Weak Flip Codes and Applications to Optimal Code Design on the Binary Erasure Channel

Intermediate Report of NSC Project

"Ultra-Short Blocklength Communication"

Date:	31 May 2013
Project-Number:	NSC 100–2221–E–009–068–MY3
Project Duration:	1 August 2011 – 31 July 2014
Funded by:	National Science Council, Taiwan

Author:	Stefan M. Moser
Co-Authors:	Po-Ning Chen, Hsuan-Yin Lin
Organization:	Information Theory Laboratory Department of Electrical and Computer Engineering National Chiao Tung University
Address:	Engineering Building IV, Office 727 1001 Daxue Rd. Hsinchu 30010, Taiwan
E-mail:	stefan.moser@ieee.org

Abstract

A new family of *nonlinear* codes, called *weak flip codes*, is presented and is shown to contain many beautiful properties. In particular, the subfamily of *fair weak flip codes* can be seen as a generalization of linear codes, i.e., they possess some *quasi-linear* properties. Different from linear codes that only exist for a number of codewords M being an integer-power of 2, the fair weak flip code can be defined for an arbitrary M. It is then noted that the fair weak flip codes are related to *binary nonlinear Hadamard codes*: both code families maximize the minimum Hamming distance and meet the Plotkin bound. However, while the binary nonlinear Hadamard codes have only been shown to possess good Hamming-distance properties, the fair weak flip codes are proven to be globally optimal — in the sense of minimizing the error probability, and under the assumption that the optimal codes can be constructed recursively in blocklength n — among all linear or nonlinear codes for the binary erasure channel (BEC) for many values of the blocklength n and for $\mathsf{M} \leq 6$. For $\mathsf{M} > 6$, similar optimality results are conjectured.

Moreover, some applications to known bounds on the error probability for a finite blocklength is introduced for comparison, while as blocklength n going to infinity, the error exponent of a BEC for a fixed number of codewords is also discussed.

The results in this work are founded upon a new powerful tool for the analysis and generation of block codes: the *column-wise* approach to the codebook matrix.

Keywords: Binary erasure channel (BEC), channel capacity, finite blocklength, flip codes, maximum likelihood (ML) decoder, minimum average error probability, optimal codes, weak flip codes.

Contents

1 Introduction

In traditional coding theory, it is the goal to find good codes that operate close to the ultimate limit of the *channel capacity* as introduced by Shannon [1]. Implicitly, by the definition of capacity, such codes have large blocklength. Moreover, due to the potential simplifications and because for large blocklength such codes do behave very well, conventional coding theory often restricts itself to *linear codes*. It is also quite common to use the *minimum Hamming distance* and the *weight enumerating function (WEF)* as a design and quality criterion [2]. This is motivated by the equivalence of Hamming weight and Hamming distance for linear codes, and by the union bound that converts the global error probability into pairwise error probabilities.

In this work we would like to break away from these traditional simplifications and instead focus on an optimal[1] design of codes for finite blocklength. Since for very short blocklength, it is not realistic to transmit large quantities of information, we

[1] With *optimal* we always mean *minimum error probability*.

start by looking at codes with only a few codewords, so called *ultra-small block-codes*. Such codes have many practical applications, e.g., in the situation of establishing an initial connection in a wireless link. There the amount of information that needs to be transmitted during the setup of the link is limited to usually only a couple of bits, however, these bits need to be transmitted in very short time (e.g., blocklength in the range of $n = 20$ to $n = 30$) with the highest possible reliability [3].

While conventional coding theory in the sense of Shannon often focuses on stating important fundamental insights and properties like, e.g., what rates are possible to achieve and what rates are not achievable, we specifically turn our attention to the concrete *code design*, i.e., we are interested in actually finding a globally optimum code for a certain given channel and a given fixed blocklength.

In this work, we introduce a new class of codes, called *fair weak flip codes*, that have many beautiful properties similar to those of linear codes. However, while linear codes are very much limited since they only can exist if the number of codewords M happens to be an integer-power of 2, our class of codes exists for arbitrary M. We will investigate these codes and show that they satisfy the Plotkin bound.

Fair weak flip codes are related to a class of binary nonlinear codes that are constructed with the help of Hadamard matrices and Levenshtein's theorem [4, Ch. 2]. These *binary nonlinear Hadamard codes* also meet the Plotkin bound. As a matter of fact, if for the parameters (M, n) of a given fair weak flip code there exists a Hadamard code, then these two codes are equivalent.[2] In this sense we can consider the fair weak flip codes to be a subclass of Hadamard codes. However, note that there is no guarantee that for every choice of parameters (M, n) for which fair weak flip codes exist, there also exists a corresponding Hadamard code.

Moreover, also note that while Levenshtein's method is only concerned with an optimal Hamming distance structure, we will show that fair weak flip codes are globally optimal for the *binary erasure channel (BEC)* under the assumption that the optimal codes can be constructed recursively in blocklength n. Hence, they are optimal with respect to error probability and not only pairwise Hamming distance, and they are best among *all* codes, linear or nonlinear. We prove this (conditional) optimality in the case of the number of codewords $\mathsf{M} \leq 6$ and conjecture it for $\mathsf{M} > 6$.

We also define a class of codes called *weak flip codes* that contains as special cases the class of fair weak flip codes, the class of binary nonlinear Hadamard codes, and the class of linear codes. We then specify some weak flip codes that are optimal for the BEC for $\mathsf{M} \leq 6$ and for *any* finite blocklength n, or for $\mathsf{M} = 5$ and for blocklength n satisfying $n \bmod 10 \in \{0, 3, 5, 7, 9\}$, or for $\mathsf{M} = 6$ and for even blocklength n.

This work is a continuation of our previous work [5], [6], where we have studied ultra-small block-codes for the situation of general binary-input binary-output channels and where we have derived the optimal code design for the two special cases of the *Z-channel (ZC)* and the *binary symmetric channel (BSC)*. We will also briefly compare our findings here with these previous results.

The foundation of our insights lies in a new very powerful way of creating and analyzing both linear and nonlinear block-codes. As is quite common, we use the *codebook matrix* containing the codewords in its rows to describe our codes. However, for our code construction and performance analysis, we look at this codebook matrix not row-wise, but *column-wise*. All our proofs and also our definition of the new "quasi-linear" codes are fully based on this new approach to a code. (This is another fundamental difference between our results and the binary nonlinear Hadamard

[2] For a precise definition of *equivalent* see Remark 11.

codes that are constructed based on Hadamard matrices and Levenshtein's theorem [4].)

The remainder of this report is structured as follows. After some comments about our notation, we will review some common definitions in Sections 2 and then introduce the channel model in Sections 3. In Section 4, we introduce some related topics in traditional information theory and coding theory. In Section 5 we introduce the new family of *weak flip codes*, that also contains the subfamily of *fair weak flip codes*. Some comparison examples between fair weak flip codes and Hadamard codes are also given. In Section 6, we review some important previous work to the error probability bounds. The main results are then summarized and discussed in Section 7.

As it is common in coding theory, vectors (denoted by bold face Roman letters, e.g., $\mathbf{x}$) are row-vectors. However, for simplicity of notation and to avoid a large number of transpose-signs we slightly misuse this notational convention for one special case: any vector $\mathbf{c}$ is a column-vector. It should be always clear from the context because these vectors are used to build codebook matrices and are therefore also conceptually quite different from the transmitted codewords $\mathbf{x}$ or the received sequence $\mathbf{y}$. Moreover, we use a bar $\bar{\mathbf{x}}$ to denote the flipped version of $\mathbf{x}$, i.e., $\bar{\mathbf{x}} \triangleq \mathbf{x} \oplus \mathbf{1}$ (where $\oplus$ denotes the componentwise XOR operation).

2 Definitions

2.1 Discrete Memoryless Channel

The probably most fundamental model describing communication over a noisy channel is the so-called *discrete memoryless channel (DMC)*. A DMC consists of a

- a finite input alphabet $\mathcal{X}$;
- a finite output alphabet $\mathcal{Y}$; and
- a conditional probability distribution $P_{Y|X}(\cdot|x)$ for all $x \in \mathcal{X}$ such that

$$P_{Y_k|X_1,X_2,\ldots,X_k,Y_1,Y_2,\ldots,Y_{k-1}}(y_k|x_1,x_2,\ldots,x_k,y_1,y_2,\ldots,y_{k-1}) = P_{Y|X}(y_k|x_k) \qquad \forall\, k. \quad (1)$$

Note that a DMC is called *memoryless* because the current output Y_k depends only on the current input x_k. Moreover also note that the channel is *time-invariant* in the sense that for a particular input x_k, the distribution of the output Y_k does not change over time.

Definition 1. We say a DMC is used *without feedback*, if

$$P(x_k|x_1,\ldots,x_{k-1},y_1,\ldots,y_{k-1}) = P(x_k|x_1,\ldots,x_{k-1}) \quad \forall\, k, \qquad (2)$$

i.e., X_k depends only on past inputs (by choice of the encoder), but not on past outputs. Hence, there is no feedback link from the receiver back to the transmitter that would inform the transmitter about the last outputs.

Note that even though we assume the channel to be memoryless, we do *not* restrict the encoder to be memoryless! We now have the following theorem.

Theorem 2. *If a DMC is used without feedback, then*

$$P(y_1,\ldots,y_n|x_1,\ldots,x_n) = \prod_{k=1}^{n} P_{Y|X}(y_k|x_k) \quad \forall\, n \geq 1. \tag{3}$$

Proof: See, e.g., [7]. □

2.2 Coding for DMC

Definition 3. A (M, n) *coding scheme* for a DMC $(\mathcal{X}, \mathcal{Y}, P_{Y|X})$ consists of

- the *message set* $\mathcal{M} = \{1,\ldots,\mathsf{M}\}$ of M equally likely random messages M;
- the (M, n) *codebook* (or simply *code*) consisting of M length-n channel input sequences, called *codewords*;
- an *encoding function* $f\colon \mathcal{M} \to \mathcal{X}^n$ that assigns for every message $m \in \mathcal{M}$ a codeword $\mathbf{x} = (x_1,\ldots,x_n)$; and
- a *decoding function* $g\colon \mathcal{Y}^n \to \hat{\mathcal{M}}$ that maps the received channel output n-sequence $\mathbf{y}$ to a guess $\hat{m} \in \hat{\mathcal{M}}$. (Usually, we have $\hat{\mathcal{M}} = \mathcal{M}$.)

Note that an (M, n) code consist merely of a unsorted list of M codewords of length n, whereas an (M, n) coding scheme additionally also defines the encoding and decoding functions. Hence, the same code can be part of many different coding schemes.

Definition 4. A code is called *linear* if the sum of any two codewords again is a codeword.

Note that a linear code always contains the all-zero codeword.

The two main parameters of interest of a code are the number of possible messages M (the larger, the more information is transmitted) and the blocklength n (the shorter, the less time is needed to transmit the message):

- we have M equally likely messages, i.e., the entropy is $H(M) = \log_2 \mathsf{M}$ bits and we need $\log_2 \mathsf{M}$ bits to describe the message in binary form;
- we need n transmissions of a channel input symbol X_k over the channel in order to transmit the complete message.

Hence, it makes sense to give the following definition.

Definition 5. The *rate*[3] of a (M, n) code is defined as

$$\mathsf{R} \triangleq \frac{\log_2 \mathsf{M}}{n} \text{ bits/transmission.} \tag{4}$$

It describes what amount of information (i.e., what part of the $\log_2 \mathsf{M}$ bits) is transmitted in each channel use.

However, this definition of a rate makes only sense if the message really arrives at the receiver, i.e., if the receiver does not make a decoding error!

[3] We define the rate here using a logarithm of base 2. However, we can use any logarithm as long as we adapt the units accordingly.

Definition 6. An (M, n) coding scheme for a DMC consists of a codebook $\mathscr{C}^{(\mathsf{M},n)}$ with M binary codewords $\mathbf{x}_m$ of length n, an encoder that maps every message m into its corresponding codeword $\mathbf{x}_m$, and a decoder that makes a decoding decision $g(\mathbf{y}) \in \{1, \ldots, \mathsf{M}\}$ for every received binary n-vector $\mathbf{y}$.

