UNIVERSITY OF CALIFORNIA, SAN DIEGO

Code Representation and Performance of Graph-Based Decoding

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

in

Electrical Engineering
(Communications Theory and Systems)

by

Junsheng Han

Committee in charge:

Professor Paul H. Siegel, Chair
Professor Laurence B. Milstein
Professor Alon Orlitsky
Professor Lance W. Small
Professor Jack K. Wolf

2008

The dissertation of Junsheng Han is approved, and it is acceptable in quality and form for publication on microfilm:

Chair

University of California, San Diego

2008

To my parents.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I am fortunate to have benefited from the help of many people during my time at UCSD.

First and foremost, I owe my deepest gratitude to my advisor, Professor Paul Siegel, who has been my mentor not only in research, but for many other things in life. When I had doubts about myself, he was always there to encourage me; when I had hesitation about my career, he was always available for a friendly chat. His kindness, wisdom and patience are more than I could ever ask for. I also wish to thank Professor Siegel's wonderful family, Darcy, Oren and Micah, for their hospitality and the many unforgettable Thanksgiving parties.

I am grateful to other members of my committee for their time and supportiveness. Professor Jack Wolf gave much valuable feedback on my research and has always been a delight to talk to. I only hope his insights are as infectious as his enthusiasm. I am thankful to Professor Alon Orlitsky and Professor Larry Milstein for their many memorable classes that among other things changed my view of the world: Professor Orlitsky showed that nerdy people can be fun, and Professor Milstein exemplifies that engineering can be rigorous, both key to keeping me from dropping out of grad school. Professor Lance Small has been very accommodating and showed much interest in my work, which I appreciate very much.

My special thanks go to my collaborators: Luis Lastras, Patrick Lee, Ron Roth and Alexander Vardy, from whom I have learned a lot.

I thank current and past STARs and affiliates, for providing a stimulating research and learning environment and a delightful home away from home. I am especially grateful to Sharon Aviran, Brian Butler, Panu Chaichanavong, Jiangxin Chen, Mike Cheng, Ismail Demirkan, Amir Hadi Djahanshadi, Federica Garin, Jilei Hou, Toshio Ito, Seyhan Karakulak, Navin Kashyap, Brian Kurkoski, Zsigmond Nagy, Henry Pfister, Kevin Quirk, Erik Rosnes, Moshe Schwartz, Ori Shental, Joseph Soriaga, Mohammad Hossein Taghavi, Zeinab Taghavi, Hao Wang, Zheng Wu, Eitan Yaakobi,Yan Zhang, and Li Zhu.

I am indebted to the CMRR staff, particularly Ray Descoteaux, Betty Manoulian and Iris Villanueva for making CMRR such a wonderful workspace. I thank the ECE staff, especially M'Lissa Michelson, Gennie Miranda, Karol Previte and Bernadette Villaluz for their excellent administrative support.

I wish to thank Sheraz Tariq, my former manager at Ericsson, for believing in my

potential and encouraging me to take on the pursuit. I also thank Dr. Olgica Milenkovic for her encouragement and interest in my work, and for her many helpful comments and suggestions.

I would like to thank my friends in San Diego, especially Patrick Amihood, Jiucang Hao, Dan Liu, Sheu-Sheu Tan, Jianfeng Xu, Hao Zheng, and Yuan Zhi, who have been very supportive during my thesis writing.

Last and most importantly, I thank my parents, Han Haizhen and Han Shuye, for their love and sacrifice that made all I did possible. I would have been nothing without them in every conceivable way, and to them I dedicate this dissertation.

Funding for this research was provided by a CalIT^2 Fellowship, and by the Center for Magnetic Recording Research (CMRR) at UCSD.

Chapter 3 is in part a reprint of the material in the papers: J. Han and P. H. Siegel, “Improved upper bounds on stopping redundancy,” *IEEE Trans. Inform. Theory*, vol. 53, no. 1, pp. 90–104, Jan. 2007, and J. Han, P. H. Siegel, and A. Vardy, “Improved probabilistic bounds on stopping redundancy,” *IEEE Trans. Inform. Theory*, vol. 54, no. 4, pp. 1749–1753, Apr. 2008. Chapter 4 is in part a reprint of the material in the papers: J. Han and P. H. Siegel, “On the stopping redundancy of MDS codes,” in *Proc. IEEE Int. Symp. Inform. Theory*, Seattle, WA, July 2006, pp. 2491–2495, J. Han and P. H. Siegel, “Improved upper bounds on stopping redundancy,” *IEEE Trans. Inform. Theory*, vol. 53, no. 1, pp. 90–104, Jan. 2007, J. Han, P. H. Siegel, and R. M. Roth, “Bounds on single-exclusion numbers and stopping redundancy of MDS codes,” in *Proc. IEEE Int. Symp. Inform. Theory*, Nice, France, June 2007, pp. 2941–2945, and J. Han, P. H. Siegel, and R. M. Roth, “Single-exclusion number and the stopping redundancy of MDS codes,” *IEEE Trans. Inform. Theory*, submitted for publication. Chapter 6 is in part a reprint of the material in the paper: J. Han and P. H. Siegel, “On ML redundancy of codes,” in *Proc. IEEE Int. Symp. Inform. Theory*, Toronto, Canada, July 2008, to appear. Chapter 7 is a reprint of the material in the paper: J. Han, P. H. Siegel and Patrick Lee, “On the probability of undetected error for over-extended Reed-Solomon codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3662–3669, Aug. 2006. The dissertation author was the primary author of all these papers.

VITA

1998	B.E., Automation, Tsinghua University, Beijing, China.
2001	M.S., Electrical Engineering, The Ohio State University, Columbus, OH.
2008	Ph.D., Electrical Engineering (Communication Theory and Systems), University of California, San Diego.

PUBLICATIONS

J. Han, P. H. Siegel, and R. M. Roth, "Single-exclusion number and the stopping redundancy of MDS codes," submitted to *IEEE Trans. Inform. Theory*, Dec. 2007.

J. Han and P. H. Siegel, "On ML redundancy of codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Toronto, Canada, July 2008, to appear.

J. Han, P. H. Siegel, and A. Vardy, "Improved probabilistic bounds on stopping redundancy," *IEEE Trans. Inform. Theory*, vol. 54, no. 4, pp. 1749–1753, Apr. 2008.

J. Han, P. H. Siegel, and R. M. Roth, "Bounds on single-exclusion numbers and stopping redundancy of MDS codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Nice, France, June 2007, pp. 2941–2945.

J. Han and L. A. Lastras-Montaño, "Reliable memories with subline accesses," in *Proc. IEEE Int. Symp. Inform. Theory*, Nice, France, June 2007, pp. 2531–2535.

J. Han and P. H. Siegel, "Improved upper bounds on stopping redundancy," *IEEE Trans. Inform. Theory*, vol. 53, no. 1, pp. 90–104, Jan. 2007.

J. Han, P. H. Siegel, and P. Lee, "On the probability of undetected error for over-extended Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3662–3669, Aug. 2006.

J. Han and P. H. Siegel, "On the stopping redundancy of MDS codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Seattle, WA, July 2006, pp. 2491–2495.

J. Han, P. H. Siegel, and P. Lee, "On the probability of undetected error for over-extended Reed-Solomon codes," in *Proc. IEEE Information Theory Workshop*, Punta del Este, Uruguay, Mar. 2006, pp. 150–154.

J. Han and P. H. Siegel, "Reducing acyclic network coding problems to single-transmitter-single-demand form," in *Proc. 42nd Allerton Conference on Communication, Control, and Computing*, Monticello, IL, Oct. 2004, pp. 316–325.

J. Han and O. Y. Takeshita, "On the decoding structure of multiple Turbo codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Washington, DC, June 2001, p. 98.

L. Gao, J. Han, C. Li, and F. Li, "Nonlinear robust auto-disturbance rejection controller for power systems," *Journal of Tsinghua (Sci. and Tech.)*, vol. 40, no. 3, pp. 59–63, Mar. 2000.

ABSTRACT OF THE DISSERTATION

Code Representation and Performance of Graph-Based Decoding

by

Junsheng Han
Doctor of Philosophy in Electrical Engineering
(Communications Theory and Systems)

University of California San Diego, 2008

Professor Paul H. Siegel, Chair

Key to the success of modern error correcting codes is the effectiveness of message-passing iterative decoding (MPID). Unlike maximum-likelihood (ML) decoding, the performance of MPID depends not only on the code, but on how the code is *represented*. In particular, the performance of MPID is potentially improved by using a *redundant* representation. We focus on Tanner graphs and study combinatorial structures therein that help explain the performance disparity among different representations of the same code. Emphasis is placed on the complexity-performance tradeoff, as more and more check nodes are allowed in the graph. Our discussion applies to MPID as well as linear programming decoding (LPD), which we collectively refer to as graph-based decoding.

On an erasure channel, it is well-known that the performance of MPID or LPD is determined by *stopping sets*. Following Schwartz and Vardy, we define the *stopping redundancy* as the smallest number of check nodes in a Tanner graph such that smallest size of a non-empty stopping set is equal to the minimum Hamming distance of the code. Roughly speaking, stopping redundancy measures the complexity requirement (in number of check nodes) for MPID of a redundant graph representation to achieve performance comparable to ML decoding (up to a constant factor for small channel erasure probability).

General upper bounds on stopping redundancy are obtained. One of our main contribution is a new upper bound based on probabilistic analysis, which is shown to be by far the strongest. From this bound, it can be shown, for example, that for a fixed minimum distance, the stopping redundancy grows just linearly with the redundancy (codimension). Specific results on the stopping redundancy of Golay and Reed-Muller codes are also obtained.

We show that the stopping redundancy of maximum distance separable (MDS) codes is bounded in between a Turán number and a single-exclusion (SE) number — a purely combinatorial quantity that we introduce. By studying upper bounds on the SE number, new results on the stopping redundancy of MDS codes are obtained. Schwartz and Vardy conjecture that the stopping redundancy of an MDS code should only depend on its length and minimum distance. Our results provide partial confirmation, both exact and asymptotic, to this conjecture.

Stopping redundancy can be large for some codes. We observe that significantly fewer checks are needed if a small number of small stopping sets are allowed. These small stopping sets can then be dealt with by "guessing" during the iterative decoding process. Correspondingly, the *guess-g stopping redundancy* is defined and it is shown that the savings in number of required check nodes are potentially significant. Another theoretically interesting question is when MPID of a Tanner graph achieves the same word error rate an ML decoder. This prompts us to define and study *ML redundancy*.

Applicability and possible extensions of the current work to a non-erasure channel are discussed. A framework based on pseudo-codewords is considered and shown to be relevant. However, it is also observed that the polytope characterization of pseudo-codewords is not complete enough to be an accurate indicator of MPID performance.

Finally, in a separate piece of work, the probability of undetected error (PUE) for over-extended Reed-Solomon codes is studied through the weight distribution bounds of the code. The resulting PUE expressions are shown to be tight in a well-defined sense.

Chapter 1

Introduction

1.1 Background

Transmission and storage of digital data has become an integral part of our daily lives. Whether it is watching a DVD movie, or making a phone call from a cellular handset, messages in digital form are effectively transmitted through a *channel* to be reconstructed at the receiver's end. Unfortunately, the channel is often inherently less reliable than we hope it is. For example, the read-back signal from a DVD may contain bursts of errors caused by scratches on the disc surface. The wireless communication channel, on the other hand, often exhibits shadowing, fading, and multicell and multiuser interference.

Transmitting information reliably over an unreliable channel has been a central topic of information theory. In his landmark paper [1], Shannon showed that, contrary to common beliefs, information can be transmitted *arbitrarily reliably* at any rate (bits per channel use) lower than a number he called the *channel capacity*, which is determined by the statistics of the channel, and conversely, reliable transmission is not possible at rates higher than the channel capacity. Shannon's key idea to reliable communication is to use *coding* to add *redundancy* to the messages to be transmitted, such that if this is done over many uses of the channel, the law of large numbers sets in and ensures that errors introduced by the channel can be corrected with high probability at the receiver's end. Figure 1.1 shows this (much) simplified view of a communication system. Here, $\boldsymbol{u} \in \mathcal{U}^k$ is the message to be sent. The encoder maps $\boldsymbol{u}$ to $\boldsymbol{x} \in \mathcal{C} \subset \mathcal{X}^n$, which is transmitted through the channel and received as $\boldsymbol{y} \in \mathcal{Y}^n$, and the decoder is a deterministic mapping from all possible received sequences to $\mathcal{U}^k$ in order to reconstruct

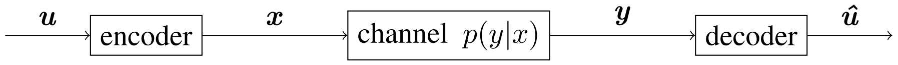

Figure 1.1 Simplified view of a communication system over a discrete memoryless channel

the original message. The channel is a stochastic device defined by the conditional probability distribution $p(y|x)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$, for each channel use. We assume that the channel is *memoryless* in the sense that conditioned on the current channel input, the current channel output is statistically independent of all past and future channel inputs. An error occurs in the system when $\hat{\boldsymbol{u}} \neq \boldsymbol{u}$. The set of all possible transmitted sequences, $\mathcal{C}$, is called a *codebook*, or simply, a *code*.

Shannon proved that good codes exist by showing that a randomly chosen code is good *on average* (in fact also with high probability). However, what he didn't show was how to find and to encode and decode good codes *efficiently*. Much of the research in coding theory has been devoted to solving the above problems.

Elias [2] showed that the capacity of a discrete memoryless channel (DMC) can be approached by using *linear codes*, which are linear vector (sub)spaces. This essentially solves the encoding problem, since linear codes can be encoded by multiplying the input vector by a *generator matrix*. What remain to be solved are efficient code construction and efficient decoding. These two requirements, however, for a long time seemed to be at odds with each other. To quote popular folk wisdom in the coding theory community, "*all codes are good, except those that we know of.* [3]" One interpretation of this difficulty is that the algebraic structure that affords simplified decoding algorithms for many linear code constructions at the same time makes the codes less random-like. To make things worse, it was recognized [4] early on that maximum likelihood decoding for the general class of linear codes is NP-hard. In fact, most of the algebraically constructed linear codes are decoded using a suboptimal *bounded distance decoder*, which only corrects up to a certain number of errors in each codeword.

Significant breakthrough was made in the 1990s with the discovery of Turbo codes [5], and later with the rediscovery of low-density parity-check (LDPC) codes [6] [7], or more generally, codes on graphs [8] [9]. What these coding schemes have in common is that they are overall random-like, but at the same time have well structured, very simple *partial descriptions*. For Turbo codes, randomness is introduced by pseudo-random interleaver(s), and a partial description is that of a constituent code, usually a recursive systematic convolutional code with a small constraint length. For LDPC codes, randomness is built in by pseudo-random construction of parity checks, and a partial description of the code is (usually) simply that of a single parity-check code corresponding to each (low-weight) parity check. In both cases, randomness ensures that the code construction is likely to be good for a reasonably long block length, while the simple partial descriptions (whose optimal decoding is low-complexity) allow for a computationally efficient, *iterative* decoding algorithm that works well in practice. The two long-standing requirements of efficient code construction and efficient decoding are finally at peace with each other.

In a sense, the code construction problem was an easy one — just pick a code at random. It is arguably the decoding problem that is more significant. Indeed, the decoding principles that underlie the success of Turbo and LDPC codes are not so much tied to their particular constructions, and have found application to many other areas. For example, the so-called "Turbo principle" has been applied to equalization [10], and to multiuser detection [11], and similar algorithms for decoding LDPC codes have also been proposed to decode other algebraically constructed linear codes [12].

Essentially, what happens in the decoding of Turbo or LDPC codes is that "constituent decoders" that have knowledge of only a partial description of the code exchange beliefs about which symbols were transmitted at which position within the codeword in an iterative fashion. For most codes of interest, there is no guarantee that such a process will converge, and when it does, that it converges to the most probable codeword (or symbols). However, in practice these algorithms work extremely well with properly designed codes and their chosen partial descriptions. For the moment, let's refer to such an algorithm as *message-passing iterative decoding (MPID)*.

One immediately notices that the performance of MPID depends not only on the code, but also on the partial descriptions that we choose to *represent* the code. In the case of Turbo codes, the choice of partial descriptions is clear. In other cases, especially the case of parity-check codes (i.e. linear codes), there is much flexibility in our choice. This flexibility comes from the fact that a linear code may be defined by many different sets of parity check equations. Interestingly, the choice of representation — using different sets of parity check equations — can indeed have a significant impact on the performance of MPID. As an example, Figure 1.2 shows

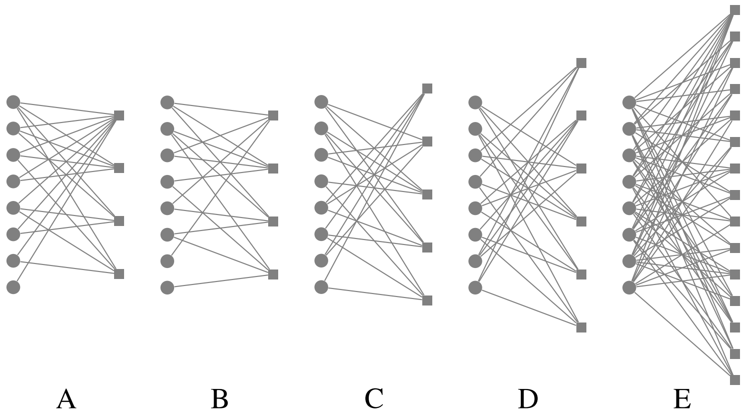

Figure 1.2 Five representations of the $(1, 3)$–RM code

five different Tanner graphs [8] all representing the same $(1, 3)$–Reed Muller (RM) code. Each graph corresponds to a different set of parity check equations. Figure 1.3 shows the performance of sum-product decoding (a specific form of MPID) using these different representations on an additive white Gaussian noise (AWGN) channel. It can clearly be seen that the performance differential between different code representations can be significant.

A particularly interesting aspect that can be seen when comparing performances of different code representations is that by allowing the use of *redundant* parity checks (thus using more than the minimal number of partial descriptions for the code), the performance of MPID can potentially be improved. As seen in Figure 1.3, going from 4 to 5 parity checks yields a significant performance improvement, and another notable improvement is obtained going from 5 to 6 checks, etc. Clearly, a complexity-performance tradeoff is at play here — by using more partial descriptions (parity checks), better performance may be achieved, but at the cost of higher decoding complexity. A large part of this dissertation has been motivated by this observation, and can be viewed as an attempt at characterizing the complexity-performance tradeoff in *redundant code representation*, especially on the minimal number of partial descriptions to use in order to achieve a certain performance goal.

One should be cautioned as to not be given the impression that the addition of parity checks always improves the performance of MPID. Although this is true for some special cases (for example, on an erasure channel), there is no simple guarantee in general. In fact, it is easy to

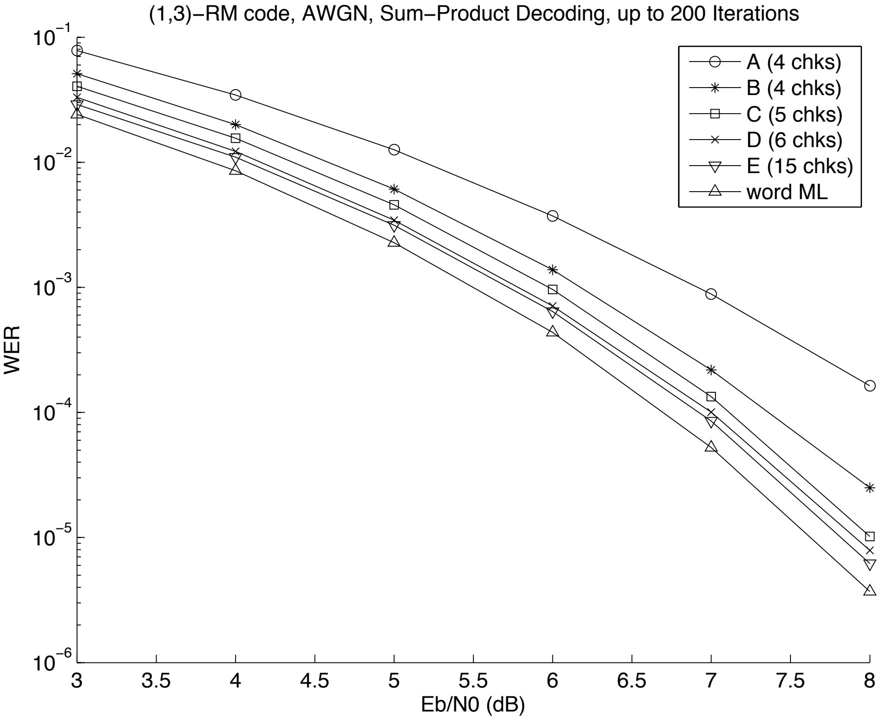

Figure 1.3 Performance of different Tanner graphs for the $(1,3)$–RM code under sum-product decoding

find examples to the contrary where the performance of MPID on an AWGN channel degrades as a result of additional parity checks [13]. Nevertheless, graph-cover decoding (GCD) [14] [15] has been proposed as an analysis tool to understand MPID, and the performance of GCD can only improve with the addition of parity checks. Same can be said for linear programming decoding (LPD) of binary linear codes [16], which is equivalent to GCD [15]. The fact that LPD/GCD and MPID are closely related [16] [15] [17] [13] gives justification to the potential performance benefit of redundant parity checks for MPID.

A precise understanding of how MPID is affected by code representation is a difficult if not near impossible task. We thus make two simplifications to the problem. First, we focus on erasure channels, where it is well-known [18] that the performance of MPID (as well as GCD/LPD) is determined by the distribution of *stopping sets* [19] [18]. Note that although the erasure channel is unique in many regards, it has proved a fruitful starting point to gain the most fundamental insights into how MPID works, especially in the case of LDPC codes. Comments on how relevant our study is to other channels and how the framework may be extended are made in 6.3. Secondly, we will pay special attention to a minimum-distance-equivalent metric for MPID performance, which in the case of an erasure channel is the *stopping distance* [20] to be defined later. Just as minimum distance is to maximum likelihood (ML) decoding, stopping distance is the appropriate figure of merit for the "high SNR" regime, i.e. in the case of the erasure channel, when the channel erasure probability is small. From a practical point of view, this attention on the "high SNR" asymptotic performance is justifiable since it addresses the "error-floor" problem [21], an important challenge that faces many modern error correcting codes for critical applications such as data storage, where error rates on the order of 10^{-15} are often required.

1.2 Dissertation Overview

After a brief review of some preliminaries, in Chapter 2 the concepts of stopping distance and stopping redundancy are motivated and defined.

Chapter 3 is devoted to general upper bounds on stopping redundancy. These bounds are general as they apply to *all* linear block codes. A bound based on the probabilistic method will be presented, together with a few explicitly constructive bounds. The probabilistic bound is further improved by a more elaborate probabilistic analysis, and is compared to various other bounds in the literature.

In Chapter 4 we study the stopping redundancy of maximum distance separable (MDS) codes. It is quickly realized that this problem is highly combinatorial, and in fact we will define and show that the *single-exclusion (SE) number* is naturally an upper bound on the stopping redundancy. Through elaborate combinatorial analysis, we obtain a number of bounds on the SE number, which in turn lead to improved upper bounds on the stopping redundancy of MDS codes. By essentially a sandwiching argument, we show that the conjecture by Schwartz and Vardy [22] that the stopping redundancy of MDS codes only depends on length and minimum distance is true for certain limited cases either exactly or in an asymptotic sense.

In Chapter 5 some results on the stopping redundancy of Reed-Muller (RM) codes [20] [23] are rediscovered through a geometric approach. The new perspective is more intuitive, and lends itself to extension to other finite geometry codes.

Chapter 6 is devoted to extending the work on stopping redundancy in a number of ways. First, we define *guess-g stopping redundancy* based on ideas in [24] [25], and show that "guessing" can significantly reduce the number of checks required in a Tanner graph to achieve the same performance target as in the case of stopping redundancy. Next, we ask the question how many check nodes are needed in a Tanner graph for MPID to have the same performance as ML decoding, and study the *ML redundancy* of codes. Finally, we comment on how the study on stopping redundancy may be extended to a non-erasure channel. A framework of study based on pseudo-codewords [14] [16] is described. The results for the erasure channel thus have relevance to the non-erasure case, since stopping sets and pseudo-codewords are closely related. On the other hand, it is shown that the set of pseudo-codewords as we define them, though sufficient to derive LPD performance, is not complete enough for an accurate characterization of MPID performance (as was noted in [13]).

Chapter 7 is a self-contained study on the probability of undetected error for over-extended Reed-Solomon (OERS) codes. Bounds on the probability of undetected error are obtained through bounds on the weight distribution of OERS codes, and are shown to be tight. This chapter is rather independent from all other chapters, and is included here for completeness.

Bibliography

[1] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, July, Oct., 1948.

[2] P. Elias, “Coding over noisy channels,” in *IRE Convention Record*, 1955, pp. 37–46.

[3] J. M. Wozencraft and B. Reiffen, *Sequential Decoding*. Cambridge, MA: MIT Press, 1961.

[4] E. R. Berlekamp, R. J. McEliece, and H. C. van Tilborg, “On the inherent intractability of certain coding problems,” *IEEE Trans. Inform. Theory*, vol. 24, no. 3, pp. 384–386, May 1978.

[5] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error correcting coding and decoding,” in *Proc. IEEE International Conference on Communications*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[6] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge: M.I.T. Press, 1963.

[7] D. J. C. MacKay and R. M. Neal, “Near Shannon limit performance of low density parity check codes,” *Electronics Letters*, vol. 32, p. 1645, Aug. 1996.

[8] R. M. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. Inform. Theory*, vol. IT-27, no. 5, pp. 533–547, Sept. 1981.

[9] N. Wiberg, “Codes and decoding on general graphs,” Ph. D. dissertation, Linköping University, 1996.

[10] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, “Iterative correction of intersymbol interference: Turbo equalization,” *European Trans. Telecomm.*, vol. 6, pp. 507–511, Sept.–Oct. 1995.

[11] X. Wang and H. V. Poor, “Iterative (Turbo) soft interference cancellation and decoding for coded CDMA,” *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, July 1999.

[12] R. Lucas, M. Bossert, and M. Breitbach, “On iterative soft-decision decoding of linear binary block codes and product codes,” *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 276–296, Feb. 1998.

[13] C. Kelley and D. Sridhara, “Pseudocodewords of Tanner graphs,” *IEEE Trans. Inform. Theory*, vol. 53, no. 11, pp. 4013–4038, Nov. 2007.

[14] R. Koetter and P. Vontobel, “Graph covers and iterative decoding of finite-length codes,” in *Proc. 3rd International Symposium on Turbo Codes and Related Topics*, Brest, France, Sept. 2003, pp. 75–82.

[15] P. O. Vontobel and R. Koetter, “Graph-cover decoding and finite-length analysis of message-passing iterative decoding of LDPC codes,” submitted to *IEEE Trans. Inform. Theory*, 2005. [Online]. Available: http://arxiv.org/abs/cs.IT/0512078

[16] J. Feldman, M. J. Wainwright, and D. R. Karger, “Using linear programming to decode binary linear codes,” *IEEE Trans. Inform. Theory*, vol. 51, no. 3, pp. 954–972, Mar. 2005.

[17] P. O.Vontobel and R. Koetter, “On the relationship between linear programming decoding and min-sum algorithm decoding,” in *Proc. International Symposium on Information Theory and Applications*, Parma, Italy, Oct. 2004, pp. 991–996.

[18] C. Di, D. Proletti, I. Telatar, T. Richardson, and R. Urbanke, “Finite length analysis of low-density parity-check codes,” *IEEE Trans. Inform. Theory*, vol. 48, no. 6, pp. 1570–1579, June 2002.

[19] T. Richardson and R. Urbanke, “Efficient encoding of low-density parity-check codes,” *IEEE Trans. Inform. Theory*, vol. 47, pp. 638–656, 2001.

[20] M. Schwartz and A. Vardy, “On the stopping distance and stopping redundancy of codes,” in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 975–979.

[21] T. Richardson, “Error-floors of LDPC codes,” in *Proc. 41st Annual Conference on Communication, Control and Computing*, Monticello, IL, Sept. 2003, pp. 1426–1435.

[22] M. Schwartz and A. Vardy, “On the stopping distance and the stopping redundancy of codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.

[23] T. Etzion, “On the stopping redundancy of Reed-Muller codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 4867–4879, Nov. 2006.

[24] H. Pishro-Nik and F. Fekri, “On decoding of low-density parity-check codes over the binary erasure channel,” *IEEE Trans. Inform. Theory*, vol. 50, no. 3, pp. 439–454, Mar. 2004.

[25] A. Shokrollahi, S. Lassen, and R. M. Karp, “Systems and processes for decoding chain reaction codes through inactivation,” U.S. Patent 6 856 263, Feb. 15, 2005.

Chapter 2

Stopping Redundancy

2.1 Preliminaries

For this thesis, an $[n,k]_q$ *code* $\mathcal{C}$ is a k-dimensional linear subspace of $\mathbb{F}_q^n$, where $\mathbb{F}_q$ is a field of size q. $\mathcal{C}$ is said to have *block length* (or *length*) n, *dimension* k, and *redundancy* (or *codimension*) $r = n - k$. The *minimum distance* of $\mathcal{C}$ is

$$\mathrm{d}(\mathcal{C}) \overset{\text{def}}{=} \min\bigl\{\mathrm{d_H}(\boldsymbol{x},\boldsymbol{y}) : \boldsymbol{x},\boldsymbol{y} \in \mathcal{C}, \boldsymbol{x} \neq \boldsymbol{y}\bigr\},$$

where

$$\mathrm{d_H}(\boldsymbol{x},\boldsymbol{y}) \overset{\text{def}}{=} \bigl|\{i : x_i \neq y_i\}\bigr|$$

is the *Hamming distance* between $\boldsymbol{x}$ and $\boldsymbol{y}$. An $[n,k]_q$ code with minimum distance d is also referred to as an $[n,k,d]_q$ code.

Let $\boldsymbol{x}$ be a vector. The *support* of $\boldsymbol{x}$ is the set of coordinates[1] where the components of $\boldsymbol{x}$ are nonzero, and is denoted by

$$\mathrm{supp}(\boldsymbol{x}) \overset{\text{def}}{=} \{i : x_i \neq 0\}.$$

The *weight* of $\boldsymbol{x}$ is

$$\mathrm{wt}(\boldsymbol{x}) \overset{\text{def}}{=} |\,\mathrm{supp}(\boldsymbol{x})|.$$

Note that due to linearity, we have

$$\mathrm{d}(\mathcal{C}) = \min\bigl\{\mathrm{d_H}(\boldsymbol{0},\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c} \neq \boldsymbol{0}\bigr\} = \min\bigl\{\mathrm{wt}(\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c} \neq \boldsymbol{0}\bigr\}.$$

[1] We use the convention that the coordinates are indexed starting from one.

Since a code is nothing but a vector subspace of $\mathbb{F}_q^n$, it is conveniently represented by a basis, usually arranged as row vectors in a $k \times n$ *generator matrix* G, such that

$$\mathcal{C} = \{\boldsymbol{u}G : \boldsymbol{u} \in \mathbb{F}_q^k\}.$$

Alternatively, $\mathcal{C}$ is fully described by its orthogonal complement, $\mathcal{C}^\perp$, commonly known as its *dual code*, or a basis of which, arranged as row vectors in an $r \times n$ *parity-check matrix* H, such that

$$\mathcal{C} = \{\boldsymbol{x}H^T = \boldsymbol{0} : \boldsymbol{x} \in \mathbb{F}_q^n\}.$$

We will relax the definition of a parity-check matrix such that H may contain more than r rows, as long as they form a spanning set of $\mathcal{C}^\perp$. That the choice of H, although immaterial for defining the code, can have a significant impact on the performance of MPID or LP decoding is a main motivating observation for this work.

Consider the communication system shown in Figure 1.1. To minimize the word error probability (WER)

$$\Pr\{\boldsymbol{u} \neq \hat{\boldsymbol{u}}\},$$

an optimal decoder selects $\boldsymbol{u} \in \mathbb{F}_q^k$ that maximizes the *a posteriori probability* (APP) $\Pr\{\boldsymbol{u}|\boldsymbol{y}\}$. Hence, (note that $\boldsymbol{x}$ is a function of $\boldsymbol{u}$)

$$\hat{\boldsymbol{u}} = \arg\max_{\boldsymbol{u}\in\mathbb{F}_q^k} \Pr\{\boldsymbol{u}|\boldsymbol{y}\} = \arg\max_{\boldsymbol{u}\in\mathbb{F}_q^k} \Pr\{\boldsymbol{x}|\boldsymbol{y}\}.$$

We refer to the above as the *word maximum a posteriori (MAP)* rule. Another very useful decoding criteria is the *word maximum likelihood (ML)* rule:

$$\hat{\boldsymbol{u}} = \arg\max_{\boldsymbol{u}\in\mathbb{F}_q^k} \Pr\{\boldsymbol{y}|\boldsymbol{u}\} = \arg\max_{\boldsymbol{u}\in\mathbb{F}_q^k} \Pr\{\boldsymbol{y}|\boldsymbol{x}\}.$$

ML decoding is useful when the *a priori* distribution of information words are not known, or when all information words are equally likely *a priori*, in which case word ML becomes equivalent to word MAP. Similarly, to minimize the error probability for the i-th information symbol $\Pr\{u_i \neq \hat{u}_i\}$, we obtain the *symbol MAP* decoding rule:

$$\hat{u}_i = \arg\max_{u_i\in\mathbb{F}_q} \Pr\{u_i|\boldsymbol{y}\} = \arg\max_{u_i\in\mathbb{F}_q} \sum_{\boldsymbol{x}\in\mathcal{C}_i(u_i)} \Pr\{\boldsymbol{x}|\boldsymbol{y}\},$$

where $\mathcal{C}_i(u_i)$ is the set of codewords corresponding to information words whose i-th component is u_i. When all information symbols are equally likely *a priori*, symbol MAP becomes equivalent to *symbol ML*:

$$\hat{u}_i = \arg\max_{u_i \in \mathbb{F}_q} \Pr\{\boldsymbol{y}|u_i\} = \arg\max_{u_i \in \mathbb{F}_q} \sum_{\boldsymbol{x} \in \mathcal{C}_i(u_i)} \Pr\{\boldsymbol{y}|\boldsymbol{x}\}.$$

Throughout the rest of this dissertation, “ML” without a qualifier refers to word ML, while “MAP” refers to symbol MAP.

Unfortunately, it is well known [1] that ML decoding of linear codes as a general problem is NP-hard, and remains NP-hard even with unlimited pre-processing [2]. Although this does not preclude the possibility that efficient ML decoding may exist for certain classes of codes, evidence both empirical and theoretical [3] seems to suggest the contrary.

2.2 Codes on Graphs

Finding good codes with efficient decoding algorithms has been a central issue in coding theory. The more traditional practice tries to reduce decoding complexity by adding intricate algebraic structures to the code. However, for most such constructions, ML decoding is still intractable. A classic solution is then to use a *bounded distance decoder*, which only corrects errors up to a certain number, typically half the minimum distance of the code. Probably the most famous example for such coding systems is the BCH codes with a Berlekamp-Massey type decoding algorithm [4] [5]. Although a bounded distance decoder that corrects errors up to half the minimum distance of the code is comparable in performance to an ML decoder as the channel approaches error-free, at practical channel conditions and reasonable error rate targets this solution leaves a significant gap from what is promised by the channel capacity.

A different route, pioneered by Gallager [6], is to divide and conquer. The basic idea is to factor the codeword membership constraint into a number of much simpler constraints (that certain symbols within the codeword belong to some other code), for example single-parity-check constraints, such that for each of the much simpler constraints, exact probabilistic inference within the “partial code” that it defines is computationally feasible. This breaks down the hard problem of decoding into a number of tractable problems. However, how to synthesize these simpler problems is not straightforward. Fortunately, an iterative procedure works very well in practice. In such an algorithm, each simpler constraint gives rise to a *constituent decoder* that does probabilistic inference based only on the knowledge of its own associated constraint and the inputs from the channel, and the constituent decoders exchange *extrinsic* likelihood information about the symbols in the codeword in an iterative fashion until some stopping criterion is met. Although this algorithm is suboptimal in almost all practically interesting cases, it performs surprisingly well for certain classes of codes, with notable examples including low-density parity-check (LDPC) codes [6] [7] [8], Turbo codes [9], and repeat accumulate (RA) codes [10]. We will generally refer to such an iterative decoding procedure as *message-passing iterative decoding (MPID).*

The principles of MPID are better described using a *Tanner graph* [11]. Let $\mathcal{C}$ be an $[n, k]_q$ linear code, and let $H = (h_{ij})_{l \times n}$ be a parity-check matrix for $\mathcal{C}$. As noted before, we only require that the rows of H span the dual code $\mathcal{C}^{\perp}$. Thus, l may be strictly larger than $(n-k)$. The Tanner graph $\mathcal{G}(H)$ is a bipartite graph with n *variable nodes*, each corresponding to one column of H, and l *check nodes*, each corresponding to one row of H, such that variable node j is adjacent to check node i if and only if $h_{ij} \neq 0$.

Example 2.1 The graph shown in Figure 2.1 represents an $[8, 4, 4]_2$ extended Hamming code with parity-check matrix

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

□

Corresponding to our earlier discussion, here in a Tanner graph each check node gives rise to a constituent decoder for what is basically a single-parity-check code involving symbols from its variable node neighbors. Each variable node also gives rise to a constituent decoder for a repetition code — as all its incident edges correspond to the same symbol. For either a repetition code or a single-parity-check code, exact probability inference is well formulated. In a typical decoding schedule, check node decoders will decode first, estimating the likelihoods of each symbol involved in the parity check based on all *other* symbols in the parity check, and pass this *extrinsic* information to the variable node decoders. The variable decoders then perform similarly probabilistic inference for all their incident edges and pass back their estimated

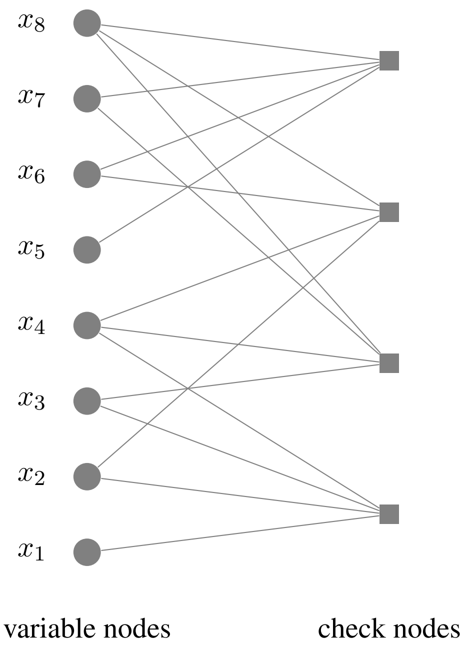

Figure 2.1 Tanner graph of an $[8, 4, 4]$ extended binary Hamming code

extrinsic likelihoods, and the process is iterated. More details of MPID, which is based on the *sum-product* algorithm or one of its variants, can be found in [12].

From the above description, it should be intuitively clear that the performance of MPID can be impacted by the Tanner graph chosen to represent a given code. In Chapter 1, we have seen that this performance impact is potentially significant. Particularly, notable performance improvements may be obtained by using a Tanner graph with redundant check nodes. And we would like to understand this complexity-performance tradeoff in code representation.

2.3 Stopping Redundancy

We will start with an erasure channel, which has proved to be a good place to get an intuitive understanding of MPID as well as insight into its performance on other channels. On an erasure channel, it is well known [13] that the performance of MPID is determined by combinatorial structures in Tanner graphs known as stopping sets. A *stopping set* in $\mathcal{G}(H)$ is a set of variable nodes such that all check nodes adjacent to the set are connected to the set at least

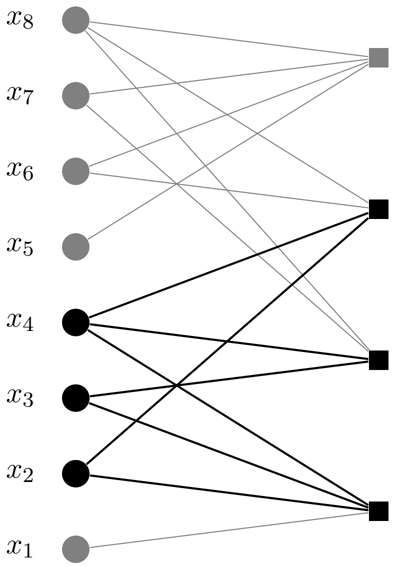

Figure 2.2 Example of a stopping set. The subgraph induced by variable node set $\{2, 3, 4\}$ is highlighted.

twice. We usually use the corresponding coordinate indices to denote the variable nodes, so a stopping set becomes a subset of $[n] \overset{\text{def}}{=} \{1, \ldots, n\}$.

Example 2.2 Figure 2.2 shows an example of a stopping set. Note that in the subgraph induced by variable nodes $\{2, 3, 4\}$, all check nodes have degree two or higher. Equivalently, observe that the submatrix of H consisting of columns 2, 3 and 4 have no rows of weight one. Therefore, $\{2, 3, 4\}$ is a stopping set.

$$H = \begin{bmatrix} 1 & \mathbf{1} & \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 \\ 0 & \mathbf{0} & \mathbf{1} & \mathbf{1} & 0 & 0 & 1 & 1 \\ 0 & \mathbf{1} & \mathbf{0} & \mathbf{1} & 0 & 1 & 0 & 1 \\ 0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 & 1 & 1 & 1 \end{bmatrix}.$$

□

It is well known [13] that iterative erasure decoding is successful if and only if the set of erasures does not contain a stopping set. The size of a smallest non-empty stopping set, known as the *stopping distance* [14] and denoted by $s(H)$, is therefore an important parameter governing the performance of the iterative decoder. Note that the stopping distance is not a property of the code $\mathcal{C}$ itself, but rather of the specific choice of a parity-check matrix H for $\mathcal{C}$. Since the support of any codeword is a stopping set, we have $s(H) \leq d$, and it is not difficult to see that equality can be achieved for any code, for example when all nonzero vectors in the dual code $\mathcal{C}^{\perp}$ are chosen as rows of H. This motivates the following definition.

Definition 2.1 Let $\mathcal{C}$ be a linear code with minimum distance d. The *stopping redundancy* of $\mathcal{C}$, denoted by $\rho(\mathcal{C})$, is the smallest integer such that there exists a parity-check matrix H for $\mathcal{C}$ with $\rho(\mathcal{C})$ rows, and $s(H) = d$.

The stopping redundancy of a linear code characterizes the minimum "complexity" (number of check nodes) required in a Tanner graph for the code, such that iterative erasure decoding achieves performance comparable to (up to a constant factor, asymptotically) maximum-likelihood (ML) decoding. It can be viewed as a basic measure of the complexity-performance tradeoff in the use of redundant parity checks (RPCs) in an iterative decoder on the erasure channel.

It is also interesting to note that there is always a "bad" choice of H, for which $s(H) \leq 3$. The following result was shown in [15] for binary codes, but is clearly also true for codes over any finite field. Here, we prove it using a slightly different argument.

Proposition 2.1 *Let $\mathcal{C}$ be an $[n, k, d]_q$ code with $d \geq 4$. There exists a parity-check matrix H for $\mathcal{C}$ such that $s(H) \leq 3$.*

Proof: Let H be a parity-check matrix for $\mathcal{C}$. If $s(H) \leq 3$, then there is nothing to prove. Otherwise, since $3 < d$, the first three columns of H are linearly independent. Therefore, by elementary row operations we can change the first row of H such that its first three components are all nonzero. Now, with this modified parity-check matrix, if $\{1, 2, 3\}$ is not a stopping set already, for each row that has a single weight in its first three components, add the first row to it. Let H' denote the parity-check matrix obtained at this point. Clearly, $s(H') \leq 3$, since $\{1, 2, 3\}$ is a stopping set. ■

The importance of $s(H)$ has been widely recognized [13], [16], [17], [18]. The relationship of $s(H)$ to the performance of iterative erasure decoding is similar to that of minimum distance to the performance of maximum-likelihood (ML) decoding. For better performance, it is desired that $s(H)$ be maximized.

Stopping redundancy was introduced by Schwartz and Vardy [14], [19], who derived general upper and lower bounds, as well as more specific bounds for Reed-Muller codes, Golay codes, and maximum distance separable (MDS) codes. The stopping redundancy of Reed-Muller and related codes was further studied by Etzion [20]. Effects of parity-check matrices on stopping set distribution were discussed by Weber and Abdel-Ghaffar [15], who found that by adding a small number of redundant parity checks, one can minimize the number of stopping sets of size 3 for a binary Hamming code. In related work, Hollmann and Tolhuizen [21], [22] consider collections of parity checks that correct all correctable erasure patterns up to a certain size for binary codes. There, emphasis was placed on (essentially) finding a *generic* r-column matrix with the least number of rows, having the property that when multiplied to *any* matrix H with r independent rows, it produces a parity-check matrix that corrects all correctable erasure patterns up to size $m \leq r$ for the code defined by the null space of H.

Bibliography

[1] E. R. Berlekamp, R. J. McEliece, and H. C. van Tilborg, “On the inherent intractability of certain coding problems,” *IEEE Trans. Inform. Theory*, vol. 24, no. 3, pp. 384–386, May 1978.

[2] J. Bruck and M. Naor, “The hardness of decoding linear codes with preprocessing,” *IEEE Trans. Inform. Theory*, vol. 36, pp. 381–385, 1990.

[3] A. Lafourcade and A. Vardy, “Asymptotically good codes have infinite trellis complexity,” *IEEE Trans. Inform. Theory*, vol. 41, pp. 555–559, 1995.

[4] E. R. Berlekamp, *Algebraic Coding Theory*. New York: McGraw-Hill, 1968.

[5] J. L. Massey, “Shift-register synthesis and BCH decoding,” *IEEE Trans. Inform. Theory*, vol. 15, pp. 122–127, 1969.

[6] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge: M.I.T. Press, 1963.

[7] D. J. C. MacKay and R. M. Neal, “Near Shannon limit performance of low density parity check codes,” *Electronics Letters*, vol. 32, p. 1645, Aug. 1996.

[8] N. Wiberg, “Codes and decoding on general graphs,” Ph. D. dissertation, Linköping University, 1996.

[9] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error correcting coding and decoding,” in *Proc. IEEE International Conference on Communications*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[10] D. Divsalar, H. Jin, and R. McEliece, “Coding theorems for turbo-like codes,” in *Proc. 36-th Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, Sept. 1998, pp. 201–210.

[11] R. M. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. Inform. Theory*, vol. IT-27, no. 5, pp. 533–547, Sept. 1981.

[12] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[13] C. Di, D. Proletti, I. Telatar, T. Richardson, and R. Urbanke, “Finite length analysis of low-density parity-check codes,” *IEEE Trans. Inform. Theory*, vol. 48, no. 6, pp. 1570–1579, June 2002.

[14] M. Schwartz and A. Vardy, “On the stopping distance and stopping redundancy of codes,” in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 975–979.

[15] J. H. Weber and K. A. Abdel-Ghaffar, “Stopping set analysis for Hamming codes,” in *Proc. IEEE ISOC Information Theory Workshop on Coding and Complexity*, Rotorua, New Zealand, Aug./Sept. 2005, pp. 244–247.

[16] A. Orlitsky, R. Urbanke, K. Viswanathan, and J. Zhang, “Stopping sets and the girth of Tanner graphs,” in *Proc. IEEE International Symposium on Information Theory*, Lausanne, Switzerland, June/July 2002, p. 2.

[17] N. Kashyap and A. Vardy, “Stopping sets in codes from designs,” in *Proc. IEEE International Symposium on Information Theory*, Yokohama, Japan, June/July 2003, p. 122.

[18] A. Orlitsky, K. Viswanathan, and J. Zhang, “Stopping set distribution of LDPC code ensembles,” *IEEE Trans. Inform. Theory*, vol. 51, no. 3, pp. 929–953, Mar. 2005.

[19] M. Schwartz and A. Vardy, “On the stopping distance and the stopping redundancy of codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.

[20] T. Etzion, “On the stopping redundancy of Reed-Muller codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 4867–4879, Nov. 2006.

[21] H. D. L. Hollmann and L. M. G. M. Tolhuizen, “Generic erasure-correcting sets: bounds and constructions,” *J. Combin. Theory Ser. A*, vol. 113, pp. 1746–1759, Nov. 2006.

[22] ——, “On parity check collections for iterative erasure decoding that correct all correctable erasure patterns of a given size,” *IEEE Trans. Inform. Theory*, vol. 53, no. 2, pp. 823–828, Feb. 2007.

Chapter 3

General Bounds on Stopping Redundancy

It is of general interest to know how large the stopping redundancy is. In this chapter we derive bounds that apply generally to all linear codes. Stopping redundancy of special classes of codes will be discussed in subsequent chapters.

By definition, when the stopping distance is maximized, there are no stopping sets of size $(d-1)$ or smaller, where d is the minimum distance of the code. This is to say that for each $X \in \bigcup_{i=1}^{d-1} [n]^i$, there exists a row $\boldsymbol{h}$ in the parity-check matrix H, such that $|\operatorname{supp}(\boldsymbol{h}) \cap X| = 1$. Let w denote the weight of $\boldsymbol{h}$. The number of i-subsets of $[n]$ that intersect $\operatorname{supp}(\boldsymbol{h})$ at a single element is

$$w\binom{n-w}{i-1}.$$

By a simple set covering argument, the following lower bound on $\rho(\mathcal{C})$ was obtained in [1]. Let $d^{\perp}$ denote the minimum distance of $\mathcal{C}^{\perp}$. For each $i \in \{1, \ldots, d-1\}$,

$$\rho(\mathcal{C}) \geq \binom{n}{i} \Big/ w_i \binom{n-w_i}{i-1},$$

where

$$w_i = \max\left\{ \left\lceil \frac{n+1}{i} \right\rceil - 1, d^{\perp} \right\}.$$

Despite its simplicity, improvement to this lower bound appears difficult.

On the other hand, an upper bound on the stopping redundancy is arguably more interesting as it shows what *can* be done, rather than what *cannot*. In this chapter, we consider two approaches to obtaining an upper bound on the stopping redundancy. The first approach is explicitly constructive. Starting with a basis of $\mathcal{C}^{\perp}$, certain linear combinations of the basis vectors are judiciously selected to form a set of parity check constraints. The second approach is based on probabilistic methods [2] [3], in which we show the probability of the desired parity-check matrix is bounded away from zero over a finite ensemble. In general, much tighter upper bounds are possible using the second approach.

