, 1/2]$ and $y \in \mathsf{supp}(Y_0)$ there exists some $z \in (0, \theta)$ such that

$$\frac{\partial h}{\partial \theta}(z, y) = \frac{h(\theta, y) - h(0, y)}{\theta}.$$

Taking into account (B.12), it follows that

$$\left|\frac{h(\theta, y) - h(0, y)}{\theta}\right| \leq g(y)$$

for all $y \in \mathsf{supp}(Y_0)$ and $\theta \in (0, 1/2]$. Recalling (B.13) and the fact that $\mathsf{supp}(Y_1) \subseteq \mathsf{supp}(Y_0)$, the dominated convergence theorem implies that

$$\begin{aligned}
\lim_{\theta \to 0^+} \sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) \cdot \frac{h(\theta, y) - h(0, y)}{\theta} &= \sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) \lim_{\theta \to 0^+} \frac{h(\theta, y) - h(0, y)}{\theta} \\
&= \sum_{y \in \mathsf{supp}(Y_0)} Y_0(y) \left(1 - \frac{Y_1(y)}{Y_0(y)}\right) \\
&= 0,
\end{aligned}$$

which yields (B.11). The desired result follows by combining (B.8), (B.9), and (B.11). □

We are now ready to prove the optimality conditions of Theorem 4.1.

Theorem B.1 (Optimality conditions of Theorem 4.1)**.** *Fix $\lambda \geq 0$ and $\mu > 0$. An input X is capacity-achieving for the channel $\mathsf{DTP}_{\lambda,\mu}$ if and only if $\mathbb{E}[X] = \mu$ and there exist constants $a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$ such that*

$$D_{\mathsf{KL}}^{(e)}(Y_x \| Y_X) \leq ax + b$$

for every $x \in \mathbb{R}_0^+$, with equality when $x \in \mathsf{supp}(X)$, where Y_X denotes the output of $\mathsf{DTP}_{\lambda,\mu}$ on input X and $Y_x \sim \mathsf{Poi}_{\lambda+x}$. In this case, we have $C(\lambda, \mu) = a\mu + b$.

Proof. First, suppose that $X^\star$ with cdf $F^\star \in \Omega_\mu$ is capacity-achieving (by the discussion at the end of Appendix B.2, we know that $\mathbb{E}[X^\star] = \mu$). Instantiate Lemma B.7 with $\Omega = \Omega_{\mathsf{fin}}$, $J(F) = -I(F)$, and $G(F) = \mathbb{E}[X_F] - \mu$. Note that $I(F), G(F) \in \mathbb{R}$ for every $F \in \Omega_{\mathsf{fin}}$ and that there exists $F \in \Omega_{\mathsf{fin}}$ with $G(F) < 0$ whenever $\mu > 0$. Moreover, Ω_{fin} is a convex subset of a vector space over $\mathbb{R}$ (the set of all functions $f : \mathbb{R} \to \mathbb{R}$ with pointwise addition and scalar multiplication), and both $-I$ and G are convex on Ω_{fin}. Therefore, there exists $z \geq 0$ such that $F^\star$ minimises $-I(\cdot) + zG(\cdot)$ over Ω_{fin}, or, equivalently, maximises the functional

$$\alpha(\cdot) = I(\cdot) - zG(\cdot)$$

over Ω_{fin}, and $zG(F^\star) = 0$. As a result, according to Lemma B.6 we must have

$$\alpha'_{F^\star}(Q) = I'_{F^\star}(Q) - zG'_{F^\star}(Q) \leq 0 \tag{B.14}$$

for all $Q \in \Omega_{\mathsf{fin}}$, provided $I'_{F^\star}(Q)$ and $G'_{F^\star}(Q)$ exist. For $\overline{x} \geq 0$, define the unit step function $Q_{\overline{x}} \in \Omega_{\mathsf{fin}}$ as

$$Q_{\overline{x}}(x) = \begin{cases} 0, & \text{if } x < \overline{x}, \\ 1, & \text{otherwise.} \end{cases}$$

Since $D_{\mathsf{KL}}(Y_x || Y_{F^\star})$ is finite for every $x \geq 0$ because $Y_{F^\star}$ has full support and $-\log Y_{F^\star}(y) = O(y \log y)$ and $H(Y_x)$ is also finite for $x \geq 0$, Lemma B.8 implies that $I'_{F^\star}(Q_{\overline{x}})$ exists and is given by

$$I'_{F^\star}(Q_{\overline{x}}) = \int D_{\mathsf{KL}}(Y_x || Y_{F^\star}) dQ_{\overline{x}}(x) - I(F^\star)$$

$$= D_{\mathsf{KL}}(Y_{\overline{x}}||Y_{F^\star}) - I(F^\star). \tag{B.15}$$

Furthermore, since G is linear on Ω_{fin} we have

$$G'_{F^\star}(Q_{\overline{x}}) = \lim_{\theta\to 0^+} \frac{G(F_\theta) - G(F^\star)}{\theta} = G(Q_{\overline{x}}) - G(F^\star) = \overline{x} - \mathbb{E}[X^\star]. \tag{B.16}$$

