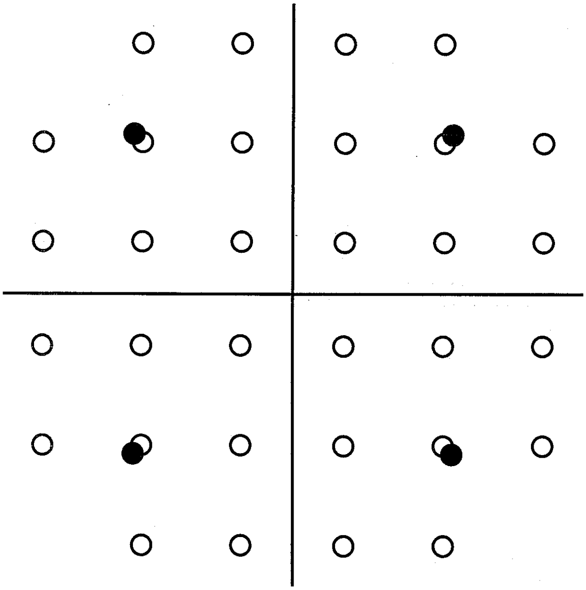
Coding strategies for channels with feedback

DOI:
10.6100/IR526152

Document status and date:
Published: 01/01/1999

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Coding Strategies for Channels with Feedback



Mortaza Shoae Bargh

Coding Strategies for Channels with Feedback

Coding Strategies for Channels with Feedback

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
Rector Magnificus, prof.dr. M. Rem, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op dinsdag 28 september 1999 om 16.00 uur

door

Mortaza Shoae Bargh

geboren te Tabriz, Iran

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. J.P.M. Schalkwijk
en
prof.dr. E.C. van der Meulen

Druk: Universiteitsdrukkerij Technische Universiteit Eindhoven

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Shoae Bargh, Mortaza.

Coding strategies for channels with feedback / by Mortaza Shoae Bargh. -
Eindhoven : Technische Universiteit Eindhoven, 1999.
Proefschrift. - ISBN 90-386-1670-8
NUGI 832
Trefw.: telecommunicatie ; codering / digitale modulatie / datatransmissie ; codes.
Subject headings: channel coding / modulation coding / Gaussian channels / telecommunication channels.

To my parents

and brothers

Summary

In communication systems there is often a feedback link from the receiver to the transmitter. For example, telephone connections are two-way channels. When one side speaks, the other listens and the corresponding channel from the listener to the speaker, i.e. the feedback channel, remains idle. Feedback strategies use this idle feedback channel in their coding process. Feedback generally enables these schemes to communicate over the forward channel with lower decoding complexity, higher reliability, higher capacity, or a combination of these advantages, compared to feedback-free communication schemes. In the so-called *information feedback* schemes the transmitter is able to observe every received symbol via a noiseless and delayless feedback channel. This enables the transmitter to correct every error that occurred in the forward channel. A class of simple and efficient information feedback schemes for Discrete Memoryless Channels (DMC's) is called Multiple Repetition Feedback Coding (MRFC) schemes, which forms the core of the presentation in this dissertation. First these schemes are applied to the channels with a noisy feedback channel. Then some modifications are made on MRFC schemes (with noiseless feedback) to improve their performance and to extend their application domain.

An important shortcoming of existing information feedback schemes is that they assume the feedback channel to be noiseless. In practical situations this is not always a realistic assumption. To deal with the problem of noisy feedback channels, we reduce the amount of information in the backward direction with respect to that in the forward direction. Therefore, this relatively low amount of information can be protected against the noise on the feedback channel. More specifically, a bandwidth efficient modulation method with MRFC schemes is designed for additive white Gaussian noise channels where both forward and feedback channels have the same Signal to Noise Ratio (SNR). The resulting method gives about 5.5 dB improvement over uncoded transmissions. In this method, the decoding complexity is considerably simplified compared to the decoding complexity of the trellis coded modulation. Given the fact that we need to transmit one bit per dimension in the backward direction, the feedback channel can essentially be considered noiseless for channels with moderate SNR's. The theoretical improvement of the proposed method is supported by running simulations.

In the remainder of this work, the scope of MRFC schemes is extended. There are block and recursive coding methods for MRFC. The performance of the block MRFC is improved by devising two block retransmission strategies. For asymmetric DMC's, one of the strategies even yields a superior performance compared to existing block and recursive coding methods. Furthermore, MRFC schemes are modified in order to be applicable to soft-output DMC's. A soft-output DMC, for example, can be obtained by quantizing the output of a continuous channel with discrete inputs. First a specific class of compound channels is used to define soft-output DMC's. For this class of compound channels, it is also shown that feedback and delayed side information at the transmitter cannot increase channel capacity. Then, the modification of MRFC schemes is presented for a two-input four-output (soft-output) DMC. Using a proper Markov source as the precoder, the scheme yields rates very close to channel capacity. Note that a MRFC scheme consists of a repetition part and a precoding part. For the repetition part, the block and recursive coding methods of the new scheme are described. Since the existing precoding methods cannot be used in the case of the new scheme, a precoding method is designed and evaluated at the end of the thesis.

Acknowledgements

I would like to express my sincere gratitude to my supervisor prof.dr.ir. J.P.M. Schalkwijk whose guidance, encouragement and support were absolutely essential to carry out my doctoral research.

I am very grateful to Frans Willems, Tjalling Tjalkens, Harry Creemers, Jan Rooyackers, Phons Bloemen, Thijs Veugen, Vladimir Balakirsky and specially to my roommates Erik Meeuwissen and Paul Volf. Special thanks go to Thijs, Vladimir and Paul for reading parts of the manuscript and for providing useful comments. Thijs's expertise and willingness to help led to many improvements in the thesis.

My acknowledgements go to the past and present members of the Radiocommunication Group for providing a nice and friendly working environment. I highly appreciate the friendship of dr. Madani and his family during the past five years. I would also like to express my gratitude to my teachers, classmates and friends in the EII/PII who have added a multicultural dimension to my stay in The Netherlands.

I am grateful to the the members of my thesis committee, prof.dr.ir. J. Biemond, prof.dr.ir. G. Brussaard, prof.dr.ing. H.J. Butterweck, prof.dr.ir. W.C. van Etten, prof.dr. E.C. van der Meulen, prof.dr.ir. J.P.M. Schalkwijk, prof.dr.ir. H.C.A. van Tilborg and dr.ir. F.M.J. Willems, for reviewing my thesis. I would also like to thank the members of the reading committee for their remarks and suggestions.

The research for this thesis was funded by the "Stan Ackermans Instituut" (SAI) and the Euler Institute for Discrete Mathematics and its Applications (EIDMA). I am deeply indebted to these institutes.

My apologies to those whom I may have inadvertently failed to mention. Last, but certainly not least, I am deeply indebted to my family who continuously supported me during my studies. Their endurance and encouragements are highly appreciated.

Eindhoven, August 16, 1999

Notation and abbreviations

$\mathbf{M} = [m_{ij}]$	Matrix with entries m_{ij}
$\mathbf{v}^N = (v_1, v_2, \dots, v_N)$	Vector
$\mathbf{v}^N = v_1 v_2 \dots v_N$	Sequence or block of symbols
$\mathbf{v}_n^{(n+N)} = v_n v_{n+1} \dots v_{n+N}$	
$\mathbf{v}_i^N = v_{i1} v_{i2} \dots v_{iN}$	The i^{th} block
$\mathbf{v} = v_1 v_2 \dots$	Sequence, unlimited length
$\mathbf{v} = \dots v_1 v_2 \dots$	
$v^n = vv \dots v$	Subsequence of n consecutive v's
$\mathbf{v}^N = v_1 v_2 \dots v_N$ $= (v_1, v_2, \dots, v_N)$	Code word
$\mathbf{v}_i^N = v_{i1} v_{i2} \dots v_{iN}$ $= (v_{i1}, v_{i2}, \dots, v_{iN})$	The i^{th} code word
$n = 1, \dots, N$	Counter or index, integer values from 1 to N
(x, y)	Interval of real values between x and y
(x, y), $(\mathcal{X}, \mathcal{Y})$	A pair of digits, sets
$\mathcal{X}$	Set
$\mathcal{X} = \{x_1, x_2, \dots, x_N\}$ $= \{x_n : n = 1, \dots, N\}$	Finite set having distinct elements
$\vert\mathcal{X}\vert$	Number of elements of set $\mathcal{X}$
$x \in \mathcal{X}$	x an element of set $\mathcal{X}$
$\mathbb{R}$	The set of real numbers
X (over $\mathcal{X}$)	Random variable, defined over set $\mathcal{X}$
$\mathbb{E}[X] = \bar{X}$	Expected value of a random variable
$\{X_i\}$	Random process
$\mathbf{X}^N = (X_1, X_2, \dots, X_N)$	Vector of random variables
$Pr\{A\}$	Probability of an event A

$Pr\{A\|B\}$	Probability of an event A conditioned on the event B (defined if $Pr\{B\} > 0$)

$*$ For discrete random variables

p_X	Distribution (or probability mass function) of $X : p_X(x) = Pr\{X = x\}$, $x \in \mathcal{X}$
$p_{X,Y}$	Joint distribution of X, Y : $p_{X,Y}(x, y) = Pr\{X = x, Y = y\}, x \in \mathcal{X}, y \in \mathcal{Y}$

$*$ For continuous random variables

f_X	Distribution (or probability density function) of $X : f_X(x)$, $x \in \mathcal{X}$
$f_{X,Y}$	Joint distribution of X, Y : $f_{X,Y}(x, y), x \in \mathcal{X}, y \in \mathcal{Y}$

ARQ: Automatic Repeat reQuest
AWGN: Additive White Gaussian Noise
BDC: Binary Duplex Channel
BEC: Binary Erasure Channel
BIQO: Binary-Input Quaternary-Output
BSC: Binary Symmetric Channel
COMC: Continuous Output Memoryless Channel
DMC: Discrete Memoryless Channel
  Input alphabet of a DMC: $\mathcal{X} = \{1, 2, \dots, |\mathcal{X}|\}$
  Output alphabet of a DMC: $\mathcal{Y} = \{1, 2, \dots, |\mathcal{Y}|\}$
  Probability transition matrix of a DMC channel: $\mathbf{P} = [p_{xy}]$, with entries $p_{xy} = p_{Y|X}(y|x)$, $x \in \mathcal{X}$ and $y \in \mathcal{Y}$
FCM: Feedback Coded Modulation
iid: independent and identically distributed
ISI: InterSymbol Interference
MRFC: Multiple Repetition Feedback Coding
MRFC scheme: MRFC with specific repetition parameters
MRFC system: to refer to all units of the MRFC as a system
psd: power spectral density
SNR: Signal to Noise Ratio
SRFC: MRFC for soft-output DMC's
TCM: Trellis Coded Modulation
□: to mark the end of an example or a remark
$\frac{N_o}{2}$: two-sided power spectral density of the AWGN random process

Contents

Chapter 1

An overview of feedback communication

In 1948, Shannon [67, 68] presented a brilliant mathematical model for the transmission of information. This pioneering work marked the beginning of information theory. Since then, information theory has grown in scope from communication theory into various scientific disciplines such as physics, statistics, economics, mathematics, computer science and etc. [16]. In what follows, we give a brief introduction to the principles of information theory in the framework of communication engineering. As we proceed, necessary definitions are given and, gradually, the course of the presentation is directed towards communication systems with feedback. Special attention will be given to the information feedback systems which are the main topic of this dissertation. In order to gain an insight into information feedback schemes, some of these schemes are explained from a new point of view which allows a unified explanation for many information feedback schemes.

1.1 A general view

A communication system is an arrangement which reliably transfers data from one point or points to another point or points. The transmitted data could be images, sound, files, documents, etc. The source and destination points can differ in location, e.g. the speaker and listener in a telephone conversation, and/or differ in time, e.g. the speaker and listener who use a telephone answering machine. Usually the goal of the system is to deliver at the destination(s) the exact data produced at the source(s). Sometimes, however, it is acceptable to deliver distorted data provided the distortion is acceptable at the destination.

The communication model of primary concern consists of two terminals A and B connected by channel K_{AB} in A to B direction and by channel K_{BA} in B to A direction. We can consider the following cases, see Fig. 1.1.

1. When the flow of information in one direction is independent from the flow of information in the other direction we have one-way communication from A to B or from B to A, see Fig. 1.1a. Most point to point communication systems are one-way systems, where channels K_{AB} and K_{BA} exist separately or one physical channel is used alternatively in either direction. Section 1.2 gives a broad view over one-way communication. Also in the course of this section, whenever it is necessary and appropriate, we will define those concepts of information theory which are essential for the presentation in the following sections or chapters.

2. When the flow of information in one direction interferes with the flow of information in the other direction we have two-way communication between A and B [7], see Fig. 1.1b. Two-way communication will be mentioned very briefly in Section 1.3.

3. When the flow of information is, let say, from A to B and channel K_{BA} is used to facilitate the flow of information in channel K_{AB}, we have feedback communication. In this case, K_{AB} and K_{BA} are called the forward channel and the feedback channel, respectively, see Fig. 1.1c. A general introduction to feedback communication is given in the second half of this chapter and the rest of the thesis covers the author's contributions to this area.

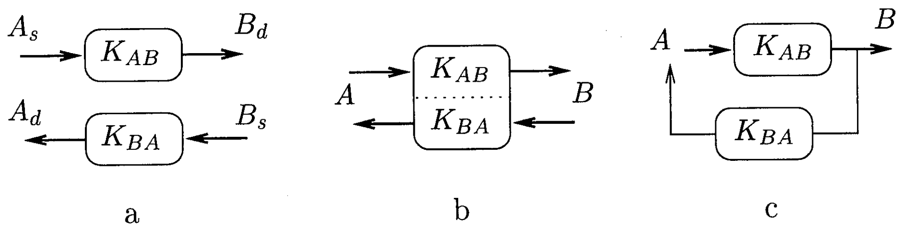


Figure 1.1: General point to point communication models; a: two one-way communication models; b: a two-way communication model; c: a feedback communication model.

Before going any further, let us draw an analogy to illustrate the subject. Assume in an hour's lecture, a tutor is supposed to teach some topics to a student. Considering the surrounding noises and student's ability of absorbing the material, the tutor must also provide the student with some extra explanations and examples in order to make sure that the student understands the topics. To achieve this goal, teacher's strategy might be to explain every lesson in several times, which would result in teaching only a few topics. Instead, the teacher could be greedy and give very short explanations. In this way, the number of topics could increase, but the student won't be able to understand some of the topics. On the other hand, an experienced tutor with a good knowledge of student's abilities and the surrounding distractions, can give efficient and sufficient explanations for the lessons. As a result, not only the number of conveyed topics per hour will be maximum, but also all topics will be understood. The disadvantage of the last solution is that it is very difficult to find such an experienced tutor. On the other hand, assume the case where the teacher is allowed to ask the student some questions in order to check whether a topic is properly understood. In this way, the tutor is able to revisit only those topics or parts of topics which have not been clearly understood. Note that the number of necessary explanations is not reduced by asking questions because the pupil does not become smarter nor disappear the surrounding noises. However, it is not necessary for the teacher to be an extremely sophisticated tutor if he is allowed to ask questions to the student. Moreover, the student learns easier if the teacher explains only those parts which are ambiguous.

One-way communication and feedback communication are analogous with such a tutorial session in which the tutor acts on his own instincts and on student's needs, respectively. In the following sections we will try to explain such communication systems from an information theory point of view.

1.2 One-way communication

One-way communication systems are designed to send information or messages from a source in point A, which generates these messages, to a destination in point B. The block diagram of the system is shown in Fig. 1.2. Physical sources produce different types of information which often contains a lot of redundancies. These redundancies can be removed using source coding techniques. Although source coding is an important part of the communication system and a substantial part of information theory deals with source coding techniques, we will assume that the output of the source encoder is the message to be transmitted via the channel. Therefore, we say that the imaginary source, shown in a dotted box in Fig. 1.2, produces message $w \in \mathcal{W} = \{1, 2, \dots, |\mathcal{W}|\}$ independent from previous messages with probability $p_W(w) = \frac{1}{|\mathcal{W}|}$, $w \in \mathcal{W}$. The remainder of the one-way communication system is described in the following subsections in detail.

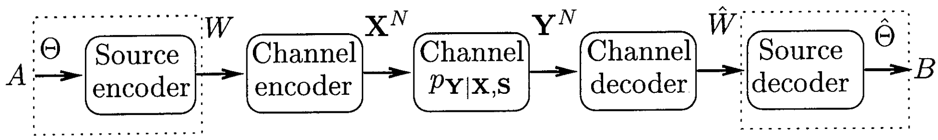


Figure 1.2: Block diagram of a one-way communication system.

1.2.1 Communication channels

The medium through which the information is transmitted is called transmission channel, or channel for short. A transmission channel can be specified in terms of the sets of input and output symbols available at the input and output terminals of the channel, i.e. $\mathcal{X}$ and $\mathcal{Y}$, respectively, and for each $x \in \mathcal{X}$, $y \in \mathcal{Y}$ and $s \in \mathcal{S}$ a conditional probability $p_{Y|X,S}(y|x,s)$, where $s \in \mathcal{S}$ represents the history of the channel up to that transmission or the state of the channel at that transmission. An ideal channel, for example, would be the one in which every output symbol uniquely specifies the input symbol. Channels, in general, change the input symbols to the output symbols such that uncertainty is produced at the output, in other words, they introduce *noise* to the input.

The input or output alphabets, i.e. $\mathcal{X}$ and $\mathcal{Y}$, can be finite or infinite sets. All channels considered in this dissertation are discrete time channels, i.e. the channel is used at discrete time instances. Discrete time channels can either exhibit memory or be memoryless. In a channel with memory the output at a given time depends statistically both on the current input and on the current state of the channel. The state of the channel, in turn, can generally depend on previous states, inputs and outputs as well as on the current input. Unless stated otherwise, we will assume that the channels are memoryless throughout the dissertation, i.e. $p_{Y|X,S} = p_{Y|X}$. More specifically, let $\mathbf{x}^N = x_1x_2\dots x_N$ be a sequence of N inputs to the channel and $\mathbf{y}^N = y_1y_2\dots y_N$ be the corresponding output sequence.

Definition 1.1 (Memoryless channels) *In a memoryless channel the output at a given time depends statistically only on the corresponding input. For the N^{th} extension of the memoryless channel we have*

$$p_{\mathbf{Y}^N|\mathbf{X}^N}(\mathbf{y}^N|\mathbf{x}^N) = \prod_{n=1}^{N} p_{Y|X}(y_n|x_n). \tag{1.1}$$

Definition 1.2 (Discrete Memoryless Channel (DMC)) *A memoryless channel is called a DMC when the input and output alphabets of the channel are finite sets. Matrix* $\mathbf{P}[p_{xy}]$*, where entries* $p_{xy} = p_{Y|X}(y|x)$ *for* $x \in \mathcal{X}$, $y \in \mathcal{Y}$*, is referred to as the probability transition matrix of the channel.*

Remark 1.1 We will also consider Continuous Output Memoryless Channels (COMC) where the output alphabet of the memoryless channel, $\mathcal{Y}$, is a subset of real numbers and the probabilistic relation between input and output is given by the conditional density function $f_{Y|X}(y|x)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$.

In [25], Gallager uses a DMC to define basic concepts of information theory. To keep the dissertation self-contained, some of these concepts are similarly defined in the next subsection, where a few related definitions are also given.

Some basic concepts of information theory

The input and output symbols of a DMC, say x and y, can be considered as the realizations of two random variables X and Y, drawn from finite sample spaces of $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let p_X, p_Y and $p_{X,Y}$ denote the distributions of random variables X and Y and their joint distribution, respectively. We want a quantitative measure of how much the occurrence of a particular output, say y, tells us about the possibility of a symbol, say x, to be the input to the channel. In other words, at the receiver, the occurrence of y changes the probability of x from the a-priori probability of $p_X(x)$ to the a-posteriori probability of $p_{X|Y}(x|y)$. In this case, the mentioned measure of the conveyed information is defined as follows.

Definition 1.3 (Mutual information) *The mutual information between events* $X = x$ *and* $Y = y$ *in the DMC, i.e. the amount of information provided about the input symbol* x *by the occurrence of the output symbol* y*, is defined as*

$$I(X = x; Y = y) = \log \frac{p_{X|Y}(x|y)}{p_X(x)} \overset{(\alpha)}{=} \log \frac{p_{Y|X}(y|x)}{p_Y(y)}, \tag{1.2}$$

where (α) *follows from the definition of conditional probability*[1].

[1] In this definition (and those of the self information and conditional self information) we assume that the probabilities involved are positive. In other cases, this assumption is not necessary due to the convention that $0 \log 0 = 0$ and $0 \log \frac{0}{0} = 0$.

From now on the assumption is that the base of all logarithms is 2, unless stated otherwise. Let us pose this question: what is the average amount of information received about X? With probability $p_{X,Y}(x,y)$ we transmit and receive symbol pair (x,y) in which the mutual information is equal to (1.2). Considering all input and output symbols, the answer is as follows.

Definition 1.4 (Average mutual information) *The average mutual information between the input and output of the DMC is given by*

$$\begin{aligned} I(X;Y) &= \sum_{x\in\mathcal{X}} p_X(x) \sum_{y\in\mathcal{Y}} p_{Y|X}(y|x) \log \frac{p_{Y|X}(y|x)}{p_Y(y)} \\ &= \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{X,Y}(x,y) \log \frac{p_{Y|X}(y|x)}{\sum_{x\in\mathcal{X}} p_X(x) p_{Y|X}(y|x)}. \end{aligned} \quad (1.3)$$

When the occurrence of a given output symbol y uniquely specifies the input symbol to be x, i.e. $p_{X|Y}(x|y) = 1$, the amount of information provided about the input symbol x by occurrence of the output symbol y is the amount of information that has already existed in the occurrence of event $X = x$.

Definition 1.5 (Self information) *The amount of information that we obtain if we know that $X = x$ is the self information of the event $X = x$ and it is defined as*

$$I(X = x) = \log \frac{1}{p_X(x)}.$$

This can also be interpreted as the a-priori uncertainty of the event $X = x$, or the amount of information required to ascertain that $X = x$.

Definition 1.6 (Entropy) *The average value of the self information of a random variable X is called the entropy of the random variable and it is defined as*

$$H(X) = \sum_{x\in\mathcal{X}} p_X(x) \log \frac{1}{p_X(x)}.$$

If we have a sequence of N realizations of the random variable X, then with arbitrarily small error probability it is possible to compress it, at most, into $NH(X)$ bits by making N sufficiently large [25, pp. 43]. Another concept is the *conditional self information* of event $X = x$ given the occurrence of $Y = y$. It is defined as

$$I(X = x|Y = y) = \log \frac{1}{p_{X|Y}(x|y)}.$$

This is the amount of information about x that we **do not know** when y is observed. Note that $I(X = x; Y = y)$ is that amount of information about x that we **do know** when $Y = y$ is observed.

Definition 1.7 (Conditional entropy) *The conditional entropy is the average of conditional self information, i.e.*

$$H(X|Y) = \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} p_{X,Y}(x, y) \log \frac{1}{p_{X|Y}(x|y)}.$$

In summary, $H(X)$ is the average amount of information that exists in the random variable X, $I(X;Y)$ is the average amount of information that we can get about X by knowing Y and $H(X|Y)$ is the average amount of information about X that we can **not** obtain by knowing Y. From this statement the following relation can be written.

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X),$$

where the last relation follows from similar arguments. Consider two probability mass functions p_X and q_X defined on $x \in \mathcal{X}$. A measure to indicate a distance between the two distributions is the relative entropy which is frequently used in information theory texts.

Definition 1.8 (Relative entropy) *The relative entropy or the Kullback Leibler distance between two probability mass functions* p_X *and* q_X *is defined as*

$$D(p \parallel q) = \sum_{x\in\mathcal{X}} p_X(x) \log \frac{p_X(x)}{q_X(x)}.$$

We can also define the above concepts for continuous random variables[2]. For example, consider the following two definitions.

[2]For the definition of a continuous random variable refer to [16, pp. 224].

Definition 1.9 (Differential entropy) *The differential entropy of a continuous random variable with probability density function f_X is defined as*

$$H(X) = -\int_{\mathcal{X}} f_X(x) \log f_X(x) dx.$$

Here $\mathcal{X}$ is the support set of continuous random variable X.

Definition 1.10 ((Average) mutual information) *The (average) mutual information between two continuous random variables X and Y with joint probability density function $f_{X,Y} = f_X \cdot f_{Y|X}$ is defined as*

$$I(X;Y) = \int_{\mathcal{X}} \int_{\mathcal{Y}} f_{X,Y}(x,y) \log \frac{f_{Y|X}(y|x)}{f_Y(y)} dxdy.$$

Now let us consider a sequence of random variables that are drawn from a finite sample space. As a general example, assume the output symbols of a source to be a stochastic process designated by $\{\Theta_i\}$. Similar to Def. 1.6, for a sequence of n random variables $\Theta^n = (\Theta_1, \Theta_2, \ldots, \Theta_n)$, the entropy per symbol can be defined as

$$\frac{H(\Theta_1, \Theta_2, \ldots, \Theta_n)}{n} = \tag{1.4}$$
$$\frac{1}{n} \sum_{\theta_1, \ldots, \theta_n} Pr\{\Theta^n = (\theta_1, \ldots, \theta_n)\} \log(\frac{1}{Pr\{\Theta^n = (\theta_1, \ldots, \theta_n)\}}).$$

By letting $n \to \infty$, the entropy per symbol of a stochastic process can be defined.

Definition 1.11 (Entropy rate) *The entropy rate of the stochastic process $\{\Theta_i\}$ is defined by*

$$H_\infty(\Theta) = \lim_{n \to \infty} \frac{1}{n} H(\Theta_1, \Theta_2, \ldots, \Theta_n),$$

under the assumption that the limit exists. If the stochastic process is stationary, the above limit exists and it can be written as $H_\infty(\Theta) = \lim_{n \to \infty} H(\Theta_n | \Theta^{(n-1)})$[3].

[3] For the proof refer to [16, pp. 64].

Being stationary means that the joint probability distribution of any sequence of source output is invariant with respect to shifts in time. The concept of entropy rate for a sequence of independent and identically distributed (iid) random variables reduces to Def. 1.6. For such variables, the entropy rate is $H_\infty(\Theta) = \frac{nH(\Theta_i)}{n} = H(\Theta_i)$, i.e. the entropy rate equals to the entropy of a single random variable.

1.2.2 Channel encoding and decoding

In a DMC, by *channel encoding* each message of the (imaginary) source, i.e. $w \in \mathcal{W} = \{1, 2, \ldots, |\mathcal{W}|\}$, is mapped into a sequence of N channel input symbols $\mathbf{x}^N = f_e(w) = (x_1, x_2, \ldots, x_N)$, where $x_n \in \mathcal{X}$, $n = 1, \ldots, N$, and function $f_e : \mathcal{W} \to \mathcal{X}^N$ represents the encoding rule. $\mathbf{x}^N$ is called the *code word* of message w. In total we will have $|\mathcal{W}|$ equiprobable code words each of length N, therefore, this code is called a $(|\mathcal{W}|, N)$ block code. The *rate* of this code is the amount of information per transmission, i.e.

$$R = \frac{\log |\mathcal{W}|}{N} \qquad \text{bits per transmission.}$$

By *channel decoding* an estimate $\hat{w}$ from the set of messages is assigned to each received sequence $\mathbf{y}^N = (y_1, y_2, \ldots, y_N)$ according to $\hat{w} = f_d(\mathbf{y}^N)$, where $f_d : \mathcal{Y}^N \to \mathcal{W}$ is the decoding function. By mapping w to its code word $\mathbf{x}^N$, we add some redundancy to the message. This redundancy protects the message from the channel noise (up to a degree) by helping the decoder to recover the transmitted message from the distorted code word $\mathbf{y}^N$, i.e. by increasing the probability that $\hat{W} = w$. Note that the term of **coding** will be used to refer to the combination of the encoding and decoding procedures.

An *error* occurs when the output of the decoder is not the same as the transmitted message, i.e. $\hat{w} \neq w$. The decoding error probability for message w is

$$\lambda_w = Pr\{\hat{W} \neq w \mid \mathbf{X}^N = f_c(w)\}$$

and the average and the maximum decoding error probabilities are respectively defined as

$$P_{e,max}^N = \max_{w \in \mathcal{W}} \lambda_w,$$

$$P_{e,av}^N = \sum_{w=1}^{|\mathcal{W}|} p_W(w) \lambda_w = \frac{1}{|\mathcal{W}|} \sum_{w=1}^{|\mathcal{W}|} \lambda_w.$$

Definition 1.12 (Achievable rate) *A rate R is said to be achievable if there exists a sequence of* $(\lceil 2^{NR} \rceil, N)$ *codes such that the maximum*[4] *probability of error* $P_{e,max}^N$ *tends to 0 as* $N \to \infty$ *[16].*

Sequential coding schemes are other coding methods that differ from block coding methods mentioned above. A sequential encoder, continuously maps a sequence of message symbols into input symbols of the channel. The decoder, on the other side, decodes every transmitted symbol after a delay of D channel transmissions. For example, convolutional codes are sequential codes. For sequential codes constraint length plays a role analogous to the block length of block codes.

Definition 1.13 (Constraint length) *The constraint length of a sequential code is the maximal number of channel symbols which are affected by a single message symbol*[5].

1.2.3 Channel capacity

At first sight, it seems reasonable to think that increasing the number of messages in a block of N transmissions, i.e. increasing the transmission rate, increases the decoding error probability. With reference to our tutor-pupil example, if the teacher cramps more material in this hour of teaching, he increases the student's confusion. Shannon in 1948 showed that the decoding error probability can still be made arbitrarily small for a DMC, as long as the transmission rate remains below channel capacity. Therefore, the (operational channel) capacity of a DMC is defined as *the supremum of all achievable rates on the channel.* First let us define the information theoretic capacity of DMC's.

Definition 1.14 (Information theoretic capacity of DMC's) *The information theoretic capacity, C, of a DMC is defined as*

$$C = \max_{p_X} I(X;Y), \tag{1.5}$$

where the maximum is taken over all possible input distributions p_X, where $\Sigma_{x \in \mathcal{X}} p_X(x) = 1$.

The following theorem shows that the (operational channel) capacity of the DMC is equal to (1.5).

[4] It can be shown that a small average error probability implies a small maximum error probability at the same rate, see [16, pp. 194].

[5] For example, see [25, pp. 263] where the constraint length is defined for convolutional codes. Note that for a sequential feedback code in [29] the average constraint length is considered.

Theorem 1.1 (Channel coding theorem for DMC's) *All rates below C are achievable. Specifically, for all $R < C$, there exists a sequence of $(\lceil 2^{NR} \rceil, N)$ codes with maximal probability of error $P_{e,max}^N \to 0$ as $N \to \infty$. Conversely, any sequence of $(\lceil 2^{NR} \rceil, N)$ codes with $P_{e,max}^N \to 0$ must have $R \leq C$.*

Proof : See [16]. Q.E.D.

Due to Theorem 1.1, C is mostly referred to as channel capacity for short. Similarly, capacity can also be determined for a continuous alphabet channel such as an Additive White Gaussian Noise (AWGN) channel. This channel is an important channel model for many applications. The output random variable, Y, of an AWGN channel can be written as

$$Y = X + Z, \qquad Z \sim N(0, \sigma^2).$$

Like in DMC's, a block code for the AWGN channel maps message $w \in \mathcal{W}$ into a sequence of N channel input symbols $\mathbf{x}^N = (x_1, x_2, \ldots, x_N)$, where x_n is a real number, $n = 1, \ldots, N$. Each code word satisfies the average energy per symbol constraint, i.e. $\frac{1}{N}\sum_{n=1}^{N} x_n^2 \leq E_s$.

Definition 1.15 (Information theoretic capacity of AWGN channels) *The capacity*[6] *of an AWGN channel with the average energy constraint E_s is*

$$C = \max_{f_X : \mathbb{E}[X^2] \leq E_s} I(X;Y) = \frac{1}{2}\log(1 + \frac{E_s}{\sigma^2}), \qquad \textit{bits per transmission.}$$

The Nyquist-Shannon sampling theorem, see [16, pp. 248], shows that a band-limited signal has only $2W$ degrees of freedom per second, where W is the bandwidth of the signal. In other words, the signal can have $2W$ independent samples per second. Therefore, the average energy per signal sample is equal to the signal power P divided by $2W$. The average energy per noise sample becomes $\sigma^2 = \frac{N_o}{2}$, where $\frac{N_o}{2}$ is the two-sided power spectral density of the AWGN random process.

Remark 1.2 (Capacity of band limited AWGN channels) The capacity of a band limited AWGN channel with power constraint P is

$$C = W \log(1 + \frac{P}{WN_o}), \qquad \text{bits per second.} \ \square$$

[6] Similar to Theorem 1.1, it can be shown that the channel capacity of AWGN channels is equal to the information theoretic capacity.

Remark 1.3 (Capacity of wide band AWGN channels) The capacity of a wide band AWGN channel with power constraint P is

$$C = \lim_{W \to \infty} W \log(1 + \frac{P}{WN_o}) = \frac{P}{N_o} \log e, \qquad \text{bits per second.} \ \Box$$

1.2.4 Reliability function

Shannon's channel coding theorem is not constructive, i.e. it does not give implementable codes. Since the beginning of information theory, many coding methods have been introduced such as block codes, convolutional codes and, recently, Turbo codes. An important aspect of a code is the rate at which $P_{e,av}^N$ vanishes with increasing $N \to \infty$. Elias [21], Shannon [73] and others showed that for the best coding schemes the convergence is exponential in the block length.

Definition 1.16 (Error exponent) *For a specific coding scheme the error exponent at a given rate is defined as*

$$E = \lim_{N \to \infty} \frac{-\log P_{e,av}^N}{N} \tag{1.6}$$

assuming that the limit exists[7].

Note that in sequential schemes, the constraint length replaces the block length, N, in (1.6). In some coding schemes which map messages into variable length code words the average block (or constraint) length $\bar{N} = \mathbb{E}[N]$ is used in the above definition, see [29, 66, 22].

Normally, the error exponent is defined in terms of the transmission rate, i.e. $E(R)$, and it is referred to as the *reliability function.* The reliability function is also a measure for the complexity of the coding scheme. To wit, a given error probability approximately corresponds to a specific value for $N \cdot E(R)$ and the complexity of the decoding usually increases sharply with N. In the first chapter of [91], Veugen gives a survey about the reliability functions of different coding methods.

1.3 Two-way communication

Another kind of point-to-point communication occurs in two-way channels. Nowadays many practical channels, such as telephone channels, are intrinsically two-way channels. So far all designers have broken a two-way channel

[7] Generally, $\limsup_{N\to\infty} \cdots$ is considered in (1.6) to avoid the conceptual possibility that the limit might not exist. We used the $\lim_{N\to\infty} \cdots$ notation because the limit exists for the cases considered in this dissertation.

into two one-way channels and used one-way codes in each direction. This seems reasonable if the transmission in one direction does not interfere with the transmission in the other direction and messages in both ends are statistically independent (even though another solution can be considered for this situation, for example see Subsection 7.3.1). A general two-way channel where transmissions in both directions interfere with each other was first studied by Shannon [72] in 1961. Since then, many authors have worked on this subject, see for example [32], [62, 87, 8, 40]. The description of two-way communication is not our intention here and the interested reader is referred to the mentioned references, specially to the first and second chapters of [40] for an overview on two-way communication.

1.4 Feedback communication

In some two-way communication systems the flow of information is often in one direction, i.e. in the forward channel, and there is no transmission in the other direction. For example, in two-way channels of telephone networks when one side speaks, the other listens and the corresponding channel from the listener to the speaker remains idle. This kind of channel is called a semi-duplex channel. The inactive channel can be used as a feedback channel through which the transmitter obtains some information about the forward transmissions. The feedback information could be, for example, the status of the decoder of the forward channel, the received symbols of the forward channel or the state of the forward channel during past transmissions. Feedback coding strategies use the feedback information in their coding process. Throughout the thesis, we will assume the feedback channel to be noiseless and delayless, unless specifically stated otherwise.[8]

1.4.1 DMC's with noiseless feedback

Assume in addition to a DMC in the forward direction, there is a feedback link from the receiver to the transmitter. As shown in Fig. 1.3, even when the feedback channel is noiseless and delayless and the transmitter can observe every received symbol via the feedback channel, i.e. the so called **complete** information feedback, the capacity of a DMC does not increase by feedback. This was shown by Shannon in [69].

[8]Chapters 3 will address the noisy feedback case.

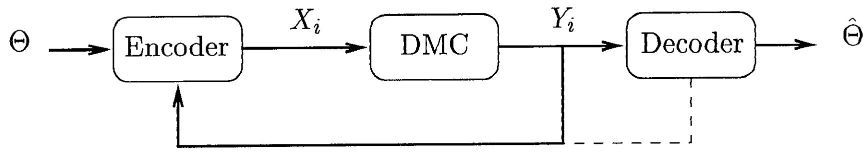


Figure 1.3: A noiseless feedback communication system, the dashed line schematically represents the decision feedback method.

Theorem 1.2 (Feedback capacity of DMC's) *Let C_{FB} and C denote the capacity of a DMC with and without feedback, respectively. Then*

$$C_{FB} = C = \max_{p_X} I(X;Y).$$

Proof : See [16] (or see Chapter 5 for a more general proof). Q.E.D.

Thus, in DMC's we cannot achieve any higher rates with feedback than we can without feedback (this also applies to other memoryless channels). However, feedback can help enormously in simplifying the coding process. For example, consider a binary erasure channel with erasure probability ϵ, see Fig. 1.4. If a feedback link exists, erasure outputs can be observed by the transmitter and the corresponding input bits can be repeated. By ignoring all erasure symbols in the received sequence, the receiver can obtain the transmitted message bits. In such a scheme, the input bit is repeated with probability ϵ, therefore, the effective transmission rate is $1-\epsilon$ bits per transmission, which coincides with channel capacity.

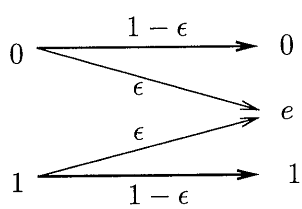


Figure 1.4: A binary erasure channel.

Coding schemes based on noiseless feedback can be categorized according to the type of information provided via the feedback channel. Distinction is made between decision feedback schemes and information feedback schemes, as it will be explained in the subsequent sections. However, let first outline three decoding methods of one-way block codes, namely: list, erasure and maximum likelihood decoding methods. The erasure decoding principle will be used to broadly explain the decision feedback methods. The list and erasure decoding principles will be used to explain the information feedback methods in Section 1.6.

1.4.2 List, erasure and maximum likelihood decoding

In [23] Forney considered the decoding problem of block codes for DMC's. In this paper any decoding rule $f_d(\cdot)$ for a $(|\mathcal{W}|, N)$ block code is characterized in terms of decision regions R_w, $w = 1, \ldots, |\mathcal{W}|$, defined over the space of received words $\mathcal{Y}^N$. If received code word $\mathbf{y}^N$ belongs to the region R_w, then the decoder puts out w, the message corresponding to the region, as an estimate of the transmitted message. Fig. 1.5 schematically shows three forms for the decision regions. In Fig. 1.5a decision regions do not overlap, but they cover the whole space of received sequences. This case occurs in the one-way systems where, for example, maximum a-posteriori probability decoding or maximum likelihood decoding is used. Here a decoding error occurs when the received sequence falls in a wrong decision region.

Fig. 1.5b shows the decision regions corresponding to erasure decoding. In erasure decoding the decision regions are also disjoint, but they do not cover the entire space $\mathcal{Y}^N$, i.e. some received words do not belong to any decision region. When a received sequence falls outside all regions, an erasure is declared. A decoding error occurs when the received sequence falls in a wrong decision region. Since decision regions are farther apart, the decoding error probability is, in general, less than that of the maximum likelihood decoding.

Finally, the decision regions of list decoding are shown in Fig. 1.5c. Here these regions are not disjoint, but they cover the whole space of received vectors. Since a received vector can belong to more than one decision region, a list decoder puts out more than one message candidate for the received vector. A decoding error occurs when the transmitted message is not on the decoded list of messages. In [23] Forney considers different applications for erasure and list decoding. For example, in systems with redundant data, in concatenated coding and in feedback schemes. In the following sections we explain the traces of erasure and list decoding principles in decision and information feedback schemes. Note that Forney did not elaborate on the relation between list decoding and feedback schemes as we do in Section 1.6.

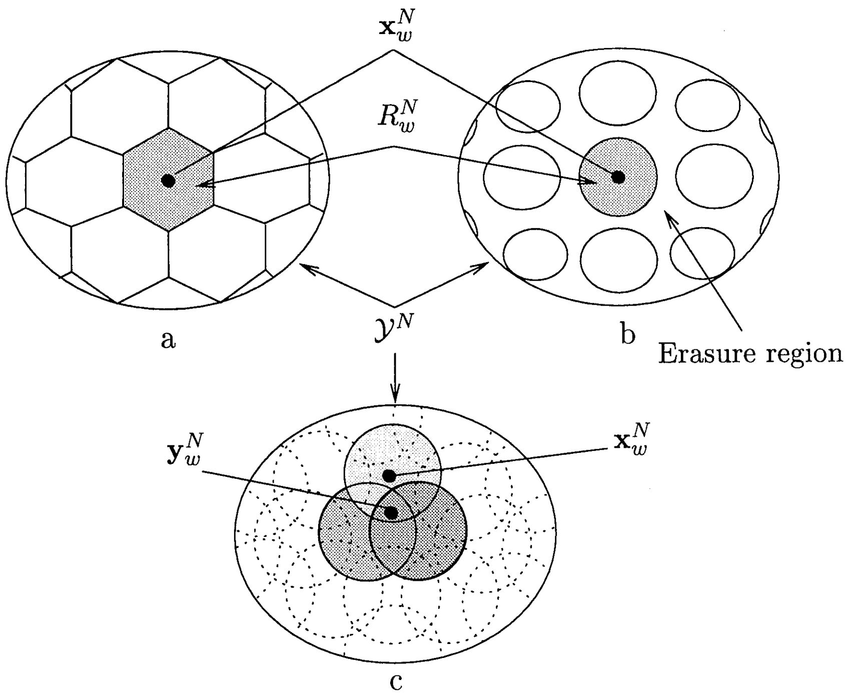


Figure 1.5: Models for decision regions in a: maximum likelihood decoding; b: erasure decoding; c: list decoding.