The rest of the chapter is arranged as follows. We start with binary codes. In Section 3.1 we propose an upper bound based on the explicit constructive approach, and relate it to several other known bounds in the literature. In Section 3.2, we obtain our first probabilistic bound, and compare it to the constructive bounds obtained in Section 3.1 as well as other known bounds based on different probabilistic arguments. Section 3.3 is devoted to improving the main bound obtained in Section 3.2, mostly by using a more elaborate probabilistic analysis. Section 3.4 shows how most of the results in this chapter can be extended to linear codes over an arbitrary field. Section 3.5 concludes the chapter.

3.1 Constructive Bounds

The first natural thought for an upper bound is to start with a non-redundant parity-check matrix, and add additional rows as linear combinations of the existing ones in a controlled manner such that all stopping sets with size smaller than d are guaranteed to have been eliminated. This idea was precisely that proposed in [4], and our first bound, as presented in the following theorem, goes along similar lines.

Theorem 3.1 *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code with $d \geq 2$. Then*

$$\rho(\mathcal{C}) \leq \sum_{i=1}^{\lfloor d/2 \rfloor} \binom{r}{2i-1}. \tag{3.1}$$

Proof: Take any basis of $\mathcal{C}^{\perp}$ as rows to form an $r \times n$ parity-check matrix H. For every set of i rows of H, where i is an odd number less than d, take their sum, and use the resulting vectors as rows to form a new parity-check matrix H'. By construction, the number of rows of H' is $\sum_{i=1}^{\lfloor d/2 \rfloor} \binom{r}{2i-1}$.

It suffices to show that $s(H') = d$. For any $X \subseteq [n]$, let $H(X)$ denote the submatrix consisting of columns of H indexed by elements of X. If $|X| < d$, then the columns of $H(X)$

are linearly independent. Hence, $H(X)$ contains a $|X| \times |X|$ full-rank submatrix, which we denote by A. Note that A^{-1} is full-rank, and hence must contain a row of odd weight. Since $A^{-1}A$ gives the identity matrix, this implies that among sums of an odd number of rows of A, at least one has weight one. By construction, this shows that $H'(X)$ contains a row of weight one, and hence X is not a stopping set. ■

The bound in (3.1) has been motivated by and can be viewed as a variant of [4, Theorem 4], which is provided below for reference.

Theorem 3.2 ([4]) *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code with $d \geq 3$. Then*

$$\rho(\mathcal{C}) \leq \sum_{i=1}^{d-2} \binom{r}{i}. \tag{3.2}$$

Another very related bound is due to Hollmann and Tolhuizen, who studied a somewhat different problem of constructing “generic erasure-correcting sets” [5, 6, 7, 8]. Nonetheless, their results can be interpreted as bounds on the stopping redundancy hierarchy (cf. [9, 10]) of general linear codes, and then specialized to bounds on $\rho(\mathcal{C})$. In particular, Hollmann and Tolhuizen prove in [7, Theorem 5.2] and [8, Theorem 4.1] the following result.

Theorem 3.3 ([7, 8]) *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code with $d \geq 3$. Then*

$$\rho(\mathcal{C}) \leq \sum_{i=0}^{d-2} \binom{r-1}{i}. \tag{3.3}$$

The proof of Theorem 3.3 is similar to the proof of Theorem 3.1 or Theorem 3.2, but uses a different criterion in selecting the sets of rows to be summed in forming the new parity-check matrix.

Using the fact $\binom{r}{i} = \binom{r-1}{i} + \binom{r-1}{i-1}$, it is easy to draw the following comparisons among the bounds (3.1), (3.2) and (3.3):

- If d is odd, then (3.1) and (3.3) are equivalent, and both are stronger than (3.2).
- If d is even, then (3.3) is generally stronger than either (3.1) or (3.2). In this case, the comparison between (3.1) and (3.2) can go both ways, since (3.1) includes the term $\binom{r}{d-1}$ while (3.2) does not. For example, if we let r grow with n and d be fixed, then compared to (3.2), the bound in (3.1) is asymptotically weaker.

Hollmann and Tolhuizen [5] also showed that Theorem 3.3 can be improved for the special case of even weight codes.

Theorem 3.4 ([5]) *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code with $d \geq 3$. If all codewords of $\mathcal{C}$ have even weight,*

$$\rho(\mathcal{C}) \leq 2\sum_{i=0}^{d-3} \binom{r-2}{i}. \tag{3.4}$$

3.2 Probabilistic Bounds

We now turn to a different approach, better known as the probabilistic method [2] [3]. The basic idea is that in order to show the existence of an object with a certain property, it suffices to show that the property holds with positive probability over a finite random ensemble of objects.

Theorem 3.5 *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code. Then*

$$\rho(\mathcal{C}) \leq \rho^*(n, d) + r - d + 1, \tag{3.5}$$

where $\rho^*(n, d)$ is the smallest integer t such that

$$\sum_{i=1}^{d-1} \binom{n}{i} \left(1 - \frac{i}{2^i}\right)^t < 1 \tag{3.6}$$

Proof: First, let's agree on some language. For a matrix H with n columns and $X \subseteq [n]$, we say that X is *covered* by H if $H(X)$ contains a row of weight one. Clearly, if H is a parity-check matrix for $\mathcal{C}$, then X is a stopping set if and only if it is not covered by H. Also, an *i-set* is generally any set that contains i elements. Similarly, an *i-subset* is a subset that contains i elements. For convenience, we will often use the term i-set to refer specifically to an i-subset of $[n]$, unless made clear otherwise. Hence, with the above notations, we have that a parity-check matrix H satisfies $s(H) = d$ if and only if H covers all i-sets, for $i = 1, \ldots, d-1$.

Now, for a given positive integer t, let H_t be a $t \times n$ matrix whose rows are drawn independently from $\mathcal{C}^\perp$ at random. Note that if H^* denotes a matrix whose rows consist of all vectors in $\mathcal{C}^\perp$, then H^* is an *orthogonal array* of strength $d-1$ [11, p. 139], i.e. for any i-set X, $i \leq d-1$, all 2^i binary i-tuples appear as rows in $H^*(X)$ the same number of times. Hence, a randomly chosen vector from $\mathcal{C}^\perp$ covers any given i-set with probability $i/2^i$, $i = 1, \ldots, d-1$, and the probability that a given i-set is *not* covered by H_t is $(1 - i/2^i)^t$, $i = 1, \ldots, d-1$. We have

$$\begin{aligned}
&\Pr\big(\{H_t \text{ covers all } i\text{-sets}, i = 1, \ldots, d-1\}\big) \\
&= 1 - \Pr\big(\{\exists X \subseteq [n], 1 \leq |X| < d, \text{not covered by } H_t\}\big) \\
&= 1 - \Pr\Bigg(\bigcup_{i=1}^{d-1} \bigcup_{X \in [n]^i} \{X \text{ is not covered by } H_t\}\Bigg) \\
&\geq 1 - \sum_{i=1}^{d-1} \sum_{X \in [n]^i} \left(1 - \frac{i}{2^i}\right)^t \\
&= 1 - \sum_{i=1}^{d-1} \binom{n}{i} \left(1 - \frac{i}{2^i}\right)^t .
\end{aligned}$$

If $\sum_{i=1}^{d-1} \binom{n}{i} \left(1 - i/2^i\right)^t < 1$, then $\Pr\big(\{H_t \text{ covers all } i\text{-sets}, i = 1, \ldots, d-1\}\big) > 0$, which implies that there exists at least one realization of H_t that covers all i-sets, $i = 1, \ldots, d-1$. Let H denote one such realization. Note that to ensure the rows of H be a spanning set of $\mathcal{C}^\perp$, we need to append no more than $r - d + 1$ additional rows to H. This is because $\mathrm{rank}(H) \geq d - 1$, which comes from the fact that H covers all i-sets up to $i = d-1$. Therefore, $\rho(\mathcal{C}) \leq t + r - d + 1$, for all t such that $\sum_{i=1}^{d-1} \binom{n}{i} \left(1 - i/2^i\right)^t < 1$. ■

The upper bound given in Theorem 3.5 involves solving an inequality. A closed form expression would be desirable. This is addressed in the following corollaries.

Corollary 3.6 *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code. If $1 < d < n/2$, then*

$$\rho(\mathcal{C}) \leq \frac{nH_2(\delta) + \frac{1}{2} \log \frac{\delta}{2\pi n(1-\delta)(1-2\delta)^2}}{-\log\left(1 - \frac{d-1}{2^{d-1}}\right)} + r - d + 1,$$

where $\delta = d/n$, and $H_2(x) = -x \log x - (1-x) \log(1-x)$ is the binary entropy function.

Proof: First, note that $(1 - i/2^i)$ is non-decreasing for $i \in \mathbb{N}$, so that

$$\sum_{i=1}^{d-1} \binom{n}{i} \left(1 - \frac{i}{2^i}\right)^t \leq \left(1 - \frac{d-1}{2^{d-1}}\right)^t \sum_{i=1}^{d-1} \binom{n}{i}. \tag{3.7}$$

Next, for $0 < \delta = d/n < 1/2$, it can be shown that

$$\sum_{i=1}^{d-1} \binom{n}{i} < \frac{\delta}{1-2\delta} \binom{n}{\delta n}. \tag{3.8}$$

Further, by Stirling's approximation it is known that ([12])

$$\binom{n}{\delta n} \leq \frac{1}{\sqrt{2\pi n \delta (1-\delta)}} 2^{nH_2(\delta)}. \tag{3.9}$$

Now, by putting together (3.7), (3.8), and (3.9), and referring to (3.6), we see that a positive solution for t to the equation

$$\frac{\delta}{1-2\delta} \frac{1}{\sqrt{2\pi n \delta (1-\delta)}} 2^{nH_2(\delta)} \left(1 - \frac{d-1}{2^{d-1}}\right)^t = 1$$

must be an upper bound on $\rho^*(n,d)$. We thus obtain

$$\rho^*(n,d) \leq \frac{nH_2(\delta) + \frac{1}{2} \log \frac{\delta}{2\pi n (1-\delta)(1-2\delta)^2}}{-\log\left(1 - \frac{d-1}{2^{d-1}}\right)}. \tag{3.10}$$

Plugging (3.10) in (3.5) we get the desired bound. ■

A weaker but much simpler closed form is also possible, without requiring $d < n/2$.

Corollary 3.7 *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code with $d > 1$. Then*

$$\rho(\mathcal{C}) \leq \frac{n}{-\log\left(1 - \frac{d-1}{2^{d-1}}\right)} + r - d + 1. \tag{3.11}$$

Proof: The argument is almost identical to the proof of Corollary 3.6, except that we instead bound $\sum_{i=1}^{d-1} \binom{n}{i}$ by

$$\sum_{i=1}^{d-1} \binom{n}{i} < 2^n.$$

■

Independently, Hollmann and Tolhuizen employed a different random-coding argument in [6, Theorem 4.2] and [8, Theorem 3.2] to show that

$$\rho(\mathcal{C}) \leq \frac{\log_2\big((2^r-1)(2^r-2)(2^r-2^2)\cdots(2^r-2^{d-2})\big)}{(d-1) - \log_2\big(2^{d-1} - (d-1)\big)} \tag{3.12}$$

As pointed out by Ludo Tolhuizen, another probabilistic bound on stopping redundancy, namely

$$\rho(\mathcal{C}) \leq \frac{(r-1)(d-2) \; - \; \log_2\big((d-2)!\big)}{(d-2) \; - \; \log_2(2^{d-2}-1)} \; + \; 1 \tag{3.13}$$

Table 3.1 Upper bounds on the stopping redundancy of the $[24, 12, 8]_2$ Golay and the $[48, 24, 12]_2$ QR codes

	$[24, 12, 8]_2$ Golay	$[48, 24, 12]_2$ QR
Schwartz-Vardy (3.2)	$2,509$	$4,540,385$
Han-Siegel (3.1)	$1,816$	$4,194,304$
Hollmann-Tolhuizen (3.3)	$1,486$	$2,842,226$
Hollmann-Tolhuizen (3.4)	$1,276$	$2,195,580$
Hollmann-Tolhuizen (3.13)	$2,488$	$147,712$
Hollmann-Tolhuizen (3.12)	$1,034$	$33,978$
Han-Siegel (3.5)	232	$4,440$

follows by combining Proposition 6.2, Lemma 6.5, and Theorem 6.14 of [7].

Although the various bounds in (3.1) – (3.5), (3.12), (3.13) are difficult to compare analytically, the following claims seem to be justified:

- For most code parameters, the probabilistic bounds (3.12), (3.13), (3.5) are much better than the constructive bounds (3.1) – (3.4).

- Among the probabilistic bounds, our bound (3.5) is by far the strongest.

Using the tables of [13], we have verified that these claims hold for *all* possible code parameters n, k, and d with $d > 4$ and $n \leq 256$. A representative picture is shown in Table 3.1, where we computed the bounds (3.1) – (3.5), (3.12), (3.13) for the $[24, 12, 8]_2$ Golay code and for the $[48, 24, 12]_2$ quadratic-residue code. Thus the new bound (3.5) appears to be the best currently known bound on the stopping redundancy of general linear codes.

On the other hand, the asymptotic behavior of the various upper bounds (as functions of n, r, and d, as $n \to \infty$) is relatively easy to obtain. Here, we consider two cases. The first case corresponds to "good" codes, i.e. codes whose rate is bounded away from zero and whose minimum distance is non-diminishing relative to the code length. The second case concerns codes with fixed minimum distance, an example of which is the family of extended binary Hamming codes.

Case 1: *$d = \delta n$, $r = \gamma n$, where $0 < \delta < 1/2$, $0 < \gamma < 1$ are constants.*

Noting that $-\log(1-x) \sim x/\ln 2$ as $x \to 0$, we see the upper bound (3.11) is $O(2^d)$, hence so is the bound (3.5). On the other hand, note that

$$\sum_{i=1}^{d-1} \binom{n}{i} \left(1 - \frac{i}{2^i}\right)^t \geq \binom{n}{d-1} \left(1 - \frac{d-1}{2^{d-1}}\right)^t.$$

Setting

$$\binom{n}{d-1} \left(1 - \frac{d-1}{2^{d-1}}\right)^t = 1,$$

and solving for t, one can readily show that $\rho^*(n,d)$ is also $\Omega(2^d)$. Therefore, the bound (3.5) is indeed $\Theta(2^d)$.

In comparison, all the constructive bounds (3.1) – (3.4) are

$$\Omega\left(\binom{r}{d}\right) = \Omega\left(\binom{\gamma n}{\delta n}\right) = \Omega\left(\frac{1}{\sqrt{n}} 2^{H_2\left(\frac{\delta}{\gamma}\right)\gamma n}\right).$$

By the asymptotic Plotkin bound, we have $\gamma \geq 2\delta$ for $0 < \delta < 1/2$. Noting further that $H_2(x) \geq 2x$ for all $0 \leq x \leq 1/2$, we have

$$\Omega\left(\frac{1}{\sqrt{n}} 2^{H_2\left(\frac{\delta}{\gamma}\right)\gamma n}\right) = \Omega\left(\frac{1}{\sqrt{n}} 2^{\frac{2\delta}{\gamma}\cdot\gamma n}\right) = \Omega\left(\frac{2^{2d}}{\sqrt{n}}\right).$$

This is clearly much worse than (3.5).

Finally, it is not hard to show that the bounds (3.12) and (3.13) are both $\Theta\big(n2^d\big)$, thus both weaker than (3.5).

Case 2: *d is a fixed constant.*

Using Corollary 3.6, it is simple to show that the bound (3.5) is $\Theta(\log n + r)$. In comparison, among the constructive bounds, (3.1) is $\Theta\big(r^{d-2}\big)$ if d is odd, $\Theta\big(r^{d-1}\big)$ if d is even, (3.2) is $\Theta\big(r^{d-2}\big)$, (3.3) is $\Theta\big((r-1)^{d-2}\big)$, and (3.4) (for even weight codes) is $\Theta\big((r-2)^{d-3}\big)$. The other probabilistic bounds (3.12) and (3.13) are both $\Theta(r)$.

Let's interpret the above results for $d > 4$. For $d \leq 4$, much more specific results are known for $\rho(\mathcal{C})$, hence a comparison of the general bounds is less interesting. (In particular, it is known [4] that $\rho(\mathcal{C}) = r$ for all $d \leq 3$. And for $d = 4$, it was shown in [14] that $\rho(\mathcal{C}) \leq 2r - 3$, and that this bound is tight.) By the sphere-packing bound, we have $r > 2\log n - 1$. From here, it is easy to deduce that (3.5) is asymptotically tighter than any of the constructive bounds (3.1) – (3.4). For the probabilistic bounds, note that (3.5), (3.12) and (3.13) all grow *linearly* with the redundancy of the code. A more detailed analysis shows that (3.5) is asymptotically at most $r + c_1(d-1)\log n$, while (3.12) is asymptotic to $c_1(d-1)r$, and (3.13) is asymptotic to $c_2(d-2)r$, where c_1, c_2 are constants that depend only on d, and $c_2 > c_1 > 1$ for $d > 4$. Since $r > 2\log n - 1$, bound (3.5) is asymptotically stronger than (3.12) or (3.13).

Before ending this section, we give another look at the $[24, 12, 8]_2$ Golay code, one of the most famous binary codes known. We present here a parity-check matrix with 34 rows that maximizes stopping distance and corrects more low-weight erasure patterns than the parity-check matrix given in [4].[1] The details of our parity-check matrix, denoted by H_{HS}, are given in Table 3.2. It was found by a greedy computer search. The idea is to start with a random selection of codewords from $\mathcal{G}_{24}$ (note that $\mathcal{G}_{24}$ is self-dual), and in each iteration, replace one codeword in the selection so that as many more i-sets ($1 \leq i \leq 7$) as possible are covered. When no such improvements can be made, an additional codeword is added to the selection and the iteration continues. The process is stopped when the desired stopping distance is achieved. We find that it is enough to only consider covering 7-sets, and verify in the end that the matrix obtained indeed covers all smaller i-sets as well. (Note that since $\mathcal{G}_{24}$ is *maximal*, there is no need to check the rank of the parity-check matrix. (cf. Proposition 3.12.))

Table 3.3 compares the number of undecodable erasure patterns by weight w (number of erased bits) for iterative decoders based on H_{HS}, H'_{24} (the 34-row parity-check matrix reported in [4]), and the maximum-likelihood decoder. We see that the iterative decoder based on H_{HS} corrects considerably more lower weight erasure patterns than does the one based on H'_{24}, which implies that it will perform better when the erasure probability is small. For a binary erasure channel with erasure probability p, a detailed comparsion shows that for all $p < 0.349$, the iterative decoder based on H_{HS} has a smaller probability of decoding failure.

3.3 Improvements to the Bound (3.5)

In this section, we present several improvements upon the probabilistic bound (3.5) using a number of ideas, most of which are based upon a more careful probabilistic analysis.

Given a linear code $\mathcal{C}$, in Theorem 3.5 we construct a parity-check matrix H for $\mathcal{C}$ by drawing codewords independently and uniformly at random from the dual code $\mathcal{C}^{\perp}$. The

[1] In early versions of [4] and [1], a 35-row parity-check matrix was initially proposed, which was improved to 34 rows following our comments.

Table 3.2 Parity check matrix with 34 rows for $\mathcal{G}_{24}$ that achieves stopping distance 8

$$H_{\text{HS}} = \begin{pmatrix}
0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0 \\
0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1 \\
0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0 \\
1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0 \\
0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1 \\
0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0 \\
0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1 \\
1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0 \\
1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \\
0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1 \\
1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1 \\
0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0 \\
1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0 \\
0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0 \\
1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0 \\
0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0 \\
0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1 \\
0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1 \\
0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0 \\
0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \\
0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1 \\
0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0 \\
1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1 \\
0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1 \\
0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \\
0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1 \\
1\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \\
0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \\
0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0 \\
1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0 \\
1\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0 \\
1\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0 \\
1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \\
0\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0
\end{pmatrix}$$

Table 3.3 Number of undecodable erasure patterns by weight w for different iterative decoders for $\mathcal{G}_{24}$

w	$\Psi_{H_{\mathrm{HS}}}(w)$	$\Psi_{H'_{24}}(w)$	$\Psi_{\mathrm{ML}}(w)$
≤ 7	0	0	0
8	3284	3598	759
9	78218	82138	12144
10	580166	585157	91080
11	1734967	1717082	425040
12	2569618	2556402	1313116
≥ 13	$\binom{24}{w}$	$\binom{24}{w}$	$\binom{24}{w}$

resulting bound (3.5) can be put in the following alternative form:

$$\rho(\mathcal{C}) \leq \min\{t \in \mathbb{N} : \mathcal{E}_{n,d}(t) < 1\} + (r-d+1) \tag{3.14}$$

where

$$\mathcal{E}_{n,d}(t) \overset{\mathrm{def}}{=} \sum_{i=1}^{d-1} \binom{n}{i} \left(1 - \frac{i}{2^i}\right)^t \tag{3.15}$$

Our first observation is this: such random choice is efficient at first, but becomes less and less efficient as successive rows of H are drawn from $\mathcal{C}^\perp$. At some point, deterministic selection becomes superior. This leads to the following result.

Theorem 3.8 *Let $\mathcal{C}$ be an $[n,k,d]_2$ code. Let $\Delta(\mathcal{C})$ denote the* **deficiency** *of $\mathcal{C}$, that is $\Delta(\mathcal{C}) \overset{\mathrm{def}}{=} n-k+1-d$. Let $\mathcal{E}_{n,d}(t)$ be the expectation function defined in* (3.15). *Then*

$$\rho(\mathcal{C}) \leq \min_{t\in\mathbb{N}} \left\{t + \lfloor \mathcal{E}_{n,d}(t) \rfloor\right\} + \Delta(\mathcal{C}) \tag{3.16}$$

Proof: Let

$$\mathscr{U} \overset{\mathrm{def}}{=} \bigcup_{i=1}^{d-1} [n]^i. \tag{3.17}$$

Let H_t be a $t \times n$ matrix whose rows are drawn from $\mathcal{C}^\perp$ independently and uniformly at random. If $\mathcal{S} \in \mathscr{U}$ is a fixed i-set and $\boldsymbol{h}$ is a fixed row of H_t, then the probability that $\boldsymbol{h}$ covers $\mathcal{S}$ is $i/2^i$. Again, this is so because $\mathcal{C}^\perp$ is an orthogonal array of strength $d-1$ (cf.[11, p. 139]), which means precisely that for all $\mathcal{S} \in \mathscr{U}$, a codeword drawn at random from $\mathcal{C}^\perp$ is equally likely to contain each of the 2^i possible vectors on the i positions indexed by $\mathcal{S}$. Let X_t denote the number of sets $\mathcal{S} \in \mathscr{U}$ that are not covered by H_t. Then X_t is a random variable, and the expected value of X_t is given by

$$\mathsf{E}[X_t] = \sum_{\mathcal{S}\in\mathscr{U}} \Pr\bigl\{\mathcal{S} \text{ not covered by } H_t\bigr\} = \sum_{i=1}^{d-1}\binom{n}{i}\Bigl(1-\frac{i}{2^i}\Bigr)^t$$

The right-hand side of the above expression is $\mathcal{E}_{n,d}(t)$, by definition. It follows that there exists a realization H of H_t that covers all but at most $\lfloor\mathcal{E}_{n,d}(t)\rfloor$ sets in $\mathscr{U}$. For each uncovered set $\mathcal{S}$, there is a codeword of $\mathcal{C}^{\perp}$ that covers $\mathcal{S}$ (again, since $\mathcal{C}^{\perp}$ is an orthogonal array). Thus we can adjoin $\lfloor\mathcal{E}_{n,d}(t)\rfloor$ rows to H to create a matrix H' with $s(H') = d$. It is possible that $\mathrm{rank}(H') < n-k$, so H' is not necessarily a parity-check matrix for $\mathcal{C}$. However, it is easy to see that $\mathrm{rank}(H') \geq d-1$. Hence, by adjoining at most $\Delta(\mathcal{C}) = (n-k)-(d-1)$ rows to H', we finally obtain a parity-check matrix H'' for $\mathcal{C}$ with at most $t + \lfloor\mathcal{E}_{n,d}(t)\rfloor + \Delta(\mathcal{C})$ rows and $s(H'') = d$. ■

Since the minimization in (3.16) contains the bound (3.5) as a special case, Theorem 3.8 is at least as strong as (3.5). In fact, it is often substantially stronger (cf. Table 3.4).

The bound of Theorem 3.8 involves solving a minimization problem; a closed-form expression would be desirable. This is addressed in the following corollary, which is similar in spirit to Corollary 3.6 and improves upon it.

Corollary 3.9 *Let $\mathcal{C}$ be an $[n,k,d]_2$ code. Let $\Delta(\mathcal{C})$ be the deficiency of $\mathcal{C}$, as before, and define*

$$C \overset{\text{def}}{=} \sum_{i=1}^{d-1}\binom{n}{i} \simeq 2^{nH_2(\frac{d}{n})}$$

$$D \overset{\text{def}}{=} -\ln\left(1-\frac{d-1}{2^{d-1}}\right) \simeq \frac{d-1}{2^{d-1}}$$

where $H_2(x) \overset{\text{def}}{=} -x\log_2 x - (1-x)\log_2(1-x)$ is the binary entropy function. Then

$$\rho(\mathcal{C}) \leq \left\lceil \frac{\ln C + \ln D + 1}{D} \right\rceil + \Delta(\mathcal{C}) \tag{3.18}$$

Proof: In order to solve the minimization problem in (3.16), albeit approximately, we upper-bound $\mathcal{E}_{n,d}(t)$ as follows

$$\mathcal{E}_{n,d}(t) \leq \sum_{i=1}^{d-1}\binom{n}{i}\left(1-\frac{d-1}{2^{d-1}}\right)^t = Ce^{-Dt}$$

Table 3.4 Improved bounds on the stopping redundancy of the $[24, 12, 8]_2$ and the $[48, 24, 12]_2$ codes

	$[24, 12, 8]_2$ Golay	$[48, 24, 12]_2$ QR
Han-Siegel (3.5)	232	$4,440$
Theorem 3.8	198	$3,655$
Corollary 3.9	213	$3,738$
Theorem 3.10	194	$3,655$
Theorem 3.11	187	$3,577$
Proposition 3.12 Theorem 3.14	182	$3,564$

where the inequality follows from the fact that $i/2^i$ is non-increasing in i. Let $f(t) = t + Ce^{-Dt}$. Then $f(t)$, regarded as a function from $\mathbb{R}$ to $\mathbb{R}$, is convex and attains its global minimum at $t_0 = (\ln C + \ln D)/D$. It is easy to see that t_0 is always positive and $f(t_0) = (\ln C + \ln D + 1)/D$. Since $f(\lceil t \rceil) \le \lceil f(t) \rceil$ for all t, the corollary now follows from Theorem 3.8. ■

Next, we observe that the random choice method used in Theorem 3.5 and Theorem 3.8 is not optimal. It would be better to select the rows of H_t from $\mathcal{C}^\perp \setminus \{\mathbf{0}\}$, rather than $\mathcal{C}^\perp$, without replacement, rather than with replacement. This leads to the following bound.

Theorem 3.10 *Let $\mathcal{C}$ be an $[n, k, d]_2$ code, let $\Delta(\mathcal{C})$ be the deficiency of $\mathcal{C}$, and let $r = n - k$. Then*

$$\rho(\mathcal{C}) \;\le\; \min_{t \in \mathbb{N}} \bigl\{ t + \lfloor \mathcal{F}_{n,d,k}(t) \rfloor \bigr\} \;+\; \Delta(\mathcal{C}) \tag{3.19}$$

where

$$\mathcal{F}_{n,d,k}(t) \;\stackrel{\text{def}}{=}\; \sum_{i=1}^{d-1} \binom{n}{i} \prod_{j=1}^{t} \left(1 - \frac{i 2^{r-i}}{2^r - j} \right) \tag{3.20}$$

Proof: We construct a parity-check matrix for $\mathcal{C}$ in the same way as in Theorem 3.8, except that the initial matrix H_t is selected differently. Specifically, the rows $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_t$ of H_t are selected as follows. Suppose that we have already chosen the first $j-1$ rows $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_{j-1}$, then the j-th row $\boldsymbol{h}_j$ is selected uniformly at random from

$$\mathcal{C}^\perp \setminus \bigl\{ \mathbf{0}, \boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_{j-1} \bigr\} \tag{3.21}$$

Now let $\mathcal{S} \in \mathscr{U}$ be a fixed i-set that is not covered by any of $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_{j-1}$. What is the probability that the j-th row covers $\mathcal{S}$? The total number of possible (equally likely) choices for

$\boldsymbol{h}_j$ is $2^r - j$. Of these, precisely $i2^{r-i}$ cover $\mathcal{S}$. This is so because there are exactly $i2^{r-i}$ codewords in $\mathcal{C}^\perp$ that cover $\mathcal{S}$ (again, since $\mathcal{C}^\perp$ is an orthogonal array), and none of these is among $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_{j-1}$ by assumption. Hence, the probability that $\boldsymbol{h}_j$ covers $\mathcal{S}$ is $i2^{r-i}/(2^r - j)$. It follows that for any fixed i-set $\mathcal{S} \in \mathscr{U}$, the probability that H_t does not cover $\mathcal{S}$ is

$$\Pr\bigl\{\mathcal{S} \text{ is not covered by } H_t\bigr\} \;=\; \prod_{j=1}^{t} \left(1 - \frac{i2^{r-i}}{2^r - j}\right)$$

Thus $\mathcal{F}_{n,d,k}(t)$ in (3.20) is the expected number of sets $\mathcal{S} \in \mathscr{U}$ that are not covered by H_t, and the theorem follows. ■

Since (3.20) is extremely close to (3.15), it is somewhat surprising that Theorem 3.10 yields any improvements upon Theorem 3.8. But it does, at least for small code parameters (cf. Table 3.4).

We now improve the construction of a parity-check matrix for $\mathcal{C}$ described in Theorem 3.10 in yet another respect. Let H_0 be a fixed $t \times n$ matrix whose rows $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_t$ are nonzero codewords of $\mathcal{C}^\perp$, and let $\mathscr{U}_0$ denote the subset of the set $\mathscr{U}$ in (3.17) consisting of those sets that are not covered by H_0. Let $X_0 = |\mathscr{U}_0|$. Suppose we adjoin another row $\boldsymbol{h}_{t+1}$ to H_0, selected uniformly at random from $\mathcal{C}^\perp \setminus \bigl\{\boldsymbol{0}, \boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_t\bigr\}$ as in (3.21), and let H_1' denote the resulting random matrix. Let X_1' be the number of sets $\mathcal{S} \in \mathscr{U}$ that are not covered by H_1'. Then, arguing as in the proof of Theorem 3.10, we find that

$$\begin{aligned} \mathsf{E}[X_1'] &= \sum_{\mathcal{S} \in \mathscr{U}_0} \Pr\bigl\{\mathcal{S} \text{ is not covered by } \boldsymbol{h}_{t+1}\bigr\} \\ &\le X_0 \left(1 - \frac{(d-1)2^{r-d+1}}{2^r - (t+1)}\right) \end{aligned}$$

It follows that there exists a $(t{+}1) \times n$ realization H_1 of H_1' that covers all but at most

$$X_1 = \left\lfloor X_0 \left(1 - \frac{(d-1)2^{r-d+1}}{2^r - (t+1)}\right) \right\rfloor \tag{3.22}$$

sets in $\mathscr{U}$. The process can be now iterated. That is, given H_1, we can construct a $(t{+}2) \times n$ matrix H_2 that covers all but at most

$$X_2 = \left\lfloor X_1 \left(1 - \frac{(d-1)2^{r-d+1}}{2^r - (t+2)}\right) \right\rfloor \tag{3.23}$$

sets in $\mathscr{U}$. And so on. To formalize this process, let us define for all $j = 1, 2, \ldots$, the function $P_j : \mathbb{N} \to \mathbb{N}$ as follows

$$P_j(k) \overset{\text{def}}{=} \left\lfloor k \left(1 - \frac{(d{-}1)2^{r-d+1}}{2^r - (t+j)} \right) \right\rfloor \qquad \text{for all } k \in \mathbb{N}$$

where the parameters d, r, and t are regarded as constant. Then, after i iterations of the foregoing procedure, we will construct a $(t+i) \times n$ matrix H_i that covers all but at most

$$Q_i(X_0) \overset{\text{def}}{=} P_i\Big(P_{i-1}\Big(\cdots\ P_2\big(P_1(X_0)\big)\Big)\Big) \tag{3.24}$$

sets in $\mathscr{U}$. If $Q_i(X_0) = 0$, we are done. This establishes the following upper bound on stopping redundancy.

Theorem 3.11 *Let $\mathcal{C}$ be an $[n,k,d]$ binary linear code, with deficiency $\Delta(\mathcal{C})$. Then the stopping redundancy $\rho(\mathcal{C})$ is at most*

$$\min_{t \in \mathbb{N}} \Big\{ t + \min\big\{ i \in \mathbb{N} : Q_i\big(\lfloor \mathcal{F}_{n,d,k}(t) \rfloor\big) = 0 \big\} \Big\} + \Delta(\mathcal{C}) \tag{3.25}$$

where $\mathcal{F}_{n,d,k}(t)$ is the expectation function defined in (3.20) *while $Q_i(\cdot)$ is the function defined in* (3.24).

Although the definition of $Q_i(\cdot)$ in (3.24) appears to be rather involved, we observe that the minimization over i in (3.25) is, in fact, very easy to compute: all it takes is a single-line while loop. The bound of Theorem 3.11 clearly includes Theorem 3.10 as a special case, and improves upon it (cf. Table 3.4).

Finally, we would like to get rid of the small, but annoying, deficiency term $\Delta(\mathcal{C})$ in Theorems 3.8 – 3.11. To this end, the following simple observation often suffices. A code $\mathcal{C} \subseteq \mathbb{F}_2^n$ is said to be *maximal* if it is not possible to adjoin any vector in $\mathbb{F}_2^n$ to $\mathcal{C}$ without decreasing its minimum distance.

Proposition 3.12 *Let $\mathcal{C}$ be an $[n,k,d]_2$ code. Let H be any matrix with $s(H) = d$ whose rows are codewords of $\mathcal{C}^\perp$. If $\mathcal{C}$ is maximal, then* $\operatorname{rank}(H) = n - k$.

Proof: Assume to the contrary that $\operatorname{rank}(H) < n - k$. Then H is a parity-check matrix for a proper supercode $\mathcal{C}'$ of $\mathcal{C}$. Since $s(H) = d$, the minimum distance of $\mathcal{C}'$ is also d, which contradicts the fact that $\mathcal{C}$ is maximal. ■

Proposition 3.12 implies that if $\mathcal{C}$ is maximal, we can drop the deficiency term $\Delta(\mathcal{C})$ in Theorems 3.5, 3.8 – 3.11. We can also (effectively) get rid of this term using a more elaborate probabilistic argument. It is intuitively clear that if we draw sufficiently many codewords uniformly at random from the dual of an $[n, k, d]$ code $\mathcal{C}$, the resulting matrix is likely to have rank close to $r = n - k$. This observation is made formal in the following lemma.

Lemma 3.13 *Let $\mathcal{C}$ be an $[n, k, d]_2$ code, and let H_t be a $t \times n$ matrix whose rows are drawn uniformly at random (either with or without replacement) from the dual code $\mathcal{C}^\perp$. Let $r = n - k$, and define the random variable $Y_t = r - \operatorname{rank}(H_t)$. Then for all $t \geq 0$, we have*

$$\mathsf{E}[Y_t] \;\leq\; \mathcal{D}_r(t), \tag{3.26}$$

where

$$\mathcal{D}_r(t) \;\overset{\text{def}}{=}\; \begin{cases} 6/7 & \textit{if } t = r \\ (r-t)^+ + \frac{1}{2^{|t-r|}}\left(1 + \frac{2/3}{2^{|t-r|+1}-1}\right) & \textit{if } t \neq r \end{cases} \tag{3.27}$$

and $(x)^+ \overset{\text{def}}{=} \max\{x, 0\}$.

Proof: See Appendix Appendix 3.A. ■

In order to combine Lemma 3.13 with our earlier results, let us define the function

$$\mathcal{G}_{n,d,k}(t) \;\overset{\text{def}}{=}\; \mathcal{F}_{n,d,k}(t) \;+\; \mathcal{D}_{n-k}(t) \tag{3.28}$$

where $\mathcal{F}_{n,d,k}(t)$ and $\mathcal{D}_r(t)$ are as defined in (3.20) and (3.27), respectively. We can now modify the proofs of Theorem 3.10 and Theorem 3.11 accordingly, thereby establishing the following result.

Theorem 3.14 *Let $\mathcal{C}$ be an $[n, k, d]_2$ code, and let $r = n - k$. Then the stopping redundancy of $\mathcal{C}$ is bounded by*

$$\rho(\mathcal{C}) \;\leq\; \min\nolimits_{t \geq r}\bigl\{t + \lfloor \mathcal{G}_{n,d,k}(t) \rfloor\bigr\} \tag{3.29}$$

Moreover, if $(r-1)(d-1) \leq 2^{d-1}$ then

$$\rho(\mathcal{C}) \leq \min_{t \geq r}\Bigl\{t + \min\bigl\{i \in \mathbb{N} : Q_i\bigl(\lfloor \mathcal{G}_{n,d,k}(t) \rfloor\bigr) = 0\bigr\}\Bigr\} \tag{3.30}$$

where the functions $\mathcal{G}_{n,d,k}(t)$ and $Q_i(\cdot)$ are as defined in (3.28) *and* (3.24)*, respectively.*

Proof: Use the same argument as in the proof of Theorem 3.10, but with respect to the random variable $Z_t = X_t + Y_t$, which is the sum of the number of sets $\mathcal{S} \in \mathscr{U}$ not covered by H_t and its rank deficiency $r - \mathrm{rank}(H_t)$. Further, note that the condition $(r-1)(d-1) \leq 2^{d-1}$ is sufficient for the argument in (3.22), (3.23), and (3.24) to be applicable in this case as well, which can be seen as follows.

Let H_0 be a fixed $t \times n$ matrix whose rows $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_t$ are nonzero codewords of $\mathcal{C}^\perp$, and let $\mathscr{U}_0$ denote the subset of the set $\mathscr{U}$ in (3.17) consisting of those sets that are not covered by H_0. Let $X_0 = |\mathscr{U}_0|$. Let $Y_0 = r - \mathrm{rank}(H_0)$ denote the rank deficiency of H_0. Suppose we adjoin another row $\boldsymbol{h}_{t+1}$ to H_0, selected uniformly at random from $\mathcal{C}^\perp \setminus \{\mathbf{0}, \boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_t\}$ as in (3.21), and let H_1' denote the resulting random matrix. We would like to apply the argument in (3.22) now to $X_0 + Y_0$. For that we may ssume $Y_0 \neq 0$. Let Y_1' be the rank deficiency of H_1'. Then we have

$$\begin{aligned}\mathsf{E}[Y_1'] &= \frac{2^{r-Y_0} - (t+1)}{2^r - (t+1)} Y_0 + \frac{2^r - 2^{r-Y_0}}{2^r - (t+1)} (Y_0 - 1) \\ &= Y_0 \left(1 - \frac{1}{Y_0} \cdot \frac{2^r - 2^{r-Y_0}}{2^r - (t+1)}\right).\end{aligned}$$

Thus, to apply the argument in (3.22) to $X_0 + Y_0$, it suffices to require

$$\frac{(d-1)2^{r-d+1}}{2^r - (t+1)} \leq \frac{1}{Y_0} \cdot \frac{2^r - 2^{r-Y_0}}{2^r - (t+1)}$$

which is equivalent to

$$\frac{d-1}{2^{d-1}} \leq \frac{1 - 2^{-Y_0}}{Y_0}$$

for all $Y_0 = 1, 2, \ldots, r-2$. (We may assume $r > 2$, because if $r \leq 2$ then $Y_0 = 0$ for all $t \geq r$. Note that for $t \geq r > 2$ we necessarily have $Y_0 \leq r - 2$.) Now, note that the right-hand side of the above equation is non-increasing in Y_0. Hence,

$$\frac{1 - 2^{-Y_0}}{Y_0} \geq \frac{1 - 2^{-(r-2)}}{r-2} \geq \frac{1}{r-1}.$$

So as long as

$$\frac{d-1}{2^{d-1}} \leq \frac{1}{r-1}$$

there exists a $(t+1) \times n$ realization H_1 of H_1' whose number of uncovered sets in $\mathscr{U}$ plus rank deficiency is at most

$$\left\lfloor (X_0 + Y_0)\left(1 - \frac{(d-1)2^{r-d+1}}{2^r - (t+1)}\right) \right\rfloor$$

The process can now be iterated, so that (3.23) and (3.24) are applicable as well. ■

It is not immediately apparent that (3.29) produces a better bound on stopping redundancy than Theorem 3.10. However, we can show that this is *always* so, except for a few trivial cases. In fact, comparing (3.20) and (3.26), we see that the second term in (3.28) decreases with t exponentially faster than the first term. Thus, unless the minimum in (3.29) and/or (3.30) is achieved for t very close to r (in which case $\rho(\mathcal{C})$ must be close to r as well), the second term in (3.28) has essentially no effect on the minimization — this term is a tiny fraction, which disappears when taking the floor of $\mathcal{G}_{n,d,k}(t)$. It follows that for virtually all code parameters, the net effect of (3.29) and (3.30) consists of eliminating the deficiency term $\Delta(C)$ in Theorems 3.10 and 3.11.

3.4 Nonbinary Codes

Up to now, we have considered only binary codes for the convenience of discussion. However, all bounds that have been obtained so far can be extended to the case of nonbinary codes rather straightforwardly. As such, we will give just a few examples in this section to illustrate the idea.

The principles behind the constructive bounds in Theorems 3.1 – 3.3 apply to nonbinary codes equally well. However, and a bit unfortunately, the various tricks to reduce the number of constructed parity checks that made these bounds different are apparently specific to binary codes. Thus, we end up with the following constructive bound on the stopping redundancy for codes over $\mathbb{F}_q$.

Theorem 3.15 *Let $\mathcal{C}$ be a linear code over $\mathbb{F}_q$. Then*

$$\rho(\mathcal{C}) \le \sum_{i=1}^{d-1} \binom{r}{i} (q-1)^{i-1}.$$

Proof: The proof is similar to that of Theorem 3.1. Here we take a basis of $\mathcal{C}^{\perp}$ and construct H such that its rows consist of linear combinations of every set of i basis vectors, $i = 1, \ldots, d-1$, with nonzero coefficients. Note that for each set of i basis vectors, we may fix the coefficient of one of the vectors as 1. ■

For a linear code $\mathcal{C}$ over $\mathbb{F}_q$, the codewords of $\mathcal{C}^\perp$ are known to form an orthogonal array of strength $d-1$ with q levels ([15, ch. 4]). Therefore, the argument we used to prove Theorem 3.5 also extends directly to nonbinary codes.

Theorem 3.16 *Let $\mathcal{C}$ be an $[n, n-r, d]_q$ code. Then*

$$\rho(\mathcal{C}) \leq \rho^*(n, d, q) + r - d + 1,$$

where $\rho^*(n, d, q)$ is the smallest integer t such that

$$\sum_{i=1}^{d-1} \binom{n}{i} \left(1 - \frac{(q-1)i}{q^i}\right)^t < 1.$$

Proof: Omitted. ■

Corollary 3.17 *Let $\mathcal{C}$ be an $[n, n-r, d]_q$ code with $1 < d < n/2$. Then*

$$\rho(\mathcal{C}) \leq \frac{nH_2(\delta) + \frac{1}{2}\log\frac{\delta}{2\pi n(1-\delta)(1-2\delta)^2}}{-\log\left(1 - \frac{(q-1)(d-1)}{q^{d-1}}\right)} + r - d + 1,$$

where $\delta = d/n$, and $H_2(x) = -x\log x - (1-x)\log(1-x)$ is the binary entropy function.

Proof: Omitted. ■

Corollary 3.18 *Let $\mathcal{C}$ be an $[n, n-r, d]_q$ code with $d > 1$. Then*

$$\rho(\mathcal{C}) \leq \frac{n}{-\log\left(1 - \frac{(q-1)(d-1)}{q^{d-1}}\right)} + r - d + 1.$$

Proof: Omitted. ■

Example 3.1 Let $\mathcal{G}_{12}$ denote the $[12, 6, 6]_3$ Golay code. The bound of Theorem 3.15 gives $\rho(\mathcal{G}_{12}) \leq 332$, while the bound of Theorem 3.16 gives $\rho(\mathcal{G}_{12}) \leq 160$. The best known result (by a greedy search, see [1]) is $\rho(\mathcal{G}_{12}) \leq 22$. □

We comment on the asymptotic comparison between the bounds of Theorem 3.15 and Theorem 3.16 as $n \to \infty$. Here we will only treat the case of "good" codes.

Let $d = \delta n$, $r = \gamma n$, where $0 < \delta < (q-1)/q$ and $0 < \gamma < 1$ are constants. Let $\theta = (q-1)/q$. For $0 < \delta < \theta$, we have $0 < \delta/\gamma \leq \theta$ by the asymptotic Plotkin bound. Noting that $H_2(p) \geq \big(H_2(\theta)/\theta\big) \cdot p$ for all $0 < p \leq \theta,\ 0 < \theta < 1$, we see that the upper bound of Theorem 3.15 is

$$\begin{aligned}
&\sum_{i=1}^{d-1} \binom{r}{i} (q-1)^{i-1} \\
&= \Omega\left(\binom{r}{d-1}(q-1)^{d-2}\right) \\
&= \Omega\left(\frac{1}{\sqrt{n}} 2^{H_2\left(\frac{\delta}{\gamma}\right)\cdot(\gamma n)} \cdot (q-1)^d\right) \\
&= \Omega\left(\frac{1}{\sqrt{n}} 2^{\frac{H_2(\theta)}{\theta}\cdot\frac{\delta}{\gamma}\cdot\gamma n} \cdot (q-1)^d\right) \\
&= \Omega\left(\frac{1}{\sqrt{n}} q^{\frac{qd}{q-1}}\right).
\end{aligned}$$

On the other hand, similar to the binary case, one can show that the bound of Theorem 3.16 is $\Theta(q^d)$. Hence, for "good" codes, the probabilistic bound of Theorem 3.16 is asymptotically tighter than the constructive bound of Theorem 3.15.

3.5 Discussion and Concluding Remarks

In this chapter we have obtained several general upper bounds on the stopping redundancy. There have been two main approaches. The first one follows the idea of Schwartz and Vardy [4] (independently Hollmann and Tolhuizen [7]), and is based on explicitly constructing a parity-check matrix by taking linear combinations of basis vectors of the dual code. The second approach, which is our main contribution, is based on probabilistic methods. We have shown through direct and asymptotic comparisons that the upper bounds obtained through the second approach are generally much tighter than those based on the first approach. Although most of our results were stated and proved herein for binary linear codes, they extend straightforwardly to linear codes over an arbitrary finite field, and we have given a few examples.

Extension to bounds on the stopping redundancy *hierarchy* [9, 10] is also straightforward: simply replace all occurrences of d in the theorems by the corresponding index of the hierarchy. Generalization to bounds on the trapping redundancy [16] should be relatively easy as well, although, perhaps, less straightforward.

In Section 3.3, we have tried to push the limits of probabilistic analysis, and have presented several improvements upon the "basic" probabilistic bound (3.5). While these improvements are not dramatic, we have seen in examples that they can be substantial in practice. We point out that, in addition to the results presented in Section 3.3, we have investigated several other ideas. For example, one could construct a parity-check matrix for a linear code $\mathcal{C}$ by selecting each codeword of the dual code $\mathcal{C}^{\perp}$ independently with some probability p, and then optimize the value of p. However, we have found that while this method, as well as other probabilistic-choice variants, improve upon (3.5), they are generally *less* efficient than the bounds presented in Section 3.3. We note that well-known techniques in "probabilistic method," such as Lovász local lemma [3, p. 64] and Rödl nibble [3, p. 53], do not seem to be applicable in our context. Thus we believe that further improvements, if any, would require drastically new ideas.

Though the various probabilistic bounds are much stronger than their explicitly constructive counterparts, it is still difficult to say if they are tight bounds in general. Particularly, all upper bounds we obtained for the case of "good" codes grow exponentially with the length of the code. Although for some codes it is known [1] that this exponential growth is necessary, it remains an interesting question whether there exist "good" codes whose stopping redundancy grows only polynomially with block length.

Improving the general lower bound on stopping redundancy seems difficult. Straightforward application of the probabilistic methods yields the same bound as in [1].

Appendix 3.A Proof of Lemma 3.13

Proof: First assume that the rows of H_t are drawn from $\mathcal{C}^{\perp}$ uniformly at random *with* replacement (as in Theorem 3.8). It is known (see [17], for example) that

$$\Pr\{\operatorname{rank}(H_t) = j\} \;=\; \frac{1}{2^{(r-j)(t-j)}} \prod_{i=0}^{j-1} \frac{(1-2^{i-r})(1-2^{i-t})}{1-2^{i-j}} \tag{3.31}$$

for all $j = 0, 1, \ldots, r$. Basically, $\operatorname{rank}(H_t) = j$ if and only if H_t can be written as $H_t = AB$, where A is $t \times j$, B is $j \times n$ and both are of rank j. There are $(2^r - 1)(2^r - 2) \ldots (2^r - 2^{j-1})$ ways to choose rows of B from $\mathcal{C}^{\perp}$ such that $\operatorname{rank}(B) = j$. Similarly, there are $(2^t - 1)(2^t - 2) \ldots (2^t - 2^{j-1})$ ways to choose A such that $\operatorname{rank}(A) = j$. Now, for any $j \times j$ binary matrix C

that is full-rank, $H_t = (AC)(C^{-1}B)$. So, by choosing A and B in separate steps, each rank-j realization of H_t is counted $(2^j-1)(2^j-2)\ldots(2^j-2^{j-1})$ times. As there are a total of 2^{rt} possible realizations of H_t, we have

$$\Pr\{\mathrm{rank}(H_t)=j\} = \frac{1}{2^{rt}}\frac{\prod_{i=0}^{j-1}(2^r-2^i)\prod_{i=0}^{j-1}(2^t-2^i)}{\prod_{i=0}^{j-1}(2^j-2^i)}$$

and (3.31) follows.