Combining (B.14), (B.15), and (B.16), we must have

$$D_{\mathsf{KL}}(Y_x||Y_{F^\star}) - I(F^\star) - z(x - \mathbb{E}[X^\star]) \leq 0$$

for every $x \geq 0$. Equivalently,

$$D_{\mathsf{KL}}(Y_x||Y_{F^\star}) \leq I(F^\star) + z(x - \mu) \tag{B.17}$$

must hold for every $x \geq 0$. This inequality holds because, according to Lemma B.7, if $G(F^\star) \neq 0$ (i.e., $\mathbb{E}[X_{F^\star}] \neq \mu$), then $z = 0$ and the inequality would still be true in this case. Suppose now that there is $x \in \mathsf{supp}(F^\star)$ such that

$$D_{\mathsf{KL}}(Y_x||Y_{F^\star}) < I(F^\star) + z(x - \mu). \tag{B.18}$$

All terms in the inequality above are continuous functions of x on $[0, \infty)$. As a result, (B.18) must hold for all $x' \in \mathcal{I} \cap [0, \infty)$, where $\mathcal{I}$ is some open interval containing x. Since $x \in \mathsf{supp}(F^\star)$, by definition of $\mathsf{supp}(F^\star)$ and the fact that $\lim_{x\to 0^-} F^\star(x) = 0$ we have $\int_{\mathcal{I}\cap[0,\infty)} dF^\star(x) = \int_{\mathcal{I}} dF^\star(x) > 0$. Therefore, recalling (B.17), it holds that

$$\begin{aligned} I(F^\star) &= \int D_{\mathsf{KL}}(Y_x||Y_{F^\star}) dF^\star(x) \\ &< I(F^\star) + z\left(\int x dF^\star(x) - \mu\right) \\ &\leq I(F^\star), \end{aligned}$$

where the second inequality follows because $\mathbb{E}[X^\star] = \int x dF^\star(x) \leq \mu$ and $z \geq 0$, which leads to a contradiction. Consequently, letting $a = z \geq 0$ and $b = I(F^\star) - z\mu$, we must have

$$D_{\mathsf{KL}}(Y_x||Y_{F^\star}) \leq ax + b$$

for every $x \in \mathbb{R}_0^+$, with equality for $x \in \mathsf{supp}(F^\star)$, as desired. In particular, this implies that $C(\lambda, \mu) = I(F^\star) = a\mu + b$.

For the converse, suppose there are constants $a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$ such that $F \in \Omega_\mu^=$ satisfies

$$D_{\mathsf{KL}}(Y_x||Y_F) \leq ax + b \tag{B.19}$$

for every $x \in \mathbb{R}_0^+$, with equality for $x \in \mathsf{supp}(F)$. Since $D_{\mathsf{KL}}(Y_x||Y_F) = ax + b$ for all $x \in \mathsf{supp}(F)$, we have

$$I(F) = \int D_{\mathsf{KL}}(Y_x||Y_F)dF(x) = a\mathbb{E}[X_F] + b = a\mu + b,$$

where the last equality holds because $F \in \Omega_\mu^=$. Combining this observation with (B.19) and the first part of Theorem 4.1 with $Y = Y_F$ implies that

$$a\mu + b = I(F) \leq C(\lambda, \mu) \leq a\mu + b$$

and F is capacity-achieving. □

B.3.1 Extension to other channels

The argument used to prove Theorem B.1 above goes through for a broad class of stationary memoryless channels under mild assumptions. Consequently, such channels enjoy analogous optimality conditions under a mean constraint. To illustrate this, we discuss the case of DMCs, which yields the optimality conditions from Theorem 2.5.

Suppose the input alphabet is $\mathbb{N}$. We define the convex sets Ω_μ, $\Omega_\mu^=$, and Ω_{fin} consisting of all distributions X over $\mathbb{N}$ satisfying $\mathbb{E}[X] \leq \mu$, $\mathbb{E}[X] = \mu$, and $\mathbb{E}[X] < \infty$, respectively, and consider a DMC Ch with input alphabet $\mathbb{N}$ and output alphabet $\mathbb{N}_0$ which maps each $x \in \mathbb{N}$ to an output Y_x. We are interested in the constrained capacities

$$C(\mu) = \sup_{X \in \Omega_\mu} I(X; Y_X) \quad \text{and} \quad C^=(\mu) = \sup_{X \in \Omega_\mu^=} I(X; Y_X),$$

where Y_X denotes the output of Ch on input X, which are concave, non-negative functions of $\mu \geq 1$. The lemma below states that these two quantities coincide always.

Lemma B.9. *We have $C(\mu) = C^=(\mu)$ for every $\mu \geq 1$. Moreover, if $C(\cdot)$ is unbounded and $X^\star \in \Omega_\mu$*

satisfies $I(X^\star; Y_{X^\star}) = C(\mu)$*, then* $\mathbb{E}[X^\star] = \mu$.