1.5 Decision feedback

In decision feedback schemes the feedback channel is used only to report the decision of the receiver that the received message is either accepted or rejected. In Fig. 1.4 a dashed feedback link schematically shows a decision feedback channel. Like in one-way communication, in decision feedback schemes a code is used to send the message through the forward channel. After receiving a block of symbols (or more generally, after receiving a specified number of symbols), the decoder either accepts or rejects the block based on its decoding criteria. Thereafter, the receiver indicates its acknowledgement or rejection to the transmitter via the feedback channel by a single yes-no message (one bit). Then, accordingly, the transmitter either moves on to the next message or repeats the same one. That is why a coding system based on decision feedback is also called the Automatic Repeat reQuest (ARQ) system.[9] Since only one bit on the return path is required for each block of symbols on the forward path, an ARQ system can be used even with noisy feedback channels. In fact, if the transmission rate of the forward channel is sufficiently higher than that of the feedback channel, then there is a greater effective Signal to Noise Ratio (SNR) in the feedback channel. Hence, a strong low-rate error correcting code can be used in the feedback direction and, consequently, the noisy feedback may for practical purposes be essentially error-free.

The decoding of ARQ systems based on block codes is similar to Forney's erasure decoding option. If the decoder sees that the received code word $\mathbf{y}^N$ is inside a region, it acknowledges the reception and if it detects that $\mathbf{y}^N$ is outside all regions, then it rejects that block and asks for a retransmission or for sending some extra information. This method dramatically reduces the probability of wrong decoding (which occurs if $\mathbf{y}^N$ falls in a wrong region), while the reduction of rate due to retransmissions is negligible. Forney showed that the error exponent of block ARQ methods approaches zero as the rate approaches capacity, similar to that of one-way block codes. However, the slope of the error exponent near capacity for ARQ methods is -1 compared to 0 for one-way block codes. This implies that the ratio of the error exponent of the decision feedback methods to that of one-way block codes becomes infinite as the rate approaches capacity.

The ARQ scheme can also be implemented using convolutional codes. A convolutional code has a better error exponent than a block code, particularly at high rates. Moreover, the decoding of convolutional codes is a sequential process, hence a retransmission can be requested before reaching the end of a code word or message. In this way, variable length ARQ schemes are possible which can reduce the rate loss due to retransmissions. Convolutional codes can be decoded by the Viterbi algorithm or the Fano algorithm and both decoding methods can be used in the ARQ systems. Refs. [22] and [100] are examples of ARQ schemes based on Viterbi decoding, for which the error exponent is asymptotically doubled in comparison with one-way convolutional codes. Examples of ARQ schemes based on the Fano algorithm are given in [20].

Since decision feedback is not the subject of this dissertation, the interested reader is referred to [91] for an overview and to references therein for more detail. Ref. [41] provides a balanced blend of theory and application about ARQ schemes.

---

[9] In [28] the decision feedback is referred to as post-decision feedback, i.e. feeding back the data derived from one or a sequence of decision results.

1.6 Information feedback

In information feedback, the transmitter is able to observe every received symbol via a noiseless and delayless feedback channel and use this information in the coding process. Here the information supplied to the transmitter is about the channel noise itself. Therefore, the information feedback is referred to as pre-decision feedback [28]. Our work is mostly concerned with information feedback schemes for DMC's and AWGN channels. Thus, we shall examine a few information feedback schemes for these channels from a new point of view, see also [81]. Note that our objective is to obtain a general insight into such information feedback schemes not to exactly evaluate them.

In fact, *many information feedback schemes yield highly reliable communication by efficiently repeating messages or message chips (i.e. message segments).* Even in one-way codes, repetition can improve the error performance or, in other words, lower the coding complexity for the same error performance. For example, consider the situation where every code word of a $(|\mathcal{W}|, N)$ block code with maximum decoding error probability $P_{e,max}^{N}$ is repeated L times. Based on a majority decision rule, an error might occur when half or more of the blocks are received erroneously. The decoding error can be upper bounded by $g_L(P_{e,max}^{N})^{\frac{L}{2}}$, where $g_L = \frac{L}{2}2^L$ and $L \ll N$. This shows that the decoded messages become more reliable as L increases. However, since the mentioned repetition mechanism is not efficient, the transmission rate also falls, accordingly.

With reference to the tutor-pupil example, repetition in one-way coding schemes as opposed to repetition in feedback coding schemes is analogous to a teacher who always gives several alternative explanations of a subject and one who gives an alternative explanation *only when* necessary, respectively. We distinguish two efficient repetition mechanisms which prevent any rate loss in most of information feedback schemes. These two mechanisms are referred to as "repeat to resolve uncertainty" and "repeat to correct erroneous receptions" in this dissertation. Generally speaking, an encoder using the first mechanism knows **what** to repeat and that of the latter one knows **when** to repeat.

1.6.1 Repeat to resolve uncertainty

In the schemes using the first mechanism, the transmitter sends the entire message in the initial transmission. Then, from the second transmission onwards, the transmitter sends corrections needed to resolve the *difference* between the true message and the receiver's latest estimate of this message. Note that the transmitted difference is sufficient to completely specify the true message provided that the difference is received correctly. Based on data received previously, the receiver has a *list* (or distribution) of candidates for the true message. Every transmitted correction can be interpreted as the index of the true message on receiver's list. As the transmission process continues, the size of the list decreases or, in other words, the correcting data is compressed relative to the previous transmissions. In this way, the scheme saves on transmission energy or, alternatively, on the number of transmissions. In the following, we explain a few schemes for DMC's and AWGN channels that operate based on this principle.

Savage's scheme

Savage [51] presented a block coding strategy which explicitly uses the concept of the list coding. The strategy for transmitting each message is a two-step procedure. For each step a one-way block code is used. In the first step, a block code is used to transmit one of the $|\mathcal{W}|$ messages. The decoder receives a distorted version of the code word and makes a list of L messages which are the most likely messages given the received code word, i.e. a list decoding of the first block code is performed. Then the feedback channel is used to inform the transmitter about the list and the order in which the messages appear on the list. In the second step, i.e. the first repetition of the message, the transmitter chooses a code word from the second code to indicate which message on the list of the L messages is the true message. For detailed description of the scheme and its performance the reader is referred to the original paper, however, we would like to draw attention to the efficient repetition mechanism employed in the second step. Instead of retransmitting one out of $|\mathcal{W}|$ messages, the scheme transmits one out of L entries on the list, where $L \ll |\mathcal{W}|$. One may argue that Savage's scheme must be considered a decision feedback scheme since the list is obtained after processing the received information. Nevertheless, we present Savage's scheme here because firstly the feedback link is used heavily in this scheme and secondly it provides an introduction to the following methods.

Ooi's scheme

In [48, 49], James Ooi used Ahlswede's idea employed in a constructive proof of the coding theorem for DMC's with feedback [1], and introduced the so called compressed-error-cancellation framework for transmitting information over DMC's with feedback. Ooi's scheme is a low complexity capacity achieving feedback scheme for DMC's though it gives a good performance for discrete finite state channels, discrete memoryless MAC's[10] and for channels that vary

[10] MAC stands for the Multiple Access Channel which is a channel with two or more transmitters and one receiver.

with time in an unknown way over a set of possible channels, see [47].

Assume a DMC with complete information feedback and capacity achieving input distribution q_X. Message $w \in \mathcal{W} = \{0,1\}^N$ is mapped into $N_1 = \frac{N}{H(X)}$ iid channel input symbols according to distribution q_X. The resulting block, i.e. the whole message, is transmitted through the channel. The channel distorts the message and the transmitter observes the received block via the feedback channel. For the first (efficient) repetition of the message, the transmitter compresses the distortion into $N_1 H(X|Y)$ bits and then maps the result into $N_2 = \frac{N_1 H(X|Y)}{H(X)}$ iid channel inputs and transmits the resulting message through the channel. In the i^{th} iteration (or the $(i-1)^{\text{st}}$ effective repetition of the message), $i = 1 \dots N$, the distortion of the previous block is compressed to $N_{i-1} H(X|Y)$ bits, where $N_{-1} = N$, then the result is mapped into $N_i = \frac{N_{i-1} H(X|Y)}{H(X)}$ iid channel inputs to be transmitted over the channel. When the process continuous indefinitely, the number of transmitted symbols is

$$\frac{N}{H(X)} + \frac{N_1 H(X|Y)}{H(X)} + \dots + \frac{N_{i-1} H(X|Y)}{H(X)} + \dots =$$
$$\frac{N}{H(X)} \left(1 + \frac{H(X|Y)}{H(X)} + \dots + \left(\frac{H(X|Y)}{H(X)} \right)^{(i-1)} + \dots \right) = \frac{N}{H(X) - H(X|Y)}$$

resulting in an average rate equal to the channel capacity. The length of the transmitted blocks, schematically shown in Fig. 1.6, decreases as the process proceeds. In practice, the iteration stops after a fixed number of times. The last block or, in other words, the base block which contains a small number of symbols is reliably sent over the channel using a strong termination (feedback) code, see Fig. 1.6. In addition to the information about the channel noise, every block contains some other information about the previous block, e.g. the length of the previous block. Before the termination block, a synchronization sequence is sent to mark the beginning of the termination code. This framework shows the problem of channel coding with feedback to be strongly related to the problem of source coding for which there are both rich theory and efficient lossless source coding algorithms.

For decoding, the decoder of the scheme looks for the synchronization sequence in order to find the termination code, decodes the last block and uses the decoded information to decode the block before the synchronization sequence. In this way, the decoded information of the i^{th} iteration is used to resolve the uncertainty in the $(i-1)^{\text{st}}$ iteration. This process continues until the uncertainty in the first block, corresponding to the original message, is removed. This decoding method is called a **right-to-left** decoding method because the receiver must receive the entire code word and decode from the

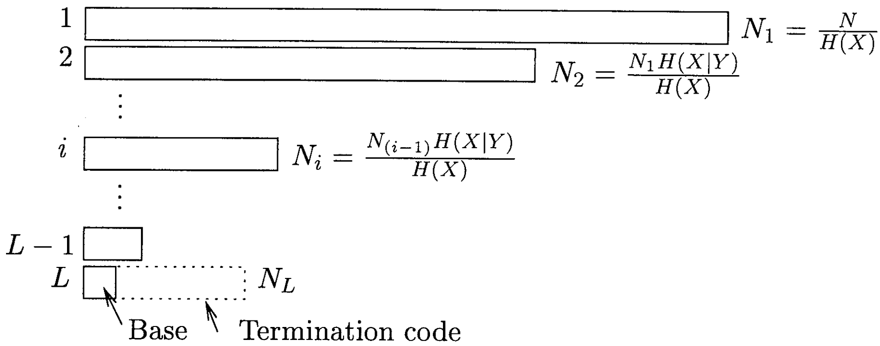


Figure 1.6: A schematic illustration of transmitted blocks in every iteration of Ooi's scheme.

end towards the beginning of the block. Note that the code words of the scheme are variable length (in [47] Ooi also presents a fixed length version of his code). The correct decoding of a received code word depends on correctly finding the synchronization sequence and correctly decoding the termination code. Then, the reliability function of the scheme becomes

$$E_{CEC}(R) = E_0 \left(1 - \frac{R}{I(X;Y)}\right), \tag{1.7}$$

where

$$E_0 = \max_{x,x' \in \mathcal{X}} D\left(p_{Y|X}(\cdot|x) \parallel p_{Y|X}(\cdot|x')\right).$$

As Ooi also mentions "Burnashev [10] has shown that $E_{CEC}(R)$ is an upper bound to the error exponent of any variable length feedback transmission scheme for a DMC".

Weldon's scheme, presented in a Bell Labs memorandum dated back in 1962 [93], is an information feedback scheme for Binary Symmetric Channels (BSC). A BSC is a DMC with two inputs and two outputs where the erroneous transition probability for each input is p. This scheme is basically similar to Ooi's scheme. In Weldon's version, however, all iterations have the same block length. In doing so, every block consists of two parts: the compressed form of the distortion of the previous block and the new information for filling the remaining of the block. Of course, some provision has to be made for separating these two parts in the decoding stage. For the last block Weldon used an ordinary linear code. Weldon's method yields an error exponent similar to (1.7), however, with smaller E_0 due to using a linear code for the last block. Fig. 1.7 shows a schematic view of the blocks transmitted in Weldon's scheme.

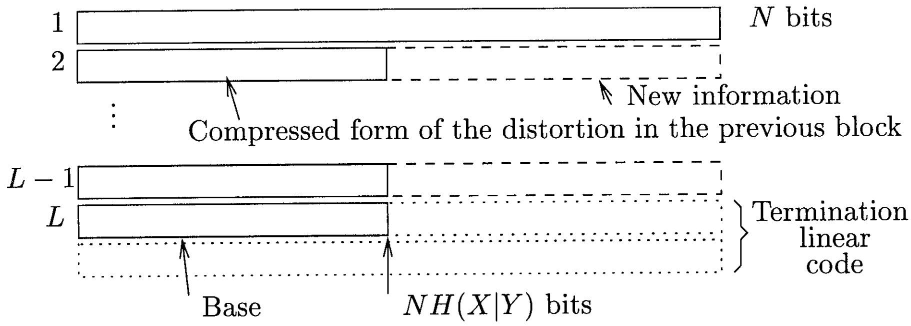


Figure 1.7: A schematic illustration of transmitted blocks in every iteration of Weldon's scheme.

Center-of-gravity methods

In [55], Schalkwijk gave the center-of-gravity information feedback model as an unifying principle for many information feedback schemes designed for AWGN channels. In these channels, transmitted signals are constrained in average power and sometimes in peak power. In this class of schemes, each transmission is also a compressed form of its preceding transmissions. To explain the center-of-gravity model, consider $\mathcal{W} = \{1, 2, \ldots, |\mathcal{W}|\}$ to be the set of $|\mathcal{W}|$ equiprobable messages. Every message $w \in \mathcal{W}$ corresponds to a D-dimensional vector $\mathbf{x}(w) = (x_1(w), \ldots, x_D(w))$ as the input of a vector channel. Therefore, the set of input symbols of the channel is defined $\mathcal{X} = I\!R^D$, where $I\!R$ is the set of real numbers. The center-of-gravity method is a repetition type scheme, but it does not simply repeat the input signal N times to have a code word like $\bar{\mathbf{x}}^N = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ where $\mathbf{x}_n = \mathbf{x}(w)$ for $n = 1, \ldots, N$. Instead, before the n^{th} transmission, $n = 1, \ldots, N$, a common signal $\mathbf{u}_{n-1} \in I\!R^D$ is computed at the receiver from already received signals and sent back to the transmitter. On the n^{th} transmission, or the $(n-1)^{\text{st}}$ repetition of the message, $\mathbf{u}_{n-1}$ is subtracted at the transmitter and added again at the receiver. Therefore, the transmitted vector is

$$\bar{\mathbf{x}}^N = ((\mathbf{x}(w) - \mathbf{u}_0), (\mathbf{x}(w) - \mathbf{u}_1), \ldots, (\mathbf{x}(w) - \mathbf{u}_{N-1}))\,.$$

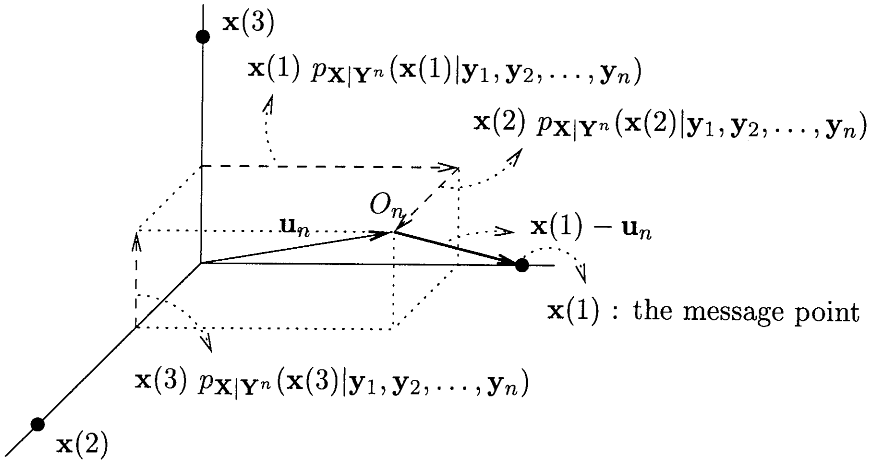


Figure 1.8: An example from [56] to illustrate the center-of-gravity method.

Now what is the optimum sequence $\mathbf{u}_{n-1}$, $n = 1, \ldots, N$, in order to minimize the average transmission power. From $(n-1)$ received signals $\mathbf{y}_1, \ldots, \mathbf{y}_{n-1}$[11], the receiver can compute $p_{\mathbf{X}|\mathbf{Y}^{(n-1)}}(\mathbf{x}(w)|\mathbf{y}_1, \ldots, \mathbf{y}_{n-1})$, i.e. the a-posteriori signal probabilities, for all $w \in \mathcal{W}$. From the receiver's point of view, the expected energy on the n^{th} forward transmission is

$$\mathbb{E}\{E_n\} = \sum_{w=1}^{|\mathcal{W}|} |\mathbf{x}(w) - \mathbf{u}_{n-1}|^2 p_{\mathbf{X}|\mathbf{Y}^{(n-1)}}(\mathbf{x}(w) \mid \mathbf{y}_1, \ldots, \mathbf{y}_{n-1}).$$

In Schalkwijk's terms, "this is the moment of inertia of the signal set around the tip O_{n-1} of the vector $\mathbf{u}_{n-1}$, see Fig. 1.8. But this moment of inertia around O_{n-1} can be minimized by placing O_{n-1} at the center of gravity" [56]. The d^{th} coordinate of the center of gravity is given by

$$u_{0d} = \sum_{w=1}^{|\mathcal{W}|} x_d(w)\pi_w, \qquad d = 1, \ldots, D,$$

$$u_{(n-1)d} = \sum_{w=1}^{|\mathcal{W}|} x_d(w)\, p_{\mathbf{X}|\mathbf{Y}^{(n-1)}}(\mathbf{x}(w) \mid \mathbf{y}_1, \ldots, \mathbf{y}_{n-1}), \qquad \begin{array}{l} d = 1, \ldots, D; \\ n = 2, \ldots, N, \end{array}$$

[11] When $n = 1$, the set of received signals is assumed to be empty.

where $\pi_1, \pi_2, \ldots, \pi_{|\mathcal{W}|}$ are the prior probabilities of messages before starting the transmission. In these methods, the center of gravity approaches the true message point in the D-dimensional signal space and, therefore, the energy of transmissions decreases as the process proceeds. On the other hand, subtracting and then again adding common signal vectors $\mathbf{u}_{n-1}$, $n = 1, \ldots, N$, does not affect the probability of error if we compare it to the original scheme that repeats the same signal N times. Moreover, it is possible to stop further transmissions as soon as the a-posteriori probability of a message at the receiver reaches a threshold. In this way a sequential decoding, or a **left-to-right** decoding, is possible. In this respect, the center-of-gravity schemes are different from Ooi-Weldon's scheme, where one must decode from the end to the beginning of the received blocks, i.e. the right-to-left decoding.

A suboptimum version of center-of-gravity uses the *receiver's best* guess in the place of the center of gravity. The best $(n-1)^{\text{st}}$ guess is the message $w \in \mathcal{W}$ which maximizes $p_{\mathbf{X}|\mathbf{Y}^{(n-1)}}(\mathbf{x}(w) \mid \mathbf{y}_1, \ldots, \mathbf{y}_{n-1})$. The best guess is inferior to the center-of-gravity, however, feeding back one out of $|\mathcal{W}|$ possible signals needs less capacity in the feedback channel than the amount needed in the center-of-gravity method. Many feedback schemes for AWGN channels can be explained by the center-of-gravity principle. In the following we briefly mention a few important information feedback schemes of this kind. For an in depth study, the interested reader is referred to the original papers [55, 56] and references therein.

Turin's scheme is the first scheme which uses the best guess and the center of gravity feedback strategies to send a message of one bit, i.e. $\mathcal{W} = \{0, 1\}$, through an AWGN channel [84]. The inputs of the channel are two one-dimensional ($D = 1$) antipodal signals. The channel has unlimited bandwidth. The suboptimum and the optimum versions of this scheme achieve approximately $\frac{2}{3}$ of channel capacity and the channel capacity, respectively.

Kramer's scheme is a best guess scheme for a set of $|\mathcal{W}|$ orthogonal signals, where each of the signals corresponds to one message [35]. At the transmitter, the same set of signals is used in every transmission, however, with different gain constants. In other words, the orthogonal signals have length $\sqrt{E_1}, \sqrt{E_2}, \ldots, \sqrt{E_N}$ on the N successive forward transmissions, respectively. For rates smaller than half the channel capacity, the scheme yields an error exponent, which is N-folded exponential

$$P_e \leq \exp\left(-\overbrace{\exp\exp\ldots\exp}^{(N-1)\text{ times}}\left((\frac{C}{2} - R)T \right) \right), \qquad \text{for } R \leq \frac{C}{2},$$

where T is the total duration of a transmission session. In [36], Kramer showed that if a peak transmission power constraint is also imposed, the N-fold exponential error exponent cannot be obtained. The peak power constraint limits the error probability expression to only a single exponential form. However, even with this constraint the proposed scheme can still provide a significant improvement over the equivalent one-way scheme.

Schalkwijk-Kailath's scheme [65] applies to AWGN channels. The input signal set $\mathcal{X}$ is the one-dimensional close interval $[\frac{-a}{2}, \frac{a}{2}]$. The input interval is divided into $|\mathcal{W}|$ equal length message intervals. The middle point of intervals is the message point θ to be transmitted to the other side. An error occurs if the output $y = \theta + z$, where $Z \sim N(0, \sigma^2)$, falls outside of the transmitted message interval. For simplicity, if every message was repeated N times, and then N received samples were averaged, we would obtain an estimate

$$\hat{\theta}(N) = \theta + \frac{1}{N}(z_1 + z_2 + \ldots + z_N),$$

which would have a normal distribution with mean θ and variance $\frac{\sigma^2}{N}$. However, the average transmitted energy during these N transmissions would be

$$\mathbb{E}[E_{N-repetitions}] = N\frac{a^2}{12}.$$

When Schalkwijk and Kailath applied the center-of-gravity principle to the transmitted signals, they greatly reduced the average transmission energy to

$$\mathbb{E}[E_{SK}] = \frac{a^2}{12} + \sigma^2 \sum_{n=1}^{N-1} \frac{1}{n}.$$

The error exponent of the scheme is double exponential[12] at the rates up to channel capacity. To wit,

$$P_e \leq \exp\left(\frac{-a^2}{4N_o}\exp\left(2T(C-R)\right)\right),$$

where T and $\frac{N_o}{2}$ are the total duration of a message transmission session and the two-sided power spectral density of the AWGN, respectively. In [54], Schalkwijk extended this result to band limited AWGN channels by expanding

[12] In [98], Wyner showed that imposing peak power constraint yields a single exponential error exponent.

the signal structure after each iterative transmission and obtained the first deterministic scheme to achieve the capacity of a band limited AWGN channel.

Butman's scheme is an extension of Schalkwijk's scheme to Gaussian channels with memory [12]. He takes into account the memory in the forward channel in making the linear least mean square estimate at the receiver. For a Gaussian channels with memory (first-order autoregressive noise), Butman's scheme increases the capacity of the feedback-free channel. **Ozarrow** [50] extended Schalkwijk's scheme to AWGN MAC's with two users and to broadcast channels. In this way, the capacity region of the MAC with feedback is shown to be larger than that of the feedback-free MAC and the achievable region for most broadcast channels with feedback is shown to lie outside the set of rates achievable in the absence of feedback. **Omura** [44], **Kashyap** [34] and **Campbell** [14] devised methods similar to Schalkwijk's scheme.

1.6.2 Repeat to correct erroneous receptions

The information feedback schemes using the second mechanism repeat messages or message chips whenever necessary. If a message (chip) is received correctly, the transmitter moves on to the next message (chip). Otherwise, the entire message (chip) is repeated. From this point of view, these schemes are similar to the ARQ schemes discussed in Section 1.5. However, here the transmitter side which observes the received message (chip) via the feedback channel initiates the acknowledgement or rejection decision. Therefore, there is a need for a mechanism to convey the acknowledgement or rejection signals to the receiver for every message (chip). In the schemes using this mechanism, the rejected message (chip) is totally neglected and it is not possible to compress the correcting repetitions or to optimize their transmission power (in fact, sometimes there is an expansion of transmissions in order to correct an erroneous reception). The optimum performance in these schemes is gained due to immediately repeating the erroneous receptions. In the following we explain a few information feedback schemes that use this mechanism to correct transmission errors.

Schalkwijk-Barron's scheme

This is a sequential signalling over AWGN channels subject to a peak power constraint [64]. The signalling has two modes: the message mode and the control mode. In the message mode, one of $|\mathcal{W}|$ messages is sent over the channel using a set of $|\mathcal{W}|$ orthogonal signals. Then the receiver sends back the decoded message or the transmitter deduces it by observing the received

message decoded	control decoded	the next transmission
correctly	correctly (ack $\to$ ack)	new message
erroneously	correctly (rej $\to$ rej)	retransmission
correctly	erroneously (ack $\to$ rej)	retransmission
erroneously	erroneously (rej $\to$ ack)	new message

Table 1.1: The decoding outcomes of the message and the control signal in Schalkwijk-Barron's scheme, where ack and rej stand for acknowledgement and rejection control signals, respectively.

signal. In the control mode, the transmitter sends the acknowledgement or the rejection signal depending on whether the decoder is correct or not, respectively. The receiver decodes the control part and if the decoded result is an acknowledgement, the receiver accepts the decoded message as correct. If the control part is decoded as a rejection, the receiver discards the decoded message and waits for a retransmission. Table 1.1 summarizes the possible alternatives in decoding the message and control parts, as well as the content of the next transmission. Notice the third row where the message is decoded correctly but the acknowledgement is decoded as a rejection. Since the transmitter is also informed about the decoded control signal, it retransmits the same message to undo the error. A decoding error occurs only when the message is decoded erroneously and the rejection control signal is decoded as an acknowledgement, see the forth row of Table 1.1.

As mentioned before, Schalkwijk and Barron used a set of orthogonal signals to send the message part. For the control part, they used a sequential decision feedback scheme for the AWGN channel, which was first described by Viterbi [92]. Two antipodal signals $\sqrt{P_{av}}$ and $-\sqrt{P_{av}}$ are repeatedly sent through the channel to convey the acknowledgement and the rejection signals, respectively. The receiver computes the log-likelihood ratio $\ln(\frac{p_+}{p_-})$, where p_+ and p_- are the posteriori probabilities for $\sqrt{P_{av}}$ and $-\sqrt{P_{av}}$ signals, respectively. As soon as this ratio reaches a positive or a negative threshold the transmission of the control signal is terminated.

Unlike the scheme of Schalkwijk and Kailath which requires unlimited peak power, this scheme has limited peak power. In the message mode, the energy of the orthogonal signals is $P_{av}T_m$ and in the control mode the average energy of the antipodal signals is $P_{av}\bar{T}_c$, where T_m and $\bar{T}_c$ are the duration of the message signal and the average duration of the control signal, respectively. This scheme was devised in order to show that even with a peak power constraint, an information feedback scheme can communicate at rates up to channel capacity with a reliability function larger than that of the one-way methods. In general, when $\alpha = \frac{P_{peak}}{P_{av}} \geq 1$ Schalkwijk and Barron derived the following error exponent

$$E_\alpha(R) = (\sqrt{\alpha C - R} + \sqrt{C - R})^2. \tag{1.8}$$

Note that we only explained the case of $\alpha = 1$.

Yamamoto-Itoh's scheme for AWGN channels [99] is a modification of Schalkwijk-Barron's scheme, where in the control mode Yamamoto and Itoh employed a new blockwise decision scheme instead of Viterbi's sequential decision scheme. In this way, a totally fixed length transmission method is obtained such that the same reliability function as (1.8) is asymptotically attained.

Yamamoto and Itoh also generalized Schalkwijk-Barron's scheme for use in DMC's. In a block of N transmissions, a block code of length γN is used to send the message part, $\gamma < 1$. They used the rest of the block for the control part to convey the acknowledgement or the rejection signals to the receiver using two code words $\mathbf{x}^{(1-\gamma)N} = (x, x, \ldots, x)$ and $\mathbf{x}^{(1-\gamma)N} = (x', x', \ldots, x')$, where x and $x' \in \mathcal{X}$, respectively. A received control part is decoded to an acknowledgement if n_y, the number of output symbol y in the received control part, satisfies

$$(1-\gamma)N \cdot p_{xy}(1-\delta) \leq n_y \leq (1-\gamma)N \cdot p_{xy}(1+\delta) \qquad \text{for all } y \in \mathcal{Y},$$

where $p_{xy} = p_{Y|X}(y|x)$. Otherwise, the control part is decoded to a rejection. Hence, the reliability function of the scheme becomes

$$E_{YI}(R) = E_0 \left(1 - \frac{R}{I(X;Y)}\right),$$

where

$$E_0 = \max_{x,x' \in \mathcal{X}} D\left(p_{Y|X}(\cdot|x) \parallel p_{Y|X}(\cdot|x')\right).$$

Note that this reliability function is the same as that of Ooi's scheme, as Ooi used Yamamoto-Itoh's scheme as the termination code.

Multiple repetition feedback coding schemes

Based on Horstein's results, see the following paragraphs, Schalkwijk [58] derived a class of simple and asymptotically optimal strategies for block coding on BSC's with noiseless feedback, which are nowadays referred to as Multiple Repetition Feedback Coding (MRFC) schemes. Resembling Horstein's strategy, but having very low coding complexity, a recursive (nonblock) coding method of MRFC schemes for BSC's was proposed in [66] by Schalkwijk and Post. Since then, MRFC schemes have been extended to a large class of DMC's, see [4] and [91].

Here we briefly explain MRFC schemes in the framework of the repeat to correct error mechanism. In this framework, we assume that every message consists of message chips to be transmitted to the receiver. Every message chip corresponds to a channel input symbol. Suppose that the transmitter sends symbol x over the DMC. Upon correct reception of the symbol, the next symbol is transmitted. However, if x is erroneously received as $y \neq x$, then x is retransmitted k_{xy} times. In the decoder, a received subsequence $yx^{k_{xy}}$, i.e. a y followed by k_{xy} x's, indicates the transmission error $x \to y$. The error is subsequently corrected by replacing subsequence $yx^{k_{xy}}$ with x to retrieve the corresponding message chip. In order to prevent removing message chips, the sequence of message chips must be constrained or precoded such that the corresponding channel inputs do not constitute the so-called forbidden subsequences $yx^{k_{xy}}$, $x \neq y$. In MRFC schemes there is no explicit acknowledgement signal but the rejection signal, which does not occur so often, is a few repetitions of the symbol in error. In fact, the acknowledgement signals are hidden in the sequence of the message chips, where a correctly received message chip, i.e. its corresponding channel input symbol, cannot be the beginning of a forbidden subsequence. This is guaranteed by the precoding and repetition rules.

Horstein's scheme [29] is a sequential scheme to transmit a message through a BSC with noiseless feedback. The message is composed of iid bits with $p_W(0) = p_W(1) = 0.5$, which is called the message sequence. Let p be the crossover probability of the BSC and $q = 1 - p$. If a binary point is put to the left of the message sequence of infinite length, the message may then be represented by a point in the interval $(0, 1)$ and it is called the message point in Horstein's terminology. In the following, we want just to show how Horstein's scheme uses the repeat to correct error mechanism.

The encoding procedure is based on receiver's distribution for the location of the message point. In every iteration, the receiver distribution $f_{\Theta}^{(i-1)}$, $i = 1, 2, \ldots$, is modified according to the i^{th} received bit. The next transmitted bit depends on the location of the message point with respect to the median of the last receiver distribution. The $(i-1)^{\text{st}}$ median, i.e. m_{i-1}, $i = 1, 2, \ldots$, is a point on the interval $(0, 1)$, which divides receiver's $(i - 1)^{\text{st}}$ distribution into

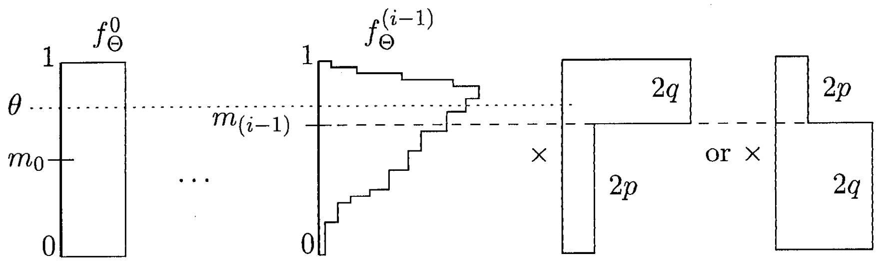


Figure 1.9: An illustration of how Horstein's scheme determines a transmission bit and how the next distribution of the receiver is determined according to the received bit.

two equiprobable parts. Initially, the receiver has no priori knowledge about the message sequence. Therefore, its initial distribution of the message point is uniform over the interval $(0, 1)$ and consequently $m_0 = 0.5$, see f^0_Θ in Fig. 1.9. In the i^{th} transmission, if θ lies under m_{i-1}, 0 is transmitted; otherwise 1 is transmitted. In this way, from the receiver's point of view, the probability of transmitting 0 or 1 is always 0.5. Upon receiving 1 at the i^{th} transmission, i.e. $y_i = 1$, the probability that the message point lies in interval $(m_{i-1}, 1)$ can be written (from the receiver's point of view) as

$$
\begin{aligned}
& Pr\{\Theta \in (m_{i-1}, 1) | \mathbf{Y}^{(i-1)} = \mathbf{y}^{(i-1)}, Y_i = 1\} \qquad (1.9) \\
= & \frac{Pr\{\Theta \in (m_{i-1}, 1), Y_i = 1 | \mathbf{Y}^{(i-1)} = \mathbf{y}^{(i-1)}\}}{Pr\{Y_i = 1 | \mathbf{Y}^{(i-1)} = \mathbf{y}^{(i-1)}\}} \\
= & \frac{Pr\{\Theta \in (m_{i-1}, 1) | \mathbf{Y}^{(i-1)} = \mathbf{y}^{(i-1)}\}}{Pr\{\Theta \in (m_{i-1}, 1) | \mathbf{y}^{(i-1)}\} \cdot Pr\{Y_i = 1 | \mathbf{y}^{(i-1)}, \Theta \in (m_{i-1}, 1)\}} \\
& \frac{\cdot Pr\{Y_i = 1 | \mathbf{Y}^{(i-1)} = \mathbf{y}^{(i-1)}, \Theta \in (m_{i-1}, 1)\}}{+ Pr\{\Theta \in (0, m_{i-1}) | \mathbf{y}^{(i-1)}\} \cdot Pr\{Y_i = 1 | \mathbf{y}^{(i-1)}, \Theta \in (0, m_{i-1})\}} \\
= & \frac{0.5 \times q}{0.5 \times q + 0.5 \times p} \\
= & q. \qquad (1.10)
\end{aligned}
$$

On the other hand, when $y_i = 1$, the upper part (above m_{i-1}) of receiver distribution $f^{(i-1)}_\Theta$ is multiplied by factor A to obtain f^i_Θ, see Fig. 1.9. Then, in the resulting distribution, the area above median m_{i-1} indicates the probability that the message point lies in interval $(m_{i-1}, 1)$, given $y_i = 1$. This area should be equal to q as seen in (1.10). Therefore, the multiplication factor A comes from

$$\int_{m_{i-1}}^{1} f_\Theta^i(\theta)\ d\theta = \int_{m_{i-1}}^{1} f_\Theta^{(i-1)}(\theta) \cdot A\ d\theta = q$$

$$\implies \qquad A = \frac{q}{\int_{m_{i-1}}^{1} f_\Theta^{(i-1)}(\theta)\ d\theta} = 2q.$$

In a similar way, we can obtain the multiplication factor $2p$ for the part of the receiver distribution under m_{i-1}, again assuming that $y_i = 1$ has been received. Likewise, upon reception of 0, the lower side of m_{i-1} in receiver's distribution is multiplied by $2q$ and its upper side is multiplied by $2p$, see Fig. 1.9. The next transmission bit will be determined according to the position of θ with respect to m_i, the median of the i^{th} distribution, where $\int_{m_i}^{1} f_\Theta^i(\theta)\ d\theta = \frac{1}{2}$.

In the i^{th} transmission, it appears that Horstein's scheme sends (repeats) the difference between message θ and median m_{i-1} (the latest estimation of the massage at the receiver). However, one must notice that the complete difference here is not sent through the channel as it is done in the schemes using the repeat to resolve uncertainty mechanism. For example, in the center-of-gravity or Ooi-Weldon schemes, the transmitted difference is sufficient to completely specify the message if the corresponding difference is received correctly, while this is not the case for Horstein's scheme. If we regard every transmitted bit here as a message chip, however, this scheme fits better in the category of the schemes using the repeat to correct error mechanism. The very similar interpretation is also used for MRFC schemes[13]. Note that the real message θ and encoding rules determine the sequence of the transmitted message chips.

Whenever a transmission error occurs, the next median drifts away from the message point θ, see the lower part of Fig. 1.10 for a schematical illustration. Therefore, the message chip in error is retransmitted a few times in order to push the median back towards θ. In this way, Horstein's scheme effectively rejects the erroneous reception of a message chip. On the other hand, when a message chip is received correctly, it can be seen that the median point moves towards θ. The upper part of Fig. 1.10 shows the medium points in a few such transmissions. For a correctly received message chip, the scheme has an

[13]Interesting to know is that Schalkwijk [58] derived MRFC schemes for BSC's by detailed study of the median path of Horstein's scheme. He noticed that for certain crossover probabilities of the channel, the $(k-1)^{\text{st}}$ median after an error coincides the one before the error ($k > 2$ is an integer number).

implicit acknowledgement procedure by letting the median move towards the message point without being rejected as in the previous case. The cost of such implicit signalling is paid when the receiver distribution is multiplied by pair $(2p, 2q)$ instead of $(0, 2)$ (even for the correctly received message-chips). With some inspection, it can be seen that multiplication factors $(2p, 2q)$ are essential in rejecting an erroneous message chip when it occurs.

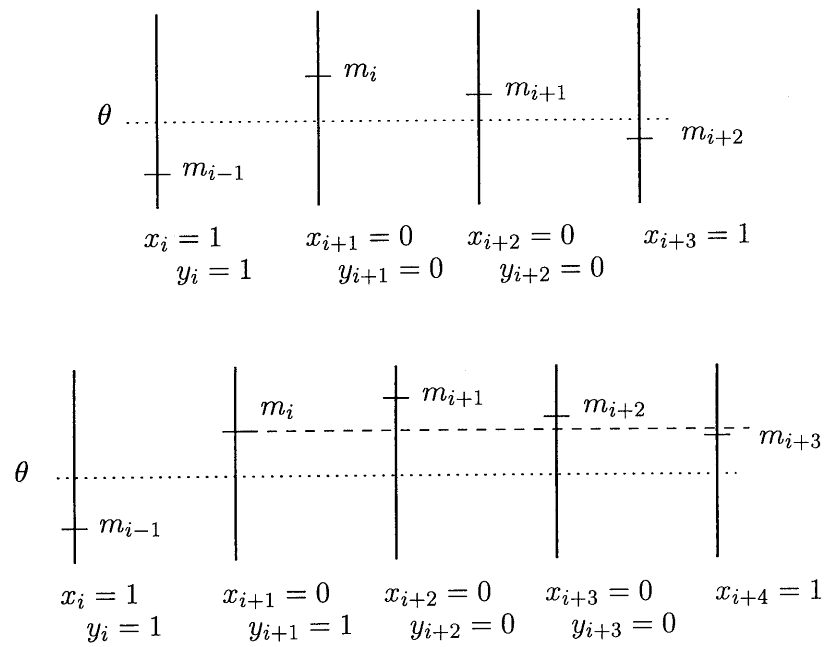


Figure 1.10: An example to illustrate the median points in a few correct transmissions (above) and those when a transmission error occurs (below).