We will always assume that the M possible messages are equally likely.

Definition 7. Given that message m has been sent, let $\lambda_m^{(n)}$ be the *probability of a decoding error* of an (M, n) coding scheme with blocklength n:

$$\lambda_m^{(n)} \triangleq \Pr[g(\mathbf{Y}) \neq m | \mathbf{X} = \mathbf{x}_m] \tag{5}$$

$$= \sum_{\mathbf{y}} P_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}_m)\ \mathrm{I}\{g(\mathbf{y}) \neq m\}, \tag{6}$$

where $\mathrm{I}\{\cdot\}$ is the indicator function

$$\mathrm{I}\{\text{statement}\} \triangleq \begin{cases} 1 & \text{if statement is true,} \\ 0 & \text{if statement is wrong.} \end{cases} \tag{7}$$

The *maximum error probability* $\lambda^{(n)}$ of an (M, n) coding scheme is defined as

$$\lambda^{(n)} \triangleq \max_{m \in \mathcal{M}} \lambda_m. \tag{8}$$

The *average error probability* $P_{\mathrm{e}}^{(n)}$ of an (M, n) coding scheme is defined as

$$P_{\mathrm{e}}^{(n)} \triangleq \frac{1}{\mathsf{M}} \sum_{m=1}^{\mathsf{M}} \lambda_m^{(n)}. \tag{9}$$

Moreover, sometimes it will be more convenient to focus on the probability of not making any error, denoted *success probability* $\psi_m^{(n)}$:

$$\psi_m^{(n)} \triangleq \Pr[g(\mathbf{Y}) = m | \mathbf{X} = \mathbf{x}_m] \tag{10}$$

$$= \sum_{\mathbf{y}} P_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}_m) \mathrm{I}\{g(\mathbf{y}) = m\}. \tag{11}$$

The definition of minimum success probability $\psi^{(n)}$ and the average success probability[4] $P_{\mathrm{c}}^{(n)}$ are accordingly.

Definition 8. For a given codebook $\mathscr{C}$, we define the *decoding region* $\mathcal{D}_m$ corresponding to the m-th codeword $\mathbf{x}_m$ as follows:

$$\mathcal{D}_m \triangleq \{\mathbf{y} \colon g(\mathbf{y}) = m\}. \tag{12}$$

Note that we will always assume that the decoder g is a *maximum likelihood (ML) decoder*:

$$g(\mathbf{y}) \triangleq \arg \max_{1 \leq m \leq \mathsf{M}} P_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}_m) \tag{13}$$

that minimizes the average error probability $P_{\mathrm{e}}^{(n)}$.

Hence, we are going to be lazy and directly concentrate on the set of codewords $\mathscr{C}^{(\mathsf{M},n)}$, called (M, n) *codebook* or usually simply (M, n) *code*. Sometimes we follow the custom of traditional coding theory and use three parameters: $\big(\mathsf{M}, n, d\big)$ *code*, where the third parameter d denotes the *minimum Hamming distance*, i.e., the minimum number of components in which any two codewords differ.

Moreover, we also make the following definitions.

[4]The subscript "c" stands for "correct."

Definition 9. By $d_{\alpha,\beta}(\mathbf{x}_m, \mathbf{y})$ we denote the number of positions j, where $x_{m,j} = \alpha$ and $y_j = \beta$. For $m \neq m'$, the *joint composition* $q_{\alpha,\beta}(m, m')$ of two codewords $\mathbf{x}_m$ and $\mathbf{x}_{m'}$ is defined as

$$q_{\alpha,\beta}(m, m') \triangleq \frac{d_{\alpha,\beta}(\mathbf{x}_m, \mathbf{x}_{m'})}{n}. \tag{14}$$

Note that $d_{\mathrm{H}}(\cdot,\cdot) \triangleq d_{0,1}(\cdot,\cdot) + d_{1,0}(\cdot,\cdot)$ and $w_{\mathrm{H}}(\mathbf{x}) \triangleq d_{\mathrm{H}}(\mathbf{x}, \mathbf{0})$ denote the commonly used Hamming distance and Hamming weight, respectively.

The following remark deals with the way how codebooks can be described. It is not standard, but turns out to be very important and is actually the clue to our derivations.

Remark 10. Usually, the codebook $\mathscr{C}^{(\mathsf{M},n)}$ is written as an $\mathsf{M} \times n$ *codebook matrix* with the M rows corresponding to the M codewords:

$$\mathscr{C}^{(\mathsf{M},n)} = \begin{pmatrix} — \ \mathbf{x}_1 \ — \\ \vdots \\ — \ \mathbf{x}_{\mathsf{M}} \ — \end{pmatrix} = \begin{pmatrix} | & | & & | \\ \mathbf{c}_1 & \mathbf{c}_2 & \cdots & \mathbf{c}_n \\ | & | & & | \end{pmatrix}. \tag{15}$$

However, it turns out to be much more convenient to consider the codebook *column-wise* rather than row-wise! We denote the column-vectors of the codebook by $\mathbf{c}$.

Remark 11. Since we assume equally likely messages, any permutation of rows only changes the assignment of codewords to messages and has therefore no impact on the performance. We thus consider two codes with permuted rows as being *equal* (this agrees with the thinking of a code being a *set* of codewords, where the ordering of the codewords is irrelevant).

Furthermore, since we only consider memoryless channels, any permutation of the columns of $\mathscr{C}^{(\mathsf{M},n)}$ will lead to another code that will result in the same error probability. We say that such two codes are *equivalent*. We would like to emphasize that two codes being equivalent is not the same as two codes being equal. However, as we are mainly interested in the performance of a code, we usually treat two equivalent codes as being the same.

In spite of this, for the sake of clarity of our derivations, we usually will define a certain fixed order of the codewords/codebook column vectors.

The most famous relation between code rate and error probability has been derived by Shannon in his landmark paper from 1948 [1].

Theorem 12 (The Channel Coding Theorem for a DMC). *Define*

$$\mathsf{C} \triangleq \max_{P_X(\cdot)} I(X;Y) \tag{16}$$

where X and Y have to be understood as input and output of a DMC and where the maximization is over all input distributions $P_X(\cdot)$.

Then for every $\mathsf{R} < \mathsf{C}$ there exists a sequence of $(2^{n\mathsf{R}}, n)$ coding schemes with maximum error probability $\lambda^{(n)} \to 0$ as the blocklength n gets very large.

Conversely, any sequence of $(2^{n\mathsf{R}}, n)$ coding schemes with maximum error probability $\lambda^{(n)} \to 0$ must have a rate $\mathsf{R} \leq \mathsf{C}$.

So we see that C denotes the maximum rate at which reliable communication is possible. Therefore C is called ***channel capacity****.*

Note that this theorem considers only the situation of n tending to infinity and thereby the error probability going to zero. However, in a practical system, we cannot allow the blocklength n to be too large because of delay and complexity. On the other hand it is not necessary to have zero error probability either.

So the question arises what we can say about "capacity" for finite n, i.e., if we allow a certain maximal probability of error, what is the smallest necessary blocklength n to achieve it? Or, vice versa, fixing a certain short blocklength n, what is the best average error probability that can be achieved? And, what is the optimal code structure for a given channel?

3 Channel Model

We consider the *binary erasure channel (BEC)* given in Figure 1. The BEC is a

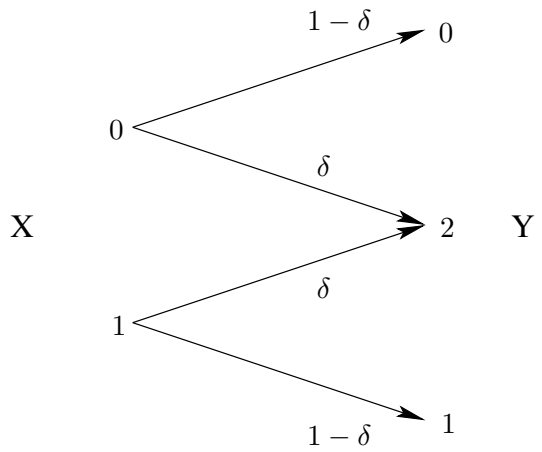

Figure 1: The binary erasure channel (BEC) with erasure probability ϵ. The channel output 2 corresponds to an erasure.

discrete memoryless channel (DMC) with binary input X and ternary output Y and with a conditional channel probability

$$P_{Y|X}(y|x) = \begin{cases} 1-\delta & \text{if } y = x,\ x \in \{0,1\}, \\ \delta & \text{if } y = 2,\ x \in \{0,1\}. \end{cases} \tag{17}$$

Here $0 \le \delta \le 1$ is called the *erasure probability*.

Due to the symmetry of the BEC, we have an additional equivalence in the codebook design.

Lemma 13. *Consider an arbitrary code $\mathscr{C}^{(\mathsf{M},n)}$ to be used on the BEC and consider an arbitrary M-vector $\mathbf{c}$. Now construct a new length-$(n+1)$ code $\mathscr{C}^{(\mathsf{M},n+1)}$ by appending $\mathbf{c}$ to the codebook matrix of $\mathscr{C}^{(\mathsf{M},n)}$ and a new length-$(n+1)$ code $\overline{\mathscr{C}}^{(\mathsf{M},n+1)}$ by appending the flipped vector $\bar{\mathbf{c}} = \mathbf{c} \oplus \mathbf{1}$ to the codebook matrix of $\mathscr{C}^{(\mathsf{M},n)}$. Then the performance of these two new codes is identical:*

$$P_{\mathrm{e}}^{(n+1)}\big(\mathscr{C}^{(\mathsf{M},n+1)}\big) = P_{\mathrm{e}}^{(n+1)}\Big(\overline{\mathscr{C}}^{(\mathsf{M},n+1)}\Big). \tag{18}$$

We remind the reader that our ultimate goal is to find the structure of an optimal code $\mathscr{C}^{(\mathsf{M},n)*}$ that satisfies

$$P_{\mathrm{e}}^{(n)}\big(\mathscr{C}^{(\mathsf{M},n)*}\big) \le P_{\mathrm{e}}^{(n)}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \tag{19}$$

for any code $\mathscr{C}^{(\mathsf{M},n)}$.

4 Preliminaries

4.1 Capacity of the BEC

The capacity of a BEC is given by

$$\mathrm{C}_{\mathrm{BEC}} = 1 - \delta \tag{20}$$

bits. Then input distribution $P_X^*(\cdot)$ that achieve the capacity is the uniform distribution given by

$$P_X^*(0) = 1 - P_X^*(1) = \frac{1}{2} \tag{21}$$

4.2 Error (and Success) Probability of the BEC

Definition 14. To make the conditional probability express shortly, we defined the number of times the symbol a occurs in one received vector $\mathbf{y}$ by $\mathrm{N}(a|\mathbf{y})$. By $\mathrm{I}(a|\mathbf{y})$ we denote the set of indices i such that $y_i = a$, hence $\mathrm{N}(a|\mathbf{y}) = |\mathrm{I}(a|\mathbf{y})|$, i.e., $\mathbf{x}_{\mathrm{I}(a|\mathbf{y})}$ is a vector of length $\mathrm{N}(a|\mathbf{y})$ containing all x_i where $i \in \mathrm{I}(a|\mathbf{y})$.