Now, we prove (3.26). If $t \geq r$, we have

$$\begin{aligned}
\mathsf{E}[Y_t] &= \sum_{j=0}^{r}(r-j)2^{-(r-j)(t-j)}\prod_{i=0}^{j-1}\frac{(1-2^{i-r})(1-2^{i-t})}{1-2^{i-j}} \\
&\leq \sum_{j=0}^{r} j2^{-j(t-r+j)}\prod_{i=1}^{j}\frac{1}{1-2^{-i}} \\
&\leq 2^{-(t-r)} + \sum_{j=2}^{r} j2^{-j(t-r+j)}\prod_{i=1}^{j}\frac{1}{1-2^{-i}} \\
&\leq 2^{-(t-r)} + \frac{4}{3}\sum_{j=2}^{r}2^{-j(t-r+1)} &&(3.32) \\
&\leq 2^{-(t-r)} + \frac{4}{3}\frac{2^{-2(t-r+1)}}{1-2^{-(t-r+1)}} \\
&= 2^{-(t-r)}\left(1+\frac{2/3}{2^{t-r+1}-1}\right) &&(3.33)
\end{aligned}$$

where (3.32) follows from the fact that

$$j2^{-j^2}\prod_{i=1}^{j}\frac{1}{1-2^{-i}} \leq \frac{4}{3}2^{-j}, \quad \forall j \geq 2.$$

Similarly, if $t < r$, we have

$$\begin{aligned}
\mathsf{E}[Y_t] &= (r-t) + \mathsf{E}\big[t-\mathrm{rank}(H_t)\big] \\
&= (r-t) + \sum_{j=0}^{t}(t-j)2^{-(r-j)(t-j)}\prod_{i=0}^{j-1}\frac{(1-2^{i-r})(1-2^{i-t})}{1-2^{i-j}} \\
&\leq (r-t) + \sum_{j=0}^{t} j2^{-j(r-t+j)}\prod_{i=1}^{j}\frac{1}{1-2^{-i}} \\
&\leq (r-t) + 2^{-(r-t)}\left(1+\frac{2/3}{2^{r-t+1}-1}\right).
\end{aligned}$$

For the special case where $t = r$, equation (3.33) shows that $\mathsf{E}[Y_r] \leq 5/3$. We show that this upper bound can be improved as follows.

$$
\begin{aligned}
\mathsf{E}[Y_r] &= \sum_{j=0}^{r} (r-j) 2^{-(r-j)^2} \prod_{i=0}^{j-1} \frac{(1-2^{i-r})^2}{1-2^{i-j}} \\
&= \sum_{j=1}^{r} j 2^{-j^2} \prod_{i=0}^{r-j-1} \frac{(1-2^{i-r})^2}{1-2^{i-(r-j)}} \\
&= \sum_{j=1}^{r} j 2^{-j^2} \frac{\phi(1/2)}{\prod_{i=1}^{j}(1-2^{-i})^2} \frac{\prod_{i=r-j+1}^{r}(1-2^{-i})}{\prod_{i=r+1}^{\infty}(1-2^{-i})} \\
&\leq \phi(1/2) \sum_{j=1}^{r} \frac{j 2^{-j^2}}{\prod_{i=1}^{j}(1-2^{-i})^2} \qquad (3.34) \\
&\leq \phi(1/2) \left(\frac{1298}{441} + \sum_{j=4}^{r} \frac{j 2^{-j^2}}{\prod_{i=1}^{j}(1-2^{-i})^2} \right) \\
&\leq \phi(1/2) \left(\frac{1298}{441} + \frac{1}{4} \sum_{j=4}^{r} 2^{-2j} \right) \qquad (3.35) \\
&\leq 0.29 \left(\frac{1298}{441} + \frac{1}{768} \right) \\
&< \frac{6}{7},
\end{aligned}
$$

where $\phi(q) \overset{\text{def}}{=} \prod_{k=1}^{\infty}(1-q^k)$ is the Euler's function. Note that (3.34) is obtained by noting

$$\prod_{i=r+1}^{\infty} (1-2^{-i}) \geq 1-2^{-r},$$

and (3.35) is based on the fact that

$$\frac{j 2^{-j^2}}{\prod_{i=1}^{j}(1-2^{-i})^2} \leq 2^{-2j-2}, \quad \forall j \geq 4.$$

Finally, we show that if the rows of H_t are drawn uniformly at random from $\mathcal{C}^{\perp} \setminus \{\mathbf{0}\}$ *without* replacement (as in Theorem 3.10), this can only reduce the expected value of Y_t.

This result is intuitively clear, since when all-zero rows and duplicate rows are avoided, the expected rank of H_t can only be higher. For a formal proof, let H_t' be a $t \times n$ matrix whose rows are drawn at random from $\mathcal{C}^{\perp} \setminus \{\mathbf{0}\}$ without replacement. Let H_t still denote the $t \times n$

matrix whose rows are drawn independently and uniformly from $\mathcal{C}^{\perp}$ with replacement. It suffices to show $\mathsf{E}\big[\mathrm{rank}(H_t')\big] \geq \mathsf{E}\big[\mathrm{rank}(H_t)\big]$.

Consider the following random experiment:

```
$H \leftarrow$ empty matrix; $H' \leftarrow$ empty matrix; $\mathcal{H} \leftarrow \{\mathbf{0}\}$;
for $i = 1$ to $t$
    $\boldsymbol{h} \leftarrow$ random vector from $\mathcal{C}^{\perp}$; $H \leftarrow \begin{bmatrix} H \\ \boldsymbol{h} \end{bmatrix}$;
    while $\boldsymbol{h} \in \mathcal{H}$
        $\boldsymbol{h} \leftarrow$ random vector from $\mathcal{C}^{\perp}$;
    end
    $H' \leftarrow \begin{bmatrix} H' \\ \boldsymbol{h} \end{bmatrix}$; $\mathcal{H} \leftarrow \mathcal{H} \cup \{\boldsymbol{h}\}$;
end
```

Clearly, at the end of the experiment, H has the same distribution as H_t and H' has the same distribution as H_t'. Effectively, the above experiment describes a way of obtaining the distribution of H_t from the distribution of H_t'. In other words, it defines a stochastic matrix M, such that

$$\boldsymbol{p}' = \boldsymbol{p}M,$$

where $\boldsymbol{p} = \big(\Pr\{H_t = \omega\}\big)_{\omega\in\Omega}$, $\boldsymbol{p}' = \big(\Pr\{H_t' = \omega'\}\big)_{\omega'\in\Omega'}$, and Ω and Ω' are the sample spaces for H_t and H_t', respectively. By construction of the random experiment, $M_{\omega,\omega'} > 0$ only if $\mathrm{rank}(\omega) \leq \mathrm{rank}(\omega')$. Hence,

$$\begin{aligned}
\mathsf{E}\big[\mathrm{rank}(H_t')\big] &= \sum_{\omega'\in\Omega'} \Pr\{H_t' = \omega'\}\,\mathrm{rank}(\omega') \\
&= \sum_{\omega\in\Omega} \Pr\{H_t = \omega\} \sum_{\omega'\in\Omega'} M_{\omega,\omega'}\,\mathrm{rank}(\omega') \\
&\geq \sum_{\omega\in\Omega} \Pr\{H_t = \omega\} \sum_{\omega'\in\Omega'} M_{\omega,\omega'}\,\mathrm{rank}(\omega) \\
&= \sum_{\omega\in\Omega} \Pr\{H_t = \omega\}\,\mathrm{rank}(\omega) \\
&= \mathsf{E}\big[\mathrm{rank}(H_t)\big].
\end{aligned}$$

■

Acknowledgment

This chapter is in part a reprint of the material in the papers: J. Han and P. H. Siegel, “Improved upper bounds on stopping redundancy,” *IEEE Trans. Inform. Theory*, vol. 53, no. 1, pp. 90–104, Jan. 2007, and J. Han, P. H. Siegel, and A. Vardy, “Improved probabilistic bounds on stopping redundancy,” *IEEE Trans. Inform. Theory*, vol. 54, no. 4, pp. 1749–1753, Apr. 2008. The dissertation author was the primary author of the above papers.

Bibliography

[1] M. Schwartz and A. Vardy, “On the stopping distance and the stopping redundancy of codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.

[2] P. Erdős and J. H. Spencer, *Probabilistic Methods in Combinatorics*. Academic Press, 1974.

[3] N. Alon and J. H. Spencer, *The Probabilistic Method*. New York: John Wiley & Sons, 1991.

[4] M. Schwartz and A. Vardy, “On the stopping distance and stopping redundancy of codes,” in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 975–979.

[5] H. D. L. Hollmann and L. M. G. M. Tolhuizen, “Generating parity check equations for bounded-distance iterative erasure decoding of even weight codes,” in *Proc. 27th Symposium on Information Theory in the Benelux*, Noordwijk, The Netherlands, June 8–9, 2006, pp. 17–24.

[6] ——, “Generating parity check equations for bounded-distance iterative erasure decoding,” in *Proc. IEEE International Symposium on Information Theory*, Seattle, WA, July 2006, pp. 514–517.

[7] ——, “Generic erasure-correcting sets: bounds and constructions,” *J. Combin. Theory Ser. A*, vol. 113, pp. 1746–1759, Nov. 2006.

[8] ——, “On parity check collections for iterative erasure decoding that correct all correctable erasure patterns of a given size,” *IEEE Trans. Inform. Theory*, vol. 53, no. 2, pp. 823–828, Feb. 2007.

[9] T. Hehn, S. Laendner, O. Milenkovic, and J. B. Huber, “The stopping redundancy hierarchy of cyclic codes,” in *Proc. 44-th Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, Sept. 2006, pp. 1271–1280.

[10] T. Hehn, O. Milenkovic, S. Laendner, and J. B. Huber, “Permutation decoding and stopping redundancy hierarchy of linear block codes,” in *Proc. IEEE International Symposium on Information Theory*, Nice, France, June 2007, pp. 2926–2930.

[11] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1978.

[12] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge: M.I.T. Press, 1963.

[13] M. Grassl. Bounds on the minimum distance of linear codes. Accessed on July 12, 2007. [Online]. Available: http://www.codetables.de

[14] T. Etzion, “On the stopping redundancy of Reed-Muller codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 4867–4879, Nov. 2006.

[15] A. S. Hedayat, N. J. A. Sloane, and J. Stufken, *Orthogonal Arrays*. New York: Springer, 1999.

[16] O. Milenkovic, E. Soljanin, and P. Whiting, “Stopping and trapping sets in generalized covering arrays,” in *Proc. 40th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, Mar. 2006, pp. 259–264.

[17] I. N. Kovalenko, “On the limit distribution of the number of solutions of a random system of linear equations in the class of boolean functions,” *Theory Probab. Appl.*, vol. 12, pp. 47–56, 1967.

Chapter 4

Stopping Redundancy of MDS Codes

4.1 Introduction

A code is *maximum distance separable (MDS)*[1] if it achieves the Singleton bound [1]. Hence, an $[n, n-r, d]$ code is MDS if and only if $d = r+1$. It turns out that very interesting things can be said about the stopping redundancy of MDS codes, which we study in detail in this chapter.

For the rest of the chapter, let $\mathcal{C}$ be an $[n, k = n-d+1, d]_q$ MDS code. In [2], it was shown that for all $d \geq 3$,

$$\frac{1}{d-1}\binom{n}{d-2} < \rho(\mathcal{C}) \leq \frac{\max\{d^{\perp}, d-1\}}{n}\binom{n}{d-2}, \tag{4.1}$$

where

$$d^{\perp} = n - d + 2$$

is the minimum distance of $\mathcal{C}^{\perp}$, the dual code of $\mathcal{C}$. The authors of [2] then made an intriguing conjecture that $\rho(\mathcal{C})$ should in fact be a function of just n and d, independent of the actual code.

We show that the upper bound on $\rho(\mathcal{C})$ can be improved by exploiting its relation to a new combinatorial quantity, the *single-exclusion number*, which will be introduced shortly. Before doing so, we first recall two related, well-studied combinatorial constructs. For positive integers $n \geq s \geq t$, and an n-set[2]N, an (n, s, t)-*Turán system* (also known as *Turán* (n, s, t)-

[1] For this thesis, an MDS code always refers to a *linear* MDS code, unless otherwise noted.

[2] An *n-set* is a set that contains n elements. Similarly, if A is any set, then a *t-subset* of A is a subset of A that contains t elements.

system) [3] is a collection of t-subsets of N, called *blocks*, such that each s-subset of the n-set contains at least one block. The (n, s, t)-*Turán number*, denoted hereafter by $T(n, s, t)$, is the smallest number of blocks in an (n, s, t)-Turán system. A concept "dual" to that of a Turán system is that of a covering design [4]. Specifically, for $n \geq s \geq t$ and an n-set N, an (n, s, t)-*covering design* is a collection of s-subsets of N, also called *blocks*, such that each t-subset of the n-set is contained in at least one block. The (n, s, t)-*covering number*, denoted by $C(n, s, t)$, is the smallest number of blocks in an (n, s, t)-covering design. Clearly, by definition,

$$T(n, s, t) = C(n, n-t, n-s).$$

The stopping redundancy of an MDS code is closely related to covering / Turán numbers. In fact, as observed in [2], suppose H is a parity-check matrix for $\mathcal{C}$ and $s(H) = d$. Since $\mathcal{C}^{\perp}$ is also MDS and hence has minimum distance $n-d+2$, a $(d-1)$-set is a stopping set unless its complement is contained in the support of some row of H that has weight $n - d + 2$. Since there are no stopping sets of size $d - 1$, the supports of minimum-weight rows of H form an $(n, n-d+2, n-d+1)$-covering design (equivalently, the complements of supports form an $(n, d-1, d-2)$-Turán system). Therefore,

$$\rho(\mathcal{C}) \geq C(n, n-d+2, n-d+1) = T(n, d-1, d-2). \tag{4.2}$$

The lower bound in (4.1) was shown as a consequence of the above fact [2].

We now introduce the notions of single-exclusion system and single-exclusion number.

Definition 4.1 For an n-set N and $t < n$, an (n, t)-*single-exclusion (SE) system* is a collection of t-subsets of N, called *blocks*, such that for each i-subset of N, $i = 1, \ldots, t+1$, there exists at least one block that contains all but one element from the i-subset. The (n, t)-*single-exclusion (SE) number*, $S(n, t)$, is the smallest number of blocks in an (n, t)-SE system.

Note that an (n, t)-SE system is a special type of $(n, t+1, t)$-Turán system. Its relevance to the stopping redundancy of MDS codes is given in the following theorem.

Theorem 4.1 *Let $\mathcal{C}$ be an $[n, n-d+1, d]$ code. Then*

$$\rho(\mathcal{C}) \leq S(n, d-2).$$

Proof: Let $\mathcal{S} \subseteq [n]^{d-2}$ be an $(n, d-2)$-SE system of smallest size. Hence, $|\mathcal{S}| = S(n, d-2)$. Since $\mathcal{C}^{\perp}$ is MDS with minimum distance $n-d+2$, for each $S \in \mathcal{S}$, we can find $\boldsymbol{h} \in \mathcal{C}^{\perp}$, such that $\mathrm{supp}(\boldsymbol{h}) = [n] \setminus S$. Choose one such $\boldsymbol{h}$ for each element of $\mathcal{S}$, and use these vectors as rows to form an $S(n, d-2) \times n$ matrix H. We show that $s(H) = d$. Indeed, for each $i \in \{1, \ldots, d-1\}$, and $X \in [n]^i$, by definition there exists $S \in \mathcal{S}$, such that $|X \setminus S| = 1$, i.e. $|X \cap ([n] \setminus S)| = 1$. By the construction of H, there exists one row $\boldsymbol{h}$, such that $\mathrm{supp}(\boldsymbol{h}) = [n] \setminus S$. Hence, $|X \cap \mathrm{supp}(\boldsymbol{h})| = 1$, so X is not a stopping set. Finally, $s(H) = d$ implies that up to row and column permutations, H contains a $(d-1) \times (d-1)$ upper triangular submatrix. Therefore, $\mathrm{rank}(H) = d - 1 = r$. ■

With (4.2) (cf. [2, equation (24)]) and Theorem 4.1, we have "sandwiched" $\rho(\mathcal{C})$ in between two combinatorial quantities that only depend on n and d. Namely,

$$T(n, d-1, d-2) \leq \rho(\mathcal{C}) \leq S(n, d-2). \tag{4.3}$$

Clearly, any upper bound on $S(n, d-2)$ is also an upper bound on $\rho(\mathcal{C})$. This reduction proves useful, as it allows us to invoke powerful combinatorial tools in tackling the problem. We will see that in some cases, the upper and lower bounds become so close that definitive conclusions may be drawn about $\rho(\mathcal{C})$, an example of which is given below, which also serves to show why a tighter upper bound on $\rho(\mathcal{C})$ is hopeful.

Theorem 4.2 *For all fixed $d > 1$, as $n \to \infty$,*

$$S(n, d-2) = T(n, d-1, d-2)\big(1 + O(n^{-1})\big).$$

Proof: Let $\mathcal{T}$ be an $(n, d-1, d-2)$-Turán system of smallest size. WLOG, let $[n]$ be the n-set. We show that no more than $O(n^{d-3})$ blocks need to be added to $\mathcal{T}$ to make it an $(n, d-2)$-SE system.

Let $\mathcal{T}' = \{Z \in [n]^{d-2} : Z \cap [d-2] \neq \emptyset\}$. Let $\mathcal{S} = \mathcal{T} \cup \mathcal{T}'$. Clearly,

$$|\mathcal{T}'| = \sum_{m=0}^{d-3} \binom{d-2}{d-2-m} \binom{n-d+2}{m} = O\big(n^{d-3}\big).$$

For any $X \subseteq [n]$ such that $1 \leq |X| \leq d-1$, if $|X| = d-1$, then by definition of Turán system, there exists $Y \in \mathcal{T}$, such that $|X \setminus Y| = 1$. Otherwise, if $|X| < d-1$, define

$$Y = \big(X \setminus \{x\}\big) \cup \Big\{ \big(d - |X| - 1\big) \text{ smallest elements of } \big([n] \setminus X\big) \Big\},$$

where $x \in X$ is an arbitrary element. It is easy to verify that $Y \in \mathcal{T}'$ and $|X \setminus Y| = 1$. Hence, $\mathcal{S}$ is an $(n, d-2)$-SE system with $T(n, d-1, d-2) + O\big(n^{d-3}\big)$ blocks. Finally, note that

$$\frac{1}{d-1}\binom{n}{d-2} \leq T(n, d-1, d-2) \leq \binom{n}{d-2}.$$

Thus $T(n, d-1, d-2) = \Theta(n^{d-2})$, and the result follows. ■

Katona, Nemetz and Simonovits [5] showed that $T(n,s,t)/\binom{n}{t}$ is non-decreasing in n and hence there exists the limit

$$\tau(s,t) \overset{\text{def}}{=} \lim_{n\to\infty} \frac{T(n,s,t)}{\binom{n}{t}}.$$

By Theorem 4.2 and (4.3), for any fixed $d > 1$, both $S(n, d-2)$ and $\rho(\mathcal{C})$ are asymptotic to[3] $T(n, d-1, d-2)$ and hence asymptotic to

$$\tau(d-1, d-2) \cdot \binom{n}{d-2}.$$

The value of $\tau(t+1, t)$, although unknown for $t > 2$, is well-studied. In fact, the determination of $\tau(s,t)$ for $s > t > 2$ has been one of the most challenging open problems in combinatorial theory (for the solution of which Erdős offered a $1000 award; see [6]). Some of the known bounds on $\tau(t+1,t)$ are summarized in Table 4.1 (cf. [7], [8], [3], [9], [10], [11], [12], [13]).

In comparison to Table 4.1, note that the upper bound in (4.1) is never better than

$$\frac{1}{2}\binom{n}{d-2}.$$

This suggests room for improvement in the upper bound, at least for the case where d is much smaller than n.

[3] Functions $f(x)$ and $g(x)$ are said to be *asymptotic to each other as* $x \to x_0$ if $\lim_{x\to x_0} \frac{f(x)}{g(x)} = 1$, and is denoted by $f(x) \sim g(x)$. In this thesis we usually talk about integer functions of n and the condition $n \to \infty$ is sometimes omitted where there is no confusion.

Table 4.1 Some known bounds on $\tau(t+1,t)$

t	Lower Bound	Upper Bound
2	$\frac{1}{2}$	$\frac{1}{2}$
3	$\frac{9-\sqrt{17}}{12}$	$\frac{4}{9}$
4	$\frac{37}{143}$	$\frac{5}{16}$
5	$\frac{37-\sqrt{345}}{80}$	$\frac{5}{16}$
6	$\frac{1}{6}$	$\frac{17}{64}$
asymp.	$\frac{1}{t}$	$\left(\frac{1}{2}+o(1)\right)\frac{\ln t}{t}$

We show that it's indeed true that much stronger upper bounds can be obtained by studying the SE number $S(n,t)$. Consequently, the new upper bounds are used to show that as $n \to \infty$, $\rho(\mathcal{C})$ is asymptotic to $S(n,d-2)$ if $d = o(\sqrt{n})$, or if $k = o(\sqrt{n}) \to \infty$.[4] Hence, in an asymptotic sense, the Schwartz-Vardy conjecture is proved in these cases. In addition, for $1 < d \leq 5$, we show that $\rho(\mathcal{C}) = S(n,d-2)$ holds exactly. We conjecture that $\rho(\mathcal{C}) = S(n,d-2)$ is a general result for all MDS codes.

The rest of the chapter is arranged as follows. Section 4.2 is devoted to upper bounds on $S(n,t)$. The upper bounds are obtained through three approaches: combinatorial constructions, probabilistic methods, and recurrent inequalities. Interesting asymptotics are observed and discussed, some of which effectively confirm the Schwartz-Vardy conjecture in an asymptotic sense. The (computable) upper bounds are compared numerically for n up to 512, and the best bounds are found. In Section 4.3, we derive a lower bound on $S(n,t)$ and observe its implications. Section 4.4 focuses on the Schwartz-Vardy conjecture. We prove the conjecture for all MDS codes with $1 < d \leq 5$, and comment on why $S(n,d-2)$ may be a reasonable estimate of $\rho(\mathcal{C})$. Section 4.5 concludes the chapter.

[4] We adopt the standard "O-notation" and related asymptotic expressions [14, Ch. 9]. Functions $f(n)$ and $g(n)$ are said to be *asymptotic* to each other, denoted by $f(n) \sim g(n)$, if $\lim_{n\to\infty} f(n)/g(n) = 1$, or equivalently, if $f(n) = \big(1+o(1)\big)g(n)$, where $o(1)$ stands for any function that goes to zero as n goes to infinity. More generally, we write $f(n) = o\big(g(n)\big)$ if $\lim_{n\to\infty} f(n)/g(n) = 0$.

4.2 Upper Bounds on $S(n,t)$

We start with some preliminaries. For any set A, let $[A]^i$ denote the set of all i-subsets of A. With some abuse of notation, we say that a set A *covers* a set B if $|B \setminus A| = 1$. (Note that previously we have defined that a matrix H *covers* a set X if $H(X)$ contains a row of weight one.) Hence, if N is an n-set, then $\mathcal{S} \subseteq [N]^t$ is an (n,t)-SE system if and only if for each $i = 1, \ldots, t+1$ and $X \in [N]^i$, there exists a block in $\mathcal{S}$ that covers X. A covering design/Turán system/SE system (with prescribed parameters) is *minimal* if it contains the least number of blocks.

By definition, an (n,t)-SE system is also an $(n,t+1,t)$-Turán system. Hence, we have

$$S(n,t) \geq T(n,t+1,t) \geq \frac{1}{n-t}\binom{n}{t+1} = \frac{1}{t+1}\binom{n}{t}, \tag{4.4}$$

where the second inequality [5] follows by noting that each block in the Turán system is contained in $(n-t)$ distinct $(t+1)$-subsets.

Let $\mathcal{C}$ be an $[n, k = n-d+1, d]$ MDS code. Note that (4.3) can be written as

$$T(n, n-k, n-k-1) \leq \rho(\mathcal{C}) \leq S(n, n-k-1).$$

Occasionally, when it's convenient to do so, we will state our results in terms of $S(n, n-k-1)$.

4.2.1 Probabilistic Bounds

Let N be an n-set. Consider the following random experiment in which we build an (n,t)-SE system, $\mathcal{S} \subseteq [N]^t$. In the first step, for a prescribed real value $p \in [0,1]$, insert into $\mathcal{S}$ each element of $[N]^t$ with probability p. The expected size of $\mathcal{S}$ at this point is $p\binom{n}{t}$, but some i-subsets, $i = 1, \ldots, t+1$, may not be covered. The probability that a given i-subset is not covered equals $(1-p)^{\varphi(n,t,i)}$, where

$$\varphi(n,t,i) = i\binom{n-i}{t-i+1} = i\binom{n-i}{n-t-1}.$$

So, as a second step, for each $X \in [N]^i$, $i = 1, \ldots, t+1$, that is not yet covered, insert into $\mathcal{S}$ some element of $[N]^t$ that covers X. The expected size of $\mathcal{S}$ is then bounded from above by

$$p\binom{n}{t} + \sum_{i=1}^{t+1} \binom{n}{i} (1-p)^{\varphi(n,t,i)}.$$

This implies the following upper bound on $S(n,t)$.

Theorem 4.3 *For all* $0 \leq p \leq 1$*,*

$$S(n,t) \leq p\binom{n}{t} + \sum_{i=1}^{t+1} \binom{n}{i} (1-p)^{i\binom{n-i}{t-i+1}}. \tag{4.5}$$

Alternatively, in the first step of the random experiment, we may instead make l random drawings from $[N]^t$. At the end of the first step, the probability that a given i-subset is not covered equals

$$\left(1 - \frac{\varphi(n,t,i)}{\binom{n}{t}}\right)^l$$

if the drawing is done with replacement, and equals

$$\prod_{j=0}^{l-1} \left(1 - \frac{\varphi(n,t,i)}{\binom{n}{t} - j}\right)^+,$$

where $(x)^+ \overset{\text{def}}{=} \max\{x, 0\}$, if the drawing is done without replacement. The results are the following bounds.

Theorem 4.4 *For all* $l \in \mathbb{N}$*,*

$$S(n,t) \leq l + \sum_{i=1}^{t+1} \binom{n}{i} \left(1 - \frac{i\binom{n-i}{t-i+1}}{\binom{n}{t}}\right)^l. \tag{4.6}$$

Theorem 4.5 *For all* $l \in \mathbb{N}$*,* $l \leq \binom{n}{t}$*,*

$$S(n,t) \leq l + \sum_{i=1}^{t+1} \binom{n}{i} \prod_{j=0}^{l-1} \left(1 - \frac{i\binom{n-i}{t-i+1}}{\binom{n}{t} - j}\right)^+. \tag{4.7}$$

Theorem 4.5 is clearly stronger than Theorem 4.4, and is closely related to Theorem 4.3. In fact, since for all $x \in \mathbb{R}$, $m \in \mathbb{N}$, $x > m > 0$,

$$\begin{aligned}\sum_{i=0}^{m} \frac{1}{x-i} &= \frac{1}{x}\sum_{i=0}^{m}\sum_{j=0}^{\infty}\left(\frac{i}{x}\right)^j = \frac{1}{x}\sum_{j=0}^{\infty} x^{-j} \sum_{i=0}^{m} i^j \\ &\geq \frac{1}{x}\sum_{j=0}^{\infty} x^{-j} \cdot \frac{m^{j+1}}{j+1} = \sum_{j=0}^{\infty} \frac{1}{j+1}\left(\frac{m}{x}\right)^{j+1} \\ &= -\ln\left(1 - \frac{m}{x}\right),\end{aligned}$$

we have, for $l \leq \binom{n}{t}$,

$$\begin{aligned}\prod_{j=0}^{l-1}\left(1-\frac{i\binom{n-i}{r-i+1}}{\binom{n}{t}-j}\right)^{+} &= \exp\left(\sum_{j=0}^{l-1}\ln\left(1-\frac{i\binom{n-i}{t-i+1}}{\binom{n}{t}-j}\right)^{+}\right)\\ &\leq \exp\left(-\sum_{j=0}^{l-1}\frac{i\binom{n-i}{t-i+1}}{\binom{n}{t}-j}\right)\\ &\leq \exp\left(i\binom{n-i}{t-i+1}\ln\left(1-\frac{l-1}{\binom{n}{t}}\right)\right)\\ &= \left(1-\frac{l-1}{\binom{n}{t}}\right)^{i\binom{n-i}{t-i+1}}.\end{aligned}$$

So the upper bound (4.7), when minimized over l, is no greater than

$$\min_{\substack{l\in\mathbb{Z}\\ 0\leq l\leq\binom{n}{t}}}\left\{l+\sum_{i=1}^{t+1}\binom{n}{i}\left(1-\frac{l}{\binom{n}{t}}\right)^{\varphi(n,t,i)}\right\}+1.$$

Note that we have strategically allowed l to take the value $\binom{n}{t}$ in the above expression.

On the other hand, letting $l = p\binom{n}{t}$ in (4.5), we see the minimum value of the upper bound (4.5) is

$$\min_{\substack{l\in\mathbb{R}\\ 0\leq l\leq\binom{n}{t}}}\left\{l+\sum_{i=1}^{t+1}\binom{n}{i}\left(1-\frac{l}{\binom{n}{t}}\right)^{\varphi(n,t,i)}\right\}.$$

Now, suppose the minimum value of the above expression is y, achieved at $l = l^*$. Then its value at $l = \lceil l^* \rceil$ is less than $y + 1$. Therefore, we conclude that the upper bound (4.7) (when minimized over l) is less than the upper bound (4.5) (when minimized over p) plus two. In practice, the difference between the two bounds is very small, while the upper bound in (4.5) is usually easier to compute.

The upper bound in (4.5) can be written as

$$\left(p+(1-p)^{t+1}\frac{n-t}{t+1}\right)\binom{n}{t}+\sum_{i=1}^{t}(1-p)^{\varphi(n,t,i)}\binom{n}{i}. \tag{4.8}$$

The first term in (4.8) is minimized when p takes the value

$$p_{\min} = 1-(n-t)^{-1/t},$$

in which case (4.8) becomes

$$\big(1-(n-t)^{-1/t}+\eta(n,t)\big)\cdot\binom{n}{t}, \tag{4.9}$$

where

$$\begin{aligned}\eta(n,t) &= \binom{n}{t}^{-1}\cdot\sum_{i=1}^{t+1}(1-p_{\min})^{\varphi(n,t,i)}\binom{n}{i}\\ &= \frac{n-t}{t+1}\cdot\sum_{i=1}^{t+1}\Big((n-t)^{-\frac{i}{t}\binom{n-i}{n-t-1}}\Big)\frac{\binom{t+1}{i}}{\binom{n-i}{n-t-1}}.\end{aligned} \tag{4.10}$$

As $n\to\infty$, it can be shown that (see Appendix Appendix 4.A) if $t<n-\ln n$, then the term corresponding to $i=t+1$ prevails in the sum (4.10). Therefore,

$$\eta(n,t)=\big(1+o(1)\big)\cdot\frac{(n-t)^{-1/t}}{t+1}.$$

Plugging the above into (4.9), we conclude that as $n\to\infty$, the bound (4.9) is[5]

$$\begin{cases}\big(1+O(n^{-1/t})\big)\binom{n}{t}, & \text{if } t\prec\ln n\\ \big(1-e^{-1/c}+o(1)\big)\binom{n}{t}, & \text{if } t=\big(c+o(1)\big)\ln n\\ \Big(1+O\big(\frac{\ln(n-t)}{t}\big)\Big)\frac{\ln(n-t)}{t}\binom{n}{t}, & \text{if } \ln n\prec t<n-\ln n\end{cases}$$

where $c>0$ is any constant. By the choice of $p_{\min}$, and the fact that for $p=p_{\min}$ the $(t+1)$-st term prevails in (4.10), no other values of p give asymptotically tighter bounds than the above.

For the case when $t\geq n-\ln n$, a different asymptotic analysis (see Appendix Appendix 4.B) shows that the bound (4.5), when minimized over p, is $O\big(\frac{\ln n}{n}\binom{n}{t}\big)$ for all t such that $2<n-t=o\big((n\ln\ln n)/\ln n\big)$, and is, particularly, $\Theta\big(\frac{\ln n}{n}\binom{n}{t}\big)$ for all $t=n-\Theta(1)$.

4.2.2 Constructive Bounds

Kim and Roush [15] proposed the following construction for Turán $(n,t+1,t)$-systems.

Construction 4.1 ([15]) Let N be an n-set. For a prescribed positive integer l, partition N into l subsets, N_i, $i=0,\ldots,l-1$, as equally as possible. Thus, $N=\bigcup_{i=0}^{l-1}N_i$, such that

[5] We write $f(n)\prec g(n)$, if $f(n)=o\big(g(n)\big)$, and similarly, $f(n)\succ g(n)$, if $g(n)=o\big(f(n)\big)$, as $n\to\infty$.

$\lfloor n/l \rfloor \leq |N_i| \leq \lceil n/l \rceil$ for all i. For all $X \subseteq N$, define

$$w(X) \overset{\text{def}}{=} \sum_{i=0}^{l-1} i|X \cap N_i|.$$

For all $j = 0, 1, \ldots, l-1$, let

$$\mathcal{K}_j = \mathcal{E} \cup \mathcal{W}_j,$$

where

$$\begin{aligned} \mathcal{E} &= \big\{X \in [N]^t : \exists i, X \cap N_i = \emptyset\big\}, \\ \mathcal{W}_j &= \big\{X \in [N]^t : w(X) \equiv j \mod l\big\}. \end{aligned} \tag{4.11}$$

We observe that $\mathcal{K}_j$ in Construction 4.1 is an (n,t)-SE system as long as $l \geq n/(n-t-1)$.

Theorem 4.6 *For all j, $\mathcal{K}_j$ as defined in Construction 4.1 is an (n,t)-SE system if $l \geq n/(n-t-1)$.*

Proof: We show that for any $X \subseteq N$ such that $1 \leq |X| \leq t+1$, there exists $Y \in \mathcal{K}_j$ such that $|X \setminus Y| = 1$.

If there exists i such that $X \cap N_i = \emptyset$, we'd like to find a superset $Z \supseteq X$ of size $t+1$, such that $Z \cap N_i = \emptyset$ as well. This is possible as long as $n - |N_i| \geq t+1$, which is guaranteed in our case by $l \geq n/(n-t-1)$. Now, let $x \in X$ be an arbitrary element and let $Y = Z \setminus \{x\}$. Then $|X \setminus Y| = 1$, and $Y \in \mathcal{K}_j$.

Otherwise, $X \cap N_i \neq \emptyset$ for all i. Let $x_i \in X \cap N_i$ be arbitrarily chosen. Let Z be any superset of X such that $|Z| = t+1$. For all i, let $Y_i = Z \setminus \{x_i\}$. Since

$$w(Y_i) = w(Z) - i, \quad \forall i = 0, \ldots, l-1,$$

by choosing i we can realize all possible values of $\big(w(Y_i) \mod l\big)$. Therefore, for any j, there exists i such that $Y_i \in \mathcal{K}_j$. By construction, $|X \setminus Y_i| = |\{x_i\}| = 1$. ■

As in [15], the smallest number of blocks in $\mathcal{K}_j$ can be estimated as follows.

$$\begin{aligned} \min_{0 \leq j \leq l-1} |\mathcal{K}_j| &\leq |\mathcal{E}| + \min_{0 \leq j \leq l-1} |\mathcal{W}_j| \\ &\leq \sum_{i=0}^{l-1} \Big|\big\{X \in [N]^t : X \cap N_i = \emptyset\big\}\Big| + \frac{1}{l}\binom{n}{t} \\ &\leq \big(l - R_l(n)\big)\binom{n - \lfloor \frac{n}{l} \rfloor}{t} + R_l(n)\binom{n - \lfloor \frac{n}{l} \rfloor - 1}{t} + \frac{1}{l}\binom{n}{t} \qquad (4.12) \\ &\leq l\binom{n - \lfloor \frac{n}{l} \rfloor}{t} + \frac{1}{l}\binom{n}{t}, \end{aligned}$$

where $R_l(n) = n \mod l$. Hence, we have the following upper bound on $S(n,t)$.

Theorem 4.7 *For all integers $l \geq n/(n-t-1)$,*

$$S(n,t) \leq l\binom{n - \lfloor \frac{n}{l} \rfloor}{t} + \frac{1}{l}\binom{n}{t}. \tag{4.13}$$

Let's look at the asymptotic behavior of the above bound as $n \to \infty$. Consider the following cases.

1. $t > 1$ *is fixed:*

   By choosing $l = \lceil t/(2\ln t) \rceil$, one can show that the upper bound (4.13) is asymptotically at most $\frac{1+2\ln t}{t}\binom{n}{t}$. This is stronger than (4.1) for all $d - 2 = t > 3$.

2. $t/n = \delta < 1$ *is fixed:*

   Choosing $l = \lceil t/(2\ln t) \rceil$, we see that the upper bound (4.13) is $O\big(\frac{\ln n}{n}\binom{n}{t}\big)$. At $t = d-2$, this is tighter than the $\Theta\big(\binom{n}{d-2}\big)$ upper bound given by (4.1). Note that the corresponding lower bound in (4.1) is $\Theta\big(\frac{1}{n}\binom{n}{d-2}\big)$.

3. $n - t - 1 = k > 3$ *is fixed:*

   Theorem 4.7 requires that $l \geq n/k$. Since $k > 3$, we can choose $l = \lfloor n/3 \rfloor$ (when n is large enough). Then bound (4.13) becomes, asymptotically,

   $$\begin{aligned} S(n, n-k-1) &\leq l\binom{n - \lfloor \frac{n}{l} \rfloor}{n-k-1} + \frac{1}{l}\binom{n}{n-k-1} \\ &\leq \left\lfloor \frac{n}{3} \right\rfloor \binom{n-3}{k-2} + \frac{3}{n-3}\binom{n}{k+1} \\ &= O(n^{k-1}) + \frac{3}{n-3}\binom{n}{k+1} \\ &= O(n^{k-1}) + \frac{3}{k+1}\binom{n}{k}, \end{aligned}$$

   which is asymptotic to $\frac{3}{k+1}\binom{n}{k}$. In comparison, at $d = n-k+1$, the upper bound in (4.1) is asymptotic to $\binom{n}{k+1}$, while the lower bound is $\frac{1}{k+1}\binom{n}{k}$.

The last part of the discussion above is interesting because the upper bound becomes asymptotically very close to the lower bound (up to a factor of 3). We summarize this result in the following theorem.

Theorem 4.8 *For all fixed k, as $n \to \infty$,*

$$S(n, n-k-1) \leq \left(1 + O(n^{-1})\right) \cdot \frac{3}{k+1}\binom{n}{k}.$$

Proof: We have just seen the upper bound holds for all $k \geq 4$. For $k = 3$, by (4.12),

$$S(n, n-k-1) \leq \left(l - R_l(n)\right)\binom{n - \lfloor \frac{n}{l} \rfloor}{n-k-1} + R_l(n)\binom{n - \lfloor \frac{n}{l} \rfloor - 1}{n-k-1} + \frac{1}{l}\binom{n}{n-k-1}.$$

Choosing $l = \lceil n/3 \rceil$ gives the desired result. (Note in this case $l - R_l(n) < 3$ if $3 \nmid n$.)

For $k = 2$, we show that we can construct an $(n, n-3)$-SE system using less than $\frac{2}{3}\binom{n}{2}$ blocks. Let $n = 3a + b$, $b = 0, 1, 2$. Consider the n-set $N = ([a] \times \{0, 1, 2\}) \cup (\{a+1\} \times \{0, \ldots, b-1\})$. Choose as blocks the complements of the following triples (if they exist in N) to construct $\mathcal{S}$:

1. $\{(x, 0), (x, 1), (x, 2)\}$, for $x = 1, \ldots, a$;

2. $\{(x, i), (y, i), (y, i{+}1)\}$ and $\{(x, i), (x, i{+}1), (y, i)\}$, for $x, y \in [a{+}1]$, $x < y$, $i = 0, 1, 2$;

3. $\{(x, 0), (x, 2), (a+1, 0)\}$, for $x = 1, \ldots, a$, if $b > 0$.

(In the above, $i + 1$ is modulo 3.) We claim that $\mathcal{S}$ is an $(n, n-3)$-SE system. Let $X \subset N$ be such that $1 \leq |X| \leq n-2$. We show that X is covered in that there exists $Y \in \mathcal{S}$ such that $|X \setminus Y| = 1$, i.e. such that $|X^c \cap Y^c| = 2$. Let's call the set of points in N that share a common first coordinate a *bin*. By the construction of $\mathcal{S}$, it is easy to verify that if X^c intersects some bin at exactly two points, then X is covered. It can also be shown that if X^c intersects some two bins each at a single point, then X is also covered. Now, excluding the two cases discussed above, we may assume that X^c intersects no bins at two points, and at most one bin at a single point. Since $|X^c| \geq 2$, this implies that X^c must intersect some bin at three points. This fact, however, also implies that X is covered (by construction rule number 2). Finally, some simple algebra shows that $|\mathcal{S}| < \frac{2}{3}\binom{n}{2}$.

For $k = 1$, it is easy to see that $S(n, n-2) = n - 1$. (Note in this case $T(n, n-1, n-2) = \lceil n/2 \rceil$.) ■

The proof of Theorem 4.8 shows that for $k = 1, 2$, a stronger result holds, namely,

$$S(n, n-k-1) \leq \frac{2}{k+1}\binom{n}{k}.$$

We shall see that the above is indeed true asymptotically for all fixed k, based on the following construction, which is modified from Construction 4.1 and improves upon it.

Construction 4.2 Let N be an n-set. Assume $t < n-2$. Let N_i, $i = 0, \dots, l-1$ and $w(X)$, $\forall X \subseteq N$ be defined as in Construction 4.1. For each $j \in \{0, \dots, l-1\}$, let

$$\mathcal{K}'_j = \mathcal{E}' \cup \mathcal{W}_j,$$

where (subscripts of N are modulo l, and similarly hereafter)

$$\mathcal{E}' = \big\{X \in [N]^t : \exists i, X \cap N_i = \emptyset, N_{i-1} \nsubseteq X\big\},$$

and $\mathcal{W}_j$ is as defined in (4.11).

Proposition 4.9 *For all j and all $l \geq n/(n-t-2)$, $\mathcal{K}'_j$ as given in Construction 4.2 is an (n,t)-SE system.*

Proof: We show that any $X \subseteq N$, $1 \leq |X| \leq t+1$ is covered by a block in $\mathcal{K}'_j$. If $X \cap N_i = \emptyset$ for some i, let $Y \in [N \setminus N_i]^{t+1}$ be selected such that $X \subseteq Y$ and $|Y \cap N_{i-1}|$ is as small as possible. Since $l \geq n/(n-t-2)$, we have $n - |N_i| \geq t+2$, which ensures that Y exists and that if $N_{i-1} \nsubseteq X$ then $N_{i-1} \nsubseteq Y$. Now, choose $x \in X$ such that if $N_{i-1} \subseteq X$ then $x \in N_{i-1}$, otherwise arbitrarily. Note that X is covered by $Y \setminus \{x\}$. But we also have $(Y \setminus \{x\}) \cap N_i = \emptyset$, and $N_{i-1} \nsubseteq (Y \setminus \{x\})$. Therefore, $Y \setminus \{x\} \in \mathcal{E}'$.

On the other hand, if $X \cap N_i \neq \emptyset$ for all i, select one element in each such intersection, say $x_i \in X \cap N_i$. Now, choose $Y \in [N]^{t+1}$ such that $X \subseteq Y$, and consider $Y \setminus \{x_i\}$, $i = 0, \dots, l-1$. All these sets cover X, and since $w(Y \setminus \{x_i\}) = w(Y) - i$, the set $\big\{w(Y \setminus \{x_i\})\big\}_{i=0}^{l-1}$ contains l consecutive integers, one of which must be congruent to j modulo l. Hence, for all j, there exists i such that X is covered by $Y \setminus \{x_i\} \in \mathcal{W}_j$. ■

Theorem 4.10 *For all integers $l \geq n/(n-t-2)$,*

$$S(n,t) \leq \frac{1}{l}\binom{n}{t} + l\left[\binom{n - \lfloor \frac{n}{l} \rfloor}{t} - \binom{n - \lfloor \frac{n}{l} \rfloor - \lceil \frac{n}{l} \rceil}{t - \lceil \frac{n}{l} \rceil}\right]. \tag{4.14}$$

Proof: From Proposition 4.9, for all $l \geq n/(n-t-2)$,

$$S(n,t) \leq \min_{0 \leq j < l} |\mathcal{K}'_j|.$$

Note that

$$\mathcal{E}' = \bigcup_{i=0}^{l-1} (\mathcal{E}_i \setminus \mathcal{E}'_i),$$

where

$$\mathcal{E}_i = \bigl\{X \in [N]^t : X \cap N_i = \emptyset\bigr\},$$
$$\mathcal{E}'_i = \bigl\{X \in [N]^t : X \cap N_i = \emptyset, N_{i-1} \subseteq X\bigr\}.$$

Thus, we have

$$\begin{aligned}\min_{0\le j<l} |\mathcal{K}'_j| &\le |\mathcal{E}'| + \min_{0\le j<l} |\mathcal{W}_j| \\ &\le \sum_{i=0}^{l-1} \left[\binom{n-|N_i|}{t} - \binom{n-|N_i|-|N_{i-1}|}{t-|N_{i-1}|}\right] + \min_{0\le j<l} |\mathcal{W}_j| \\ &\le l\left[\binom{n-\lfloor\frac{n}{l}\rfloor}{t} - \binom{n-\lfloor\frac{n}{l}\rfloor-\lceil\frac{n}{l}\rceil}{t-\lceil\frac{n}{l}\rceil}\right] + \frac{1}{l}\binom{n}{t}.\end{aligned}$$ ■

An alternative (slightly weaker) form of the upper bound is given in the following theorem.

Theorem 4.11 *For all integers $l \ge n/(n-t-2)$,*

$$S(n,t) \le \frac{1}{l}\binom{n}{t} + l\left\lceil\frac{n}{l}\right\rceil\binom{n-\lfloor\frac{n}{l}\rfloor-1}{t}. \tag{4.15}$$

Proof: Note that

$$\mathcal{E}' = \bigcup_{k=0}^{l-1} \bigcup_{\alpha\in N_{k-1}} \Bigl[N \setminus \bigl(N_k \cup \{\alpha\}\bigr)\Bigr]^t.$$

The rest of the proof is similar to that of Theorem 4.10. ■

Corollary 4.12 *For all fixed k, as $n \to \infty$,*

$$S(n, n-k-1) \le \left(\frac{2}{k+1} + O\bigl(n^{-1}\bigr)\right)\binom{n}{k}.$$

Proof: Theorem 4.11 applies provided that $l \ge n/(k-1)$. If $k \ge 4$, let $l = \lfloor n/2\rfloor$. From (4.15), we have

$$\begin{aligned}S(n,n-k-1) &\le \frac{2}{n-2}\binom{n}{n-k-1} + \frac{3n}{2}\binom{n-3}{n-k-1} \\ &= \frac{2}{k+1}\binom{n}{k} + O(n^{k-1}).\end{aligned}$$

For $k = 3$, let $l = \lceil n/2 \rceil$. If n is even, the above derivation is still valid. If n is odd, note that there is one bin that contains a single element, and the remaining $(n-1)/2$ bins all contain two elements. From the proof of Theorem 4.11, we have

$$\begin{aligned} |\mathcal{E}'| &\le \sum_{i=0}^{\lceil n/2 \rceil - 1} |N_{i-1}| \cdot \binom{n - |N_i| - 1}{n-4} \\ &= (n-2) \cdot \binom{n-3}{n-4} + 2 \cdot \binom{n-2}{n-4} \\ &= O\big(n^2\big) \end{aligned}$$

Hence,

$$\begin{aligned} S(n, n-4) &\le \frac{2}{n+1} \binom{n}{n-4} + O\big(n^2\big) \\ &= \frac{1}{2} \binom{n}{3} + O\big(n^2\big) \end{aligned}$$

For $k < 3$, the result is already shown in the proof of Theorem 4.8. ■

Note that the above bound is sharp for $k = 1$. For $k > 1$, stronger results can be obtained using recurrent inequalities, which are discussed in the next section.

We remark that Construction 4.2 also serves as a construction for Turán systems. In that sense, it is an improved version of the construction by Kim and Roush [15].

Proposition 4.13 *For all l and j, $\mathcal{K}'_j$ as given in Construction 4.2 is an $(n, t+1, t)$-Turán system.*

Proof: The proof is similar to that of Proposition 4.9. ■

Our next two constructions for SE systems are inspired by the following construction for $(n, t+1, t)$-Turán systems due to Frankl and Rödl [16].

Construction 4.3 ([16]) Let N be an n-set. Let N_i, $i = 0, \ldots, l-1$ and $w(X)$, $\forall X \subseteq N$ be defined as in Construction 4.1. Also, for all $X \subseteq N$, define $\Lambda(X) \overset{\text{def}}{=} \{i : X \cap N_i \neq \emptyset\}$ and $\lambda(X) \overset{\text{def}}{=} |\Lambda(X)|$. So $\lambda(X)$ is the number of partitions that X intersects. Now, for $j \in \{0, \ldots, l-1\}$, let

$$\mathcal{F}_j = \Big\{ X \in [N]^t : \Big(\big(w(X) + j\big) \mod l \Big) \in \big\{0, 1, \ldots, l - \lambda(X)\big\} \Big\}.$$

Theorem 4.14 ([16]) *For all l and j, $\mathcal{F}_j$ constructed according to Construction 4.3 is an $(n, t+1, t)$-Turán system.*

Proof: Note that in general, if $x \in (X \cap N_i)$, then $w(X \setminus \{x\}) = w(X) - i$. Let X be a $(t+1)$-set. Since X intersects $\lambda(X)$ partitions, $\{(w(Y) + j) \mod l : Y \in [X]^t\}$ contains $\lambda(X)$ distinct values. Hence, there exists $Y \in [X]^t$, such that $(w(Y)+j) \mod l \in \{0, 1, \ldots, l - \lambda(X)\}$. Now, note that $\lambda(Y) \leq \lambda(X)$ since $Y \subseteq X$. Therefore, $(w(Y) + j) \mod l \in \{0, 1, \ldots, l - \lambda(Y)\}$, which implies that $Y \in \mathcal{F}_j$. ■

We observe that if $l \leq n/(t+1)$, then the above construction actually produces an (n, t)-SE system.