The first message bit is decoded at, let say, the n^{th} transmission if the area of receiver distribution exceeds $(1 - P_e)$ in one of the half intervals (0, 0.5) or (0.5, 1). Then, the first decoded bit, i.e. $\hat{w}_1$, is set to 0 or to 1, respectively. The interval corresponding to probability P_e is discarded and the area corresponding to probability $(1 - P_e)$ is expanded to 1. Assuming that the first bit was decoded to 0, the second message bit is determined as soon as either area of the intervals (0, 0.25) or (0.25, 0.5) exceeds $(1 - P_e)$, and so on. Here n is the number of transmitted symbols influenced by a given message bit, or according to Horstein, it is the constraint length of the scheme, which is a random variable designated by N. Let P_e^0 denote the probability of a $w_i = 0 \rightarrow \hat{w}_i = 1$ error[14], $i = 1, 2, \ldots$. The error exponent is defined and given as

$$E_H^0 = \lim_{P_e^0 \to 0} \frac{-\log P_e^0}{\bar{N}} = D(0.5 \parallel p) = 0.5 \log \frac{1}{4pq}, \tag{1.11}$$

where $\bar{N} = \mathbb{E}[N]$, see [29]. Due to symmetry, the error exponent of a $1 \rightarrow 0$ error can be written as $E_H^1 = E_H^0$. This exponent holds when the transmission rate is equal to channel capacity. Because, according to Horstein, every transmitted bit is equally probable from the receiver's point of view so the average amount of conveyed information equals

$$R = p \log \frac{p}{0.5} + q \log \frac{q}{0.5} = h(0.5) - h(p) = C.$$

Zigangirov and **Burnashev** used Horstein's idea in [104, 101, 102, 103] and in [10, 11], respectively, where they modified Horstein's scheme and improved some shortcomings of Horstein's presentation.

1.7 Advantages of feedback communication

Nowadays, two separate one-way coding schemes are applied to many two-way channels. However, in these duplex channels information usually flows in one direction only and, therefore, there is a potential to employ a feedback communication scheme. In this section we summarize the advantages of feedback coding schemes over feedback-free, i.e. one-way, coding schemes. Knowing these advantages is essential for a communication engineer in order to design an efficient communication system for which all available resources are exploited and the best trade-offs are made among the possible alternatives.

Channel capacity

Shannon [69] proved that the capacity of a memoryless channel is not increased by feedback. Alajaji [2] showed that this statement is also true for discrete channels which add modulo-$|\mathcal{X}|$ an input-independent noise process with memory. However, there are many channels with memory where capacity is increased by feedback. For some simple channels with memory see [19]. For an additive Gaussian channel with memory (first order autoregressive noise), Butman [12] proposed a feedback scheme that yields a higher rate than the

[14] Note that for message bit w_i, Horstein assumed that no decoding errors are made during interval of the N transmissions.

capacity of the feedback-free channel. Gaarder and Wolf [24] showed that the capacity region of a memoryless MAC can be increased by feedback. Extending Schalkwijk's scheme to a two-user AWGN MAC with feedback and to broadcast channels with feedback, Ozarrow [50] showed that the capacity regions of the channels with feedback are larger than those of the feedback-free channels. Shannon in [69] showed that the zero-error capacity of some memoryless channel can be increased by feedback.[15] Making use of feedback for enlarging the channel capacity appears to be appealing when we notice that, on the one hand, many communication channels have memory and, on the other hand, the number of multiuser channels such as MAC's is growing in the era of data communication networks.

Coding complexity and error performance

There are methods to define the concept of system complexity, for example see [52], [53] or Chapter 7 of [16], but we confine ourselves to a common sense definition of the complexity. We can consider the complexity as the amount of computation and memory needed in the coding process, i.e. encoding and decoding. The reliability function is one of the best tools that gives an insight into a tradeoff between the error performance and the complexity of different methods, though comparison based on the value of the error exponent is not fair for feedback schemes. The reliability functions of feedback schemes are mostly derived from deterministic procedures, while their one-way counterparts are based on random coding arguments.

Feedback coding schemes require less complexity compared to feedback-free coding schemes for the same error performance. This statement can be illustrated by the following argument. Let $E_{fb}(R)$ and N_{fb} denote the error exponent and the block or constraint length of a feedback coding scheme and $E_{ow}(R)$ and N_{ow} denote the corresponding parameters of a one-way coding scheme. For a given error performance we have $E_{fb}(R)N_{fb} \approx E_{ow}(R)N_{ow}$. Generally $E_{fb}(R) > E_{ow}(R)$, therefore, $N_{fb} < N_{ow}$. The complexity of different coding schemes is related to the block or the constraint length of the coding scheme. In fact, the complexity of feedback coding schemes is usually linear with the block or constraint length. On the other hand, the complexity of many one-way coding schemes is exponential on block or constraint length (for example the well known the Viterbi decoder). Hence for an identical constraint or block length, feedback scheme are, in general, much easier to implement. We finish this section with a statement by Erik Meeuwissen [40],

[15] Capacity is the supremum of rates at which the decoding error probability goes to zero (with increasing the block length) and zero-error capacity is the supremum of rates at which the decoding error probability is equal to zero.

which nicely captures the dual role of feedback in performance and complexity aspects.

> "The real problem in communication engineering is to find a tradeoff between performance and complexity. The attractiveness of freely available feedback is that it can increase performance and decrease complexity at the same time."

1.8 Overview of the thesis

This dissertation is organized in eight chapters. Chapter 2 gives a summary of MRFC schemes to which we will refer in the remainder of the dissertation.

To tackle the problem of noisy feedback channels, we present a feedback strategy for AWGN channels in Chapter 3. Using MRFC schemes, a bandwidth efficient modulation method is designed for AWGN channels where both forward and feedback channels have the same signal to noise ratio. The resulting method gives about 5.5 dB improvement over uncoded transmissions. In this method, the decoding complexity is considerably simplified as one compares it with the complexity of the trellis coded modulation. Given the fact that only one bit per dimension is fed back, the feedback channel can essentially be considered errorless for channels with moderate signal to noise ratios. The performance of the proposed method is supported by running simulations and the results are compared to those of the existing methods.

In Chapter 4, it is shown how the error performance of block MRFC can be improved by block retransmission strategies. For asymmetric DMC's, one of the presented strategies even yields a superior error performance compared to that of the already existing MRFC schemes.

Chapters 5 and 6 deal with the modification of MRFC schemes to use them in soft-output DMC's. A soft-output DMC, for example, can be obtained by quantizing the output of a binary-input AWGN channel. A class of compound channels is used to define soft-output DMC's. For this class of compound channels, it is shown that feedback and delayed side information at the transmitter cannot increase the capacity of the channel. Then, the modification of MRFC schemes is presented for a two-input four-output DMC. The new repetition scheme yields rates very close to channel capacity by using a proper Markov source as the precoder. Note that MRFC schemes consist of a precoding part and a repetition part. For the repetition part, the block and recursive coding methods of the new scheme are investigated in Chapter 6. This is followed by describing a practical precoding scheme. The extension of the new scheme to other channels, e.g. the Gilbert-Elliot channel and the block interference channels, with channel side information at the receiver is briefly outlined.

Finally in Chapter 7 we end the thesis by describing some useful applications for the obtained results, outlining some open problems and giving a few suggestions for future research.

Chapter 2

Multiple repetition feedback coding schemes

Multiple Repetition Feedback Coding (MRFC) schemes are a family of simple and efficient information feedback schemes for a large class of DMC's. In MRFC schemes, an erroneous reception is immediately corrected by repeatedly sending the correct input symbol. Therefore, the repeat to correct error mechanism as explained in the first chapter is used in these schemes. Because MRFC is the core of the presentation in the following chapters, this chapter summarizes the important aspects of MRFC schemes which are relevant to the dissertation.

2.1 History of MRFC

One of the many schemes that exploits a feedback channel is Horstein's sequential coding scheme [29] for a BSC. Horstein's scheme, briefly explained in the first chapter, suffers from implementation complexity. As message transmission proceeds, the required accuracy of the calculations for the receiver distribution and the median points increases. Another drawback of the scheme is that it is not mathematically tractable. Some modifications to Horstein's scheme, see for example [104, 101, 102, 103, 10, 11], were made to enable rigorous analysis, often, however, at the expense of even greater complexity of implementation. In 1971, Schalkwijk noticed that the $(k+1)^{\mathrm{st}}$ median coincides with the first one for certain channel crossover probabilities. In Figs. 2.1 and 2.2 the median paths of a few successions of transmissions are shown in two different cases, θ being the message point, f_Θ^i and m_i are the receiver distribution and the corresponding median after receiving the i^{th} bit, respectively. Let p be the crossover probability of the BSC. In Fig. 2.1, it is assumed

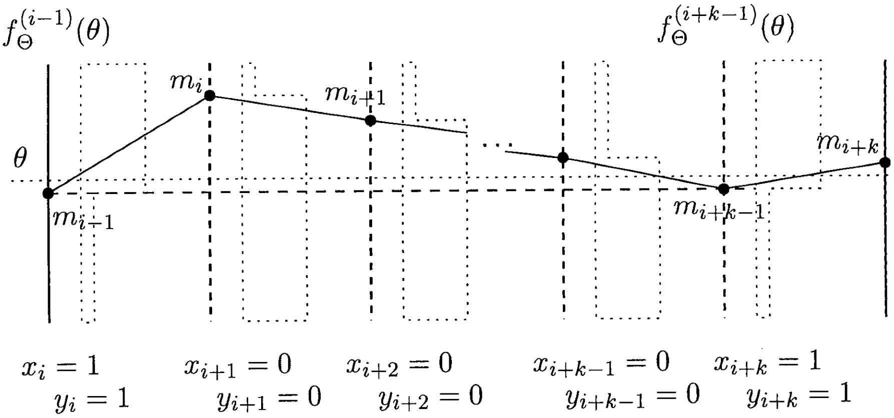


Figure 2.1: The median path for a few successive transmissions in Horstein's scheme with no transmission error.

that there is no transmission error. In this case, in order to $m_{i-1} = m_{i+k-1}$ we must have

$$\int_0^{m_{i-1}} f_\Theta^{(i-1)}(\theta)d\theta = \int_0^{m_{i+k-1}} f_\Theta^{(i-1)}(\theta)\ 2p\ (2q)^{(k-1)}\ d\theta = \frac{1}{2}$$

or

$$2p\ (2q)^{(k-1)} = 1. \tag{2.1}$$

Let crossover probability $p_k = 1 - q_k$ be the solution of equation (2.1) for a given $k \geq 3$. In Fig. 2.2 an error occurs at the i^{th} transmission. Therefore, the i^{th} median jumps to m_i, and not to m'_i as it should. The following $(k-1)$ transmissions bring the median back to the original place, as mentioned above, and the k^{th} signal puts the median to its correct value m'_i, the same place as it would be if the error did not occur, i.e.

$$\int_0^{m'_i} f_\Theta^{(i-1)}(\theta)\ 2qd\theta = \int_0^{m_{i+k}} f_\Theta^{(i-1)}(\theta)\ 2p\ (2q)^{(k-1)}\ 2q\ d\theta = \frac{1}{2}.$$

Due to (2.1), the above equation follows from the fact that the receiver's distribution of the message point does not change in the interval below the erroneous excursion, i.e. interval $(0, m_{i-1})$ in Fig. 2.2. Therefore, even when an error occurs, the position of the subsequent medians does not change once the error is corrected. This is rigorously proven by Veugen [89]. Schalkwijk

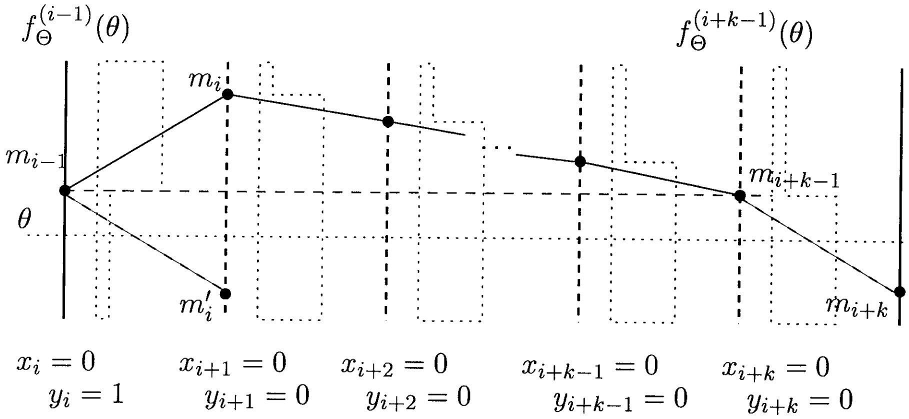


Figure 2.2: The median path for a few successive transmissions in Horstein's scheme with one transmission error.

considered the path of medians as an indicator of the true message point, which jumps to an erroneous path when an error occurs and returns to the correct path after k correctly received bits. Schalkwijk also noticed that an error-free median path does not allow transmitting subsequences 10^k, i.e. 1 followed by k 0's (see Fig. 2.1), and 01^k. When these subsequences occur in the received sequence, they indicate a $0 \to 1$ error and a $1 \to 0$ error, respectively. This observation of Schalkwijk led to the creation of a class of simple and asymptotically optimal strategies for block coding on BSC's with noiseless feedback, which are nowadays referred to as MRFC schemes.

2.2 MRFC for BSC's

Schalkwijk's observation considerably simplified Horstein's scheme and eliminated the computation of the receiver distribution and its median at every transmission. The error correcting mechanism of MRFC schemes forces the output symbol of the channel to equal the corresponding input symbol by employing the repeat to correct error mechanism. Fig. 2.3 shows the block diagram of the encoder of a MRFC scheme for the BSC. The encoder consists of two parts: the precoder and the transmitter. When 0 is erroneously received as 1, the **transmitter** repeats 0 k-times and when a 1 is erroneously received as 0, the transmitter repeats 1 k-times. By observing subsequence 10^k or 01^k in the received sequence, the decoder detects a transmission error and replaces the subsequence with 0 or 1, respectively. Of course, the input sequence to the transmitter should be constrained by the precoder in such a way that 10^k and 01^k subsequences do not occur in $\mathbf{v}$, see Fig. 2.3. Therefore, the **precoder** maps message w into a sequence satisfying these constraints.

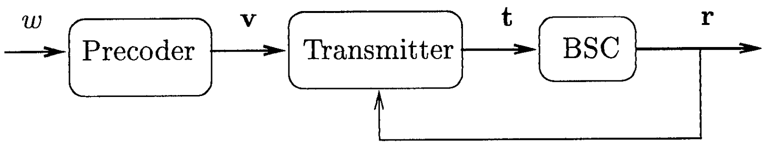


Figure 2.3: Block diagram of the encoder of the MRFC scheme for a BSC.

Definition 2.1 (Forbidden subsequence) *Those subsequences which are not allowed to occur at the output of the precoder are called forbidden subsequences.*

Note that forbidden subsequences are error indicators in the received sequence. In MRFC schemes for BSC's, 10^k and 01^k are forbidden subsequences for repetition parameter k. During transmission of the correcting bits of a precoded bit in error, new errors may occur. Each new error again causes k repetitions.

Definition 2.2 (Correction subsequence) *The correction subsequence of a precoded symbol that is erroneously received is the total sequence of repetitions needed to correct the error.*

Example 2.1 Consider the following outcomes in a MRFC scheme for BSC's with $k = 3$.

$$\begin{array}{rcl}
\mathbf{v} & = & \ldots\ 0\,1\ \qquad\qquad\qquad 1\ \ldots \\
\mathbf{t} & = & \ldots\ 0\,1\,\underline{1\,1\,1\,1\,1\,1}\,1\ \ldots \\
\mathbf{r} & = & \ldots\ 0\,\underline{0\,1\,1\,\underline{0\,1\,1\,1}}\,1\ \ldots
\end{array}$$

where $\mathbf{v}$, $\mathbf{t}$ and $\mathbf{r}$ are the precoded, transmitted and received sequences, respectively, as indicated in Fig. 2.3. The line in the sequence $\mathbf{t}$ indicates the correction subsequence of a precoded bit in error and the lines in sequence $\mathbf{r}$ indicate the forbidden subsequences in the received sequence. In the decoding stage, the net effect of the inner forbidden subsequence in $\mathbf{r}$ is a 1 which constitutes one bit of the outer forbidden subsequence. □

2.2.1 Precoding Rate

The precoder of MRFC schemes is a constrained channel whose capacity, or the so-called precoding rate, was studied in Shannon's original paper [67]. The precoder of the scheme mentioned in the previous subsection can be seen as a Finite State Machine (FSM) with non-transient states $Q_0^1, Q_0^2, \ldots, Q_0^{k-1}$ and $Q_1^1, Q_1^2, \ldots, Q_1^{k-1}$. For example, see the FSM with $k = 5$ shown on the left-hand side of Fig. 2.4, where a solid transition and a dashed transition correspond to producing 1 and 0, respectively. The **precoding rule** implied by the FSM can be expressed as follows. *After every 0 or 1 there should not be k ones or zeroes, respectively.* Or due to symmetry, *after every change in the output bit of the precoder, there should not be k similar output bits*, see the FSM consisting of states $Q^1, Q^2, \ldots, Q^{k-1}$ in the right hand-side of Fig. 2.4 where a solid transition and a dashed transition correspond to producing a precoded bit that is the same as the previous precoded bit and a precoded bit that is different from the previous precoded bit, respectively. Due to the restrictions mentioned above, a change is forced in state $Q^{(k-1)}$ (or in states $Q_1^{(k-1)}$ and $Q_0^{(k-1)}$).

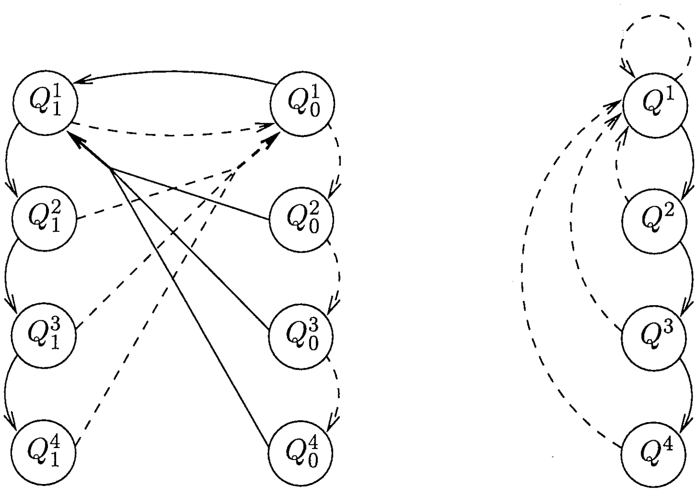


Figure 2.4: Two FSM diagrams of the repetition scheme for a BSC with $k = 5$.

Two general procedures for calculating the precoding rate of such schemes are presented in Appendix A. Nevertheless, to obtain an intuitive understanding, we also explain here two other approaches. Remember from Horstein's scheme that the interval $(0, 1)$ consists of M message subintervals such that each message point is located in the middle of the corresponding subinterval. Let us assume that all bits are transmitted correctly over a BSC with the channel crossover probability satisfying condition (2.1). The corresponding median paths constitute a so-called zero-error tree, see [66]. In this case, the distribution in the true message interval of the width $\frac{1}{M}$ will be multiplied by a factor of $(2q_k)^L$ in L transmissions and the corresponding message probability becomes

$$\frac{1}{M}(2q_k)^L \approx 1 \qquad \Rightarrow \qquad M \approx (2q_k)^L.$$

Here we assume that the subintervals are so small that the medians do not fall in the true message subinterval. The precoding rate indicates how many information bits are conveyed by a precoded symbol, i.e.

$$R_0^k = \lim_{L\to\infty} \frac{\log M}{L} = \log(2q_k). \tag{2.2}$$

According to this intuitive interpretation of the precoding rate [58, 91], a precoder which itself is a constrained channel and independent from any BSC is related to Horstein's scheme, whose distribution is always correctly modified by a factor pair $(2q_k, 2p_k)$.

Another way to calculate the precoding rate is to enumerate the number of precoded sequences. Let $M(L)$ denote the number of distinct precoded sequences of length L. The corresponding recurrent equation can be written as

$$M(L) = 2M(L-1) - M(L-k) \qquad \text{for } L > k. \tag{2.3}$$

To solve this recurrent equation, let us assume that $M(L) = K\alpha^L$ in (2.3), where K is independent of L. Then, we obtain the characteristic equation

$$\alpha^k = 2\alpha^{(k-1)} - 1. \tag{2.4}$$

Let α_k be the largest real solution of (2.4), then

$$R_0^k = \lim_{L\to\infty} \frac{\log M(L)}{L} = \log \alpha_k.$$

Equation (2.4) can be written as $(2-\alpha)\alpha^{(k-1)} = 1$ and comparing it with (2.1), we have

$$\alpha_k = 2q_k,$$

which shows that these two approaches give the same result. In the following subsections, two coding methods of MRFC for BSC's will be explained.

2.2.2 Block coding

The block diagram of the block MRFC is shown in Fig. 2.5, where L precoded bits are transmitted in N transmissions. If e errors occur in these transmissions, then $N = L + ke$. In block coding, when L precoded bits are to be transmitted through a BSC, there is no guarantee of exactly e errors and, therefore, a block length of $N = L + ke$ bits.

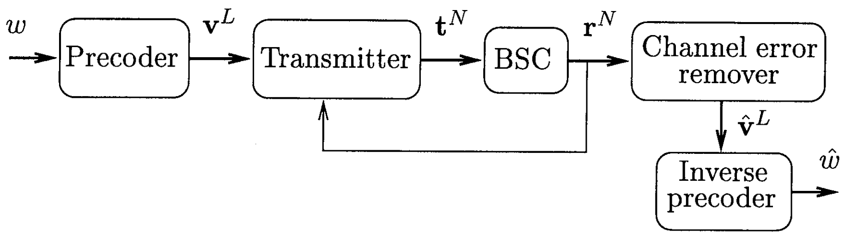


Figure 2.5: The block diagram of the block coding of MRFC for BSC's.

Block encoding

Schalkwijk [58] followed the precoded symbols by a tail to ensure that the scheme is able to correct **up to** e errors in a block. In his version, a certain sequence of length $N - L$, denoted by $\mathbf{z}_{L+1}^{N} = z_{L+1}, \ldots, z_N$, is appended to the precoded sequence before transmission. This added sequence is called the **tail** or the redundancy of the repetition part of the encoding. In MRFC for the BSC, the tail can be constructed by alternating the last bit of $\mathbf{v}^L$ until length N is reached, i.e. $\mathbf{z}_{L+1}^{N} = \bar{v}_L, v_L, \bar{v}_L, \ldots, v_L/\bar{v}_L$, where $\bar{0} = 1$ and $\bar{1} = 0$, see Fig. 2.6.

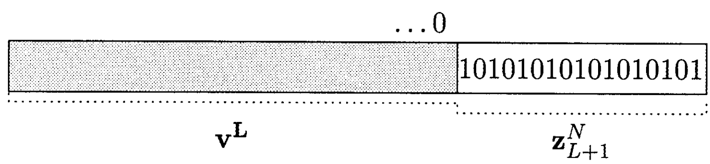


Figure 2.6: A code word of the MRFC scheme (before transmission).

During transmission, whenever a channel error occurs and, consequently, the corresponding transmitted bit is repeated k times, the last k bits of the tail are deleted. A tail of $T = N - L = k\ fN + 1$ bits is sufficient to correct up to $e = fN$ channel errors in a block, see [4] and [91] for the proofs. Here f is called the *correctable error fraction* of the block (for convenience fN is assumed to be an integer number). Therefore, the rate corresponding to the transmitter unit in Fig. 2.5 can be written as

$$\frac{L}{N} = 1 - kf - \frac{1}{N}$$

and since every precoded bit asymptotically contains R_0^k information bits, the asymptotic transmission rate can be written as

$$R(f) = R_0^k(1 - kf). \tag{2.5}$$

For a BSC with $p = 0.02$, the transmission rate of MRFC scheme with $k = 4$ is depicted in Fig. 2.7 in terms of rate verses correctable error fraction f. The rate performance lines in (2.5) are straight lines and tangent to the Hamming bound, i.e. $1 - h(f)$[1], at $f = p_k$, where p_k is the solution of (2.1) for a given k. To show this, we can write

$$\begin{aligned} R(p_k) &= R_0^k(1 - kp_k) = (1 - p_k)R_0^k - (k-1)p_k R_0^k \\ &\overset{(\alpha)}{=} q_k \log(2q_k) - p_k \log(2q_k)^{(k-1)} \overset{(\beta)}{=} q_k \log(2q_k) + p_k \log(2p_k) \\ &= 1 - h(p_k), \end{aligned}$$

where in (α) and (β) relations (2.2) and (2.1) are used, respectively. At this intersection point, the rate performance line of the corresponding scheme is tangent to the Hamming bound because

$$\frac{d}{df}\left(1 - h(f)\right)_{f=p_k} = \log(\frac{p_k}{q_k}) = \log\left(\frac{\frac{1}{2^k q_k^{(k-1)}}}{q_k}\right) = -k\log(2q_k) = -kR_0^k.$$

Block decoding

The block decoding of the MRFC is carried out in two stages. The second stage is the so-called inverse precoder, which exists in both block and recursive MRFC schemes and maps the precoded sequence into the corresponding message. The issue of the precoding will be addressed in Chapter 6.

In the first stage, channel errors are detected and corrected by locating and appropriately replacing the forbidden subsequences. For doing so, the decoder moves from the end towards the beginning of received block $\mathbf{r}^N$ and looks

[1]Function $h(f) = -f\log f - (1-f)\log(1-f)$.

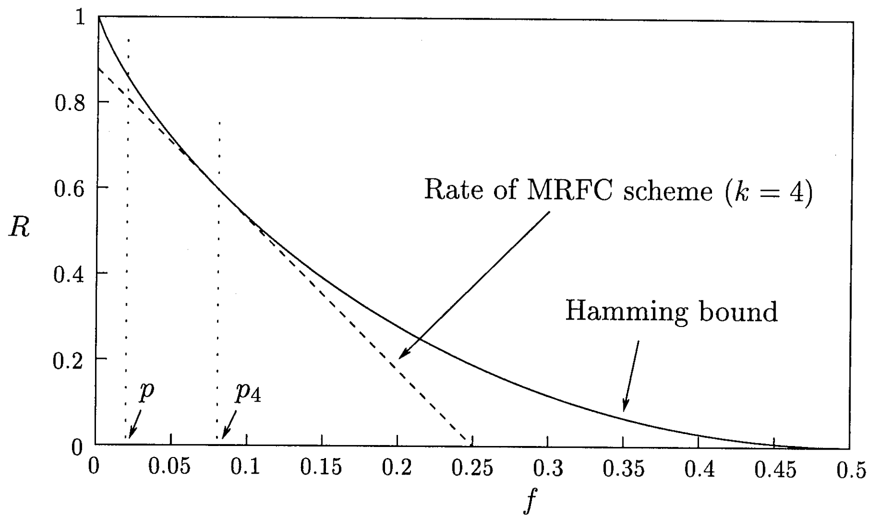


Figure 2.7: The rate of MRFC scheme for BSC's with $k = 4$.

for forbidden subsequences, e.g. subsequences 01^k and 10^k in the mentioned MRFC scheme for BSC's. As soon as a forbidden subsequence is found, the last bit of the subsequence replaces the whole subsequence. For example, $r_i \dots r_{i+k} = 01^k$ indicates that an error has occurred at the i^{th} position, therefore, $r_{i+k} = 1$ substitutes the subsequence. After the replacement, and again moving from right to left, the decoder looks for the next forbidden subsequence which starts at position $i' < i$. For this reason, this method is also referred to as the **right-to-left** decoding.

Let sequence $\hat{\mathbf{v}}^L$ denote the first L bits of the resulting sequence when the decoder finally reaches the beginning of the received block. A block decoding error might occur when $e > fN$, i.e.

$$P_b = Pr\{\hat{\mathbf{V}}^L \neq \mathbf{v}^L\} \leq Pr\{kE + 1 > T\} = Pr\{E \geq fN + 1\}, \tag{2.6}$$

where random variable E denotes the number of occurred errors in a block. If p is the crossover probability of the BSC, then E is a Bernoulli random variable with the parameter p. An upper bound to the right hand side of (2.6) can be obtained from the Chernoff bound

$$P_o = Pr\{E \geq fN + 1\} \leq 2^{-D(f\|p)N}, \qquad \text{for } p < f \leq \frac{1}{k}, \; k \geq 3, \tag{2.7}$$

where $D(\cdot\|\cdot)$ is the divergence function or the Kullback Leibler distance defined in Def. 1.8. The equality in (2.7) holds asymptotically (with some factor of the order $N^{\frac{-1}{2}}$ on the right-hand side) [25, pp. 130] and, therefore, we can define the exponent

$$E_o(f) = \lim_{N\to\infty} \frac{-\log_2 Pr\{E \geq fN+1\}}{N} = D(f \parallel p), \tag{2.8}$$

for $p < f \leq \frac{1}{k}$, $k \geq 3$. From (2.6), (2.7) and (2.8) we can write

$$P_b \leq 2^{-E_o(f)N} \qquad \text{for } p < f \leq \frac{1}{k},\ k \geq 3. \tag{2.9}$$

We will frequently use the upper bound (2.9) to estimate P_b. The exponent $E_o(f)$ can be related to the transmission rate via (2.5) to obtain the reliability function (corresponding to probability P_o). This reliability function of the MRFC scheme with repetition parameter $k = 4$ is depicted in Fig. 2.8 for a BSC with $p = 0.02$.

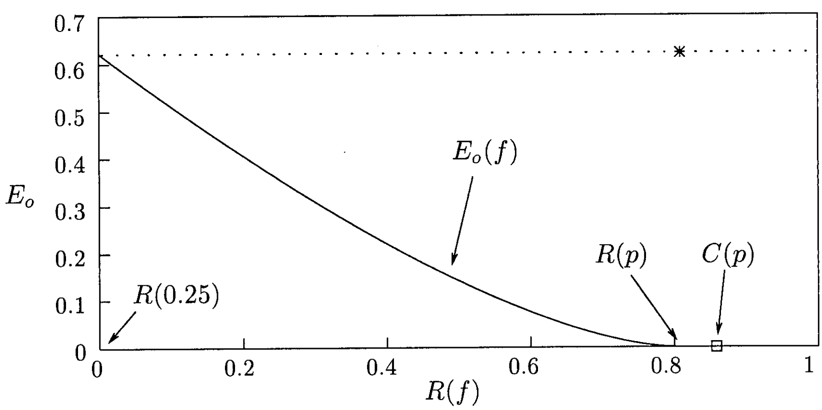


Figure 2.8: The $E_o(f)$ exponent of the MRFC scheme with $k = 4$ used in a BSC with $p = 0.02$, where $p < f \leq 0.25$.

Remark 2.1 Note that if $p = p_k$, then $R(p_k) = 1 - h(p_k) = C(p_k)$, i.e. the transmission rate (more precisely, the supremum of achievable rates, see also Example 2.4) coincides with the capacity of the BSC. As we can see from Fig. 2.8, the $E_o(f)$ is very small at the rates close to $R(p)$ (or at the rates close to channel capacity when $p = p_k$). □

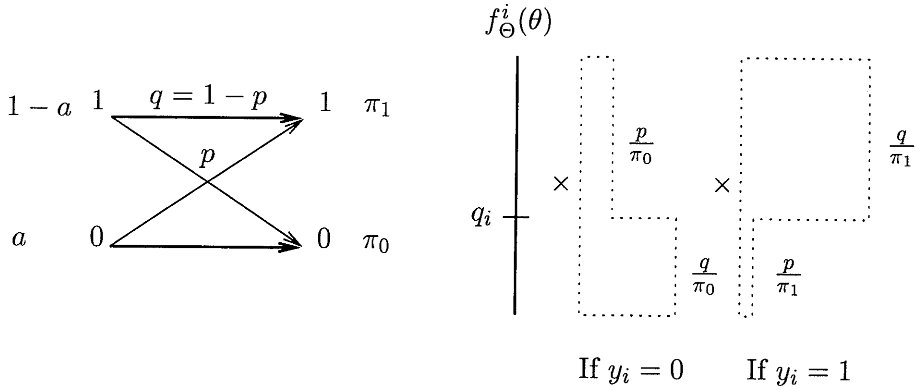


Figure 2.9: Left: a BSC with asymmetric input distribution; Right: the multiplication factors of the receiver distribution.

2.2.3 Recursive coding

The recursive (non-block or sequential) coding method of MRFC schemes for BSC's was proposed in [66] by Schalkwijk and Post. They considered Horstein's scheme for a BSC with the crossover probability p and assumed that 0 is transmitted with probability a instead of 0.5, see Fig. 2.9. The quantiles, the term which the authors used instead of the medians because $a \neq 0.5$, in the $(i+1)^{\mathrm{st}}$ transmission are therefore computed according to

$$\int_0^{q_i} f^i_\Theta(\theta)\, d\theta = a, \qquad i = 0, 1, \ldots .$$

Such an input distribution induces the output distribution

$$(\pi_0, \pi_1) = (p(1-a) + qa,\ pa + q(1-a)) . \tag{2.10}$$

According to Horstein's scheme, the lower and upper parts of the quantile in the receiver distribution must be multiplied by the pairs $(\frac{q}{\pi_0}, \frac{p}{\pi_0})$ or $(\frac{p}{\pi_1}, \frac{q}{\pi_1})$ upon receiving 0 or 1, respectively, see also Fig. 2.9. The probabilities p and a for which the quantiles coincide before and after reception of $10^{(k-1)}$ and $10^{(l-1)}$ subsequences can be computed. In fact, the following conditions, see (5) and (6) of [66], must hold

$$\frac{p}{\pi_1}\left(\frac{q}{\pi_0}\right)^{(k-1)} = 1, \tag{2.11}$$

$$\frac{p}{\pi_0}\left(\frac{q}{\pi_1}\right)^{(l-1)} = 1. \tag{2.12}$$

For a given pair of repetition parameters (k, l), solving equations (2.11) and (2.12) gives $p = p_{kl}$ and $\pi_0 = \pi_{0,kl}$, which results in a regular quantile path of 1-up $(k-1)$-down and 1-down $(l-1)$-up[2]. By using (2.10), one can also obtain the corresponding $a = a_{kl}$.

Recursive encoding

A MRFC scheme which corrects one $0 \to 1$ error by transmitting 0 k-times and corrects a $1 \to 0$ error by transmitting 1 l-times performs similarly to such a Horstein scheme used for a BSC with $p = p_{kl}$ and $a = a_{kl}$. The block diagram of the recursive coding system is shown in Fig. 2.10. The transmitter sends precoded bits continuously and every transmitted 0 or 1 which is received erroneously is repeated k or l times, respectively. Since the forbidden subsequences of the resulting repetition scheme are 10^k and 01^l, the corresponding precoder has an asymmetric FSM (imagine the one shown in the left hand side of Fig. 2.4 but with $(k-1)$ states and $(l-1)$ states for producing consecutive 0's and 1's, respectively).

Recursive decoding

Unlike the previous method, recursive MRFC schemes [66] use a **left-to-right** decoding method. This means that the decoding is realized with a small delay. A recursive decoder consists of an estimation unit, a channel-error removing unit and an inverse-precoding unit, see Fig. 2.10. The inverse precoding is the same for both block MRFC and recursive MRFC. In [91], Veugen addresses different precoding and inverse-precoding procedures. Moreover, a specific precoding method will be considered in this dissertation in Chapter 6. In the following subsections only the estimator and the channel-error remover will be explained.

Estimator with a fixed coding delay

In the recursive decoder with a fixed coding delay, every *transmitted* symbol is estimated based on the following D received symbols, therefore, only a delay of D transmissions is necessary to estimate every transmitted symbol. Let the sequence $\hat{\mathbf{t}}$ be the estimate of transmitted sequence $\mathbf{t}$. To obtain symbol $\hat{t}_i$

[2]Sometimes this terminology will be used to refer to a MRFC scheme with forbidden subsequences 10^k and 01^l.

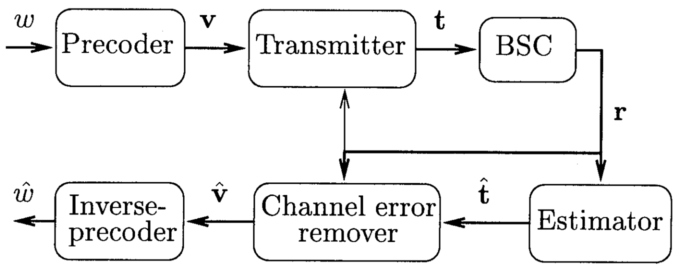


Figure 2.10: Block diagram of the recursive MRFC system.

from $\mathbf{r}_i^{(i+D)}$, $i = 1, 2, \ldots$, there are two kinds of estimators with a fixed coding delay, namely

- right-to-left estimator [91], which proceeds from r_{i+D} towards r_i;
- left-to-right estimator [66], which proceeds from r_i towards r_{i+D}.

The right-to-left estimator will be explained in Subsection 2.3.2 and can be used for all MRFC schemes. A left-to-right estimator is applicable only to the schemes for binary input DMC's. The main advantage of a left-to-right estimator over the right-to-left estimator is its capability for the variable coding delay estimation, as will be explained in the next subsection.

To obtain bit $\hat{t}_i$ from $\mathbf{r}_i^{(i+D)}$, $i = 1, 2, \ldots$, the left-to-right estimator uses a state diagram with integer numbers as its states, see Fig. 2.11. Starting from the 0-state, the estimator walks to the right or to the left if the j^{th} received bit is 1 or 0, respectively, $j = i \ldots i + D$. All transitions towards the 0-state have unit length. All transitions from the 0-state to the right and to the left have the length of $k - 1$ units and $l - 1$ units, respectively. After D steps, the estimate is 0 or 1 depending on whether the walk ends up in a negative or a positive state, respectively. In case that the last state of the walk is the 0-state, a fair coin is flipped.

In the channel-error removing unit a transmission error is detected when the estimated bit differs from the corresponding received bit, i.e. when $\hat{t}_i \neq r_i$. The channel-error remover, see the following example, deletes from the received sequence the erroneous bit and its following $(k-1)$ or $(l-1)$ bits for every detected $0 \to 1$ error or $1 \to 0$ error, respectively.

Example 2.2 Let $\mathbf{r}$ and $\hat{\mathbf{t}}$ denote the received sequence and the estimation of the transmitted sequence, respectively, as indicated in Fig. 2.10. Assume

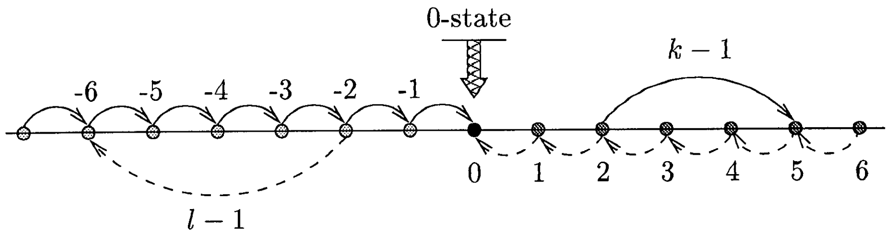


Figure 2.11: State diagram of the left-to-right estimator ($k = 4$ and $l = 5$).

$k = 4$ and $l = 3$ and consider the following outcomes

$$\begin{aligned} \mathbf{r} &= \ldots\ 0\,0\,\underline{1\,0\,\underline{1\,1}\,1}\,1\,1\,\underline{1\,0\,0\,0}\,0\,1\ \ldots \\ \hat{\mathbf{t}} &= \ldots\ 0\,1\,1\,1\,1\,1\,1\,1\,1\,0\,0\,0\,0\,0\,1\ \ldots \\ \hat{\mathbf{v}} &= \ldots\ 0\qquad\qquad 1\,1\qquad\quad 0\,1\ \ldots, \end{aligned}$$

where $\hat{\mathbf{v}}$ represents the output sequence of the channel-error remover. The lines in sequence $\mathbf{r}$ indicate those bits that are removed by the channel-error remover. □

Let τ_0 be the average number of transmissions needed to send 0 over the channel. We have $\tau_0 = 1 + kp\tau_0$, or $\tau_0 = (1-kp)^{-1}$. Similarly $\tau_1 = (1-lp)^{-1}$. As long as $1 - kp > 0$ and $1 - lp > 0$ all channel errors will eventually be corrected. However, due to a limited decoding delay a decoding error is caused by a wrong estimation[3], i.e.

$$P_r = Pr\{\hat{T}_i \neq t_i\},$$

where $\hat{T}_i$ is the random variable denoting $\hat{t}_i$. In this case, a random walk stops in the wrong region at the D^{th} step. Therefore, an estimation error probability can be upper bounded by the probability of the events for which a random walk enters the 0-state after D steps.[4] In [66], the asymptotic error exponent for this estimation error is proved for a BSC with $p = p_{kl}$ to be

$$E_F = \lim_{D\to\infty} \frac{-\log P_r}{D} = \min\{D(\frac{1}{k}\|\pi_1), D(\frac{1}{l}\|\pi_0)\},$$

[3]Note that an estimation error can cause error propagation in the channel-error remover and eventually in the inverse-precoder [91, pp. 120], i.e. a bit error causes an event error.

[4]More precisely, if P_r is the probability of entering the 0-state after D steps, aP_r and $(1-a)P_r$ are the probabilities of wrong estimation depending on whether a 1 or a 0 was decided at the end of the walk.

where the divergence notation of the exponents is due to Veugen [91]. The asymptotic error exponent also gives an upper bound to P_r with a finite delay [66, Theorem 4.2] at $p = p_{kl}$ and when $\frac{1}{l} < \pi_0 = \pi_{0,kl} < \frac{k-1}{k}$, i.e.

$$P_r \leq \frac{k+l}{kl-(k+l)} 2^{-E_F \cdot D}. \tag{2.13}$$

Note that

$$D(\frac{1}{k}\|\pi_1) \overset{(\alpha)}{=} D(\frac{1}{k}\|p) \quad \text{and} \quad D(\frac{1}{l}\|\pi_0) \overset{(\beta)}{=} D(\frac{1}{l}\|p),$$

where (α) and (β) follow from (2.11) and (2.12), respectively.