It is often easier to maximize the success probability instead of minimizing the error probability. For the convenience of later derivations, we are going to derive its error and success probabilities:

$$P_{\mathrm{c}}(\mathcal{C}^{(\mathrm{M},n)}) = \frac{1}{\mathrm{M}} \sum_{m=1}^{\mathrm{M}} \sum_{\substack{\mathbf{y} \\ g(\mathbf{y})=m}} (1-\epsilon)^{n-\mathrm{N}(2|\mathbf{y})} \cdot \epsilon^{\mathrm{N}(2|\mathbf{y})} \cdot \mathrm{I}\left\{ d_{\mathrm{H}}\left(\mathbf{x}_{m\,\mathrm{I}(b|\mathbf{y})}, \mathbf{y}_{\mathrm{I}(b|\mathbf{y})} \right) = 0 \right\}, \tag{22}$$

where $b \in \{0, 1\}$. The error probability formula is accordingly

$$P_{\mathrm{e}}(\mathcal{C}^{(\mathrm{M},n)}) = \frac{1}{\mathrm{M}} \sum_{m=1}^{\mathrm{M}} \sum_{\substack{\mathbf{y} \\ g(\mathbf{y})\neq m}} (1-\epsilon)^{n-\mathrm{N}(2|\mathbf{y})} \cdot \epsilon^{\mathrm{N}(2|\mathbf{y})} \cdot \mathrm{I}\left\{ d_{\mathrm{H}}\left(\mathbf{x}_{m\,\mathrm{I}(b|\mathbf{y})}, \mathbf{y}_{\mathrm{I}(b|\mathbf{y})} \right) = 0 \right\} \tag{23}$$

4.3 Pairwise Hamming Distance

The minimum Hamming distance is a well-known and often used quality criterion of a code. Unfortunately, a design based on the minimum Hamming distance can be strictly suboptimal even for a very symmetric channel like the BSC and even for linear codes, although the error probability performance of a BSC is completely specified by the Hamming distances between codewords and received vectors [6].

We therefore define a slightly more general and more concise description of a code: the *pairwise Hamming distance vector*.

Definition 15. Given a code $\mathscr{C}^{(\mathsf{M},n)}$ with codewords $\mathbf{x}_m$ we define the *pairwise Hamming distance vector* $\mathbf{d}\big(\mathscr{C}^{(\mathsf{M},n)}\big)$ of length $\frac{(\mathsf{M}-1)\mathsf{M}}{2}$ as

$$\begin{aligned}
&\mathbf{d}\big(\mathscr{C}^{(\mathsf{M},n)}\big)\\
&\quad\triangleq \Big(d_{\mathrm{H}}(\mathbf{x}_1,\mathbf{x}_2),\\
&\qquad d_{\mathrm{H}}(\mathbf{x}_1,\mathbf{x}_3),\ d_{\mathrm{H}}(\mathbf{x}_2,\mathbf{x}_3),\\
&\qquad d_{\mathrm{H}}(\mathbf{x}_1,\mathbf{x}_4),\ d_{\mathrm{H}}(\mathbf{x}_2,\mathbf{x}_4),\ d_{\mathrm{H}}(\mathbf{x}_3,\mathbf{x}_4),\\
&\qquad \ldots,\\
&\qquad d_{\mathrm{H}}(\mathbf{x}_1,\mathbf{x}_{\mathsf{M}}),\ d_{\mathrm{H}}(\mathbf{x}_2,\mathbf{x}_{\mathsf{M}}),\ldots,\ d_{\mathrm{H}}(\mathbf{x}_{\mathsf{M}-1},\mathbf{x}_{\mathsf{M}})\Big),
\end{aligned} \tag{24}$$

The *minimum Hamming distance* $d_{\min}\big(\mathscr{C}^{(\mathsf{M},n)}\big)$ is defined as the minimum component of the pairwise Hamming distance vector $\mathbf{d}\big(C^{(\mathsf{M},n)}\big)$.

5 Weak Flip Codes and Hadamard Codes

We next introduce some special families of binary codes. We start with a family of codes with two codewords.

Definition 16. The *flip code of type* t for $t \in \big\{0,1,\ldots,\lfloor\frac{n}{2}\rfloor\big\}$ is a code with $\mathsf{M}=2$ codewords defined by the following codebook matrix $\mathscr{C}_t^{(2,n)}$:

$$\mathscr{C}_t^{(2,n)} \triangleq \begin{pmatrix}\mathbf{x}\\ \bar{\mathbf{x}}\end{pmatrix} = \begin{pmatrix}0 & \cdots & 0 & \overbrace{1 & \cdots & 1}^{t \text{ columns}}\\ 1 & \cdots & 1 & 0 & \cdots & 0\end{pmatrix}. \tag{25}$$

Defining the column vectors

$$\left\{\mathbf{c}_1^{(2)} \triangleq \begin{pmatrix}0\\1\end{pmatrix},\ \mathbf{c}_2^{(2)} \triangleq \begin{pmatrix}1\\0\end{pmatrix}\right\}, \tag{26}$$

we see that a flip code of type t is given by a codebook matrix that consists of $n-t$ columns $\mathbf{c}_1^{(2)}$ and t columns $\mathbf{c}_2^{(2)}$.

We again remind the reader that due to the memorylessness of the BEC, other codes with the same columns as $\mathscr{C}_t^{(2,n)}$, but in different order are equivalent to $\mathscr{C}_t^{(2,n)}$. Moreover, we would like to point out that while the flip code of type 0 corresponds to a repetition code, the general flip code of type t with $t>0$ is neither a repetition code nor is it even linear.

We have shown in [6] that for any blocklength n and for a correct choice[5] of t, the flip codes are optimal on *any* binary-input binary-output channel for arbitrary channel parameters. In particular, they are optimal for the BSC and the ZC [6].

The columns given in the set in (26) are called *candidate columns*. They are flipped versions of each other, therefore also the name of the code.

The definition of a flip code with one codeword being the flipped version of the other cannot be easily extended to a situation with more than two codewords. Hence, for $\mathsf{M}>2$, we need a new approach. We give the following definition.

[5] We would like to emphasize that the optimal choice of t for many binary channels is not 0, i.e., the linear repetition code is not optimal!

Definition 17. Given an $\mathsf{M} > 2$, a length-M candidate column $\mathbf{c}$ is called a *weak flip column* if its first component is 0 and its Hamming weight equals to $\lfloor \frac{\mathsf{M}}{2} \rfloor$ or $\lceil \frac{\mathsf{M}}{2} \rceil$. The collection of all possible weak flip columns is called *weak flip candidate columns set* and is denoted by $\mathcal{C}^{(\mathsf{M})}$.

We see that a weak flip column contains an almost equal number of zeros and ones. The restriction of the first component to be zero is based on the insight of Lemma 13. For the remainder of this work, we introduce the shorthand

$$\ell \triangleq \left\lceil \frac{\mathsf{M}}{2} \right\rceil. \tag{27}$$

Lemma 18. *The cardinality of a weak flip candidate columns set is*

$$\left|\mathcal{C}^{(\mathsf{M})}\right| = \binom{2\ell - 1}{\ell}. \tag{28}$$

Proof: If $\mathsf{M} = 2\ell$, then we have $\binom{2\ell-1}{\ell}$ possible choices, while if $\mathsf{M} = 2\ell - 1$, we have $\binom{2\ell-2}{\ell-1} + \binom{2\ell-2}{\ell} = \binom{2\ell-1}{\ell}$ choices. □

We are now ready to generalize Definition 16.

Definition 19. A *weak flip code* is a codebook that is constructed only by weak flip columns.

Concretely, for $\mathsf{M} = 3$ or $\mathsf{M} = 4$, we have the following.

Definition 20. The *weak flip code of type* (t_2, t_3) for $\mathsf{M} = 3$ or $\mathsf{M} = 4$ codewords is defined by a codebook matrix $\mathscr{C}_{t_2,t_3}^{(\mathsf{M},n)}$ that consists of $t_1 \triangleq n - t_2 - t_3$ columns $\mathbf{c}_1^{(\mathsf{M})}$, t_2 columns $\mathbf{c}_2^{(\mathsf{M})}$, and t_3 columns $\mathbf{c}_3^{(\mathsf{M})}$, where

$$\left\{ \mathbf{c}_1^{(3)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \mathbf{c}_2^{(3)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \mathbf{c}_3^{(3)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \right\} \tag{29}$$

or

$$\left\{ \mathbf{c}_1^{(4)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \mathbf{c}_2^{(4)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \mathbf{c}_3^{(4)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \right\}, \tag{30}$$

respectively. We often describe the weak flip code of type (t_2, t_3) by its *code parameters*

$$[t_1, t_2, t_3] \tag{31}$$

where t_1 can be computed from the blocklength n and the type (t_2, t_3) as $t_1 = n - t_2 - t_3$. Moreover, we use

$$\mathcal{D}_{t_2,t_3;m}^{(\mathsf{M},n)} \triangleq \{\mathbf{y} \colon g(\mathbf{y}) = m\} \tag{32}$$

to denote the decoding region of the mth codeword of $\mathscr{C}_{t_2,t_3}^{(\mathsf{M},n)}$.

An interesting subfamily of weak flip codes of type (t_2, t_3) for $\mathsf{M} = 3$ or $\mathsf{M} = 4$ is defined as follows.

Definition 21. A *fair weak flip* code of type (t_2, t_3), $\mathscr{C}_{t_2,t_3}^{(\mathsf{M},n)}$, with $\mathsf{M} = 3$ or $\mathsf{M} = 4$ codewords satisfies that

$$t_1 = t_2 = t_3. \tag{33}$$

Note that the fair weak flip code of type (t_2, t_3) is only defined provided that the blocklength satisfies $n \bmod 3 = 0$. In order to be able to provide convenient comparisons for every blocklength n, we define a *generalized fair weak flip code* for every n, $\mathscr{C}_{\lfloor \frac{n+1}{3} \rfloor, \lfloor \frac{n}{3} \rfloor}^{(\mathsf{M},n)}$, where

$$t_2 = \left\lfloor \frac{n+1}{3} \right\rfloor, \quad t_3 = \left\lfloor \frac{n}{3} \right\rfloor. \tag{34}$$

If $n \bmod 3 = 0$, the generalized fair weak flip code actually is a fair weak flip code.

The following lemma follows from the respective definitions in a straightforward manner. We therefore omit its proof.

Lemma 22. *The pairwise Hamming distance vector of a weak flip code of type* (t_2, t_3) *can be computed as follows:*

$$\begin{aligned}
\mathbf{d}^{(3,n)} &= (t_2 + t_3, t_1 + t_3, t_1 + t_2), \\
\mathbf{d}^{(4,n)} &= (t_2 + t_3, t_1 + t_3, t_1 + t_2, t_1 + t_2, t_1 + t_3, t_2 + t_3).
\end{aligned}$$

A similar definition can be given also for larger M, however, one needs to be aware that the number of weak flip candidate columns is increasing fast. For $\mathsf{M} = 5$ or $\mathsf{M} = 6$ we have ten weak flip candidate columns:

$$\left\{ \mathbf{c}_1^{(5)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \mathbf{c}_2^{(5)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \mathbf{c}_3^{(5)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \right.$$
$$\mathbf{c}_4^{(5)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mathbf{c}_5^{(5)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \mathbf{c}_6^{(5)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \mathbf{c}_7^{(5)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix},$$
$$\left. \mathbf{c}_8^{(5)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \mathbf{c}_9^{(5)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \mathbf{c}_{10}^{(5)} \triangleq \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \right\}, \tag{35}$$

and

$$\left\{ \mathbf{c}_1^{(6)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \mathbf{c}_2^{(6)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \mathbf{c}_3^{(6)} \triangleq \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix}, \right.$$

$$\mathbf{c}_4^{(6)} \triangleq \begin{pmatrix}0\\0\\1\\1\\1\\0\end{pmatrix}, \mathbf{c}_5^{(6)} \triangleq \begin{pmatrix}0\\1\\0\\0\\1\\1\end{pmatrix}, \mathbf{c}_6^{(6)} \triangleq \begin{pmatrix}0\\1\\0\\1\\0\\1\end{pmatrix}, \mathbf{c}_7^{(6)} \triangleq \begin{pmatrix}0\\1\\0\\1\\1\\0\end{pmatrix},$$

$$\left.\mathbf{c}_8^{(6)} \triangleq \begin{pmatrix}0\\1\\1\\0\\0\\1\end{pmatrix}, \mathbf{c}_9^{(6)} \triangleq \begin{pmatrix}0\\1\\1\\0\\1\\0\end{pmatrix}, \mathbf{c}_{10}^{(6)} \triangleq \begin{pmatrix}0\\1\\1\\1\\0\\0\end{pmatrix}\right\}, \tag{36}$$

respectively.