Theorem 4.15 *If $l \leq n/(t+1)$, then for all j, $\mathcal{F}_j$ constructed according to Construction 4.3 is an (n, t)-SE system.*

Proof: Let $X \subseteq N$ be such that $1 \leq |X| \leq t+1$. Choose $Z \in [N]^{t+1}$, such that $X \subseteq Z$ and $\Lambda(Z) = \Lambda(X)$. This is possible as $|\bigcup_{i \in \Lambda(X)} N_i| \geq \lambda(X) \lfloor n/l \rfloor \geq t+1$. Consider the class of t-sets, $\mathcal{Y} = \{Z \setminus \{x\} : x \in X\}$. For all $Y \in \mathcal{Y}$, we have $|X \setminus Y| = 1$. Note that $\{(w(Y) + j) \mod l : Y \in \mathcal{Y}\}$ contains $\lambda(X)$ distinct values. Hence, there exists $Y \in \mathcal{Y}$, such that $(w(Y)+j) \mod l \in \{0, 1, \ldots, l - \lambda(X)\}$. Now, note that $Y \subseteq Z$ implies that $\lambda(Y) \leq \lambda(Z) = \lambda(X)$. Therefore, $(w(Y) + j) \mod l \in \{0, 1, \ldots, l - \lambda(Y)\}$, which implies that $Y \in \mathcal{F}_j$. ■

It was shown in [3] that

$$\sum_{j=0}^{l-1} |\mathcal{F}_j| = \binom{n}{t} + l\binom{n - \lfloor \frac{n}{l} \rfloor}{t}.$$

Therefore,

$$\min_j |\mathcal{F}_j| \leq \frac{1}{l} \sum_{j=0}^{l-1} |\mathcal{F}_j| = \frac{1}{l}\binom{n}{t} + \binom{n - \lfloor \frac{n}{l} \rfloor}{t}.$$

Thus, we have the following theorems.

Theorem 4.16 *For all positive integers $l \leq n/(t+1)$,*

$$S(n, t) \leq \frac{1}{l}\binom{n}{t} + \binom{n - \lfloor \frac{n}{l} \rfloor}{t}.$$

The requirement that l be no greater than $n/(t+1)$ can be rather restrictive, and makes the upper bound less useful unless t is small relative to n. We may get around this problem by adding some additional blocks to $\mathcal{F}_j$. One way of doing so is described in the following construction. For clarity, we first assume $l \mid n$.

Construction 4.4 Let N, $N_0, N_1, \ldots, N_{l-1}$ and $w(X), \forall X \subseteq N$ be defined as in Construction 4.1. Arrange elements of N into an $(n/l) \times l$ array such that the i-th column consists of the elements of N_i. Note that rows of this matrix also partition N, and we denote the sets of elements contained in each row by $M_0, \ldots, M_{n/l-1}$. For all $X \subseteq N$, define

$$w'(X) \overset{\text{def}}{=} \sum_{i=0}^{n/l-1} i|X \cap M_i|.$$

For $j = 0, \ldots, l-1$ and $m = 0, \ldots, n/l-1$, let

$$\mathcal{F}'_{j,m} = \mathcal{F}_j \cup \mathcal{M}_m,$$

where $\mathcal{F}_j$ is as defined in Construction 4.3, and

$$\mathcal{M}_m = \big\{X \in [N]^t : w'(X) \equiv m \mod (n/l)\big\}.$$

We show that for all l, $\mathcal{F}'_{j,m}$ as defined in Construction 4.4 is an (n,t)-SE system.

Lemma 4.17 *Let $l \geq 2$ be an integer. Let $L = \{0, 1, \ldots, l-1\}$. For all $X \subseteq L$, define*

$$\|X\| \overset{\text{def}}{=} \sum_{x \in X} x.$$

Then, for all $i = 1, \ldots, l-1$,

$$\big\{\|X\| \mod l : X \in [L]^i\big\} = L.$$

Proof: The case for $i = 1$ is trivial. For $i = 2$, the claim is proved simply by noting that $j = \|\{0, j\}\|$ for $j = 1, \ldots, l-1$, and $l = \|\{1, l-1\}\|$.

In general, if the claim is true for i, then it is also true for $l - i$, because

$$\big\{\|X\| \mod l : X \in [L]^{l-i}\big\} = \big\{(\|L\| - \|Y\|) \mod l : Y \in [L]^i\big\}.$$

Hence, the claim is also true for $i = l - 1$ and $i = l - 2$.

Now, for the general case, let's assume $i \leq l - 3$. The idea is to consider pairs of elements in L that sum to 0 modulo l. We will only prove the case where l is even. The case for odd l is similar. Let

$$Z_j \overset{\text{def}}{=} \{j, l - j\}, \quad j = 1, \ldots, l/2 - 1.$$

We show that for all $m \in L$, we can find $X \in [L]^i$, such that $\|X\| \equiv m \mod l$.

Indeed, if i is odd, then we can simply choose $X = \{m\} \cup \bigcup_{j \in \mathcal{J}} Z_j$, where $\mathcal{J}$ is any $(i-1)/2$-subset of $[l/2 - 1]$ that does not contain $\min\{m, l - m\}$. If i is even, then X can be found by

- $X = \bigcup_{j \in \mathcal{J}} Z_j$, for $m = 0$, where $\mathcal{J}$ is an arbitrary $(i/2)$-subset of $[l/2 - 1]$;
- $X = \{0, m\} \cup \bigcup_{j \in \mathcal{J}} Z_j$, for all $m \neq 0$, where $\mathcal{J}$ is any $(i/2 - 1)$-subset of $[l/2 - 1]$ that does not contain $\min\{m, l - m\}$. ■

Theorem 4.18 *For all l, j, and m, $\mathcal{F}'_{j,m}$ as defined in Construction 4.4 is an (n, t)-SE system.*

Proof: Since it is known [16] that $\mathcal{F}_j$ is an $(n, t + 1, t)$-Turán system, all $(t + 1)$-subsets of N are covered. It remains to show that any $X \subseteq N$, $1 \leq |X| \leq t$, is covered by a block in $\mathcal{F}'_{j,m}$. In the following, let $x \in X$ be an arbitrary element.

First, suppose $N_i \nsubseteq X$ for all i. If $t \leq n - l$, then we can find a t-set $Z \supseteq X$ such that $N_i \nsubseteq Z$ for all i. Now, let $Y_i = (Z \setminus \{x\}) \cup \{y_i\}$, where $y_i \in N_i \setminus Z$ are arbitrarily chosen. Clearly, $|X \setminus Y_i| = 1$ for all i. For any gvien j, by varying i we can realize all l possible values of $\big(w(Y_i) + j\big) \mod l$. Hence, there exists i such that $w(Y_i) + j \equiv 0 \mod l$, which implies that $Y_i \in \mathcal{F}_j$.

On the other hand, if $t > n - l$, then we can find an $(n - l)$-set $Z \supseteq X$ such that $N_i \nsubseteq Z$ for all i. Clearly, $N \setminus Z$ intersects each N_i at exactly one element. Consider t-sets that consist of the union of $Z \setminus \{x\}$ and an $(t - n + l + 1)$-subset of $N \setminus Z$. By Lemma 4.17, for any given j, there exists $W \in [N \setminus Z]^{t-n+l+1}$ such that if $Y = (Z \setminus \{x\}) \cup W$ then $w(Y) + j \equiv 0 \mod l$. Therefore, $Y \in \mathcal{F}_j$ and clearly $|X \setminus Y| = 1$.

Next, consider the case where $N_i \subseteq X$ for some i. By construction, N_i contains elements from each of $M_0, \ldots, M_{n/l-1}$. Let $Z \supseteq X$ be a superset of X that contains $t + 1$

elements. Let $\mathcal{Y} = \big\{Z \setminus \{x\} : x \in X\big\}$. By choosing $Y \in \mathcal{Y}$, we can realize any value of $w'(Y)$ $\bmod (n/l)$. Hence, for any m, there exists a t-set $Y \in \mathcal{M}_m$ such that $|X \setminus Y| = 1$. ■

Although our results have been stated assuming l divides n, both Construction 4.4 and Theorem 4.18 can be straightforwardly extended to the general case. Specifically, if $l \nmid n$, we can define $M_0, \ldots, M_{\lfloor n/l \rfloor - 1}$ by applying Construction 4.4 to some $l\lfloor n/l \rfloor$ elements of N and letting $M_{\lfloor n/l \rfloor - 1}$ include the remaining $(n \mod l)$ elements.

Clearly,

$$\sum_{m=0}^{\lfloor n/l \rfloor - 1} |\mathcal{M}_m| = \binom{n}{t}.$$

Hence,

$$\min_m |\mathcal{M}_m| \le \frac{1}{\lfloor n/l \rfloor} \binom{n}{t}.$$

By the union bound, $|\mathcal{F}'_{j,m}| \le |\mathcal{F}_j| + |\mathcal{M}_m|$, hence we arrive at the following bounds.

Theorem 4.19 *For all integers l, $1 \le l \le n$,*

$$S(n,t) \le \begin{cases} \binom{n-\lfloor n/l \rfloor}{t} + \frac{1}{l}\binom{n}{t} & \text{if } l \le \frac{n}{t+1}, \\ \binom{n-\lfloor n/l \rfloor}{t} + \left(\frac{1}{l} + \frac{1}{\lfloor n/l \rfloor}\right)\binom{n}{t} & \text{if } l > \frac{n}{t+1}. \end{cases} \tag{4.16}$$

Proof: Omitted, as it is clear from the preceding discussion. ■

Note that when $l > n/(t+1)$, the above bound is never better than $\frac{2}{\sqrt{n}}\binom{n}{t}$ due to the second term. This implies that the above bound is still rather weak when t is not significantly smaller than n.

We now consider a second way of augmenting Construction 4.3.

Construction 4.5 Let N be an n-set. Assume $t < n$. Let N_i, $i = 0, \ldots, l-1$, and $w(X)$, $\forall X \subseteq N$, be defined as in Construction 4.1. We will call each N_i a *bin*. For all $X \subseteq N$ and $j = 0, \ldots, l-1$, define

$$w_j(X) \overset{\text{def}}{=} (w(X) + j) \mod l.$$

Now, for all $j = 0, \ldots, l-1$, let

$$\mathcal{F}''_j = \tilde{\mathcal{F}}''_j \cup \mathcal{U},$$

where

$$\tilde{\mathcal{F}}''_j = \big\{X \in [N]^t : w_j(X) \leq \max\{e(X), f(X)\}\big\},$$

where

$$e(X) = |\{i : X \cap N_i = \emptyset\}|,$$
$$f(X) = |\{i : N_i \subseteq X\}|$$

are the number of "empty" and "full" bins for X, respectively, and the set $\mathcal{U}$ is constructed as follows.

Fix an arbitrary total order on N. Let $I \subseteq \{0, \dots, l-1\}$ be an index set that satisfies $\sum_{i \in I} |N_i| > t$, and is *minimal* in the sense that all proper subsets of I violate this condition. For each such I and $i, j \in I$, $i \neq j$, let $\mathcal{U}$ include the t-set that consists of all elements from bins $N_m, m \in I \setminus \{i, j\}$, the smallest $|N_i| - 1$ elements of N_i, and the smallest $\big(t + 1 - \sum_{m \in I, m \neq j} |N_m|\big)$ elements of N_j.

Proposition 4.20 *For all l and j, $\mathcal{F}''_j$ as given in Construction 4.5 is an (n,t)-SE system.*

Proof: We show that any $X \subseteq N$, $1 \leq |X| \leq t+1$, is covered by a block in $\mathcal{F}''_j$. If $|X| = t+1$, note that all t-subsets of X can be written as $X \setminus \{x\}$, for some $x \in X$. Since $w(X \setminus \{x\}) = w(X) - w(\{x\})$ for all $x \in X$, by choosing x from different bins that X intersects, we can make $w(X \setminus \{x\})$ take on $l - e(X)$ different values. Since no two of these values differ by more than $l - 1$, this also means that we can realize $l - e(X)$ different values for $w_j(X \setminus \{x\})$. Since only $l - e(X) - 1$ numbers in $\{0, \dots, l-1\}$ are greater than $e(X)$, there exists $x \in X$ such that $w_j(X \setminus \{x\}) \leq e(X) \leq e(X \setminus \{x\})$, hence $X \setminus \{x\} \in \tilde{\mathcal{F}}''_j$, and it covers X.

If $i \leq t$, consider two cases. First, let us assume that there exists i, such that $X \cap N_i \neq \emptyset$ and $N_i \nsubseteq X$. In this case, remove from X an arbitrary element in $X \cap N_i$, add in $t - |X|$ other elements from N using as few elements from N_i as possible, and call the resulting $(t-1)$-set $\tilde{X}$. That is, $\tilde{X} = (X \setminus \{x\}) \cup Y$, for some $x \in X \cap N_i$ and some $Y \in [N \setminus X]^{t - |X|}$ that has a minimal number of elements from N_i. Note that the choice of $\tilde{X}$ ensures that $f(\tilde{X} \cup \{x\}) = f(\tilde{X})$. Since $w(\tilde{X} \cup \{z\}) = w(\tilde{X}) + w(\{z\})$ for all $z \notin \tilde{X}$, by choosing $z \notin \tilde{X}$, $z \neq x$, from different bins where possible, we can make $w(\tilde{X} \cup \{z\})$ take on $l - f(\tilde{X} \cup \{x\}) = l - f(\tilde{X})$ different values. This also means that we can realize $l - f(\tilde{X})$ different values for $w_j(\tilde{X} \cup \{z\})$. Since only $l - f(\tilde{X}) - 1$ numbers in $\{0, \dots, l-1\}$ are greater than $f(\tilde{X})$, there exists z such that $w_j(\tilde{X} \cup \{z\}) \leq f(\tilde{X}) \leq f(\tilde{X} \cup \{z\})$, hence $\tilde{X} \cup \{z\} \in \tilde{\mathcal{F}}''_j$, and it covers X.

Next, if no i exists such that $X \cap N_i \neq \emptyset$ and $N_i \nsubseteq X$, this means that for all i such that $X \cap N_i \neq \emptyset$, we have $N_i \subseteq X$. Figuratively, it means that X consists of a number of full bins. Let N_i be any bin that X intersects. Let $x \in N_i$ be its largest element. Take $X \setminus \{x\}$, and add to it elements from bins that X does not intersect, one bin after another, from smallest to the largest within each bin, until $X \setminus \{x\}$ is augmented to contain t elements. By construction, the t-subset thus obtained is contained in $\mathcal{U}$. ■

Theorem 4.21 *For all positive integers l,*

$$S(n,t) \leq \frac{1}{l}\binom{n}{t} + \binom{n - \lfloor \frac{n}{l} \rfloor}{t} + \binom{n - \lfloor \frac{n}{l} \rfloor}{t - \lfloor \frac{n}{l} \rfloor} + g(n,t,l), \tag{4.17}$$

where

$$g(n,t,l) = \sum_{\frac{t+1}{\lceil n/l \rceil} \leq i \leq \left\lceil \frac{t+1}{\lfloor n/l \rfloor} \right\rceil} \binom{l}{i} i(i-1).$$

Proof: Note that

$$\begin{aligned}\sum_{X \in [N]^t} f(X) &= \sum_{X \in [N]^t} \sum_i 1_{\{N_i \subseteq X\}} \\ &= \sum_i \Big(\sum_{X \in [N]^t} 1_{\{N_i \subseteq X\}} \Big) \\ &\leq l \binom{n - \lfloor \frac{n}{l} \rfloor}{t - \lfloor \frac{n}{l} \rfloor}.\end{aligned}$$

Similarly,

$$\sum_{X \in [N]^t} e(X) \leq l \binom{n - \lfloor \frac{n}{l} \rfloor}{t}.$$

Each $X \in [N]^t$ is contained in precisely $1 + \max\{e(X), f(X)\}$ of the $\tilde{\mathcal{F}}_j''$'s. Therefore,

$$\begin{aligned}\sum_j |\tilde{\mathcal{F}}_j''| &= \sum_{X \in [N]^t} \big(1 + \max\{e(X), f(X)\}\big) \\ &\leq \binom{n}{t} + \sum_{X \in [N]^t} \big(e(X) + f(X)\big) \\ &\leq \binom{n}{t} + l\binom{n - \lfloor \frac{n}{l} \rfloor}{t} + l\binom{n - \lfloor \frac{n}{l} \rfloor}{t - \lfloor \frac{n}{l} \rfloor}.\end{aligned}$$

Hence,

$$\min_j |\tilde{\mathcal{F}}_j''| \leq \frac{1}{l}\binom{n}{t} + \binom{n - \lfloor \frac{n}{l} \rfloor}{t} + \binom{n - \lfloor \frac{n}{l} \rfloor}{t - \lfloor \frac{n}{l} \rfloor}.$$

Finally, note that $\mathcal{U}$ contains $|I|(|I|-1)$ size-t subsets for each valid I, which must satisfy $|I|\lceil n/l\rceil \geq t+1$ and $(|I|-1)\lfloor n/l\rfloor < t+1$. Therefore, $|\mathcal{U}| \leq g(n,t,l)$. ■

4.2.3 Recurrent Bounds

We now turn to a different approach. The key observation here is that an (n,t)-SE system can be constructed from an $(n-1,t-1)$-SE system and an $(n-1,t+1,t)$-Turán system, as shown in the following lemma.

Lemma 4.22 *For all* $0 < t < n-1$,

$$S(n,t) \leq S(n-1,t-1) + T(n-1,t+1,t),$$

or equivalently, for all $0 < k < n-1$,

$$S(n,n-k-1) \leq S(n-1,n-k-2) + C(n-1,k,k-1).$$

Proof: Let N be an n-set and $a \in N$ be an arbitrary element. Let $\mathcal{S} \subseteq [N\setminus\{a\}]^{t-1}$ be a minimal $(n-1,t-1)$-SE system, and $\mathcal{T} \subseteq [N\setminus\{a\}]^{t}$ be a minimal $(n-1,t+1,t)$-Turán system. Define $\mathcal{S}' = \{Y \cup \{a\} : Y \in \mathcal{S}\}$. Then $\mathcal{S}' \cup \mathcal{T}$ is an (n,t)-SE system. Indeed, for all $X \in [N]^i$, $i = 1,\ldots,t+1$, if $1 \leq |X \setminus \{a\}| \leq t$, then there exists $Y \in \mathcal{S}$ such that $|(X\setminus\{a\})\setminus Y| = 1$, which implies that $|X \setminus (Y \cup \{a\})| = 1$, i.e. X is covered by a block in $\mathcal{S}'$. The only cases left are when $X = \{a\}$, and when $X \in [N\setminus\{a\}]^{t+1}$. In either case, X is covered by a block in $\mathcal{T}$.■

Theorem 4.23 *For all* $0 < t < n-1$,

$$S(n,t) \leq \sum_{i=0}^{t} T(n-t+i-1,i+1,i), \tag{4.18}$$

or equivalently, for all $0 < k < n-1$,

$$S(n,n-k-1) \leq \sum_{i=k}^{n-1} C(i,k,k-1). \tag{4.19}$$

Proof: Recursively apply Lemma 4.22. ■

Interesting results follow. When $k=1$, (4.19) implies that $S(n,n-2) \leq n-1$, which is sharp. When $k=2$, since $C(i,2,1) = \lceil i/2\rceil$, (4.19) gives

$$S(n,n-3) \leq \left\lceil \frac{n}{2}\right\rceil \left\lfloor \frac{n}{2}\right\rfloor - 1,$$

which is asymptotically tighter than Corollary 4.12.

Generally, since exact values of most Turán / covering numbers are not known, the upper bounds in Theorem 4.23 often cannot be directly evaluated. To get a computable upper bound, one can simply replace each Turán / covering number in the sum by an explicit upper bound. We show several ways to do this. The first one is based on a result by Erdős and Spencer [17].

Theorem 4.24 ("Recurrent A") *For all* $0 < k < n-1$,

$$S(n, n-k-1) \le \frac{1+\ln k}{k}\left(\binom{n}{k} - 1\right). \tag{4.20}$$

Proof: In [17], it was shown that for all $n \ge s \ge t$,

$$C(n,s,t) \le \left(1 + \ln\binom{s}{t}\right)\frac{\binom{n}{t}}{\binom{s}{t}}. \tag{4.21}$$

Plugging (4.21) into (4.19), we obtain the claimed result after some algebraic manipulations. ■

The second one is based on an upper bound on Turán numbers due to Sidorenko [3, Construction 4].

Theorem 4.25 ("Recurrent B") *For all* $0 < t < n-1$ *and positive integers* $l_0, l_1, \ldots, l_t$,

$$S(n,t) \le \sum_{i=0}^{t} f_{n,t}(i, l_i), \tag{4.22}$$

where

$$f_{n,t}(i,l_i) = \left[\frac{1}{2l_i} + \frac{1}{2}\left(3 + \frac{i}{l_i - 1 - \frac{l_i(i-1)}{m+i}}\right)\left(1 - \frac{1}{l_i}\right)^i\right] \cdot \binom{m+i}{i},$$

and $m = n - t - 1$.

Proof: Omitted. ■

A third way to obtain an explicit upper bound from Theorem 4.23 is based on a construction of an $(n, k, k-1)$ covering design due to Kuzjurin [18], although we count blocks in a slightly different manner.

Lemma 4.26 *For all positive integers $n \geq k$,*

$$C(n,k,k-1) \leq \frac{1}{k}\binom{n}{k-1} + \frac{k-1}{k}\binom{n-1}{k-2}.$$

Proof: Let N be an n-set. WLOG, let $N = \{1, 2, \ldots, n\}$. Let $Q_i = \{X \in [N]^k : \sum_{x \in X} x \equiv i \mod n\}$, and $C_i = \{X \in [N]^{k-1} : \nexists Y \in Q_i,$ s.t. $X \subset Y\}$, $i = 0, 1, \ldots, n-1$. For each $X \in C_i$, we can add one block $Y \in [N]^k$ to Q_i, such that $X \subset Y$. Hence, by adding no more than $|C_i|$ blocks to Q_i, we construct an $(n,k,k-1)$ covering design. Therefore, for all i,

$$C(n,k,k-1) \leq |Q_i| + |C_i|.$$

Now note that for all $Y, Z \in Q_i, Y \neq Z$, we have $|Y \cap Z| \leq k-2$. Therefore, for all $X \in [N]^{k-1}$ there is at most one block Y in Q_i such that $X \subset Y$; on the other hand, for every $Y \in [N]^k$ there are k elements $X \in [N]^{k-1}$ such that $X \subset Y$. Hence,

$$|C_i| = \binom{n}{k-1} - k|Q_i|.$$

We have

$$\begin{aligned}\sum_{i=0}^{n-1}\big(|Q_i| + |C_i|\big) &= \sum_{i=0}^{n-1}\left(\binom{n}{k-1} - (k-1)|Q_i|\right)\\ &= n\binom{n}{k-1} - (k-1)\binom{n}{k}.\end{aligned}$$

Therefore, there exists i, such that

$$\begin{aligned}|Q_i| + |C_i| &\leq \binom{n}{k-1} - \frac{k-1}{n}\binom{n}{k}\\ &= \frac{1}{k}\binom{n}{k-1} + \frac{k-1}{k}\binom{n-1}{k-2}.\end{aligned}$$ ■

Plugging the bound in the preceding lemma into (4.19), we obtain the following theorem.

Theorem 4.27 ("Recurrent C") *For all $0 < k < n-1$,*

$$\begin{aligned}S(n,n-k-1) &\leq \frac{1}{k}\binom{n}{k} + \frac{k-1}{k}\binom{n-1}{k-1} - 1 \qquad (4.23)\\ &= \left(1 + \frac{k^2-k}{n}\right)\cdot\frac{1}{k}\binom{n}{k} - 1.\end{aligned}$$

Proof: Omitted. ■

Corollary 4.28 *For all* $0 < k < n-1$,

$$S(n, n-k-1) \le \left(1 + \frac{1}{k} + \frac{k^2-1}{n}\right) T(n, n-k, n-k-1)$$

In particular, as $n \to \infty$*, if* $k = o(\sqrt{n})$ *and* $1 = o(k)$*, then*

$$S(n, n-k-1) \sim T(n, n-k, n-k-1).$$

Proof: Omitted. ■

Interestingly, a similar asymptotic relation can be shown between $S(n,t)$ and $T(n,t+1,t)$ when t is small compared to n. In fact, we have already seen in Theorem 4.2 that $S(n,t)$ is asymptotically $\big(1 + O(n^{-1})\big)T(n,t+1,t)$ for all fixed t. Let's first revisit the construction in Theorem 4.2.

Theorem 4.29

$$S(n,t) \le T(n,t+1,t) + \binom{n}{t} - \binom{n-t}{t}$$

Proof: Let $\mathcal{T}$ be a minimal $(n,t+1,t)$-Turán system whose blocks are t-subsets of some n-set N. We show that $\mathcal{T}$ can be augmented to an (n,t)-SE system by adding no more than $\binom{n}{t} - \binom{n-t}{t}$ blocks. Let L be a t-subset of N. Let $\mathcal{T}' = [N]^t \setminus [N \setminus L]^t$. For any $X \in [N]^i$, $i = 1, \dots, t$, and $x \in X$, consider size-t supersets of $X \setminus \{x\}$. Since $t - i + 1 \ge 1$ elements are chosen freely, at least one of the supersets has non-empty intersection with L, and is thus contained in $\mathcal{T}'$. Therefore, $\mathcal{T} \cup \mathcal{T}'$ is an (n,t)-SE system, and it contains no more than

$$T(n,t+1,t) + \binom{n}{t} - \binom{n-t}{t}$$

blocks. ■

Now note that

$$\binom{n}{t} - \binom{n-t}{t} = \sum_{i=1}^{t} \binom{n-i}{t-1} \le t\binom{n-1}{t-1},$$

and on the other hand

$$\frac{t\binom{n-1}{t-1}}{T(n,t+1,t)} \le \frac{t\binom{n-1}{t-1}}{\frac{1}{t+1}\binom{n}{t}} = \frac{t^2(t+1)}{n}.$$

Hence, for all $t = o(\sqrt[3]{n})$, $S(n,t)$ is asymptotic to $T(n,t+1,t)$.

The above result can be slightly improved using Theorem 4.23 and the following result relating SE and Turán numbers.

Theorem 4.30 *For all* $0 < t < n-1$,

$$S(n,t) \leq \left(1 - \frac{t}{n}\right) T(n,t+1,t) + \binom{n-1}{t-1}.$$

Proof: We will use the fact [19] that

$$T(n,s,t) \geq \frac{n}{n-t} T(n-1,s,t).$$

From Theorem 4.23, we have

$$\begin{aligned} S(n,t) &\leq T(n-1,t+1,t) + \sum_{i=0}^{t-1} T(n-t+i-1,i+1,i) \\ &\leq \left(1 - \frac{t}{n}\right) T(n,t+1,t) + \sum_{i=0}^{t-1} \binom{n-t+i-1}{i} \\ &= \left(1 - \frac{t}{n}\right) T(n,t+1,t) + \sum_{i=n-t-1}^{n-2} \binom{i}{n-t-1} \\ &= \left(1 - \frac{t}{n}\right) T(n,t+1,t) + \binom{n-1}{t-1}. \end{aligned}$$

■

Corollary 4.31 *For all* $0 < t < n-1$,

$$S(n,t) \leq \left(1 + \frac{t^2}{n}\right) T(n,t+1,t),$$

In particular, as $n \to \infty$*, if* $t = o(\sqrt{n})$*, then*

$$S(n,t) \sim T(n,t+1,t).$$

Proof: Simply note that

$$\frac{\binom{n-1}{t-1}}{T(n,t+1,t)} \leq \frac{\binom{n-1}{t-1}}{\frac{1}{t+1}\binom{n}{t}} = \frac{t^2+t}{n}.$$

The result follows immediately from Theorem 4.30. ■

4.2.4 Comparison of Upper Bounds

We numerically computed several of the upper bounds on $S(n, d-2)$, and hence on $\rho(\mathcal{C})$, for all $5 < d \leq n \leq 512$. For each (n, d) pair, the tightest bound is identified, and the results are shown in Fig. 4.1. In the figure, "Construction 4.2" refers to Theorem 4.10, "Construction 4.5" refers to Theorem 4.21, "Probabilistic" refers to Theorem 4.3, "Recurrent B" refers to Theorem 4.25, and "Recurrent C" refers to Theorem 4.27. Other bounds included in the comparison (but not appearing in the figure) are "Schwartz-Vardy" (4.1), "Construction 4.1" (Theorem 4.7), "Construction 4.4" (Theorem 4.19), and "Recurrent A" (Theorem 4.24). Wherever a bound contains an auxiliary variable (l, p, etc.), the minimum is taken over the auxiliary variable. Note that $d \leq 5$ is not considered, as in this case $S(n, d-2)$ (or $\rho(\mathcal{C})$) is at most $T(n, d-1, d-2) + 1$, for which either precise formulas are known, or tighter special upper bounds exist. These will be discussed in detail in Section 4.4,

We observe that the most "successful" bounds are Recurrent B, Recurrent C, Probabilistic, and Construction 4.2. A minor exception is Construction 4.5, which excels occasionally for certain small values of (n, d). As n gets larger, a trend can be seen. Roughly speaking, for code rate $(n - d + 1)/n \geq 1/5$, Recurrent B is the best bound. For code rate less than $1/5$, the best bounds are Probabilistic, Construction 4.2, and Recurrent C, in that respective order, as code rate gets progressively smaller.

A few samples of the upper bounds are given in Table 4.2. The tightest are highlighted in boldface. For comparison, a lower bound on $T(n, d-1, d-2)$ (hence on $S(n, d-2)$ and $\rho(\mathcal{C})$ as well) has been included, based on (4.24) (see Section 4.3). Compared to upper bounds previously known, significant improvements can clearly be seen. As a side note, we caution that while Recurrent C is an excellent bound when d is very close to n, it gets loose quickly as d gets smaller, and should be avoided if the code rate is greater than $1/2$.

4.3 Lower Bounds on $S(n, t)$

A few other lower bounds on Turán / covering numbers are known, besides the simple lower bound in (4.4). For example, Schönheim [19] showed that

$$T(n, t+1, t) \geq \left\lceil \frac{n}{n-t} \left\lceil \frac{n-1}{n-t-1} \left\lceil \cdots \left\lceil \frac{t+2}{2} \right\rceil \cdots \right\rceil \right\rceil \right\rceil . \qquad (4.24)$$

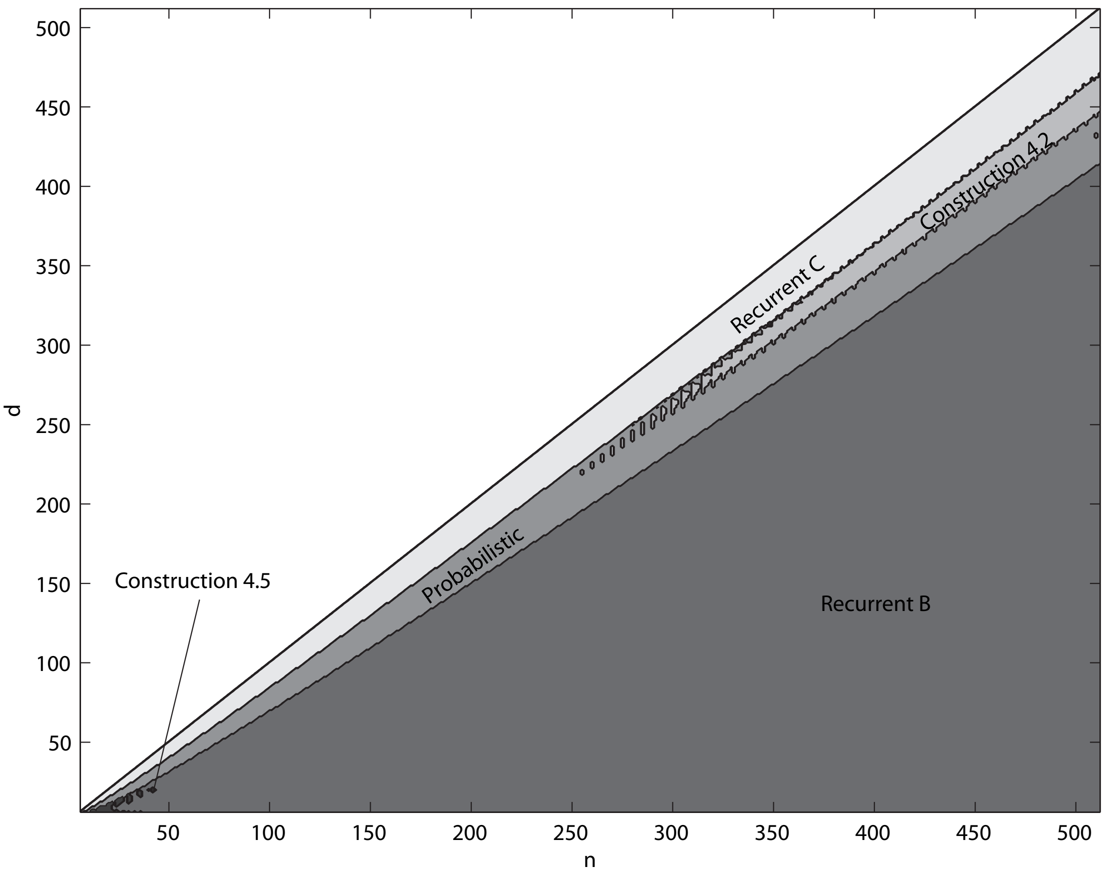

Figure 4.1 Best upper bounds on $S(n, d-2)$ (hence on $\rho(\mathcal{C})$), for $5 < d \leq n \leq 512$.

Table 4.2 Upper bounds on $S(n, d-2)$ and $\rho(\mathcal{C})$

$(n, d) =$	$(31, 7)$	$(31, 23)$	$(31, 27)$
Probabilistic (4.5)	96,112	**6,412,596**	77,298
Construction 4.1 (4.13)	93,691	7,786,707	106,388
Construction 4.2 (4.14)	93,691	7,693,683	86,148
Construction 4.4 (4.16)	76,986	16,275,110	269,970
Construction 4.5 (4.17)	76,986	12,151,903	299,697
Recurrent A (4.20)	124,250	7,161,809	88,673
Recurrent B (4.22)	**71,891**	9,665,343	520,847
Recurrent C (4.23)	599,474	7,442,607	**55,905**
Schwartz-Vardy (4.1)	142,506	31,475,730	617,526
Lower Bound (4.24)	33,981	2,103,660	29,450

Another useful bound is due to De Caen [20]:

$$T(n,t+1,t) \geq \frac{1}{t} \cdot \frac{n-t}{n-t+1} \binom{n}{t}.$$

Note that a lower bound on $T(n,d-1,d-2)$ is in turn a lower bound on $\rho(\mathcal{C})$ and on $S(n,d-2)$.

In this section, we study a lower bound result on $S(n,t)$, which is not a lower bound on $T(n,t+1,t)$ in general. Note that a lower bound (just) on $S(n,d-2)$ is not necessarily a lower bound on $\rho(\mathcal{C})$.

Theorem 4.32 *For all* $0 < t < n-1$,

$$\left(1 + \frac{n-t}{n(n-t-1/2)}\right) S(n,t) \geq T(n-1,t+1,t) + \frac{1}{n-t-1/2}\binom{n-1}{t}.$$

Proof: Let N be an n-set, and $\mathcal{S} \subseteq [N]^t$ be a minimal (n,t)-SE system. For each $j \in N$, $\mathcal{S}$ can be partitioned into blocks that contain j and those that do not, namely,

$$\mathcal{S} = \mathcal{A}_j \cup \mathcal{B}_j,$$

where $\mathcal{A}_j = \{X \in \mathcal{S} : j \in X\}$, and $\mathcal{B}_j = \{X \in \mathcal{S} : j \notin X\}$. Note that for all j, $\mathcal{B}_j$ is an $(n-1,t+1,t)$-Turán system. Further, if we let $\mathcal{A}'_j = \{A \setminus \{j\} : A \in \mathcal{A}_j\}$, then each t-set in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$ contains at least one element of $\mathcal{A}'_j$. To see this, suppose $X \in [N \setminus \{j\}]^t \setminus \mathcal{B}_j$. Then $X \cup \{j\}$ is a $(t+1)$-set and so there exists $Y \in \mathcal{S}$ such that $Y \subset \big(X \cup \{j\}\big)$. Since $X \notin \mathcal{S}$, we have $j \in Y$ and hence $\mathcal{A}'_j \ni (Y \setminus \{j\}) \subset X$.

On the other hand, since each element of $\mathcal{A}'_j$ is contained in

$$(n-1) - (t-1) = n-t$$

t-subsets of $N \setminus \{j\}$, it is contained in at most $(n-t)$ distinct t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$. Therefore,

$$|\mathcal{A}_j| = |\mathcal{A}'_j| \geq \frac{1}{n-t}\left(\binom{n-1}{t} - |\mathcal{B}_j|\right).$$

This lower estimate can be improved by a more careful argument as follows. Let

$$
\begin{aligned}
\mathcal{A}'_{j1} &= \{A \in \mathcal{A}'_j : \exists B \in \mathcal{B}_j, A \subset B\},\\
\mathcal{A}'_{j2} &= \{A \in \mathcal{A}'_j \setminus \mathcal{A}'_{j1} : \exists A' \in \mathcal{A}'_{j1}, |A \setminus A'| = 1\},\\
\mathcal{A}'_{j3} &= \{A \in \mathcal{A}'_j \setminus (\mathcal{A}'_{j1} \cup \mathcal{A}'_{j2}) : \exists A' \in \mathcal{A}'_{j2}, |A \setminus A'| = 1\},\\
&\ \ \vdots\\
\mathcal{A}'_{ji} &= \left\{A \in \mathcal{A}'_j \setminus \bigcup_{l=1}^{i-1} \mathcal{A}'_{jl} : \exists A' \in \mathcal{A}'_{j(i-1)}, |A \setminus A'| = 1\right\},\\
&\ \ \vdots
\end{aligned}
$$

Note that $\mathcal{A}'_{ji} \cap \mathcal{A}'_{jl} = \emptyset$ for all $i \neq l$. Since $\mathcal{A}'_j$ is finite, there exists i such that $\mathcal{A}'_{jl} = \emptyset$ for all $l > i$. Regardless, define

$$\tilde{\mathcal{A}}'_j = \bigcup_{l=1}^{\infty} \mathcal{A}'_{jl}.$$

We claim that on average, an element in $\tilde{\mathcal{A}}'_j$ is contained in at most $(n-t-1)$ t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$. To see this, consider a process in which we enumerate elements of $\mathcal{A}'_j$, and for each element, "mark" the t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$ that contain it. We start with elements in $\mathcal{A}'_{j1}$ and proceed to $\mathcal{A}'_{j2}, \mathcal{A}'_{j3}$, and so on. Each element in $\mathcal{A}'_{j1}$ is contained in $(n-t)$ t-sets in $[N \setminus \{j\}]^t$, at least one of which lies in $\mathcal{B}_j$. Therefore, for each element in $\mathcal{A}'_{j1}$, at most $(n-t-1)$ t-sets are marked. Now, for any $X \in \mathcal{A}'_{j2}$, by definition, there exists $Y \in \mathcal{A}'_{j1}$, such that $|X \setminus Y| = 1$. Hence, among the $(n-t)$ t-sets that contain X, at least one of them, namely, $X \cup Y$, is already marked. Therefore, processing any $X \in \mathcal{A}'_{j2}$ marks at most $(n-t-1)$ additional t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$. A similar argument shows that among the $(n-t)$ t-sets that contain an element of $\mathcal{A}'_{ji}$, at least one of them is already marked after elements of $\mathcal{A}'_{j(i-1)}$ have been processed.

For $\mathcal{A}'_j \setminus \tilde{\mathcal{A}}'_j$, we show that on average, each element marks at most $(n-t-1/2)$ t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$. Let $X \in \mathcal{A}'_j \setminus \tilde{\mathcal{A}}'_j$. As $X \cup \{j\} \in [N]^t$, there exists $Y \in \mathcal{S}$, such that $|(X \cup \{j\}) \setminus Y| = 1$. Since $X \notin \mathcal{A}'_{j1}$, we have $j \in Y$ and hence $Y \setminus \{j\} \in \mathcal{A}'_j$. In fact, $Y \setminus \{j\} \in \mathcal{A}'_j \setminus \tilde{\mathcal{A}}'_j$, since otherwise it would imply that $X \in \tilde{\mathcal{A}}'_j$. Now, let $Z = X \cup Y \setminus \{j\}$, and denote by l the number of elements of $\mathcal{A}'_j \setminus \tilde{\mathcal{A}}'_j$ that are contained in Z. Note that $l \geq 2$, since Z contains both X and $Y \setminus \{j\}$. Therefore, of the l elements that are contained in Z, each on average marks

$$n - t - (l-1)/l \leq n - t - 1/2$$

t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$. For other elements in $\mathcal{A}'_j \setminus \tilde{\mathcal{A}}'_j$, the above argument can be repeated until all elements have been considered.

Based on the preceding discussions, we conclude that on average, each block in $\mathcal{A}'_j$ is contained in no more than $(n-t-1/2)$ t-sets in $[N \setminus \{j\}]^t \setminus \mathcal{B}_j$. Hence,

$$\begin{aligned} |\mathcal{S}| &= |\mathcal{A}'_j| + |\mathcal{B}_j| \\ &\geq \frac{1}{n-t-1/2}\left(\binom{n-1}{t} - |\mathcal{B}_j|\right) + T(n-1, t+1, t). \end{aligned} \tag{4.25}$$

Since each block of $\mathcal{S}$ appears in $(n-t)$ $\mathcal{B}_j$'s, we have

$$\sum_{j \in N} |\mathcal{B}_j| = (n-t)|\mathcal{S}|.$$

Summing (4.25) over all j, dividing both sides by n, and noting that $|\mathcal{S}| = S(n,t)$ (since $\mathcal{S}$ was chosen to be minimal) gives the desired inequality. ■

Corollary 4.33 *For all $0 < t < n-1$, we have*

$$S(n,t) \geq \frac{1}{n-t-(t/n)+1/2}\binom{n}{t+1}.$$

Proof: Simply use the facts that

$$T(n-1, t+1, t) \geq \frac{1}{n-t-1}\binom{n-1}{t+1},$$

and

$$\binom{n}{t+1} = \binom{n-1}{t+1} + \binom{n-1}{t}.$$

■

Equivalently, if we let $k = n-t-1$, then we have that for all $0 < k < n-1$,

$$S(n, n-k-1) \geq \frac{1}{k+(k+1)/n+1/2}\binom{n}{k}.$$

To relate this to the asymptotic results shown earlier, we note the following corollary.

Corollary 4.34 *For all $k > 0$, as $n \to \infty$,*

$$S(n, n-k-1) \geq \big(1 - O(n^{-1})\big)\frac{1}{k+1/2}\binom{n}{k}.$$

Proof: Trivial. ■

For fixed k, $T(n, n-k, n-k-1)$ is asymptotic to $\frac{1}{k+1}\binom{n}{k}$ (cf. [21]). So the above corollary shows that for *any* fixed k, the ratio $S(n, n-k-1) \,/\, T(n, n-k, n-k-1)$ is bounded away from 1 as $n \to \infty$.

4.4 On the Schwartz-Vardy Conjecture

Schwartz and Vardy [2] conjectured that the stopping redundancy of an MDS code should only depend on its length and minimum distance.

In Section 4.2, using recurrent inequalities, we were able to show that this conjecture holds true in an asymptotic sense under the following scenarios:

- $d = o(\sqrt{n})$, or
- $k = o(\sqrt{n})$, $1 = o(k)$.

In both cases, our upper and lower bounds on $\rho(\mathcal{C})$ in (4.3) become asymptotic to each other.

We now show that the Schwartz-Vardy conjecture holds in the exact sense for $1 < d \leq 5$. When $d = 2$, it is straightforward to see that $\rho(\mathcal{C}) = T(n, 1, 0) = S(n, 0) = 1$. The case with $d = 3$ is only slightly less obvious.

Theorem 4.35 *Let $\mathcal{C}$ be an $[n, n-2, 3]$ MDS code. Then*

$$\rho(\mathcal{C}) = S(n, 1) = T(n, 2, 1) = n - 1.$$

Proof: It suffices to show that $T(n, 2, 1) \geq n - 1$ and $S(n, 1) \leq n - 1$. Let N be an n-set. It is easy to verify that any $(n-1)$-subset of $[N]^1$ is an $(n, 1)$-SE system. On the other hand, an $(n, 2, 1)$-Turán system clearly cannot have $(n-2)$ or fewer blocks, or there would exist $i, j \in N$, such that $\{i, j\}$ does not contain any block. ■

The case for $d = 4$ needs a bit more work.

Lemma 4.36 *For all $n \geq 3$,*

$$T(n, 3, 2) \leq \binom{n-3}{2} + 3.$$

Proof: The proof is by construction. Let N be an n-set. Let L be a 3-subset of N, and $R = N \setminus L$. Let $\mathcal{T} = [L]^2 \cup [R]^2$. It is easy to verify that $\mathcal{T}$ is an $(n, 3, 2)$-Turán system, and that $|\mathcal{T}| = \binom{n-3}{2} + 3$. ■

Theorem 4.37 *Let $\mathcal{C}$ be an $[n, n-3, 4]$ MDS code. Then*

$$\rho(\mathcal{C}) = S(n, 2).$$

Further, if $n \geq 6$, then we also have

$$\rho(\mathcal{C}) = T(n, 3, 2) = \left\lfloor \frac{n}{2} \right\rfloor \left(\left\lceil \frac{n}{2} \right\rceil - 1 \right).$$

Proof: The formula for $T(n, 3, 2)$ is a well-known result first documented by Mantel [22]. Later, Turán [7], [8] solved the more general case of $T(n, s, 2)$.

It is not hard to verify that for $n = 4$, $\rho(\mathcal{C}) = S(4, 2) = 3$, and for $n = 5$, $\rho(\mathcal{C}) = S(5, 2) = 5$. (In comparison, note that we have $T(4, 3, 2) = 2$, and $T(5, 3, 2) = 4$.)

Now, assume $n \geq 6$. To show $\rho(\mathcal{C}) = S(n, 2) = T(n, 3, 2)$, it suffices to show that $S(n, 2) \leq T(n, 3, 2)$. Let N be an n-set. Let $\mathcal{T} \subset [N]^2$ be an $(n, 3, 2)$-Turán system of smallest size. We show that $\mathcal{T}$ must also be an $(n, 2)$-SE system. Since $\mathcal{T}$ is an $(n, 3, 2)$-Turán system, all 3-sets are covered. It remains to show that all 1- and 2-sets are covered as well.

Suppose there is a 1-set, say $\{i\}$, that is not covered. Then i is contained in all blocks of $\mathcal{T}$. But this implies that all 3-subsets of $N \setminus \{i\}$ are not covered, contradicting the fact that $\mathcal{T}$ is a Turán $(n, 3, 2)$-system.

Suppose there is a 2-set, say $\{i, j\}$, that is not covered. This implies that blocks of $\mathcal{T}$ either are $\{i, j\}$, or are disjoint from $\{i, j\}$. Indeed, $\{i, j\}$ itself must be a block in $\mathcal{T}$, or any of its size-3 supersets would not be covered. All 2-sets disjoint from $\{i, j\}$ must also be blocks of $\mathcal{T}$. Otherwise, if some $\{a, b\} \subseteq N \setminus \{i, j\}$ is not contained in $\mathcal{T}$, then the 3-set $\{a, b, i\}$ would not be covered. All together, the above discussion shows that $T(n, 3, 2) = |\mathcal{T}| = \binom{n-2}{2} + 1$. But $\binom{n-2}{2} + 1 > \binom{n-3}{2} + 3$ for all $n \geq 6$, which contradicts Lemma 4.36. ■

Since the precise formula for $T(n, 3, 2)$ is known, Lemma 4.36 may seem unnecessary. It is nonetheless given here for its simplicity, and the fact that it suffices for the proof even if the expression for $T(n, 3, 2)$ were not known.

Now, consider $d = 5$. We first note a couple of bounds on $T(n, 4, 3)$.

Lemma 4.38
$$T(n,4,3) \leq \left\lfloor\frac{n}{3}\right\rfloor \left\lfloor\frac{n-1}{3}\right\rfloor \left(2\left\lfloor\frac{n-2}{3}\right\rfloor + 1\right),$$

where equality holds for $n \leq 13$*.*

Proof: The upper bound comes from a construction of Turán $(n,4,3)$-systems due to Ringel [23], which has been verified to be optimal for $n \leq 13$ (cf. [4]). ■

Lemma 4.39 *For* $n \geq 13$*,*
$$T(n,4,3) \geq \frac{56}{143}\binom{n}{3}.$$

Proof: It can be shown [5] that $T(n,s,t)/\binom{n}{t}$ is non-decreasing in n, hence for any given n_0,
$$T(n,s,t) \geq \frac{T(n_0,s,t)}{\binom{n_0}{t}}\binom{n}{t}, \quad \text{for } n \geq n_0.$$

Choose $n_0 = 13$. Since $T(13,4,3) = 112$ by Lemma 4.38, the result follows. ■

Theorem 4.40 *Let* $\mathcal{C}$ *be an* $[n, n-4, 5]$ *MDS code. Then*
$$\rho(\mathcal{C}) = S(n,3).$$

Further, if $n \geq 6$*, then it also holds that*
$$\rho(\mathcal{C}) = T(n,4,3).$$

Proof: For $n = 5$, it is easily verified that $\rho(\mathcal{C}) = S(5,3) = 4$. (In comparison, note that $T(5,4,3) = 3$.)

Now, it remains to show that $S(n,3) = T(n,4,3)$ for all $n \geq 6$. Let N be an n-set and $\mathcal{T} \subseteq [N]^3$ be a minimal $(n,4,3)$-Turán system. If $\mathcal{T}$ is also an $(n,3)$-SE system then we are done. Otherwise, let X be a *smallest* uncovered subset of N. Then $|X| = 1$, 2, or 3. (All 4-sets are covered since $\mathcal{T}$ is a Turán $(n,4,3)$-system.)

First, suppose $|X| = 1$. Since X is not covered, it is contained in all blocks of $\mathcal{T}$. Then any 4-subset of $N \setminus X$ is not covered. This is a contradiction.

Next, suppose $|X| = 2$, say $X = \{i, j\}$. Then any block of $\mathcal{T}$ either contains X or is disjoint from X. Out of the $n-2$ size-3 supersets of X, at least $n-3$ must be in $\mathcal{T}$. Otherwise we could find $a, b \in N \setminus X$ such that $\{i, j, a\}, \{i, j, b\} \notin \mathcal{T}$. But then the 4-set $\{i, j, a, b\}$ would not be covered. On the other hand, all of the $\binom{n-2}{3}$ 3-sets that are disjoint from X must be blocks of $\mathcal{T}$. Otherwise, if $\{a, b, c\} \subseteq N \setminus X$ is not a block, then $\{i, a, b, c\}$ would not be covered. All in all, we see that $\mathcal{T}$ contains at least $\binom{n-2}{3} + n - 3$ blocks. But since $\binom{n-2}{3} + n - 3 > \lfloor \frac{n}{3} \rfloor \lfloor \frac{n-1}{3} \rfloor \left(2 \lfloor \frac{n-2}{3} \rfloor + 1\right)$ for $n \geq 6$, this contradicts Lemma 4.38.