Estimator with a variable coding delay

Variable coding delay [66] is one of the nice features of the left-to-right decoding method with a left-to-right estimator. A left-to-right estimator uses a state diagram as described in the previous subsection. However, unlike in the fixed coding delay where the estimation is made after D steps of the random walk, in variable coding delay a decision about the transmitted bit is made as soon as the random walk enters an absorbing state. For a 1-up $(k-1)$-down and 1-down $(l-1)$-up scheme the absorbing states are $\{(r+1),(r+2),\dots,(r+k-1)\}$, at which the estimated bit is 1, and $\{-(s+1),-(s+2),\dots,-(s+l-1)\}$, at which the estimated bit is 0.

In the variable coding delay method, a random walk which reaches state $(r+1)$ is in error when in the case of the continuation of the walk, the 0-state would have been reached, see Fig. 2.12. We designate the probability of such an error by $P_V(r+1)$. Note that $a \cdot P_V(r+1)$ is the probability of the erroneous paths entering the 0-state after reaching a positive absorbing state $(r+1)$.

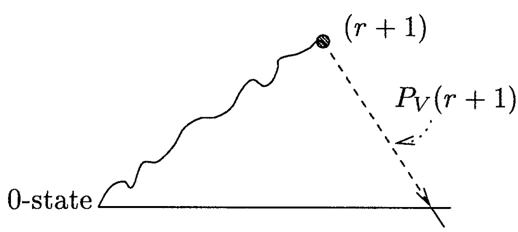


Figure 2.12: An illustration of a random walk absorbed in a wrong state.

Let P_u denote the probability of reaching the 0-state from the state $u > 0$. Since in the region of positive states, the path moves $(k-1)$ units upwards with probability π_1 and it moves one unit downwards with probability π_0, the recurrent equation for P_u can be written as

$$P_u = \pi_1 P_{u+k-1} + \pi_0 P_{u-1}.$$

Assuming that $P_u = \omega^u$, the characteristic equation of the recurrent equation can be written as

$$\omega = \pi_1 \omega^k + \pi_0. \tag{2.14}$$

Let ω_0 denote the unique solution of equation (2.14) in the interval $(0, 1)$. Therefore, $P_u = \omega_0^u$. Hence

$$P_V(r+1) = \omega_0^{(r+1)} \qquad 0 < \omega_0 < 1. \tag{2.15}$$

The error exponent of this error can be written as

$$\bar{E}_V^0 = \lim_{P_V(r+1)\to 0} \frac{-\log(P_V(r+1))}{\bar{D}_1}, \tag{2.16}$$

where $\bar{D}_1$ is the average number of steps needed to reach a positive absorption state. In [66] it is shown that

$$\lim_{r\to\infty} \frac{r}{\bar{D}_1} = d_1 = (k-1)\pi_1 - \pi_0, \tag{2.17}$$

where d_1 represents the drift away from the 0-state in the region of positive states. From (2.15), (2.16) and (2.17) one can obtain

$$\bar{E}_V^0 \quad = \quad -d_1 \log \omega_0 \overset{(\alpha)}{=} D(\pi_1 \| p),$$

where (α) is due to Veugen [91]. Similarly, for $1 \to 0$ error, one can obtain

$$\bar{E}_V^1 = ((l-1)\pi_0 - \pi_1) \log \frac{1}{\tilde{\omega}_0} = D(\pi_0 \| p),$$

where $0 < \tilde{\omega}_0 < 1$ is the root of the characteristic equation

$$\omega = \pi_0 \omega^l + \pi_1.$$

Example 2.3 If $k = l$, then the transmission rate is equal to the channel capacity of a BSC with $p = p_k$, where p_k is the solution of (2.1) for a given k, and $\pi_0 = \pi_1 = \frac{1}{2}$. The estimation error exponent of the fixed coding delay for both $0 \to 1$ and $1 \to 0$ errors is equal to

$$E_F(C) = D(\frac{1}{k}\|\frac{1}{2}) \overset{(\alpha)}{=} D(\frac{1}{k}\|p_k), \qquad k \geq 3, \tag{2.18}$$

where (α) follows from (2.1). The corresponding exponent of the variable coding delay is equal to

$$\bar{E}_V(C) = D(\frac{1}{2}\|p_k), \qquad k \geq 3.$$

Note that $\bar{E}_V(C) > E_F(C)$ because $\frac{1}{2} > \frac{1}{k}$ for $k \geq 3$. □

2.3 MRFC for DMC's

Since Schalkwijk's work in 1971, MRFC schemes have been extended to be applicable for a large class of DMC's. Let $\mathcal{X} = \{1, 2, \ldots, |\mathcal{X}|\}$ and $\mathcal{Y} = \{1, 2, \ldots, |\mathcal{Y}|\}$ be the input and output symbol sets of a DMC, respectively. Becker [4] generalised MRFC for the three following classes of DMC's with $|\mathcal{X}| = |\mathcal{Y}|$. The DMC's of the first group are called *strict-sense symmetric* channels, where the channel transition probability $p_{Y|X}(y|x) = p_{xy} = p$ for any $y \neq x$. The DMC's of the second group are called *wide-sense symmetric* channels, where the channel transition probability p_{xy} depends on the difference between x and y modulo $|\mathcal{X}|$. DMC's with no restriction on channel transition probabilities belong to the third group. Becker showed that his block MRFC schemes are asymptotically optimal for the first two classes of DMC's.

Recently in [91], Veugen further extended MRFC schemes to be used for DMC's with $|\mathcal{X}| \leq |\mathcal{Y}|$. He divided the set $\mathcal{Y}$ into two subsets $\mathcal{X}$ and $\mathcal{Y} - \mathcal{X}$ and considered all outputs of subset $\mathcal{Y} - \mathcal{X}$ as erasures, see Fig. 2.13. In [90], Veugen also proved the optimality of MRFC for these DMC's. We will mostly describe the highlights of Veugen's work in this section.

2.3.1 Block coding

A MRFC scheme for the defined DMC repeats the input symbol $x \in \mathcal{X}$ which is received as $y \in \mathcal{Y}$ k_{xy}-times. Obviously, a correct reception will not be repeated, i.e. $k_{xx} = 0$ for $x \in \mathcal{X}$. If $y \in \mathcal{Y} - \mathcal{X}$, then $k_{xy} = 1$ for each $x \in \mathcal{X}$ because these outputs are erasure outputs known for both decoder and transmitter. When an erasure output occurs, the transmitter repeats the corresponding input symbol. On the other side, the decoder simply ignores

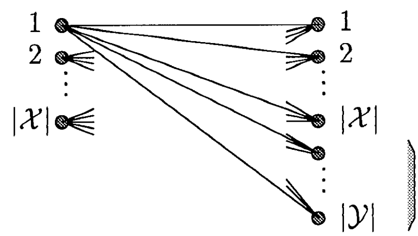


Figure 2.13: A general DMC considered by Veugen in [91].

all erasure output symbols. Consequently, the forbidden subsequences of a general MRFC scheme are $yx^{k_{xy}}$, $x, y \in \mathcal{X}$[5].

In the block MRFC, precoded sequences of length L are transmitted in blocks of N symbols by appending a tail of length $N-L$. There are four kinds of tails discovered by Becker [4] (three kinds) and Veugen [91] (the fourth one). After discarding the erasure outputs, a right-to-left (block) decoder moves from the end towards the beginning of the resulting received block and replaces forbidden subsequences $yx^{k_{xy}}$ with x. When the decoder reaches the beginning of the block, first L symbols of the resulting block constitutes $\hat{\mathbf{v}}^L$, see the block diagram of the system in Fig. 2.5.

Theorem 2.1 (Becker-Veugen) *Let e_{xy} denote the total number of $x \to y$ errors in a block of N transmissions and k_{xy} denote the corresponding repetition parameter, $(x, y) \in (\mathcal{X}, \mathcal{Y})$. As long as*

$$\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} k_{xy}e_{xy} < N - L$$

all channel errors are corrected, provided that a proper tail has been appended, i.e. $\hat{\mathbf{v}}^L = \mathbf{v}^L$.

Proof : See [91, 4]. Q.E.D.

Let the random variables E_{xy}, $(x, y) \in (\mathcal{X}, \mathcal{Y})$, denote the number of $x \to y$ errors in a transmitted block. A block transmission error occurs with probability

$$P_b^N = Pr\{\hat{\mathbf{V}}^L \neq \mathbf{v}^L\} \leq P_o^N = Pr\{\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} k_{xy}E_{xy} \geq N - L\}.$$

[5]The decoder can consider also subsequence yx, $x \in \mathcal{X}$ and $y \in \mathcal{Y} - \mathcal{X}$, as a forbidden subsequence, however, from precoder's point of view such a constraint does not exist.

Furthermore assume $M(L)$ to be the number of precoded sequences of length L. Veugen defines an achievable rate for block MRFC schemes as follows.

Definition 2.3 *A rate R is achievable if, for each sufficiently large block length N, it is possible to find $M(L)$ precoded messages such that $R \leq \frac{\log M(L)}{N}$ and $P_o^N \to 0$ with increasing N.*

Note that if a rate is achievable in the Veugen sense, it is also achievable in the Shannon sense, i.e. if $P_o^N \to 0$ then $P_b^N \to 0$ (as $N \to \infty$). The supremum over all achievable rates is designated by $\hat{R}$.

Example 2.4 A MRFC scheme with repetition parameter k for a BSC with crossover probability p can achieve $R(f) = R_0^k(1 - kf)$ as long as $\frac{1}{k} \leq f < p$, because from (2.8) we have $E_o(f) = D(f\|p) > 0$. Therefore, the supremum rate $\hat{R} = R(p)$ and if $p = p_k$ then $\hat{R} = R(p_k) = C$. □

2.3.2 Recursive coding

The block diagram of a recursive MRFC system for the general class of DMC's is similar to the one shown in Fig. 2.10. A recursive encoder continuously transmits a precoded sequence of unlimited length according to the repetition rules. At the receiver side, only a right-to-left estimator can be used in MRFC systems for nonbinary-input DMC's. Proposed by Veugen in [91], a right-to-left estimator applies the block decoding method to the received sequence $\mathbf{r}_i^{(i+D)}$ to estimate the i^{th} transmitted symbol. That is to move from the end to the beginning of this sequence and replace every $yx^{k_{xy}}$ with x until reaching the beginning of the sequence. The last, i.e. the leftmost, remaining symbol after all replacements is the estimate $\hat{t}_i$. Since we are interested in the leftmost symbol, the right-to-left estimator is simpler than the right-to-left decoder for a block decoding method where one wants to decode all symbols of the block.

Basically, a counter and a symbol register are needed to implement a right-to-left estimator. Let $\mathbf{r'}_i^{(i+D')}$ denote the sequence obtained from $\mathbf{r}_i^{(i+D)}$ by removing all erasure symbols of $\mathbf{r}_i^{(i+D)}$. The following algorithm [91] describes the right-to-left estimator applied on sequence $\mathbf{r'}_i^{(i+D')}$, where $\hat{T}_R$ and C_R represent the contents of the register and the counter, respectively.

Right-to-left estimator

1. Set $j := D'$, $C_R := 1$ and $\hat{T}_R := r'_{i+D'}$.

2 Set $j := j - 1$ and
if $r'_{i+j} = \hat{T}_R$ then $C_R := C_R + 1$;
else $C_R := C_R - (k_{xy} - 1)$, where $x = \hat{T}_R$ and $y = r'_{i+j}$.

3 If $C_R < 1$, then set $C_R := 1$ and $\hat{T}_R := r_{i+j}$.

4 If $j > 0$, then go to the second step.

5 The content of the register is the estimation of t_i, i.e. $\hat{t}_i := \hat{T}_R$.

An event error occurs in recursive decoding when the estimator makes an erroneous estimation, i.e.

$$P_r(i) = Pr\{\hat{T}_i \neq t_i\},$$

where $\hat{T}_i$ is the random variable denoting $\hat{t}_i$. Since it is assumed that $1 - \sum_{y\in\mathcal{Y}} k_{xy}p_{xy} > 0$ for all $x \in \mathcal{X}$, all channel errors are eventually corrected. However, using a fixed coding delay D may result in an estimation error. Veugen [91] recognised two different types of estimation errors for a recursive decoder with finite coding delay D.

1. P_1 is the probability that an erroneous reception cannot be corrected due to insufficient delay. Let us that assume $t_i = x$ is erroneously received as y. As there is insufficient delay to correct this error, the transmitter will send x for the following D transmissions. Consider a Markov process with states $s = 0, 1, \ldots$, which starts at state k_{xy} and has transition probabilities

   $$Pr\{s + k_{xy} - 1|s\} = p_{xy} \qquad \text{for } y \in \mathcal{Y}.$$

   An $x \to y$ error is not corrected with probability P_1^x in D transmissions if the mentioned random walk reaches the 0-state after D steps. Define

   $$E_1^x = \lim_{D\to\infty} \frac{-\log P_1^x}{D}.$$

   Veugen in [91, pp. 62] showed that

   $$E_1^x = -\log_2 \left(\sum_{y\in\mathcal{Y}} p_{xy}\ k_{xy}\ \gamma^{(k_{xy}-1)} \right), \tag{2.19}$$

where $\gamma > 0$ is the solution of

$$\sum_{y \in \mathcal{Y}} p_{xy} \ (k_{xy} - 1)\gamma^{k_{xy}} = 0. \tag{2.20}$$

The exponent of P_1 is defined as

$$E_1 = \lim_{D \to \infty} \frac{-\log P_1}{D} = \min_{x \in \mathcal{X}} E_1^x.$$

2. The other estimation error is caused by certain precoded sequences called the flip sequences. A flip sequence corresponding to symbol x is defined as

$$y_1 x^{(k_{xy_1}-1)} y_2 x^{(k_{xy_2}-1)} \dots y_n x^{(k_{xy_n}-1)} \qquad \text{where } y_i \neq x, \quad i = 1, \dots, n.$$

A few transmission errors at the end of such a sequence drive the estimator to a wrong conclusion, e.g. a single $y \to x$ error which follows the mentioned sequence yields x as the estimated symbol. Let P_2^x be the estimation error probability due to a flip sequence corresponding to symbol x and $P_2 = \max_x P_2^x$. In [91, pp. 132] the asymptotic exponent of P_2^x is given by

$$E_2^x = R_0 + \log \gamma_x - \log p_{xx},$$

where R_0 is the number of information bits per precoded symbol and $\gamma_x > 0$ is the solution of $\sum_{y \neq x} p_{yy} \gamma^{k_{xy}} = p_{xx}$. Therefore, the error exponent of P_2 can be expressed as

$$E_2 = \lim_{D \to \infty} \frac{-\log P_2}{D} = \min_{x \in \mathcal{X}} E_2^x.$$

Define the error exponent of the estimator as

$$E_r = \lim_{D \to \infty} \frac{-\log P_r(i)}{D}.$$

Then $E_r \leq \min\{E_r, E_2\}$. In [91, pp. 63] it is conjectured that $E_r = E_1$ when the corresponding repetition parameters are suitably chosen, see [91, Chapter 5] how to choose the repetition parameters. The following example confirms the conjecture for a special case. In Chapter 4, it is shown that $\max_{x \in \mathcal{X}} E_1^x$ is the asymptotic error exponent for a retransmission strategy in a less restrictive sense (see Rem. 4.5).

Example 2.5 In recursive MRFC for BSC's with the fixed coding delay, the exponent of $0 \to 1$ error probability, which cannot be corrected at the D^{th} step, follows from (2.19) and (2.20) as

$$\begin{cases} q = p(k-1)\gamma_0^k \\ E_1 = -\log(pk\gamma_0^{(k-1)}) \end{cases} \quad \Rightarrow \quad E_1 = D(\frac{1}{k} \| \, p),$$

which agrees with (2.18) for $p = p_k$. More precisely, according to this conjecture E_1 is the exponent of the estimation error if p is in the range where $k =< -\log p >$, where $< a >$ is the closest integer to $a \in I\!R$. □

2.3.3 Achieving channel capacity

In this subsection Veugen's result [91] is briefly mentioned to indicate when the $\hat{R}$ of a MRFC scheme (for a given set of repetition parameters) coincides with the capacity of a DMC.

Theorem 2.2 (Veugen)

1. *Let channel transition probabilities* p_{xy} $(x \in \mathcal{X},\ y \in \mathcal{Y})$, *and repetition numbers* k_{xy} $(x \in \mathcal{X}\ ,\ y \in \mathcal{Y})$ *be given.*
2. *Assume that* $k_{xy} = 1$ *for* $(x \in \mathcal{X},\ y \in \mathcal{Y} - \mathcal{X})$.
3. *Let* $\mathbf{P}^*$ *be* $|\mathcal{X}|$ *by* $|\mathcal{X}|$ *matrix with entries* $p^*_{xy} = p_{xy}$.
4. *Let* π_y $(y \in \mathcal{Y})$ *be the unique capacity achieving output distribution.*

If the channel probabilities satisfy the equations

$$\frac{p_{xy}}{\pi_y}\left(\frac{p_{xx}}{\pi_x}\right)^{(k_{xy}-1)} = 1 \qquad \textit{for each } (x \in \mathcal{X},\ y \in \mathcal{Y}) \tag{2.21}$$

and $\det \mathbf{P}^* \neq 0$, *then there exists a symbol precoding distribution* $\tilde{\mathbf{q}}$ *such that the maximal achievable rate of the corresponding MRFC scheme* $\hat{R}(\tilde{\mathbf{q}})$ *is equal to the capacity of the discrete memoryless channel with channel transition probabilities* p_{xy} $(x \in \mathcal{X},\ y \in \mathcal{Y})$.

Proof : See [91]. Q.E.D.

Conditions (2.21) are equivalent to (2.11) and (2.12) corresponding to a regular median path in MRFC for BSC's. In [91, pp. 68] Veugen says:

Led by the idea of repetition feedback coding ...., the probability that an x was transmitted, should not be altered by receiving $r_T \dots r_{T+k_{xy}} = yx^{k_{xy}}$, i.e.

$$\begin{aligned} \Pr\{t_T = x | r_0 \dots r_{T-1}, r_T = x\} = \\ \Pr\{t_{T+k_{xy}} = x | r_0 \dots r_{T-1}, r_T \dots r_{T+k_{xy}} = yx^{k_{xy}}\} \end{aligned} \tag{2.22}$$

Equation (2.22) should hold for each $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ to obtain a regular "median" path. In order to achieve channel capacity, the channel output probabilities have to be equal to π_y ($y \in \mathcal{Y}$), the unique capacity achieving output distribution. Furthermore, the channel outputs have to be independent. Since $t_{T+k_{xy}} = x$ and $r_T \dots r_{T+k_{xy}} = yx^{k_{xy}}$ implies that $t_T = \dots t_{T+k_{xy}} = x$, condition (2.22) reduces to $\frac{p_{xx}}{\pi_x} = \frac{p_{xy}}{\pi_y}\left(\frac{p_{xx}}{\pi_x}\right)^{k_{xy}}$. It turns out that this necessary condition to achieve capacity is also sufficient.

Example 2.6 Consider a binary asymmetric channel where $\mathcal{X} = \mathcal{Y} = \{0,1\}$ and channel transition probabilities $p_{10} = 1 - p_{11}$ and $p_{01} = 1 - p_{00}$ (where $p_{xy} = p_{Y|X}(y|x)$ and $x, y \in \{0,1\}$). Consider a pair of repetition parameters (k, l) corresponding to forbidden subsequences 10^k and 01^l. A regular median path will be obtained if

$$\frac{p_{01}}{\pi_1}\left(\frac{p_{00}}{\pi_0}\right)^{(k-1)} = 1 \quad \text{and} \quad \frac{p_{10}}{\pi_0}\left(\frac{p_{11}}{\pi_1}\right)^{(l-1)} = 1. \tag{2.23}$$

If $(\pi_0, \pi_1 = 1 - \pi_0)$ is a capacity achieving output distribution, it must satisfy $I(X = 0; Y) = I(X = 1; Y)$ or

$$D(p_{01} \| \pi_1) = D(p_{10} \| \pi_0). \tag{2.24}$$

For a given pair of (k, l), the solution of the three equations in (2.23) and (2.24) determines the binary asymmetric channel, i.e. p_{10} and p_{01} probabilities, for which the capacity can be achieved by a 1-up $(k-1)$-down and 1-down $(l-1)$-up repetition scheme. □

Chapter 3

Feedback Coded Modulation

In this chapter, we consider the transmission of digital information over AWGN channels. The AWGN channel is a model for various physical communication channels, including wire lines and some radio and satellite channels. Using MRFC schemes, we devise a simple coding method for AWGN channels with high Signal to Noise Ratios (SNR's). The feedback channel is assumed to be an AWGN channel as well, therefore, the new method can operate with noisy feedback channels. Since the resulting method is a combination of coding and modulation, it is referred to as Feedback Coded Modulation (FCM) in this dissertation. This terminology is analogous to that of the famous Trellis Coded Modulation (TCM) method. TCM is also explained briefly in Section 3.2 to set a benchmark for evaluating the decoding complexity of the FCM. Let us first describe the channel model.

3.1 Discrete input AWGN channels

An ideal waveform AWGN channel is a channel whose output signal $y(t)$ can be related to its input signal $x(t)$ by

$$y(t) = x(t) + n(t),$$

where $n(t)$ is a Gaussian random process with two-sided power spectral density $N(f) = \frac{N_o}{2}$. Orthogonal modulation techniques, see Refs. [96, 63, 74], convert such a waveform channel into a (discrete-time) vector AWGN channel. A D-dimensional vector AWGN channel is a channel with input and output vectors $\mathbf{x}^D = (x_1, x_2, \dots, x_D)$ and $\mathbf{y}^D = (y_1, y_2, \dots, y_D)$, respectively, where $x_d, y_d \in I\!R$, $d = 1, \dots, D$. These input and output vectors are related by

$$\mathbf{y}^D = \mathbf{x}^D + \mathbf{n}^D, \tag{3.1}$$

where $\mathbf{n}^D = (n_1, n_2, \ldots, n_D)$ represents the noise vector. Let X_d, Y_d and N_d denote the input, output and noise random variables of the d^{th} component or subchannel of the vector channel, respectively, $d = 1, \ldots, D$. For the mentioned channel $N_d \sim N(0, \sigma^2)$, where $\mathbb{E}[N_d^2] = \sigma^2 = \frac{N_o}{2}$ and the covariance matrix of the noise vector is given by

$$\Delta = \frac{N_o}{2}\mathbf{I},$$

where I is the $D \times D$ identity matrix. The (information theoretic) capacity of the d^{th} subchannel is calculated from Def. 1.15 as

$$C_d = \max_{f_X : \mathbb{E}[X_d^2] \leq E_d} I(X_d; Y_d) = \frac{1}{2}\log(1 + \frac{E_d}{\frac{N_o}{2}}), \tag{3.2}$$

bits per dimension. This capacity is attained when the input to the d^{th} subchannel has a normal distribution, i.e. $X_d \sim N(0, E_d)$. If the total energy per transmitted signal, defined as $\sum_{d=1}^{D} \mathbb{E}[X_d^2] = \sum_{d=1}^{D} E_d$, is constrained to E, the water filling argument [16, pp. 252] shows that

$$E_1 = \ldots = E_D = \frac{E_s}{D}, \tag{3.3}$$

which is also evident from the symmetry of the subchannels. From (3.2), (3.3) and the noise samples of subchannels being independent, the total capacity per transmission or the capacity of the vector channel can be written as

$$C = D\ C_d = \frac{D}{2}\log(1 + \frac{E_s}{D\frac{N_o}{2}}) \qquad \text{bits per transmission.} \tag{3.4}$$

Definition 3.1 (Signal to noise ratio (SNR)) *For the vector channel, the SNR is defined as the ratio of the average energy per transmitted signal to the average energy of the D-dimensional noise, i.e.*

$$SNR = \frac{E_s}{D\sigma^2} = \frac{E_s}{D\frac{N_o}{2}} = \frac{E_d}{\frac{N_o}{2}} = SNR_d. \tag{3.5}$$

Often, $10\log_{10} SNR$ *is considered and the resulting value is expressed in dB units.*

Assuming that the bandwidth of the waveform channel is limited to W and the input signal is subjected to power constraint P, we can also write

$$\text{SNR} = \frac{P}{N_o W}.$$

In general the input vector of a vector AWGN channel is a signal point from D-dimensional Euclidean space $\mathbb{R}^D$, i.e. $x_d \in \mathbb{R}$, $d = 1, \ldots, D$. However, in a discrete-input vector AWGN channel, the d^{th} component of the input vector can take one of the L-equiprobable values from

$$\mathcal{X}' = \{\pm\frac{\gamma}{2}, \pm\frac{3\gamma}{2}, \ldots, \pm\frac{(L-1)\gamma}{2}\} \qquad \text{(L even)},$$

where γ is the *minimum Euclidean distance* between two signal points. The set of input signal points of the discrete-input vector channel is called *signal constellation* and denoted by $\mathcal{X} = \mathcal{X}'^D$. This signal constellation is a D-dimensional cubic. In Fig. 3.1 the one and two dimensional signal constellations are depicted, which are used in Pulse Amplitude Modulation (PAM) and multilevel Quadrature Amplitude Modulation (QAM), respectively.

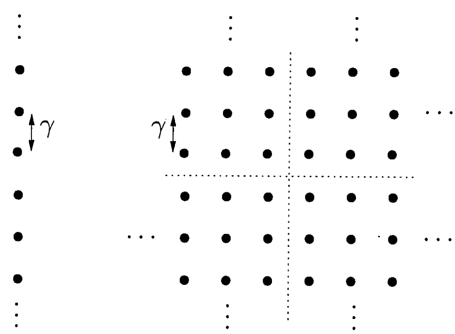


Figure 3.1: Signal constellations of PAM and QAM ($D = 1$ and $D = 2$).

Now the random variable X_d can uniformly take on a discrete value, therefore, the average signal energy per dimension can be written as

$$E_d = \mathbb{E}[X_d^2] = \frac{1}{L} \sum_{x \in \mathcal{X}'} x^2 = \frac{L^2 - 1}{12} \cdot \gamma^2. \tag{3.6}$$

The average energies of the transmitted signal and noise vector are $E_s = DE_d$ and $D\sigma^2 = D\frac{N_o}{2}$, respectively. Note that $\text{SNR} = \text{SNR}_d$ for a D-dimensional cubic constellation.

The described vector channel is a COMC whose inputs are from a finite alphabet, i.e. it has discrete inputs and continuous outputs. In Appendix B a procedure is outlined to numerically evaluate the capacity, or more precisely the average mutual information, for such a channel achieved with an uniform input distribution. Fig. 3.2 shows the capacity achieved with equiprobable $|\mathcal{X}|$-QAM (for $|\mathcal{X}| = 4, 16, 64$ and 256) as a function of SNR. Moreover, as a reference and the ultimate limit, the corresponding capacity of the vector AWGN channel is plotted from (3.4), which is achieved by a Gaussian input distribution. The points at which a signal error probability of 10^{-6} is achieved are shown in Fig. 3.2 as well.

In the low-SNR's an equiprobable binary input alphabet is almost optimal to achieve the channel capacity [26]. In the high-SNR's, however, the capacity of the equiprobable $|\mathcal{X}|$-QAM constellation asymptotically approaches a straight[1] line parallel to the capacity of the vector AWGN channel, see Fig. 3.2. The gap of $\frac{\pi e}{6}$ or 1.53 dB is due to using a uniform rather than a Gaussian distribution over the input signal set [26]. Constellation shaping techniques are used to gain the remaining 1.53 dB by producing a Gaussian-like distribution for every sequence of a fixed number of QAM signals, see Subsection 3.6.1 for an explanation.

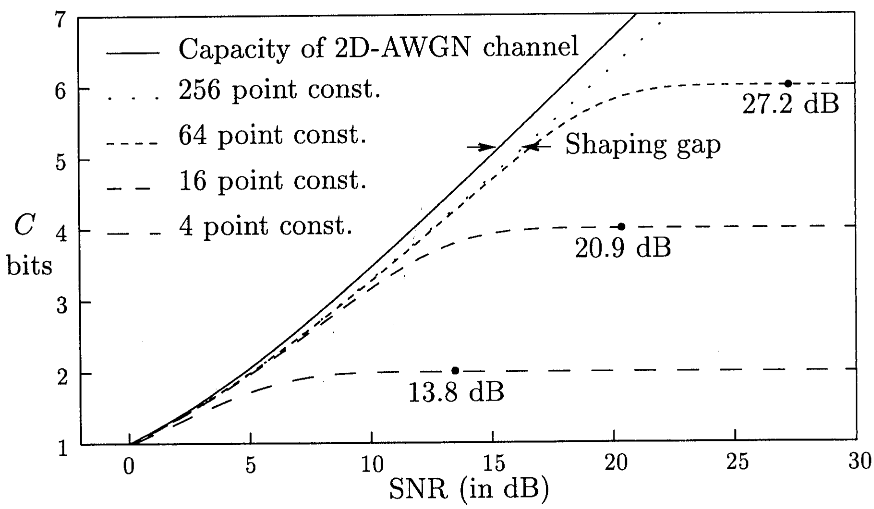


Figure 3.2: The capacities of two dimensional 4, 16, 64 point constellations.

[1]When the SNR is expressed in dB.

3.1.1 Union bound

Assume an uncoded (coded) sequence of n signals denoted by $(\mathbf{x}^D)_i^{i+n-1} \triangleq \mathbf{x}^{D'}$ is transmitted through the vector AWGN channel, where $D' = nD$. Since the received sequence is corrupted by an additive Gaussian signal $\mathbf{n}^{D'} = \mathbf{y}^{D'} - \mathbf{x}^{D'}$, maximum likelihood (ML) decoding is equivalent to minimum distance decoding [26]. The probability for the uncoded (coded) sequence $\mathbf{x}^{D'}$ to be decoded as an uncoded (coded) sequence $\mathbf{z}^{D'}$ is a function of $d(\mathbf{z}^{D'}, \mathbf{x}^{D'})$, the Euclidean distance between uncoded (coded) sequences $\mathbf{x}^{D'}$ and $\mathbf{z}^{D'}$. More precisely,

$$Pr\{\hat{W} = \mathbf{z}^{D'} | W = \mathbf{x}^{D'}\} = Q(\frac{d(\mathbf{z}^{D'}, \mathbf{x}^{D'})}{2\sigma})$$

where random variables $\hat{W}$ and W denote the transmitted and decoded D'-dimensional sequences, respectively, and $Q(\cdot)$ represents the Gaussian error integral of

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{y^2}{2}} dy. \tag{3.7}$$

The error probability of ML decoding is the probability that the received vector sequence $\mathbf{y}^{D'}$ is closer to any $\mathbf{z}^{D'}$ in the constellation but other than to $\mathbf{x}^{D'}$ and can be upper bounded by

$$Pr\{\text{error} \mid W = \mathbf{x}^{D'}\} \leq \sum_{\mathbf{z}^{D'} \neq \mathbf{x}^{D'}} Q(\frac{d(\mathbf{z}^{D'}, \mathbf{x}^{D'})}{2\sigma}).$$

The average probability of error over all uncoded (coded) sequences $\mathbf{x}^{D'}$ is upper bounded by the *union bound*

$$P_e \leq \sum_{\gamma} N_\gamma Q(\frac{\gamma}{2\sigma}),$$

where N_γ is the average number of uncoded (coded) sequences at distance γ from a sequence. The Gaussian error integral function $Q(x)$ decays exponentially with respect to x. Therefore, if N_γ does not increase too rapidly with γ, then the union bound is dominated by its initial terms corresponding to the smallest γ. The **union bound estimate** then becomes

$$P_e \approx N_f Q(\frac{\gamma_{fed}}{2\sigma}). \tag{3.8}$$

Here γ_{fed}, the so-called **free Euclidean distance**, is the minimum Euclidean distance between uncoded (coded) sequences and N_f indicates the average number of nearest neighbour uncoded (coded) sequences at distance γ_{fed} from a given uncoded (coded) sequence. This approximation expresses the fact that at high SNR's the probability of error events associated with a distance larger than γ_{fed} becomes negligible.

Example 3.1 In an uncoded constellation (the one defined in the previous section)

$$\gamma_{fed} = \gamma \quad \text{and} \quad N_f = D \cdot \frac{2(L-2)+2}{L} = D\frac{2(L-1)}{L}. \;\square \tag{3.9}$$

In the following section the union bound estimate is used to depict the error performance of an uncoded 2-dimensional signal constellation.

3.1.2 SNR normalization

Assume that a coding scheme delivers R_t information bits per transmission (per D-dimensions) at given SNR_t. From Shannon's famous formula (3.4), a vector AWGN channel requires a SNR of SNR_{Sh} in order to have a capacity equal to R_t bits, i.e. $R_t = \frac{D}{2}\log(1+\text{SNR}_{Sh})$ or

$$\text{SNR}_{Sh} = 2^{\frac{2R_t}{D}} - 1.$$

To normalize the operating SNR of any scheme, we divide SNR_t by SNR_{Sh}, i.e.

$$\text{SNR}_n = \frac{\text{SNR}_t}{\text{SNR}_{Sh}} = \frac{\text{SNR}_t}{2^{\frac{2R_t}{D}} - 1} = \frac{\text{SNR}_d}{2^{2R_d} - 1}, \tag{3.10}$$

where $R_d = \frac{R_t}{D}$ is the rate per dimension of the given coding scheme. For a capacity achieving scheme $\text{SNR}_n = 1$ (0 dB). This kind of normalization is called *normalization with respect to the Shannon rate.* In the high-SNR channels, $\text{SNR}_n > 1$ is the fundamental figure of merit of uncoded and coded modulation schemes, which specifies how far a given scheme operates from the Shannon limit [26]. In Fig. 3.3 the probability of symbol error $P_{s,uc}$ for uncoded equiprobable 4, 16, 64 point QAM's are depicted as a function of SNR_n from (3.8) using (3.9). At $P_{s,uc} = 10^{-6}$, an uncoded QAM operates about 9 dB away from the Shannon limit, therefore, a coding gain of 9 dB can potentially be attained.

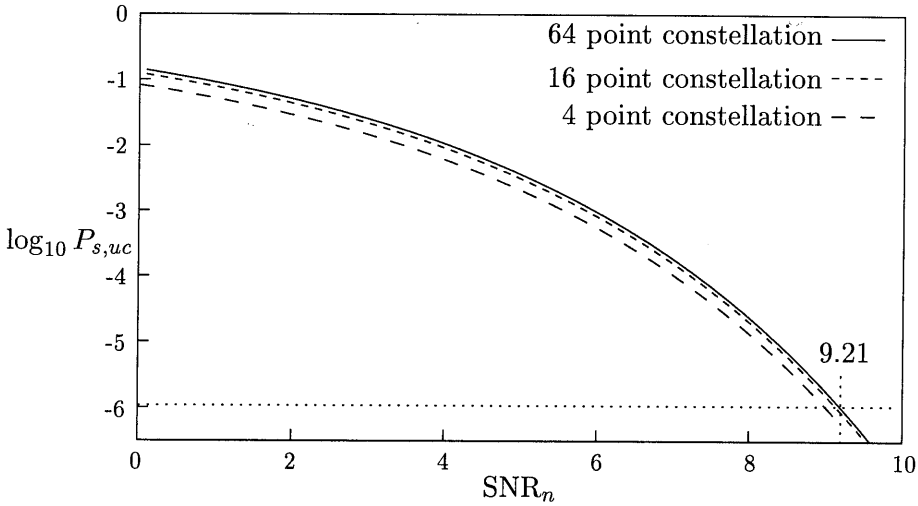


Figure 3.3: The signal error probability for 2-dimensional constellations in terms of SNR_n.

Remark 3.1 Note that the union bound and its approximation blow up at cutoff rate R_0. For high-SNR channels, the cutoff rate corresponds to $\text{SNR}_n = \frac{4}{e}$ (1.68 dB) [26]. □

3.2 Trellis coded modulation

In the past 20 years, a series of advances has occurred that now makes it possible to state that the channel capacity of a band-limited Gaussian channel with high SNR's can be approached quite closely, not just in theory as Shannon indicated, but also in practice. The most striking of these advances was the invention of TCM by Ungerboeck [86]. In this section we are going to explain TCM in order to set a benchmark relative to which we can judge the decoding complexity of our new scheme.

In a conventional system, used before introducing TCM, the channel encoder and modulator were two separate units. The channel encoder added redundant symbols to its input information symbols and the modulator mapped the outputs of the channel encoder into signal points of a D-dimensional constellation. Because the added redundancies in the channel coding unit appeared as extra transmissions, this method was not an efficient way to transmit data via a band-limited channel in which the number of transmissions per second is limited.

Example 3.2 Consider a binary encoder of rate $R = \frac{3}{4}$ and a constellation of 16 signal points. In the channel encoder, 3 information bits are encoded into 4 coded bits which are used to choose a signal point from the constellation. Thus, every transmitted signal represents 4 coded bits or 3 information bits. In the following we show how using TCM makes it possible to transmit 4 information bits per transmission and yet have a good error correcting capability. □

Unlike in conventional systems, in TCM channel encoding and signal mapping are combined in order to directly maximize the overall free distance between coded signal sequences. Ungerboeck took an elegant approach to compensate for the rate loss without transmitting fewer information bits per transmission. Basically, two important techniques, namely: signal set expansion and set partitioning led to a significant gain in TCM as compared to uncoded modulation. In this section these two techniques are explained and then the coding process of TCM is briefly outlined.

3.2.1 Signal set expansion

The first and the most essential new concept of TCM was to use the signal set expansion technique to provide the required redundancy for coding. To make it possible, the number of signal points in the constellation is increased. For example, the uncoded constellation of Example 3.2 contains 16 signal points where each signal can potentially transfer 4 bits. To transmit also 4 information bits per signal in TCM, 4 information bits must enter the encoder-modulator unit in every transmission interval. If the encoder adds one redundant bit, then we need $2^{4+1} = 32$ signal points in the constellation. This means that the original signal constellation has to be enlarged. *To keep the average energy per transmitted signal constant*, the extra points are uniformly placed among the original signal points[2], see Fig. 3.4. In this way, the minimum distance between two adjacent signal points is reduced from γ to $\gamma_0 = \frac{\gamma}{\sqrt{2}}$ in a 2-dimensional Euclidean space. In general one can show that doubling the number of signal points in an D-dimensional constellation decreases the minimum signal distance from γ to $\gamma_0 = 2^{-\frac{1}{D}}\gamma$, provided that the average signal energy is kept constant. So far, we have worsened the minimum

[2]From another point of view, having the extra points can be considered as the extension of the boundaries of the original constellation towards outside without changing the minimum distance between signal points. In this interpretation, the average transmitted energy increases and this increase must be taken into account in computing the overall coding gain. This gain normalization is not needed in our interpretation mentioned in the text above.

distance of the (original) uncoded constellation. Subsequently, however, the applied code increases the minimum distance (the free Euclidean distance) between any two *sequences* of signal points from the enlarged constellation relative to that of the uncoded constellation.

3.2.2 Set partitioning

The second technique used in TCM is mapping by set partitioning. The key point in partitioning of a constellation is to find subsets of the constellation that are similar and the signal points inside each partition are maximally separated. Starting from the extended constellation, we partition it into two subsets that are congruent and the points within each partition are maximally separated. Then apply the same principle to each partition and continue. The point at which the partitioning is stopped depends on the code that we are using, which will be explained in the following subsection.

Example 3.3 Consider Example 3.2 in which the original 16 point signal constellation is extended to the 32 point constellation. In Fig. 3.4, the 32 point constellation is partitioned in the first stage into two congruent partitions denoted by B_0 and B_1 with the minimum partition distance of $\gamma_1 = \sqrt{2}\gamma_0 = \gamma$, where γ and γ_0 are the minimum signal distances in the original constellation and in the extended constellation, respectively. Now we further partition B_0 and B_1 to obtain C_0, C_1, C_2, C_3. The inner partition distance now has increased to $\gamma_2 = 2\gamma_0 = \sqrt{2}\gamma$. We go one step further to obtain eight partitions, each containing four points. The corresponding subsets are denoted by D_0 through D_7, for which the minimum signal distance is $\gamma_3 = \sqrt{8}\gamma_0 = 2\gamma$. □

3.2.3 Coded modulation

In TCM a code is used to reliably transmit *the sequence of partitions* while the signal points inside each partition are generally sent without encoding. Basically, the sequence of partitions can be encoded using block or convolutional (or Turbo) codes. Due to the existence of a simple and optimal soft decision decoding algorithm for convolutional codes (the Viterbi algorithm), it has been used mostly with convolutional codes. When used with convolutional codes the resulting coded modulation is known as trellis coded modulation. The trellis terminology is used because these schemes can be described by a state transition (trellis) diagram like that of binary convolutional codes. To be more precise, the term TCM is used when a general trellis code is used.