We will next introduce a generalized fair weak flip codes, as we will see in Section 7, possess particularly beautiful properties.

Definition 23. A weak flip code is called *fair* if it is constructed by an equal number of all possible weak flip candidate columns in $\mathcal{C}^{(\mathrm{M})}$. Note that by definition the blocklength of a fair weak flip code is always a multiple of $\binom{2\ell-1}{\ell}$, $\ell \geq 2$.

Fair weak flip codes have been used by Shannon *et al.* [8] for the derivation of error exponents, although the codes were not named at that time. Note that the error exponents are defined when the blocklength n goes to infinity, but in this work we consider finite n.

Related to the weak flip codes and the fair weak flip codes are the families of *Hadamard codes* [4, Ch. 2].

Definition 24. For an even integer n, a (*normalized*) *Hadamard matrix* H_n of order n is an $n \times n$ matrix with entries $+1$ and -1 and with the first row and column being all $+1$, such that

$$\mathsf{H}_n \mathsf{H}_n^{\mathsf{T}} = n\mathsf{I}_n, \tag{37}$$

if such a matrix exists. Here I_n is the identity matrix of size n. If the entries $+1$ are replaced by 0 and the entries -1 by 1, H_n is changed into the *binary Hadamard matrix* A_n.

Note that a necessary (but not sufficient) condition for the existence of H_n (and the corresponding A_n) is that n is a 1, 2 or multiple of 4 [4, Ch. 2].

Definition 25. The binary Hadamard matrix A_n gives rise to three families of Hadamard codes:

1. The $\left(n, n-1, \frac{n}{2}\right)$ *Hadamard code* $\mathscr{H}_{1,n}$ consists of the rows of A_n with the first column deleted. The codewords in $\mathscr{H}_{1,n}$ that begin with 0 form the $\left(\frac{n}{2}, n-2, \frac{n}{2}\right)$ *Hadamard code* $\mathscr{H}'_{1,n}$ if the initial zero is deleted.

2. The $\left(2n, n-1, \frac{n}{2}-1\right)$ *Hadamard code* $\mathscr{H}_{2,n}$ consists of $\mathscr{H}_{1,n}$ together with the complements of all its codewords.

3. The $\left(2n, n, \frac{n}{2}\right)$ *Hadamard code* $\mathscr{H}_{3,n}$ consists of the rows of A_n and their complements.

Further Hadamard codes can be created by an arbitrary combinations of the codebook matrices of different Hadamard codes.

Example 26. Consider a $(6, 10, 6)$ $\mathscr{H}'_{1,12}$ code:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \tag{38}$$

From this code, see the candidate columns (36) for $\mathsf{M} = 6$, it is identical to the fair weak flip code for $\mathsf{M} = 6$. Since the fair weak flip code already used up all the possible weak flip candidate columns, hence, there is only one $(6, 10, 6)$ $\mathscr{H}'_{1,12}$ in column-wise respect. ◊

Example 27. Consider an $(8, 7, 4)$ $\mathscr{H}_{1,8}$ code:

$$\mathscr{H}^1_{1,8} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{39}$$

and the other $(8, 7, 4)$ $\mathscr{H}^2_{1,8}$ code:

$$\mathscr{H}^2_{1,8} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}. \tag{40}$$

From these codes, an $(8, 35, 20)$ Hadamard code can be constructed by simply concatenating $\mathscr{H}^1_{1,8}$ five times, or concatenating $\mathscr{H}^1_{1,8}$ three times and $\mathscr{H}^2_{1,8}$ two times. ◊

Note that since the rows of H_n are orthogonal, any two rows of A_n agree in $\frac{1}{2}n$ places and differ in $\frac{1}{2}n$ places, i.e., they have a Hamming distance $\frac{1}{2}n$. Moreover, by definition the first row of a binary Hadamard matrix is the all-zero row. Hence, we see that all Hadamard codes are weak flip codes, i.e., the family of weak flip codes is a superset of Hadamard codes.

On the other hand, every Hadamard code of parameters (M, n), for which fair weak flip codes exist, is not necessarily equivalent to a fair weak flip code. We also would like to remark that the Hadamard codes rely on the existence of Hadamard matrices. So in general, it is very difficult to predict whether for a given pair (M, n), a Hadamard code will exist or not. This is in stark contrast to weak flip codes (which exist for all M and n) and fair weak flip codes (which exist for all M and all n being a multiple of $\binom{2\ell-1}{\ell}$).

Example 28. We continue with Example 27 and note that the $(8, 35, 20)$ Hadamard code that is constructed by five repetitions of the matrix given in (39) is actually not a fair weak flip code, since we have to use up all possible weak flip candidate columns to get a $(8, 35, 20)$ fair weak flip code. $\Diamond$

Note that two Hadamard matrices can be *equivalent* if one can be obtained from the other by permuting rows and columns and multiplying rows and columns by -1. In other words, Hadamard codes can actually be constructed from weak candidate columns. This also follows directly from the already mentioned fact that Hadamard codes are weak flip codes.

6 Previous Work

6.1 SGB Bounds on the Average Error Probability

In [8], Shannon, Gallager, and Berlekamp derive upper and lower bounds on the average error probability of a given code used on a DMC. We next quickly review their results.

Definition 29. For $0 < s < 1$ we define

$$\mu_{\alpha,\beta}(s) \triangleq \log \sum_y P_{Y|X}(y|\alpha)^{1-s} P_{Y|X}(y|\beta)^s. \tag{41}$$

Then the *discrepancy* $\mathsf{D}^{(\mathrm{DMC})}(m, m')$ between $\mathbf{x}_m$ and $\mathbf{x}_{m'}$ is defined as

$$\mathsf{D}^{(\mathrm{DMC})}(m, m') \triangleq - \min_{0 \le s \le 1} \sum_\alpha \sum_\beta q_{\alpha,\beta}(m, m') \mu_{\alpha,\beta}(s) \tag{42}$$

with $q_{\alpha,\beta}(m, m')$ given in Def. 9.

Note that the discrepancy is a generalization of the Hamming distance, however, it depends strongly on the channel crossover probabilities. We use a superscript "(DMC)" to indicate the channel which the discrepancy refers to.

Definition 30. The *minimum discrepancy* $\mathsf{D}_{\min}^{(\mathrm{DMC})}(\mathscr{C}^{(\mathsf{M},n)})$ for a codebook is the minimum value of $\mathsf{D}^{(\mathrm{DMC})}(m, m')$ over all pairs of codewords. The *maximum minimum discrepancy* is the maximum value of $\mathsf{D}_{\min}^{(\mathrm{DMC})}(\mathscr{C}^{(\mathsf{M},n)})$ over all possible $\mathscr{C}^{(\mathsf{M},n)}$ codebooks: $\max_{\mathscr{C}^{(\mathsf{M},n)}} \mathsf{D}_{\min}^{(\mathrm{DMC})}(\mathscr{C}^{(\mathsf{M},n)})$.

Theorem 31. *If $\mathbf{x}_m$ and $\mathbf{x}_{m'}$ are a pair of codewords in a code of blocklength n, then either*

$$P_{\mathrm{e},m} > \frac{1}{4} \exp\left(-n \left[\mathsf{D}^{(\mathrm{DMC})}(m, m') + \sqrt{\frac{2}{n}} \log\left(1/P_{\min}\right) \right] \right) \tag{43}$$

or

$$P_{\mathrm{e},m'} > \frac{1}{4} \exp\left(-n \left[\mathsf{D}^{(\mathrm{DMC})}(m, m') + \sqrt{\frac{2}{n}} \log\left(1/P_{\min}\right) \right] \right), \tag{44}$$

where $P_{\min}$ is the smallest nonzero transition probability for the channel.

Conversely, one can also show that

$$P_{\mathrm{e},m} \le (\mathsf{M} - 1) \exp\left(-n \mathsf{D}_{\min}^{(\mathrm{DMC})}\left(\mathscr{C}^{(\mathsf{M},n)}\right) \right), \quad \text{for all } m. \tag{45}$$

Theorem 32 (SGB Bounds on Average Error Probability [8]). *For an arbitrary DMC, the average error probability* $P_{\mathrm{e}}\big(\mathscr{C}^{(\mathsf{M},n)}\big)$ *of a given code* $\mathscr{C}^{(\mathsf{M},n)}$ *with* M *codewords and blocklength* n *is upper- and lower-bounded as follows:*

$$\frac{1}{4\mathsf{M}} e^{-n\left(\mathsf{D}_{\min}^{(\mathrm{DMC})}(\mathscr{C}^{(\mathsf{M},n)})+\sqrt{\frac{2}{n}}\log\frac{1}{P_{\min}}\right)} \leq P_{\mathrm{e}}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \leq (\mathsf{M}-1)e^{-n\mathsf{D}_{\min}^{(\mathrm{DMC})}(\mathscr{C}^{(\mathsf{M},n)})} \tag{46}$$

where $P_{\min}$ *denotes the smallest nonzero transition probability of the channel.*

Note that these bounds are specific to a given code design (via $\mathsf{D}_{\min}^{(\mathrm{DMC})}$). Therefore, the upper bound is a generally valid upper bound on the optimal performance, while the lower bound only holds in general if we apply it to the optimal code or to a suboptimal code that achieves the optimal $\mathsf{D}_{\min}$.

The bounds (46) are tight enough to derive the *error exponent* of the DMC (for a fixed number M of codewords).

Theorem 33 ([8]). *The error exponent of a DMC for a fixed number* M *of codewords*

$$\mathsf{E}_{\mathsf{M}} \triangleq \varlimsup_{n\to\infty} \max_{\mathscr{C}^{(\mathsf{M},n)}} \left\{-\frac{1}{n}\log P_{\mathrm{e}}\big(\mathscr{C}^{(\mathsf{M},n)}\big)\right\} \tag{47}$$

is given as

$$\mathsf{E}_{\mathsf{M}} = \lim_{n\to\infty} \max_{\mathscr{C}^{(\mathsf{M},n)}} \mathsf{D}_{\min}^{(\mathrm{DMC})}\big(\mathscr{C}^{(\mathsf{M},n)}\big). \tag{48}$$

Unfortunately, in general the evaluation of the error exponent is very difficult. For some cases, however, it can be done. For example, for $\mathsf{M}=2$, we have

$$\mathsf{E}_2 = \max_{\mathscr{C}^{(2,n)}} \mathsf{D}_{\min}^{(\mathrm{DMC})}\big(\mathscr{C}^{(2,n)}\big) = \max_{\alpha,\beta}\left\{-\min_{0\leq s\leq 1}\mu_{\alpha,\beta}(s)\right\}. \tag{49}$$

Also for the class of so-called *pairwise reversible channels*, the calculation of the error exponent turns out to be uncomplicated.

Definition 34. A *pairwise reversible channel* is a DMC that has $\mu'_{\alpha,\beta}(\frac{1}{2}) = 0$ for any inputs α, β.

Clearly, the BSC is a pairwise reversible channel.

Note that it is easy to compute the pairwise discrepancy of a linear code on a pairwise reversible channel, so linear codes are quite suitable for computing (46).

Theorem 35 ([8]). *For pairwise reversible channels with* $\mathsf{M} > 2$,

$$\mathsf{E}_{\mathsf{M}} = \frac{1}{\mathsf{M}(\mathsf{M}-1)} \max_{\substack{\mathsf{M}_x \text{ s.t.} \\ \sum_x \mathsf{M}_x = \mathsf{M}}} \sum_{\substack{\text{all input} \\ \text{letters } x}} \sum_{\substack{\text{all input} \\ \text{letters } x'}} \mathsf{M}_x \mathsf{M}_{x'} \cdot \left(-\log \sum_y \sqrt{P_{Y|X}(y|x)P_{Y|X}(y|x')}\right) \tag{50}$$

where M_x *denotes the number of times the channel input letter* x *occurs in a column. Moreover,* E_{M} *is achieved by fair weak flip codes.*[6]

[6] While throughout we only consider binary inputs and $\mathsf{M}=3$ or $\mathsf{M}=4$, the definitions of our fair weak flip codes can be generalized to nonbinary inputs and larger M. Also these generalized fair weak flip codes will achieve the corresponding error exponents. Note that Shannon *et al.* did not actually name their exponent-achieving codes.