Lastly, suppose $|X| = 3$, say $X = \{i, j, k\}$. Then for all $Y \in \mathcal{T}$, $|Y \cap X| \neq 2$. Note the following facts:

1. X itself must be a block of $\mathcal{T}$, otherwise 4-sets like $\{i, j, k, a\}$ would not be covered.

2. For each 2-set $\{a, b\} \subseteq N \setminus X$, at least two of $\{a, b, i\}$, $\{a, b, j\}$, and $\{a, b, k\}$ must be blocks of $\mathcal{T}$. This is true because if, say, $\{a, b, i\}$ and $\{a, b, j\}$ both were not blocks of $\mathcal{T}$, then $\{a, b, i, j\}$ would not be covered.

3. All blocks that are disjoint from X form an $(n-3, 4, 3)$-Turán system.

Together, these imply that

$$S(n, 3) \geq T(n, 4, 3) = |\mathcal{T}| \geq T(n-3, 4, 3) + 2\binom{n-3}{2} + 1.$$

However, from Theorem 4.23,

$$S(n, 3) \leq T(n-1, 4, 3) + T(n-2, 3, 2) + T(n-3, 2, 1) + 1. \tag{4.26}$$

And since (cf. [3])

$$T(n, s, t) \leq T(n-1, s, t) + T(n-1, s-1, t-1),$$

we have

$$T(n-1, 4, 3) \leq T(n-3, 4, 3) + T(n-3, 3, 2) + T(n-2, 3, 2).$$

Plugging the above in (4.26), we obtain

$$S(n, 3) \leq T(n-3, 4, 3) + 2T(n-2, 3, 2) + T(n-3, 3, 2) + n - 3.$$

Putting the upper and lower bounds on $S(n,3)$ together, we have

$$2T(n-2,3,2)+T(n-3,3,2)+n-3 \geq 2\binom{n-3}{2}+1.$$

However, since it is known (cf. [22] [8]) that $T(n,3,2)=\lfloor n/2\rfloor(\lceil n/2\rceil-1)$, the above inequality results in a contradiction for all $n \geq 9$.

For $n=6,7,8$, a different contradiction results directly from

$$T(n,4,3)=|\mathcal{T}| \geq T(n-3,4,3)+2\binom{n-3}{2}+1,$$

in relation to Lemma 4.38 and Lemma 4.39. ■

Corollary 4.41 *Let $\mathcal{C}$ be an $[n,n-4,5]$ MDS code. Then*

$$\rho(\mathcal{C})=\left\lfloor\frac{n}{3}\right\rfloor\left\lfloor\frac{n-1}{3}\right\rfloor\left(2\left\lfloor\frac{n-2}{3}\right\rfloor+1\right), \quad \textit{for } n=6,\ldots,13.$$

Our approach regarding the conjecture has been so far to show that in some cases the upper and lower bounds on stopping redundancy (SE and Turán numbers, respectively) converge, either exactly or asymptotically. However, we have seen that in other cases the corresponding SE and Turán numbers can be provably different, even in the asymptotic sense (for example, when k is a fixed constant), which shows the limitation of the current approach in fully resolving the conjecture.

In fact, it is our belief that for an $[n,n-d+1,d]$ MDS code $\mathcal{C}$,

$$\rho(\mathcal{C})=S(n,d-2),$$

the proof of which would in turn prove the Schwartz-Vardy conjecture. We have shown that this is true (or close to being true) when either d or k is $o(\sqrt{n})$. A reasonable question to ask is: what if d and k are both larger than $o(\sqrt{n})$? For example, what if k/n approaches a constant? The current approach only bounds $\rho(\mathcal{C})$ to within a factor of up to $\ln n$. For example, using the result of Theorem 4.24, we have

$$\rho(\mathcal{C}) \leq \frac{n-d+2}{n-d+1}\cdot\frac{1+\ln(n-d+1)}{d-1}\cdot\binom{n}{d-2},$$

while, for the lower bound, we saw that

$$\rho(\mathcal{C}) \geq \frac{1}{d-1}\binom{n}{d-2}.$$

Alternatively, let's make the following observation. Suppose we are given one optimal parity-check matrix, i.e. one with $\rho(\mathcal{C})$ rows that maximizes stopping distance. It is not apparent that all rows should have minimum weight, but suppose T' rows are of minimum weight and the rest are not. We can replace each row that is not of minimum weight (and whose weight is, of course, at most n) with no more than $\lceil n/(n-d+2) \rceil$ minimum-weight rows, such that the union of supports of these rows is precisely the support of the row they replaced. It is simple to verify that the replacement procedure does not decrease the stopping distance, which also implies that the rank of the matrix is not reduced. After all rows that are not of minimum weight have been replaced, we obtain a parity-check matrix with at most

$$T' + \left\lceil \frac{n}{n-d+2} \right\rceil \bigl(\rho(\mathcal{C}) - T'\bigr)$$

rows, all having minimum weight, that achieves maximum stopping distance. Therefore,

$$T' + \left\lceil \frac{n}{n-d+2} \right\rceil \bigl(\rho(\mathcal{C}) - T'\bigr) \geq S(n, d-2).$$

Now note that

$$T' \geq T(n, d-1, d-2),$$

so we have

$$\rho(\mathcal{C}) \geq \lambda \cdot S(n, d-2) + (1-\lambda) \cdot T(n, d-1, d-2),$$

where

$$\lambda = \lambda(n, d) = 1 \Bigg/ \left\lceil \frac{n}{n-d+2} \right\rceil .$$

Without knowing better how $T(n, d-1, d-2)$ compares with $S(n, d-2)$, if we just ignore the second term, we obtain

$$\lambda \cdot S(n, d-2) \leq \rho(\mathcal{C}) \leq S(n, d-2).$$

This shows that in many cases $S(n, d-2)$ is a good estimate of $\rho(\mathcal{C})$. For example, if the code rate $R = (n-d+1)/n \geq 1/2$, then

$$\frac{1}{2} S(n, d-2) \leq \rho(\mathcal{C}) \leq S(n, d-2).$$

And, clearly, for any constant code rate, $\rho(\mathcal{C})$ is within a constant factor of $S(n, d-2)$.

4.5 Concluding Remarks

We have attacked the problem of stopping redundancy of MDS codes from a combinatorial perspective, by introducing the single-exclusion (SE) number as a proper upper bound on the stopping redundancy. We have seen that this proved to be a powerful technique that has led to much stronger bounds on the stopping redundancy of MDS codes. In some cases, we have even been able to find the value of $\rho(\mathcal{C})$, either precisely or in an asymptotic sense, as a function of n and d, thus proving the Schwartz-Vardy conjecture that the stopping redundancy of an MDS code should only depend on its length and minimum distance.

While we have obtained a fairly good understanding of the stopping redundancy of MDS codes as well as the SE number, many interesting questions remain unanswered. For example, what is the asymptotic value of $S(n, n-k-1)$ for a fixed k? (According to [24], it can be shown that $S(n, n-k-1)$ is asymptotic to $\frac{1}{k+1/2}\binom{n}{k}$, i.e. the asymptotic lower bound in Corollary 4.34 is sharp.) And how does $S(n,t)$ compare with $T(n, t+1, t)$ in general? (Do they differ by at most a constant factor?) Finally, is it true that the stopping redundancy of an $[n, n-d+1, d]$ MDS code equals $S(n, d-2)$?

Clearly, besides their application to the stopping redundancy of MDS codes, SE systems warrant further study for their intrinsic mathematical appeal and interest. They also have practical relevance; for example, the definition of SE system can be readily mapped to a problem in the design of experiments. We hope that our results on SE system obtained in this chapter will become useful in other settings.

Appendix 4.A Asymptotics of (4.10) for $t < n - \ln n$

Rewrite (4.10) as

$$\eta(n,t) = \frac{n-t}{t+1} \cdot \sum_{i=1}^{t+1} f(i) \tag{4.27}$$

where

$$f(i) = \frac{\binom{t+1}{i}}{\binom{n-i}{n-t-1}} \cdot (n-t)^{-\frac{i}{t}\binom{n-i}{n-t-1}}$$

Assuming

$$n - t > \ln n,$$

we show that the $(t+1)$-st term,

$$f(t+1) = (n-t)^{-(t+1)/t},$$

prevails in the sum of (4.27).

It suffices to show that each of the other t terms is a $o(1/t)$ fraction of $f(t+1)$. For $i = t$, it is easy to verify that

$$f(t) = \frac{t+1}{n-t} \cdot (n-t)^{-(n-t)} = o(1/t)f(t+1).$$

In general, we have

$$\frac{f(i)}{f(t+1)} = \frac{\binom{t+1}{i}}{\binom{n-i}{n-t-1}} \cdot (n-t)^{-\frac{i}{t}\binom{n-i}{n-t-1}+\frac{t+1}{t}}. \tag{4.28}$$

For $i = 1, \ldots, t-1$, consider two cases. For $i \leq 2n/\ln n = o(n)$, we have

$$\binom{n-i}{n-t-1} \geq \binom{n(1-2/\ln n)}{2} = \Omega(n^2).$$

In this case, (4.28) decreases super-exponentially with n, and is certainly $o(1/t)$. On the other hand, for $2n/\ln n < i \leq t-1$, we show that $f(i) < f(t) = o(1/t)f(t+1)$, by showing that $f(i)$ is monotonically increasing for $2n/\ln n < i \leq t$. Indeed, for $2n/\ln n < i \leq t-1$, we have

$$\begin{aligned}
\frac{f(i+1)}{f(i)} &= \frac{n-i}{i+1} \cdot (n-t)^{-\frac{i+1}{t}\binom{n-i-1}{n-t-1}+\frac{i}{t}\binom{n-i}{n-t-1}} \\
&= \frac{n-i}{i+1} \cdot (n-t)^{\frac{1}{t}\left(i-\frac{t-i+1}{n-t-1}\right)\binom{n-i-1}{n-t-2}} \\
&> \frac{\ln n}{n} \cdot (\ln n)^{\frac{1}{t}\left(\frac{2n}{\ln n}-\frac{n}{\ln n}\right)\binom{\ln n}{2}} \\
&> \left(\frac{\sqrt{\ln n}}{e}\right)^{\ln n} \\
&> 1.
\end{aligned}$$

Appendix 4.B Asymptotics of (4.5) for $t \geq n - \ln n$

Let $k = n-t-1$, and $j = t+1-i$. The right-hand side of (4.5) becomes

$$p\binom{n}{k+1} + \sum_{j=0}^{t} \binom{n}{k+j} (1-p)^{(n-k-j)\binom{k+j}{k}}. \tag{4.29}$$

Let $p = (\ln n)/n$. Assume that

$$1 < k = o\left(\frac{n \ln\ln n}{\ln n}\right)$$

($n - \ln n \le t < n - 2$ being a special case). We show that the first term in (4.29) prevails. Note that

$$\begin{aligned}\left(1 - \frac{\ln n}{n}\right)^{(n-k-j)\binom{k+j}{k}} &\le e^{-\frac{\ln n}{n}\cdot(n-k-j)\binom{k+j}{k}} \\ &= n^{-\frac{n-k-j}{n}\binom{k+j}{k}}.\end{aligned}$$

For $1 \le j \le n\sqrt{2/\ln n}$, we have

$$\frac{n-k-j}{n}\binom{k+j}{k} \ge \big(1 - o(1)\big)\binom{j+2}{2} > j + 1.$$

For $j > n\sqrt{2/\ln n}$, we have

$$\frac{\ln n}{n} \cdot (n-k-j)\binom{k+j}{k} \ge \frac{\ln n}{n}\binom{j+2}{2} > n.$$

Noting further that for $j \ge 1$,

$$\binom{n}{k+j} \le \binom{n}{k+1} \cdot n^{j-1},$$

we see that the second term in (4.29) is at most

$$\begin{aligned}
&\binom{n}{k} \cdot n^{-\frac{n-k}{n}} + \sum_{j=1}^{\left\lfloor n\sqrt{\frac{2}{\ln n}}\right\rfloor} \binom{n}{k+1} \cdot n^{j-1} \cdot n^{-\frac{n-k-j}{n}\binom{k+j}{k}} \\
&+ \sum_{j=\left\lfloor n\sqrt{\frac{2}{\ln n}}\right\rfloor+1}^{r} \binom{n}{k+j} \cdot e^{-\frac{\ln n}{n}\cdot(n-k-j)\binom{k+j}{k}} \\
&\le \binom{n}{k+1} \cdot n^{-1+\frac{k}{n}} + \binom{n}{k+1} \sum_{j=1}^{\left\lfloor n\sqrt{\frac{2}{\ln n}}\right\rfloor} n^{-2} \\
&\quad + e^{-n} \sum_{j=\left\lfloor n\sqrt{\frac{2}{\ln n}}\right\rfloor+1}^{r} \binom{n}{k+j} \\
&\le \frac{1}{n}\binom{n}{k+1}\left(e^{\frac{k\ln n}{n}} + \sqrt{\frac{2}{\ln n}}\right) + 2^n \cdot e^{-n} \\
&= \frac{1}{n}\binom{n}{k+1}\Big(o(\ln n) + o(1)\Big) + o(1) \\
&= o\left(\frac{\ln n}{n}\binom{n}{k+1}\right).
\end{aligned}$$

Hence, at $p = (\ln n)/n$, the upper bound (4.5) is asymptotic to $\frac{\ln n}{n}\binom{n}{t}$ for all t such that $2 < n - t = o\big((n \ln\ln n)/\ln n\big)$, which implies that for $2 < n - t = o\big((n \ln\ln n)/\ln n\big)$, the upper bound (4.5), when minimized over p, is $O\big(\frac{\ln n}{n}\binom{n}{t}\big)$.

Note that the $O\big(\frac{\ln n}{n}\binom{n}{t}\big)$ estimate is not always tight. For example, when $t \approx n - \ln n$, we have shown using a different analysis that the upper bound (4.5), when minimized over p, is in fact $O\big(\frac{\ln\ln n}{n}\binom{n}{t}\big)$. However, note that by keeping just the $i = t$ term in the sum, the upper bound (4.5) is at least

$$\begin{aligned}
&\Big(p + (1-p)^{t(n-t)}\Big)\binom{n}{t} \\
&\ge \Big(1 - \big(1 - 1/a\big) \cdot a^{-1/(a-1)}\Big)\binom{n}{t} \\
&= \left[\frac{\ln a}{a} + O\left(\frac{1}{a}\right)\right]\binom{n}{t},
\end{aligned}$$

where $a = t(n-t)$. In particular, this shows that the $O\big(\frac{\ln n}{n}\binom{n}{t}\big)$ estimate is tight if $n - t$ is $\Theta(1)$. That is, the bound (4.5) is $\Theta\big(\frac{\ln n}{n}\binom{n}{t}\big)$ when minimized over p, for all $t = n - \Theta(1)$.

Acknowledgment

This chapter is in part a reprint of the material in the papers: J. Han and P. H. Siegel, "On the stopping redundancy of MDS codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Seattle, WA, July 2006, pp. 2491–2495, J. Han and P. H. Siegel, "Improved upper bounds on stopping redundancy," *IEEE Trans. Inform. Theory*, vol. 53, no. 1, pp. 90–104, Jan. 2007, J. Han, P. H. Siegel, and R. M. Roth, "Bounds on single-exclusion numbers and stopping redundancy of MDS codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Nice, France, June 2007, pp. 2941–2945, and J. Han, P. H. Siegel, and R. M. Roth, "Single-exclusion number and the stopping redundancy of MDS codes," *IEEE Trans. Inform. Theory*, submitted for publication. The dissertation author was the primary author of the above papers.

Bibliography

[1] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1978.

[2] M. Schwartz and A. Vardy, "On the stopping distance and the stopping redundancy of codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.

[3] A. Sidorenko, "Upper bounds for Turán numbers," *J. Combin. Theory Ser. A*, vol. 77, pp. 134–147, 1997.

[4] W. H. Mills and R. C. Mullin, "Coverings and packings," in *Contemporary Design Theory*, J. H. Dinitz and D. R. Stinson, Eds. New York: Wiley, 1992, ch. 9, pp. 371–399.

[5] G. Katona, T. Nemetz, and M. Simonovits, "On a graph problem of Turán," *Mat. Lapok*, vol. 15, pp. 228–238, 1964, (in Hungarian).

[6] F. Chung and R. Graham, *Erdős on Graphs – His Legacy of Unsolved Problems*. Wellesley, Massachusetts: A K Peters, 1998.

[7] P. Turán, "Egy gráfelméleti szélsőértékfeladatról," *Mat. Fiz. Lapok*, vol. 48, pp. 436–452, 1941, (in Hungarian).

[8] P. Turán, "An extremal problem in graph theory," in *Collected Papers of Paul Turán*, P. Erdős, Ed. Budapest: Akadémiai Kiadó, 1990, pp. 231–256.

[9] F. Chung and L. Lu, "An upper bound for the Turán number $t_3(n,4)$," *J. Combin. Theory Ser. A*, vol. 87, pp. 381–389, 1999.

[10] D. de Caen, D. L. Kreher, and J. Wiseman, “On constructive upper bounds for the Turán numbers $T(n, 2r+1, 2r)$,” *Congr. Numer.*, vol. 65, pp. 277–280, 1988.

[11] A. F. Sidorenko, “Systems of sets that have the T-property,” *Moscow Univ. Math. Bull.*, vol. 36, no. 5, pp. 22–26, 1981.

[12] ——, “The method of quadratic forms and Turán’s combinatorial problem,” *Moscow Univ. Math. Bull.*, vol. 37, no. 1, pp. 1–5, 1982.

[13] D. Applegate, E. M. Rains, and N. J. A. Sloane, “On asymmetric coverings and covering numbers,” *J. Combin. Des.*, vol. 11, pp. 218–228, 2003.

[14] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*, 2nd ed. Reading, MA: Addison-Wesley, 1994.

[15] K. H. Kim and F. W. Roush, “On a problem of Turán,” in *Studies in Pure Mathematics: To the Memory of Paul Turán*, P. Erdős, Ed. Basel: Birkhäuser Verlag, 1983, pp. 423–425.

[16] P. Frankl and V. Rödl, “Lower bounds for Turán’s problem,” *Graphs Combin.*, vol. 1, pp. 213–216, 1985.

[17] P. Erdős and J. H. Spencer, *Probabilistic Methods in Combinatorics*. Academic Press, 1974.

[18] N. N. Kuzjurin, “Minimal coverings and maximal coverings of $(k-1)$-subsets by k-subsets,” *Matematicheskie Zametki*, vol. 21, no. 4, pp. 565–571, Apr. 1977, (in Russian).

[19] J. Schönheim, “On coverings,” *Pacific J. Math.*, vol. 14, pp. 1405–1411, 1964.

[20] D. de Caen, “Extension of a theorem of Moon and Moser on complete subgraphs,” *Ars Combin.*, vol. 16, pp. 5–10, 1983.

[21] V. Rödl, “On a packing and covering problem,” *European J. Combin.*, vol. 5, pp. 69–78, 1985.

[22] W. Mantel, “Vraagstuk XXVIII,” *Wiskundige Opgaven met de Oplossingen*, vol. 10, pp. 60–61, 1907.

[23] G. Ringel, “Extremal problems in the theory of graphs,” in *Theory of Graphs and Their Applications*, M. Fiedler, Ed. Prague: Czechoslovak Academy of Sciences, 1964.

[24] O. Milenkovic and Z. Furedi, private communication, 2008.

Chapter 5

Stopping Redundancy of Reed-Muller Codes

The binary Reed-Muller (RM) codes [1] are among the first few classes of codes whose stopping redundancy has been carefully studied. Let $\mathcal{R}(l, m)$ denote the l-th order Reed-Muller code with parameter m. Schwartz and Vardy [2] showed that

$$\rho\big(\mathcal{R}(l, m)\big) \leq \sum_{i=0}^{m-l-1} \binom{l+i}{i} 2^i.$$

The stopping redundancy of RM codes was further studied by Etzion [3]. In both papers, the arguments were based on the recursive construction of RM codes, using an elaborate analysis of a specific recursion of parity-check matrices.

In this chapter, we rediscover a number of results in [2] and [3] on the stopping redundancy of RM codes using a geometric approach, which we feel is more intuitive. A couple of new bounds based on probabilistic methods are also presented.

5.1 Main Results

Recall that $\mathcal{R}(l, m)$ has length 2^m, dimension $\sum_{i=0}^{l} \binom{m}{i}$, minimum distance 2^{m-l}, and its dual code is $\mathcal{R}(l, m)^{\perp} = \mathcal{R}(m-l-1, m)$. $\mathcal{R}(l, m)$ is generated by its minimum-weight codewords, which can be interpreted as incidence vectors of $(m-l)$-flats in affine geometry $\mathrm{AG}(m, 2)$. For more information about RM codes, the reader is referred to [1].

First, we show the following result.

Theorem 5.1

$$\rho\big(\mathcal{R}(l,m)\big) \leq \binom{m}{l+1} 2^{m-l-1}$$

Proof: Let S be a set of coordinates with $|S| < 2^{m-l}$. We think of S as a set of points in $\mathbb{F}_2^m$. Let $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m\}$ be a basis of $\mathbb{F}_2^m$. We show that there exists an $(l+1)$-flat defined by $m-l-1$ linear equations in the form $\boldsymbol{x}\boldsymbol{a}_{i_j}^T = b_j$, $b_j \in \mathbb{F}_2$, $j = 1, \ldots, m-l-1$, that intersects S at a single point. (Recall that any $(l+1)$-flat is a minimum weight codeword in $\mathcal{R}(l,m)^\perp$.) For that, it suffices to show that S has a single point contained in a t-flat defined by $\boldsymbol{x}\boldsymbol{a}_{i_j}^T = b_j, j = 1, \ldots, m-t$ where $t \geq l+1$. Indeed, if that single point is denoted by $\boldsymbol{s}$, then we can choose any $t-l-1$ more basis vectors $\boldsymbol{a}_{i_{m-t+1}}, \ldots, \boldsymbol{a}_{i_{m-l-1}}$ and append the equations $\boldsymbol{x}\boldsymbol{a}_{i_j}^T = \boldsymbol{s}\boldsymbol{a}_{i_j}^T, j = m-t+1, \ldots, m-l-1$. This specifies an $(l+1)$-flat contained in the t-flat, that also contains $\boldsymbol{s}$.

For any set of points P with $|P| > 1$, we say that a vector $\boldsymbol{a}$ *cuts* P if the inner products of $\boldsymbol{a}$ with points in P yield both 0 and 1. The word "cut" is chosen for the fact that in this case the two hyperplanes $\{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\boldsymbol{a}^T = 0\}$ and $\{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\boldsymbol{a}^T = 1\}$ partition P into two nonempty subsets. Referring to this fact, we will also say that $\boldsymbol{a}$ cuts P into two subsets, i.e. $\{\boldsymbol{x} \in P : \boldsymbol{x}\boldsymbol{a}^T = 0\}$ and $\{\boldsymbol{x} \in P : \boldsymbol{x}\boldsymbol{a}^T = 1\}$.

Now, if S itself contains just a single point, then we are done. If S contains more than one point, then there exists some $\boldsymbol{a}_{i_1}$ that cuts S, otherwise all points of S must satisfy m independent linear equations, contradicting the fact that $|S| > 1$. Thus $\boldsymbol{a}_{i_1}$ cuts S into two subsets. Choose the smaller subset and call it S_1. Now note that either $|S_1| = 1$, or else there exists a vector $\boldsymbol{a}_{i_2} \in \{\boldsymbol{a}_i\}_{i \neq i_1}$ that cuts S_1 into two subsets. Pick the smaller subset and call it S_2. The above process can be iterated until after some p steps, we have $|S_p| = 1$. Since $|S| < 2^{m-l}$ and at least half of the points are removed in each step, we have $p \leq m-l-1$. Therefore, we have found an $(m-p)$-flat with $m-p \geq l+1$, defined by $\boldsymbol{x}\boldsymbol{a}_{i_j}^T = b_j, j = 1, \ldots, p$ where b_j's are constants in $\mathbb{F}_2$, which intersects S at a single point.

Finally we need to count the number of $(l+1)$-flats for any given set of basis vectors, which is simply

$$\binom{m}{m-l-1} 2^{m-l-1}.$$

■

The above theorem is the most natural upper bound on $\rho\big(\mathcal{R}(l,m)\big)$ that can be obtained following this "set-cutting" idea. The bound can be improved by noting that we do not need to choose *all* $(l+1)$-flats defined by the set of basis vectors, as made precise in the following theorem.

Theorem 5.2

$$\rho\big(\mathcal{R}(l,m)\big) \leq \sum_{i=0}^{m-l-1} \binom{l+i}{l} 2^i$$

Proof: It is clear from the proof of the previous theorem that we are simply looking for a set of up to $m-l-1$ vectors from a basis of $\mathbb{F}_2^m$, that cut S down to a single point, where S is any set of points (binary m-tuples) with $|S| < 2^{m-l}$.

Let $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m\}$ denote the basis in question. If we can cut S down to a single point without using $\boldsymbol{a}_1$, then that only requires choosing the

$$\binom{m-1}{m-l-1} 2^{m-l-1}$$

$(l+1)$-flats defined by vectors in $\{\boldsymbol{a}_2, \ldots, \boldsymbol{a}_m\}$. If $\boldsymbol{a}_1$ has to be used, we claim that we can stick to using $\boldsymbol{x}\boldsymbol{a}_1^T = 0$. (The 0 can be replaced by a 1. The point is that we do not need to allow both.) Indeed, when we cannot cut down the point set any more without using $\boldsymbol{a}_1$, the remaining points must be a line defined by $\boldsymbol{x}\boldsymbol{a}_i^T = b_i$, $i = 2, \ldots, m$, for some $b_i \in \mathbb{F}_2$. Hence, we can cut the set down to a single point using $\boldsymbol{a}_1$, and it does not matter what constant term is used. This results in an additional

$$\binom{m-1}{m-l-2} 2^{m-l-2}$$

$(l+1)$-flats to be chosen.

The above idea can be carried further, so when $\boldsymbol{a}_1$ has to be used and $\boldsymbol{x}\boldsymbol{a}_1^T = 0$ is chosen, we further differentiate between whether $\boldsymbol{a}_2$ has to be used, or not. If $\boldsymbol{a}_2$ does not have to be used, then we just need to choose $m-l-2$ vectors from $\{\boldsymbol{a}_3, \ldots, \boldsymbol{a}_m\}$ and the corresponding constants, to be appended to the equation $\boldsymbol{x}\boldsymbol{a}_1^T = 0$, that is

$$\binom{m-2}{m-l-2} 2^{m-l-2}$$

$(l+1)$-flats. On the other hand, if we also must use $\boldsymbol{a}_2$, then for the same reason as before, it does not matter if we use $\boldsymbol{x}\boldsymbol{a}_2^T = 0$ or $\boldsymbol{x}\boldsymbol{a}_2^T = 1$. So we can just stick to using one of them. This results in an additional

$$\binom{m-2}{m-l-3}2^{m-l-3}$$

$(l+1)$-flats. Clearly, the case where both $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ have to be used can be further broken down according to whether $\boldsymbol{a}_3$ has to be used, and so on. If this is done to the first t basis vectors, then it shows that $\rho\big(\mathcal{R}(l,m)\big)$ is at most

$$\sum_{i=1}^{t}\binom{m-i}{m-l-i}2^{m-l-i}+\binom{m-t}{m-l-t-1}2^{m-l-t-1}$$

It's not hard to see that this function is nonincreasing in t. For the best result, we choose $t \geq m-l-1$ and the claimed bound is obtained after simple algebra. ■

The above theorem gives the same result as was shown in [4] but through a different approach. Note that the two proofs are related. In fact, one can show that the parity-check matrix used in the above proof and the one constructed in [4] are essentially the same (up to row and column permutations). For first-order RM codes, the above result shows that $\rho\big(\mathcal{R}(1,m)\big) \leq (m-2)2^{m-1}+1$. This has been improved to $\rho\big(\mathcal{R}(1,m)\big) \leq \big((6m-7)2^{m-1}+(-1)^{m-1}\big)/9$ in [3]. For extended Hamming codes, the above result shows that $\rho\big(\mathcal{R}(m-2,m)\big) \leq 2m-1$. This bound has been shown [3] to be sharp, i.e. $\rho\big(\mathcal{R}(m-2,m)\big) = 2m-1$. We now give an alternative (and shorter) proof of this fact.

Theorem 5.3

$$\rho\big(\mathcal{R}(m-2,m)\big) = 2m-1.$$

Proof: It suffices to show that $\rho\big(\mathcal{R}(m-2,m)\big) \geq 2m-1$. Besides the all-zero and all-one vectors, every codeword in the dual code $\mathcal{R}(1,m)$ is the incidence vector of a hyperplane. Clearly, to use the minimum number of checks, only those corresponding to hyperplanes need to be considered. It suffices to show that we cannot find a set of $2m-2$ checks that cover all sets of size up to 3, that is, that we cannot find a set of $2m-2$ hyperplanes such that any set of up to 3 points intersects a hyperplane at a single point.

Suppose we could, then let the $2m-2$ hyperplanes be represented as

$$\{\boldsymbol{x}\in\mathbb{F}_2^m : \boldsymbol{x}\boldsymbol{a}_i^T = b_i\}, \quad i=1,\ldots,2m-2.$$

First, note that $\{\boldsymbol{a}_i\}_{i=1}^{2m-2}$ must contain a basis of $\mathbb{F}_2^m$. Otherwise, choose a maximal independent set out of $\{\boldsymbol{a}_i\}_{i=1}^{2m-2}$, then the intersection of the corresponding hyperplanes is a coset of dimension at least one. Any line in this coset contains two points and is not covered. So let's assume that $\{\boldsymbol{a}_i\}_{i=1}^{2m-2}$ contains a basis. WLOG, assume that $\{\boldsymbol{a}_i\}_{i=1}^{m}$ is a basis of $\mathbb{F}_2^m$. Consider the set of equations

$$\boldsymbol{x}\boldsymbol{a}_i^T = b_i + 1, \quad i = 1, \ldots, m,$$

which can be written in matrix form as

$$\boldsymbol{x}A^T = \boldsymbol{b}^c, \tag{5.1}$$

where

$$A = \begin{pmatrix} \boldsymbol{a}_1 \\ \vdots \\ \boldsymbol{a}_m \end{pmatrix},$$

$$\boldsymbol{b}^c = (b_1 + 1, \ldots, b_m + 1).$$

Since $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m\}$ is a basis, there exists an $m \times m$ full-rank matrix K, such that $\boldsymbol{a}_{m+1}, \ldots, \boldsymbol{a}_{2m-2}$ are contained in the linear span of the first $m-2$ rows of KA. Multiplying both sides of (5.1) by K^T, we obtain

$$\boldsymbol{x}\Lambda^T = \boldsymbol{\beta},$$

where $\Lambda = KA$, $\boldsymbol{\beta} = \boldsymbol{b}^c K^T$. Let Λ_1 denote the first $m-2$ rows of Λ, and $\boldsymbol{\beta}_1$ denote the first $m-2$ elements of $\boldsymbol{\beta}$. Note that

$$\{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\Lambda_1^T = \boldsymbol{\beta}_1\}$$

defines a two-flat. If we remove from it the point satisfying $\boldsymbol{x}\Lambda^T = \boldsymbol{\beta}$, we obtain a set, S, of size three. That is, let

$$S = \{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\Lambda_1^T = \boldsymbol{\beta}_1, \boldsymbol{x}\Lambda^T \neq \boldsymbol{\beta}\}.$$

We claim that none of the $2m-2$ hyperplanes intersect S at a single point, thus resulting in a contradiction. This is not obvious only for those hyperplanes $\{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\boldsymbol{a}_i^T = b_i\}$, $i = 1, \ldots, m$, such that $\boldsymbol{a}_i$ is linearly independent of the rows of Λ_1. In such cases, note that $\{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\boldsymbol{a}_i^T = b_i\}$ intersects the two-flat $\{\boldsymbol{x} \in \mathbb{F}_2^m : \boldsymbol{x}\Lambda_1^T = \boldsymbol{\beta}_1\}$ at two points. It remains to show that both points are contained in S. Indeed, since both points satisfy $\boldsymbol{x}\boldsymbol{a}_i^T = b_i$, they both satisfy $\boldsymbol{x}A^T \neq \boldsymbol{b}^c$ and hence $\boldsymbol{x}\Lambda^T \neq \boldsymbol{\beta}$. ■

Next, let's try to combine probabilistic methods with the use of designs. A byproduct of our first theorem is that for RM codes, it is sufficient to choose minimum weight parity checks in order to maximize the stopping distance. Now, consider $\mathcal{R}(1,m)$. Its dual code is $\mathcal{R}(m-2,m)$, whose minimum weight codewords form a $(3,4,2^m)$ Steiner system. Let us choose each minimum weight codeword in $\mathcal{R}(m-2,m)$ independently with probability p. If S is set of coordinates with $|S| < 2^{m-1}$, then the probability that S is not covered at this point is

$$(1-p)^{f(S)},$$

where $f(S)$ denotes the number of blocks in the Steiner system that intersect S at a single point. We need to estimate or bound $f(S)$. Let $n = 2^m$. If S contains just a single point, then clearly

$$f(S) = \frac{1}{3}\binom{n-1}{2}.$$

If S contains two points, we need to add together the number of blocks that contain each point, then subtract off two times the number of blocks that contain both points. Hence,

$$f(S) = \frac{2}{3}\binom{n-1}{2} - n + 2.$$

If S contains three points, then we need three times the number of blocks that contain any particular point, minus two times $\binom{3}{2}$ times the number of blocks that contain any two given points, and add back three times the number of blocks that contain all three points.

$$f(S) = \binom{n-1}{2} - 2\binom{3}{2}\frac{n-2}{2} + 3.$$

Now it is rather clear that in general,

$$f(S) = \frac{|S|}{3}\binom{n-1}{2} - 2\binom{|S|}{2}\frac{n-2}{2} + 3\binom{|S|}{3} - 4g(S),$$

where $g(S)$ denotes the number of blocks contained in S. Except for $g(S)$, all terms only depend on $|S|$. Unfortunately $g(S)$ apparently depends on the actual choice of S. However, note that $g(S)$ is at most the size of a $(3,4,|S|)$ Steiner system (assuming one exists), i.e.

$$g(S) \leq \frac{1}{4}\binom{|S|}{3}.$$

Therefore, if we let

$$h(i) = \frac{i}{3}\binom{n-1}{2} - (n-2)\binom{i}{2} + 2\binom{i}{3},$$

then

$$f(S) \geq h(|S|).$$

Theorem 5.4

$$\rho\big(\mathcal{R}(1,m)\big) \le \min_{0\le p\le 1}\left\{\frac{p}{4}\binom{2^m}{3} + \sum_{i=1}^{2^{m-1}-1}\binom{2^m}{i}(1-p)^{h(i)}\right\} + 2^{m-1} - m.$$

Proof (sketch): The result is shown by using probabilistic methods in expectation. The additional term $2^{m-1} - m$ is to ensure that the parity check matrix has maximum rank. ■

Unfortunately, this bound is rather weak. One possibility for improvement is to pre-select a basis and then apply the probabilistic method. Also a better bound on $g(S)$ might be helpful.

Let's pursue the first idea. Suppose we pre-selected a basis. We claim that if $2^{i-1} \le |S| < 2^i$, then there exist i or more $(m-i+1)$-flats that intersect S at a single point.

If $|S| > 2^{i-1}$, then S must be cut by at least i vectors in the basis. (Otherwise S would be contained in an $(i-1)$-flat.) On the other hand, since $|S| < 2^i$, S can be cut down to a single point in at most $i-1$ steps. Let $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m\}$ denote the selected basis. WLOG, suppose $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_i$ cut S. Take all $(i-1)$-subsets of $\{\boldsymbol{a}_j\}_{j=1}^i$. If we can cut S down to a single point by using vectors in each of the i subsets, then we are done. Otherwise, it implies that to cut S down to a single point, some $\boldsymbol{a}_t$ has to be used, and hence we can choose to stay in either of the hyperplanes defined by $\boldsymbol{a}_t$. There are $(i-1)$ $(i-1)$-subsets of $\{\boldsymbol{a}_j\}_{j=1}^i$ that contain $\boldsymbol{a}_t$. By fixing the equation $\boldsymbol{x}\boldsymbol{a}_t^T = 0$ or $\boldsymbol{x}\boldsymbol{a}_t^T = 1$, we get a total number of $2(i-1)$ possibilities to potentially cut S down to a single point. If all of them are successful, then we are done ($2(i-1) \ge i$). Otherwise, it implies that given $\boldsymbol{x}\boldsymbol{a}_t^T = 0$ (or $\boldsymbol{x}\boldsymbol{a}_t^T = 1$), to cut S down to a single point, some $\boldsymbol{a}_p$ has to be used, and hence we can choose to stay in either of the hyperplanes defined by $\boldsymbol{a}_p$. Now we can further fix $\boldsymbol{x}\boldsymbol{a}_p^T = 0$ or $\boldsymbol{x}\boldsymbol{a}_p^T = 1$, and continue. Note that the number of "options" never decrease and in the end all options have to be successful. Since we started with i options, we have at least i $(m-i+1)$-flats constructed at the end.

In the special case where $|S| = 2^{i-1}$, if there are i or more vectors that cut S then the above reasoning applies. Otherwise, S must be an $(i-1)$-flat, to be cut by precisely $i-1$ vectors in the basis. Any coset of the subspace orthogonal to these $i-1$ vectors intersects S at a single point, and there are $2^{i-1} \ge i$ of them.

Now, recall that we are looking for $(l+1)$-flats, which can be constructed by appending

$$m - i + 1 - l - 1 = m - l - i$$

linear constraints to the defining equations of the $(m-i+1)$-flats. Recall that in constructing the $(m-i+1)$-flats that intersect S at a single point, we have chosen from no more than i basis vectors. Now, the $m-l-i$ additional equations can be chosen from the rest of the basis vectors. This results in a total of at least

$$i\binom{m-i}{m-l-i} = i\binom{m-i}{l}$$

$(l+1)$-flats that intersect S at a single point, and it is simple to verify that they are all distinct.

Applying probabilistic methods to the restricted set of $\binom{m}{l+1}2^{m-l-1}$ $(l+1)$-flats corresponding to any given basis of $\mathbb{F}_2^m$, we immediately obtain the following result.

Theorem 5.5

$$\rho\big(\mathcal{R}(l,m)\big) \leq \min_{0\leq p\leq 1}\left\{ p\binom{m}{l+1}2^{m-l-1} + \sum_{i=1}^{2^{m-l}-1}\binom{2^m}{i}(1-p)^{\alpha(i)} \right\} + r - 2^{m-l} + 1,$$

where

$$\alpha(i) = \lceil \log(i+1) \rceil \binom{m - \lceil \log(i+1) \rceil}{l},$$

$$r = \sum_{i=0}^{m-l-1}\binom{m}{i}.$$

The above bound is unfortunately not a significant improvement over that obtained using the whole restricted set we started with. It appears that $\alpha(i)$ is still too small so that p has to be very close to 1 in order to drive down the second term, so the minimum value for the first two terms is very close to $\binom{m}{l+1}2^{m-l-1}$.

Bibliography

[1] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1978.

[2] M. Schwartz and A. Vardy, “On the stopping distance and stopping redundancy of codes,” in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 975–979.

[3] T. Etzion, “On the stopping redundancy of Reed-Muller codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 4867–4879, Nov. 2006.

[4] M. Schwartz and A. Vardy, “On the stopping distance and the stopping redundancy of codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.

Chapter 6

Beyond Stopping Redundancy

The stopping redundancy is a logical first step towards understanding the complexity-performance tradeoff in MPID over Tanner graphs with redundant check nodes. On the other hand, it clearly has its limitations. First of all, it only addresses one particular tradeoff point, where stopping distance is maximized with the smallest number of check nodes. It will be interesting to investigate other regions of this tradeoff, especially considering that the number of check nodes required to maximize stopping distance can be very large (growing exponentially with block length) for some codes, for example, as shown in [1]. Secondly, the definition of stopping redundancy is based on erasure channels. How relevant it is to various non-erasure channels remains to be seen.

In this chapter, we address some of the limitations mentioned above.

6.1 Guess-g Stopping Redundancy

6.1.1 Introduction

The definition of stopping redundancy aims at maximizing the stopping distance. However, the stopping redundancy may be prohibitively large for certain codes. Even when the stopping redundancy is at a manageable level, the number of checks could have been significantly reduced if we allowed the existence of just a few small stopping sets. The source of the above issue partly lies in the "principle of diminishing return," which in this case is to say that the *incremental* number of stopping sets that get eliminated by each *additional* check gets progressively smaller as the total number of parity checks increases.

A straightforward way to address the above concerns is to have a more moderate performance goal, i.e. to achieve stopping distance $t \leq d$ rather than d. The sequence of numbers

$$\rho^{(t)}(\mathcal{C}) \overset{\text{def}}{=} \min\bigl\{\text{number of rows of } H : \mathcal{C} = \text{Null}(H), s(H) = t\bigr\}, \quad t = 2, \ldots, d$$

is known as the *stopping redundancy hierarchy*, and has been studied in [2] [3]. Clearly,

$$\rho^{(2)}(\mathcal{C}) \leq \rho^{(3)}(\mathcal{C}) \leq \cdots \leq \rho^{(d)}(\mathcal{C}) = \rho(\mathcal{C}).$$

A different route is to keep the *same* performance goal, i.e. to be able to correct all sets of $d-1$ or fewer erasures, but allow a more graceful tradeoff with complexity by modifying the iterative decoding algorithm. Recall that iterative decoding on an erasure channel is equivalent to repeatedly finding and solving parity check equations with single unknowns. One way to generalize this decoding procedure is to let the decoder solve for up to p unknowns, rather than single unknowns, in each iteration. This idea was proposed in [4], where the concepts of stopping set and stopping redundancy were correspondingly extended to *stopping sets of order* p, and *stopping redundancy of order* p, respectively. Let $\rho^{\{p\}}(\mathcal{C})$ denote the stopping redundancy of order p. It is not hard to show [4] that

$$\rho(\mathcal{C}) = \rho^{\{1\}}(\mathcal{C}) = \rho^{\{2\}}(\mathcal{C}) \geq \rho^{\{3\}}(\mathcal{C}) \geq \cdots \geq \rho^{\{d-1\}}(\mathcal{C}) = \rho^{\{d\}}(\mathcal{C}) = n - k.$$

A slightly different approach, which we will explore, is motivated by another modification to the decoding procedure based on the idea of "guessing" [5] [6]. The decoding algorithm is sometimes known as the Maxwell decoder [7]. The idea is that when the iterative decoder gets stuck (i.e. when no check equations with a single unknown can be found), instead of declaring decoding failure, it treats one of the remaining unknown bits as known but with a variable value. At this point, iterative decoding may be able to resume. If not, the decoder assigns variable values to more erased bits until it is able to find a parity-check equation with a single unknown, or until there are no unknown bits left. Note that in the end, all decoded bits are generally linear combinations of the variable-valued bits and a constant. The variable-valued bits can then be solved for using the set of parity check equations that they participate in. It can be shown that this procedure achieves the same performance as maximum likelihood (ML) decoding if there is no limit on the number of "guesses" (i.e. variable-valued bits). In practice, the decoder usually puts a limit on the number of guesses, so that a failure is declared if all erased bits are not decoded after the maximum number of guesses have been made. Note that the term “guess” here is somewhat of a misnomer that has its roots in a more brute-force implementation of the same idea, where instead of assigning variable value to a bit, the bit is assumed to be a 0 or a 1, so that iterative decoding continues with more bits being guessed as necessary, and the process backtracks when any parity check is violated.

A similar modification to MPID has been proposed for use on other channels (e.g. AWGN) with promising results [8] [9].

We would like to understand how guessing helps to reduce the number of redundant check nodes in the Tanner graph. For all nonnegative integers g, let us define the *guess-g stopping redundancy* of a code $\mathcal{C}$ as the smallest number of check nodes in a Tanner graph for $\mathcal{C}$, such that all erasure patterns up to size $d-1$ can be decoded with the aforementioned modified iterative decoding algorithm using at most g guesses. If for some Tanner graph $\mathcal{T}$ (parity-check matrix H) this is the case, then we say that $\mathcal{C}$ is *guess-g decodable* on $\mathcal{T}$ (with H). Note that in the above definitions, it is assumed that the decoder can choose the best set of erased bits to guess. In practice, to choose a bit to be guessed, heuristic rules usually work well. For example, one can choose a bit with the largest degree in the subgraph induced by the erased bit nodes and their check neighbors [5].

Let $\rho_g(\mathcal{C})$ denote the guess-g stopping redundancy. Clearly, $\rho_g(\mathcal{C})$ is non-increasing in g, and is reduced to the normal definition of stopping redundancy when $g = 0$, i.e. we have

$$\rho(\mathcal{C}) = \rho_0(\mathcal{C}) \geq \rho_1(\mathcal{C}) \geq \cdots \geq \rho_{d-1}(\mathcal{C}) = n - k.$$

Let H be a parity-check matrix for $\mathcal{C}$. If an erasure pattern $S \subseteq [n] = \{1, 2, \ldots, n\}$ is correctable by the iterative decoder without guessing, then $H(S)$, the submatrix of H constrained to columns indexed by elements of S, must contain a row of weight one. The submatrix of $H(S)$ omitting this row and the column where the nonzero entry is must in turn contain a row of weight one, and so on. This implies that with row and column permutations $H(S)$ can be put into the following form

$$H(S) = \begin{bmatrix} \Delta \\ A \end{bmatrix}$$

where Δ is a lower-triangular matrix with nonzero diagonal. Similarly, an erasure pattern S can be decoded by the iterative decoder while guessing the value of no more than g bits if and only if $H(S)$ has the following form up to row and column permutations:

$$H(S) = \begin{bmatrix} A & \Delta \\ B & C \end{bmatrix} \tag{6.1}$$

where A has g or fewer columns, and Δ is a lower-triangular matrix with nonzero diagonal. Clearly, if the bits corresponding to columns of A are known, then iterative decoding will suffice to find the values of the rest.

Note that any matrix M can be viewed as having the form in (6.1), i.e.

$$M = \begin{bmatrix} A & \Delta \\ B & C \end{bmatrix}$$

where Δ is a lower-triangular matrix with nonzero diagonal entries (or as a special case, the empty matrix). Define the *triangulation deficiency* of M as the smallest number of columns in A over all column and row permutations of M, and denote it by $\mathrm{TD}(M)$. If M is an $a \times b$ matrix, then by definition, $0 \leq \mathrm{TD}(M) \leq b$.

Our definition of guess-g stopping redundancy is now equivalently stated as the smallest number of rows in H such that for all $S \subseteq [n]$ with $0 < |S| < d$, we have $\mathrm{TD}\big(H(S)\big) \leq g$.

6.1.2 Upper Bounds

Note that if all stopping sets have size at least $d - g$, then after g guesses any set of $d - 1$ or fewer erasures is stopping set free and thus can be decoded iteratively. In other words, relating to the definition of stopping redundancy hierarchy, we have

$$\rho_g(\mathcal{C}) \leq \rho^{(d-g)}(\mathcal{C}). \tag{6.2}$$

It turns out that most upper bounds on $\rho(\mathcal{C})$ can be adapted to $\rho^{(d-g)}(\mathcal{C})$ rather straightforwardly, thus leading to upper bounds on $\rho_g(\mathcal{C})$. For example, the works by Hollmann and Tolhuizen [10] [11] directly imply the following result.

Theorem 6.1 ([10] [11]) *Let $\mathcal{C}$ be an $[n, n-r, d]_2$ code. Then for all $0 \leq g < d - 2$,*

$$\rho_g(\mathcal{C}) \leq \sum_{i=0}^{d-g-2} \binom{r-1}{i}.$$

Proof: Omitted. ■

Corollary 6.2 *For all $g \geq d-3$,*

$$\rho_g(\mathcal{C}) = r.$$

Using probabilistic arguments similar to those used in Theorem 3.5 and Theorems 3.8–3.14, we can obtain a series of upper bounds on $\rho_g(\mathcal{C})$. For example, corresponding to Theorem 3.14, we have the following result.

Theorem 6.3 *Let $\mathcal{C}$ be an $[n,k,d]_2$ code, and let $r = n-k$. Then the guess-g stopping redundancy of $\mathcal{C}$ is bounded by*

$$\rho_g(\mathcal{C}) \;\leq\; \min\nolimits_{t \geq r}\big\{t + \lfloor \mathcal{G}_{n,d-g,k}(t) \rfloor\big\}, \tag{6.3}$$

where $\mathcal{G}_{n,d,k}(t)$ is as defined in (3.28). *Moreover, if $(r-1)(d-g-1) \leq 2^{d-g-1}$ then*

$$\rho(\mathcal{C}) \leq \min_{t \geq r}\Big\{t + \min\big\{i \in \mathbb{N} : Q_i^{(d-g)}\big(\lfloor \mathcal{G}_{n,d-g,k}(t) \rfloor\big) = 0\big\}\Big\} \tag{6.4}$$

where

$$Q_i^{(d-g)}(l) \;\stackrel{\text{def}}{=}\; P_i^{(d-g)}\bigg(P_{i-1}^{(d-g)}\Big(\cdots\, P_2^{(d-g)}\big(P_1^{(d-g)}(l)\big)\Big)\bigg),$$

$$P_j^{(d-g)}(l) \;\stackrel{\text{def}}{=}\; \left\lfloor l\left(1 - \frac{(d-g-1)2^{r-d+g+1}}{2^r - (t+j)}\right)\right\rfloor, \qquad \textit{for all } l \in \mathbb{N}.$$

Proof: Similar to that of Theorem 3.14. ■

More upper bounds on $\rho_g(\mathcal{C})$ can be obtained similarly by utilizing (6.2). However, note that in using (6.2) we are over-constraining the problem by requiring that no stopping sets of size less than $d-g$ exist, while by definition of the guess-g stopping redundancy, smaller stopping sets can exist as long as they are "weak" in the sense that they are "completely broken" if g bits within the set are known. In the discussion that follows, we try to translate the definition of $\rho_g(\mathcal{C})$ in other ways in hope of finding stronger upper bounds.

We first modify Theorem 6.3 using one additional "twist."