The block diagram for a TCM (the encoder-modulator unit) is shown in Fig. 3.5. The frame of k_1+k_2 input message bits enters the encoder-modulator

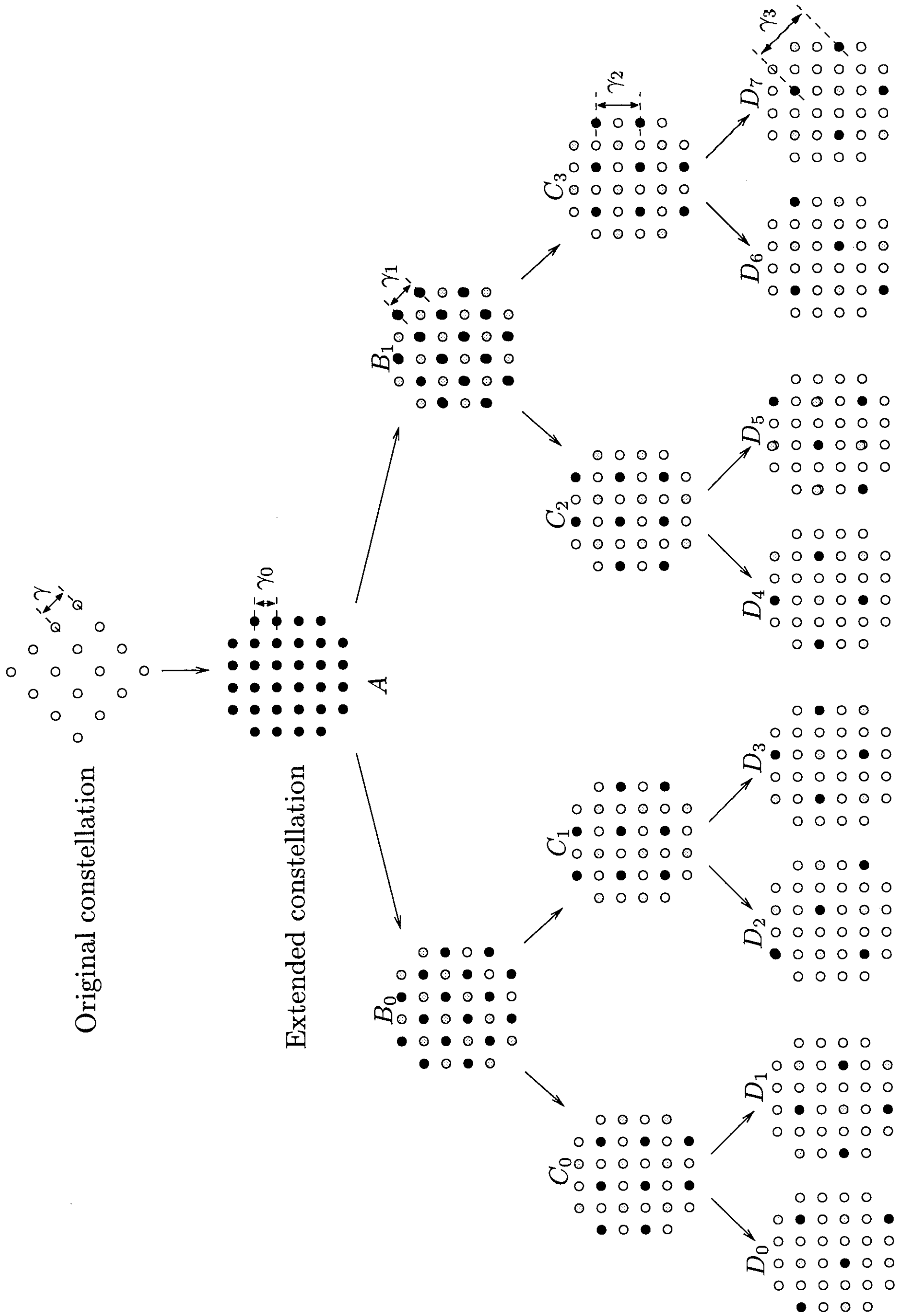


Figure 3.4: Signal set expansion and corresponding set partitioning.

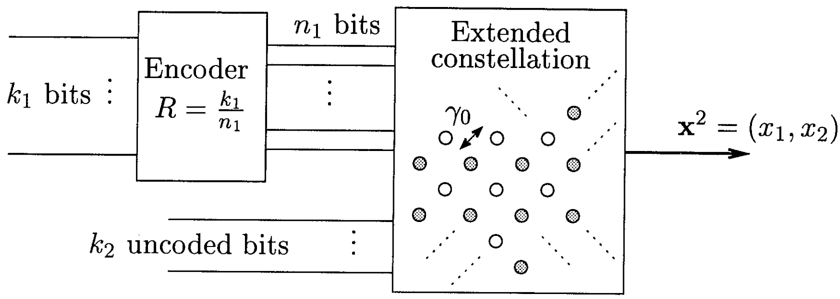


Figure 3.5: A block diagram of TCM, the encoder-modulator unit.

unit. Using an encoder of rate $R = \frac{k_1}{n_1}$, k_1 bits of this frame are encoded into n_1 bits which are used to choose one of 2^{n_1} partitions. After choosing the partition, the remaining k_2 bits are used to choose one of the points in the chosen partition. I.e. the extended constellation should be partitioned into 2^{n_1} partitions or subsets, where each partition contains 2^{k_2} signal points. This gives us a rule for determining how large a constellation is required and how many steps must be taken in the partitioning of this constellation. Doubling the size of the original constellation provides enough redundancy [86] where the encoder has to add one redundant bit, i.e. $n_1 = k_1 + 1$.

Example 3.4 For the constellation of Example 3.3 there are eight partitions $D_0, \ldots, D_7$, see Fig. 3.4. Thus $n_1 = 3$, $R = \frac{2}{3}$ and $k_1 = k_2 = 2$. The 2^{k_2} signal points in each subset are so far apart that there is no need to encode them. For a given partition, $\gamma_3 = \sqrt{8}\gamma_0 = 2\gamma$ which already yields $10 \log_{10} 2^2 = 6$ dB improvement with respect to an uncoded system. Hence for most applications, it is only necessary to encode the sequence of subsets and decode them correctly at the receiver. □

The trellis diagram of TCM is similar to that of a convolutional code. However, there are 2^{k_2} *parallel paths* connecting every two states of the trellis in TCM. A bunch of parallel branches corresponds to a partition of the constellation and each branch in the bunch corresponds to a signal point in the partition. Fig. 3.6 shows a trellis with a few parallel paths, each designated by the name of the corresponding partition, and two paths for the sequence of partitions. The sequence of the partitions follows a path on the trellis according to the values of the frames of k_1 input bits. The performance of TCM can well be determined from the free Euclidean distance, γ_{fed}, between any two

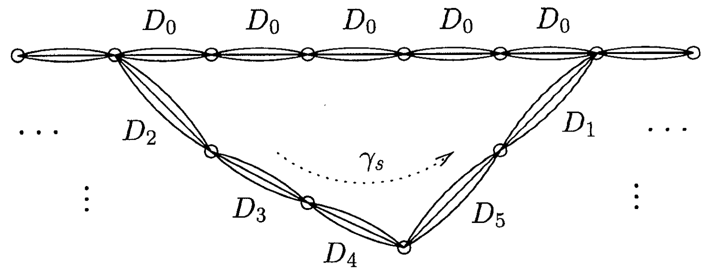


Figure 3.6: Two schematic paths of partitions on the trellis of a TCM.

paths originating from a node and merging into another node of the trellis. There are two types of paths which affect γ_{fed} in TCM.

- The first kind is the parallel paths for which γ_p is the minimum Euclidean distance between signal points in a partition. For example, in Fig. 3.4 we have $\gamma_p = \gamma_3 = 2\gamma$.
- The other kind of paths can be seen on the trellis of a TCM if we consider every set of parallel paths as a single path represented by its partition index. In fact, now we are talking about the Euclidean distance of a classic convolutional code. Let γ_s denote the minimum Euclidean distance between two paths of this kind. For example, in the trellis of Fig. 3.6 the Euclidean distance between two indicated paths is $\gamma_s^2 = d^2(D_0, D_2) + d^2(D_0, D_3) + d^2(D_0, D_4) + d^2(D_0, D_5) + d^2(D_0, D_1)$, where $d(D_i, D_j)$ indicates the smallest Euclidean distance between a signal point of partition D_i and a signal point of partition D_j.

Finally, the free Euclidean distance is determined from

$$\gamma_{fed} = \min\{\gamma_p, \gamma_s\}.$$

This γ_{fed} indicates the amount of improvement that we have achieved by coding and set partitioning when the minimum signal distance is increased from γ in the uncoded constellation to γ_{fed} with TCM (assuming that the average signal energy does not change by the signal set expansion), see also Subsection 3.5.3.

3.3 Feedback coded modulation

FCM is a coded modulation method for AWGN channels, which extends the domain of MRFC to AWGN channels. The main motive behind this extension was to design a feedback coding scheme that is able to operate with a noisy feedback channel. In fact, here we assume that both forward and feedback channels are AWGN channels with the same SNR. Using the MRFC schemes for BSC's, FCM yields a simple solution to the problem of such a noisy feedback channel. This new scheme is similar to TCM in the sense that it uses the signal set partitioning technique, where the signal points inside each partition are transmitted without coding. On the other hand, FCM is similar to conventional modulation methods in the sense that there is no constellation expansion and adding redundancy is accomplished by executing more transmissions. However, the amount of redundancy (the rate loss) is very small in FCM as we compare it with that of conventional modulation methods due to set partitioning and operating the MRFC schemes at the highest possible rate. In the following sections, this coded modulation method is introduced, see also [78] and its decoding complexity is compared to that of the TCM. Some related features of FCM are also outlined at the end of the chapter.

Let us consider the signal constellation $\mathcal{X} = \mathcal{X}'^D$ as defined in Subsection 3.1, where γ is the minimum distance between signal points along each dimension. Since the signal space is assumed to be orthogonal, γ is also the minimum distance of signal points in the D-dimensional Euclidean space. The signal points in $\mathcal{X}$ (or signal coordinate points in $\mathcal{X}'$) are equiprobable, i.e. $Pr\{\mathbf{X}^D = \mathbf{x}^D\} = \frac{1}{L^D}$ and $Pr\{X_d = x\} = \frac{1}{L}$, for all $\mathbf{x}^D \in \mathcal{X}$ and $x \in \mathcal{X}'$.

3.3.1 Encoder for FCM

We explain the encoding and decoding of FCM by using the 16 point 2-dimensional constellation of Example 3.2. In Fig. 3.7 the schematic diagram of the system is depicted where one MRFC scheme (for BSC's) is used in each dimension. The output bit of the first MRFC scheme, represented by a/b, chooses every other row of the constellation and the output bit of the second MRFC scheme, denoted by c/d, chooses every other column of the constellation. Therefore, the two a/b and c/d bits determine one of the $a-c$, $a-d$, $b-c$ and $b-d$ partitions, see Fig. 3.8. Here bits a/b and c/d are called the **partitioning bits**. The remaining two uncoded bits choose one signal point from the selected partition. As we see from Fig. 3.8 and Fig. 3.4, for the same minimum distance in each partition, the number of the partitions of the FCM is half of that of the corresponding TCM since here we do not double the number of the constellation points. At the receiver, a simple hard decision operation is done on each coordinate of the received signal to determine the corresponding received partitioning bit, i.e. being a or b, and c or d. The received partitioning bits are fed back via a noiseless and delayless feedback channel to the transmitter in order to be used in their corresponding encoders.

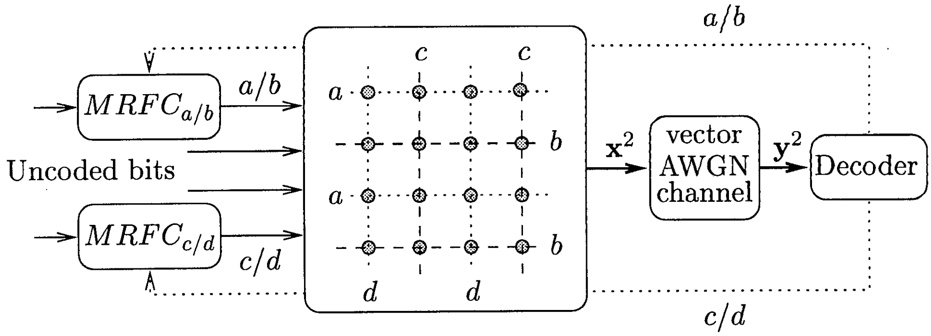


Figure 3.7: The block diagram of the FCM with a 2-dimensional 16-point constellation.

In the general case of D-dimensional signalling, the encoding is accomplished by using one MRFC per dimension. In a dimension, the MRFC scheme yields the partitioning bit which specifies the corresponding coordinate to be from one of the two congruent subsets $B_a = \{-\frac{(L-1)\gamma}{2}, \frac{(L-3)\gamma}{2}, \ldots, (-1)^{\frac{L}{2}}\frac{\gamma}{2}\}$ or $B_b = \mathcal{X}' - B_a$. The chosen subsets for all coordinates determine the transmitted partition. Then, the uncoded bits are used to pick up a D-dimensional signal point from the chosen partition.

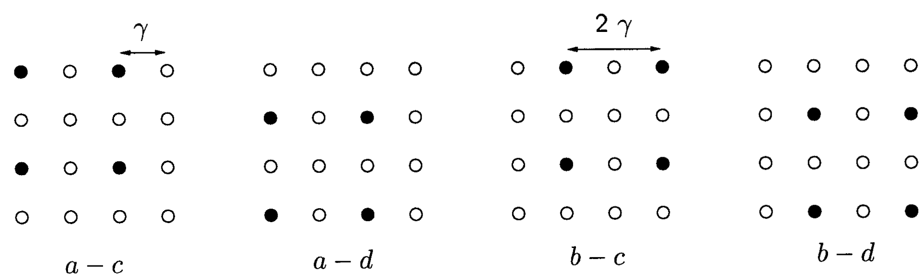


Figure 3.8: Set partitioning of a 16-point constellation in FCM ($D = 2$).

3.3.2 Decoder for FCM

The decoding process of FCM is explained in three parts: storing the received data, estimating the transmitted partitions and extracting the information bits.

Receiver buffer

The coordinates of the received vector must be stored in separate buffers until the code words of the corresponding MRFC schemes are decoded. For each coordinate, a buffer of N or $D_r + 1$ units is needed to store N or $D_r + 1$ received points of the coordinate, depending on whether a block or a recursive MRFC method is used, respectively. To store each coordinate of an input signal point, $x_d \in \mathcal{X}'$, $\log(L)$ bits are needed. The corresponding output coordinate $y_d = x_d + n_d$ is a real number, however, it is not necessary to be stored with a high precision. In fact, one bit per dimension is the cost of real (soft) valued information at the output, which is introduced by the AWGN. Therefore, $\log(L)+1$ bits are enough to store each received coordinate point. In the following we describe how an output coordinate point, e.g. corresponding to partitioning bit a/b, can be stored. For other coordinates, the same rules apply.

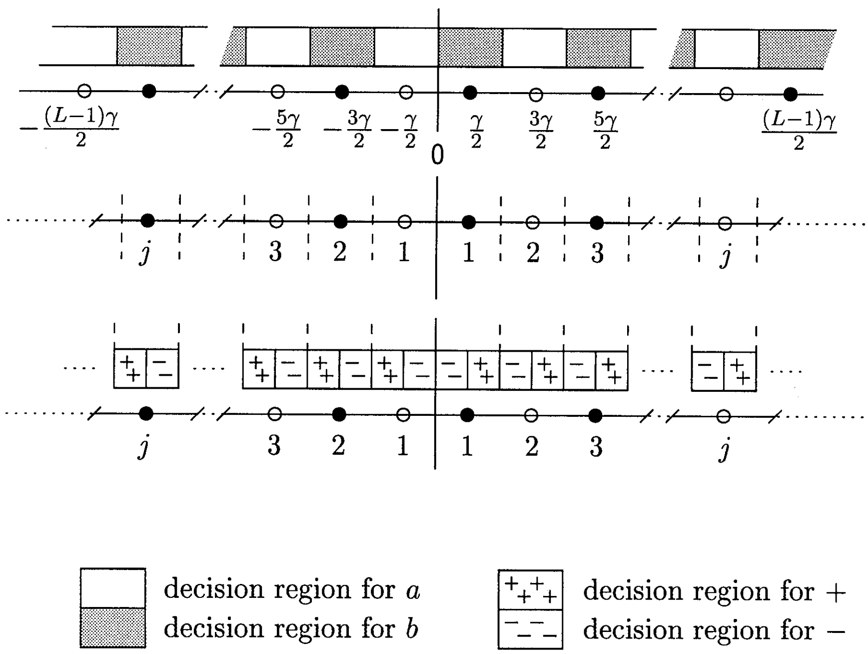


Figure 3.9: The decision regions for each dimension of the received signal.

1. One bit, denoted by a/b, records the partitioning bit of the received coordinate point. As shown in the upper part of Fig. 3.9, bit a/b indicates

that the received coordinate point is either in the decision region of the B_a signal coordinate points or in the decision region of the B_b signal coordinate points, respectively.

2. The amplitude of signal coordinate points is indicated by *index* number $j = 1, \ldots, \frac{L}{2}$. In each dimension, two signal coordinate points with the same index are from two different partitions. Note that the left most signal coordinate point corresponds to partitioning bit a. As shown in the middle part of Fig. 3.9, if signal coordinate point $x_d \in B_a$, then signal coordinate point $x'_d = -x_d$ is in B_b. We can imagine that when the uncoded information chooses a specific amplitude (or index number) in a dimension, the corresponding partitioning bit chooses the sign of the index. The reason for this indexing method will be discussed in Subsection 3.6.1.

3. One extra +/− bit per dimension is needed to record the position of the received coordinate point in the upper or lower part of the decision region of the corresponding a/b bit. In the lower part of Fig. 3.9 the decision regions of +/− bit are indicated. As we see, the interval of length γ, centered at a signal coordinate point, is divided into two parts. The half with higher absolute value is assigned + and the other part is assigned −. The +/− bit will be used to modify the index of the received coordinate when a transmission error is detected in the corresponding partitioning bit, see Table 3.1.

The three points mentioned above indicate that we have to store

$$\log(\frac{L}{2}) + 1 + 1 = \log(L) + 1 \quad \text{bits per signal per dimension,}$$

which asserts our claim.

Estimation of the partitioning bit

To decode uncoded information bits, the receiver must estimate every transmitted partition or, in other words, the receiver must estimate every transmitted partitioning bit. The right-to-left or left-to-right estimators of the recursive decoder directly estimate the transmitted partitioning bits. An upper bound to such estimation error at the certain crossover probabilities of the partitioning bit is given in (2.13). Also the block decoder can yield the estimate of the transmitted block of partitioning bits as shown in the following theorem.

Theorem 3.1 *In a block MRFC scheme (see Fig. 2.5) where an appropriate tail of $N-L$ symbols is used, the receiver can exactly determine $\mathbf{t}^N$, provided that $\hat{\mathbf{v}}^L = \mathbf{v}^L$.*

Proof : The proof is straightforward. Knowing $\mathbf{v}^L$ and accordingly the tail $z_{L+1}\ldots z_N$, the decoder knows the codeword that the transmitter intended to send. Having also the received sequence $\mathbf{r}^N$, the decoder is able to reconstruct the pertaining transmission process and obtain the transmitted sequence $\mathbf{t}^N$ precisely. Q.E.D.

According to Theorem 2.1 a sufficient condition for $\hat{\mathbf{v}}^L = \mathbf{v}^L$ in the MRFC for a BSC is that $ke < N - L$, where e represents the number of errors in a block. Therefore, (2.9) is also an upper bound to this block estimation error probability, which becomes exponentially tight for $Pr\{kE \geq N - L\}$ as N increases.

Extracting the information bits

After decoding the MRFC schemes, we have acquired the part of information associated with the partitioning bits, i.e. the precoded sequences corresponding to the partitioning bits can be obtained and then inverse-precoded, see Figs. 2.5 and 2.10. Correctly estimated partitioning bits enable us to know the transmitted partitions. In a given partition, we find the closest signal point of the partition to the received point. In this way, information associated with uncoded bits are extracted using ML detection. Here we explain a simple technique to obtain the closest signal point of the estimated partition to the received point. Let $i = 1, 2, \ldots$ denote the i^{th} received signal point.

- When no error is detected in the partitioning bits of the i^{th} received signal point, i.e. $\hat{t}_{id} = r_{id}$, $d = 1, \ldots, D$, the corresponding stored index numbers and partitioning bits determine the closest signal point of the partition to the received point.
- When a transmission error is detected in a partitioning bit, i.e. $\hat{t}_{id} \neq r_{id}$, in addition to changing the received partitioning bit to the correct bit, the corresponding index number stored in the receiver buffer must be modified according to Table 3.1. Such a modification gives the most likely neighbour coordinate point in the correct partition.

Given a correctly estimated partition at the receiver, from Fig. 3.8 it is clear that the minimum distance between signal points in a partition is twice that in the original constellation. Therefore, for a same error performance, the

+/− bit	$j=1$	$2 \le j \le \frac{L}{2}-1$	$j=\frac{L}{2}$
+	2	$j+1$	$\frac{L}{2}-1$
−	1	$j-1$	$\frac{L}{2}-1$

Table 3.1: Modification of the index number when the received partitioning bit is in error.

minimum signal distance in the coded system can be taken half of that in the uncoded system. Since the average signal energy is proportional to the square of the minimum distance, FCM saves by a factor 4 in energy (or 6 dB). This 6 dB improvement also holds for a general D-dimensional cube constellation.

3.3.3 BSC of the partitioning bit

In every dimension one partitioning bit is used to differentiate between every other signal coordinate point, i.e. between B_a and $B_b = \mathcal{X}' - B_0$ sets of signal coordinate points. The partitioning bit is encoded using an MRFC scheme for BSCs. The corresponding channel for the partitioning bit is a BSC with crossover probability obtained from the union bound estimate (3.8) as

$$p \approx N_{f,d} Q(\frac{\gamma}{2\sigma}) \tag{3.11}$$

where $N_{f,d} = \frac{2(L-2)+2}{L} = 2\frac{L-1}{L}$. Here p is the probability that a transmitted signal coordinate point from B_a is received in the decision regions of those in B_b or vice versa. The argument of the Q function can be written in terms of the SNR of the received vector. From (3.6) it follows that

$$\frac{\gamma^2}{4} = \frac{3}{L^2-1}E_d = \frac{3}{L^2-1}\frac{E_s}{D}$$

and since $\sigma^2 = \frac{N_o}{2}$, we have

$$\frac{\gamma}{2\sigma} = \sqrt{\frac{3}{L^2-1}\frac{E_d}{\frac{N_o}{2}}} = \sqrt{\frac{3}{L^2-1}SNR_d} = \sqrt{\frac{3}{L^2-1}SNR}.$$

Example 3.5 In the lower part of Fig. 3.10 the average mutual information corresponding to a 64 point 2-dimensional constellation (8×8) is depicted using the procedure of Appendix B. In the upper part of the figure, the BSC capacity of the corresponding partitioning bit (in one dimension) is depicted from $1-h(p)$ where $h(p) = -p \log p - (1-p) \log(1-p)$ and p is given by (3.11). From the union bound estimate (3.8) and (3.9), the signal error probability for the uncoded constellation is 10^{-6} at point A where $\mathrm{SNR}_A = 27.2$ dB. If we want to have an improvement of 6 dB by our coding scheme as mentioned before, then the coded system has to work at point B where $\mathrm{SNR}_B = 27.2-6 = 21.2$ dB. At this SNR, the BSC capacity of a partitioning bit is 0.914 bits. Since every transmitted signal consists of two partitioning bits and four uncoded bits, this signal can transfer fewer than 6 information bits at point B. Note that an uncoded system transfers 6 information bits at point A. Therefore, the normalization of the SNR with respect to the transmission rate is necessary to fairly measure the SNR improvement by the mentioned coding scheme, see also Example 3.6. □

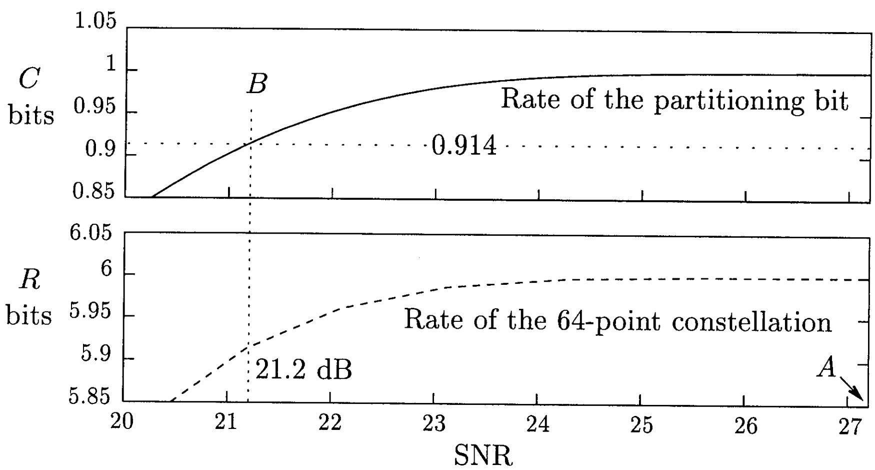


Figure 3.10: Maximum rate of a partitioning bit and that of the corresponding constellation.

3.4 Error performance of FCM

In a FCM system there are three sources of error that affect the performance of the coding process. The corresponding error probabilities are: the intra-partition error probability designated by P_p (the signal error probability within a given partition), the inter-partition error probability designated by P_m (the decoding error probability of the MRFC schemes) and the feedback channel error probability designated by P_{fb}. In what follows, the parameters of the scheme are chosen such that P_p determines P_t, the total error probability of FCM. I.e. $P_t \approx P_p + P_m + P_{fb} \approx P_p$, when $P_p \gg P_m$ and $P_p \gg P_{fb}$.

3.4.1 Intra-partition error

This is the probability that, in a given partition, a transmitted signal point is received in the decision region of another signal point. For example, consider a FCM scheme with 64-point 2-dimensional signal constellation (8 $\times$ 8). After decoding of the MRFC schemes in both dimensions, the transmitted partition is known at the receiver. Given a correctly estimated partition, the intra-partition error probability is the probability that the transmitted signal point $\mathbf{x}_w^D$ is received outside decision region R_w, see Fig. 3.11. Note that the estimating process of the partitioning bits and the index modification procedure mentioned in Table 3.1 automatically determine the decision region of every received signal point within the estimated partition.

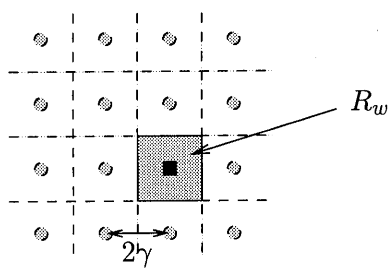


Figure 3.11: The decision regions in a partition of a 64-point 2-dimensional constellation.

The signal error probability in a partition of D-dimensional cubic constellation $\mathcal{X}$ can well be approximated by the union bound estimate (3.8) as

$$P_{s,p} \approx N_{f,p} Q(\frac{2\gamma}{2\sigma}), \tag{3.12}$$

where $N_{f,p} = D\frac{2(\frac{L}{2}-2)+2}{\frac{L}{2}} = 2D\frac{L-2}{L}$ is the average number of nearest neighbours in each partition of $\mathcal{X}$. This bound is tight in our range of interest. It is useful to mention that the signal error probability here is the same as that in the parallel branches of the trellis in Fig 3.6.

As mentioned before, we are going to make the error performance of FCM be dominated by P_p, i.e. by $P_{s,p}$. Therefore, we have to compare $P_{s,p}$ with $P_{s,uc}$, the signal error probability of the uncoded system. For the original (uncoded) constellation one obtains

$$P_{s,uc} \approx N_{f,uc} Q(\frac{\gamma}{2\sigma}), \tag{3.13}$$

where $N_{f,uc} = D^{\frac{2(L-2)+2}{L}} = 2D^{\frac{L-1}{L}}$ (also mentioned in Example 3.2).

Example 3.6 In the constellation of Example 3.5, each partition contains 16 signal points, thus, $N_{f,p} = 3$ while $N_{f,uc} = 3.5$. The uncoded and coded (FCM) signal error probabilities of the constellation are depicted in Fig. 3.12 from (3.13) and (3.12), respectively. In order to compare the results fairly, the SNR's are normalized according to criterion (3.10). For the uncoded scheme $R = 2R_d = 6$ bits per transmission and for the coded scheme $R = 2R_d = 4+2x$ bits per transmission, where $x = R_0^k(1 - kp)$ and p is given by (3.11). The signal error probability of FCM is normalized with respect to the transmission rate for $k = 4, 5, 6$. The curves of Fig. 3.12 bend because of the SNR normalization. With decreasing SNR, the rate of the MRFC scheme falls faster and the bending effect becomes more evident as k increases. □

3.4.2 Inter-partition error

We can speak about the intra-partition signal error probability in the previous subsection as long as the partition of the transmitted signal is correctly estimated at the receiver. In other words, the MRFC schemes of all coordinates must also be decoded correctly in order to deliver the information associated with both partitioning bits and uncoded bits.

The inter-partition error occurs if the partition of a transmitted signal is erroneously estimated, i.e. at least one of the partitioning bits is estimated erroneously in a block or sequence of transmissions (depending on whether block or recursive MRFC is used in FCM, respectively). In this subsection, the effect of the partition estimation error is taken into account for evaluating the performance of FCM with block MRFC and with recursive MRFC.

Performance of FCM with block MRFC

Let N be the block length of MRFC schemes. In a dimension, designate the block decoding error probability of the MRFC scheme by P_b^N. The block error probability of MRFC schemes in D dimensions is upper bounded by

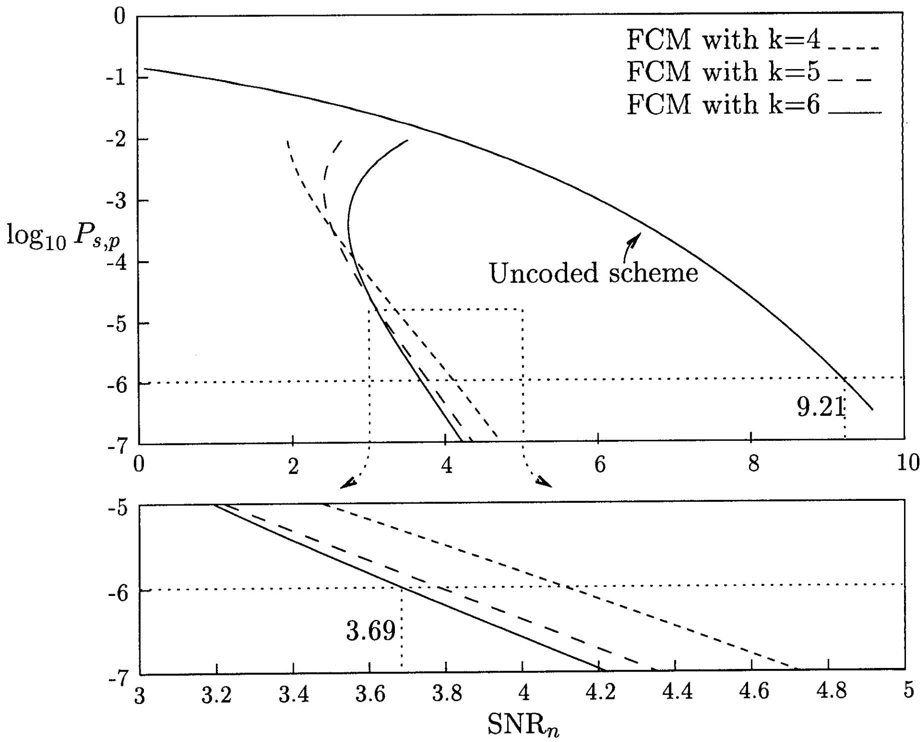


Figure 3.12: The symbol error probability for FCM.

$$Pr\{\text{Block inter-partition error}\} = P_m^N \leq D \cdot P_b^N.$$

The total block error probability of FCM can be written as

$$P_t^N \approx P_m^N + P_p^N + P_{fb}^N \overset{(\alpha)}{\approx} P_m^N + P_p^N, \tag{3.14}$$

where in (α) the error probability of the noisy feedback channel is ignored conform to the argument of Subsection 3.4.3. In (3.14) P_p^N is the block error probability caused by the intra-partition signal error (in a correctly estimated partition). It is

$$P_p^N = 1 - (1 - P_{s,p})^N,$$

where $P_{s,p}$ is given by (3.12). To make $P_{s,p}$ (i.e. P_p^N) dominant in (3.14), let $P_p^N \gg P_m^N$. Therefore, (3.14) can be written as

$$P_t^N \approx P_p^N,$$

which means that the normalized error probability of FCM per transmitted signal is determined by $P_{s,p}$. Normally, we are interested in points where $P_{s,p} = 10^{-6}$ or $P_{s,uc} = 10^{-6}$, which explains the graphs of Fig. 3.12.

In order to decrease the block error probability of MRFC schemes, i.e. to make $P_p^N \gg P_m^N$, one solution is to increase the tail length of the corresponding blocks from $kpN+1$ to $kfN+1$. In this way, each block can certainly correct up to fN errors, $f > p$. For example, assume that

$$P_p^N \geq 10 \times D \times 2^{-ND(f||p)} \overset{(\beta)}{\geq} 10 \times D \times P_b^N \geq 10P_m^N. \tag{3.15}$$

In (β) the upper-bound of (2.9) is used for P_b^N and p is obtained from (3.11). Having condition (3.15) causes a drop in the transmission rate (due to increasing the tail lengths), which must be taken into account in the SNR normalization, see (3.16) in the following example.

Example 3.7 For the 64-point 2-dimensional constellation mentioned in Example 3.5, if $P_{s,p} = 10^{-6}$, then $p = 0.011$. The SNR$_n$ loss due to increasing the tail of both MRFC schemes is plotted in Fig. 3.13 according to the following procedure. First, for each block length N, the smallest error correction fraction f that satisfies the left inequality of (3.15), i.e

$$1 - (1 - 10^{-6})^N \geq 10 \times 2 \times 2^{-ND(f||0.011)},$$

is computed. Then, the total SNR$_n$ loss due to increasing the tail lengths of the two MRFC schemes is calculated from (3.10) as

$$10 \log_{10}\left(\frac{2^{4+2R_0^k(1-kf)} - 1}{2^{4+2R_0^k(1-kp)} - 1}\right). \tag{3.16}$$

This SNR$_n$ loss is plotted in Fig. 3.13 for $k = 6$ in terms of N. Note that $k = 6$ is the optimum repetition value at $P_{s,c} = 10^{-6}$ where $p = 0.011$, see also Subsection 3.6.2. With increasing block length, the SNR loss is decreasing. Note that increasing N does not affect the computational complexity of the decoding, and only the length of the buffers increases. As indicated in Fig.

3.13 for 0.2 dB loss, the length of the buffers should be 6650 according to (3.15).

One may argue that (3.15) is not a good measure when $N = 6650$ because then $P_p^N \approx 10^{-2}$. Let $P_p^N = 10^{-5}$, for example by using a high-rate error-correcting code for the uncoded bits in Fig. 3.7. Using the same computation shows that $N = 11,000$ is enough for 0.2 dB loss. □

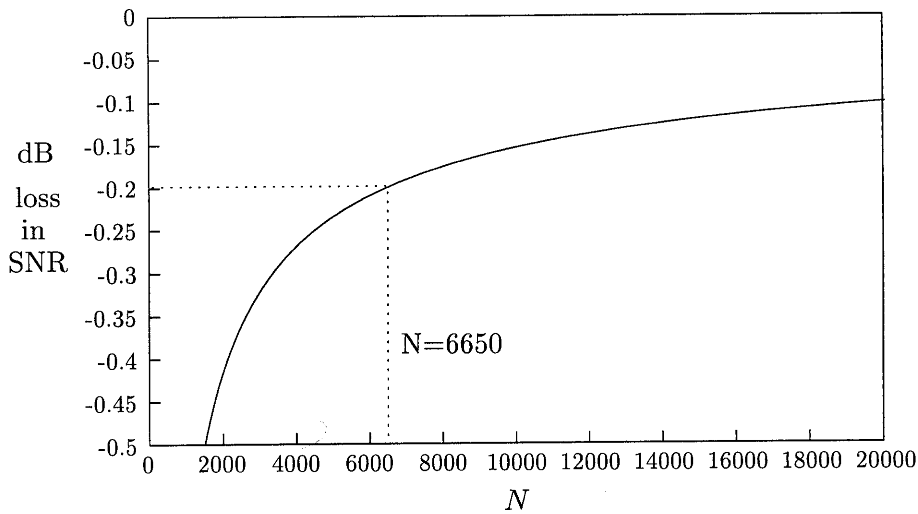


Figure 3.13: The SNR loss due to increasing the tail length in FCM with block MRFC.

Remark 3.2 The effective coding gain of the FCM with block MRFC for the 2-dimensional 64-point constellation is 5.32 dB, which can be computed from Fig. 3.12 as 9.21 - 3.69 = 5.52 dB minus 0.2 dB tail increasing loss when $N = 6650$. As seen from Fig. 3.13, the SNR_n loss due to increasing the tail length could be significant for short block lengths. In Chapter 4 two block retransmission strategy are presented for MRFC schemes that improve the error performance of the block MRFC. In this way, it is possible to shorten the gap between f and p, and reduce the SNR_n loss in (3.16). □

Performance of FCM with recursive MRFC

The recursive MRFC is described in Section 2.2.3. In the recursive decoding, each transmitted partitioning bit is estimated based on only the D_r following received partitioning bits (from the same coordinate). The length of the buffers in this method, i.e. (D_r+1), can be significantly smaller than that in the block MRFC. However, a random walk of D_r+1 steps must be performed for **every** partitioning bit using the state diagram of Fig. 2.11.

The total error probability (per transmitted symbol) of FCM with recursive MRFC can be written as

$$P_t^r \approx P_{s,p} + P_m^r + P_{s,fb} \overset{(\alpha)}{\approx} P_{s,p} + P_m^r \overset{(\beta)}{\approx} P_{s,p} \tag{3.17}$$

where in (α) $P_{s,fb}$, the error probability of the noisy feedback channel (per signal), is ignored conform to the argument of Subsection 3.4.3. The partition estimation error probability, i.e. the inter-partition error probability of the recursive method, is $P_m^r \leq DP_r(i)$, where $P_r(i)$ is the estimation error probability of a partitioning bit at the i^{th} transmission. An upper bound to the probability of this event is given by (2.13), which can be rewritten as

$$P_r(i) = Pr\{\hat{T}_{id} \neq t_{id}\} \leq \frac{2}{k-2} \cdot 2^{-D_r \cdot D(\frac{1}{k}\|p)} \tag{3.18}$$

when $p = p_6 = 0.017$ (p_6 is the solution of (2.1) when $k = 6$). According to the argument of Example 2.5, $D(\frac{1}{k}\|p)$ is the asymptotic exponent of $P_r(i)$ even when $p = 0.011$. Therefore, in the following example we use the upper bound of (3.18) to give a measure on the value of D_r. For a rigorous statement, one can consider the performance of the system at $p = p_6 = 0.017$ which is about 0.6 dB lower than the point at which $P_{s,p} = 10^{-6}$ in Fig. 3.12 (or use $k = 7$ and consider the performance of the system at $p = p_7 = 0.0082$ which is about 0.4 dB higher than the point at which $P_{s,p} = 10^{-6}$ in Fig. 3.12).

Example 3.8 In Fig. 3.14, the upper bound of $\log_{10} P_r(i)$ is plotted in terms of buffer length D_r from (3.18) for optimum repetition parameter $k = 6$ at $p = 0.011$. If we apply the rule of thumb $P_r(i) \leq 10^{-14}$, then the MRFC schemes are practically error free and one can make the approximation of (β) in (3.17). The required buffer length for $k = 6$ at $p = 0.011$ when $P_r(i) \leq 10^{-14}$ is 101. See also the simulation results at the end of Subsection 3.5.3. □

Remark 3.3 The effective coding gain of the FCM with recursive MRFC for the 2-dimensional 64-point constellation is 5.52 dB, which can be computed from Fig. 3.12 as 9.21 - 3.69 = 5.52 dB when $D_r = 101$. The right-to-left or left-to-right bit estimating methods can also be used in the block coding of

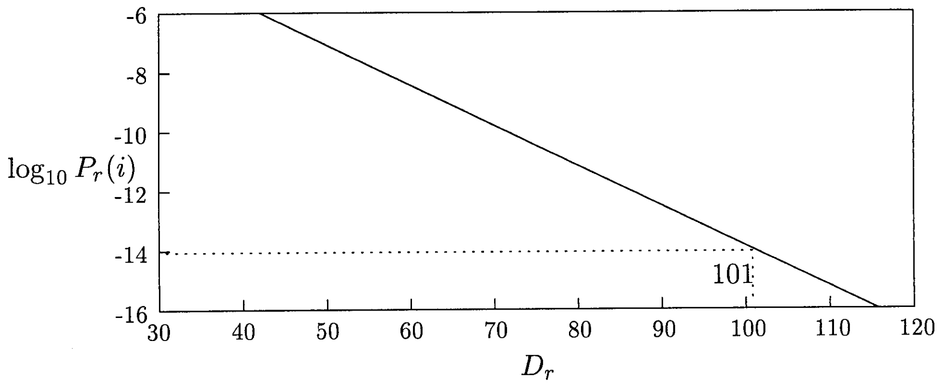


Figure 3.14: The exponent of the (upper bound of the) estimation error probability in terms of delay length.

FCM to decrease the error probability of the estimated partitioning bits. In this way, using large block lengths is more feasible where uncoded data can only be decoded after a short delay. Nevertheless, the information associated with the partitioning bits must be decoded when the whole block is received. □

3.4.3 Feedback channel error

So far we have practically considered the feedback channel to be noiseless, i.e P_{fb}^N and $P_{s,fb}$ are ignored in (3.14) and (3.17), respectively. On the other hand, the assumption is that the feedback channel is noisy, i.e. the SNR's in the forward channel and in the feedback channel are equal or

$$\text{SNR}_{fb} = \text{SNR}. \qquad (3.19)$$

Our approach is to make this noisy feedback channel perform like a noiseless channel. This is done, perhaps being clear to the reader by now, by reducing the amount of information in the backward direction relative to that of the forward direction and spending the same signal energy of the forward direction on this, the so-called, **low rate** feedback information.