We would like to emphasize that while Shannon *et al.* proved that fair weak flip codes achieve the error exponent, they did not investigate the error performance of fair weak flip codes for finite n. As we will show later, fair weak flip might be strictly suboptimal codes for finite n (see also [9]).

6.2 Gallager Bound

Another famous bound is by Gallager [10].

Theorem 36 ([10]). *For an arbitrary DMC, there exists a code $\mathscr{C}^{(\mathsf{M},n)}$ with $\mathsf{M} = \lfloor e^{n\mathsf{R}} \rfloor$ such that*

$$P_{\mathrm{e}}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \leq e^{-n\mathsf{E}_{\mathrm{G}}(\mathsf{R})} \tag{51}$$

where $\mathsf{E}_{\mathrm{G}}(\cdot)$ is the Gallager exponent and is given by

$$\mathsf{E}_{\mathrm{G}}(\mathsf{R}) = \max_{Q(\cdot)} \max_{0\leq\rho\leq 1} \big\{\mathsf{E}_0(\rho, Q) - \rho\mathsf{R}\big\} \tag{52}$$

with

$$\mathsf{E}_0(\rho, Q) \triangleq -\log\left(\sum_y \left(\sum_x Q(x) P_{Y|X}(y|x)^{\frac{1}{1+\rho}}\right)^{1+\rho}\right). \tag{53}$$

6.3 PPV Bounds for the BEC

In [11], Polyanskiy, Poor, and Verdú present upper and lower bounds on the optimal average error probability for finite blocklength for the BEC. The upper bound is based on *random coding*.

Theorem 37. *For the BEC with crossover probability δ, the average error probability for an random code is given by*

$$\begin{aligned}&\mathsf{E}\Big[P_{\mathrm{e}}\big(\mathbb{C}^{(\mathsf{M},n)}\big)\Big] \\ &\quad = 1 - \sum_{j=0}^{n} \binom{n}{j} (1-\delta)^j \delta^{n-j} \sum_{\ell=0}^{\mathsf{M}-1} \frac{1}{\ell+1} \binom{\mathsf{M}-1}{\ell} (2^{-j})^{\ell} (1-2^{-j})^{\mathsf{M}-1-\ell}.\end{aligned} \tag{54}$$

Note that there must exist a codebook whose average error probability achieves (54), so Th. 37 provides a general achievable upper bound, although we do not know its concrete code structure.

Polyanskiy, Poor, and Verdú also provide a new general converse for the average error probability for a BEC.

Theorem 38. *For the BEC with erasure probability δ, the average error probability of a $\mathscr{C}^{(\mathsf{M},n)}$ code satisfies*

$$P_{\mathrm{e}}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \geq \sum_{\ell=\lfloor n-\log_2 \mathsf{M}\rfloor+1}^{n} \binom{n}{\ell} \delta^{\ell} (1-\delta)^{n-\ell} \left(1 - \frac{2^{n-\ell}}{\mathsf{M}}\right). \tag{55}$$

7 Main Results

7.1 Characteristics of Weak Flip Codes

In conventional coding theory, most results are restricted to so called *linear codes* that possess very powerful algebraic properties. For the following definitions and proofs see, e.g., [2], [4].

Definition 39. Let $\mathsf{M} = 2^k$, where $k \in \mathbb{N}$. The binary code $\mathscr{C}_{\text{lin}}^{(\mathsf{M},n)}$ is *linear* if its codewords span a k-dimensional subspace of $\{0,1\}^n$.

One of the most important property of a linear code is as follows.

Proposition 40. *Let $\mathscr{C}_{\text{lin}}$ be linear and let $\mathbf{x}_m \in \mathscr{C}_{\text{lin}}$ be given. Then the code that we obtain by adding $\mathbf{x}_m$ to each codeword of $\mathscr{C}_{\text{lin}}$ is equal to $\mathscr{C}_{\text{lin}}$.*

Another property concerns the column weights.

Proposition 41. *If an (M, n) binary code is linear, then each column of its codebook matrix has Hamming weight $\frac{\mathsf{M}}{2}$, i.e., the code is a weak flip code.*

Hence, linear codes are weak flip codes. Note, however, that linear codes only exist if $\mathsf{M} = 2^k$, where $k \in \mathbb{N}$, while weak flip codes are defined for any M. Also note that the converse of Proposition 41 does not hold, i.e., even if $\mathsf{M} = 2^k$ for some $k \in \mathbb{N}$, a weak flip code $\mathscr{C}^{(\mathsf{M},n)}$ is not necessarily linear. It is not even the case that a fair weak flip code for $\mathsf{M} = 2^k$ is necessarily linear!

Now the question arises as to which of the many powerful algebraic properties of linear codes are retained in weak flip codes.

Theorem 42. *Consider a weak flip code $\mathscr{C}^{(\mathsf{M},n)}$ and fix some codeword $\mathbf{x}_m \in \mathscr{C}^{(\mathsf{M},n)}$. If we add this codeword to all codewords in $\mathscr{C}^{(\mathsf{M},n)}$, then the resulting code $\tilde{\mathscr{C}}^{(\mathsf{M},n)} \triangleq \left\{\mathbf{x}_m \oplus \mathbf{x} \,\middle|\, \forall\, \mathbf{x} \in \mathscr{C}^{(\mathsf{M},n)}\right\}$ is still a weak flip code, however, it is not necessarily the same one.*

Proof: Let $\mathscr{C}^{(\mathsf{M},n)}$ be according to Definition 19. We have to prove that

$$\begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{\mathsf{M}} \end{pmatrix} \oplus \begin{pmatrix} \mathbf{x}_m \\ \mathbf{x}_m \\ \vdots \\ \mathbf{x}_m \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \oplus \mathbf{x}_m \\ \vdots \\ \mathbf{x}_m \oplus \mathbf{x}_m = \mathbf{0} \\ \vdots \\ \mathbf{x}_{\mathsf{M}} \oplus \mathbf{x}_m \end{pmatrix} \triangleq \tilde{\mathscr{C}}^{(\mathsf{M},n)} \tag{56}$$

is a weak flip code. Let $\mathbf{c}_i$ denote the column vectors of $\mathscr{C}^{(\mathsf{M},n)}$. Then $\tilde{\mathscr{C}}^{(\mathsf{M},n)}$ has the column vectors

$$\tilde{\mathbf{c}}_i = \begin{cases} \mathbf{c}_i & \text{if } x_{m,i} = 0, \\ \bar{\mathbf{c}}_i & \text{if } x_{m,i} = 1, \end{cases} \tag{57}$$

for $1 \le i \le n$. Since $\mathbf{c}_i$ is a weak flip column, either $w_{\text{H}}(\mathbf{c}_i) = \left\lfloor \frac{\mathsf{M}}{2} \right\rfloor$ and therefore $w_{\text{H}}(\bar{\mathbf{c}}_i) = \left\lceil \frac{\mathsf{M}}{2} \right\rceil$, or $w_{\text{H}}(\mathbf{c}_i) = \left\lceil \frac{\mathsf{M}}{2} \right\rceil$ and therefore $w_{\text{H}}(\bar{\mathbf{c}}_i) = \left\lfloor \frac{\mathsf{M}}{2} \right\rfloor$. So we only need to interchange the first codeword of $\tilde{\mathscr{C}}$ and the all-zero codeword in the mth row in $\tilde{\mathscr{C}}$ (which is always possible, see discussion after Definition 7), and we see that $\tilde{\mathscr{C}}$ is also a weak flip code. □

Theorem 42 is a beautiful property of weak flip codes; however, it still represents a considerable weakening of the powerful property of linear codes given in Proposition 40. This can be fixed by considering the subfamily of fair weak flip codes.

Theorem 43 (Quasi-Linear Codes). *Let $\mathscr{C}$ be a **fair** weak flip code and let $\mathbf{x}_m \in \mathscr{C}$ be given. Then the code $\tilde{\mathscr{C}} = \left\{\mathbf{x}_m \oplus \mathbf{x} \,\middle|\, \forall\, \mathbf{x} \in \mathscr{C}^{(\mathsf{M},n)}\right\}$ is equivalent to $\mathscr{C}$.*

Proof: We have already seen in Theorem 42 that adding a codeword will result in a weak flip code again. In the case of a fair weak flip code, however, all possible candidate columns will show up again with the same equal frequency. It only remains to rearrange some rows and columns. □

If we recall Proposition 41 and the discussion after it, we realize that the definition of the quasi-linear fair weak flip code is a considerable enlargement of the set of codes having the property given in Theorem 43.

The following corollary is a direct consequence of Theorem 43.

Corollary 44. *The Hamming weights of each codeword of a fair weak flip code are all identical except the all-zero codeword $\mathbf{x}_1$. In other words, if we let $w_{\mathrm{H}}(\cdot)$ be the Hamming weight function, then*

$$w_{\mathrm{H}}(\mathbf{x}_2) = w_{\mathrm{H}}(\mathbf{x}_3) = \cdots = w_{\mathrm{H}}(\mathbf{x}_{\mathsf{M}}). \tag{58}$$

Before we next investigate the minimum Hamming distance for the quasi-linear fair weak flip codes, we quickly recall an important bound that holds for any $\big(\mathsf{M}, n, d\big)$ code.

Lemma 45 (Plotkin Bound [4]). *The minimum distance of an (M, n) binary code $\mathscr{C}^{(\mathsf{M},n)}$ always satisfies*

$$d_{\min}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \leq \begin{cases} \frac{n\cdot\frac{\mathsf{M}}{2}}{\mathsf{M}-1} & \mathsf{M} \text{ even}, \\ \frac{n\cdot\frac{\mathsf{M}+1}{2}}{\mathsf{M}} & \mathsf{M} \text{ odd}. \end{cases} \tag{59}$$

Proof: We show a quick proof. We sum the Hamming distance over all possible pairs of two codewords apart from the codeword with itself:

$$\mathsf{M}(\mathsf{M}-1)\cdot d_{\min}(\mathscr{C}^{(\mathsf{M},n)}) \leq \sum_{\mathbf{u}\in\mathscr{C}^{(\mathsf{M},n)}} \sum_{\substack{\mathbf{v}\in\mathscr{C}^{(\mathsf{M},n)} \\ \mathbf{v}\neq\mathbf{u}}} d_{\mathrm{H}}(\mathbf{u},\mathbf{v}) \tag{60}$$

$$= \sum_{j=1}^{n} 2b_j\cdot(\mathsf{M}-b_j) \tag{61}$$

$$\leq \begin{cases} n\cdot\frac{\mathsf{M}^2}{2} & \text{if } \mathsf{M} \text{ even (achieved if } b_j = \mathsf{M}/2), \\ n\cdot\frac{\mathsf{M}^2-1}{2} & \text{if } \mathsf{M} \text{ odd (achieved if } b_j = (\mathsf{M}\pm 1)/2). \end{cases} \tag{62}$$

Here in (61) we rearrange the order of summation: instead of summing over all codewords (rows), we approach the problem column-wise and assume that the jth column of $\mathscr{C}^{(\mathsf{M},n)}$ contains b_j zeros and $\mathsf{M}-b_j$ ones: then this column contributes $2b_j(\mathsf{M}-b_j)$ to the sum. □

Note that from the proof of Lemma 45 we can see that a necessary condition for a codebook to meet the Plotkin-bound is that the codebook is composed by weak flip candidate columns. Furthermore, Levenshtein [4, Ch. 2] proved that the Plotkin bound can be achieved, provided that Hadamard matrices exist.