Theorem 6.4 *Let $\mathcal{C}$ be an $[n,k,d]_2$ code, and let $r = n-k$. Then the guess-g stopping redundancy of $\mathcal{C}$ is bounded by*

$$\rho_g(\mathcal{C}) \;\leq\; \min\nolimits_{t \geq r}\big\{t + \lfloor \mathcal{H}_{n,d,g,k}(t) \rfloor\big\}, \tag{6.5}$$

where

$$\mathcal{H}_{n,d,g,k}(t) \overset{\text{def}}{=} \mathcal{G}_{n,d-g-1,k}(t) + \binom{n}{d-1} \prod_{i=1}^{t} \left(1 - \frac{2^{r-d+1} \sum_{j=1}^{g+1} \binom{d-1}{j}}{2^r - i}\right)^+ \tag{6.6}$$

where $\mathcal{G}_{n,d,k}(t)$ *is as defined in* (3.28)*, and* $(x)^+ \overset{\text{def}}{=} \max\{x, 0\}$.

Proof: To be able to iteratively decode any set of $d-1$ or fewer erasures without making more than g guesses, the following two requirements suffice. First, we require that no stopping sets of size $d-g-2$ or smaller exist. This ensures that any set of $d-2$ or fewer erasures can be iteratively decoded with no more than g guesses. Next, for all $S \in [n]^{d-1}$, we require that $H(S)$ must contain a row whose weight is at least one and at most $g+1$. Thus, if all bits in S are erased, we can select a row from $H(S)$ with weight $j+1$ for some $0 \leq j \leq g$, guess j of the bits, and decode one additional bit using just that one parity check. The remaining $d-j-2$ erasures can then be iteratively decoded using no more than $g-j$ guesses.

Let H_t be a $t \times n$ matrix whose rows are drawn from $\mathcal{C}^\perp \setminus \{\mathbf{0}\}$ successively at random, without replacement. That is, if the row vectors of H_t are denoted by $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_t$, then for all j, $\boldsymbol{h}_j$ is chosen uniformly at random from $\mathcal{C}^\perp \setminus \{\mathbf{0}, \boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_{j-1}\}$. Given H_t, let X_t denote the number of non-empty stopping sets with size less than or equal to $d-g-2$, Y_t denote the number of $(d-1)$-sets S such that all rows of $H_t(S)$ have weights less than one or larger than $g+1$, and $Z_t = r - \operatorname{rank}(H_t)$. For all $S \in [n]^{d-1}$, let $\mathcal{A}_t(S)$ denote the event that all rows of $H_t(S)$ have weights less than one or greater than $g+1$. We have

$$\Pr\{\mathcal{A}_t(S)\} = \prod_{i=1}^{t} \left(1 - \frac{2^{r-d+1} \sum_{j=1}^{g+1} \binom{d-1}{j}}{2^r - i}\right)^+.$$

Hence, for all $t \geq r$,

$$\mathsf{E}[X_t + Y_t + Z_t] \leq \mathcal{G}_{n,d-g-1,k}(t) + \binom{n}{d-1} \prod_{i=1}^{t} \left(1 - \frac{2^{r-d+1} \sum_{j=1}^{g+1} \binom{d-1}{j}}{2^r - i}\right)^+.$$

The rest of the proof is similar to that of Theorem 3.14. ■

Note that Theorem 6.4 can be slightly improved following similar ideas that underlie Theorem 3.11.

We next modify the probabilistic argument while following the definition of $\rho_g(\mathcal{C})$ more closely, rather than trying to eliminate stopping sets of any particular size.

Theorem 6.5 *Let $\mathcal{C}$ be an $[n,k,d]_2$ code, and let $r = n - k$. Then the guess-g stopping redundancy of $\mathcal{C}$ is bounded by*

$$\rho_g(\mathcal{C}) \leq \min_{t \geq r}\bigl\{t + \lfloor \mathcal{I}_{n,d,g,k}(t) \rfloor\bigr\}, \tag{6.7}$$

where

$$\mathcal{I}_{n,d,g,k}(t) \overset{\text{def}}{=} \mathcal{D}_r(t) + \sum_{i=g+3}^{d-1} \binom{n}{i}\Bigl(1 - \frac{i}{2^i}\Bigr)^t \mathcal{J}_{t,g}(i), \tag{6.8}$$

$$\mathcal{J}_{t,g}(i) \overset{\text{def}}{=} \begin{cases} \lambda_i - g + 1 & \text{if } g < 2 \\ \min\bigl\{\lambda_i,\, a_i\lambda_i + \delta_i,\, b_i(1+\lambda_i) + \delta_i\bigr\} & \text{if } g \geq 2 \end{cases} \tag{6.9}$$

$$\lambda_i \overset{\text{def}}{=} \sum_{j=0}^{i-2}\Biggl(1 - \frac{2^{i-j-1} - i + j + 1}{2^i - i(i-1)/2}\Biggr)^t \tag{6.10}$$

$$a_i \overset{\text{def}}{=} \sum_{j=g}^{i-3} \frac{1}{j} \tag{6.11}$$

$$b_i \overset{\text{def}}{=} \sum_{j=g+1}^{i-2} \frac{1}{j} \tag{6.12}$$

$$\delta_i \overset{\text{def}}{=} \frac{(2^i - i(i-1)/2 - 1)(2^t - 1) + 2}{\bigl(2^i - i(i-1)/2\bigr)^t} \tag{6.13}$$

and $\mathcal{D}_r(t)$ is as defined in (3.27).

Proof: If $g \leq d - 3$, then $\rho_g(\mathcal{C}) = r$, but clearly $\min_{t \geq r}\{t + \lfloor \mathcal{I}_{n,d,g,k}(t) \rfloor\} \geq r$.

Now suppose $0 \leq g < d-3$. Let H_t be a $t \times n$ matrix whose rows are drawn from $\mathbb{F}_2^i$ independently and uniformly at random. Corresponding to H_t, let $\mathcal{S}_t$ denote the set of stopping sets whose sizes are less than d but greater than $g + 2$. Let

$$X_t = \sum_{S \in \mathcal{S}_t} \Bigl(\mathrm{TD}\bigl(H_t(S)\bigr) - g\Bigr)^+$$

and $Y_t = r - \mathrm{rank}(H_t)$. Then by appending no more than $X_t + Y_t$ rows to H_t we can construct another matrix with which $\mathcal{C}$ is guess-g decodable. Indeed, let us first append Y_t rows to H_t and call the resulting matrix H'. These Y_t additional rows are chosen such that $\mathrm{rank}(H') = r$. Let E denote a set of erasures with $|E| < d$. If E does not contain a stopping set, then all erasures can be iteratively decoded without making any guesses. Otherwise, let $S \subseteq E$ denote the maximal stopping set contained in E. Note that all erasures in $E \setminus S$ can be decoded without making any guesses. Now, if $|S| \leq g+2$ and/or $H_t(S) \leq g$, then with H', all erasures in S can be iteratively decoded using no more than g guesses. (Note that H' does not contain all-zero or identical columns.) Otherwise we have $g+2 < |S| < d$ and $H_t(S) = j > g$. In this case, we reduce the number of required guesses from j to g by adding at most $j-g$ appropriately chosen parity checks to H', which is possible since $\mathcal{C}^\perp$ is an orthogonal array of strength $d-1$ (cf. [12, p. 139]). We do this for each and every stopping set $S \in \mathcal{S}_t$, appending rows to H' as needed, and let H denote the matrix obtained. By construction, H has no more than $t + X_t + Y_t$ rows, and as argued above, $\mathcal{C}$ is guess-g decodable with H. Therefore, any realization of $t + X_t + Y_t$ is an upper bound on $\rho_g(\mathcal{C})$, which also implies

$$\rho_g(\mathcal{C}) \leq \big\lfloor \mathsf{E}[t + X_t + Y_t] \big\rfloor = t + \big\lfloor \mathsf{E}[X_t] + \mathsf{E}[Y_t] \big\rfloor .$$

From Lemma 3.13, we already know that $\mathsf{E}[Y_t] \leq \mathcal{D}_r(t)$ for all $t \geq 0$. It remains to find an expression for or an upper bound on $\mathsf{E}[X_t]$. We have

$$\begin{aligned}
\mathsf{E}[X_t] &= \mathsf{E}\Big[\sum_{S \in \mathcal{S}_t} \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \Big] \\
&= \sum_{\substack{S \subseteq [n] \\ g+2 < |S| < d}} \mathsf{E}\Big[1_{S \in \mathcal{S}_t} \cdot \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \Big] \\
&= \sum_{i=g+3}^{d-1} \sum_{S \in [n]^i} \mathsf{E}\Big[\big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,\big|\, S \in \mathcal{S}_t \Big] \Pr\{S \in \mathcal{S}_t\}
\end{aligned}$$

For all $S \in [n]^i$,

$$\Pr\{S \in \mathcal{S}_t\} = \Big(1 - \frac{i}{2^i}\Big)^t,$$

and $H_t(S)$ given $\{S \in \mathcal{S}_t\}$ has the same conditional distribution. Hence,

$$\mathsf{E}[X_t] = \sum_{i=g+3}^{d-1} \binom{n}{i} \Big(1 - \frac{i}{2^i}\Big)^t \mathsf{E}\Big[\big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,\big|\, S \in \mathcal{S}_t \Big] \tag{6.14}$$

Given $\{S \in \mathcal{S}_t\}$, $H_t(S)$ has a well-defined distribution. However, finding the distribution of $\mathrm{TD}\big(H_t(S)\big)$ appears to be difficult. To simplify the problem, we first introduce an upper bound on the function $\mathrm{TD}(\cdot)$. We say a binary vector *ends with a run of j zeros* if its last $j+1$ entries are

$$1\underbrace{0\ldots0}_{j}.$$

For any $a \times b$ binary matrix M, let $\mathcal{X}_j(M)$, $j = 0, 1, \ldots, b-1$ denote the event that M does not contain a row that ends with a run of j zeros. Define

$$\mathrm{TD}'(M) \overset{\mathrm{def}}{=} \sum_{j=0}^{b-1} 1_{\mathcal{X}_j(M)}.$$

Then for all M,

$$\mathrm{TD}(M) \le \mathrm{TD}'(M),$$

since $\mathrm{TD}'(M)$ can be interpreted as the number of columns in A when M is put into the form in (6.1) following a particular procedure (rather than the minimum number of columns in A over all row and column permutations, as in the case of $\mathrm{TD}(M)$).

For all $S \in [n]^i$, $g + 2 < i < d$, conditioned on $\{S \in \mathcal{S}_t\}$, $H_t(S)$ is a $t \times i$ matrix whose rows are drawn from $\{\boldsymbol{x} \in \mathbb{F}_2^i : \mathrm{wt}(\boldsymbol{x}) \ne 2\}$ independently and uniformly at random. Therefore, it is not hard to show that

$$\begin{aligned}
\mathsf{E}\big[\, \mathrm{TD}'\big(H_t(S)\big) \,|\, S \in \mathcal{S}_t \,\big] &= \sum_{j=0}^{i-1} \Pr\big\{\, \mathcal{X}_j\big(H_t(S)\big) \,|\, S \in \mathcal{S}_t \,\big\} \\
&= 1 + \sum_{j=0}^{i-2} \left(1 - \frac{2^{i-j-1} - i + j + 1}{2^i - i(i-1)/2} \right)^t \\
&= 1 + \lambda_i
\end{aligned}$$

In general, we have

$$\begin{aligned}
&\mathsf{E}\Big[\, \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,|\, S \in \mathcal{S}_t \Big] \\
&= \sum_{j=g+1}^{i} \Pr\big\{\, \mathrm{TD}\big(H_t(S)\big) \ge j \,|\, S \in \mathcal{S}_t \,\big\} \\
&\le \sum_{j=g+1}^{i-2} \Pr\big\{\, \mathrm{TD}'\big(H_t(S)\big) \ge j \,|\, S \in \mathcal{S}_t \,\big\} + \delta_i && (6.15) \\
&\le \sum_{j=g+1}^{i-2} \frac{1}{j}\, \mathsf{E}\big[\, \mathrm{TD}'\big(H_t(S)\big) \,|\, S \in \mathcal{S}_t \,\big] + \delta_i && (6.16) \\
&= b_i(1 + \lambda_i) + \delta_i
\end{aligned}$$

where (it is not hard to show that) δ_i, as defined in (6.13), is the expression for

$$\Pr\big\{\, \mathrm{TD}\big(H_t(S)\big) \ge i \,|\, S \in \mathcal{S}_t \,\big\} + \Pr\big\{\, \mathrm{TD}\big(H_t(S)\big) \ge i - 1 \,|\, S \in \mathcal{S}_t \,\big\}.$$

Note that (6.16) follows from Markov's inequality.

Since $\mathrm{TD}'\big(H_t(S)\big) \geq 1$ when S is a stopping set, (6.15) can alternatively be derived as

$$\begin{aligned}
&\mathsf{E}\Big[\, \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,\big|\, S \in \mathcal{S}_t \Big] \\
&\leq \sum_{j=g+1}^{i-2} \Pr\big\{\, \mathrm{TD}'\big(H_t(S)\big) - 1 \geq j - 1 \,|\, S \in \mathcal{S}_t \,\big\} + \delta_i \\
&\leq \sum_{j=g+1}^{i-2} \frac{1}{j-1}\, \mathsf{E}\big[\, \mathrm{TD}'\big(H_t(S)\big) - 1 \,|\, S \in \mathcal{S}_t \big] + \delta_i \\
&= a_i \lambda_i + \delta_i
\end{aligned}$$

In fact, based on the same observation, for $g = 0, 1$ we also have

$$\begin{aligned}
\mathsf{E}\Big[\, \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,\big|\, S \in \mathcal{S}_t \Big] &\leq \mathsf{E}\big[\, \mathrm{TD}'\big(H_t(S)\big) - g \,\big|\, S \in \mathcal{S}_t \,\big] \\
&= 1 + \lambda_i - g
\end{aligned}$$

In summary, for all $S \in [n]^i$ where $g + 2 < i < d$, $\mathsf{E}\big[\, \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,\big|\, S \in \mathcal{S}_t \,\big]$ is bounded from above by $\mathcal{J}_{t,g}(i)$, as defined in (6.9). Note that the inclusion of λ_i in the minimization in (6.9) comes from the observation that $\mathsf{E}\big[\, \big(\mathrm{TD}\big(H_t(S)\big) - g\big)^+ \,\big|\, S \in \mathcal{S}_t \,\big]$ should be non-increasing in g. Inequality (6.9) can now be plugged in (6.14) to find an upper bound on $\mathsf{E}[X_t]$, which completes the proof. ∎

Let us apply the bounds we have obtained so far to an example. Let $\mathcal{G}_{24}$ denote the $[24, 12, 8]_2$ Golay code. Table 6.1 lists various upper bounds on $\rho_g(\mathcal{G}_{24})$ for different values of g. We see that the probabilistic bounds in Theorems 6.3, 6.4 and 6.5 all fare very well against the constructive bound of Theorem 6.1. Among them, Theorem 6.4 gives the tightest bounds for $\mathcal{G}_{24}$. Also, all but the bound in Theorem 6.5 predict a drastic reduction in the number of required parity checks as g increases from 0 upwards. Even by allowing just a single guess, the bounds can be very dramatically lower. This observation is partly corroborated by the "empirical" bounds, which are found via greedy search on a computer (using ideas of the proof of Theorem 6.4). The fact the bound in Theorem 6.5 decreases slowly with g is due to the use of Markov's inequality (e.g. in (6.16)), rather than weakness of the probabilistic model. The true

Table 6.1 Upper bounds on $\rho_g(\mathcal{G}_{24})$

	$g=0$	$g=1$	$g=2$	$g=3$	$g=4$	$g\geq 5$
Theorem 6.1	1486	1024	562	232	67	12
Theorem 6.3	182	105	57	31	16	12
Theorem 6.4	189	58	31	16	12	12
Theorem 6.5	196	171	171	163	150	12
Empirical	34	17	13	12	12	12

tail probability $\Pr\{\,\mathrm{TD}'\big(H_t(S)\big) \geq j \,|\, S \in \mathcal{S}_t\,\}$ decreases much faster than $1/j$. Finally, observe that the bounds given by Theorem 6.4 (almost) coincide with those by Theorem 6.3 using one more guess. This is because for this example, the second term in (6.6) becomes insignificant relative to the first term when t is fairly large.

6.2 ML Redundancy

6.2.1 Introduction

The focus on stopping distance is justified only for "high SNR" scenarios, or as in the case of an erasure channel, for when the erasure probability is small (approaches zero). For the more general case, maximum likelihood (ML) decoding may be a more appropriate benchmark. Whereas in general ML decoding performance may never be matched by the iterative decoder, no matter which Tanner graph is used, on an erasure channel things are particularly well-defined. As we shall see, if we allow enough check nodes in the Tanner graph, on an erasure channel iterative decoding can always achieve the same word error rate as ML decoding. It is therefore a theoretically interesting question to ask how many check nodes are required for this to be possible, thus leading to the definition of ML redundancy, which will be the subject of this section.

Let H be a parity-check matrix for some code $\mathcal{C}$. The *iterative H-decoder* is the iterative erasure decoder over a Tanner graph defined by H. As noted before, an iterative H-decoder works effectively by finding, at each iteration, a parity-check equation that involves a single erasure, solving for the erasure, and repeating the process until either all erasures have been decoded, or no such equations can be found (in which case a decoding failure results). We are interested in when an iterative H-decoder achieves the same failure rate as the word ML decoder, and the minimum number of rows in H required for doing so.

An *erasure pattern*, i.e. a set of code coordinates that are erased, can be corrected by the ML decoder if and only if it does not contain the support of a codeword, in which case we say that the erasure pattern is *correctable*. In comparison, for the iterative H-decoder, an erasure pattern can be corrected if and only if it does not contain a stopping set. Hence, the iterative H-decoder is ML if and only if none of the correctable erasure patterns contains a stopping set. Since any subset of a correctable erasure pattern is still correctable, this is equivalent to requiring that the set of correctable erasure patterns and the set of stopping sets be disjoint. As in Chapter 3, we say that a matrix M *covers* a set S if $M(S)$ contains a row of weight one. (Note that M can be a vector as a special case.) Thus, the iterative H-decoder achieves ML performance if and only if all correctable erasure patterns are covered by H.

We define the *ML redundancy* of a code $\mathcal{C}$, denoted by $\gamma(\mathcal{C})$, as the smallest number of rows in a parity-check matrix H such that the iterative H-decoder for $\mathcal{C}$ achieves ML performance. In this section, we will develop bounds on $\gamma(\mathcal{C})$ both in general and for certain classes of codes.

Let H^* denote a matrix consisting of all codewords of $C^\perp$ as rows. The following lemma is a known result [5], which shows that $\gamma(\mathcal{C})$ is well-defined, i.e. it is always possible to achieve ML performance with an iterative decoder on the erasure channel.

Lemma 6.6 *The iterative H^*-decoder is ML.*

Proof: If $S \subseteq [n]$ is correctable, then for any parity-check matrix H, the column vectors of $H(S)$ are linearly independent. Hence, the rows of $H(S)$ contain a basis of $\mathbb{F}_q^{|S|}$. Therefore, all possible q-ary $|S|$-tuples appear as rows in $H^*(S)$ the same number of times. In particular, $H^*(S)$ must contain a row of weight one. ■

The next lemma shows that if a matrix H has all its rows taken from $\mathcal{C}^\perp$ and covers all correctable erasure patterns for $\mathcal{C}$, then it must indeed be a parity-check matrix for $\mathcal{C}$. Hence, to verify that an iterative H-decoder achieves ML performance, it suffices to verify that H covers all correctable erasure patterns.

Lemma 6.7 *If $\mathcal{C} \subseteq \mathrm{Null}(H)$ and H covers all correctable erasure patterns for $\mathcal{C}$, then $\mathcal{C} = \mathrm{Null}(H)$, where $\mathrm{Null}(H)$ is the null space of H.*

Proof: The redundancy of $\mathcal{C}$ being r implies that correctable erasure patterns of size r exist. Let S be such an erasure pattern. Since S is covered by H, it is covered by some row of H, say $\boldsymbol{h}_1$. Let s_1 denote the coordinate that $\mathrm{supp}(\boldsymbol{h}_1)$ and S have in common. Then $S \setminus \{s_1\}$ is a correctable erasure pattern of size $r - 1$. And so it must be covered by some other row of H, say $\boldsymbol{h}_2$. Repeating the above argument shows that up to row and column permutations, H contains an $r \times r$ lower triangular matrix with nonzero diagonal. Hence, $\mathrm{rank}(H) = r$. Since $\mathcal{C} \subseteq \mathrm{Null}(H)$, it follows that $\mathcal{C} = \mathrm{Null}(H)$. ■

The following result is obtained as an easy extension of Theorem 3.1.

Theorem 6.8 *Let $\mathcal{C}$ be a binary code with redundancy r. Then*

$$\gamma(\mathcal{C}) \leq 2^{r-1}. \tag{6.17}$$

Proof: Note that any set of $r + 1$ or more erasures is not correctable. Thus, in the proof of Theorem 3.1, if we replace d with $r+1$ in the construction for H', then H' covers all correctable erasure patterns for $\mathcal{C}$. Now, it suffices to note that the number of rows in H' is

$$\sum_{i=1}^{\lfloor (r+1)/2 \rfloor} \binom{r}{2i-1} = 2^{r-1}.$$

■

Hollmann and Tolhuizen [11] considered general constructions of parity-check collections for binary codes that correct all correctable erasure patterns up to a given size. Specifically, the study in [11] focused on finding an r-column matrix A with the least number of rows, having the property that given any $r \times n$ matrix H with rank r, the iterative AH-decoder corrects all correctable erasure patterns of size m or smaller, for the code defined by the null space of H. Such a matrix (A), or rather, collection of r-tuples (row vectors of A), was defined as a *generic* (r, m)-*correcting set*, whose smallest size is denoted by $F(r, m)$. It was found that $F(r, r) = 2^{r-1}$, which implied Theorem 6.8. The authors of [11] also showed that equality is achieved in (6.17) for binary Hamming codes.

Theorem 6.9 ([11]) *Let $\mathcal{H}_m$ denote the $[2^m-1, 2^m-m-1, 3]_2$ Hamming code. Then*

$$\gamma(\mathcal{H}_m) = 2^{m-1}.$$

Other work related to the subject includes that of [13].

6.2.2 Upper Bounds

Theorem 6.8 can be extended to q-ary codes rather straightforwardly using similar ideas as in [11].

Theorem 6.10 *Let $\mathcal{C}$ be a q-ary code with redundancy $r > 0$. Then*

$$\gamma(\mathcal{C}) \le q^{r-1}.$$

Proof: Let H be an $r \times n$ parity-check matrix for $\mathcal{C}$, where n is the length of $\mathcal{C}$. Let the row vectors of H be denoted by $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$. Define $V \stackrel{\text{def}}{=} \{\boldsymbol{h}_1 + \boldsymbol{v} : \boldsymbol{v} \in \operatorname{span}(\{\boldsymbol{h}_2, \ldots, \boldsymbol{h}_r\})\}$. Clearly, $|V| = q^{r-1}$. We show that every correctable erasure pattern is covered by some vector in V.

Let $S \subseteq [n]$ be correctable. Then $\operatorname{rank}\big(H(S)\big) = |S|$. Let $B \subseteq \{\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r\}$ be such that $B(S) \stackrel{\text{def}}{=} \{\boldsymbol{b}(S) : \boldsymbol{b} \in B\}$ form a basis of $\mathbb{F}_q^{|S|}$. If $\boldsymbol{h}_1 \in B$, note that $\operatorname{span}\big(B(S) \backslash \boldsymbol{h}_1(S)\big)$ has rank $|S|-1$, hence does not contain all weight one $|S|$-tuples. Therefore, there exists $\boldsymbol{e} \in \mathbb{F}_q^{|S|}$, such that $\operatorname{wt}(\boldsymbol{e}) = 1$ and $\boldsymbol{e} = \alpha \boldsymbol{h}_1(S) + \boldsymbol{x}(S)$, where $\alpha \neq 0$ and $\boldsymbol{x} \in \operatorname{span}(B \setminus \boldsymbol{h}_1) \subseteq \operatorname{span}(\{\boldsymbol{h}_2, \ldots, \boldsymbol{h}_r\})$. Since $\operatorname{wt}(\alpha^{-1}\boldsymbol{e}) = 1$, we see S is covered by $(\boldsymbol{h}_1 + \alpha^{-1}\boldsymbol{x}) \in V$. Finally, if $\boldsymbol{h}_1 \notin B$, simply note that $V(S) = \operatorname{span}\big(B(S)\big) = \mathbb{F}_q^{|S|}$. ■

Many techniques for obtaining bounds on stopping redundancy can be adapted to ML redundancy as well. One such example is given in the following theorem, which is based on an idea similar to that of Theorem 4.3.

Theorem 6.11 *Let $\mathcal{C}$ be an $[n, k]_q$ code. Let $r = n - k$. Then for all $0 \le p \le 1$,*

$$\gamma(\mathcal{C}) \le pq^r + \sum_{i=1}^{r} \binom{n}{i} (1-p)^{i(q-1)q^{r-i}}.$$

Proof: For some prescribed real value p, $0 \leq p \leq 1$, select each codeword in $\mathcal{C}^{\perp}$ with probability p, and let matrix H consist of all selected codewords as rows (with arbitrary ordering). At this point, the expected number of rows in H is

$$p \cdot |\mathcal{C}^{\perp}| = pq^r.$$

Not all correctable erasure patterns may be covered by H. Particularly, for any given $X \in [n]^i$, $i = 1, \ldots, r$, that is correctable, the probability that X is not covered by H is

$$(1-p)^{i(q-1)q^{r-i}}.$$

This is because X being correctable implies that $\mathcal{C}^{\perp}(X)$ (as a multiset) contains all q-ary i-tuples the same number of times. To cover all correctable erasure patterns, as a second step, for each $X \in [n]^i$, $i = 1, \ldots, r$, that is correctable but not yet covered by H, append to H a codeword from $\mathcal{C}^{\perp}$ that covers X, until no such X can be found. At this point, the expected number of rows in H is at most

$$\begin{aligned} & pq^r + \sum_{i=1}^{r} \sum_{\substack{X \in [n]^i \\ X \text{correctable}}} (1-p)^{i(q-1)q^{r-i}} \\ & \leq pq^r + \sum_{i=1}^{r} \binom{n}{i} (1-p)^{i(q-1)q^{r-i}}. \end{aligned}$$

Therefore, there exists at least one realization of H with at most the above number of rows. ■

We now consider a different approach based on minimal codewords. A nonzero codeword is *minimal* if its support does not contain the support of another codeword of smaller weight.[1] The set of minimal codewords of code $\mathcal{C}$ is denoted by $\mathcal{M}(\mathcal{C})$. Minimal codewords were introduced by Hwang [14], and have found applications from decoding algorithms [14] [15] [16] to secret sharing [17]. Basic properties of minimal codewords were studied by Massey [17], and by Ashikhmin and Barg [16], who also characterized minimal codewords for certain well-known classes of codes.

Some properties of minimal codewords [17] [16] that we will use are summarized in the following lemma.

[1] Some authors require in addition that the first nonzero coordinate of minimal codewords be one.

Lemma 6.12 *Let $\mathcal{C}$ be an $[n,k]_q$ code. Then*

1. *For all $\boldsymbol{v} \in \mathcal{C} \setminus \{\mathbf{0}\}$, there exist $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_m \in \mathcal{M}(\mathcal{C})$, such that $\boldsymbol{v} = \sum_{i=1}^{m} \boldsymbol{c}_i$ and $\operatorname{supp}(\boldsymbol{c}_i) \subseteq \operatorname{supp}(\boldsymbol{v})$, $i = 1, \ldots, m$.*
2. *For all $\boldsymbol{c} \in \mathcal{M}(\mathcal{C})$, we have $\operatorname{wt}(\boldsymbol{c}) \leq n-k+1$.*

We make a key observation that to achieve ML performance, it suffices to select parity checks from $\mathcal{M}(\mathcal{C}^{\perp})$, which leads to the following upper bound on ML redundancy.

Theorem 6.13 *If $\mathcal{C}$ is a q-ary code, then*

$$\gamma(\mathcal{C}) \leq \frac{1}{q-1} \big| \mathcal{M}(\mathcal{C}^{\perp}) \big|.$$

Proof: Let $H_{\mathcal{M}}$ denote the matrix consisting of all codewords of $\mathcal{M}(\mathcal{C}^{\perp})$ as rows. If $S \subseteq [n]$ is correctable, then by Lemma 6.6, it is covered by some $\boldsymbol{c} \in \mathcal{C}^{\perp}$. If $\boldsymbol{c} \in \mathcal{M}(\mathcal{C}^{\perp})$, then $H_{\mathcal{M}}$ covers S. If $\boldsymbol{c} \notin \mathcal{M}(\mathcal{C}^{\perp})$, then by Lemma 6.12, $\boldsymbol{c}$ can be written as a sum of codewords from $\mathcal{M}(\mathcal{C}^{\perp})$, whose supports are contained in $\operatorname{supp}(\boldsymbol{c})$. Thus, one of these minimal codewords must cover S. Finally, note that just one codeword is needed from $\mathcal{M}(\mathcal{C}^{\perp})$ for each support set. ■

It is often hard to find the number of minimal codewords in $\mathcal{C}^{\perp}$. Lemma 6.12 tells us that it suffices to consider codewords of weights $k+1$ or less, leading to the following corollary. Note that the same result was also shown in [13, Theorem 3], albeit through a very different argument.

Corollary 6.14 *If $\mathcal{C}$ is an $[n,k]_q$ code, then*

$$\gamma(\mathcal{C}) \leq \frac{1}{q-1} \sum_{i=1}^{k+1} B_i,$$

where B_i is the number of weight-i codewords in $\mathcal{C}^{\perp}$.

Let us consider a few examples.

Example 6.1 Let $\mathcal{H}_m$ denote the binary Hamming code with redundancy m, and $\mathcal{S}_m$ denote its dual code, the $[2^m-1, m, 2^{m-1}]_2$ Simplex code. Since all nonzero codewords of $\mathcal{S}_m$ have the same weight, all of them are minimal. Theorem 6.13 tells us that $\gamma(\mathcal{H}_m) \leq 2^m - 1$, which is trivially true and is about twice the true value of $\gamma(\mathcal{H}_m)$ as given in Theorem 6.9.

On the other hand, it is known [16] that the number of minimal codewords in $\mathcal{H}_m$ of weight w is

$$M_w = \begin{cases} \frac{1}{w!}\prod_{i=0}^{w-2}(2^m - 2^i) & \text{if } 3 \leq w \leq m+1, \\ 0 & \text{otherwise.} \end{cases}$$

So by Theorem 6.13, $\gamma(\mathcal{S}_m) \leq \left|\mathcal{M}(\mathcal{H}_m)\right| = O\left(2^{m^2}/m\right)$, which is much stronger than the upper bound of $2^{2^m - m - 2}$, as given by Theorem 6.8. □

Example 6.2 Let $\mathcal{G}_{24}$ denote the $[24, 12, 8]_2$ self-dual Golay code. It is known [16] that $\mathcal{M}(\mathcal{G}_{24}) = \{\boldsymbol{c} \in \mathcal{G}_{24} \setminus \{\boldsymbol{0}\} : \mathrm{wt}(\boldsymbol{c}) \leq 12\}$. By Theorem 6.13 and the well-known weight distribution of $\mathcal{G}_{24}$ [12], we have $\gamma(\mathcal{G}_{24}) \leq 3335$. Note that Theorem 6.8 and Theorem 6.11 both perform better in this case, yielding $\gamma(\mathcal{G}_{24}) \leq 2048$ and $\gamma(\mathcal{G}_{24}) \leq 2435$, respectively. A greedy search shows that in fact $\gamma(\mathcal{G}_{24}) \leq 370$. □

Example 6.3 For MDS codes, minimal codewords and minimum-weight codewords are the same, because all nonzero codewords with weight not exceeding $r + 1$ are minimum-weight. Also, ML redundancy and stopping redundancy become equivalent, since no correctable erasure pattern is of size larger than $r = d - 1$.

Let $\mathcal{C}$ be an $[n, k]_q$ MDS code. Then $\mathcal{C}^\perp$ is an $[n, n-k]_q$ MDS code. Theorem 6.13 gives us $\gamma(\mathcal{C}) \leq \binom{n}{k+1}$, which was obtained in [1] as an upper bound on $\rho(\mathcal{C})$. Much stronger upper- and lower-bounds on $\rho(\mathcal{C}) = \gamma(\mathcal{C})$ for MDS codes have been presented in Chapter 4 of this thesis. □

Example 6.4 Let $\mathcal{C}$ be a random code whose $(n - k) \times n$ parity-check matrix consists of independent and equiprobable entries drawn from $\mathbb{F}_q$. The following result was shown in [16].

Theorem 6.15 ([16]) *Let $k = Rn$, where $R \in (0, 1)$ is fixed. Then*

$$\lim_{n\to\infty} \frac{1}{n} \log_q \mathsf{E}\left[|\mathcal{M}(\mathcal{C})|\right] = \begin{cases} H_q(1-R) - (1-R) & \text{if } R > 1/q \\ R & \text{if } R \leq 1/q \end{cases}$$

where $H_q(\cdot)$ is the base-q entropy function.

Theorem 6.15 was proved by noting that the average number of minimal codewords of weight w is at least a constant fraction (> 0.288) of the average total number of weight-w codewords, for all $w \leq n-k+1$. This shows that Corollary 6.14 in general should give a bound that is not much larger than that given by Theorem 6.13.

To compare $\mathsf{E}\big[|\mathcal{M}(\mathcal{C})|\big]$ with the total number of codewords, note that $\mathsf{E}\big[|\mathcal{C}|\big] = q^k - q^{-r} + 1$. Therefore, if $R \leq 1/q$, almost all codewords in $\mathcal{C}$ are minimal; if $R > 1/q$, the number of minimal codewords as a fraction of $|\mathcal{C}|$ decreases to 0 exponentially fast as $n \to \infty$. Accordingly, Theorem 6.13 is likely to give us a non-trivial bound if $R < (q-1)/q$. This observation is corroborated by our earlier examples of Hamming, Simplex, and Golay codes. □

6.2.3 Codes with Designs

As before, let $\mathcal{H}_m$ and $\mathcal{S}_m$ denote the $[2^m-1, 2^m-m-1, 3]_2$ Hamming code and the $[2^m-1, m, 2^{m-1}]_2$ Simplex code, respectively. We first show that the ML redundancy of $\mathcal{S}_m$ is small — at most quadratic in the length of the code. This is contrasted with the very large number of available parity checks. It is also much lower than the $O\big(2^{m^2}/m\big)$ bound given by Theorem 6.13. Interestingly, the stopping redundancy of $\mathcal{S}_m$ is also very small. Indeed, it can be shown [18] that the stopping redundancy of $\mathcal{S}_m$ is equal to its redundancy, $2^m - m - 1$.

Theorem 6.16 *For all m,*

$$\gamma(\mathcal{S}_m) \leq \frac{n^2-n}{6},$$

where $n = 2^m - 1$ is the block length of $\mathcal{S}_m$.

Proof: It is well known [12] that the number of weight-3 codewords in $\mathcal{H}_m$ is $\frac{1}{3}\binom{n}{2} = (n^2 - n)/6$. We will show that any correctable erasure pattern for $\mathcal{S}_m$ is covered by a weight-3 codeword of $\mathcal{H}_m$. Let S be a correctable erasure pattern. Then there exists $\boldsymbol{c} \in \mathcal{H}_m$, such that $\mathrm{wt}\big(\boldsymbol{c}(S)\big) = 1$. If $\mathrm{wt}(\boldsymbol{c}) = 3$, then we are done. Otherwise, note that the supports of weight-3 codewords in $\mathcal{H}_m$ form a $(2,3,n)$ Steiner system [12]. Hence, we can find a weight-3 codeword, $\boldsymbol{x}_1 \in \mathcal{H}_m$, that matches $\boldsymbol{c}$ at its nonzero position within S and at least one other nonzero bit. If $\mathrm{wt}\big(\boldsymbol{x}_1(S)\big) = 1$, then we are done. Otherwise, let $\boldsymbol{c}_1 = \boldsymbol{c} - \boldsymbol{x}_1$, and note that $\mathrm{wt}\big(\boldsymbol{c}_1(S)\big) = 1$ and $\mathrm{wt}(\boldsymbol{c}_1) < \mathrm{wt}(\boldsymbol{c})$. Now we can repeat the above procedure and find weight-3 codewords

$\boldsymbol{x}_2, \boldsymbol{x}_3, \ldots \in \mathcal{H}_m$, until some $\boldsymbol{x}_i$ is found that covers S. This must happen since with each unsuccessful step the weight of the codeword that covers S (i.e. $\boldsymbol{c}_i = \boldsymbol{c}_{i-1} - \boldsymbol{x}_i$) is reduced by at least one. ■

The above bound can be slightly improved by noting that a subset of the weight-3 codewords of $\mathcal{H}_m$ suffices as parity checks, giving the following result.

Theorem 6.17 *For all m,*

$$\gamma(\mathcal{S}_m) \leq \frac{n^2 - 4n + 9}{6},$$

where $n = 2^m - 1$ is the block length of $\mathcal{S}_m$.

Proof: Let $\mathcal{A}_m$ denote the set of weight-3 codewords of $\mathcal{H}_m$. For $j = 1, \ldots, n$, define $\mathcal{A}_{m,j} \overset{\text{def}}{=} \{\boldsymbol{x} \in \mathcal{A}_m : j \in \operatorname{supp}(\boldsymbol{x})\}$. Let $\mathcal{A}'_{m,j} \overset{\text{def}}{=} \mathcal{A}_m \setminus \big(\mathcal{A}_{m,j} \setminus \{\boldsymbol{a}_{m,j}\}\big)$, where $\boldsymbol{a}_{m,j}$ is an arbitrary element in $\mathcal{A}_{m,j}$. We show that any correctable erasure pattern is covered by a vector in $\mathcal{A}'_{m,j}$ for all j.

Basically, we are saying that among all weight-3 codewords of $\mathcal{H}_m$ whose supports contain the j-th coordinate, we may discard all but one of them, and still cover all correct erasure patterns. Let $S \subseteq [n]$ be a correctable erasure pattern. Then there exists $\boldsymbol{c} \in \mathcal{H}_m$, such that $\operatorname{wt}\big(\boldsymbol{c}(S)\big) = 1$. Observe that if $j \notin \operatorname{supp}(\boldsymbol{c})$, then the recursive procedure in the previous proof can be followed through without using any vectors in $\mathcal{A}_{m,j}$. This is because at each step, we have multiple choices for $\boldsymbol{x}_i$ by varying the choice of the position outside of S, and only one of the resulting 3-sets possibly contains j.

We now show that either S is trivially covered, or we can find $\boldsymbol{c} \in \mathcal{H}_m$ such that $\operatorname{wt}\big(\boldsymbol{c}(S)\big) = 1$ and $j \notin \operatorname{supp}(\boldsymbol{c})$, so the above observation suffices to complete the proof. We need the following fact, which can be seen from the proof of Lemma 6.6:

$$\forall s \in S,\ \exists \boldsymbol{c} \in \mathcal{H}_m,\ \text{s.t. } \operatorname{supp}\big(\boldsymbol{c}(S)\big) = \{s\}. \tag{6.18}$$

Now, consider two cases. If $j \in S$, then by (6.18), unless $S = \{j\}$ (in which case S is covered by $\boldsymbol{a}_{m,j}$), we can always choose $\boldsymbol{c} \in \mathcal{H}_m$ such that $\operatorname{wt}\big(\boldsymbol{c}(S)\big) = 1$ and $j \notin \operatorname{supp}(\boldsymbol{c})$. If $j \notin S$, let's further assume that $\operatorname{wt}\big(\boldsymbol{a}_{m,j}(S)\big) \neq 1$ (or S is covered by $\boldsymbol{a}_{m,j}$). Thus, $\operatorname{wt}\big(\boldsymbol{a}_{m,j}(S)\big) = 0$ or 2. If $\operatorname{wt}\big(\boldsymbol{a}_{m,j}(S)\big) = 0$, choose $\boldsymbol{v} \in \mathcal{H}_m$ such that $\operatorname{wt}\big(\boldsymbol{v}(S)\big) = 1$. If $\operatorname{wt}\big(\boldsymbol{a}_{m,j}(S)\big) = 2$, by (6.18) we can choose $\boldsymbol{v} \in \mathcal{H}_m$ such that $\operatorname{wt}\big(\boldsymbol{v}(S)\big) = 1$ and $\operatorname{supp}\big(\boldsymbol{v}(S)\big) \subset \operatorname{supp}(\boldsymbol{a}_{m,j})$. Now note that in either case, $\boldsymbol{v}$ and $\boldsymbol{v} + \boldsymbol{a}_{m,j}$ both cover S, but only one of them has j in its support. The one whose support does not contain j can then be chosen as $\boldsymbol{c}$.

Finally, since $\{\text{supp}(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{A}_m\}$ is a $(2, 3, n)$ Steiner system, $|\mathcal{A}_{m,j}| = (n-1)/2$ for all j. Therefore, $|\mathcal{A}'_{m,j}| = |\mathcal{A}_m| - |\mathcal{A}_{m,j}| + 1 = n(n-1)/6 - (n-1)/2 + 1 = (n^2 - 4n + 9)/6.$ ■

The above bound appears to be reasonably good, but is not always sharp. For example, Theorem 6.17 shows that $\gamma(\mathcal{S}_3) \leq 5$ and $\gamma(\mathcal{S}_4) \leq 29$. While it can be verified using an exhaustive search that indeed $\gamma(\mathcal{S}_3) = 5$, a greedy search shows that $\gamma(\mathcal{S}_4) \leq 21$.

Next, consider Reed-Muller (RM) codes. Let $\mathcal{R}(l, m)$ denote the l-th order RM code with parameter m. The first order RM code, $\mathcal{R}(1, m)$, is closely related to the Simplex code $\mathcal{S}_m$. Whereas $\mathcal{S}_m$ is the dual code of the Hamming code $\mathcal{H}_m$, $\mathcal{R}(1, m)$ is the dual code of the corresponding extended Hamming code, which we denote by $\mathcal{H}_m^{\text{ext}}$. Using similar ideas from Theorem 6.16 and Theorem 6.17, we obtain the following bound on the ML redundancy of $\mathcal{R}(1, m)$.

Theorem 6.18 *For all m,*

$$\gamma\big(\mathcal{R}(1, m)\big) \leq \frac{1}{4}\binom{n}{3} - \frac{n}{2} + 2.$$

where $n = 2^m$ is the block length of $\mathcal{R}(1, m)$.

Proof (Sketch): Let $\mathcal{B}_m$ denote the set of weight-4 codewords of $\mathcal{H}_m^{\text{ext}}$. It is well known [12] that $\{\text{supp}(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{B}_m\}$ is a $(3, 4, n)$ Steiner system. For each $D \in [n]^2$, define $\mathcal{B}_{m,D} \overset{\text{def}}{=} \{\boldsymbol{x} \in \mathcal{B}_m : D \subset \text{supp}(\boldsymbol{x})\}$, and let $\mathcal{B}'_{m,D} \overset{\text{def}}{=} \mathcal{B}_m \setminus \big(\mathcal{B}_{m,D} \setminus \{\boldsymbol{b}_{m,D}\}\big)$, where $\boldsymbol{b}_{m,D}$ is an arbitrary element in $\mathcal{B}_{m,D}$.

Arguing similarly as in the proofs of Theorem 6.16 and Theorem 6.17, one can show that any correctable erasure pattern is covered by a vector in $\mathcal{B}'_{m,D}$ for all D. Finally, note that $|\mathcal{B}'_{m,D}| = |\mathcal{B}_m| - |\mathcal{B}_{m,D}| + 1 = \frac{1}{4}\binom{n}{3} - \frac{n-2}{2} + 1.$ ■

The essence of Theorems 6.16–6.18 can be generalized to any length-n linear code whose dual code contains an $(n, t+1, t)$ covering design (by which we mean precisely that the supports of weight-$(t+1)$ codewords in the dual code form an $(n, t+1, t)$ covering design) for some t.

Theorem 6.19 *Let $\mathcal{C}$ be a linear code of length n over $\mathbb{F}_q$. If for some $t \geq q$, $\mathcal{C}^{\perp}$ contains an $(n, t+1, t)$ covering design, then*

$$\gamma(\mathcal{C}) \leq \binom{n}{t} + \frac{1}{q-1} \sum_{i=1}^{t} B_i. \tag{6.19}$$

In particular, if $\mathcal{C}^{\perp}$ contains a $(t, t+1, n)$ Steiner system, then

$$\gamma(\mathcal{C}) \leq \frac{1}{t+1} \binom{n}{t} + \frac{1}{q-1} \sum_{i=1}^{t} B_i, \tag{6.20}$$

where B_i is the number of weight-i codewords in $\mathcal{C}^{\perp}$.

Proof: Choose a subset of weight-$(t+1)$ codewords from $\mathcal{C}^{\perp}$ as parity checks, such that their supports form an $(n, t+1, t)$ covering design. This subset can be chosen to contain less than $\binom{n}{t}$ vectors (or, precisely $\frac{1}{t+1}\binom{n}{t}$ vectors, if $\mathcal{C}^{\perp}$ contains a $(t, t+1, n)$ Steiner system). Choose also a subset of codewords from $\mathcal{C}^{\perp}$ whose weights are less than $(t+1)$ (if there are any), such that exactly one codeword is chosen for each support set. The total number of vectors we have chosen from $\mathcal{C}^{\perp}$ is no greater than $\binom{n}{t} + \frac{1}{q-1}\sum_{i=1}^{t} B_i$ (or, $\frac{1}{t+1}\binom{n}{t} + \frac{1}{q-1}\sum_{i=1}^{t} B_i$, if $\mathcal{C}^{\perp}$ contains a $(t, t+1, n)$ Steiner system). Let this set of vectors (as parity checks) be denoted by $\mathcal{A}$. We show that any correctable erasure pattern for $\mathcal{C}$ is covered by a vector in $\mathcal{A}$.

If S is correctable, then there exists $\boldsymbol{v} \in \mathcal{C}^{\perp}$ such that $\mathrm{wt}\big(\boldsymbol{v}(S)\big) = 1$. Let $\boldsymbol{c} = \alpha \boldsymbol{v}$ where $\alpha \in \mathbb{F}_q$, $\alpha \neq 0$ is a constant of our choice. If there exists α such that $\boldsymbol{c} \in \mathcal{A}$, then we are done. Otherwise, we have $|\operatorname{supp}(\boldsymbol{c})| \geq t+1$ and $|\operatorname{supp}(\boldsymbol{c}) \setminus S| \geq t$. Let X be any t-subset of $\big(\operatorname{supp}(\boldsymbol{c}) \setminus S\big)$. Since $\mathcal{C}^{\perp}$ contains an $(n, t+1, t)$ covering design, there exists $\boldsymbol{x}_1 \in \mathcal{A}$ such that $\mathrm{wt}(\boldsymbol{x}_1) = t+1$ and $\operatorname{supp}(\boldsymbol{x}_1) \supset X$. If $\mathrm{wt}(\boldsymbol{x}_1(S)) = 1$, then we are done. Otherwise, note that by choosing α we can ensure that $\boldsymbol{c}$ matches $\boldsymbol{x}_1$ at no less than $\lceil t/(q-1) \rceil$ positions. Let $\boldsymbol{c}_1 = \alpha_1(\boldsymbol{c} - \boldsymbol{x}_1)$, where $\alpha_1 \in \mathbb{F}_q$, $\alpha_1 \neq 0$ can be chosen freely. Note that $\mathrm{wt}(\boldsymbol{c}_1(S)) = 1$. And since $t \geq q$, we also have

$$\mathrm{wt}(\boldsymbol{c}_1) \leq \mathrm{wt}(\boldsymbol{c}) - \lceil t/(q-1) \rceil + 1 < \mathrm{wt}(\boldsymbol{c}).$$

Now the above procedure can be repeated to find weight-$(t+1)$ codewords $\boldsymbol{x}_2, \boldsymbol{x}_3, \ldots \in \mathcal{A}$, and stopped if some $\boldsymbol{x}_i$ is found that covers S. Since at each unsuccessful step the weight of the parity-check vector that covers S (i.e. $\boldsymbol{c}_i = \alpha_i(\boldsymbol{c}_{i-1} - \boldsymbol{x}_i)$) is reduced by at least one, the process must terminate in a finite number of steps, at the end of which we will either have found some $\boldsymbol{x}_i \in \mathcal{A}$ that covers S, or we will be left with some $\mathbf{0} \neq \boldsymbol{c}_i = \alpha_i(\boldsymbol{c}_{i-1} - \boldsymbol{x}_i)$ that covers S and has weight no greater than t. In the latter case, note that by construction, there exists α_i such that $\boldsymbol{c}_i \in \mathcal{A}$. ■

The recursive procedure used in the above proof is slightly different from the one used in the proof of Theorem 6.16, but the two are similar in principle.

If the minimum distance of $\mathcal{C}^\perp$ is $t+1$, then the second term in (6.19) is zero, as in the case of Simplex and first order Reed-Muller codes.

As examples, let $\mathcal{G}_{11}$ and $\mathcal{G}_{12}$ denote the ternary and extended ternary Golay codes, respectively. It is well known [12] that the supports of weight-6 codewords in $\mathcal{G}_{12}$ and those of weight-5 codewords in $\mathcal{G}_{11}$ form $(5, 6, 12)$ and $(4, 5, 11)$ Steiner systems, respectively. Furthermore, note that $\mathcal{G}_{12}$ is self-dual. By (6.20) we immediately have

$$\gamma(\mathcal{G}_{11}^\perp) \leq 66,$$

$$\gamma(\mathcal{G}_{12}) \leq 132.$$

It is worth noting that the recursive procedure used in proving Theorems 6.16 – 6.19 is reminiscent of the zero-neighbors algorithm [19]. It will be interesting to see if such intuition can be materialized so that instead of using all minimal codewords in the dual code (sans those with repeated support), an appreciably smaller subset can be used.

6.3 Non-Erasure Channels

It is widely believed [20] [1] [21] that concepts such as stopping sets and stopping redundancy that are directly motivated by the erasure channel are relevant to other channels as well. We will see an example that supports this claim later in this section. On the other hand, their applicability to a non-erasure channel is clearly limited. An erasure channel is fundamentally different from a non-erasure one in that a symbol transmitted through the channel can be *lost* but is never *distorted*. Consequently, there is no error propagation during message-passing, making iterative decoding more “well-behaved” on an erasure channel. For example, on an erasure channel the addition of parity checks can only improve performance of MPID, while on a non-erasure channel things can go both ways — it is easy to construct examples where the addition of parity check degrades the performance of MPID. One such example is given in [22], and we will see another example later in the section. Also, observe that adding parity checks to a Tanner graph does not remove any cycles but generally introduces new ones, thus can only reduce its girth. While girth is not important for choosing Tanner graph representation of a code on an erasure channel, the presence of small cycles, especially four-cycles, are generally considered detrimental to the performance of MPID on a non-erasure channel (although this position has been contended [23]).