For every forward transmission in FCM scheme, we need to feed back D bits (one bit per dimension) over the noisy feedback channel. Therefore, in the signal constellation of the feedback channel there are two signal points per dimension or 2^D signal points in D dimensions. If γ_{fb} is the distance between two signal coordinate points in the constellation of the feedback channel, then condition (3.19) means that

$$\mathrm{SNR} = \mathrm{SNR}_d = \frac{\gamma^2 \frac{L^2-1}{12}}{\sigma^2} = \frac{\gamma_{fb}^2 \frac{2^2-1}{12}}{\sigma^2} = \mathrm{SNR}_{fb,d} = \mathrm{SNR}_{fb}$$

or

$$\gamma_{fb} = \gamma\sqrt{\frac{L^2-1}{3}}.$$

If the 6 dB improved coded system works at signal error probability 10^{-6}, then for such minimum distance γ the signal error probability of the uncoded constellation is about 10^{-2}. By choosing $\gamma_{fb} \approx 3\gamma$, the symbol error probability in the feedback channel will be in the order of 10^{-14}. This error probability in the feedback link is so small that it practically enables us to consider the feedback channel noiseless. Thus, the criterion for the assumption that the feedback error probability per dimension $P_{s,fb} < 10^{-14}$ becomes

$$\gamma_{fb} \geq 3\gamma \quad \overset{(3.19)}{\Longrightarrow} \quad L^2 \geq 28 \quad \text{or} \quad L \geq 6. \tag{3.20}$$

In Fig. 3.15 the signal constellation of the feedback channel, the filled points, is depicted relative to the signal constellation of the forward channel in the case where criterion (3.20) holds with equality. Here, if $P_{s,uc}$ is about 10^{-2} then $P_{s,fb}$ is about 10^{-14}.

Remark 3.4 In order to make the feedback channel virtually noiseless, a SNR_{fb} that yields equality on the left side of (3.20) suffices. If the number of signal points per dimension of the signal constellation of the forward channel is larger than 6, then SNR_{fb} can be smaller than the SNR of the forward channel. Note that criterion (3.20) is a rule of thumb and can be relaxed. For example, in FCM with block MRFC, we can make the block error probability due to the feedback channel error 10 times smaller than that due to the intra-partition signal error. □

3.5 Decoding complexity, a comparison

Here the important justification for using the feedback information in the coding process is to reduce the coding complexity, in general, and to reduce the complexity of the decoder, in particular. Therefore, the decoding complexities of TCM and FCM are roughly evaluated and compared in this section in terms of the types of computations and the amount of memory needed in the decoder of each method.

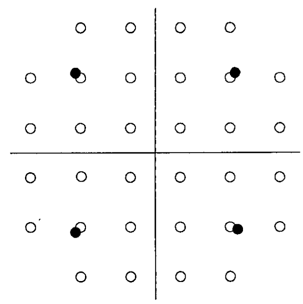


Figure 3.15: Constellation of the feedback channel, with filled points, relative to that of the forward channel (an illustration).

3.5.1 Decoding complexity of TCM

The decoding complexity of TCM can be considered for the general scheme of Fig. 3.5. Here we do not insist on evaluating the complexity of the most efficient decoding techniques of TCM. Our intention is to give an indication of the types and the number of major processes and facilities needed in a general decoder of TCM. For every received symbol, the following operations are carried out in the decoder of a TCM system.

1. The most likely signal point in each partition is found in the subset decoding. This is accomplished by finding the signal point in each partition that is closest in Euclidean distance to the received point. For each partition, the Euclidean distances of the received point to all 2^{k_2} signal points of the partition have to be calculated and the smallest distance and the corresponding signal point have to be found.

2. After performing the first step, corresponding to each set of parallel transitions in the trellis there exists only one point from the partition (the most likely one) and one corresponding Euclidean distance. The next decoding step is to use these Euclidean distances to find the path on the trellis whose total Euclidean distance from the received sequence is minimum. This step is done by applying the Viterbi algorithm. In a 2^{ν}-state Viterbi decoder, the following operations and facilities are needed. Note that in the trellis diagram 2^{k_1} branches are entering each state and the same number of branches are emerging from it.

   i. The add-compare-select part finds the survival branch arriving at each state and it also updates the metric of the corresponding state.

In each state, we have to choose the incoming survivor branch, therefore, 2^{k_1} additions and $2^{k_1} - 1$ comparisons of real valued numbers are necessary. The survivor metrics are stored for the next signal.

ii. In the path-storage unit, the survivor branch of the previous step is used to update the survivor path of each state. For each state, the survivor path leading to that state is stored up to a certain depth. Normally, in the Viterbi decoder the path records of all states have a depth of $5 \times (\nu + 1)$ branches. For each state we have to store k_1 bits as the index of the survivor branch and k_2 bits as the index of the closest signal point in the corresponding partition.

iii. The survivor path selection is performed by comparing the metrics of the states and choosing the smallest one. The survivor path of the chosen state is then traced back for a depth of $5 \times (\nu + 1)$ branches. In this way, the most likely symbol at $5 \times (\nu + 1)$ signal periods in the past is obtained.

In TCM, the total amount of memory needed to implement the decoder and the type and the number of operations that have to be carried out for **every** received signal at the decoder are summarized in Table 3.2. The real valued numbers can be processed by quantizing the corresponding real values. The precision of the quantization determines the complexity of the operations and the required memory size. In order not to go in too much detail, we refer to them as real values in Table 3.2.

3.5.2 Decoding complexity of FCM

Let us investigate the same requirements for the decoder of FCM. In the following, subscripts b and r denote that the corresponding requirement applies to FCM with block and recursive MRFC, respectively. The FCM with block MRFC needs the following operations and facilities in the decoder.

1_b. D buffers of length N are needed to store the received signal points. Since $\log L + 1$ bits per dimension per signal are enough to store a received coordinate point, the total required memory is $DN(\log L + 1)$ bits.

2_b. For every received partitioning bit, a pattern search is needed to look for the forbidden subsequences 10^k and 01^k. On average, there occur DNp errors for partitioning bits in a block of N transmissions. For every error, one forbidden subsequence replacement and one index modification are needed (p is given by (3.11)). In total DN pattern searches and on average DNp modifications must be performed in a block.

No.	The type and number of operations	The amount of needed memory
1	$2^{n_1} \cdot 2^{k_2}$ distance calculations (real values)	2^{n_1} distances (real values)
	$2^{n_1} \cdot (2^{k_2} - 1)$ comparisons (real values)	$2^{n_1} \cdot k_2$ bits signal point indexes
2.i	$2^{\nu} \cdot 2^{k_1}$ additions (real values)	2^{ν} metrics of states (real values)
	$2^{\nu} \cdot (2^{k_1} - 1)$ comparisons (real values)	
2.ii	2^{ν} surviving path updating (real values)	$5(\nu + 1) \cdot 2^{\nu}(k_1 + k_2)$ bits records of surviving paths
2.iii	$2^{\nu} - 1$ comparisons (real values)	
	$5(\nu + 1)$ back-tracings	

Table 3.2: The requirements needed in a typical decoder of the TCM.

Similarly, the following requirements are needed in the decoder of the FCM system with recursive MRFC.

1_r. As D buffers of length $(D_r + 1)$ are used, a memory of size $D(D_r + 1)(\log L + 1)$ bits is needed.

2_r. A random walk of $(D_r + 1)$ steps is performed for every received partitioning bit in the estimator. One forbidden subsequence replacement and one index modification are needed on average per $\frac{1}{p}$ partitioning bits.

Precoding and inverse precoding are needed in both block and recursive methods. In [91, Section 7.2] an overview of precoding methods for MRFC schemes are given. It appears that for the symmetric MRFC scheme that is used in the FCM method, there are efficient methods that only require a few operations per precoded bit. In the following, we mention only two methods and the complexities of the corresponding inverse precoders without going into implementation detail.

3_b. For block MRFC with short block lengths, one may use an enumerative scheme for precoding. The enumerative coding was introduced by

Cover in [15]. However, Becker [15] independently developed a similar method for the precoding of MRFC from Schalkwijk's source coding algorithm [59]. As mentioned in [91, pp. 101], an inverse precoder based on the enumerative scheme has to store k L-bits numbers. Moreover, one comparison and at most k L-bit additions are needed per precoded bit. Therefore, this method turns to be rather complex when L is large.

$3_{b,r}$. For block MRFC with large block lengths or recursive MRFC, an arithmetic decoder and an arithmetic encoder can be used as the precoder and inverse precoder, respectively. An arithmetic coding method with a fixed rate for mapping messages into constrained sequences is mentioned in [38, 83]. Here we consider the efficient and fast arithmetic coding algorithm presented in [33], which also employs the FSM of the precoder (e.g. see Fig. 2.4) and the corresponding state transition probabilities given in Appendix A (see Chapter 6 for more detail). The resulting inverse precoder needs $(k-1)$ registers to store the state transition probabilities of the FSM, two registers per dimension to store the beginning point and the length of the corresponding code interval and one register per dimension to store the number of runs, see [33]. For every precoded bit, the arithmetic encoder requires: one multiplication, at most two additions and (for our application) at most two scalings, i.e. shifting the contents of two registers.

Note that the given requirements per precoded bit is an overestimation, because the number of precoded bits is about 10 percents smaller than the number of transmitted partitioning bits.

3.5.3 Comparison of decoding complexities

For a coding scheme the *asymptotic* coding gain can be determined from the union bound. The amount of improvement in the minimum signal distance γ_{fed} as defined in Subsection 3.2.3 determines the asymptotic coding gain. This gain affects the argument of the Q function in (3.8). In Table 3.3, γ_{fed} and the asymptotic coding gain of different codes for TCM are presented from [85]. The first column gives the number of states in the trellis code. The N_f column indicates the (average) number of nearest neighbour signal sequences with distance γ_{fed} that diverge at any state from a transmitted signal sequence and merge with it after one or more transmissions.

Due to term N_f in (3.8), however, the coding gains of the codes mentioned in Table 3.3 are not feasible. The *effective* coding gain of a modulation scheme is measured by the reduction in required SNR_n to achieve a certain target

No. of states	Asympt. $\frac{\gamma_{fed}^2}{\gamma_0^2}$	Asympt. $\frac{\gamma_{fed}^2}{\gamma^2}$	G_c in dB	N_f
4	4*	2*	3.01	4
8	5	2.5	3.89	16
16	6	3	4.77	56
32	6	3	4.77	16
64	7	3.5	5.44	56
128	8	4*	6.02	344
256	8	4*	6.02	44

Table 3.3: The asymptotic coding gains of different codes for three level two-dimensional TCM. The $*$ indicates that γ_{fed} is determined by γ_p.

signal error probability (e.g. 10^{-6}) relative to a baseline uncoded scheme [26], see also Fig. 3.12 for an illustration. Fig. 6 from [26] gives the effective coding gains of different codes designed for TCM, where the effect of nearest neighbour signals are also taken into account. In [26], Forney and Ungerboeck say:

> "It is noteworthy that no one has improved on the performance complexity tradeoff of the original 1D and 2D trellis codes of Ungerboeck or the subsequent multidimensional codes of Wei. By this time it seems safe to predict that no one will ever do so. There have, however, been new trellis codes that feature other properties and have about the same performance and complexity, ..., and there may still be room for further improvements of this kind."

Therefore, according to [26, Fig. 6] a 256-state or a 512-state trellis code of Ungerboeck can be chosen as a comparison benchmark in the range of 5.3-5.5 dB (effective) coding gain. Also FCM yields a coding gain in this range depending on whether a block or a recursive MRFC scheme is used, see Rems. 3.2 and 3.3. In the following, we summarize the comparison results between the decoding complexity of the TCM (with 256-state trellis code) and that of FCM.

Comparison results

Assume a 2-dimensional signalling scheme for which the original constellation contains 64 signal points. If we apply a three level TCM with a 256 state trellis code, then the parameters of Fig. 3.5 are $k_1 = 2$, $n_1 = 3$ and $k_2 = 4$. From Table 3.2, where $\nu = 8$, the decoder needs

– a 8644 byte memory and a memory for 264 real values;

– for every signal: 128 Euclidean distance calculations, 1024 additions and 1143 comparisons of real values, 256 metric updating and a back tracing of 45 branches.

Again we emphasize that these figures are not the requirements of an optimum TCM decoder, however, we can say that they present a good estimation of those major requirements. On the other hand, if the FCM scheme is applied to the same constellation, for $k = 6$, $N = 6650$ and $(D_r + 1) = 101$ we need

+ a 6650 byte memory in the block decoding or a 101 byte memory in the recursive decoding;

+ for every signal: 2 pattern searches in the block decoding or 2 random walks in the recursive decoding as well as 2 forbidden subsequence replacement and 2 index modifications per 90 signals (same in both methods);

+ in the inverse precoder of both methods: a 22 byte memory, and for every received signal 2 multiplications, at most 4 additions and 4 scalings.

From the above figures, the decoding simplicity of the FCM scheme (or the decoding complexity of the TCM) is evident. Note that we could have used a 512-state trellis code as the benchmark in the coding gain range of the recursive FCM. For a 512-state trellis code, almost a factor of two appears in the requirements of TCM's decoder, mentioned above.

Simulation results

The FCM with recursive MRFC is simulated for two cases of PAM and QAM signalling, where an 8-point and a 64-point constellations are considered, respectively. The length of the buffers is chosen as $D_r = 75$. During 10^9 transmissions of random data, the estimators of the partitioning bits did not yield any error at the SNR where the signal error probability within partitions became $P_{s,p} = 10^{-6}$. Compared to uncoded schemes, coding gains of 5.41 dB and 5.49 dB were observed in the PAM and QAM cases, respectively.

D'	2	4	8	12	16	24	32	64	...	∞
dB	0.20	0.46	0.73	0.88	0.98	1.10	1.17	1.31	...	1.53

Table 3.4: The maximum shaping gain for spherical constellation over cubic constellation.

3.6 Some peripheral aspects of FCM

3.6.1 Signal shaping

The signal shaping problem is that of choosing the region $\mathcal{R}_C$ that bounds the D-dimensional signal constellation. Until now we have chosen $\mathcal{R}_C$ to be a D-dimensional cubic. In such a constellation the average signal energy is not optimum. For example, consider the 2-dimensional constellation of Fig. 3.16. If instead of signal point A, we choose signal point B from the outside of the square, then the average signal energy will be reduced. It is easy to see that the optimum $\mathcal{R}_C$ is a circle. Similarly in D dimensions, the optimum $\mathcal{R}_C$ will be a D-dimensional sphere. The shape of the enclosing region of the constellation does not change the minimum distance of signal points and, consequently, the error performance of the constellation remains practically unchanged. It is important to be emphasized that if we use the signal shaping techniques in FCM, the resulting improvement in SNR does not affect the minimum distance between the signal points and, therefore, it does not affect the transmission of the partitioning bits.

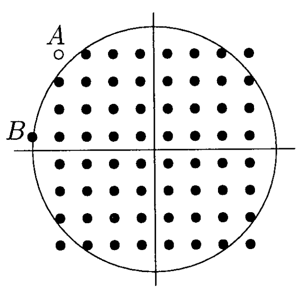


Figure 3.16: An illustration for signal shaping in a 2-dimensional constellation.

In D' dimensions, the amount of maximum gain that can be obtained by choosing a sphere constellation over a cubic constellation, i.e. by the so-called *signal shaping*, is presented in Table 3.4 from [13]. Note that the number of dimensions in the signal shaping, denoted by D' here, is not necessarily the same as D, the number of dimensions in the coded modulation. Signal shaping codes or techniques are investigated in many papers, see [18, pp. 10] for an overview. In this section we briefly mention a simple signal shaping method which was first proposed in [97, 94] and later was studied in [18]. We illustrate that this method can be adapted for using in FCM. Based on enumerative source coding [59], this method keeps the total energy of D' dimensions below a certain limit, i.e.

$$\sum_{d'=1}^{D'} |x_{d'}|^2 \leq E_{max},$$

where $x_{d'} \in \mathcal{X}'$ is the $(d')^{\text{th}}$ coordinate of the shaped signal. In this way, only the signal points which are at the inside of the D'-dimensional sphere of radius $\sqrt{E_{max}}$ are transmitted.

The index assignment of Fig. 3.9 is chosen for signal shaping purposes, where the energy of the transmitted signal points depends on indexes and is independent of the partitioning bits. Therefore, a signal shaping code which operates on the indexes can be independent of the MRFC schemes and the on-fly repetitions of partitioning bits, see Fig. 3.17. Finally, we present in Table 3.5 one of the results of [18] to indicate how much SNR improvement can be gained with this simple method.

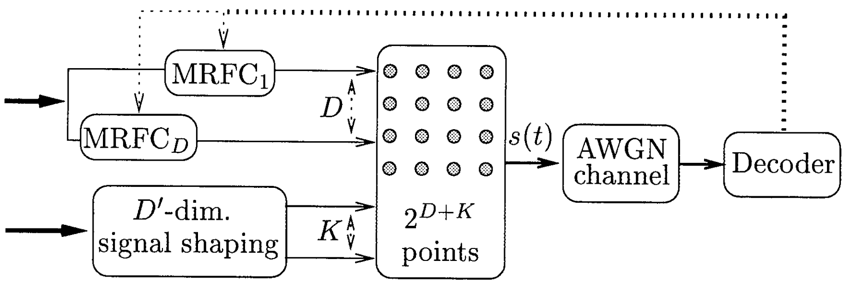


Figure 3.17: A block diagram of the FCM with signal shaping in the encoder.

D'	4	8	16	32	64
dB	0.218	0.465	0.709	0.901	1.039

Table 3.5: The shaping gains for the method reported in [18].

3.6.2 Choice of the repetition parameter

In [91, Chapter 5] Veugen presents a rule to choose the repetition parameters in MRFC for a given DMC so that the the maximum transmission rate is obtained. For a BSC with crossover probability p, the optimum repetition parameter is $k = - < \log p >$ where the $< z >$ operation yields the closest integer to real number z. This seems intuitively reasonable because a transmission error, which occurs with probability p, needs $\log \frac{1}{p}$ bits (or transmissions in the case of a BSC) to be clarified for the decoder according to Def. 1.6, see also [91]. One must notice that $\log \frac{1}{p}$ bits are not enough to correct an error, a precoding mechanism is further needed to distinguish between correct receptions and erroneous ones, i.e. to produce an acknowledgement mechanism as mentioned in the first chapter. Thus, a combination of precoding and repetitions is the cost paid to correct any channel error. As $p \to 0$, the optimum $k \to \infty$ and the precoding rate $R_0^k \to 1$, therefore, the precoding redundancy goes to zero and only the repetition cost is paid for correcting an error. This explanation is in accordance with [91, Theorem 5.1].

For recursive MRFC, one must choose a scheme which gives maximum transmission rate at p. A good approximation is the mentioned rule $k = - < \log p >$. For block MRFC, we saw that the correctable error fraction must be increased from p to $f = \frac{N-L}{kN}$ in order to have the desired error exponent $D(f\|p)$. Here the scheme which maximizes the transmission rate at f must be chosen, i.e. $k = - < \log f >$, according to the mentioned rule.

3.7 Conclusions

In this chapter FCM is introduced and analyzed. It is shown that in this way we can gain about 5.5 dB improvement in the SNR. This effective coding gain is in the range of the effective coding gain of the most complex TCM with a 256 or 512 state trellis code. In FCM the decoding process is considerably simplified compared to that of the TCM. The decoder of FCM mainly needs D buffers of a limited length. In the recursive FCM, where only a few recently received points are needed, the length of the buffers can even be shortened considerably compared to the buffer length of block FCM. The amount of processing per signal in the decoder of FCM is extremely smaller with respect to that of TCM. More importantly, the feedback channel considered in FCM is a noisy channel, which is virtually made errorless by the scheme. It is also shown how to use signal shaping techniques in the FCM case. With a very simple signal shaping code, an extra shaping gain of 1 dB can be obtained on top of the 5.5 dB gain. See also Chapter 7 for some applications and practical aspects of FCM.

Chapter 4

Retransmission strategies with MRFC schemes

The transmitter in information feedback schemes can easily discover if a received block is decoded correctly. In this chapter we show how the transmitter of a MRFC system can use the resulting knowledge and apply retransmission strategies. In Chapter 3, the error performance of FCM with block MRFC is improved by increasing the tail length of the corresponding MRFC schemes. However, increasing the tail length also decreases the transmission rate for small block lengths, see Fig. 3.13. The retransmission strategies presented here are another solution for improving the error performance of the block MRFC schemes.

4.1 The basic idea

In a block retransmission strategy, the transmission of every block has two modes: the message mode and the control mode. In the message mode, the code word of a message is sent over the channel. Then the receiver sends back the decoded message or the transmitter deduces it by observing the received signal. In the control mode, the transmitter sends the acknowledgement or the rejection signal depending on whether the decoder is correct or not, respectively. The receiver decodes the control part and if the decoded result is an acknowledgement, the receiver accepts the decoded message as correct. If the control part is decoded as a rejection, the receiver discards the decoded message and waits for a retransmission. In conclusion, the transmitter of a block retransmission strategy must fulfil the following functions.

- Send (the code word of) a message in the message mode.

- Detect whether or not the decoded message is the same as the transmitted one.
- Send the acknowledgement or the rejection signals in the control mode, accordingly.

A code word in this strategy consists of the message signal and the corresponding control signal, see Fig. 4.1 for an illustration. Some schemes that use this principle, e.g. [64, 99], are explained in more detail in Subsection 1.6.2. Two factors stimulate us to use retransmission strategies and MRFC schemes together, namely: the existence of block and recursive MRFC methods with two different error performances at high transmission rates and the fact that it is easy for the transmitter in MRFC systems to detect a correct block decoding. In the following sections, two retransmission strategies with MRFC are devised and evaluated, see also [79].

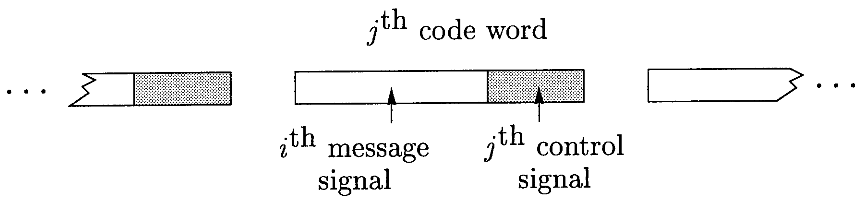


Figure 4.1: Transmitted code words in a block retransmission strategy.

4.2 A two-block delay strategy for BSC's

To begin with, let us consider the three requirements mentioned in the previous section for the case of a retransmission strategy with MRFC for BSC's. Fig. 4.2 shows the block diagram of the system.

4.2.1 Detection of block decoding error

To know whether a block is decoded correctly or not, the transmitter of a MRFC system itself can decode the received block, which is known to the transmitter using the feedback channel. However, the transmitter can also use a very simple method to detect a correct decodable block. First we define the following.

Definition 4.1 (Block overflow) *In block MRFC, denote the total number of $x \to y$ errors in a block of N transmissions by e_{xy}, $(x, y) \in (\mathcal{X}, \mathcal{Y})$. Let k_{xy}*

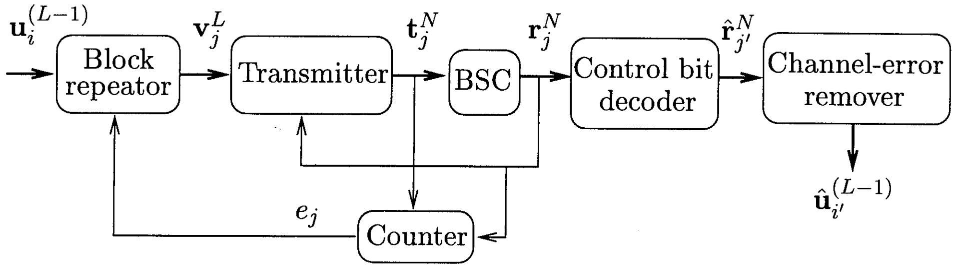


Figure 4.2: The block diagram of the retransmission strategy for BSC's.

denote the repetition parameter of the $x \to y$ error and L be the length of pre-coded sequences. A block overflow is defined as the event $\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} k_{xy}e_{xy} \geq N - L$.

In Fig. 4.2, $\mathbf{t}_j^N = (t_{j1}, \dots, t_{jN})$ and $\mathbf{r}_j^N = (r_{j1}, \dots, r_{jN})$ denote the j^{th} transmitted block and the corresponding received block, respectively, $j = 1, 2, \dots$. Let e_j denote the number of channel errors during the transmission of the j^{th} block, i.e.

$$e_j = \sum_{n=1}^{N}(1 - \delta(r_{jn}, t_{jn})),$$

where the Kronecker δ function is defined as

$$\delta(a,b) = \begin{cases} 0 & \text{if} \quad a \neq b \\ 1 & \text{if} \quad a = b. \end{cases}$$

In block MRFC for BSC's, obviously, a block overflow occurs only if $e_j > fN$, where f is the correctable error fraction of the block such that $N - L = k\ fN + 1$. We say that the j^{th} block is **successfully** transmitted if the block does not overflow[1]. Note that a non-overflowing block is always decoded correctly according to Theorem 2.1, however, an overflowing block is not necessarily decoded erroneously. Nevertheless, considering the number of errors in a block is a very good and simple measure to indicate a correct block decoding (specially for large block lengths).

[1]This agrees with the definition of a rate being achievable in the Veugen sense, Def. 2.3.

4.2.2 Message sequence and control signals

In MRFC scheme for BSC's with repetition parameter k, precoded sequence $\mathbf{v}^L$ (see Fig. 2.5) does not contain any 01^k and 10^k subsequences. Sequence $\mathbf{v}^L = v_1 \dots v_L$ can be seen as a combination of **message sequence** $\mathbf{u}^{(L-1)} = u_1 \dots u_{L-1}$ and **control signal (bit)** v_1, where

$$u_n = v_n \oplus v_{n+1} \qquad \text{for } n = 1, \dots, (L-1).$$

The $\oplus$ operator denotes modulo 2 addition, i.e. the eXclusive-OR operation. The 1 or 0 symbols in $\mathbf{u}^{(L-1)}$ correspond to a "change" or a "no-change" in sequence $\mathbf{v}^L$. If 01^k and 10^k are forbidden subsequences of $\mathbf{v}^L$, then sequence $\mathbf{u}^{(L-1)}$ does not contain subsequence 0^{k-1}. In other words, sequences $\mathbf{u}$ and $\mathbf{v}$ correspond to the FSM's on the right and left side of Fig. 2.4, respectively. Both constrained channels have the same capacity or precoding rate because they only differ in one bit, which is asymptotically negligible. For convenience let us define the following.

Definition 4.2 *Consider the binary sequence* $\mathbf{b}^{(N-1)}$ *of length* $(N-1)$ *and bit* c. *Define operation* $\Psi(\mathbf{b}^{(N-1)}, c) = \mathbf{a}^N$ *of length* N, *where*

$$\begin{cases} a_1 = c \\ a_n = a_{n-1} \oplus b_{n-1} \quad \textit{for } n = 2, \dots, N. \end{cases} \tag{4.1}$$

The precoded sequence of the j^{th} transmitted block, i.e. $\mathbf{v}_j^L$ as shown in Fig. 4.2, can be written as

$$\mathbf{v}_j^L = \Psi(\mathbf{u}_i^{(L-1)}, cs_j),$$

where $\{\mathbf{u}_i^{(L-1)}\}$, $i = 1, 2, \dots$, is the sequence of messages to be transmitted over the channel, see also Fig. 4.2, and cs_j is the acknowledgement/rejection control bit of the $(j-1)^{\text{st}}$ block.

The transmitter in Fig. 4.2 appends to $\mathbf{v}_{j-1}^L$ the tail sequence $z_{L+1} \dots z_N$ of length $N - L = k\, fN + 1$ bits so that up to fN channel errors can be corrected. Remember that the transmitter shortens the tail from the end by k bits and carries out k (bit) repetitions whenever a transmission error occurs. If the $(j-1)^{\text{st}}$ block, i.e. t_{j-1}^N in Fig. 4.2, is transmitted successfully, i.e. $e_{j-1} \le fN$, a new message will be transmitted in the j^{th} transmission. Otherwise, i.e. $e_{j-1} > fN$, the same message will be retransmitted in the j^{th}

transmission. Without loss of generality, we assume that $cs_j = 0$ and $cs_j = 1$ convey to the receiver the acknowledgement signal and the rejection signal, respectively, of the $(j-1)^{\mathrm{st}}$ received block.

Fig. 4.3 gives a schematic illustration of the transmitted blocks and the concept of a code word in the resulting retransmission strategy. The control bit decoding unit in Fig. 4.2 decodes the control symbol of a block and accepts or discards the previously received block, accordingly. The control symbol must be decoded with a very low error probability because, as will be seen later, the error performance of the strategy depends on the decoding error probability of the control symbols. Since a left-to-right decoder of MRFC schemes for BSC has a large error exponent at rates close to channel capacity, an either left-to-right estimator or right-to-left estimator can be used to estimate the first bit of every transmitted block, i.e. to have $\hat{t}_{j1}$ as an estimate of $t_{j1} = cs_j$.

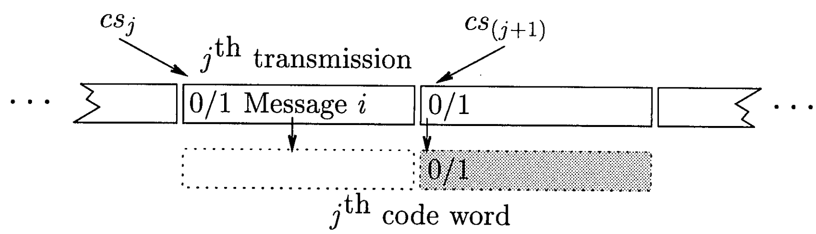


Figure 4.3: The concept of a code word in the 2-block delay strategy.

4.2.3 2-block delay strategy

Given that the control bits are estimated correctly, the erroneous (overflowing) blocks can be corrected. As long as the transmitted blocks corresponding to message $\mathbf{u}_i^{(L-1)}$ overflow, i.e. $e_j > fN$ during transmission of t_j^N, the same message is repeated with initial bit $v_{(j+1)1} = cs_{j+1} = 1$, see Fig. 4.4. As soon as $e_j \leq fN$ for a j, then the next message $\mathbf{u}_{i+1}^{(L-1)}$ is transmitted with $v_{(j+1)1} = cs_{j+1} = 0$, which is schematically shown in Fig. 4.4 by moving to the next row. Only the last block on each row will be decoded by the (right-to-left) block decoding method in the channel-error removing unit of Fig. 4.2.

Decoding

In the decoder, one has to store the two latest received blocks. Let buffers $\mathbf{B_1}$ and $\mathbf{B_2}$ contain two latest received blocks $\mathbf{r}_{j-1}^N$ and $\mathbf{r}_j^N$, respectively. Every

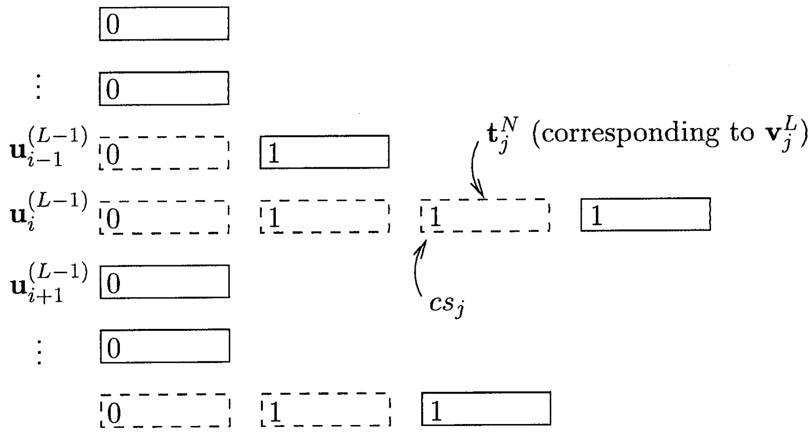


Figure 4.4: An illustration of the transmitted blocks in the 2-block delay strategy. The dashed blocks are discarded at the receiver.

arriving bit of block $\mathbf{r}_j^N$ is written in buffer $\mathbf{B_2}$ and a left-to-right estimator can simultaneously process the estimation of t_{j1}. When r_{jN} is received, the estimation $\hat{t}_{j1}$ is known. Based on $\hat{t}_{j1}$, the decoding of the strategy can be summarized as follows ($j > 1$).

Decoding:

1
- If $\hat{t}_{j1} = 0$, then shift $\mathbf{B_1}$ to the channel-error removing unit.
- If $\hat{t}_{j1} = 1$, then discard $\mathbf{B_1}$.

2 Shift $\mathbf{B_2}$ to $\mathbf{B_1}$.

As we can see, only one bit per received block is decoded by the left-to-right decoding algorithm and those blocks which are given to the channel-error removing unit are decoded by the right-to-left decoding method.

Encoding

In this strategy the decoding error can be caused by the erroneous estimation of the first bit of a block, where an acknowledgement signal is estimated as a rejection signal and vice versa. With a slight modification of the encoding in the block repeating unit, the estimation error of the acknowledgement signal to the rejection signal can be corrected. To achieve this goal, the transmitter itself must also estimate the first transmitted bit of every received block the exact way that the receiver does. Thus, the transmitter becomes aware of the outcome in which the first bit is estimated incorrectly by the estimator of the receiver. There are two following cases.

- $cs_j = t_{j1} = 0$ is estimated as $\hat{t}_{j1} = 1$ ($0 \to 1$ error). In spite of being decodable, $\mathbf{v}_{j-1}^L$ will be discarded by the decoder, see the upper part of Fig. 4.5. Therefore, in the $(j+1)^{\text{st}}$ block we have to retransmit $\mathbf{u}_{i-1}^{(L-1)}$, i.e. the message corresponding to $\mathbf{v}_{j-1}^L$, with $cs_{j+1} = 1$ in order to undo the error. In this way, we can correct the $0 \to 1$ estimation error of the first bit at the cost of discarded blocks $\mathbf{v}_{j-1}^L$ and $\mathbf{v}_j^L$.

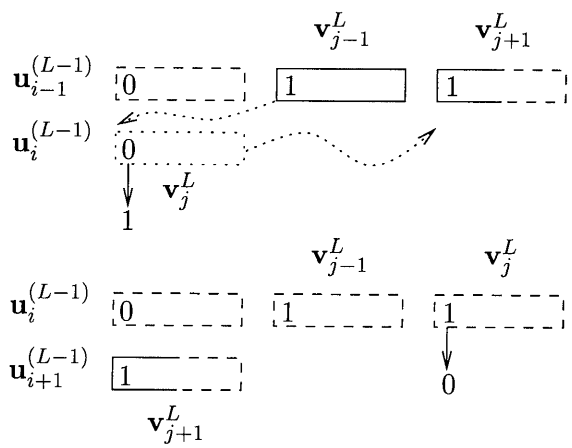


Figure 4.5: The encoding procedure when an acknowledgement to rejection error (above) and when a rejection to acknowledgement error (below) occur.

- $cs_j = t_{j1} = 1$ is estimated as $\hat{t}_{j1} = 0$ ($1 \to 0$ error). In this case the $(j-1)^{\text{st}}$ block will be considered as a decodable block and, therefore, an erroneous block will be produced, see the lower part of Fig. 4.5. The $1 \to 0$ estimation error of the first bit cannot be corrected. In order to prevent the j^{th} block from being considered as another legitimate block, the next block is produced with a new message and $cs_{j+1} = 1$. In this way, every $1 \to 0$ estimation error results in only one erroneous block.

To initiate the encoding process, we have $j = 1$ and $cs_1 = 0$, $i = 1$. To transmit the subsequent blocks, i.e. $j > 1$, the following steps are performed in the encoder.

Encoding:

1
- If $\hat{t}_{(j-1)1} = t_{(j-1)1}$ and
  - $e_{j-1} > fN$ then $cs_j := 1$ and $i := i$,
  - $e_{j-1} \leq fN$ then $cs_j := 0$ and $i := i + 1$.
- An acknowledgement to rejection error[2], i.e. $t_{(j-1)1} = 0$ but $\hat{t}_{(j-1)1} = 1$, then $cs_j := 1$ and (when $i > 1$) $i := i - 1$.
- A rejection to acknowledgement error[3], i.e. $t_{(j-1)1} = 1$ but $\hat{t}_{(j-1)1} = 0$, then $cs_j := 1$ and $i := i + 1$.

2 Obtain $\mathbf{v}_j^L = \Psi(\mathbf{u}_i^{(L-1)}, cs_j)$, append a proper tail and transmit the resulting sequence through the channel.

4.2.4 Error exponent

In this retransmission strategy, a block is erroneously given to the channel-error remover, see Fig. 4.2, with probability P_E when the $1 \to 0$ estimation error of the first bit occurs for the content of buffer $\mathbf{B_2}$.

Given that $b_{21} = 1$ (the first transmitted bit of the block stored in buffer $\mathbf{B_2}$) is estimated to 0, Table 4.1 summarizes the probabilities of different events according to whether b_{11} (the first transmitted bit of the block stored in buffer $\mathbf{B_1}$) is 0 or 1. Let P_r^0 and P_r^1 denote the estimation probabilities corresponding to a $0 \to 1$ error and a $1 \to 0$ error, respectively, and P_o denote the overflow probability of each block. The asymptotic exponents of P_r^0 and P_r^1, denoted by E_F^0 and E_F^1 respectively, are given by (2.18) and that of P_o, denoted by E_o, is given by (2.8). Without loss of generality, let us assume that $P_r^1 \leq P_r^0$. Note that the equality holds in the cases that we are interested in, i.e. the symmetric MRFC schemes for BSC's with forbidden subsequences 01^k and 10^k. We can write

[2]See the upper part of Fig. 4.5.
[3]See the lower part of Fig. 4.5.

	$b_{11} \to \hat{b}_{11}$	Overflowing $\mathbf{B_1}$	Probability
Given $b_{21}=1$ $\downarrow$ 0	$0 \to 0$	no	0 ($b_{21}=1$ is not possible)
	$0 \to 0$	yes	$\leq Pr\{\text{overflowing } \mathbf{B_1}\}$
	$0 \to 1$	–	$Pr\{b_{11}: 0 \to 1\}$
	$1 \to 1$	no	0 ($b_{21}=1$ is not possible)
	$1 \to 1$	yes	$\leq Pr\{\text{overflowing } \mathbf{B_1}\}$
	$1 \to 0$	–	$Pr\{b_{11}: 1 \to 0\}$

Table 4.1: The probabilities of possible events when $b_{21}=1$ is estimated to 0.

$$\begin{aligned} P_E &\leq P_r^1\left(Pr\{B_{11}=0\}(P_o+P_r^0)+Pr\{B_{11}=1\}(P_o+P_r^1)\right) \\ &= P_r^1 P_o + P_r^1\left(Pr\{B_{11}=0\}P_r^0+Pr\{B_{11}=1\}P_r^1\right) \\ &\overset{(\alpha)}{\leq} P_r^1 P_o + P_r^1 P_r^0 \\ &\overset{(\beta)}{\leq} 2P_r^1 P_o, \end{aligned}$$

where (α) comes from the assumption that $P_r^1 \leq P_r^0$ and (β) holds because $P_r^1 \leq P_o$ (according to the corresponding exponents). Since, obviously, $P_E \geq P_r^1 P_o$, we can say that

$$P_E \rightleftharpoons P_r^1 P_o \tag{4.2}$$

where $A \rightleftharpoons B$ means that $\lim_{N\to\infty} \frac{1}{N}\log A = \lim_{N\to\infty} \frac{1}{N}\log B$. To express the exponent, we need to define the constraint length of the strategy. According to Horstein, the constraint length is the number of channel symbols needed to decode an information bit from the moment of entering the transmitter until the time of being decoded at the receiver. In other words, the constraint length is the length of the receiver's buffer needed to decode a message (and the corresponding information bits), see also [37]. The decoder asymptotically needs to have $2N$ channel symbols to decode a message, see also the following subsection. Therefore, from (4.2) we have

$$E_{ret,2} = \lim_{N\to\infty} \frac{-\log P_E}{2N} = \frac{E_F^1 + E_o}{2} \quad \overset{f\to p}{\Longrightarrow} \quad E_{ret,2} = \frac{E_F^1}{2}, \tag{4.3}$$

despite using the left-to-right decoding method only for a single bit.

Remark 4.1 The exponents E_F^1 and E_F^0 hold when p, the channel crossover probability of the BSC, satisfies (2.1), i.e. $p = p_k$. Moreover, according to a conjecture due to Veugen and mentioned in Section 2.3.2, $E_F^1 = D(\frac{1}{k}||p)$ as long as the repetition parameters are chosen as $k =< -\log p >$, where the notation $< a >$ means the closest integer to a. In the retransmission strategy presented in the following section, this exponent rigorously holds even when $p \neq p_k$. □

Remark 4.2 The real error probability, i.e. $Pr\{\hat{\mathbf{U}}^{(L-1)} \neq \mathbf{u}^{(L-1)}\}$, is upper bounded by P_E. If the crossover probability of the BSC $p = p_k$, where p_k satisfies (2.1), then $\hat{R} = R(p_k) = C(p_k)$, see Example 2.4. At $p = p_k$, E_F^1 rigorously holds and the error exponent of any feedback block coding goes to 0 as $R \to C$ (i.e. as $f \to p_k$), see [6, pp. 3]. Therefore, the right-hand side of (4.3) is also the exponent of the real error probability when $p = p_k$ and $f \to p_k$. □

4.2.5 Transmission rate

As shown in Fig. 4.4 and the upper part of Fig. 4.5, the average number of transmissions to send a message correctly satisfies

$$\bar{N}_c \doteq N + P_o\, \bar{N}_c + P_r^0\, (\bar{N}_c + N). \qquad \text{So} \qquad \bar{N}_c \doteq \frac{N(1+P_r^0)}{1-P_o-P_r^0}. \tag{4.4}$$

Notation $\doteq$ means that the equality holds when $1-P_r^1$, $1-P_r^0$ and $1-P_o$ approach one, i.e. for large N's. The probability P_E of erroneous decoding can be made arbitrarily small, as shown before. Therefore, $\bar{N} = \bar{N}_c$ for large N. Also from (4.4) the total transmission rate as $N \to \infty$ (and $L \to \infty$, since $\frac{L}{N} = 1 - kf$) is

$$R_t = \lim_{N\to\infty} \frac{\log M(L)}{\bar{N}} = \lim_{N,L\to\infty} \frac{\log M(L)}{L} \cdot \frac{L}{\bar{N}} = R_0^k(1-k\,f) = R(f),$$

as long as $p < f \leq \frac{1}{k}$.