Theorem 46. *Fix some M and a blocklength n with $n \bmod \binom{2\ell-1}{\ell} = 0$. Then a fair weak flip code $\mathscr{C}^{(\mathsf{M},n)}$ achieves the largest minimum Hamming distance among all codes of given blocklength and satisfies*

$$d_{\min}\big(\mathscr{C}^{(\mathsf{M},n)}\big) = \frac{n\cdot\ell}{2\ell-1}. \tag{63}$$

Proof: For $\mathsf{M} = 2\ell$, we know that by definition the Hamming weight of each column of the codebook matrix is equal to ℓ. Hence, when changing the sum from column-wise to row-wise, where we can ignore the first row of zero weight (from the all-zero codeword $\mathbf{x}_1$), we get

$$n \cdot \ell = \sum_{j=1}^{n} w_{\mathrm{H}}(\mathbf{c}_j) = \sum_{m=2}^{2\ell} w_{\mathrm{H}}(\mathbf{x}_m) \tag{64}$$

$$= \sum_{m=2}^{2\ell} d_{\min}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \tag{65}$$

$$= (2\ell - 1) \cdot d_{\min}\big(\mathscr{C}^{(\mathsf{M},n)}\big). \tag{66}$$

Here, (65) follows from Theorem 43 and from Corollary 44. For $\mathsf{M} = 2\ell - 1$, the Hamming distance remains the same due to the fair construction.

It remains to show that a fair weak flip code achieves the largest minimum Hamming distance among all codes of given blocklength. From Corollary 44 we know that (apart from the all-zero codeword) all codewords of a fair weak flip code have the same Hamming weight. So, if we flip an arbitrary 1 in the codebook matrix to become a 0, then the corresponding codeword has a decreased Hamming weight and is therefore closer to the all-zero codeword. If we flip an arbitrary 0 to become a 1, then the corresponding codeword is closer to some other codeword that already has a 1 in this position. Hence, in both cases we have reduced the minimum Hamming distance. Finally, based on the concept of looking at the code in column-wise, it can be seen that whenever we change more than one bit, we either get back to a fair weak flip code or to another code who is worse. □

7.2 Optimal Codes on BEC

The definition of the flip, the weak flip, and the fair weak flip codes is interesting not only due to their generalization of the concept of linear codes, but also because we can show that they are optimal for the BEC for many values of the blocklength n.

Theorem 47. *For a BEC and for any $n \geq 1$, an optimal codebook with $\mathsf{M} = 2$ codewords is the flip code of type t for any $t \in \{0, 1, \ldots, \lfloor \frac{n}{2} \rfloor\}$.*

Proof: Omitted. □

Theorem 48. *For a BEC and for any $n \geq 2$, if the optimal codebook can be recursively constructed in blocklength n, an optimal codebook with $\mathsf{M} = 3$ or $\mathsf{M} = 4$ codewords is the weak flip code of type (t_2^*, t_3^*), where*

$$t_2^* \triangleq \left\lfloor \frac{n}{3} \right\rfloor, \quad t_3^* \triangleq \left\lfloor \frac{n+1}{3} \right\rfloor. \tag{67}$$

This optimal codebook can be constructed recursively in the blocklength n. We start with an optimal codebook for $n = 2$:

$$\mathcal{C}_{\mathrm{BEC}}^{(\mathsf{M},2)*} = \left(\mathbf{c}_1^{(\mathsf{M})}, \mathbf{c}_3^{(\mathsf{M})}\right). \tag{68}$$

Then, from the optimal code $\mathcal{C}_{\mathrm{BEC}}^{(\mathsf{M},n-1)*}$ *of blocklength* $n-1$*, we can recursively construct the optimal codebook of blocklength* n *by appending*

$$\begin{cases} \mathbf{c}_2^{(\mathsf{M})} & \text{if } n \bmod 3 = 0, \\ \mathbf{c}_1^{(\mathsf{M})} & \text{if } n \bmod 3 = 1, \\ \mathbf{c}_3^{(\mathsf{M})} & \text{if } n \bmod 3 = 2. \end{cases} \tag{69}$$

This theorem suggests that for a given fixed code size M, a sequence of good codes can be generated by appending proper columns to the code of smaller blocklength. We are going to sketch a proof for this theorem that is based on this recursive generation. The proof follows similar ideas as in [6, App. C], i.e., it is based on a column-wise analysis of the codebook matrix and on a mathematical induction on n. For a given DMC and code of blocklength n, we ask the question what is the optimal improvement (i.e., the maximum reduction of error probability) when increasing the blocklength n to $n+\gamma$, where $\gamma = 1$ when $\mathsf{M} = 3$ or 4 (and may be larger than 1 when $\mathsf{M} > 5$). The answer to this question then leads to the recursive construction of (69). We conclude here by a remark. While it is very intuitive to construct the codes recursively, i.e., to start from an optimal code for n and then to add one column that maximizes the total probability increase, unfortunately, from a proof perspective, such a recursive construction only guarantees local optimality: one still needs a proof that the achieved code of blocklength $n+1$ is globally optimum.

We start with the following lemma.

Lemma 49. *Fix the number of codewords* M *and a DMC. The success probability* $P_{\mathrm{c}}(\mathscr{C}^{(\mathsf{M},n)})$ *for a sequence of codes* $\{\mathscr{C}^{(\mathsf{M},n)}\}_n$*, where each code is generated by appending proper columns to the code of smaller blocklength, is nondecreasing with respect to the blocklength* n*.*

Proof of Lemma 49: For a given code $\mathscr{C}^{(\mathsf{M},n)}$, the average success probability is given as

$$P_{\mathrm{c}}\big(\mathscr{C}^{(\mathsf{M},n)}\big) = \frac{1}{\mathsf{M}} \sum_{m=1}^{\mathsf{M}} \psi_m^{(n)} \tag{70}$$

$$= \frac{1}{\mathsf{M}} \sum_{m=1}^{\mathsf{M}} \sum_{\mathbf{y}^{(n)} \in \mathcal{D}_m^{(n)}} P_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^{(n)}|\mathbf{x}_m^{(n)}). \tag{71}$$

Now we consider a new codebook $\mathscr{C}^{(\mathsf{M},n+\gamma)}$, which is formed by appending γ columns to the original codebook $\mathscr{C}^{(\mathsf{M},n)}$. For convenience, we express the new codewords by

$$\mathbf{x}_m^{(n+\gamma)} = \big[\mathbf{x}_m^{(n)}\ \mathbf{x}_m^{(\gamma)}\big] = (x_{m,1}, x_{m,2}, \ldots, x_{m,n}, \ldots, x_{m,n+\gamma}), \tag{72}$$

and likewise the extended received vector by

$$\mathbf{y}^{(n+\gamma)} = \big[\mathbf{y}^{(n)}\ \mathbf{y}^{(\gamma)}\big] = (y_1, y_2, \ldots, y_{n+\gamma}). \tag{73}$$

Assume that a length-n received vector $\mathbf{y}^{(n)}$ is in the mth decoding region, $\mathbf{y}^{(n)} \in \mathcal{D}_m^{(n)}$. According to the ML decoding rule, a corresponding new received vector $\mathbf{y}^{(n+\gamma)}$ will change to another decoding region $\mathcal{D}_{m'}^{(n+\gamma)}$ if

$$\frac{P_{\mathbf{Y}|\mathbf{X}}\Big(\big[\mathbf{y}^{(n)}\ \mathbf{y}^{(\gamma)}\big]\Big|\big[\mathbf{x}_{m'}^{(n)}\ \mathbf{x}_{m'}^{(\gamma)}\big]\Big)}{P_{\mathbf{Y}|\mathbf{X}}\Big(\big[\mathbf{y}^{(n)}\ \mathbf{y}^{(\gamma)}\big]\Big|\big[\mathbf{x}_{m}^{(n)}\ \mathbf{x}_{m}^{(\gamma)}\big]\Big)} \geq 1. \tag{74}$$

Obviously, if no extended received vectors change its original decoding region from its length-n counterpart, then

$$P_{\mathrm{c}}\big(\mathscr{C}^{(\mathsf{M},n+\gamma)}\big) = \frac{1}{\mathsf{M}} \sum_{m=1}^{\mathsf{M}} \Bigg[\sum_{\mathbf{y}^{(n)} \in \mathcal{D}_m^{(n)}} P_{\mathbf{Y}|\mathbf{X}}\big(\mathbf{y}^{(n)} \big| \mathbf{x}_m^{(n)}\big) \cdot \underbrace{\sum_{\mathbf{y}^{(\gamma)} \in \mathcal{Y}^{\gamma}} P_{\mathbf{Y}|\mathbf{X}}\big(\mathbf{y}^{(\gamma)} \big| \mathbf{x}_m^{(\gamma)}\big)}_{=1} \Bigg] \tag{75}$$

$$= P_{\mathrm{c}}\big(\mathscr{C}^{(\mathsf{M},n)}\big), \tag{76}$$

where $\mathcal{Y}$ denotes the output alphabet. If however some $\mathbf{y}^{(n+\gamma)}$ change its original decoding region of blocklength n, the new success probability will be

$$\begin{aligned} &P_{\mathrm{c}}\big(\mathscr{C}^{(\mathsf{M},n+\gamma)}\big) \\ &\quad = P_{\mathrm{c}}\big(\mathscr{C}^{(\mathsf{M},n)}\big) \\ &\qquad + \frac{1}{\mathsf{M}} \sum_{m=1}^{\mathsf{M}} \sum_{\substack{\mathbf{y}^{(n+\gamma)} \\ \text{s.t. } \mathbf{y}^{(n)} \in \mathcal{D}_m^{(n)} \\ \text{but } \mathbf{y}^{(n+\gamma)} \in \mathcal{D}_{m'}^{(n)}}} \Big[P_{\mathbf{Y}|\mathbf{X}}\big(\mathbf{y}^{(n+\gamma)} \big| \mathbf{x}_{m'}^{(n+\gamma)}\big) \\ &\qquad\qquad - P_{\mathbf{Y}|\mathbf{X}}\big(\mathbf{y}^{(n+\gamma)} \big| \mathbf{x}_{m}^{(n+\gamma)}\big) \Big] \qquad (77) \\ &\quad \triangleq P_{\mathrm{c}}\big(\mathscr{C}^{(\mathsf{M},n)}\big) + \Delta\Psi\big(\mathscr{C}^{(\mathsf{M},\gamma)}\big). \qquad (78) \end{aligned}$$

The proof of Lemma 49 is completed by noting from (74) that $\Delta\Psi(\mathscr{C}^{(\mathsf{M},\gamma)})$ is always no less than zero. □

Definition 50. The term $\Delta\Psi\big(\mathscr{C}^{(\mathsf{M},n+\gamma)}\big)$ in (78) is called *total probability increase for a step-size* γ and describes the amount by which the average success probability of the code $\mathscr{C}^{(\mathsf{M},n)}$ grows when γ column vectors are appended to its codebook matrix.

In the proof of Theorem 48, our goal is to maximize $\Delta\Psi\big(\mathscr{C}^{(\mathsf{M},\gamma)}\big)$ among all possible $\mathscr{C}^{(\mathsf{M},\gamma)}$; hence, for every blocklength n, we can maximize the improvement of performance. Note that our optimal codes based on an important assumption: if the optimal codes can be constructed recursively in maximizing the improvement of performance for every blocklength n. This induction proof for a BEC follows the lines of the proof for the BSC shown in [6, App. C] with some modifications that take into account the details of the decoding rules for the BEC. Similarly to [6, App. C], we need a case distinction depending on n mod 3. For space reason, we only outline the case from $n = 3k-1$ to $n = 3k$.