Proof. Regarding the first statement, note that it is trivial for $\mu = 1$. Therefore, suppose that $C(\mu) > C^{=}(\mu)$ for some $\mu > 1$. Then, there is X satisfying $\mathbb{E}[X] = \mu' < \mu$ such that $C^{=}(\mu') \geq I(X; Y_X) > C^{=}(\mu)$. As a result, by concavity of $C^{=}$, the quantity $C^{=}(\mu'')$ must lie below the negative-slope line defined by $(\mu', C^{=}(\mu'))$ and $(\mu, C^{=}(\mu))$ for $\mu'' > \mu$, and so $C^{=}(\mu'') < 0$ for μ'' large enough, contradicting the fact that $C^{=}(\cdot)$ is non-negative.

To see the second statement, suppose that $I(X^\star; Y_{X^\star}) = C(\mu)$ and $\mathbb{E}[X^\star] = \mu' < \mu$. Then, we have $C(\mu') = C(\mu)$. Since C is concave and non-decreasing, it must hold that $C(\mu'') = C(\mu)$ for all $\mu'' > \mu$, contradicting the fact that $C(\cdot)$ is unbounded. □

The argument from Appendix B.3 (with only one minor modification which we discuss below) yields analogous optimality conditions for a broad class of DMCs.

Theorem B.2. *Fix* $\mu > 1$ *and suppose that the DMC* Ch *satisfies* $H(Y_X) < \infty$ *for every* $X \in \Omega_{\mathsf{fin}}$*. If an input distribution* $X \in \Omega_\mu$ *satisfies* $I(X; Y_X) = C(\mu)$*, there exist constants* $a \in \mathbb{R}_0^+$, $b \in \mathbb{R}$ *such that*

$$D_{\mathsf{KL}}(Y_x \| Y_X) \leq ax + b \tag{B.20}$$

for every $x \in \mathbb{N}$ *with equality when* $x \in \mathsf{supp}(X)$*. Moreover, if* $X \in \Omega_\mu^{=}$ *and* (B.20) *holds for all* $x \in \mathbb{N}$ *with equality when* $x \in \mathsf{supp}(X)$*, then* $I(X; Y_X) = a\mu + b = C(\mu)$*.*

Proof. The argument proceeds exactly like in the proof of Theorem B.1, except that we employ a more general approach to show that $D_{\mathsf{KL}}(Y_x \| Y_{X^\star}) < \infty$ for all $x \in \mathbb{N}$ if $X^\star \in \Omega_\mu$ is capacity-achieving. Since Ω_μ is convex, it holds that (see [178, Theorem 4.4]) for every $Z \in \Omega_\mu$ we have

$$\sum_{x \in \mathsf{supp}(Z)} Z(x) D_{\mathsf{KL}}(Y_x \| Y_{X^\star}) \leq I(X^\star; Y_{X^\star}) = C(\mu) < \infty. \tag{B.21}$$

If we consider the singleton distribution Z_x with support $\{x\}$ for integer $x \in [1, \mu]$, it follows that $D_{\mathsf{KL}}(Y_x \| Y_{X^\star}) < \infty$ for all such x. For integer $x > \mu$ consider $W_x = \lambda Z_1 + (1 - \lambda) Z_x$ with $\lambda = \frac{x - \mu}{x - 1} \in (0, 1)$ so that $\mathbb{E}[W_x] = \lambda + (1 - \lambda)x = \mu$. From (B.21) with $Z = W_x$, we conclude that

$$\lambda D_{\mathsf{KL}}(Y_1 \| Y_{X^\star}) + (1 - \lambda) D_{\mathsf{KL}}(Y_x \| Y_{X^\star}) < \infty,$$

and so $D_{\mathsf{KL}}(Y_x \| Y_{X^\star}) < \infty$. □

It remains to see that Theorem B.2 yields the optimality conditions from Theorem 2.5 (the capacity upper bounds of that theorem can also be obtained from Lemma 4.1). Fix a replication rule R with finite expected value $\lambda > 0$ and let Ch_R be the associated DMC with input alphabet $\mathbb{N}$ and output alphabet $\mathbb{N}_0$. First, note that $\mathbb{E}[Y_X] = \lambda\mathbb{E}[X]$ for any input distribution X. Therefore, imposing an output mean constraint $\mathbb{E}[Y_X] = \mu > \lambda$ is equivalent to imposing an input mean constraint $\mathbb{E}[X] = \mu/\lambda > 1$, and so $\mathsf{Cap}_\mu(\mathsf{Ch}_R) = C^{=}(\mu/\lambda)$. Moreover, we have $\mathbb{E}[Y_X] < \infty$ when $X \in \Omega_{\mathsf{fin}}$, and so $H(Y_X) < \infty$. Finally, if $X \in \Omega^{=}_{\mu/\lambda}$ achieves $C^{=}(\mu/\lambda)$, then Lemma B.9 ensures that X also achieves $C(\mu/\lambda)$. Combining these properties with Theorem B.2 leads to the optimality conditions of Theorem 2.5.