4.3 M-block delay strategy for BSC's

In order to have a larger error exponent than that of the mentioned strategy, a strategy must be able to reject not only one previous block, but also any number of the previous blocks that are received erroneously. Assume that there are M buffers of length N at the receiver, which are denoted by $\mathbf{B_1}, \ldots, \mathbf{B_M}$, see also Fig. 4.6. For the acknowledgement of the previous block, the current

(j^{th}) block is started with $cs_j = 0$. The remainder bits are obtained from $\mathbf{v}_j^L = \Psi(\mathbf{u}_i^{(L-1)}, 0)$, where $\mathbf{u}_i^{(L-1)}$ is a new message[4]. This is similar to the case mentioned in Subsection 4.2. However, for the rejection of the previous block, not only the current block is started with $v_{j1} = 1$, but also $v_{jn} = 1$ for $n = 2, \ldots, N$. Therefore, the resulting rejection signal, denoted by $\mathbf{v}_{rej}^N$, becomes a **rejection code word** which does not contain any forbidden subsequence nor any message information. Note that the rejection code word does not need a tail, however, the code word is shortened by l bits when a channel error occurs, where l is the number of repetitions needed to correct the $1 \to 0$ error.

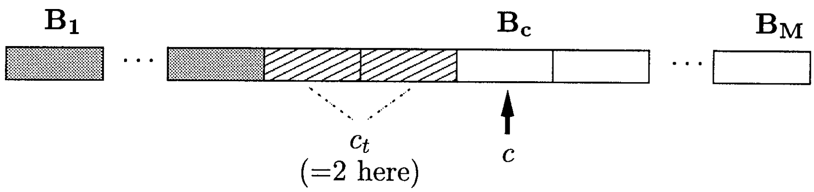


Figure 4.6: Receiver's buffers and the corresponding counters in the M-block delay strategy.

Decoding

Assume that pointer c, $c = 1, 2, \ldots, M$, points to the first empty buffer of the receiver, see Fig. 4.6. The currently received block $\mathbf{r}_j^N$ is stored in $\mathbf{B_c}$, whose first bit is estimated by a fixed coding delay left-to-right (or right-to-left) estimator. The decoding procedure can be summarized as follows.

Decoding:

1. If $\hat{t}_{j1} = 1$, then the stored block in the previous buffer is discarded, i.e. if $c > 1$, then $c := c - 1$.

2. If $\hat{t}_{j1} = 0$, then store the received block in buffer $\mathbf{B_c}$, i.e.
   - if $c < M$, then $c := c + 1$.
   - if $c = M$, then shift the content of buffer $\mathbf{B_1}$ to the channel-error removing unit. Set $\mathbf{B_m} := \mathbf{B_{m+1}}$ for $m = 1, \ldots, (M-1)$.

[4] In general, here the transmitted messages can be considered as those precoded sequences which start with 0, without any reference to a $\mathbf{u}$-like sequence.

Encoding

Let counter c_t at the transmitter denote the number of rejection code words to be sent in order to erase the erroneously accumulated blocks on the top of receiver's buffers, where $0 \leq c_t < c$ (see Fig. 4.6). The erroneous blocks are due to either the wrong decoding of a rejection code word or an overflowing block, for which $\hat{t}_{j1} = 0$. To initiate the encoding process, we have $i = 1$, $j = 1$, $c_t = 0$ and $c = 1$. To transmit the subsequent blocks, i.e. $j \geq 1$, the encoding procedure can be summarized as follows. Note that c in the following denotes the value of receiver's counter at the beginning of each transmitted block (before its modification at the receiver).

Encoding:

1 When $c_t = 0$, a new message is transmitted, i.e. $\mathbf{v}_j^L = \Psi(\mathbf{u}_i^{(L-1)}, 0)$.

- If $\hat{t}_{j1} = 0$, then
  * if $e_j \leq fN$, then $i := i + 1$;
  * if $e_j > fN$, then $c_t := c_t + 1$.
- If $\hat{t}_{j1} = 1$, then[5] (when $i > 1$) $i := i - 1$.

2 When $M > c_t > 0$, rejection message $\mathbf{v}_{rej}^N$ is transmitted.

- If $\hat{t}_{j1} = 1$, then $c_t := c_t - 1$.
- If $\hat{t}_{j1} = 0$,
  * if $c_t < M - 1$, then $c_t := c_t + 1$.
  * if $c_t = M - 1$, then[6] $i := i + 1$.

In this way, the error occurs when buffer $\mathbf{B_1}$ contains an erroneous block (either an overflowing block or a rejection code word whose first bit is estimated erroneously) and buffers $\mathbf{B_m}$, $m = 2, \ldots, M$, are the erroneous receptions of the rejection code word. Therefore, considering the exponent of P_E, i.e. the probability of the error, we can write

$$P_o \left(P_{rej}^1\right)^{(M-1)} \leq P_E \leq \left(P_o + P_{rej}^1\right) \left(P_{rej}^1\right)^{(M-1)},$$

[5] Provided that the $(i-1)^{\text{st}}$ message is not already passed on to the channel-error remover.

[6] Because the erroneous i^{th} message is passed on to the channel-error remover by now.

where P_o and P^1_{rej} denote the probabilities of a block overflow and the estimation of $v_{j1} = 1 \to 0$ (v_{j1} being the first bit of $\mathbf{v}^N_{rej}$), respectively. Let E_o and E^1_{rej} denote the asymptotic exponents of P_o and P^1_{rej}, respectively. Now the constraint length of the coding is $M \cdot N$ [37] and, therefore, the asymptotic exponent of P_E becomes

$$E_{ret,M} = \lim_{N\to\infty} \frac{-\log P_E}{M \cdot N} = \frac{E_o + (M-1)E^1_{rej}}{M} \tag{4.5}$$

By letting $M \to \infty$, we have

$$E_{ret,\infty} = E^1_{rej}. \tag{4.6}$$

Note that as M increases the decoding delay and, consequently, the implementation complexity increase. When $M \to \infty$, the average number of transmissions for a message, $\bar{N}_t \geq N$, can also be written as

$$\begin{aligned} \bar{N}_t &= N + P^0_r(2\bar{N}_t) + Pr\{\text{ overflow} \mid \text{first bit } 0 \to 0\}(\bar{N}_{rej} + \bar{N}_t) \\ &\leq N + P^0_r(2\bar{N}_t) + P_o(\bar{N}_{rej} + \bar{N}_t), \end{aligned}$$

where P^0_r denotes the estimation error probability of the first bit of a non-rejection code word, which starts with a 0, and $P^0_r \leq P_o$. In the above inequality we used the approximate

$$\begin{aligned} Pr\{\text{ overflow} \mid \text{first bit } 0 \to 0\} &= \frac{Pr\{\text{overflow and first bit } 0 \to 0\}}{Pr\{\text{ first bit } 0 \to 0\}} \\ &\leq \frac{P_o}{1 - P^0_r} \doteq P_o \end{aligned}$$

in a received block containing a message, where $\doteq$ indicates that the equality holds when $1 - P^0_r$ approaches one, i.e. for large N. Here $\bar{N}_{rej}$ is the average transmission length of the rejection code word. Since

$$\bar{N}_{rej} = N + 2P^1_{rej}\bar{N}_{rej}, \qquad \text{we have} \qquad \bar{N}_{rej} = \frac{N}{1 - 2P^1_{rej}}.$$

Since $\lim_{N\to\infty} \bar{N}_t = N$, the transmission rate is asymptotically equal to $R(f)$ as given by (2.5), as long as $p < f \leq \frac{1}{k}$.

Remark 4.3 As M increases, the effect of E_o in (4.5) diminishes. Similarly, the effect of the exponent of the real error probability $Pr\{\hat{\mathbf{U}}^{(L-1)} \neq \mathbf{u}^{(L-1)}\}$ will disappear as $M \to \infty$. Therefore, (4.6) holds also for the exponent of $Pr\{\hat{\mathbf{U}}^{(L-1)} \neq \mathbf{u}^{(L-1)}\}$. □

Remark 4.4 The decoding error exponent of (4.6) holds even if the crossover probability of the BSC does not satisfy (2.1). When a right-to-left (or left-to-right) estimator is used for a rejection code word, the probability that its first bit is erroneously received as 0 without being corrected by the end of the estimation delay can be determined from Veugen's approach mentioned in Subsection 2.3.2. As also mentioned in Example 2.5, the exponent of such probability, i.e. $D(\frac{1}{k}\| p)$, is derived from (2.19) and (2.20). Since the rejection code word is not a flip sequence, $D(\frac{1}{k}\| p)$ is the error exponent of the estimator.

To illustrate this, assume that the first bit of the rejection code word is correctly received as 1. Remember that the transmitter always sends 1's for the rejection code word. Similar to the approach of Subsection 2.3.2, a Markov process with states $0, 1, 2, \ldots$ can be defined here. The process starts from state k, moves $(k-1)$ steps upwards when a 1 is received (with probability $q = 1-p$) and one step downwards when a 0 is received (with probability p). Now let us see the exponent of the probability of *asymptotically* entering the 0-state (and perhaps producing the wrong output 0)[7]. Using (2.19) and (2.20), we can write this exponent as

$$\begin{cases} p = q(k-1)\gamma_0^k \\ E_c = -\log(qk\gamma_0^{(k-1)}) \end{cases} \qquad \Rightarrow \qquad E_c = D(\frac{1}{k}\| q).$$

Since $q > p$ for $p < 0.5$, $D(\frac{1}{k}\| q) > D(\frac{1}{k}\| p)$. So it is more likely that the first bit of the rejection code word is received erroneously and cannot be corrected by the end of the estimation delay than that it is correctly received but asymptoticly driven to the wrong estimation. I.e. in the symmetric MRFC schemes for BSC's we have

$$E_{ret,\infty} = E_{rej}^1 = D(\frac{1}{k}\| p),$$

when $f > p$ and $R(f) > 0$. □

Remark 4.5 In the asymmetric 1-up $(k-1)$-down and 1-down $(l-1)$-up schemes for the binary asymmetric channel of Example 2.6, we can also have an argument similar to that of Rem. 4.4 and obtain

$$E_{ret,\infty} = E_{rej}^1 = D(\frac{1}{l}\| p_{10}),$$

[7]Note that if the process enters the 0-state after a limited number of steps, then the error exponent becomes the same as the one mentioned in the previous paragraph.

which holds as long as $D(\frac{1}{l}\|p_{10}) < D(\frac{1}{k}\|(1-p_{10}))$ without requiring that p_{10} and p_{01} must satisfy (2.23) and (2.24). Note that we assume

$$D(\frac{1}{l}\|\ p_{10}) = \max\{D(\frac{1}{l}\|\ p_{10}), D(\frac{1}{k}\|\ p_{01})\},$$

which is the reason to choose the all 1 code word instead of the all 0 code word as $\mathbf{v}_{rej}^N$. Note further that a recursive 1-up $(k-1)$-down and 1-down $(l-1)$-up MRFC scheme on this channel would have $\min\{D(\frac{1}{l}\|\ p_{10}), D(\frac{1}{k}\|\ p_{01})\}$ as the estimation error exponent[8]. □

4.4 Retransmission strategy for general DMC's

The retransmission strategies can easily be generalized to be applicable in non-binary input DMC's. To reject or to acknowledge a received block, the transmitter observes e_{xy}, the total number of $x \to y$ errors $(y \neq x)$ in the block. According to Def. 4.1 a block overflows if $z_j = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} e_{xy}k_{xy} \geq N-L$.

In the two-block delay strategy for strict-sense symmetric channels or wide-sense symmetric channels, see the first paragraph of Section 2.3 for the definitions, two precoded sequences should correspond to every message, one starting with x and the other starting with a symbol from $\mathcal{X} - \{x\}$. To reject or to acknowledge the previous block, the transmitter uses the first or the second precoded sequence, respectively. In this way, the error exponent of $0.5E_1$ can be obtained using the conjecture mentioned in Subsection 2.3.2. In such symmetric MRFC schemes, where the repetition parameter k_{xy} depends on modulo $|\mathcal{X}|$ difference between x and y, one may define a modulo $|\mathcal{X}|$ operation similar to (4.1) in order to simplify the adaptation of the precoded sequences to the control symbols.

For the M-block delay strategy, the error exponent $E_1^{x'} = \max_x E_1^x$ can be attained $(M \to \infty)$ because rejection code word $\mathbf{v}_r^N = (x', x', \ldots, x')$ is not a flip sequence (provided that for the DMC an argument similar to that of Rem. 4.5 can be established using relations (2.19) and (2.20)). As an insightful case, consider the MRFC schemes for the strict-sense symmetric channels. In these channels $p_{xy} = p$ for $y \neq x$ and, therefore, $(1-|\mathcal{X}|)\,p = 1-p_{xx}$. The repetition parameters of the corresponding MRFC scheme are $k_{xy} = k$ for $y \neq x$. The exponent of the probability that the first bit of the rejection code word is received as y and cannot be corrected by the end of the estimation delay, is

$$E_{ret,\infty} = E_r^x = D(\frac{1}{k}\|\ (1-p_{xx})).$$

[8]The proof of this statement at crossover probabilities p_{10} and p_{01} satisfying (2.23) and (2.24) follows the one presented in [66] (or can directly be derived from (2.19) and (2.20)).

Assume that the first bit of the rejection code word is correctly received as x. The *asymptotic* probability that the control symbol is estimated as, let us say, y can be determined by using a Markov process with states $0, 1, 2, \ldots$. The process starts from state k, moves $(k-1)$ steps upwards when a symbol from $\mathcal{X}-\{y\}$ is received (with probability $1-p$) and one step downwards when a y is received (with probability p). The exponent of the probability of *asymptotically* entering the 0-state (and perhaps producing the wrong output y) is $D(\frac{1}{k}\|(1-p))$. Since $p_{xx} > p$, $D(\frac{1}{k}\| (1-p_{xx})) < D(\frac{1}{k}\|(1-p))$ and, therefore, $D(\frac{1}{k}\| (1-p_{xx}))$ is the error exponent of the strategy.

4.5 Conclusions

There are two different coding methods for MRFC schemes, namely right-to-left and left-to-right decoding methods. A brief comparison shows that in the right-to-left decoder, one needs a k_{xy} symbol pattern search per received symbol while in the left-to-right decoder one needs a walk of D symbols per received symbol, $D \gg k_{xy}$. On the other hand, the left-to-right decoder needs a buffer for D symbols while the right-to-left decoder needs a buffer for N symbols, $D \ll N$.

Using a two-block delay retransmission strategy and at the cost of an extra buffer of size N (to store one previous block), the block error probability of the strategy is significantly improved with respect to the block error probability of MRFC schemes. Later, another strategy with unlimited delay is presented, whose exponent of the block error probability is equal to or even larger than that of the corresponding (block and) recursive MRFC, e.g. see Rem. 4.5. Here we assume that there is no buffering problem at the transmitter, i.e. there is no limitation on the number of stored data symbols at the transmitter. This assumption can be justified in cases where, for example, data is stored in large files. With some extra calculation, the effect of the sequential arriving of data can be taken into consideration, see Kudryashov [37].

Chapter 5

MRFC for soft-output DMC's, basics

In this and the following chapters MRFC scheme are modified to be used on the so-called soft-output DMC's. Using a specific class of compound channels, we set up a base for defining the soft-output DMC's in the beginning of this chapter. This setting up is also used to prove the feedback capacity of DMC's in a general sense in Section 5.2. In the remainder of the chapter, we explain the basics of the mentioned modification of the MRFC schemes for using them in the soft-output DMC's. The detailed description of the coding methods of the new scheme is presented in the following chapter.

5.1 A class of compound channels

Let us start with a finite collection $\{K_1, K_2, \dots, K_h\}$ of DMC's with the same input and output alphabet set $\mathcal{X}$. Denote the capacity of the channel K_s by C_s, $s \in \mathcal{S} = \{1, \dots, h\}$. Consider a channel K which has h distinct possible states $s \in \mathcal{S}$. During every block of N transmissions, channel K is in state $s \in \mathcal{S}$ where it behaves like DMC K_s. In [95] channel K is called a **compound** channel which consists of *component* channels $K_1, K_2, \dots, K_h$. Note that the state of the compound channel K cannot be recognized from the output of the channel because the output alphabets are the same for all component channels.

To study the general class of compound channels the interested reader is referred to [95]. As a special case, we are only interested in a class of compound channels, where the state of the channel is stochastically determined at *every* transmission, see [95, pp. 43]. Let random variables X, Y and S denote the input symbol, the output symbol and the state of channel K, respectively. Denote the probability transition matrix of channel K_s by $\mathbf{P}^s = [p^s_{xy}]$ with entries $p^s_{xy} = p_{Y|X,S}(y|x,s)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$ and $s \in \mathcal{S}$. The index $i = 1, 2, \ldots$ will be used to denote the i^{th} transmission. We assume that the states of the channel are determined by chance according to distribution q_s, $s \in \mathcal{S}$, independently of all previous states and of all previously transmitted and received symbols and the current input of the channel. I.e.

$$Pr\{S_i = s|\mathbf{X}^i = \mathbf{x}^i, \mathbf{Y}^{(i-1)} = \mathbf{y}^{(i-1)}, \mathbf{S}^{(i-1)} = \mathbf{s}^{(i-1)}\} = Pr\{S_i = s\} = q_s.$$

Such a compound channel was considered by Shannon in [71], where the information regarding the channel state is called *side information.* Depending on whether and when the transmitter and/or the receiver know the side information, many cases can arise. Four of these cases are considered in [95, pp. 43], which are also mentioned in the following subsections. Note that only two of the cases mentioned here concern us in this chapter.

5.1.1 No side information at the transmitter and receiver

Designate the resulting channel by K_I when neither the transmitter nor the receiver know the channel state. Channel K_I is a DMC having probability transition matrix $\mathbf{P} = [p_{xy}]$ with entries

$$p_{xy} = \sum_{s=1}^{h} q_s p^s_{xy} \qquad x, y \in \mathcal{X}. \tag{5.1}$$

Let C_I denote the capacity of DMC K_I. For this channel we will show in Section 5.2 that the channel capacity is not increased if the encoder is provided with information regarding the previous outputs of the channel and the previous states of the channel. Here the previous states of the channel are referred to as *delayed side information.*

5.1.2 Side information only at the transmitter

Shannon in [71] considered an interesting case where only the transmitter knows the state of the channel just before the corresponding symbol is sent. In other words, the transmitter knows the state of the channel *casually* and it is not aware of the future states. Designate the resulting channel by $K_{I\!I}$ in this case. To transmit message w in a block of N transmissions, the n^{th} channel input of the block is determined according to encoding rule

$$x_n = f_n(w, \mathbf{s}^n) \qquad n = 1, \ldots, N.$$

In the *non-causal* case, the transmitter anticipates the future states. In other words, the transmitter knows beforehand the state sequence of the channel for the whole block and the encoding rule is $x_n = f(w, \mathbf{s}^N)$, $n = 1, \ldots, N$.

Here we only outline the capacity of the channel for the casual case from [71]. Shannon showed that C_{II}, the capacity of channel K_{II}, is given by the ordinary (without side information) capacity of a DMC, designated by K'_{II}, with the same output alphabet $\mathcal{Y} = \mathcal{X}$ and an input alphabet $\mathcal{X}^h$ consisting of super letters $\mathbf{x}^h = (x_1, x_2, \ldots, x_h)$, where $x_s \in \mathcal{X}$, $s = 1, \ldots, h$. The transition probabilities $p'_{\mathbf{x}y}$ for channel K'_{II} are given by

$$p'_{\mathbf{x}y} = Pr\{Y = y | \mathbf{X}^h = (x_1, x_2, \ldots, x_h)\} = \sum_{s=1}^{h} q_s p^s_{x_s y}.$$

The capacity of channel K'_{II} can be achieved if the encoder uses only $|\mathcal{Y}|$ suitably chosen super inputs, as shown by Shannon for DMC's with more inputs than outputs [70]. Let super input $\mathbf{x}^{*h} = (x^*_1, x^*_2, \ldots, x^*_h)$ be one of these chosen inputs. To transmit symbol $\mathbf{x}^{*h}$ from channel K'_{II}, the real transmitter will put symbol x^*_s to the input of channel K_{II} if the corresponding state of the channel is s, $s \in \mathcal{S}$.

5.1.3 Side information only at the receiver

Assume that the channel side information for each letter is available only to the receiver[1] and denote the resulting channel by K_{III}. Such side information is sometimes referred to as soft decision information [39]. Channel K_{III} can be considered as a DMC with input and output alphabet $\mathcal{X}$ and $\mathcal{X} \times \mathcal{S}$, respectively, and transition probabilities

$$p_{(y,s)x} = Pr\{S = s\} Pr\{Y = y | X = x, S = s\} = q_s p^s_{xy}, \quad x, y \in \mathcal{X}, s \in \mathcal{S},$$

see [95, pp. 45]. Channel K_{III} will be referred to as a **soft-output** DMC. We have

$$\begin{aligned} I(X; Y, S) &= I(X; S) + I(X; Y|S) \\ &\overset{(a)}{=} I(X; Y|S) \\ &= \sum_{s=1}^{h} q_s I(X; Y|S = s) \end{aligned}$$

[1]Here we assume that every state of the channel is known to the receiver immediately after receiving the corresponding symbol (to facilitate the concept of feedback for this channel).

where (α) is due to the independence of X and S. Let us denote the capacity of K_{III} by C_{III}. Then

$$C_{III} = \max_{p_X(\cdot)} \sum_{s=1}^{h} q_s I(X;Y|S=s).$$

which is the same as relation (4.6.11) in [95]. If we further assume that the capacity achieving input distribution is the same for all component channels $K_1, \ldots, K_h$, then we can write

$$C_{III} = \sum_{s=1}^{h} q_s \max_{p_X(\cdot)} I(X;Y|S=s) = \sum_{s=1}^{h} q_s C_s, \qquad (5.2)$$

i.e. the capacity is the average of the individual capacities of the component channels. Notice that channel K_{III} is a DMC, therefore, C_{III} is also the capacity of the channel with feedback.

5.1.4 Side information at both transmitter and receiver

The assumption in this case is that the transmitter knows the channel state just before the corresponding symbol is sent or the transmitter knows the channel states of a block just before the corresponding code word is sent. The receiver might have this side information with a delay (since the decoding is normally carried out with some delay). The capacity of the resulting channel is given by (4.6.2) in [95] and can be written as

$$C_{IV} = \sum_{s=1}^{h} q_s C_s.$$

5.2 Delayed side information at the transmitter

In this section we assume that at every transmission, the transmitter knows all previous states[2] of channel K_I. Moreover, suppose that the transmitter observes the outputs of the channel via a noiseless feedback channel (complete feedback), see Fig. 5.1. For some channels the complete feedback can be seen as a form of delayed side information at the transmitter. For example, in a BSC with crossover probabilities p, two component BSC's can be recognized with crossover probabilities 1 and 0, which occur with probability p and $1-p$,

[2]Therefore, we use the term "delayed side information" to refer to such information.

respectively. Knowing the previous state of the channel and the corresponding input symbol, the transmitter can know the corresponding output of the channel.

Let us define a $(2^{N \cdot R}, N)$ "feedback & delayed side information" code as mappings

$$x_n = f_n(\mathbf{y}^{(n-1)}, \mathbf{s}^{(n-1)}, w) \qquad n = 1, \ldots, N. \tag{5.3}$$

The encoding rule $f_n(\cdot,\cdot,\cdot)$ is a function of message $w \in \mathcal{W} = \{1, \ldots, 2^{N \cdot R}\}$ and of the previous received-symbols and channel states. The decoding function is defined as $\hat{w} = g(\mathbf{y}^N)$, where $\hat{w} \in \mathcal{W}$. The decoding error probability for the message w is equal to

$$\lambda_w = Pr\{\hat{W} \neq w\}, \tag{5.4}$$

where we assume that w is uniformly distributed on $\mathcal{W}$. The following theorem addresses $C_{FB\&DSI}$, the capacity of such a channel with feedback and delayed side information.

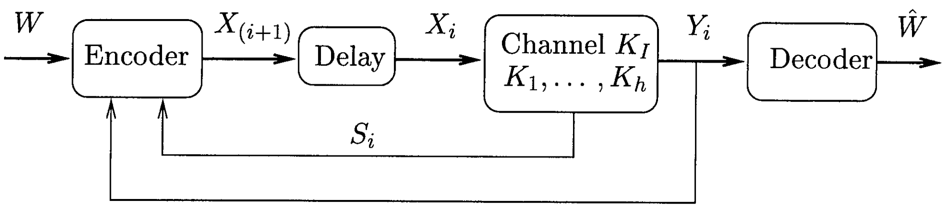


Figure 5.1: A system with complete feedback and delayed side information at the transmitter.

Theorem 5.1 (Complete feedback and delayed side information at the transmitter) *Let C_I denote the capacity of channel K_I defined in Subsection 5.1.1. Then, the channel capacity does not increase if the information about the previous channel outputs and the previous channel states is used in the encoding process, i.e.*

$$C_{FB\&DSI} = C_I. \tag{5.5}$$

Proof : This proof closely follows a similar proof in [16]. Since a one-way code is a special case of a "feedback & delayed side information" code (in such a scheme, use neither the feedback information nor the side information), any rate which can be achieved without feedback and delayed side information can also be achieved with this information, therefore,

$$C_{FB\&DSI} \geq C_I. \tag{5.6}$$

Now let random variable W denote the message, which uniformly takes values from $\mathcal{W}$. Let random variables X and Y denote the input and output symbols of channel K_I, where the transmitter does not know the state of the channel at the time of transmitting X and knowing Y does not give any side information to the receiver. Therefore, the channel transition probability governing random variables X and Y is given by (5.1). We can write

$$H(W) = \log(2^{N \cdot R}) = H(W|\mathbf{Y}^N) + I(W;\mathbf{Y}^N). \tag{5.7}$$

According to the Fano inequality [16, pp. 204]

$$H(W|\mathbf{Y}^N) \leq 1 + NR\lambda_W, \tag{5.8}$$

where λ_W is defined in (5.4). On the other hand, $I(W;\mathbf{Y}^N)$ in (5.7) can be bounded as

$$\begin{aligned}
I(W;\mathbf{Y}^N) &= H(\mathbf{Y}^N) - H(\mathbf{Y}^N|W) \\
&= H(\mathbf{Y}^N) - \sum_{n=1}^{N} H(Y_n|\mathbf{Y}^{(n-1)}, W) \\
&\overset{(\alpha_1)}{\leq} H(\mathbf{Y}^N) - \sum_{n=1}^{N} H(Y_n|\mathbf{Y}^{(n-1)}, \mathbf{S}^{(n-1)}, W) \\
&\overset{(\alpha_2)}{=} H(\mathbf{Y}^N) - \sum_{n=1}^{N} H(Y_n|X_n) \\
&\overset{(\alpha_3)}{\leq} \sum_{n=1}^{N} H(Y_n) - \sum_{n=1}^{N} H(Y_n|X_n) \\
&\overset{(\alpha_4)}{=} \sum_{n=1}^{N} I(X_n;Y_n) \\
&\overset{(\alpha_5)}{\leq} NC_I.
\end{aligned} \tag{5.9}$$

Where (α_1) follows from the fact that entropy decreases by conditioning, i.e.

$$H(Y_n|\mathbf{Y}^{(n-1)}, W) \geq H(Y_n|\mathbf{Y}^{(n-1)}, \mathbf{S}^{(n-1)}, W)$$

and (α_2) holds because X_n is a deterministic function of $\mathbf{Y}^{(n-1)}$, $\mathbf{S}^{(n-1)}$ and W (see (5.3)) and conditional on X_n, Y_n is independent of W and the past samples of Y and S. For inequality (α_3) a bound of entropy is used, see [16]. In (α_4) and (α_5) the definition of the average mutual information function and the definition of the capacity of a DMC are used. Combining (5.7), (5.8) and (5.9), we obtain

$$NR \leq 1 + NR\lambda_W + NC_I.$$

Dividing both sides of this inequality by N and letting $N \to \infty$, we have

$$R \leq C_I. \tag{5.10}$$

Thus, we cannot achieve any higher rates with complete feedback and delayed side information at the transmitter than we can without it for the mentioned class of channels. From (5.6) and (5.10) we conclude relation (5.5). Q.E.D.

5.3 BIQO channel

Consider a compound BSC having a set of component channels consisting of a strong BSC and a weak BSC, where the corresponding states are designated by s and w, respectively. The crossover probabilities of the strong and weak component channels are denoted by p_s and p_w, respectively, where $p_s \ll p_w$. The states of the channel are independently determined according to $Pr\{S_i = s\} = \varphi$ and $Pr\{S_i = w\} = 1 - \varphi$. The corresponding channel with side information at the receiver is a two-input four-output symmetric[3] DMC, here referred to as **Binary-Input Quaternary-Output (BIQO)** channel. The transition probabilities of the BIQO channel shown in Fig. 5.2 can be related to those of the component BSC's by

$$\begin{aligned} q &= \varphi\,(1-p_s), \\ q' &= (1-\varphi)\,(1-p_w), \\ p' &= (1-\varphi)\,p_w, \\ p &= \varphi\,p_s. \end{aligned}$$

We have such a BIQO channel when, for example, the output of an AWGN channel with antipodal signalling at the input is quantized at four levels. The outputs of the BIQO channel can be characterized by their type and status.

[3] See [25, pp. 92] , for the definition of symmetric DMC's

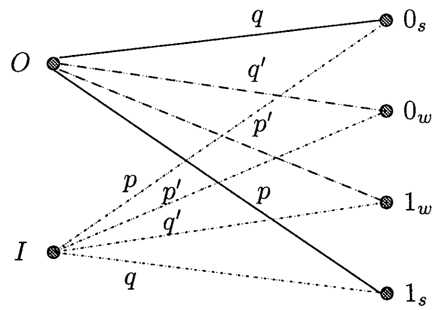


Figure 5.2: A BIQO channel.

Definition 5.1 (Type and status bits) *The* **type** *of an output symbol of a BIQO channel indicates whether the output symbol of the corresponding component BSC is 1 or 0. The* **status** *of an output symbol of a BIQO channel indicates the state of the channel at that reception. Subscripts "s" and "w" designate the strong and weak status of an output symbol, respectively.*

The extension of the domain of MRFC to soft-output DMC's will be illustrated by describing the corresponding scheme for the BIQO channel. For a reference, consider the following example.

Example 5.1 Consider the BIQO channel of Fig. 5.2 with $p = 0.0081$, $p' = 0.0439$, $q' = 0.1457$ and $q = 0.8023$. The capacity of the channel from (5.2) is $C_{BIQO} = 0.7865$ bits. For this channel, if only the type of the output symbols is considered, i.e. completely ignoring the side information at the receiver, then we have a BSC with crossover probability $p + p' = 0.052$ and capacity $C_{BSC} = 0.7052$ bits. On the other hand, if 1_w and 0_w are considered as erasure outputs as Veugen considers in [91], then we have a Binary Erasure Channel (BEC)[4] with capacity $C_{BEC} = 0.7450$ bits. By using existing MRFC schemes, only the capacities of the mentioned BSC and BEC can be approached while there is still a substantial gap to the capacity of the BIQO channel. □

5.4 A new MRFC for BIQO channels

The objective here is to design a simple MRFC method that yields a larger transmission rate than the rates of existing MRFC schemes for soft-output

[4]The BEC defined here is a binary-input ternary-output symmetric DMC, where $p' + q'$ is the crossover probability of the erasure output and p is the crossover probability of the erroneous output. Note that such a BEC differs from the channel shown in Fig. 1.4.

DMC's. In other words, approaching the channel capacity of soft-output DMC's by using a simple repetition strategy is the main concern here, see also [76, 80]. To illustrate the basic idea behind the new technique, we describe the corresponding repetition scheme for the BIQO channel in this and the following chapters. The resulting scheme can be generalized for the use on other soft-output DMC's. Contrary to the previous MRFC methods, here it is necessary to precode adaptively according to the channel output. However, the main advantage of the MRFC schemes, i.e. their simple decoding methods, is preserved in the new repetition schemes. The block and recursive coding methods of the new scheme will be explained in the next chapter.

5.4.1 Introduction

In the encoding process of existing MRFC schemes, the precoding and transmitting units are separate and independent. Nevertheless, we can imagine that the precoder-transmitter in Fig. 2.3 is a single unit which walks through the FSM, produces precoded bits according to message m and transmits them over the BSC. As soon as a channel error is detected via the feedback channel, the precoder-transmitter stalls the walk and immediately inserts k correction bits in front of the list of bits yet to be transmitted. These inserted bits which are called *correction* bits are identical to the erroneously transmitted bit. In a way, we can say that an MRFC scheme forces channel outputs to be the same as channel inputs, see also Section 2.2. Note that the correction subsequence of a precoded bit might be more than k bits due to errors that may occur while transmitting correction bits. Each such error causes k extra transmissions, see also Def. 2.2.

We use the error correcting mechanism and the precoding rule of MRFC schemes to design a similar repetition scheme for the BIQO channels. Let $t_i \in \{0, 1\}$ be the i^{th} transmitted bit and $r_i \in \{0_s, 0_w, 1_w, 1_s\}$ denote the corresponding received symbol. Symbol r_i can be represented by type bit $x_i \in \{0, 1\}$ and status bit $y_i \in \{w, s\}$. Sometimes we will use 0 and 1 notations to indicate a status bit, where $0 \equiv w$ and $1 \equiv s$. Since, for the BIQO channel, the number of inputs is smaller than the number of outputs, we can only force the type of the output symbol to be the same as the input bit. Therefore, the following definitions are made.

Definition 5.2 *A reception is correct if the type of the received symbol is the same as the transmitted bit, i.e. if* $x_i = t_i$*. There is an erroneous reception when* $x_i \neq t_i$*.*

Remark 5.1 From Definition 5.2 it can be concluded that every correct or erroneous reception has either a strong status or a weak status. □

We can summarize the repetition rules (the error correcting mechanisms) of the new scheme as follows.

- Every 0 of the transmitted sequence is allowed to be received as 0_s or 0_w and every 1 is allowed to be received as 1_w or 1_s, i.e. no repetition is needed in these cases.
- If a weak error occurs (e.g. a 0 is received as 1_w) the input bit is repeated k'-times and
- if a strong error occurs (e.g. a 0 is received as 1_s), the input bit is repeated k-times, $k > k' \geq 2$.

When $k' = k$ or $k' = 1$ $(k > 2)$, the resulting scheme is the MRFC scheme for BSC's or for BEC's , respectively. Hereafter, we refer to the new repetition schemes for soft output DMC's as Soft Repetition Feedback Coding (SRFC) schemes.

5.4.2 Encoding

As mentioned earlier, for SRFC schemes it is **necessary** to consider the precoder and the transmitter as a single unit and precode adaptively according to the channel output. This is due to the fact that the state sequence of the channel that determines the lengths of forbidden subsequences is not known beforehand. The encoder of the SRFC scheme for the BIQO channel is depicted in Fig. 5.3. The encoding process comprises the following steps.

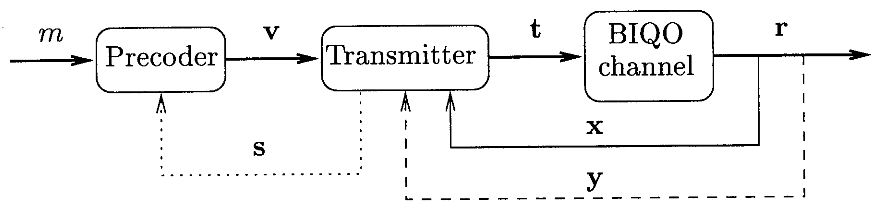


Figure 5.3: Block diagram of the encoder of the SRFC scheme for the BIQO channel.

E1 Being in generic state Q of the FSM and based on message m, the precoder produces the j^{th} precoded bit v_j with $Pr\{V_j = 1|Q\} = \beta$, independently of its previous output bits, see the left hand side of Fig. 5.4.

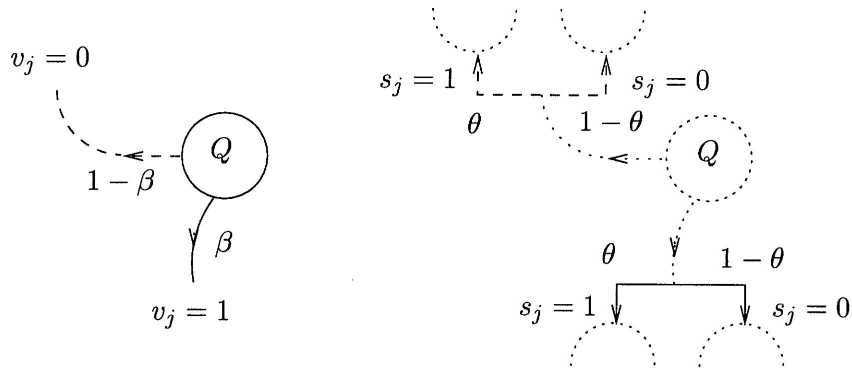


Figure 5.4: Left: producing a precoded bit at a state in the FSM of the SRFC scheme; Right: moving to the next state in the FSM after the **correct** reception of the precoded bit.

E2 The transmitter sends v_j in the i^{th} transmission, i.e. $t_i = v_j$. If t_i is received correctly, set $i' = i$ and go to E3. Otherwise, in case of a weak or a strong error, the transmitter immediately sends the correction bits by repeating t_i for k' or k times, respectively. The repetition rules also apply for every error within these correction bits. Let us say that the last correction bit of v_j is transmitted at the $(i')^{\text{th}}$ transmission. Therefore, subsequence $t_{(i+1)} \dots t_{i'}$ is referred to as the *correction subsequence* of v_j and we can say that v_j is **received correctly** by the $(i')^{\text{th}}$ transmission.

E3 For every precoded bit v_j there exists a corresponding status $s_j = y_{i'}$, where index i' is determined in E2. Having v_j and then knowing s_j determines the next state of the precoder in the FSM, see the right hand side of Fig. 5.4. Since the channel states are drawn according to an iid distribution in every transmission and according to E2 every precoded bit is eventually received correctly, the probability that precoded bit v_j is received strongly, i.e. $s_j = s$, is

$$\theta = Pr\{S_j = s\} = \frac{q}{q+q'}. \tag{5.11}$$

Hereafter, we refer to v_j and s_j as the j^{th} *precoded bit* and its status, respectively. Note that the message determines a precoded bit and the channel determines the corresponding status, see also the following example.

Example 5.2 Let $k' = 3$ and $k = 6$ and consider the following outcomes.

$$\begin{array}{lcl}
\mathbf{v} & = & \ldots\ 1\ \ \ \ \ \ \ \ 0\ \ \ldots\ 1\ 0\ \ \ldots \\
\mathbf{t} & = & \ldots\ 1\ \underline{1\ 1\ 1}\ 0\ \ \ldots\ 1\ \underline{1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1}\ 0\ \ldots \\
\mathbf{x} & = & \ldots\ \underline{0\ 1\ 1\ 1}\ 0\ \ \ldots\ \underline{0\ 1\ 1\ \underline{0\ 1\ 1\ 1}\ 1\ 1\ 1}\ 0\ \ldots \\
\mathbf{y} & = & \ldots\ \underline{w\ -\ -\ *}\ *\ \ \ldots\ \underline{s\ -\ -\ \underline{w\ -\ -\ -}\ -\ -\ *}\ *\ \ldots
\end{array}$$

where a dash represents a don't care status and a $*$ (being s or w) represents the corresponding status of a precoded bit. Underlines in **t** indicate the correction subsequences and those in **x** and **y** represent the forbidden subsequences. □

Remark 5.2 Resulting from the error correcting mechanisms (the repetition rules), $1_w 0_-^{k'}$, $0_w 1_-^{k'}$, $1_s 0_-^{k}$ and $0_s 1_-^{k}$ are the **forbidden subsequences** of the SRFC scheme for the BIQO channel. □

Remark 5.3 The precoding rules of the scheme can be expressed as follows. If a precoded bit changes, then it should not be followed by k' or k similar precoded bits depending on whether its received status is weak or strong, respectively. In summary,

$$\begin{array}{llll}
\text{if} & (v_j = 1 \text{ and } s_j = w) & \text{then} & v_{j+1}, \cdots, v_{j+k'} \neq 0^{k'}; \\
\text{if} & (v_j = 0 \text{ and } s_j = w) & \text{then} & v_{j+1}, \cdots, v_{j+k'} \neq 1^{k'}; \\
\text{if} & (v_j = 1 \text{ and } s_j = s) & \text{then} & v_{j+1}, \cdots, v_{j+k} \neq 0^{k}; \\
\text{if} & (v_j = 0 \text{ and } s_j = s) & \text{then} & v_{j+1}, \cdots, v_{j+k} \neq 1^{k}.
\end{array}$$

As a result, none of the precoded bits will be removed in the decoding stage by the following precoded bits. □

Based on these constraints, let us draw up guidelines on the structure of the FSM of the SRFC scheme (for the BIQO channel). Similar to the guidelines mentioned in Rem. A.1, all precoded sequences **v** with status sequences **s** that end up at given state Q must be restricted by the identical constraints from *producing the next precoded bit* point of view. According to Rem. A.1, the state Q must identify (as an illustrative example, let us assume that $\mathbf{v} = \ldots, 01^l$ and $\mathbf{s} = \ldots, s, -^{(l-1)}, w)$

1. the last bit of the precoded sequence (1 here),

2. the number of identical bits at the end of the precoded sequence (l here) and

3. the previous precoded bit (non identical) to the last precoded bit (0 here).