For $\mathsf{M} = 3$, we note that similarly to the proof for the BSC and due to the symmetry of the BEC (see Lemma 13), we can reduce the number of candidate columns to $\mathbf{c}_1^{(3)}, \mathbf{c}_2^{(3)}, \mathbf{c}_3^{(3)}$. We start with the code $\mathscr{C}_{t_2^*,t_3^*}^{(3,n-1)}$, whose code parameters, pairwise Hamming distance vector, and minimum Hamming distance are as follows:

$$[t_1^*, t_2^*, t_3^*] = [k, k-1, k]; \tag{79}$$

$$\mathbf{d}\big(\mathscr{C}_{t_2^*,t_3^*}^{(3,n-1)}\big) = (2k-1, 2k, 2k-1); \tag{80}$$

$$d_{\min}\big(\mathscr{C}_{t_2^*,t_3^*}^{(3,n-1)}\big) = 2k-1. \tag{81}$$

We require to show that appending $\mathbf{c}_2^{(3)}$ yields a larger success probability than appending $\mathbf{c}_1^{(3)}$ or $\mathbf{c}_3^{(3)}$. Note that appending $\mathbf{c}_1^{(3)}$ will result in the same success probability as appending $\mathbf{c}_3^{(3)}$.

Consider the three possible extended decoding regions of blocklength n, i.e., $\left[\mathcal{D}_m^{(n-1)}\ 0\right]$, $\left[\mathcal{D}_m^{(n-1)}\ 1\right]$, and $\left[\mathcal{D}_m^{(n-1)}\ 2\right]$. Owing to $P_{Y|X}(0|1) = P_{Y|X}(1|0) = 0$, we know for the mth new codeword of blocklength n with $x_{m,n} = b$, where $b \in \{0, 1\}$, its extended decoding region $\mathcal{D}_m^{(n)}$ should include both $\left[\mathcal{D}_m^{(n-1)}\ b\right]$ and $\left[\mathcal{D}_m^{(n-1)}\ 2\right]$, and all the received vectors in $\left[\mathcal{D}_m^{(n-1)}\ \bar{b}\right]$ will be decoded to one of the other two codewords. Since $\psi_m^{(n-1)}$ is equal to the occurrence probabilities of those received vectors in the union of $\left[\mathcal{D}_m^{(n-1)}\ b\right]$ and $\left[\mathcal{D}_m^{(n-1)}\ 2\right]$, $\psi_m^{(n)}$ is no less than $\psi_m^{(n-1)}$. As a result, the increment of success probability for each codeword will be determined by how the received vectors in $\left[\mathcal{D}_m^{(n-1)}\ \bar{b}\right]$ are decoded to the other two codewords.

The following claim is going to help answering this question.

Claim 51. *Let m, m' and m'' be distinct numbers in $\{1, 2, 3\}$. If $d_{\mathrm{H}}\left(\mathbf{x}_m^{(n-1)}, \mathbf{x}_{m'}^{(n-1)}\right) \geq d_{\mathrm{H}}\left(\mathbf{x}_m^{(n-1)}, \mathbf{x}_{m''}^{(n-1)}\right)$ and if $x_{m,n} = b$ is different from $x_{m',n} = x_{m'',n} = \bar{b}$, then the received vectors in $\left[\mathcal{D}_m^{(n-1)}\ \bar{b}\right]$ should be assigned to $\mathcal{D}_{m''}^{(n)}$ rather than to $\mathcal{D}_{m'}^{(n)}$, as this will result in a higher success probability.*

Proof of Claim 51: To facilitate the explanation of our idea behind the proof of Claim 51, we assume without loss of generality that $m = 1$, $m' = 2$ and $m'' = 3$, and consider $\mathbf{y}^{(n-1)} \in \mathcal{D}_1^{(n-1)}$, whose components must be either an erasure 2 or equal to the corresponding component of the first codeword: $y_j \in \{x_{1,j}, 2\}$ (where actually $x_{1,j} = 0$; also note that since $m = 1$, we have $b = 0$). Now we investigate all those length-n received vectors $\mathbf{y}^{(n)}$ in $\left[\mathcal{D}_1^{(n-1)}\ \bar{b}\right]$ with positive probability. Note that because of the last digit $y_n = \bar{b} = 1$ these vectors cannot be assigned to $\mathcal{D}_1^{(n)}$.

If there exists a position y_j of $\mathbf{y}^{(n)}$ that corresponds to a code matrix column $\mathbf{c}_1^{(3)}$ and that takes value $y_j = x_{1,j}$ $(= 0)$, then this received vector must be assigned to $\mathcal{D}_2^{(n)}$, where we can infer from the assumption of $\mathbf{y}^{(n)}$ having positive probability that all positions in $\mathbf{y}^{(n)}$ corresponding to code matrix columns $\mathbf{c}_2^{(3)}$ or $\mathbf{c}_3^{(3)}$ must be erased to 2. Likewise, if there exists a position y_j that corresponds to a code matrix column $\mathbf{c}_2^{(3)}$ and $y_j = x_{1,j}$ $(= 0)$, then such received vectors will be classified to $\mathcal{D}_3^{(n)}$, where we can infer that all positions of $\mathbf{y}^{(n)}$ corresponding to code matrix columns $\mathbf{c}_1^{(3)}$ or $\mathbf{c}_3^{(3)}$ must be 2.

Since by assumption $d_{\mathrm{H}}\left(\mathbf{x}_1^{(n-1)}, \mathbf{x}_2^{(n-1)}\right)$ is larger than $d_{\mathrm{H}}\left(\mathbf{x}_1^{(n-1)}, \mathbf{x}_3^{(n-1)}\right)$, in the code matrix of length $n-1$, $\mathbf{c}_2^{(3)}$ will occur more often than $\mathbf{c}_1^{(3)}$. We will therefore gain a higher increase in the success probability if the vectors in $\left[\mathcal{D}_1^{(n-1)}\ \bar{b}\right]$ are assigned to $\mathcal{D}_3^{(n)}$. □

Using a similar approach as shown in the proof of Claim 51 together with $d_{12}^{(n-1)} = 2k - 1 < d_{13}^{(n-1)} = 2k$, we can proceed to show that we gain a larger increment of success probability if we append $\mathbf{c}_2^{(3)}$ as the nth code matrix column rather than appending $\mathbf{c}_1^{(3)}$. This then completes the proof of the exemplified special case in Theorem 48.

Similar arguments can be applied to $\mathsf{M} = 4$.

Note that the idea of designing an optimal code recursively promises to be a very powerful approach. Unfortunately, for larger values of M, we might need a recursion

from n to $n+\gamma$ with a step-size $\gamma > 1$, and this step-size γ might be a function of blocklength n. However, based on our definition of fair weak flip codes and on Theorem 53 below, we conjecture that the necessary step-size satisfies $\gamma \leq \binom{2\ell-1}{\ell}$.

We have successfully applied this recursive approach also to the cases of $\mathsf{M} = 5$ and $\mathsf{M} = 6$.

Theorem 52. *For a BEC and for any $n \geq 3$, if the optimal codebook can be recursively constructed in blocklength n, an optimal codebook with $\mathsf{M} = 5$ codewords can be constructed recursively in the blocklength n. We start with an optimal codebook for $n = 3$:*

$$\mathscr{C}_{\mathrm{BEC}}^{(\mathsf{M},3)*} = \left(\mathbf{c}_1^{(\mathsf{M})}, \mathbf{c}_2^{(\mathsf{M})}, \mathbf{c}_5^{(\mathsf{M})}\right) \tag{82}$$

and recursively construct the optimal codebook for $n \geq 5$ by using $\mathscr{C}_{\mathrm{BEC}}^{(\mathsf{M},n-\gamma)*}$, $\gamma \in \{1,2,3\}$, and appending

$$\begin{cases} \left(\mathbf{c}_1^{(\mathsf{M})}, \mathbf{c}_2^{(\mathsf{M})}, \mathbf{c}_5^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 3, \\ \left(\mathbf{c}_3^{(\mathsf{M})}, \mathbf{c}_6^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 5, \\ \left(\mathbf{c}_9^{(\mathsf{M})}, \mathbf{c}_{10}^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 7, \\ \left(\mathbf{c}_4^{(\mathsf{M})}, \mathbf{c}_7^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 9, \\ \mathbf{c}_8^{(\mathsf{M})} & \text{if } n \bmod 10 = 0. \end{cases} \tag{83}$$

For $\mathsf{M} = 6$ codewords, an optimal codebook can be constructed recursively in the blocklength n by starting with an optimal codebook for $n = 4$:

$$\mathscr{C}_{\mathrm{BEC}}^{(\mathsf{M},3)*} = \left(\mathbf{c}_1^{(\mathsf{M})}, \mathbf{c}_2^{(\mathsf{M})}, \mathbf{c}_6^{(\mathsf{M})}, \mathbf{c}_8^{(\mathsf{M})}\right). \tag{84}$$

Then we recursively construct the optimal codebook for $n \geq 6$ by using $\mathscr{C}_{\mathrm{BEC}}^{(\mathsf{M},n-2)*}$ and appending

$$\begin{cases} \left(\mathbf{c}_1^{(\mathsf{M})}, \mathbf{c}_2^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 2, \\ \left(\mathbf{c}_6^{(\mathsf{M})}, \mathbf{c}_8^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 4, \\ \left(\mathbf{c}_3^{(\mathsf{M})}, \mathbf{c}_5^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 6, \\ \left(\mathbf{c}_4^{(\mathsf{M})}, \mathbf{c}_7^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 8, \\ \left(\mathbf{c}_9^{(\mathsf{M})}, \mathbf{c}_{10}^{(\mathsf{M})}\right) & \text{if } n \bmod 10 = 0. \end{cases} \tag{85}$$

For space reasons we omit the proof and only remark once again that the ideas of the derivation follow the same ideas as shown above in Lemma 49 and Claim 51.

An interesting special case of Theorem 52 is as follows.

Theorem 53. *For a BEC and for any n being a multiple of 10, an optimal codebook with $\mathsf{M} = 5$ or $\mathsf{M} = 6$ codewords is the corresponding fair weak flip code.*

Note that the restriction on n comes from the restriction that fair weak flip codes are only defined for n with $n \bmod \binom{2\ell-1}{\ell} = n \bmod 10 = 0$. Even though Theorem 53 actually follows as special case from Theorem 52, it can be proven directly and more elegantly using the properties of fair weak flip codes derived in Section 7.1.

How about the optimal codes on BEC for higher number of codewords M? We strongly believe that Theorem 53 can be generalized to arbitrary M.

Conjecture 54. *For a BEC and for an arbitrary M, the optimal code for a blocklength n that satisfies $n \bmod \binom{2\ell-1}{\ell} = 0$ is the corresponding fair weak flip code.*

7.3 Quick Comparison between BSC and BEC

In [6] it has been shown that optimal codes for $\mathsf{M} = 3$ or $\mathsf{M} = 4$ are weak flip codes with code parameters:

$$[t_1^*, t_2^*, t_3^*] = \begin{cases} [k+1, k-1, k] & \text{if } n \bmod 3 = 0, \\ [k+1, k, k] & \text{if } n \bmod 3 = 1, \\ [k+1, k, k+1] & \text{if } n \bmod 3 = 2, \end{cases} \tag{86}$$

where we use

$$k \triangleq \left\lfloor \frac{n}{3} \right\rfloor. \tag{87}$$

The corresponding pairwise Hamming distance vectors (see Lemma 22) are

$$\begin{cases} (2k-1, 2k, 2k+1) & \text{if } n \bmod 3 = 0, \\ (2k, 2k+1, 2k+1) & \text{if } n \bmod 3 = 1, \\ (2k+1, 2k+2, 2k+1) & \text{if } n \bmod 3 = 2. \end{cases} \tag{88}$$

If we compare this to Theorem 48:

$$[t_1^*, t_2^*, t_3^*] = \begin{cases} [k, k, k] & \text{if } n \bmod 3 = 0, \\ [k+1, k, k] & \text{if } n \bmod 3 = 1, \\ [k+1, k, k+1] & \text{if } n \bmod 3 = 2 \end{cases} \tag{89}$$

with corresponding pairwise Hamming distance vectors

$$\begin{cases} (2k, 2k, 2k) & \text{if } n \bmod 3 = 0, \\ (2k, 2k+1, 2k+1) & \text{if } n \bmod 3 = 1, \\ (2k+1, 2k+2, 2k+1) & \text{if } n \bmod 3 = 2, \end{cases} \tag{90}$$

we can conclude the following.