For binary codes, a natural way to extend our discussion to the case of non-erasure channels is to consider pseudo-codewords instead of stopping sets. There are several definitions of pseudo-codewords, some of which equivalent. The most complete definition, pioneered by Wiberg [24], is based on valid configurations of computation trees (CT) that arise from iterative decoding. The idea was further studied in [25] and [26]. Here we follow the polytope characterization introduced by Koetter and Vontobel [27] [21], who essentially showed that a subset of the CT-based pseudo-codewords, when properly scaled, can be compactly represented by a polytope which they called the fundamental polytope. Koetter and Vontobel argued that the main weakness of MPID is that it works *locally* and cannot distinguish the Tanner graph from its finite covers, hence the valid configurations on graph covers is a major contribution to the suboptimality of MPID. They consequently defined graph-cover decoding (GCD) as a tool for analysis of MPID. It turned out that GCD is equivalent to the linear programming decoding (LPD) formulated in [28], and the fundamental polytope coincides with the feasible region of LPD.

We now give a brief description of pseudo-codeword and related concepts following mainly the language of LPD. For more details, the reader is referred to [28] [27] [21].

Let $\mathcal{C} \subseteq \mathbb{F}_2^n$ be a binary linear code. We will also view $\mathcal{C}$ as a subset of $\mathbb{R}^n$ by simply mapping $0 \in \mathbb{F}_2$ to $0 \in \mathbb{R}$ and $1 \in \mathbb{F}_2$ to $1 \in \mathbb{R}$. Assuming a binary-input memoryless symmetric channel and that all information words (hence codewords of $\mathcal{C}$) are equally likely *a priori*, the optimal block decoder works according to the (block) ML rule:

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{c} \in \mathcal{C}} \Pr\{\boldsymbol{y} | \boldsymbol{x} = \boldsymbol{c}\}, \tag{6.21}$$

where $\hat{\boldsymbol{x}}$ is the estimate of the transmitted codeword $\boldsymbol{x}$, and $\boldsymbol{y}$ is the received vector of n channel outputs. Let $\boldsymbol{\gamma}$ denote the vector of log-likelihood ratios (LLR), i.e. for the i-th bit we have

$$\gamma_i = \ln \frac{\Pr\{y_i \mid x_i = 0\}}{\Pr\{y_i \mid x_i = 1\}}.$$

It can be shown [28] that (6.21) is equivalent to solving the following linear program (LP):

$$\boxed{\begin{array}{ll} \text{minimize} & \boldsymbol{\gamma}\boldsymbol{c}^T \\ \text{subject to} & \boldsymbol{c} \in \operatorname{conv}(\mathcal{C}) \end{array}} \tag{6.22}$$

where $\operatorname{conv}(\mathcal{C})$, the *codeword polytope*, is the convex hull of $\mathcal{C}$ in $\mathbb{R}^n$. Since an exact description of $\operatorname{conv}(\mathcal{C})$ is often intractable, the following relaxation was proposed. For any given parity check matrix H, let $\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_m$ denote its rows. Each $\boldsymbol{h}_i$ defines a super code of $\mathcal{C}$, that is

$$\mathcal{C}_i \overset{\text{def}}{=} \{\boldsymbol{x} \in \mathbb{F}_2^n : \boldsymbol{x}\boldsymbol{h}_i^T = 0\},$$

which we call a *local* code. Note that $\mathcal{C}$ is the intersection of all local codes. The codeword polytope of a local code can be easily obtained as

$$\operatorname{conv}(\mathcal{C}_i) = \left\{\boldsymbol{x} \in [0,1]^n : \sum_{j \in V} x_j - \sum_{j \in U_i \setminus V} x_j \le |V| - 1, \forall V \subseteq U_i, |V| \text{ odd}\right\},$$

where $U_i = \operatorname{supp}(\boldsymbol{h}_i)$. Now, let

$$\mathcal{P}(H) \overset{\text{def}}{=} \bigcap_i \operatorname{conv}(\mathcal{C}_i).$$

Then instead of solving (6.22), one can solve the following *relaxed* LP

$$\boxed{\begin{array}{ll} \text{minimize} & \boldsymbol{\gamma}\boldsymbol{c}^T \\ \text{subject to} & \boldsymbol{c} \in \mathcal{P}(H). \end{array}} \tag{6.23}$$

$\mathcal{P}(H)$ is known as the *fundamental polytope* [27]. It can be shown that due to symmetry, we may assume that the all-zero codeword was transmitted. Then (6.23) has the same error probability as the following LP

$$\boxed{\begin{array}{ll} \text{minimize} & \boldsymbol{\gamma}\boldsymbol{c}^T \\ \text{subject to} & \boldsymbol{c} \in \mathcal{K}(H), \end{array}} \tag{6.24}$$

where $\mathcal{K}(H) \overset{\text{def}}{=} \operatorname{conic}\big(\mathcal{P}(H)\big)$, the *fundamental cone*, is the conic hull of $\mathcal{P}(H)$. Following [27] [21], we call any vector in $\mathcal{K}(H)$ a *pseudo-codeword*. A pseudo-codeword is *minimal* if it cannot be expressed as a nonnegative linear combination of other pseudo-codewords. Geometrically, minimal pseudo-codeword form edges of $\mathcal{K}(H)$. Let M be a set that contains one pseudo-codeword from each edge of $\mathcal{K}(H)$. Then $\mathbf{0}$ is a solution to (6.24) if and only if $\boldsymbol{x}\boldsymbol{\gamma}^T \geq 0$ for all $\boldsymbol{x} \in M$. For a given channel, by putting the *pairwise* (between $\mathbf{0}$ and $\boldsymbol{x}$) error probability $\Pr\{\boldsymbol{x}\boldsymbol{\gamma}^T < 0\}$ into formal analogy to that when $\boldsymbol{x} \in \mathcal{C}$, an appropriate definition of *pseudo-weight* can be obtained. By the way pseudo-weight is defined, we see that the performance of LPD (or the equivalent GCD) is largely determined by the pseudo-weight distribution of minimal pseudo-codewords (i.e. edges of the fundamental cone), much like how ML decoding performance is governed by the Hamming weight distribution.

For the AWGN channel, the pseudo-weight is defined as

$$\mathrm{wt}_{\mathrm{AWGN}}(\boldsymbol{x}) \overset{\text{def}}{=} \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{\sum_{i=1}^{n} x_i^2}.$$

For the binary erasure channel (BEC), the pseudo-weight is

$$\mathrm{wt}_{\mathrm{BEC}}(\boldsymbol{x}) \overset{\text{def}}{=} |\operatorname{supp}(x)|.$$

For the BSC, a proper definition can also obtained. As expected, if $\boldsymbol{x} \in \{0,1\}^n$, then pseudo-weight reduces to the Hamming weight.

Let us revisit the example shown in Figure 1.2. Four of the Tanner graphs representing the same $(1,3)$-RM code are reproduced in Figure 6.1, together with their respective AWGN pseudo-weight distribution of minimal pseudo-codewords. The corresponding WER curves using MPID with the min-sum algorithm are shown in Figure 6.2. Note that graph C is based on the construction in [29] (or equivalently, Theorem 5.2) that maximizes stopping distance (and in this case uses the least number of check nodes), and D is based on Theorem 5.1 which has a very neat geometric interpretation. Let H_D denote the parity-check matrix corresponding to Tanner graph D. It can be verified that H_D covers all correctable erasure patterns, hence the iterative H_D-decoder is ML. For perspective, the number of stopping sets of size 3 and 4 are listed in Table 6.2.

We make two qualitative observations. First, we note that there is positive correlation between the distribution of stopping set size and the distribution of AWGN pseudo-weight. In

Table 6.2 Number of stopping sets of sizes 3 and 4 for Tanner graphs in Figure 6.1

	A	B	C	D
size-3	10	2	0	0
size-4	33	24	18	14

some sense this is not very surprising, since essentially we are looking at the same set of pseudo-codewords, it is just the pseudo-weight that is defined differently.[2] More specific results relating stopping sets to pseudo-codewords can be found in [21] [22]. Secondly, from the point of view of pseudo-weight distribution, the fewer the number of smaller-weight pseudo-codewords the better, and we observe that this prediction matches the WER performance comparison very well in this example, i.e. a Tanner graph with better pseudo-weight distribution produces less errors in MPID.

Up to now everything looks very good: the above example seems to suggest that the stopping set size distribution is a good indicator of the pseudo-weight distribution, which in turn is a good indicator of MPID performance. While as just shown, there are cases where these claims are true, it is not very clear to what extent they hold in general.

In fact, there are examples [21] where stopping set size and AWGN pseudo-weight can be dramatically different. On the other hand, neither is the distribution of pseudo-weight always an accurate indicator of MPID performance. For example, consider the set of four Tanner graphs with their respective distribution of AWGN pseudo-weight of minimal pseudo-codewords shown in Figure 6.3. Two of them, B and C, are replicated from Figure 6.1. The other two, F and G, are both obtained from B by including one additional check. In fact, C can be viewed in this way as well. Thus, taking B as baseline, this comparison is targeted at how choice of an additional parity check may have positive or negative impact on the MPID performance, and how this impact may or may not be explained by the change in pseudo-weight distribution. For reference, the number of size-3 and size-4 stopping sets for the graphs in question are provided in Table 6.3.

We observe that while the correlation between distributions of stopping set size and AWGN pseudo-weight can again be seen, the pseudo-weight distribution no longer gives a sat-

[2] To be precise, by counting stopping sets we are not distinguishing between pseudo-codewords that have the same support, i.e. all pseudo-codewords with the same support are counted as just one stopping set.

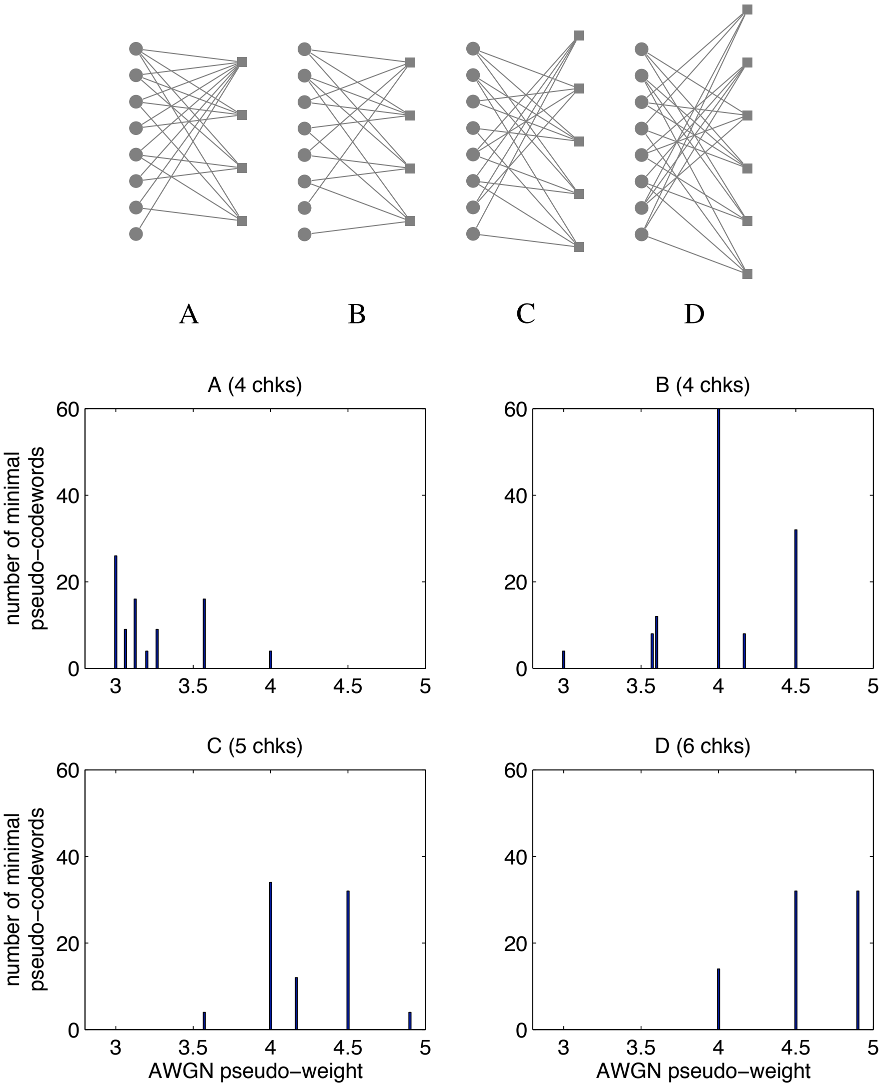

Figure 6.1 Four Tanner graphs representing the $(1,3)$–RM code and their respective AWGN pseudo-weight distribution of minimal pseudo-codewords. A is the natural choice if the code is viewed as an extended Hamming code. B comes from the $(\boldsymbol{u} \mid \boldsymbol{u}+\boldsymbol{v})$ recursive construction of RM codes [12]. C is based on the construction in [29] (or equivalently, Theorem 5.2) that maximizes stopping distance (in this case with the least number of check nodes). D is based on Theorem 5.1 which has a neat geometric interpretation. It can be verified that MPID over graph D in fact achieves ML performance on the binary erasure channel.

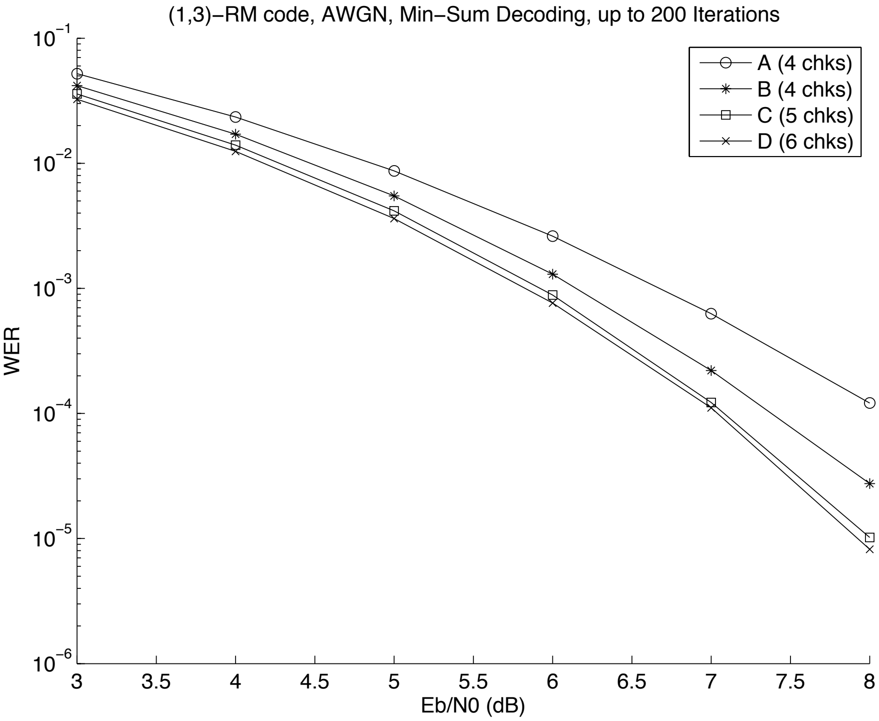

Figure 6.2 WER performance for Tanner graphs in Figure 6.1. Min-sum algorithm with up to 200 iterations.

Table 6.3 Number of stopping sets of sizes 3 and 4 for Tanner graphs in Figure 6.3

	B	C	F	G
size-3	2	0	1	2
size-4	24	18	21	20

isfactory explanation of the differences in MPID performance among different Tanner graphs. In fact, having one additional parity check, graph F recorded *worse* WER performance than graph B, despite its superior pseudo-weight distribution. Another unusual comparison is between graphs C and G. From their pseudo-weight distributions, one would expect C to have a notable advantage. However, the two graphs perform very similarly under iterative decoding. And if there was any difference, G seems to perform very slightly better.

The main cause for such inconsistencies, as pointed out in [22], is that the set of pseudo-codewords as we have defined it is not enough to fully describe the behavior of MPID. The more general set of pseudo-codewords according to [24], [26] or [25] would be appropriate, but unfortunately does not have a tractable mathematical characterization.

Despite its limitations, evaluating Tanner graphs based on pseudo-codewords distribution is certainly appropriate for LPD/GCD, and due to the close relation between LPD/GCD and MPID, still provides a valuable means to understand the effects of Tanner graph representation on the performance of MPID. Work on the relation between LPD/GCD and MPID can be found in [28] [21] [30] [22]. Notably, it is observed in [31] that LPD can be viewed as a high SNR limit of MPID. Also, similarities in decision regions have been noted for LPD and the min-sum algorithm, and a detailed discussion of the relation between the two algorithms can be found in [30]. On the BEC, it can be shown [28] that LPD and MPID are equivalent.

To obtain the WER curves in Figure 6.2 and Figure 6.4, the min-sum algorithm has been used due to its closer relation to LPD. Although not shown here, similar performance comparison is observed when the sum-product algorithm is used.

Since performance of LPD/GCD can only improve with the addition of parity checks, there exists a similar complexity-performance tradeoff as in the case of erasure decoding. At high SNRs, LPD/GCD performance (and in the limit, MPID performance) is governed by the minimum pseudo-weight among all nonzero pseudo-codewords. Since the set of pseudo-codewords

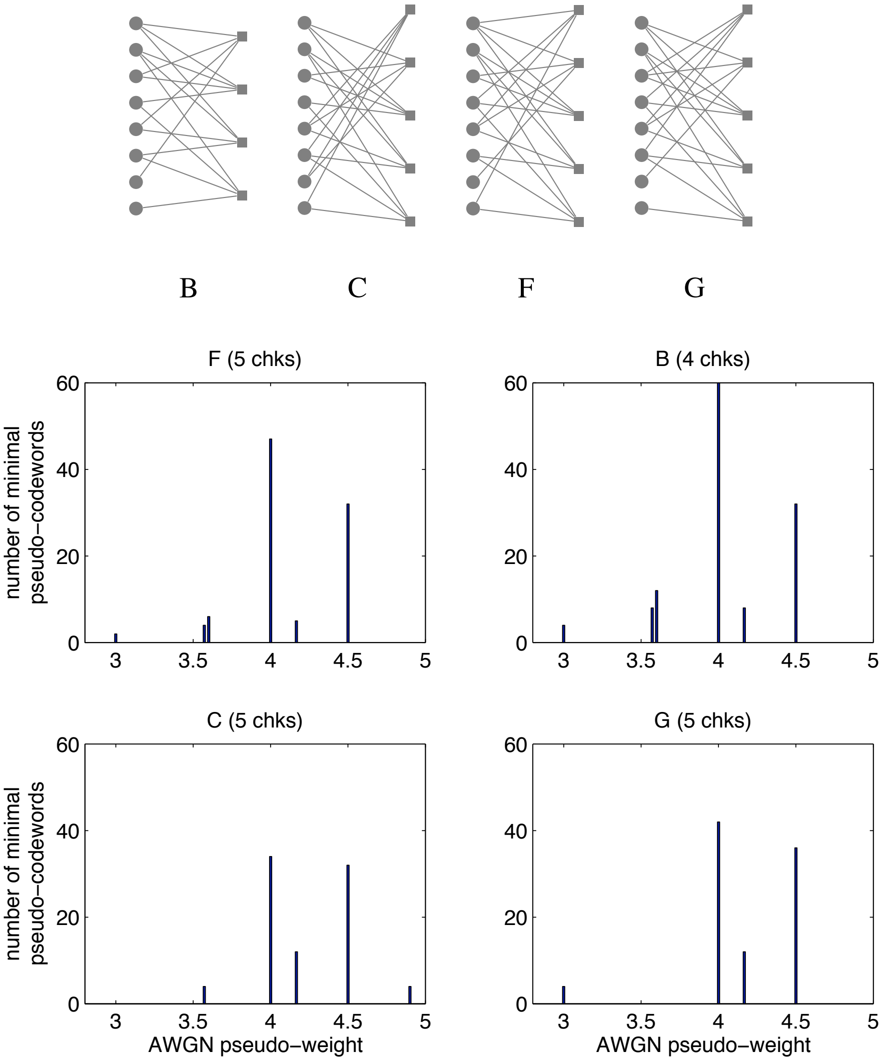

Figure 6.3 Four Tanner graphs representing the $(1, 3)$–RM code and their respective AWGN pseudo-weight distribution of minimal pseudo-codewords. Graphs B and C are the same as in Figure 6.1. F and G are both obtained from B by adding one additional check node. Note that C can be viewed in a similar way. Thus, the difference among C, F and G lies in the choice of the additional check node to be added to the Tanner graph B.

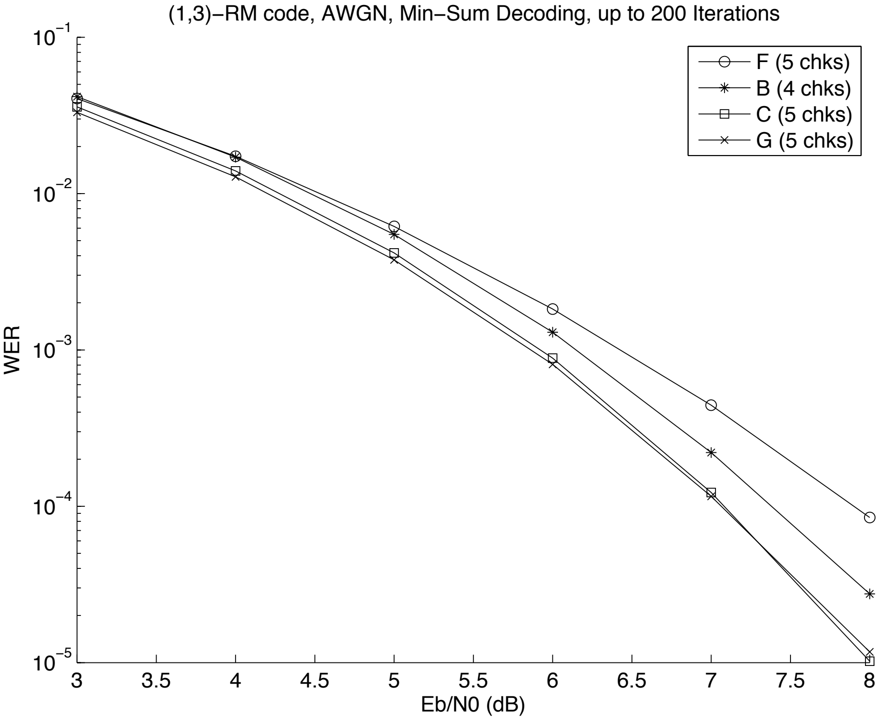

Figure 6.4 WER performance for Tanner graphs in Figure 6.3. Min-sum algorithm with up to 200 iterations.

contains the code itself as a subset, it can easily be shown that the minimum pseudo-weight for BSC, BEC and the AWGN channel are all upper bounded by the minimum Hamming distance of the code. In the case of BEC, this upper bound can always to be achieved and the least number of check nodes for doing so is the stopping redundancy. Recently, Kelley and Sridhara showed in [32] that the upper bound is also always achievable on the BSC, and correspondingly defined the *pseudo-weight redundancy* for the BSC. It was then shown that many of the results on stopping redundancy extend to the BSC case with little modification (though the proofs can be quite different).

We have seen that pseudo-weight distribution as an indicator of MPID performance is not without its limitations. Hence, it may be worthwhile to consider alternatives. Some worthy candidates include *trapping sets* [33] [34] and *extrinsic message degree (EMD)* [35]. An initial study based on trapping sets can be found in [36] [20]. In [20], the authors defined *trapping redundancy* in analogy to stopping redundancy. We note that since trapping sets and EMD share similar combinatorial structures with stopping sets, much of the the tools and techniques that we used can be extended rather easily.

6.4 Further Discussion

While the study in this thesis has added to our understanding of how code representation affects its performance under MPID, many interesting questions remain and further study is certainly warranted. Some of the open questions are already mentioned in earlier chapters, here we add a few more.

An important aspect that we have not paid much attention to is how one practically finds a good representation of a code suitable for MPID. We indeed talked about a very specific construction for Reed-Muller codes, which, although coinciding with the construction in [29] up to automorphism, provides an interesting geometric perspective that can be applied to other finite geometry codes. The more general results in this area so far focused on three approaches. The first one is based on using parity checks of minimum weight to form the parity-check matrix [37]. The second method focuses on small loops in the Tanner graph, and form new parity checks by linearly combining check equations involved in a loop [5] [38]. This method is partly justified by the fact that additional parity checks obtained by linearly combining less than $g/2$

parity checks in a Tanner graph with girth g does not change the fundamental polytope [21] [39]. A third method (e.g. [36]) is to select the additional parity checks based on trapping set analysis. Due to its clear practical interest, more work to further our understanding in this area is much desired.

Another interesting pursuit is to find codes that have small stopping/pseudo-weight redundancy. Clearly, if the stopping redundancy is too large, then it is practically impossible to use a corresponding Tanner graph for MPID due to complexity reasons. By "small", we usually require that the stopping redundancy grow linearly with the redundancy of the code. Some initial work in this direction can be found in [29] [18]. As shown in [29], for a fixed order l, the stopping redundancy of Reed-Muller codes $\mathcal{R}(l, m)$ is linear in the redundancy of the code. Moreover, as discussed in Chapter 3, the upper bound (3.5) implies that for a fixed d, the stopping redundancy of any binary linear code grows linearly with the redundancy of the code. Another example is the class of LDPC codes based on finite geometries [40]. It can be shown [41] [42] that the stopping/pseudo-weight redundancy of these codes is at most polynomial in the redundancy. Note that in all the above examples, either the rate or the distance of the code (as a fraction of block length) approaches zero as the block length is increased. It remains a question whether there exists a family of good linear codes whose stopping redundancy grows polynomially with redundancy of the code.

Moving on to non-erasure channels, we see that pseudo-codewords provide a reasonable framework for the evaluation of different Tanner graphs. We have seen that the minimum pseudo-weight is upper bounded by the minimum Hamming distance for any channel. For the BEC and BSC, it has been shown that this upper bound can always be achieved. It is a theoretically interesting question whether the same conclusion holds for other channels, e.g. the binary-input AWGN channel. We also talked about ML redundancy. Note that when we move to a non-erasure channel it is usually impossible to achieve ML performance via MPID on *any* Tanner graph. Even for the LPD/GCD, there are many codes for which ML performance is not achievable by adding more parity checks, and a precise characterization of these codes can be found using results in Matroid theory [43] [44]. For those codes, even when all possible parity checks are present, the vertex set of the fundamental polytope will still contain non-codeword pseudo-codewords. Nevertheless, corresponding to ML redundancy, it may still be an interesting question to consider, for example, the number of parity checks required to minimize the volume of the fundamental polytope.

We should also mention that we have focused on just one dimension in varying the Tanner graph for a code: we choose how many and which parity checks to use. By using redundant parity checks, we expand along the row dimension of a parity-check matrix. Note that we can also expand along the column dimension to create “auxiliary” variable nodes which are considered punctured out before transmission, and are treated as erasures at the decoder. This idea was introduced in [45], and by allowing both redundant rows and redundant columns, the authors defined the *generalized parity-check matrix*. There are certain benefits of having punctured variable nodes in a Tanner graph, for example, to break small cycles. How this added dimension affects the performance-complexity tradeoff in code representation remains to be seen.

Finally, we should be reminded that even pseudo-codewords (as we have defined them) cannot accurately predict MPID performance, and thus should be used only as an approximate tool for the analysis. Indeed, an accurate characterization of MPID performance by mathematically tractable structures in a Tanner graph is likely a near impossible task. As more and more redundant parity checks are employed, LPD/GCD performance steadily improves, but it is not clear to what extent the MPID performance will follow suit. When there are many check nodes in the graph, problems may arise from having too many short cycles and/or too many pseudo-codewords that are not included in the polytope characterization.

But another way to look at this problem is that in a sense a larger graph never hurts: in principle we could emulate the performance of a subgraph of it by *scheduling*. So if in some case MPID performance degrades due to the addition of parity checks, it really is that we are not using the over-defined description efficiently. A variation of the normal MPID algorithm may take better advantage of the redundant representation. One such idea is to represent the same code with *multiple* Tanner graphs in a turbo-like fashion [46] or to perform list decoding [47] [48]. Another powerful idea is to *adapt* the representation based on the received vector (hence we are selecting a subset of parity checks to use from a large pool), which was considered in [49] [50] [38]. Further investigation in these above directions may prove fruitful.

Acknowledgment

This chapter is in part a reprint of the material in the paper: J. Han and P. H. Siegel, "On ML redundancy of codes," in *Proc. IEEE Int. Symp. Inform. Theory*, Toronto, Canada, July 2008, to appear. The dissertation author was the primary author of this paper.

Bibliography

[1] M. Schwartz and A. Vardy, "On the stopping distance and the stopping redundancy of codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.

[2] T. Hehn, S. Laendner, O. Milenkovic, and J. B. Huber, "The stopping redundancy hierarchy of cyclic codes," in *Proc. 44-th Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, Sept. 2006, pp. 1271–1280.

[3] T. Hehn, O. Milenkovic, S. Laendner, and J. B. Huber, "Permutation decoding and stopping redundancy hierarchy of linear block codes," in *Proc. IEEE International Symposium on Information Theory*, Nice, France, June 2007, pp. 2926–2930.

[4] K. A. Abdel-Ghaffar and J. H. Weber, "Generalized stopping sets and stopping redundancy," in *Proc. IEEE Information Theory and Applications Workshop*, San Diego, CA, Jan.–Feb. 2007, pp. 400–404.

[5] H. Pishro-Nik and F. Fekri, "On decoding of low-density parity-check codes over the binary erasure channel," *IEEE Trans. Inform. Theory*, vol. 50, no. 3, pp. 439–454, Mar. 2004.

[6] A. Shokrollahi, S. Lassen, and R. M. Karp, "Systems and processes for decoding chain reaction codes through inactivation," U.S. Patent 6 856 263, Feb. 15, 2005.

[7] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[8] H. Pishro-Nik and F. Fekri, "Improved decoding algorithms for low-density parity-check code," in *Proc. 3rd International Symposium on Turbo Codes and Related Topics*, Brest, France, 2003.

[9] N. Varnica and M. Fossorier, "Belief-propagation with information correction: Improved near maximum-likelihood decoding of low-density parity-check codes," in *Proc. IEEE International Symposium on Information Theory*, Chicago, USA, June/July 2004, p. 343.

[10] H. D. L. Hollmann and L. M. G. M. Tolhuizen, "Generic erasure-correcting sets: bounds and constructions," *J. Combin. Theory Ser. A*, vol. 113, pp. 1746–1759, Nov. 2006.

[11] ——, “On parity check collections for iterative erasure decoding that correct all correctable erasure patterns of a given size,” *IEEE Trans. Inform. Theory*, vol. 53, no. 2, pp. 823–828, Feb. 2007.

[12] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1978.

[13] J. H. Weber and K. A. S. Abdel-Ghaffar, “Results on parity-check matrices with optimal stopping and/or dead-end set enumerators,” *IEEE Trans. Inform. Theory*, vol. 54, no. 3, pp. 1368–1374, Mar. 2008.

[14] T.-Y. Hwang, “Decoding linear block codes for minimizing word error rate,” *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 733–737, Nov. 1979.

[15] E. Agrell, “Voronoi regions for binary linear block codes,” *IEEE Trans. Inform. Theory*, vol. 42, no. 1, pp. 310–316, Jan. 1996.

[16] A. Ashikhmin and A. Barg, “Minimal vectors in linear codes,” *IEEE Trans. Inform. Theory*, vol. 44, no. 5, pp. 2010–2017, Sept. 1998.

[17] J. L. Massey, “Minimal codewords and secret sharing,” in *Proc. 6th Joint Swedish-Russian Workshop on Information Theory*, Rölle, Sweden, 1993, pp. 246–249.

[18] T. Etzion, “On the stopping redundancy of Reed-Muller codes,” *IEEE Trans. Inform. Theory*, vol. 52, no. 11, pp. 4867–4879, Nov. 2006.

[19] L. B. Levitin and C. R. P. Hartmann, “A new approach to the general minimum distance decoding problem: The zero-neighbors algorithm,” *IEEE Trans. Inform. Theory*, vol. 31, no. 3, pp. 378–384, May 1985.

[20] O. Milenkovic, E. Soljanin, and P. Whiting, “Stopping and trapping sets in generalized covering arrays,” in *Proc. 40th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, Mar. 2006, pp. 259–264.

[21] P. O. Vontobel and R. Koetter, “Graph-cover decoding and finite-length analysis of message-passing iterative decoding of LDPC codes,” submitted to *IEEE Trans. Inform. Theory*, 2005. [Online]. Available: http://arxiv.org/abs/cs.IT/0512078

[22] C. Kelley and D. Sridhara, “Pseudocodewords of Tanner graphs,” *IEEE Trans. Inform. Theory*, vol. 53, no. 11, pp. 4013–4038, Nov. 2007.

[23] R. McEliece, “Girth doesn’t matter (or does it?),” presented at the ITA Inauguration Workshop, UCSD, La Jolla, CA, 2006.

[24] N. Wiberg, “Codes and decoding on general graphs,” Ph. D. dissertation, Linköping University, 1996.

[25] B. J. Frey, R. Koetter, and A. Vardy, "Signal-space characterization of iterative decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 766–781, Feb. 2001.

[26] G. D. Forney, R. Koetter, F. R. Kschischang, and A. Reznik, "On the effective weights of pseudocodewords for codes defined on graphs with cycles," in *Codes, Systems and Graphical Models*. New York: Springer-Verlag, 2001, pp. 101–112.

[27] R. Koetter and P. Vontobel, "Graph covers and iterative decoding of finite-length codes," in *Proc. 3rd International Symposium on Turbo Codes and Related Topics*, Brest, France, Sept. 2003, pp. 75–82.

[28] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inform. Theory*, vol. 51, no. 3, pp. 954–972, Mar. 2005.

[29] M. Schwartz and A. Vardy, "On the stopping distance and stopping redundancy of codes," in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 975–979.

[30] P. O.Vontobel and R. Koetter, "On the relationship between linear programming decoding and min-sum algorithm decoding," in *Proc. International Symposium on Information Theory and Applications*, Parma, Italy, Oct. 2004, pp. 991–996.

[31] M. Chertkov and M. Stepanov, "Pseudo-codeword lanscape," in *Proc. IEEE International Symposium on Information Theory*, Nice, France, June 2007, pp. 1546–1550.

[32] C. A. Kelley and D. Sridhara, "On the pseudocodeword weight and parity-check matrix redundancy of linear codes," in *Proc. IEEE Information Theory Workshop*, Lake Tahoe, California, Sept. 2007, pp. 1–6.

[33] D. MacKay and M. Postol, "Weaknesses of Margulis and Ramanujan-Margulis low-density parity-check codes," *Electronic Notes in Theoretical Computer Science*, vol. 74, 2003. [Online]. Available: http://www.elsevier.nl/locate/entcs/volume74.html

[34] T. Richardson, "Error-floors of LDPC codes," in *Proc. 41st Annual Conference on Communication, Control and Computing*, Monticello, IL, Sept. 2003, pp. 1426–1435.

[35] T. Tian, C. R. Jones, J. D. Villasenor, and R. D. Wesel, "Selective avoidance of cycles in irregular LDPC code constructions," *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1242–1247, 2004.

[36] S. Laendner, T. Hehn, O. Milenkovic, and J. B. Huber, "When does one redundant parity-check equation matter?" submitted to Globecom 2006, 2006.

[37] R. Lucas, M. Bossert, and M. Breitbach, "On iterative soft-decision decoding of linear binary block codes and product codes," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 276–296, Feb. 1998.

[38] M. H. Taghavi and P. H. Siegel, “Adaptive linear programming decoding,” in *Proc. IEEE International Symposium on Information Theory*, Seattle, WA, July 2006, pp. 1374–1378.

[39] M. J. Wainwright, “Codeword polytopes and linear programming relaxations for error control coding,” presented at the Workshop on Applications of Statistical Physics to Coding Theory, Santa Fe, New Mexico, 2005. [Online]. Available: http://cnls.lanl.gov/˜chertkov/EC_Talks/Wainwright/

[40] Y. Kou, S. Lin, and M. P. C. Fossorier, “Low-density parity-check codes based on finite geometries: A rediscovery and new results,” *IEEE Trans. Inform. Theory*, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.

[41] N. Kashyap and A. Vardy, “Stopping sets in codes from designs,” in *Proc. IEEE International Symposium on Information Theory*, Yokohama, Japan, June/July 2003, p. 122.

[42] C. Kelley, D. Sridhara, J. Xu, and J. Rosenthal, “Pseudocodeword weights and stopping sets,” in *Proc. IEEE International Symposium on Information Theory*, Chicago, USA, June/July 2004, p. 68.

[43] P. D. Seymour, “Matroids and multicommodity flows,” *European J. Combin.*, no. 2, pp. 257–290, 1981.

[44] F. Barahona and M. Grötschel, “On the cycle polytope of a binary matroid,” *J. Combin. Theory Ser. B*, vol. 40, pp. 40–62, 1986.

[45] J. Yedidia, J. Chen, and M. Fossorier, “Generating code representations suitable for belief propagation decoding,” in *Proc. 40-th Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, Sept. 2002, pp. 447–456.

[46] T. R. Halford and K. M. Chugg, “Random redundant soft-in soft-out decoding of linear block codes,” in *Proc. IEEE International Symposium on Information Theory*, Seattle, USA, July 2006, pp. 2231–2235.

[47] M. Marrow, “Detection and modeling of 2-dimensional signals,” Ph. D. dissertation, University of California, San Diego, 2004.

[48] T. Hehn, J. B. Huber, S. Laendner, and O. Milenkovic, “Multiple-bases belief-propagation for decoding of short block codes,” in *Proc. IEEE International Symposium on Information Theory*, Nice, France, June 2007, pp. 311–315.

[49] J. Jiang and K. R. Narayanan, “Iterative soft decision decoding of Reed Solomon codes based on adaptive parity check matrices,” in *Proc. IEEE International Symposium on Information Theory*, Chicago, USA, June/July 2004, p. 258.

[50] A. Kothiyal and O. Y. Takeshita, “A comparison of adaptive belief propagation and the best graph algorithm for the decoding of linear block codes,” in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 724–728.

Chapter 7

Probability of Undetected Error for Over-Extended Reed-Solomon Codes

7.1 Introduction

In some applications, error correcting codes have been used as pure error detection codes. In particular, Reed-Solomon (RS) codes have been used for error detection in some disk drives since the 1990's because they have excellent error detection capabilities and do not exhibit the undesirable behavior characteristic of certain shortened binary cyclic redundancy check (CRC) codes [1]. A further example is the USB interface standard [2], which specifies the use of a Hamming code for error detection.

Typically, the error detecting capabilities of these codes are guaranteed only when the codeword length is limited to some maximum number of symbols. For RS codes defined over a finite field with q elements, $\mathbb{F}_q$, the maximum length is $q-1$ symbols (or q symbols for an extended code). However, for various reasons such as format efficiency, we sometimes use an *over-extended code*, where the codeword length is allowed to exceed this maximum length. For example, a 16-bit, binary CRC is most often used to protect codewords consisting of $n = 2^{15} - 1$ or fewer bits. However, the Ultra DMA mode in the ATA standard [3] specifies the use of a 16-bit CRC for protecting data packets of length much greater than n bits.

When a block code is used solely for error detection, the decoder announces the received word to be free of error if it is found in the codebook. However, errors may have occured in such a way that the received word is a codeword different from the one transmitted, in which case the errors will not be detected. The probability of such an event is known as the *probability of undetected error*, and is denoted by P_{ud}.

Consider an $[n, k]$ linear block code over $\mathbb{F}_q$, transmitted over a q-ary symmetric channel, where each transmitted symbol is received correctly with probability $1 - p$, and as any of the other $q - 1$ symbols with equal probability $p/(q - 1)$. Clearly, for this channel, P_{ud} can be calculated as a function of p as follows:

$$P_{ud}(p) = \sum_{i=1}^{n} A_i \left(\frac{p}{q-1} \right)^i (1-p)^{n-i}, \tag{7.1}$$

where A_i is the number of codewords with Hamming weight i. Equation (7.1) relates the probability of undetected error directly to the weight distribution of the code. Alternatively, $P_{ud}(p)$ can also be obtained from the weight distribution of the dual code, as follows:

$$P_{ud}(p) = q^{-(n-k)} \sum_{i=0}^{n} A_i^{\perp} \left(1 - \frac{qp}{q-1} \right)^i - (1-p)^n, \tag{7.2}$$

where $A_i^{\perp}$ is the number of codewords with Hamming weight i in the dual code. This can be conveniently shown from (7.1) using the MacWilliams identity [4], [5].

When $p = (q - 1)/q$, the received symbols appear to be uniformly distributed no matter which codeword was transmitted. Therefore, undetected error occurs when the received word is any codeword except the one sent and each such codeword appears with probability q^{-n}. Since there are $q^k - 1$ such incorrect codewords, we have

$$P_{ud}\left(\frac{q-1}{q} \right) = (q^k - 1)q^{-n} < q^{-(n-k)}.$$

The same result can be obtained directly from (7.1). Note that this "purely random" case does not necessarily correspond to the worst-case error detection performance [6], [7], [8], [9], for $0 \leq p \leq \frac{q-1}{q}$. Intuitively, if the weight distribution of the code is concentrated near certain weights, it is more likely that a codeword is confused with another when typically certain numbers of errors occur, rather than when typically an exceedingly large number of errors occur. For the same reasons, $P_{ud}(p)$ is not guaranteed to be a monotonic function of p for $0 \leq p \leq \frac{q-1}{q}$, though in [8] the authors were able to show that except for certain trivial classes of codes, $P_{ud}(p)$ is well-behaved in the vicinity of $\frac{q-1}{q}$ (i.e. $P'_{ud}(\frac{q-1}{q}) > 0$).

Following [7], [10], [11], we call a code *good*[1] if $P_{ud}(p) < q^{-(n-k)}$ for all $0 \leq p \leq \frac{q-1}{q}$, and *proper* if $P_{ud}(p)$ is monotonic in p for $0 \leq p \leq \frac{q-1}{q}$. (Some authors have used $q^{-(n-k)} - q^{-n}$ as the goodness threshold. See [12].) Proper codes are necessarily good, but not vice versa. Properness and goodness properties of certain classes of codes are addressed in [6], [7], [10], [11], [9]. In particular, MDS codes (e.g. RS codes) are known to be good and proper [11]. Note also that for the ensemble of all $[n, k]$ linear block codes over $\mathbb{F}_q$, it is known [12] that the average probability of undetected error is

$$P_{ud}^{\text{avg}}(p) = \frac{q^k - 1}{q^n - 1}\big(1 - (1-p)^n\big).$$

For systematic codes, a similar result is known [13], [8]:

$$P_{ud}^{\text{sys}}(p) = q^{-(n-k)}\big(1 - (1-p)^k\big).$$

Note that in either case, the average performance of a randomly chosen code satisfies the conditions for both goodness and properness.

In this paper, we consider *Over-Extended Reed-Solomon (OERS)* codes. From a practical point of view, these codes are constructed by using a (shift register type) RS encoder but allowing a longer input. Let $\mathcal{C}$ be a RS code over $\mathbb{F}_q$ with length $n = q-1$ and minimum distance d. Then $\mathcal{C}$ can be described as the set of polynomials $c(x)$ such that

$$c(x) = -r(x) + x^{d-1}u(x), \tag{7.3}$$

where $u(x)$ is the data polynomial of degree at most $n-d$, and $r(x)$ is the remainder of $x^{d-1}u(x)$ divided by $g(x)$, the generator polynomial of $\mathcal{C}$. An OERS code $\mathcal{C}'$ can then be defined simply by allowing $u(x)$ in (7.3) to have degree higher than $n-d$, such that the length of the code is extended to $n' > n$. This results in a linear $[n', n'-d+1]$ code over $\mathbb{F}_q$.

The rest of the paper is arranged as follows. In Section 7.2, we derive upper and lower bounds on the weight distribution of OERS codes. In Section 7.3, we apply the results of Section 7.2 to obtain bounds on the probability of undetected error for OERS codes on q-ary symmetric channels. We show that the bounds are asymptotically tight, which is corroborated by an example. Section 7.4 concludes the paper. Proofs, where not given, are either evident or can be found in Appendix Appendix 7.A.

[1] This use of the term "good" is to be distinguished from as in *good family* of codes that we came across in previous chapters, which refers a family of codes whose distance and rate (as fractions of block length) are both bounded away from zero as the block length approaches infinity.

7.2 Weight Distribution

First, a few remarks on notation. Throughout the rest of the paper, unless otherwise stated, $\mathcal{C}$ is a RS code over $\mathbb{F}_q$ with length $n = q - 1$, minimum distance d, and generator polynomial $g(x)$; $\mathcal{C}'$ is the OERS code constructed from $\mathcal{C}$ with length $n' > n$. In most of our discussions, $\mathcal{C}$ and $\mathcal{C}'$ will be interpreted as subsets of $\mathbb{F}_q[x]$, the ring of polynomials with coefficients in $\mathbb{F}_q$. If $c(x) \in \mathbb{F}_q[x]$, then $\deg\bigl(c(x)\bigr)$ is the degree of $c(x)$, and $\mathrm{wt}\bigl(c(x)\bigr)$ is the number of nonzero terms in $c(x)$, i.e. the Hamming weight of the corresponding vector of coefficients. For any Euclidean domain D and $a, b \in D$, $R_a[b]$ is the remainder of b divided by a. Vectors are indicated in bold. If $\boldsymbol{x}$ is a vector, then $|\boldsymbol{x}|$ is the dimension of $\boldsymbol{x}$.

From the definition of OERS codes given in the previous section, it is easy to show that $\mathcal{C}'$ is also the set of polynomials in $\mathbb{F}_q[x]$ that have degrees less than n' and are divisible by $g(x)$. This is the definition that we will use most often.

Since $g(x) \mid x^n - 1$, we know that $x^n - 1 \in \mathcal{C}'$ for all $n' > n$. Therefore, all OERS codes contain codewords of weight-2, and thus have minimum distance $\min(d, 2)$.

Let $\mathcal{A}_i'$ denote the set of weight-i codewords of $\mathcal{C}'$. We are interested in finding $A_i' = |\mathcal{A}_i'|$ for all i. For very low weights, the problem of determining the corresponding term in the weight enumerator is tractable – we can fully characterize all codewords of a given low weight and thereby count them. The results for weight-2 and weight-3 codewords are summarized in the following propositions.

Proposition 7.1 *The number of weight-*2 *codewords in an OERS code is*

$$A_2' = \begin{cases} \binom{a}{2}(q-1)^2 + ab(q-1) & \textit{if } d > 2 \\ \binom{n'}{2}(q-1) & \textit{if } d = 2 \\ \binom{n'}{2}(q-1)^2 & \textit{if } d = 1, \end{cases}$$

where a and b are integers such that $n' = an + b$, $0 \le b < n$.

Proof: See Appendix Appendix 7.A. ■

Corollary 7.2 *If $d > 2$ and $n \mid n'$, then $A_2' = \binom{n'/n}{2}(q-1)^2$.*

For example, if the OERS code has twice the length of the original RS code, then $A_2' = (q-1)^2$ if $d > 2$.

Proposition 7.3 *The number of weight-*3 *codewords in an OERS code is*

$$A'_3 = \begin{cases} \Big(\binom{a}{3}(q-1) + \binom{a}{2}b\Big)(q-1)(q-2) & \textit{if } d > 3 \\ \Big[\binom{n'}{3} - (n'-2)\Big(\binom{a}{2}(q-1) + ab\Big) + q\Big(\binom{a}{3}(q-1) + \binom{a}{2}b\Big)\Big](q-1) & \textit{if } d = 3 \\ \binom{n'}{3}(q-1)(q-2) & \textit{if } d = 2 \\ \binom{n'}{3}(q-1)^3 & \textit{if } d = 1, \end{cases}$$

where a *and* b *are integers such that* $n' = an + b$*,* $0 \le b < n$*.*

Proof: See Appendix Appendix 7.A. ■

The study of these special cases motivates a general approach to understanding the entire weight distribution of OERS codes. The following two lemmas, though elementary, are the basis of much of the discussion that follows.

Lemma 7.4 *If* $c(x) \in \mathbb{F}_q[x]$ *and* $\deg(c) < n'$*, then* $c(x) \in \mathcal{C}'$ *if and only if* $R_{x^n-1}[c(x)] \in \mathcal{C}$*.*

Proof: Note $R_{g(x)}[c(x)] = R_{g(x)}\big[R_{x^n-1}[c(x)]\big]$. ■

Lemma 7.5 *For all* $c(x) \in \mathbb{F}_q[x]$*,*

$$\mathrm{wt}\big(c(x)\big) \ge \mathrm{wt}\big(R_{x^n-1}[c(x)]\big).$$

Proof: If $c(x) = \sum_{i=0}^{m} c_i x^i$, then

$$R_{x^n-1}[c(x)] = \sum_{i=0}^{m} c_i x^{R_n[i]} = \sum_{j=0}^{n-1} r_j x^j,$$

where $r_j = \sum_{i:R_n[i]=j} c_i$. For each j such that $r_j \neq 0$, there exists i, $i \equiv j \mod n$, such that $c_i \neq 0$. ■

From Lemma 7.4, since $0 \in \mathcal{C}$, if we define $\mathcal{B}'_i \stackrel{\text{def}}{=} \{c(x) \in \mathbb{F}_q[x] : \mathrm{wt}\big(c(x)\big) = i, \deg\big(c(x)\big) < n', x^n - 1 \mid c(x)\}$, then $\mathcal{B}'_i \subseteq \mathcal{A}'_i$. We first show how $B'_i \stackrel{\text{def}}{=} |\mathcal{B}'_i|$ can be calculated. For $n' \le 2n$, the situation is particularly simple.