Corollary 55. *Apart from $n \bmod 3 = 0$, the optimal codes for a BSC are identical to the optimal codes for a BEC for $\mathsf{M} = 3$ or $\mathsf{M} = 4$ codewords.*

It is interesting to note that for $n \bmod 3 = 0$ the optimal codes for the BEC are fair and therefore maximize the minimum Hamming distance, while this is not the case for the (very symmetric!) BSC. However, note that the converse is *not* true: if a code maximizes the minimum Hamming distance, then it is not necessarily an optimal code for the BEC! So, in particular, it is not clear if binary nonlinear Hadamard codes are optimal.

7.4 Application to Known Bounds on the Error Probability for a Finite Blocklength

We again provide a comparison between the performance of the optimal code to the known bounds of Sec. 6.

Note that the error exponents for $\mathsf{M} = 3, 4$ codewords are

$$\mathsf{E}_3 = \mathsf{E}_4 = -\frac{2}{3} \log \delta. \tag{91}$$

Moreover, for $\mathsf{M} = 3, 4$,

$$\mathsf{D}_{\min}^{(\mathrm{BEC})}\left(\mathscr{C}^{(\mathsf{M},n)}_{\lfloor\frac{n+1}{3}\rfloor,\lfloor\frac{n}{3}\rfloor}\right) = \begin{cases} -\frac{2}{3}\log\delta & \text{if } n \bmod 3 = 0 \\ -\frac{\lfloor\frac{n}{3}\rfloor+\lfloor\frac{n+1}{3}\rfloor}{n}\log\delta & \text{if } n \bmod 3 = 1 \\ -\frac{\lfloor\frac{n}{3}\rfloor+\lfloor\frac{n+1}{3}\rfloor}{n}\log\delta & \text{if } n \bmod 3 = 2. \end{cases} \tag{92}$$

Figs. 2 and 3 compare the exact optimal performance for $\mathsf{M} = 3$ and $\mathsf{M} = 4$, respectively, with some bounds: the SGB upper bound based on the weak flip code used by Shannon *et al.*,[7], the SGB lower bound based on the weak flip code (which is suboptimal, but achieves the optimal $\mathsf{D}_{\min}^{(\mathrm{DMC})}$ and is therefore a generally valid lower bound), the Gallager upper bound, and also the PPV upper and lower bounds.

We can see that the SGB upper bound is tighter to the exact optimal performance than the PPV upper bound. Note, however, the PPV upper bound does not exhibit the correct error exponent. It is shown in [12] that, for n going to infinity, the random coding (PPV) upper bound tends to the Gallager exponent for $R = 0$ [10], which is of course not necessarily equal to E_{M} for finite M.

Concerning the lower bounds, we see that the PPV lower bound (converse) is much better for finite n than the SGB bound. However, for n large enough, its exponential growth will approach that of the sphere-packing bound [8], which does not equal to E_{M} either.

Once more we would like to point out that even though the fair weak flip codes achieve the error exponent, they are optimal codes in the BEC, however, they are strictly suboptimal for every $n \bmod 3 = 0$ in the BSC.

8 Conclusion

In this work, we have introduced the *weak flip codes*, a new class of codes containing both the class of binary nonlinear Hadamard codes and the class of linear codes as special cases. We have shown that weak flip codes have many desirable properties; in particular, we have been able to prove that besides retaining many of the good Hamming distance properties of Hadamard codes, they are actually optimal with respect to the minimum error probability over a binary erasure channel (BEC) for certain numbers of codewords M and many finite blocklengths n.

We have also introduced the subclass of *fair weak flip codes* that can be seen as a generalization of linear codes to arbitrary numbers of codewords M. We have shown that, if the optimal codes can be constructed recursively in blocklength n, the fair weak flip codes are optimal with respect to the error probability for the BEC for $\mathsf{M} \leq 6$ and a blocklength that depends on M, and we have conjectured that this result continues to hold also for $\mathsf{M} > 6$.

Note that while it has been known for quite some time that binary nonlinear Hadamard codes have good Hamming distance properties [4], so far not much has been known about their behavior with respect to error probability for finite n. Furthermore, also note that while fair weak flip codes have been used before (although without being named) in the derivation of results related to error probability [8], so far it is only showed that the optimal error exponents can be achieved by fair weak

[7]The SGB upper bound based on the optimal code performs almost identically (because the BSC is pairwise reversible) and is therefore omitted.

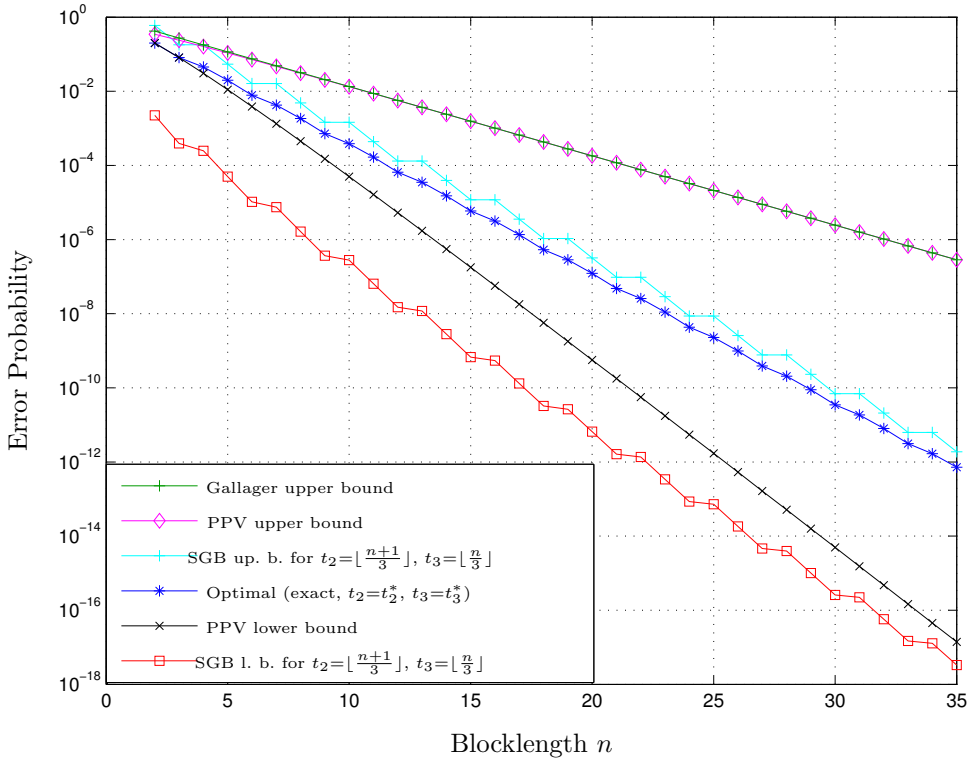

Figure 2: Exact value of, and bounds on, the performance of an optimal code with $\mathsf{M} = 3$ codewords on the BEC with $\delta = 0.3$ as a function of the blocklength n.

flip codes, but they have not been proven to be actually optimal in error probability among all possible linear or nonlinear codes for finite blocklength.

In conclusion, we try to build a bridge between coding theory, which usually is concerned with the design of codes with good Hamming distance properties (like, e.g., the binary nonlinear Hadamard codes), and information theory, which deals with error probability and the existence of codes that have good or optimal error probability behavior.

References

[1] Claude E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, July and October 1948.

[2] Shu Lin and Daniel J. Costello, Jr., *Error Control Coding*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2004.

[3] Chia-Lung Wu, Po-Ning Chen, Yunghsiang S. Han, and Yan-Xiu Zheng, "On the coding scheme for joint channel estimation and error correction over block fading channels," in *Proceedings IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Tokyo, Japan, September 13–16, 2009, pp. 1272–1276.

[4] F. Jessy MacWilliams and Neil J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1977.

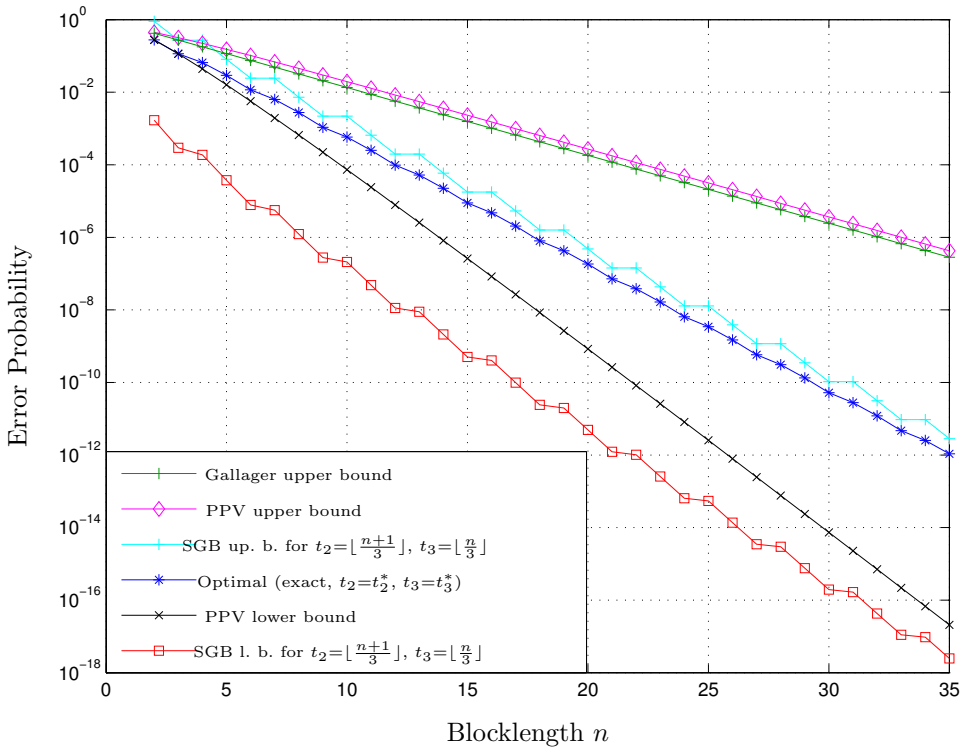

Figure 3: Exact value of, and bounds on, the performance of an optimal code with $\mathsf{M} = 4$ codewords on the BEC with $\delta = 0.3$ as a function of the blocklength n.

[5] Po-Ning Chen, Hsuan-Yin Lin, and Stefan M. Moser, "Ultra-small block-codes for binary discrete memoryless channels," in *Proceedings IEEE Information Theory Workshop (ITW)*, Paraty, Brazil, October 16–20, 2011, pp. 175–179.

[6] ——, "Optimal ultra-small block-codes for binary discrete memoryless channels," 2013, to appear in *IEEE Transactions on Information Theory*. [Online]. Available: http://moser.cm.nctu.edu.tw/publications.html

[7] Stefan M. Moser, *Information Theory (Lecture Notes)*, version 1, fall semester 2011/2012, Information Theory Lab, Department of Electrical Engineering, National Chiao Tung University (NCTU), September 2011. [Online]. Available: http://moser.cm.nctu.edu.tw/scripts.html

[8] Claude E. Shannon, Robert G. Gallager, and Elwyn R. Berlekamp, "Lower bounds to error probability for coding on discrete memoryless channels," *Information and Control*, pp. 522–552, May 1967, part II.

[9] Po-Ning Chen, Hsuan-Yin Lin, and Stefan M. Moser, "Equidistant codes meeting the Plotkin bound are not optimal on the binary symmetric channel," January 2013, submitted to *IEEE International Symposium on Information Theory (ISIT)*. [Online]. Available: http://moser.cm.nctu.edu.tw/publications.html

[10] Robert G. Gallager, *Information Theory and Reliable Communication*. New York: John Wiley & Sons, 1968.

[11] Yury Polyanskiy, H. Vincent Poor, and Sergio Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[12] Yury Polyanskiy, "Saddle point in the minimax converse for channel coding," 2013, to appear in *IEEE Transactions on Information Theory*.