Proposition 7.6 *If* $n' \le 2n$*, then for all* i*,*

$$B'_i = \begin{cases} \binom{n'-n}{i/2}(q-1)^{i/2} & \textit{if } i \textit{ is even} \\ 0 & \textit{if } i \textit{ is odd} \end{cases} \tag{7.4}$$

Proof: Note that $\mathcal{B}'_i = \{c(x) \in \mathbb{F}_q[x] : \mathrm{wt}(c(x)) = i, c(x) = (x^n - 1)a(x), a(x) \in \mathbb{F}_q[x], \deg(a) < n' - n\}$. If $n' \leq 2n$, then $\deg(a(x)) < n$, which implies that $a(x)$ and $x^n a(x)$ have no powers of x in common. Therefore, $i = \mathrm{wt}(c) = 2\,\mathrm{wt}(a)$. This is only possible if i is even. And the number of such $c(x)$'s is precisely the number of $a(x)$'s such that $\deg(a) < n' - n$ and $\mathrm{wt}(a) = i/2$. ■

In general, for every $c(x) = \sum_{j=0}^{\deg(c)} c_j x^j \in \mathbb{F}_q[x]$, denote its support set as

$$\mathcal{W}(c) \overset{\text{def}}{=} \{j : 0 \leq j \leq \deg(c), c_j \neq 0\}.$$

Given a positive integer n, we can write

$$\mathcal{W}(c) = \bigcup_{l=0}^{n-1} \mathcal{W}(c) \cap (l + n\mathbb{Z}) \tag{7.5}$$

$$= \bigcup_{l \in \mathcal{L}_n(c)} \mathcal{W}_{n,l}(c) \tag{7.6}$$

where $\mathcal{W}_{n,l}(c) \overset{\text{def}}{=} \mathcal{W}(c) \cap (l + n\mathbb{Z})$ are those indices in the support set of $c(x)$ that are congruent to l modulo n, and $\mathcal{L}_n(c) \overset{\text{def}}{=} \{l : 0 \leq l < n, \mathcal{W}_{n,l}(c) \neq \emptyset\}$. Clearly, $(\mathcal{W}_{n,l})_{l \in \mathcal{L}_n(c)}$ is a partition of the set $\mathcal{W}(c)$. Hence,

$$\mathrm{wt}(c) = |\mathcal{W}(c)| = \sum_{l \in \mathcal{L}_n(c)} |\mathcal{W}_{n,l}(c)|.$$

Let $\mathcal{L}_n(c)$ be ordered such that $\mathcal{L}_n(c) = \{l_1, l_2, \ldots, l_{|\mathcal{L}_n(c)|}\}$, where $l_1 < l_2 < \ldots < l_{|\mathcal{L}_n(c)|}$. Define the *$n$-ary support profile* of $c(x)$ as

$$\boldsymbol{w}_n(c) \overset{\text{def}}{=} \left(|\mathcal{W}_{n,l_1}(c)|, |\mathcal{W}_{n,l_2}(c)|, \ldots, |\mathcal{W}_{n,l_{|\mathcal{L}_n(c)|}}(c)|\right).$$

Then $\boldsymbol{w}_n(c)$ is an ordered partition of $\mathrm{wt}(c)$. We count $\mathcal{B}'_i$ by counting subsets of $\mathcal{B}'_i$ corresponding to specific n-ary support profiles. Let $\mathcal{P}_i$ be the set of all ordered partitions of i, i.e., $\mathcal{P}_i \overset{\text{def}}{=} \{\boldsymbol{\delta} \in \mathbb{N}^* : \sum_j \delta_j = i\}$, where $\mathbb{N}^* = \bigcup_{j=1}^{\infty} \mathbb{N}^j$ is the set of vectors of natural numbers. For all $\boldsymbol{\delta} \in \mathcal{P}_i$, define

$$\mathcal{B}'_{i,\boldsymbol{\delta}} \overset{\text{def}}{=} \{c(x) : c(x) \in \mathcal{B}'_i, \boldsymbol{w}_n(c) = \boldsymbol{\delta}\}.$$

Then $\{\mathcal{B}'_{i,\boldsymbol{\delta}}\}_{\boldsymbol{\delta} \in \mathcal{P}_i}$ is a set partition of $\mathcal{B}'_i$. Hence, for all i,

$$B'_i = \sum_{\boldsymbol{\delta} \in \mathcal{P}_i} |\mathcal{B}'_{i,\boldsymbol{\delta}}|. \tag{7.7}$$

We are now ready to give the formula for B'_i.

Lemma 7.7 *Let $\phi_q(t)$, $t \geq 1$, be the number of solutions to $\sum_{j=1}^{t} x_j = 0$, such that $x_j \in \mathbb{F}_q$, $x_j \neq 0$, $\forall j$. Then*

$$\phi_q(t) = \frac{q-1}{q}\big((q-1)^{t-1} - (-1)^{t-1}\big). \tag{7.8}$$

Proof: See Appendix Appendix 7.A. ■

Proposition 7.8 *For all i,*

$$B'_i = \sum_{\boldsymbol{\delta} \in \mathcal{P}_i} \sum_{j=0}^{|\boldsymbol{\delta}|} \binom{b}{j} \binom{n-b}{|\boldsymbol{\delta}|-j} \prod_{l=1}^{|\boldsymbol{\delta}|} \binom{a + 1_{\{l \leq j\}}}{\delta_l} \phi_q(\delta_l), \tag{7.9}$$

where

$$1_{\{l \leq j\}} = \begin{cases} 1 & \textit{if } l \leq j \\ 0 & \textit{otherwise,} \end{cases}$$

$\phi_q(t)$ is as given in (7.8), and a and b are integers such that $n' = an + b$, $0 \leq b < n$.

Proof: Note that for all $c(x) = \sum_{j=0}^{n'-1} c_j x^j \in \mathbb{F}_q[x]$,

$$R_{x^n-1}\big[c(x)\big] = \sum_{l \in \mathcal{L}_n(c)} \Big(\sum_{j \in \mathcal{W}_{n,l}(c)} c_j \Big) x^l.$$

Therefore, codewords in $\mathcal{B}'_{i,\boldsymbol{\delta}}$ can be enumerated with the following process.

1. Choose $\mathcal{L}_n(c) \subseteq \{0, 1, \ldots, n-1\}$ such that $|\mathcal{L}_n(c)| = |\boldsymbol{\delta}|$.

2. For each $l \in \mathcal{L}_n(c)$, choose $\mathcal{W}_{n,l}(c) \subseteq \{0, 1, \ldots, n'-1\} \cap (l + n\mathbb{Z})$ such that $|\mathcal{W}_{n,l}(c)| = \delta_l$.

3. For each $\mathcal{W}_{n,l}(c)$, choose $c_j \in \mathbb{F}_q \setminus \{0\}$ for all $j \in \mathcal{W}_{n,l}(c)$, such that $\sum_{j \in \mathcal{W}_{n,l}(c)} c_j = 0$.

In step 2, δ_l numbers are chosen from $\{0, \ldots, a\}$ if $l \leq b$, and from $\{0, \ldots, a-1\}$ otherwise. In step 1, there are $\binom{b}{j}\binom{n-b}{|\boldsymbol{\delta}|-j}$ choices such that $\mathcal{L}_n(c)$ contains exactly j numbers that are no greater than b. For each such choice, there are $\prod_{l=1}^{j} \binom{a+1}{\delta_l} \prod_{l=j+1}^{|\boldsymbol{\delta}|} \binom{a}{\delta_l}$ choices in step 2, for each of which there are $\phi_q(\delta_l)$ choices in step 3. Summing over all possible values of j, and noting (7.7), we immediately obtain (7.9). ■

Corollary 7.9 *If $n \mid n'$, then for all i,*

$$B'_i = \sum_{\boldsymbol{\delta} \in \mathcal{P}_i} \binom{n}{|\boldsymbol{\delta}|} \prod_{l=1}^{|\boldsymbol{\delta}|} \binom{n'/n}{\delta_l} \phi_q(\delta_l). \tag{7.10}$$

Remark Note that in (7.7), and consequently (7.9) and (7.10), we have summed over all partitions of i. However, not all partitions of i are valid n-ary support profiles for codewords in $\mathcal{B}'_i$. For example, if $\boldsymbol{\delta} = \boldsymbol{w}_n(c)$ for some $c(x) \in \mathcal{B}'_i$, then by definition of the n-ary support profile, it must be true that $|\boldsymbol{\delta}| \leq n$, and $\delta_l \leq \lceil n'/n \rceil$ for all l. Further, since $x^n - 1 \mid c(x)$, we have $\delta_l \neq 1$ for all l. Therefore, it suffices to consider

$$\mathcal{P}_i(n', n) \overset{\text{def}}{=} \{\boldsymbol{\delta} \in \mathcal{P}_i : |\boldsymbol{\delta}| \leq n, 2 \leq \delta_l \leq \lceil n'/n \rceil, \forall l\}. \tag{7.11}$$

In all our formulas for calculation of B'_i, $\mathcal{P}_i$ can be replaced by $\mathcal{P}_i(n', n)$.

We now show that $A'_i = B'_i$ for all $i < d$.

Proposition 7.10 *For all* $i < d$*,* $A'_i = B'_i$*.*

Proof: We show that $\mathcal{A}'_i = \mathcal{B}'_i$. By Lemma 7.4, $\mathcal{B}'_i \subseteq \mathcal{A}'_i$. To show $\mathcal{A}'_i \subseteq \mathcal{B}'_i$, note that if $c(x) \in \mathcal{A}'_i$, then $R_{x^n-1}[c(x)] \in \mathcal{C}$. On the other hand, $\text{wt}\big(R_{x^n-1}[c(x)]\big) \leq \text{wt}\big(c(x)\big) = i < d$, which implies that $R_{x^n-1}[c(x)] = 0$. ■

For $i \geq d$, we find bounds on A'_i.

Proposition 7.11

$$A'_i \leq \begin{cases} \binom{n'}{i}(q-1)^{i-d+1} + B'_i & \textit{if } d \leq i \leq \lceil n'/n \rceil (d-2) \\ \binom{n'}{i}(q-1)^{i-d+1} & \textit{if } i > \lceil n'/n \rceil (d-2). \end{cases} \tag{7.12}$$

Proof: Let $c(x)$ be a polynomial of weight i, denoted by $c(x) = \sum_{j=1}^{i} c_{k_j} x^{k_j}$, $c_{k_j} \neq 0$, $\forall j$. Recall that the generator polynomial of $\mathcal{C}$ has the form $g(x) = \prod_{l=0}^{d-2}(x - \omega^{s+l})$, where ω is a primitive n-th root of unity in $\mathbb{F}_q$, and s is an integer. Hence, $c(x) \in \mathcal{A}'_i$ if and only if $c(\omega^{s+l}) = 0$, for all $0 \leq l \leq d-2$. This condition can be written as

$$\begin{pmatrix} \tilde{c}_{k_1} & \tilde{c}_{k_2} & \ldots & \tilde{c}_{k_i} \end{pmatrix} \begin{pmatrix} 1 & \omega^{k_1} & \cdots & \omega^{(d-2)k_1} \\ 1 & \omega^{k_2} & \cdots & \omega^{(d-2)k_2} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \omega^{k_i} & \cdots & \omega^{(d-2)k_i} \end{pmatrix} = \mathbf{0}, \tag{7.13}$$

where $\tilde{c}_{k_j} = c_{k_j}\omega^{sk_j}$ for all j. Note that counting $\begin{pmatrix} c_{k_1} & c_{k_2} & \ldots & c_{k_i} \end{pmatrix}$ is equivalent to counting $\begin{pmatrix} \tilde{c}_{k_1} & \tilde{c}_{k_2} & \ldots & \tilde{c}_{k_i} \end{pmatrix}$.

Recall that $i = \sum_{l\in\mathcal{L}_n(c)} |\mathcal{W}_{n,l}(c)| \leq |\mathcal{L}_n(c)|\lceil n'/n\rceil$. If $i > \lceil n'/n\rceil(d-2)$, then $|\mathcal{L}_n(c)| \geq d-1$, which implies that $\{\omega^{k_j}\}_{j=1}^{i}$ contains at least $d-1$ distinct values. Without loss of generality, assume $\omega^{k_1},\dots,\omega^{k_{d-1}}$ are distinct. Rewrite (7.13) as

$$\begin{pmatrix}\tilde{c}_{k_1} & \dots & \tilde{c}_{k_{d-1}}\end{pmatrix}\begin{pmatrix}1 & \omega^{k_1} & \cdots & \omega^{(d-2)k_1}\\ \vdots & \vdots & \cdots & \vdots\\ 1 & \omega^{k_{d-1}} & \cdots & \omega^{(d-2)k_{d-1}}\end{pmatrix} = \\ -\begin{pmatrix}\tilde{c}_{k_d} & \dots & \tilde{c}_{k_i}\end{pmatrix}\begin{pmatrix}1 & \omega^{k_d} & \cdots & \omega^{(d-2)k_d}\\ \vdots & \vdots & \cdots & \vdots\\ 1 & \omega^{k_i} & \cdots & \omega^{(d-2)k_i}\end{pmatrix}. \quad (7.14)$$

The matrix on the left is a Vandermonde matrix and hence is invertible. For all choices of $(\tilde{c}_{k_j})_{j=d}^{i}$ that are nonzero, $(\tilde{c}_{k_j})_{j=1}^{d-1}$ is uniquely determined. By enumerating all nonzero $(\tilde{c}_{k_j})_{j=d}^{i}$, we can enumerate all valid choices of $(\tilde{c}_{k_j})_{j=1}^{i}$ that satisfy (7.13), possibly more (since $(\tilde{c}_{k_j})_{j=1}^{d-1}$ so determined may contain zeros). Therefore, for each given choice of $\{k_j\}_{j=1}^{i}$, there are at most $(q-1)^{i-d+1}$ codewords of weight i. The total number of weight-i codewords is hence at most $\binom{n'}{i}(q-1)^{i-d+1}$.

On the other hand, if $d \leq i \leq \lceil n'/n\rceil(d-2)$, we break A'_i into two parts:

$$A'_i = B'_i + |\mathcal{A}'_i \setminus \mathcal{B}'_i|.$$

By Lemma 7.4, any codeword $c(x)$ in $\mathcal{A}'_i \setminus \mathcal{B}'_i$ must satisfy $0 \neq R_{x^n-1}[c(x)] \in \mathcal{C}$. Therefore, $\mathrm{wt}\big(R_{x^n-1}[c(x)]\big) \geq d$, which implies that $\{\omega^{k_j}\}_{j=1}^{i}$ contains at least d distinct values. The reasoning for the first case now applies and we see that $|\mathcal{A}'_i \setminus \mathcal{B}'_i|$ is upper bounded by $\binom{n'}{i}(q-1)^{i-d+1}$. ■

Proposition 7.12

$$A'_i \geq \begin{cases} K'_i(q-d)(q-1)^{i-d} + B'_i & \textit{if } d \leq i \leq \lceil n'/n\rceil(d-1)\\ \binom{n'}{i}(q-d)(q-1)^{i-d} & \textit{if } i > \lceil n'/n\rceil(d-1),\end{cases} \quad (7.15)$$

where

$$K'_i = \sum_{j=0}^{i}\binom{b}{j}\binom{n-b}{i-j}(a+1)^j a^{i-j}, \quad (7.16)$$

and a and b are such that $n' = an + b$, $0 \leq b < n$.

Proof: Let $c(x)$ be a polynomial of weight i, denoted by $c(x) = \sum_{j=1}^{i} c_{k_j} x^{k_j}$, $c_{k_j} \neq 0$, $\forall j$. If $i > \lceil n'/n \rceil (d-1)$, then $|\mathcal{L}_n(c)| \geq d$, which implies that $\{\omega^{k_j}\}_{j=1}^{i}$ contains at least d distinct values. Without loss of generality, assume that $\omega^{k_1}, \ldots, \omega^{k_d}$ are distinct. Following the notation used in the proof of Proposition 7.11, we rewrite (7.14) as

$$\begin{pmatrix} \tilde{c}_{k_1} & \ldots & \tilde{c}_{k_{d-1}} \end{pmatrix} \begin{pmatrix} 1 & \omega^{k_1} & \cdots & \omega^{(d-2)k_1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \omega^{k_{d-1}} & \cdots & \omega^{(d-2)k_{d-1}} \end{pmatrix} = \\ - \begin{pmatrix} \tilde{c}_{k_{d+1}} & \ldots & \tilde{c}_{k_i} \end{pmatrix} \begin{pmatrix} 1 & \omega^{k_{d+1}} & \cdots & \omega^{(d-2)k_{d+1}} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \omega^{k_i} & \cdots & \omega^{(d-2)k_i} \end{pmatrix} - \tilde{c}_{k_d} \begin{pmatrix} 1 & \omega^{k_d} & \cdots & \omega^{(d-2)k_d} \end{pmatrix} \tag{7.17}$$

Call the matrix on the left V and the one on the right W. Note that V is invertible, so we can write

$$\begin{aligned} & \begin{pmatrix} \tilde{c}_{k_1} & \ldots & \tilde{c}_{k_{d-1}} \end{pmatrix} \\ = \; & -\begin{pmatrix} \tilde{c}_{k_{d+1}} & \ldots & \tilde{c}_{k_i} \end{pmatrix} W V^{-1} - \tilde{c}_{k_d} \begin{pmatrix} 1 & \omega^{k_d} & \cdots & \omega^{(d-2)k_d} \end{pmatrix} V^{-1} \\ = \; & \boldsymbol{r} + \tilde{c}_{k_d} \boldsymbol{v}, \end{aligned}$$

where $\boldsymbol{r} = -\begin{pmatrix} \tilde{c}_{k_{d+1}} & \ldots & \tilde{c}_{k_i} \end{pmatrix} W V^{-1}$ and $\boldsymbol{v} = -\begin{pmatrix} 1 & \omega^{k_d} & \cdots & \omega^{(d-2)k_d} \end{pmatrix} V^{-1}$.

We now show that for all choices of $(\tilde{c}_{k_j})_{j=d+1}^{i}$ that are nonzero, no matter what $\boldsymbol{r}$ comes out to be, we always have at least $q-d$ choices of $\tilde{c}_{k_d} \neq 0$ to make $c(x)$ a weight-i codeword, i.e., at least $q-d$ choices of $\tilde{c}_{k_d}$ such that the values $\{\tilde{c}_{k_j}\}_{j=1}^{d-1}$ determined from (7.17) are all nonzero.

First, we claim that $\boldsymbol{v}$ does not contain zero elements. Suppose otherwise, that for some $1 \leq j \leq d-1$, $v_j = 0$. By definition,

$$\boldsymbol{v} V = -\begin{pmatrix} 1 & \omega^{k_d} & \cdots & \omega^{(d-2)k_d} \end{pmatrix} \tag{7.18}$$

Since $v_j = 0$, we can ignore the j-th row in V and rearrange (7.18) as

$$\begin{pmatrix} v_1 & \cdots & v_{j-1} & v_{j+1} & \cdots & v_{d-1} & 1 \end{pmatrix} \begin{pmatrix} 1 & \omega^{k_1} & \cdots & \omega^{(d-2)k_1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \omega^{k_{j-1}} & \cdots & \omega^{(d-2)k_{j-1}} \\ 1 & \omega^{k_{j+1}} & \cdots & \omega^{(d-2)k_{j+1}} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \omega^{k_d} & \cdots & \omega^{(d-2)k_d} \end{pmatrix} = \mathbf{0}$$

Note that since $\{\omega^{k_j}\}_{j=1}^{d}$ are all distinct, the matrix on the left is invertible, which implies that $\begin{pmatrix} v_1 & \cdots & v_{j-1} & v_{j+1} & \cdots & v_{d-1} & 1 \end{pmatrix} = \mathbf{0}$, a contradiction.

Now, since $v_j \neq 0$ for all $1 \leq j \leq d-1$, for any given j, there is at most one nonzero value that $\tilde{c}_{k_d}$ can take such that $r_j + \tilde{c}_{k_d} v_j = 0$. Therefore, there are at least $(q-1)-(d-1) = q-d$ nonzero values that $\tilde{c}_{k_d}$ can take such that $\tilde{c}_{k_j} = r_j + \tilde{c}_{k_d} v_j \neq 0$ for all j. Thus, for any given $\{k_j\}_{j=1}^{i}$, there are at least $(q-1)^{i-d}(q-d)$ codewords of weight i. So the total number of weight-i codewords is at least $\binom{n'}{i}(q-1)^{i-d}(q-d)$.

If $d \leq i \leq \lceil n'/n \rceil (d-1)$, we break A'_i into two parts:

$$A'_i = B'_i + |\mathcal{A}'_i \setminus \mathcal{B}'_i|.$$

Consider the subset of codewords in $\mathcal{A}'_i$ whose n-ary support profile is the all-ones vector, i.e., $\boldsymbol{w}_n(c) = (1, \dots, 1)$. All these codewords must be contained in $\mathcal{A}'_i \setminus \mathcal{B}'_i$, as codewords in $\mathcal{B}'_i$ have n-ary support profiles whose elements are no less than 2. There are $K'_i = \sum_{j=0}^{i} \binom{b}{j}\binom{n-b}{i-j}(a+1)^j a^{i-j}$ choices of $\{k_j\}_{j=1}^{i}$ (i.e. $\mathcal{W}(c)$) corresponding to the all-ones support profile and they all satisfy $|\mathcal{L}_n(c)| = i \geq d$. Therefore, the reasoning of the first case applies and we see that the number of codewords in $\mathcal{A}'_i \setminus \mathcal{B}'_i$ is at least $K'_i(q-d)(q-1)^{i-d}$. ■

7.3 Probability of Undetected Error

First, note that any term in (7.1) is a lower bound on $P_{ud}(p)$. In particular, for a code with length n and minimum distance d, we have

$$P_{ud}(p) \geq A_d \left(\frac{p}{q-1}\right)^d (1-p)^{n-d}.$$

This bound is interesting because for any given code, it is the dominant term in the sum as $p \to 0$. From Proposition 7.1, we immediately obtain the following result.

Proposition 7.13 *If $d > 2$, then*

$$P_{ud}(p) \geq P_{ud}^{(2)}(p),$$

where

$$P_{ud}^{(2)}(p) = \left(\binom{a}{2} + \frac{ab}{q-1}\right) p^2 (1-p)^{n'-2}, \tag{7.19}$$

and a and b are integers such that $n' = a(q-1) + b$, $0 \leq b < q-1$.

Note that

$$\max_{0 \leq p \leq 1} P_{ud}^{(2)}(p) = \left(\binom{a}{2} + \frac{ab}{q-1}\right) \left(\frac{2}{n'}\right)^2 \left(1 - \frac{2}{n'}\right)^{n'-2},$$

which we denote by $P_{\max}^{(2)}$. In many cases, we have $P_{\max}^{(2)} > q^{-(d-1)}$, which implies that the corresponding OERS codes are not good. For example, if n'/n is fixed, then as $q \to \infty$, $P_{\max}^{(2)} \sim Cq^{-2}$, where C is a constant that depends only on n'/n. Therefore, for all $d > 3$ and $q > 1/C$, the corresponding OERS codes are not good. The intuition here is that the number of weight-2 codewords in an OERS code does *not* depend on the number of parity-check symbols ($d-1$ in this case). While P_{ud}^{avg} is expected to decrease exponentially with d, $P_{\max}^{(2)}$ is not affected. For practical values of q, $P_{\max}^{(2)}$ can be orders of magnitude larger than $q^{-(d-1)}$, even when d is just moderately larger than 3.

Next, better bounds can be obtained by simply plugging the results of Proposition 7.10, Proposition 7.12, and Proposition 7.11 into (7.1).

Proposition 7.14 *For OERS codes,*

$$\underline{P}_{ud}(p) \leq P_{ud}(p) \leq \overline{P}_{ud}(p),$$

where

$$\begin{aligned}
\underline{P}_{ud}(p) &= \sum_{i=1}^{\lceil \frac{n'}{n} \rceil (d-1)} B'_i \left(\frac{p}{q-1}\right)^i (1-p)^{n'-i} \\
&\quad + \frac{q-d}{(q-1)^d} \left(1 - \sum_{i=0}^{\lceil \frac{n'}{n} \rceil (d-1)} \binom{n'}{i} p^i (1-p)^{n'-i}\right) \\
&\quad + \frac{q-d}{(q-1)^d} \sum_{i=d}^{\lceil \frac{n'}{n} \rceil (d-1)} K'_i p^i (1-p)^{n'-i},
\end{aligned} \tag{7.20}$$

$$\begin{aligned}\overline{P}_{ud}(p) &= \sum_{i=1}^{\lceil \frac{n'}{n} \rceil (d-2)} B'_i \left(\frac{p}{q-1}\right)^i (1-p)^{n'-i} \\ &\quad + \frac{1}{(q-1)^{d-1}} \left(1 - \sum_{i=0}^{d-1} \binom{n'}{i} p^i (1-p)^{n'-i}\right), \end{aligned} \tag{7.21}$$

where B'_i is given by (7.9), and K'_i by (7.16).

The the worst-case probability of undetected error, $P_{ud}^{\max} \overset{\text{def}}{=} \max_{0 \le p \le 1} P_{ud}(p)$, is then bounded between the maximum values of the upper and lower bounds.

Corollary 7.15 *For OERS codes,*

$$\underline{P}_{ud}^{\max} \le P_{ud}^{\max} \le \overline{P}_{ud}^{\max},$$

where $\underline{P}_{ud}^{\max} \overset{\text{def}}{=} \max_{0 \le p \le 1} \underline{P}_{ud}(p)$, $\overline{P}_{ud}^{\max} \overset{\text{def}}{=} \max_{0 \le p \le 1} \overline{P}_{ud}(p)$.

We now discuss the tightness of the bounds that we have derived. First, we show that the bounds given by Proposition 7.14 are asymptotically tight for all p as $q \to \infty$.

Lemma 7.16 *Let n'/n be fixed. For any fixed i, as $q \to \infty$,*

$$K'_i \sim \binom{n'}{i}, \tag{7.22}$$

where K'_i is as defined in (7.16).

Proof: See Appendix Appendix 7.A. ■

Proposition 7.17 *Let n'/n and d be fixed. Then for all $0 < p \le 1$, as $q \to \infty$,*

$$\underline{P}_{ud}(p) \sim P_{ud}(p) \sim \overline{P}_{ud}(p). \tag{7.23}$$

Proof: It suffices to show that $\overline{P}_{ud}(p) - \underline{P}_{ud}(p) = o\big(\overline{P}_{ud}(p)\big)$.

First, note that $\overline{P}_{ud}(p)$ can be rewritten as

$$\overline{P}_{ud}(p) = \sum_{i=1}^{\lceil \frac{n'}{n} \rceil (d-2)} B'_i \left(\frac{p}{q-1}\right)^i (1-p)^{n'-i} + P_1(p) + P_2(p),$$

where

$$P_1(p) = \frac{1}{(q-1)^{d-1}}\left(1 - \sum_{i=0}^{\lceil \frac{n'}{n}\rceil(d-1)} \binom{n'}{i} p^i (1-p)^{n'-i}\right),$$

$$P_2(p) = \frac{1}{(q-1)^{d-1}} \sum_{i=d}^{\lceil \frac{n'}{n}\rceil(d-1)} \binom{n'}{i} p^i (1-p)^{n'-i}.$$

Next, from the expressions above and (7.20), it is easy to show that

$$\overline{P}_{ud}(p) - \underline{P}_{ud}(p) \leq \Delta_1(p) + \Delta_2(p),$$

where

$$\Delta_1(p) = \frac{d-1}{(q-1)^d}\left(1 - \sum_{i=0}^{\lceil \frac{n'}{n}\rceil(d-1)} \binom{n'}{i} p^i (1-p)^{n'-i}\right),$$

$$\Delta_2(p) = \frac{1}{(q-1)^{d-1}} \sum_{i=d}^{\lceil \frac{n'}{n}\rceil(d-1)} \left(\binom{n'}{i} - \frac{q-d}{q-1}K'_i\right) p^i (1-p)^{n'-i}.$$

Finally, note that $\Delta_1(p) = o\big(P_1(p)\big)$. And by Lemma 7.16, $\Delta_2(p) = o\big(P_2(p)\big)$. Therefore, $\overline{P}_{ud}(p) - \underline{P}_{ud}(p) = o\big(\overline{P}_{ud}(p)\big)$. ■

Since the result of Proposition 7.17 holds for all $p \in (0, 1]$, it follows that $\underline{P}_{ud}^{\max}$ and $\overline{P}_{ud}^{\max}$ are also tight bounds for $P_{ud}^{\max}$.

Corollary 7.18 *Let n'/n and d be fixed. Then for all $0 < p \leq 1$, as $q \to \infty$,*

$$\underline{P}_{ud}^{\max} \sim P_{ud}^{\max} \sim \overline{P}_{ud}^{\max}.$$

We now show that in many cases, $P_{ud}^{\max}$ consists predominantly of the contribution from weight-2 codewords.

Lemma 7.19 *Let n'/n be fixed. For any fixed i, as $q \to \infty$,*

$$B'_i = O(q^i). \tag{7.24}$$

Proposition 7.20 *Let n'/n and d, $d > 3$, be fixed. As $q \to \infty$,*

$$P_{\max}^{(2)} \sim P_{ud}^{\max}.$$

Proof: Since $P^{(2)}_{\max} \leq P^{\max}_{ud} \leq \overline{P}^{\max}_{ud}$, it suffices to show that $P^{(2)}_{\max} \sim \overline{P}^{\max}_{ud}$. Note

$$\begin{aligned}
&\overline{P}^{\max}_{ud} - P^{(2)}_{\max} \\
&= \max_p \overline{P}_{ud}(p) - \max_p P^{(2)}_{ud}(p) \\
&\leq \max_p \{\overline{P}_{ud}(p) - P^{(2)}_{ud}(p)\} \\
&= \max_p \left\{ \sum_{i=3}^{\lceil \frac{n'}{n} \rceil (d-2)} B'_i \Big(\frac{p}{q-1}\Big)^i (1-p)^{n'-i} + \frac{1}{(q-1)^{d-1}} \left(1 - \sum_{i=0}^{d-1} \binom{n'}{i} p^i (1-p)^{n'-i}\right)\right\} \\
&\leq \sum_{i=3}^{\lceil \frac{n'}{n} \rceil (d-2)} \max_p \left\{ B'_i \Big(\frac{p}{q-1}\Big)^i (1-p)^{n'-i} \right\} + \frac{1}{(q-1)^{d-1}} \\
&= \sum_{i=3}^{\lceil \frac{n'}{n} \rceil (d-2)} \frac{B'_i}{(q-1)^i} \left(\frac{i}{q-1}\right)^i \left(1 - \frac{i}{q-1}\right)^{n'-i} + \frac{1}{(q-1)^{d-1}} \\
&= \sum_{i=3}^{\lceil \frac{n'}{n} \rceil (d-2)} O(q^{-i}) + O\big(q^{-(d-1)}\big) \\
&= O(q^{-3}).
\end{aligned}$$

Recall that in the discussion following Proposition 7.13, we have shown that for this case $P^{(2)}_{\max} = \Theta(q^{-2})$. Therefore, $\overline{P}^{\max}_{ud} - P^{(2)}_{\max} = o\big(P^{(2)}_{\max}\big)$, implying $P^{(2)}_{\max} \sim P^{\max}_{ud}$. ■

We should note that since RS codes are usually used with relatively short block lengths, the asymptotic analysis might not seem very useful. However, it is clear from the proofs that as long as $n' >> \lceil \frac{n'}{n} \rceil (d-1)$, i.e. $n >> d-1$, the actual behavior of the code should be well approximated by the asymptotic analysis. In applications such as data storage, where high rate codes are commonly used, this condition is usually satisfied.

The asymptotic results can also be used to simplify the bounds. For example, from Lemma 7.16 and Lemma 7.19, we see that in the lower bound of (7.15), B'_i is negligibly small and can be safely ignored. This would reduce the number of B'_i's that need be calculated in the lower bound of (7.20).

We end our discussion with an example. Let $\mathcal{C}'$ be an OERS code obtained by doubling the code length of $\mathcal{C}$, a RS code over $\mathbb{F}_{2^m}$ with minimum distance $d = 4$. Fig. 7.1 plots $\underline{P}_{ud}(p)$ and $\overline{P}_{ud}(p)$ together with the true $P_{ud}(p)$ for various values of m. Fig. 7.2 shows more explicitly

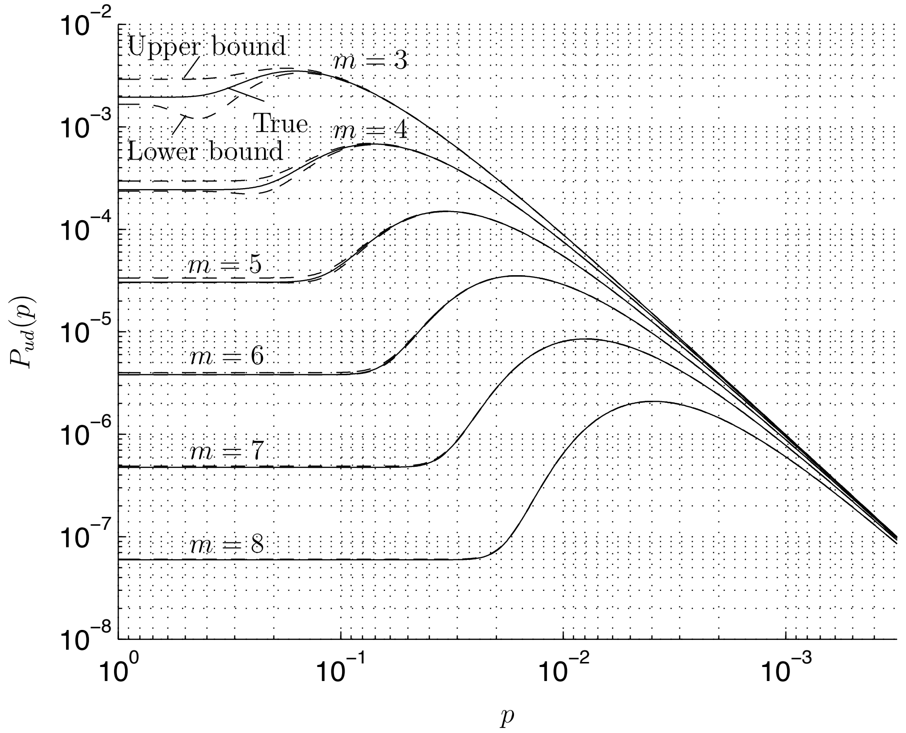

Figure 7.1 Upper and lower bounds on $P_{ud}(p)$ for $[2^{m+1}-2, 2^{m+1}-5]$ OERS codes over $\mathbb{F}_{2^m}$

how the upper and lower bounds converge to the true probability as m increases. The true $P_{ud}(p)$ values are obtained through equation (7.2), where the weight distribution of the dual code of $\mathcal{C}'$ is found through enumeration of codewords. It should be noted that this brute-force procedure is only possible when d is small (so that the dual code is of low dimension). Even for $d = 4$, as in our case, for $m > 6$, the enumeration becomes a rather long process.

From the figures, we see that our upper and lower bounds are very tight, except for very small values of m. As the field size increases, the bounds converge to the true probability of undetected error very quickly. This is especially true for the bounds on $P_{ud}^{\max}$. Even $P_{\max}^{(2)}$ converges rather fast. The numerical values of bounds on $P_{ud}^{\max}$ are shown in Table 7.1. Also, from Fig. 7.1, we note that when p is small, both bounds are tight regardless of the field size. This is expected because as $p \to 0$, both bounds consist of predominantly $P_{ud}^{(2)}(p)$, which is asymptotically p^2.

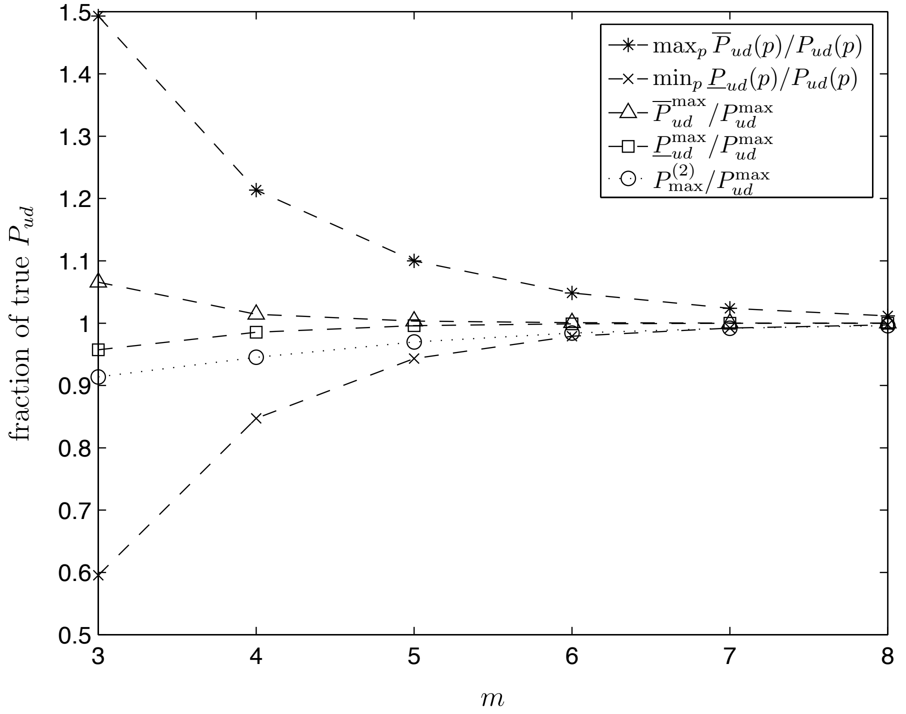

Figure 7.2 Convergence of P_{ud} bounds for $[2^{m+1}-2, 2^{m+1}-5]$ OERS codes over $\mathbb{F}_{2^m}$

Table 7.1 Bounds on $P_{ud}^{\max}$ for $[2^{m+1}-2, 2^{m+1}-5]$ OERS codes $\mathbb{F}_{2^m}$

	$P_{ud}^{\max}$	$\underline{P}_{ud}^{\max}$	$\overline{P}_{ud}^{\max}$	$P_{\max}^{(2)}$
$m=3$	3.5120e-03	3.3622e-03	3.7427e-03	3.2095e-03
$m=4$	6.8138e-04	6.7156e-04	6.9102e-04	6.4394e-04
$m=5$	1.5004e-04	1.4946e-04	1.5054e-04	1.4550e-04
$m=6$	3.5204e-05	3.5170e-05	3.5232e-05	3.4647e-05
$m=7$	8.5262e-06	8.5241e-06	8.5279e-06	8.4573e-06
$m=8$	2.0980e-06	2.0979e-06	2.0981e-06	2.0895e-06

7.4 Conclusion

OERS codes are of practical interest as they have been used in data storage systems. In this work, we have examined their performance in terms of probability of undetected error when the codes are used solely for error detection over a q-ary symmetric channel. We have obtained upper and lower bounds on $P_{ud}(p)$, which have been shown to be asymptotically tight. The bounds are also relatively easy to evaluate for high rate codes, which are commonly used in storage systems.

Our bounds on the probability of undetected error have been derived by bounding the weight distribution of the code. The techniques involved in obtaining the weight distribution bounds can potentially be applied to the study of other RS- or BCH-derived codes.

Appendix 7.A Additional Proofs

Proof (Proposition 7.1): If $d > 2$, we show that every $c(x) \in \mathcal{A}'_2$ is of the form $\alpha(x^i - x^j)$, where $i \equiv j \mod n$. Clearly, if $c(x) = \alpha(x^i - x^j)$ and $i \equiv j \mod n$, then $g(x) \mid x^n - 1 \mid a(x)$; hence $c(x) \in \mathcal{A}'_2$. On the other hand, if $c(x) = \alpha x^i + \beta x^j \in \mathcal{A}'_2$, then $g(x) \mid c(x)$. For $d > 2$, $(x - \omega^s)(x - \omega^{s+1}) \mid g(x)$ for some primitive n-th root of unity $\omega \in \mathbb{F}_q$ and some integer s. Hence, $c(\omega^s) = c(\omega^{s+1}) = 0$, from which it is easy to show that $\omega^i = \omega^j$ (hence $i \equiv j \mod n$) and $\alpha + \beta = 0$. Now, note that $\mathcal{A}'_2$ can be written as the following union of disjoint subsets:

$$\mathcal{A}'_2 = \bigcup_{j=1}^{a} \Big\{ \alpha(x^{jn+i} - x^i) : \alpha \in \mathbb{F}_q \backslash \{0\}, i = 0, 1, \ldots, \max\{n' - jn - 1, b - 1\} \Big\}.$$

The result follows from simple counting.

If $d = 2$, then $g(x) = x - \omega^s$. A weight-2 polynomial $c(x) = \alpha x^i + \beta x^j \in \mathbb{F}_q[x]$ is a codeword if and only if $c(\omega^s) = 0$. That is, if and only if $\beta = -\alpha\omega^{s(i-j)}$. So the number of weight-2 codewords is simply the number of $\{i, j\}$ pairs times the number of nonzero choices of α.

If $d = 1$, any polynomial in $\mathbb{F}_q[x]$ that has degree less than n' is a codeword. Hence, $A'_2 = \binom{n'}{2}(q-1)^2$. ■

Proof (Proposition 7.3): For $d > 3$, the proof is similar in spirit to the first part of the proof of Proposition 7.1, and the result is a specialization of Proposition 7.8 and Proposition 7.10.

Details of the proof are thus omitted. Essentially, we show that every $c(x) \in \mathcal{A}'_3$ has the form $\alpha x^i + \beta x^j + \gamma x^l$, such that $\alpha + \beta + \gamma = 0$ and $i \equiv j \equiv l \mod n$.

If $d = 3$, then $g(x) = (x - \omega^s)(x - \omega^{s+1})$ for some primitive n-th root of unity $\omega \in \mathbb{F}_q$ and some integer s. A weight-3 polynomial $c(x) = \alpha x^i + \beta x^j + \gamma x^l \in \mathbb{F}_q[x]$ is a codeword if and only if $c(\omega^s) = c(\omega^{s+1}) = 0$, i.e.

$$\begin{pmatrix} \alpha & \beta & \gamma \end{pmatrix} \begin{pmatrix} \omega^{si} & \omega^{(s+1)i} \\ \omega^{sj} & \omega^{(s+1)j} \\ \omega^{sl} & \omega^{(s+1)l} \end{pmatrix} = \mathbf{0} \tag{7.25}$$

If $i \equiv j \equiv l \mod n$, the equation above is satisfied if and only if $\alpha + \beta + \gamma = 0$. This corresponds to $\left(\binom{a}{3}(q-1) + \binom{a}{2}b\right)(q-1)(q-2)$ weight-3 codewords as has been calculated for $d > 3$. Otherwise, without loss of generality, suppose $i \not\equiv j \mod n$ so that $\omega^i \neq \omega^j$. From the fact that $\alpha, \beta, \gamma \neq 0$, it is easy to show that it must also be true that $j \not\equiv l \mod n$ and $l \not\equiv i \mod n$. This implies that if $\{i, j, l\}$ is chosen, we can fix any non-zero value for γ and (α, β) will be uniquely determined. From the inclusion-exclusion principle, we see that the number of $\{i, j, l\}$'s such that no two of the numbers are congruent modulo n is $\binom{n'}{3} - \binom{n'-2}{1}\left[\binom{a}{2}(q-1) + ab\right] + 2\left[\binom{a}{3}(q-1) + \binom{a}{2}b\right]$. The result now follows after some algebra.

If $d = 2$, then $g(x) = x - \omega^s$. We have $c(x) = \alpha x^i + \beta x^j + \gamma x^l \in \mathbb{F}_q[x]$ is a codeword if and only if $c(\omega^s) = 0$, which is true if and only if $\gamma = -\alpha\omega^{s(i-l)} - \beta\omega^{s(j-l)}$. There are $\binom{n'}{3}$ choices of $\{i, j, l\}$, for each of which there are $(q-1)$ choices of $\alpha \neq 0$ and subsequently $(q-2)$ choices of $\beta \neq 0$ and $\beta \neq -\alpha\omega^{s(i-j)}$. Therefore, $A'_3 = \binom{n'}{3}(q-1)(q-2)$.

If $d = 1$, any polynomial in $\mathbb{F}_q[x]$ that has degree less than n' is a codeword. Hence, $A'_3 = \binom{n'}{3}(q-1)^3$. ■

Proof (Lemma 7.7): Clearly, $\phi_q(1) = 0$. For $t \geq 2$, note that to satisfy the equation we must have $x_t = -\sum_{j=1}^{t-1} x_j$. So x_t is determined if $(x_j)_{j=1}^{t-1}$ are chosen. Among the $(q-1)^{t-1}$ choices of $(x_j)_{j=1}^{t-1}$ that are nonzero, a choice is a valid solution if and only if $\sum_{j=1}^{t-1} x_j \neq 0$. By definition the number of such choices is $\phi_q(t)$. On the other hand, also by definition, the number of choices corresponding to $\sum_{j=1}^{t-1} x_j = 0$ is $\phi_q(t-1)$. Therefore,

$$\phi_q(t) + \phi_q(t-1) = (q-1)^{t-1}, \quad \forall t \geq 2. \tag{7.26}$$

Consider the sequence of equations

$$\begin{aligned}
\phi_q(2) &= q-1, \\
\phi_q(3)+\phi_q(2) &= (q-1)^2 \\
&\vdots \\
\phi_q(t)+\phi_q(t-1) &= (q-1)^{t-1}
\end{aligned}$$

Multiplying the equation corresponding to $(q-1)^{t-1-j}$ by $(-1)^j$ and summing up, we obtain

$$\begin{aligned}
\phi_q(t) &= \sum_{j=0}^{t-2}(-1)^j(q-1)^{t-1-j} \\
&= (q-1)^{t-1}\sum_{j=0}^{t-2}\left(\frac{-1}{q-1}\right)^j \\
&= \frac{q-1}{q}\big((q-1)^{t-1}-(-1)^{t-1}\big), \quad \forall t \geq 2.
\end{aligned}$$

It is readily verified that the expression above also holds for $t=1$. ■

Proof (Lemma 7.16): Let $n' = an+b$, where a, b are integers such that $0 \leq b < n$. Note that for all fixed α and i, as $n \to \infty$,

$$\binom{n}{i}\alpha^i \sim \binom{\alpha n}{i}. \tag{7.27}$$

If $b=0$, we have

$$K_i' = \binom{n}{i}a^i \sim \binom{an}{i} = \binom{n'}{i}.$$

If $b \neq 0$, then $b, n-b \to \infty$ as $n \to \infty$. Hence, we have

$$\begin{aligned}
K_i' &= \sum_{j=0}^{i}\binom{b}{j}\binom{n-b}{i-j}(a+1)^j a^{i-j} \\
&\sim \sum_{j=0}^{i}\binom{(a+1)b}{j}\binom{a(n-b)}{i-j} \\
&= \sum_{j=0}^{i}\binom{(a+1)b}{j}\binom{n'-(a+1)b}{i-j} \\
&= \binom{n'}{i}.
\end{aligned}$$

■

Proof (of Lemma 7.19): We have

$$\begin{aligned}
B_i' &\leq \sum_{\boldsymbol{\delta}\in\mathcal{P}_i(n',n)} \binom{n}{|\boldsymbol{\delta}|} \prod_{l=1}^{|\boldsymbol{\delta}|} \binom{\lceil \frac{n'}{n} \rceil}{\delta_l} \phi_q(\delta_l) \\
&\sim \sum_{\boldsymbol{\delta}\in\mathcal{P}_i(n',n)} \frac{q^{|\boldsymbol{\delta}|}}{|\boldsymbol{\delta}|!} \prod_{l=1}^{|\boldsymbol{\delta}|} \binom{\lceil \frac{n'}{n} \rceil}{\delta_l} \prod_{l=1}^{|\boldsymbol{\delta}|} q^{\delta_l - 1} \\
&= \sum_{\boldsymbol{\delta}\in\mathcal{P}_i(n',n)} \frac{q^{|\boldsymbol{\delta}|}}{|\boldsymbol{\delta}|!} \left(\prod_{l=1}^{|\boldsymbol{\delta}|} \binom{\lceil \frac{n'}{n} \rceil}{\delta_l} \right) q^{i-|\boldsymbol{\delta}|} \\
&= q^i \sum_{\boldsymbol{\delta}\in\mathcal{P}_i(n',n)} \frac{1}{|\boldsymbol{\delta}|!} \prod_{l=1}^{|\boldsymbol{\delta}|} \binom{\lceil \frac{n'}{n} \rceil}{\delta_l} \\
&\leq q^i \sum_{\boldsymbol{\delta}\in\mathcal{P}_i} \frac{1}{|\boldsymbol{\delta}|!} \prod_{l=1}^{|\boldsymbol{\delta}|} \binom{\lceil \frac{n'}{n} \rceil}{\delta_l} \\
&= O(q^i)
\end{aligned}$$

■

Acknowledgment

This chapter is a reprint of the material in the paper: J. Han, P. H. Siegel and Patrick Lee, "On the probability of undetected error for over-extended Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3662–3669, Aug. 2006. The dissertation author was the primary author of this paper.

Bibliography

[1] J. K. Wolf and R. D. Blakeney, II, "An exact evaluation of the probability of undetected error for certain shortened binary CRC codes," in *Proc. MILCOM 88*, Oct. 1988, pp. 287–292.

[2] "Universal Serial Bus specification," Revision 2, Apr. 2000.

[3] "ATA 6 specification," Document Number T13/1410D, Revision 3B.

[4] F. J. M. Williams, "A theorem on the distribution of weights in a systematic code," *Bell Syst. Tech. J.*, vol. 42, pp. 79–94, Jan. 1963.

[5] S.-C. Chang and J. K. Wolf, “A simple derivation of the MacWilliams identity for linear codes,” *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 476–477, July 1980.

[6] S. K. Leung-Yan-Cheong and M. E. Hellman, “Concerning a bound on undetected error probability,” *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 235–237, Mar. 1976.

[7] S. K. Leung-Yan-Cheong, E. R. Barnes, and D. U. Friedman, “On some properties of the undetected error probability of linear codes,” *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 110–112, Jan. 1979.

[8] J. K. Wolf, A. M. Michelson, and A. H. Levesque, “On the probability of undetected error for linear block codes,” *IEEE Trans. Commun.*, vol. COM-30, pp. 317–324, Feb. 1982.

[9] T. Fujiwara, T. Kasami, A. Kitai, and S. Lin, “On the undetected error probability for shortened Hamming codes,” *IEEE Trans. Commun.*, vol. COM-33, pp. 570–574, June 1985.

[10] T. Kasami, T. Kløve, and S. Lin, “Linear block codes for error detection,” *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 131–136, Jan. 1983.

[11] T. Kasami and S. Lin, “On the probability of undetected error for the maximum distance separable codes,” *IEEE Trans. Commun.*, vol. COM-32, pp. 998–1006, Sept. 1984.

[12] T. Kløve and V. I. Korzhik, *Error Detecting Codes: General Theory and Their Application in Feedback Communication Systems*. Boston, MA: Kluwer, 1995.

[13] J. Massey, “Coding techniques for digital networks,” in *Proc. International Conference on Information Theory Systems*, Berlin, Germany, Sept. 1978, pp. 307–315.