Furthermore, for the case of SRFC scheme for the BIQO channel, the state must also identify

4. the status bit of the last precoded bit (w here) and

5. the status bit of the previous precoded bit (non identical) to the last precoded bit (s here).

The first and forth rules are essential when the next precoded bit changes with respect to the last precoded bit. The second, third and fifth rules are essential when the next precoded bit does not change.

The FSM of the SRFC scheme with $k' = 3$ and $k = 5$ takes the form indicated in Fig. 5.5. Due to the symmetry of the transmissions of 0's and 1's, the FSM refers to precoded bit changes. The solid transition edges indicate that the corresponding output bit of the precoder is the same as the previous output bit. The dashed transitions indicate a change. Also indicated in Fig. 5.5 are the state transition probabilities, which reflect the Markovian nature of the FSM (see Subsection 5.5.1). After having received the j^{th} precoded bit and knowing its status, moving to state $Q^l_{\bar{s}w}$ indicates that the last $l+1$ bits of $\mathbf{v}$ and $\mathbf{s}$ are $v_{j-l}\dots v_j = 01^l$ or 10^l (the ambiguity between these two is resolved by knowing v_1) and s_{j-l}, s_{j-l+1}, $\dots$, s_{j-1}, $s_j = s, -, \dots, -, w$. Here $-$ denotes a don't care status in accordance with the rules mentioned, i.e. when the current precoded bit is the same as the previous precoded bit, the status of the previous precoded bit is not relevant. This fact is reflected by the FSM having two parallel sets of states which are interconnected such that the status of the last precoded bit at a given depth dictates the state column. In terms of the transition probabilities of the Markov model of the FSM, we have

$$Pr\{V_{j+1} = v_j | V_j = v_j, V_{j-1} = v_j, S_j = s_j, S_{j-1} = s_{j-1}\}$$
$$= Pr\{V_{j+1} = v_j | V_j = v_j, V_{j-1} = v_j, S_j = s_j\}.$$

5.5 Transmission rate

Due to the symmetry of the BIQO channel and the corresponding SRFC scheme with respect to producing and transmitting of 0's and 1's, the average number of correcting transmissions needed to correct a channel error is

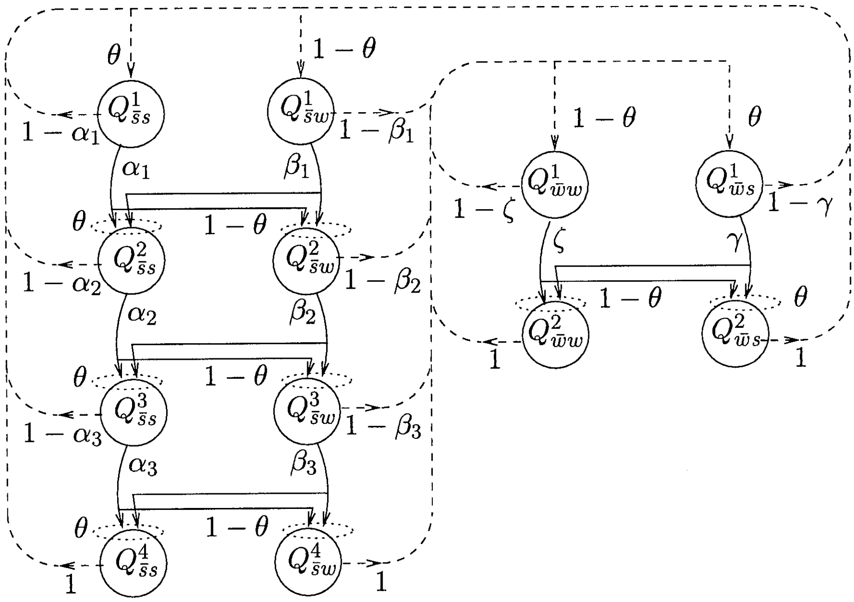


Figure 5.5: The FSM of the SRFC scheme for $k' = 3$ and $k = 5$.

independent from the input of the channel. As a result, we can compute the rate of the precoder and that of the repeater separately. Let us first investigate the FSM of the precoder of the SRFC scheme.

5.5.1 Markov model

The FSM's of the previous MRFC schemes do not depend on the outcome of the symbol transmissions and mapping of messages into constrained sequences can be done before starting the transmissions. However, for the SRFC scheme, the FCM is dependent on parameter θ defined in (5.11) or, in other words, on channel outputs. In the following, we show that the Markov source representing the FSM of the SRFC scheme is an irreducible and unifilar Markov source *given the precoded (type) sequence* $\mathbf{v}$ *and the status sequence* $\mathbf{s}$. Momentarily and only for this purpose, we assume that the Markov source produces a joint symbol vs at every time unit.

Denote the states of the Markov source representing the FSM by Q_ψ, $\psi = 1, \ldots, \Psi$. Let random variable Z_j denote the state of the Markov source in which two independent bits of v_j and s_j (or the joint symbol of $(vs)_j$) are produced at the j^{th} instant, $j = 1, 2, \ldots$, according to joint distribution

$Pr\{(VS)_j = (vs)_j | Z_j = Q_\psi\} = Pr\{V_j = v_j | Z_j = Q_\psi\} \cdot Pr\{S_j = s_j | Z_j = Q_\psi\}$

where

$$\begin{cases} Pr\{V_j = 1|Z_j = Q_\psi\} = 1 - Pr\{V_j = 0|Z_j = Q_\psi\} = p_\psi \\ Pr\{S_j = s_j|Z_j = Q_\psi\} = Pr\{S_j = s_j\} = \begin{cases} \theta & \text{if } s_j = s \\ 1-\theta & \text{if } s_j = w. \end{cases} \end{cases}$$

Then the state of the Markov source is changed from Q_ψ to $Q_{\psi'}$. In fact, the true source produces bit v_j and nature, i.e. the channel, independently produces bit s_j. Thus, the state transition probability from state Q_ψ to state $Q_{\psi'}$, during which joint symbol $(sv)_j$ is produced, becomes

$$q_{\psi\psi'} = Pr\{V_j = v_j|Z_j = Q_\psi\} \cdot Pr\{S_j = s_j\}. \tag{5.12}$$

Remark 5.4 In accordance with the precoding rules, we can deduce that p_ψ is 0 or 1 for those states at which there is only one choice for the next precoded bit (see states $Q^4_{\bar{s}s}$, $Q^4_{\bar{s}w}$, $Q^2_{\bar{w}s}$ and $Q^2_{\bar{w}w}$ in Fig. 5.5). From now on we assume that states Q_ψ, $\psi = 1, \ldots, \Psi'$ are those states at which there are two possible choices for the next precoded bit, i.e. $0 < p_\psi < 1$ for $\psi = 1, \ldots, \Psi'$. □

The state set of this Markov source is *irreducible*, i.e. from each state in the set, no state outside of the set can be reached and every state within the set can be reached within one or more transitions. Moreover, the irreducible set of states is *ergodic* because starting from any state, the numbers of transitions required for the first return to the state have the greatest common divisor one. For an ergodic set of states there is a limiting state distribution $\mathbf{q}^T = (q_1, \ldots, q_\Psi)$ given by the solution to the equations

$$\mathbf{Q}^T \mathbf{q} = \mathbf{q}, \tag{5.13}$$

where $\mathbf{Q} = [q_{\psi\psi'}]$ is the state transition matrix with entries given in (5.12), see [25, pp. 65]. For any ψ and ψ' we have

$$\lim_{j\to\infty} Pr\{Z_j = Q_\psi | Z_1 = Q_{\psi'}\} = q_\psi.$$

This limit is approached exponentially in j [25, pp. 66].

Remark 5.5 If a Markov source with an ergodic set of states is started in a given state infinitely far in the past, then at a given instant j

$$Pr\{Z_j = Q_\psi\} = q_\psi.$$

and the source is stationary and ergodic [25, pp. 66]. □

According to Def. A.1, the Markov source of the SRFC scheme is a unifilar Markov source because the current state Q_ψ and current output $(sv)_j$ uniquely determine the next state $Q_{\psi'}$. Therefore, the entropy rate of the Markov source can be written as

$$\begin{aligned} H_M(V,S) &= \sum_{\psi=1}^{\Psi} q_\psi H((VS)_\psi) \\ &\overset{(\alpha_1)}{=} \sum_{\psi=1}^{\Psi} q_\psi \left(H(V_\psi)+H(S_\psi)\right) \\ &= \sum_{\psi=1}^{\Psi} q_\psi \left(h(p_\psi)+h(\theta)\right) \\ &\overset{(\alpha_2)}{=} \sum_{\psi=1}^{\Psi'} q_\psi h(p_\psi)+h(\theta) \end{aligned} \tag{5.14}$$

where $h(p) = -p\log p - (1-p)\log(1-p)$, (α_1) holds because V_ψ and S_ψ are independent and (α_2) follows from Rem. 5.4. The entropy per precoded bit v (associated with the true data) is $\sum_{\psi=1}^{\Psi'} q_\psi h(p_\psi)$ and the entropy per status bit (associated with channel states) is $h(\theta)$.

Remark 5.6 The Markov source representing the FSM of the precoder practically produces precoded sequence $\mathbf{v}$ with entropy rate $\sum_{\psi=1}^{\Psi'} q_\psi h(p_\psi)$. Therefore, from now on we can forget the notion of the joint source symbol used in Subsection 5.5.1. The significance of knowing both $\mathbf{v}$ and $\mathbf{s}$ sequences by the decoder, see the following chapter, is that the inverse-precoder can precisely trace the state path of the Markov source and subsequently extract the message embedded in sequence $\mathbf{v}$. □

5.5.2 Transmission rate

Let L be the number of precoded bits, i.e. the length of sequence $\mathbf{v}$, in a block of N transmissions. For a given $\theta = \frac{q}{q+q'}$, let $R_0^{k,k'}(q,q')$ denote the precoding rate or, in other words, the asymptotic entropy per output bit of the precoder. Therefore, there are approximately $M = R_0^{k,k'}(q,q')\ L$ information bits in sequence $\mathbf{v}^L$ of L precoded bits. Given state sequence $\mathbf{s}^L$, the precoding rate $R_0^{k,k'}(q,q')$ can be calculated from (5.14) as

$$R_0^{k',k}(q,q') = \lim_{L\to\infty} \frac{M}{L} = \sum_{\psi=1}^{\Psi'} q_\psi h(p_\psi), \tag{5.15}$$

where q_ψ, $\psi = 1, \dots, \Psi$, are the solutions of (5.13).

Remark 5.7 For a given BIQO channel, i.e. given θ, the probabilities of producing 1's and 0's in the states of the Markov source, i.e p_ψ, $\psi = 1, \dots, \Psi'$, can be chosen such that $R_0^{k',k}(q,q')$ be maximized. Let

$$\hat{R}_0^{k',k}(q,q') = \max_{\{0<p_\psi<0:\psi=1,\dots,\Psi'\}} \sum_{\psi=1}^{\Psi'} q_\psi h(p_\psi), \tag{5.16}$$

where q_ψ, $\psi = 1, \dots, \Psi$, are the solutions of (5.13). This, in turn, will maximize the asymptotic transmission rate of the scheme, see (5.17). □

The total transmission rate of the new scheme is the product of the precoding rate and the rate of the repetition part. As $N \to \infty$, the expected numbers of weak and strong errors in the block approach $p'N$ and pN, respectively. According to repetition rules, we have

$$N = L + k'\, p'N + k\, pN \quad \Rightarrow \quad \frac{L}{N} = 1 - k'\, p' - k\, p.$$

Therefore, the total *asymptotic* transmission rate, as $L \to \infty$ and $N \to \infty$, can be written as

$$R = \hat{R}_0^{k',k}(q,q')\ (1 - k'\, p' - k\, p). \tag{5.17}$$

Example 5.3 If we apply the SRFC scheme on the BIQO channel of Example 5.1, the asymptotic transmission rate would be 0.7837 when $k' = 2$ and $k = 7$. This rate is very close to the capacity of the BIQO channel (a difference of 0.0028 bits per transmission) and far better than the rates which can be achieved by the MRFC for BSC's and BEC's. The upper part of Table 5.1 gives the transition probabilities and also the state probabilities of the corresponding FSM obtained from (5.13) and (5.16), where we obtain $\hat{R}_0^{2,7}(q,q') = 0.916041$ and $(1 - k'\, p' - k\, p) = 0.8555$. Using a $k' = 3$ and $k = 7$ SRFC scheme on the same channel yields the transmission rate of 0.7765. The lower part of Table 5.1 gives the transition probabilities and also the state probabilities of the corresponding FSM obtained from (5.13) and (5.16), where we have $\hat{R}_0^{3,7}(q,q') = 0.956691$ and $(1 - k'\, p' - k\, p) = 0.8116$.

i	$Pr\{Q^i_{\bar{s}s}\}$	α_i	$Pr\{Q^i_{\bar{s}w}\}$	β_i
1	0.397743	0.494482	0.072231	0.674635
2	0.207689	0.484411	0.037717	0.665930
3	0.106401	0.463968	0.019323	0.648881
4	0.052391	0.421781	0.009514	0.605879
5	0.023580	0.321738	0.004282	0.5
6	0.008233	0	0.001495	0
i	$Pr\{Q^i_{\bar{w}s}\}$	γ_i	$Pr\{Q^i_{\bar{w}w}\}$	ζ_i
1	0.050272	0	0.009130	0

i	$Pr\{Q^i_{\bar{s}s}\}$	α_i	$Pr\{Q^i_{\bar{s}w}\}$	β_i
1	0.391296	0.495485	0.071061	0.567776
2	0.198229	0.486407	0.035999	0.558741
3	0.098624	0.467720	0.017910	0.539398
4	0.047215	0.427867	0.008574	0.500226
5	0.020727	0.329708	0.003764	0.397614
6	0.007050	0	0.001280	0
i	$Pr\{Q^i_{\bar{w}s}\}$	γ_i	$Pr\{Q^i_{\bar{w}w}\}$	ζ_i
1	0.062046	0.396811	0.011268	0.330178
2	0.021122	0	0.003836	0

Table 5.1: The state and transition probabilities of the Markov sources in Example 5.3, when $k = 7$ and $k' = 2$ (above), $k' = 3$ (below). Note that the parameters in these tables are indicated based on the notations of Fig. 5.5.

5.5.3 Achieving channel capacity

Veugen proved that for every traditional MRFC scheme with given repetition parameters there is a DMC for which capacity can be achieved [90]. However, the repetition rules of the SRFC scheme are such that the correction subsequences are produced regardless of the status of the correcting bits that are received without error. Discarding the status of the properly received correction bits entails a small penalty in rate, but greatly simplifies the encoding and decoding of SRFC schemes as seen in the following chapter.

Chapter 6

MRFC for soft-output DMC's, coding methods

In Chapter 5 the principles of the SRFC scheme for the BIQO channel were presented. In this chapter two coding methods, namely block and recursive methods, of the scheme are discussed in detail. Furthermore, in Section 6.3 we explain a precoding method which can be used for the precoding of all MRFC schemes, including the SRFC scheme. At the end of the chapter, some possible extensions of the SRFC scheme are outlined.

6.1 Block SRFC for the BIQO channel

The block diagram of the block SRFC system for the BIQO channel is shown in Fig. 6.1. We assume that every block consists of L precoded bits denoted by $\mathbf{v}^L$. If e_w weak errors and e_s strong errors occur during the transmission of a block, then $n = L + k'e_w + ke_s$ is the index of the transmission at which v_L is transmitted correctly[1]. Actually n is a random number, but let us first consider an ideal case where somehow the receiver knows n.

Similar to the block decoding of the MRFC scheme for the BSC, the block decoding of the SRFC scheme is carried out in two stages. The **second** stage is the inverse-precoding which is explained in Section 6.3. In the **first** stage, channel errors are detected and corrected by locating and replacing forbidden subsequences. To wit, the decoder moves from the end towards the beginning of received block $\mathbf{r}^n$ (or equivalently $\mathbf{x}^n$ and $\mathbf{y}^n$) and looks for forbidden subsequences as defined in Rem. 5.2. As soon as a forbidden subsequence is found, the type and the status of its last symbol replace the sub-

[1]To see when a precoded bit is transmitted correctly, refer to rule E2 in Chapter 5.

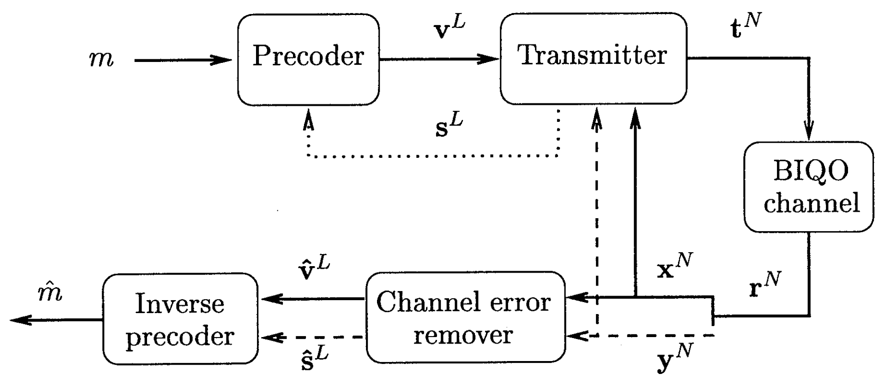


Figure 6.1: The block diagram of the block SRFC system for the BIQO channel.

sequence, compare rule E3 in Chapter 5. For example, $x_i, \ldots, x_{i+k} = 01^k$ and $y_i, \ldots, y_{i+k} = s, -, \ldots, -, *^2$, indicate that a strong $0 \to 1_s$ error has occurred at the i^{th} position, therefore, $x_{i+k} = 1$ and $y_{i+k} = *$ substitutes the subsequences. After the replacements, while moving leftwards to the beginning of the block, the decoder looks for the next forbidden subsequence which starts at position $i' < i$ (the right-to-left decoding). Let $\hat{\mathbf{v}}^L$ and $\hat{\mathbf{s}}^L$ denote the first L bits of the resulting sequences from $\mathbf{x}^N$ and $\mathbf{y}^N$, respectively, when the decoder finally reaches the beginning of the received block. Then, we have the following.

Theorem 6.1 (Ideal block decoding) *Assume L precoded bits are transmitted over a BIQO channel by a SRFC scheme with repetition parameters $k \geq k' \geq 2$. Let n be the index of the transmission corresponding to the L^{th} precoded bit, if received correctly, or its last correction bit, otherwise. If all forbidden subsequences are replaced with their last correction symbols according to the method mentioned above, then*

$$\hat{\mathbf{v}}^L = \mathbf{v}^L \qquad \textit{and} \qquad \hat{\mathbf{s}}^L = \mathbf{s}^L.$$

Proof : Let e_w and e_s denote the number of the weak and strong errors that occurred during the transmission of the L precoded bits, respectively. Thus, $n = L + k'e_w + ke_s$. Let j be the position of the right-most forbidden

[2]Notation $-$ indicates a don't care status bit and $* \in \{w, s\}$.

subsequence in $\mathbf{r}^n$ (or in $\mathbf{x}^n$ and $\mathbf{y}^n$). Without loss of generality, this forbidden subsequence is assumed to be that of a strong error, i.e. $x_j, \dots, x_{j+k} = \bar{a}a^k$ and $y_j = 1$, where $a \in \{0,1\}$, $\bar{0} = 1$ and $\bar{1} = 0$. After removing the error and its following $k-1$ correction symbols, we have the following received sequences

$$\mathbf{x}'^{n'} = \dots, x'_{j-1}, x'_j, x'_{j+1}, \dots, x'_{n'} = \dots, x_{j-1}, x_{j+k}, x_{j+k+1}, \dots, x_n;$$
$$\mathbf{y}'^{n'} = \dots, y'_{j-1}, y'_j, \ y'_{j+1}, \dots, y'_{n'} = \dots, y_{j-1}, y_{j+k}, y_{j+k+1}, \dots, y_n;$$

where $n' = n - k$. We can imagine an other ideal block SRFC scheme at work with transmitted sequence

$$\mathbf{t}'^{n'} = \dots, t'_{j-1}, t'_j, \ t'_{j+1}, \dots, t'_{n'} = \dots, t_{j-1}, t_{j+k}, t_{j+k+1}, \dots, t_n$$

and received sequences $\mathbf{x}'^{n'}$ and $\mathbf{y}'^{n'}$, during which there occurred $e'_w = e_w$ and $e'_s = e_s - 1$ weak and strong errors, respectively. We can see that there is no lost information regarding the status sequence $\mathbf{s}^L$ (and, of course, the type sequence $\mathbf{v}^L$) in the new scenario. Because, when x_{j+k} and y_{j+k} are the last correction bit of a precoded bit, the replacement preserves the corresponding status bit (as well as the type bit). And when x_{j+k} is a middle correction bit of an erroneous precoded bit, then status bit $y'_j = y_{j+k}$ is irrelevant in retrieving the status of the erroneous precoded bit. Note that the next forbidden subsequence of the original scenario, i.e. the first one of the new scenario, starts from position $j' < j$ in $\mathbf{x}'^{n'}$ and $\mathbf{y}'^{n'}$. If this procedure is continued until $j = 1$, all channel errors will be removed from $\mathbf{x}^n$ and $\mathbf{y}^n$ and only the precoded sequence and the corresponding status sequence will be left, i.e. $\hat{\mathbf{v}}^L = \mathbf{v}^L$ and $\hat{\mathbf{s}}^L = \mathbf{s}^L$.
Q.E.D.

6.1.1 Tail adding technique

In the block decoding method of MRFC schemes, only the length of the transmitted block and the length of the precoded sequence, i.e. N and L in the block diagram of Fig. 6.1, are known for both encoder and decoder. Thus, one cannot assume that the receiver knows the transmission position of the last bit of $\mathbf{v}^L$ (or its last correction bit). To perform block SRFC with fixed N and L, we use Schalkwijk's tail adding technique mentioned in [58]. In his version, a tail of length $N - L$, denoted by $z_{L+1} \dots z_N$, is appended to the precoded sequence before transmission. A tail construction is explained in Section 2.2.2 for the case of MRFC scheme for the BSC, where the last precoded bit v_L is alternated. Unlike the precoding of all previous MRFC schemes, which is independent from the transmission procedure, the precoding in SRFC is adaptively done to channel outputs. Therefore, the last bit of precoded sequence $\mathbf{v}^L$ in SRFC is not known beforehand and, consequently, an adaptive tail appending technique must be used in the SRFC schemes.

In a block of N transmissions, the encoder should accumulate L precoded bits as in the ideal case. Let us assume that v_L (or its last correction bit) is transmitted at the n^{th} transmission.

- If $n > N$, then a block decoding error occurs by definition.
- If $n = N$, we have the ideal block coding and the block can be correctly decoded according to Theorem 6.1.
- If $n < N$, then encoder appends an alternating tail of length $N - n$, starting with $z_{n+1} = \bar{v}_L$, where $\bar{0} = 1$ and $\bar{1} = 0$. For the transmission of the tail, whenever a weak or a strong error occurs, the encoder unit uses the same repetition principles and deletes k' or k bits from the end of the tail, respectively.

In the decoder, after removing all forbidden subsequences from $\mathbf{x}^N$ and $\mathbf{y}^N$ by the right-to-left decoding method, the first L bits of the remainders are denoted by $\hat{\mathbf{v}}^L$ and $\hat{\mathbf{s}}^L$, respectively. When $n < N$, the following theorem which is a modification of [91, Theorems 2.1 and 2.2] gives conditions according to which $\hat{\mathbf{v}}^L = \mathbf{v}^L$ and $\hat{\mathbf{s}}^L = \mathbf{s}^L$.

Theorem 6.2 (Block decoding condition) *Let e_w and e_s denote the total number of weak and strong errors in a block of N transmissions in the block SRFC scheme for the BIQO channel. As long as $k'e_w + ke_s < N - L$ and $k \geq k' > 2$, all channel errors can be detected and corrected by the tail appending method, i.e.*

$$\hat{\mathbf{v}}^L = \mathbf{v}^L \qquad \textit{and} \qquad \hat{\mathbf{s}}^L = \mathbf{s}^L.$$

Proof : See Appendix C. Q.E.D.

The following example shows that the block decoding of the SRFC scheme does not necessarily result in correct decoding for $k' = 2$ with conditions mentioned in Theorem 6.2.

Example 6.1 Assume $L = 1$, $v_1 = 1$, $e_w = 1$, $e_s = 0$ and $N = 4$. Consider the following outcomes for a SRFC scheme with $k' = 2$ and $k > 2$.

$$\begin{array}{rcl} \mathbf{v} & = & 1 \\ \mathbf{z} & = & \quad\; 0 \;\; 1 \\ \mathbf{t} & = & 1 \;\; 0 \;\; \underline{1 \;\; 1} \\ \mathbf{x} & = & \underline{1 \;\; 0 \;\; 0} \;\; 1 \\ \mathbf{y} & = & \underline{w \;\; - \;\; w} \;\; - . \end{array}$$

After replacing the forbidden subsequence $\hat{v}_1 = 0 \neq v_1$, even though inequality $k'e_w + ke_s < N - L$ is satisfied. □

6.1.2 Channel-decoding error probability

An upper bound on $P_b = Pr\{\hat{\mathbf{v}}^L \neq \mathbf{v}^L \text{ or } \hat{\mathbf{s}}^L \neq \mathbf{s}^L\}$, the channel-decoding[3] error probability of the block SRFC (with tail appending technique) for the BIQO channel, can be calculated using the Chernoff bound. In a block of N transmissions with L precoded bits, the tail length is

$$T = N - L = \tau N,$$

where τ is the tail fraction of the block[4]. The asymptotic transmission rate can be written as

$$R(\tau) = R_0^{k',k}(q, q')\ (1 - \tau),$$

where $R_0^{k',k}(q, q')$ is the precoding rate. Let random variables E_w and E_s denote the number of weak and strong errors, respectively, in a block. If the number of the required correction transmissions exceeds the tail length, then a block decoding error might occur according to Theorem 6.2, i.e.

$$P_b \leq P_o = Pr\{k'E_w + kE_s \geq T\}, \tag{6.1}$$

where P_o denotes the block overflow probability in accordance with Def. 4.1. We can obtain the asymptotic exponent of the right-hand side of (6.1) as follows. Define

$$E_o = \lim_{N \to \infty} \frac{-\log Pr\{k'E_w + kE_s \geq T\}}{N}. \tag{6.2}$$

[3]The channel-decoding error probability corresponds to the decoding error probability for a block of L **precoded** bits, i.e. a block of N transmissions. As shown in Section 5.5.1 there are asymptotically $LR_0^{k',k}$ information bits in every transmitted block.

[4]For convenience, τN is assumed to be an integer.

Let random variable W denote the necessary number of retransmissions in the block, i.e. $W = k'E_w + kE_s$. An upper bound on $Pr\{W \geq T\}$ can be obtained from the Chernoff bound as

$$P_o = Pr\{W \geq T\} = Pr\{e^{sW} \geq e^{sT}\} \leq \frac{\overline{e^{sW}}}{e^{sT}}, \tag{6.3}$$

where $s > 0$ and the last inequality comes from the Chebyshev bound. The function $g_W(s) = \overline{e^{sW}}$ is called the generating function of random variable W. Random variable W can also be written as $W = \sum_{n=1}^{N} W_n$, where random variable W_n represents the number of induced retransmissions by the n^{th} transmission, $n = 1, \ldots, N$. Random variables $W_1, W_2, \ldots, W_N$ are N iid random variables with $Pr\{W_n = k'\} = p'$, $Pr\{W_n = k\} = p$ and $Pr\{W_n = 0\} = q' + q$, $n = 1, \ldots, N$. Therefore, the generating function can be written as

$$g_W(s) = \overline{e^{sW}} = \prod_{n=1}^{N} \overline{e^{sW_n}} = \prod_{n=1}^{N} g_n(s),$$

where

$$g_n(s) = \overline{e^{sW_n}} = (q + q') + p'e^{sk'} + p\ e^{sk}.$$

Using these notations, the right-hand side of (6.3) can be written as

$$\begin{aligned} g_W(s)e^{-sT} &= \prod_{n=1}^{N} g_n(s)e^{-sN\tau} \\ &= ((q + q')e^{-s\tau} + p'e^{s(k'-\tau)} + pe^{s(k-\tau)})^N. \end{aligned} \tag{6.4}$$

Let us define

$$f(s) = (q + q')e^{-s\tau} + p'e^{s(k'-\tau)} + pe^{s(k-\tau)}, \tag{6.5}$$

which can be minimized at $s_0 \geq 0$, s_0 being the solution of $\frac{d\ f(s)}{d\ s} = 0$, i.e. $s_0 \geq 0$ is the root of equation

$$p'(k' - \tau)e^{sk'} + p(k - \tau)e^{sk} = \tau(q + q'). \tag{6.6}$$

From (6.2), (6.3), (6.4) and (6.5) we have

$$E_o \geq \lim_{N \to \infty} \frac{-\log(g_W(s_0)e^{-s_0 T})}{N} = -\log(f(s_0)), \tag{6.7}$$

where $s_0 \geq 0$ is the solution of (6.6).

Using the approach of [91, pp. 93], one can show that the Chernoff bound is exponentially tight as $N \to \infty$. We have

$$\begin{aligned} Pr\{W \geq T\} &\geq Pr\{k'E_w + kE_s = T\} \\ &\geq Pr\{E_w = e_w, E_s = e_s : k'e_w + ke_s = T\}, \end{aligned} \tag{6.8}$$

where

$$Pr\{E_w = e_w, E_s = e_s\} = \binom{N}{e_w, e_s} (p')^{e_w} p^{e_s} (q+q')^{(N-e_w-e_s)}.$$

Assume $\epsilon_w = \frac{e_w}{N}$ and $\epsilon_s = \frac{e_s}{N}$. From Stirlings formula, $\frac{-\log Pr\{E_w=e_w, E_s=e_s\}}{N}$ can asymptotically be written as

$$\begin{aligned} & \epsilon_w \log \epsilon_w + \epsilon_s \log \epsilon_s + (1-\epsilon_w-\epsilon_s)\log(1-\epsilon_w-\epsilon_s) \\ & -\epsilon_w \log p' - \epsilon_s \log p - (1-\epsilon_w-\epsilon_s)\log(q+q') \\ = \; & \epsilon_w \log \frac{\epsilon_w}{p'} + \epsilon_s \log \frac{\epsilon_s}{p} + (1-\epsilon_w-\epsilon_s)\log \frac{1-\epsilon_w-\epsilon_s}{q+q'}. \end{aligned} \tag{6.9}$$

Using Lagrange multipliers, (6.9) can be minimized under the condition that

$$k'e_w + ke_s = T \qquad \Rightarrow \qquad k'\epsilon_w + k\epsilon_s = \tau, \tag{6.10}$$

where $\tau = \frac{T}{N}$. This results in another condition

$$(\frac{\epsilon_w}{p'})^k = (\frac{\epsilon_s}{p})^{k'} (\frac{1-\epsilon_w-\epsilon_s}{q+q'})^{(k-k')}. \tag{6.11}$$

Relation (6.9) can further be manipulated as

$$\begin{aligned} = \; & \epsilon_w \log \frac{\epsilon_w}{p'} \frac{q+q'}{1-\epsilon_w-\epsilon_s} + \epsilon_s \log \frac{\epsilon_s}{p} \frac{q+q'}{1-\epsilon_w-\epsilon_s} - \log \frac{q+q'}{1-\epsilon_w-\epsilon_s} \\ = \; & \epsilon_w \log \frac{\epsilon_w}{p'} \frac{q+q'}{1-\epsilon_w-\epsilon_s} + \epsilon_s \log \frac{\epsilon_s}{p} \frac{q+q'}{1-\epsilon_w-\epsilon_s} - \\ & -\log\left((q+q') + \epsilon_w \frac{q+q'}{1-\epsilon_w-\epsilon_s} + \epsilon_s \frac{q+q'}{1-\epsilon_w-\epsilon_s} \right) \end{aligned} \tag{6.12}$$

and then can be minimized by using conditions (6.10) and (6.11). For doing so, let us first assume

$$(e^{s_1})^k = \frac{\epsilon_s}{p}\frac{q+q'}{1-\epsilon_w-\epsilon_s} \tag{6.13}$$

and have

$$(e^{s_1})^{k'} = \frac{\epsilon_w}{p'}\frac{q+q'}{1-\epsilon_w-\epsilon_s} \tag{6.14}$$

using (6.11). These equations change (6.12) to

$$\begin{aligned}
&= \epsilon_w \log(e^{s_1})^{k'} + \epsilon_s \log(e^{s_1})^k - \log\left((q+q') + p'(e^{s_1})^{k'} + p(e^{s_1})^k\right)\\
&\overset{(\alpha)}{=} (k'\ \epsilon_w + k\ \epsilon_s)\log e^{s_1} - \log\left((q+q') + p'(e^{s_1})^{k'} + p(e^{s_1})^k\right)\\
&= \tau \log e^{s_1} - \log\left((q+q') + p'(e^{s_1})^{k'} + p(e^{s_1})^k\right)\\
&= -\log\left((q+q')e^{-s_1\tau} + p'(e^{s_1})^{(k'-\tau)} + p(e^{s_1})^{(k-\tau)}\right)\\
&= -\log\left(f(s_1)\right), \qquad (6.15)
\end{aligned}$$

where in (α) condition (6.10) is applied. Furthermore, (6.13) and (6.14) satisfy equation (6.6) as

$$\begin{aligned}
p'(k'-\tau)\frac{\epsilon_w}{p'}\frac{q+q'}{1-\epsilon_w-\epsilon_s} + p(k-\tau)\frac{\epsilon_s}{p}\frac{q+q'}{1-\epsilon_w-\epsilon_s} &=\\
\frac{(k'\epsilon_w + k\epsilon_s) - \tau(\epsilon_w+\epsilon_s)}{1-\epsilon_w-\epsilon_s}(q+q') = \tau(q+q').
\end{aligned}$$

In other words $s_1 = s_0$. From this and (6.8), (6.15) we conclude that

$$E_o = \lim_{N\to\infty} \frac{-\log Pr\{W \geq T\}}{N} \leq -\log(f(s_0)), \tag{6.16}$$

From (6.7) and (6.16), we have $E_o = -\log(f(s_0))$.

Remark 6.1 Note that 2^{-NE_o} is an upper bound on P_b for any N, see (6.1), (6.3), (6.4) and $E_o = -\log(f(s_0))$. □

Example 6.2 Fig. 6.2 shows exponent E_o versus the tail fraction of the block SRFC scheme mentioned in Example 5.3, where repetition parameters $k' = 3$ and $k = 7$ are used. □

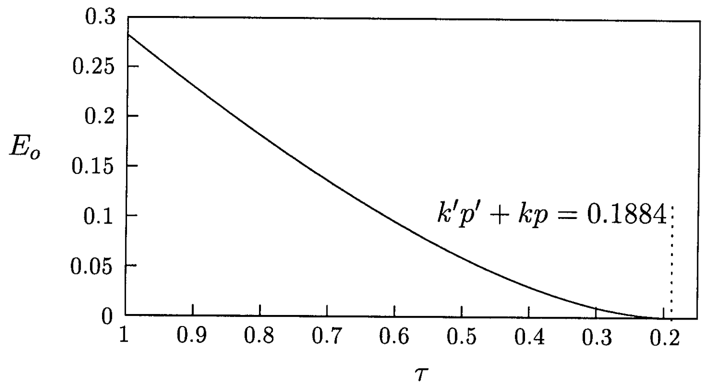


Figure 6.2: Exponent E_o.

6.2 Recursive SRFC for the BIQO channel

Using a left-to-right decoding method, it is possible to decode SRFC schemes with repetition parameters $k > k' \geq 2$. Therefore, SRFC schemes with $k' = 2$ are decodable only by the recursive decoding method. According to our calculations, SRFC with $k' = 2$ gives the best transmission rate for a BIQO channel obtained from quantizing the output of a binary-input AWGN channel. These two facts increase the importance of the left-to-right decoding method. The block diagram of a recursive SRFC system is shown in Fig. 6.3. The decoder consists of three units: an estimator, a channel-error remover and an inverse-precoder. We describe first two units in the following subsections. The inverse-precoder (and the precoder) will be explained in Section 6.3.

6.2.1 Estimator

In this unit the decoder obtains an estimate of the *transmitted* sequence. Let $\hat{\mathbf{t}}$ be the estimate of $\mathbf{t}$. To obtain $\hat{t}_i$, the estimator uses the received sequence $\mathbf{r}_i^{(i+D)} = r_i \ldots r_{i+D}$ and, therefore, a delay of only D transmissions is necessary to estimate every transmitted bit. Two estimation techniques of the recursive MRFC schemes, namely right-to-left and left-to-right estimation methods, are modified in the following subsections in order to be applicable in the new scheme.

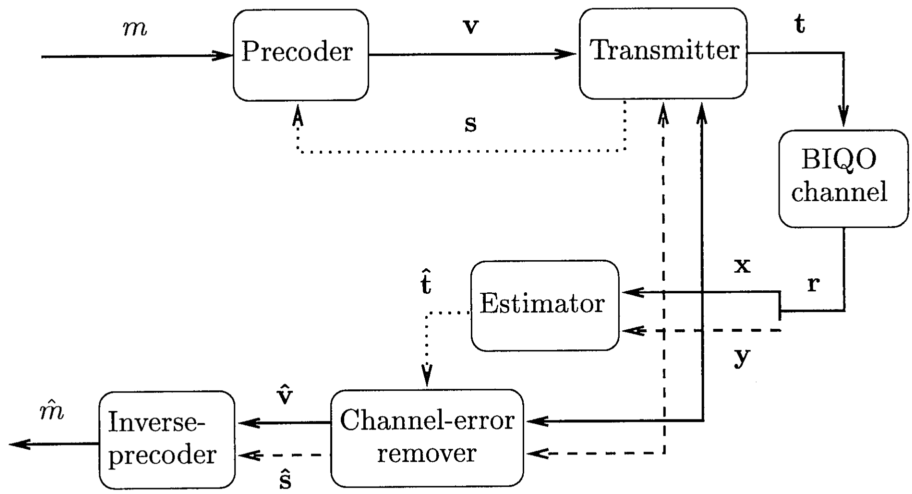


Figure 6.3: Block diagram of the recursive SRFC system.

Left-to-right estimator

Until now, Schalkwijk-Post's left-to-right estimation method has only been applicable in MRFC for binary symmetric and asymmetric channels. The possibility for variable coding delay is the main advantage of the left-to-right estimator over the right-to-left estimator. For binary-input recursive SRFC schemes, a modification of Schalkwijk-Post's left-to-right estimator is presented in this subsection. To estimate t_i from received symbols $\mathbf{r}_i^{(i+D)}$ (or x_{i+j} and y_{i+j}, $j = 0 \ldots D$), the state diagram of Fig.6.4 is used. Starting from the 0-state, the random walk moves right or left according to x_{i+j} and y_{i+j} at the j^{th} step. Let z_j denote the state of the estimator after the j^{th} step with initial state $z_{-1} = 0$. The rules of the random walk can be formalized as follows for step j, $j = 0 \ldots D$.

Left-to-right estimator

1 If $x_{i+j} = 0$, then

$$z_j = \begin{cases} z_{j-1} - 1 & \text{for} \quad z_{j-1} > 0 \\ z_{j-1} - (k-1)y_{i+j} - (k'-1)(1 - y_{i+j}) & \text{for} \quad z_{j-1} \leq 0. \end{cases}$$

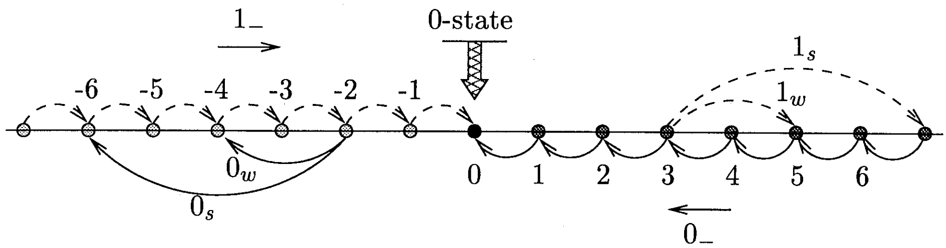


Figure 6.4: State diagram of the left-to-right estimator, $k' = 3$ and $k = 5$.

2 If $x_{i+j} = 1$, then

$$z_j = \begin{cases} z_{j-1} + 1 & \text{for} \quad z_{j-1} < 0 \\ z_{j-1} + (k-1)y_{i+j} + (k'-1)(1-y_{i+j}) & \text{for} \quad z_{j-1} \geq 0. \end{cases}$$

In summary, at the j^{th} step, $j = 0 \ldots D$, the estimator moves left or right if x_{i+j} is a zero or a one, respectively. All transitions towards the 0-state have unit length. All transitions away from the 0-state have a length of $k'-1$ units, if the corresponding received symbol is weak ($y_{i+j} = 0$), and a size of $k-1$ units, otherwise. At the end of the walk, the estimate $\hat{t}_i$ is 0 or 1, according to whether z_D is negative or positive, respectively. In case that $z_D = 0$, Schalkwijk-Post's estimator flips a fair coin. We here propose to choose $\hat{t}_i$ to be 0 or 1, according to whether the last transition to state $z_D = 0$ is initiated from a negative state or from a positive one, respectively. In this way, both right-to-left and left-to-right estimators always yield the same result, see also Theorem 6.3.

Let us briefly explain how the left-to-right estimation method operates by considering the situations which are illustrated in Fig. 6.5. Note that different sizes for the steps away from the 0-state are due to the status bit of the corresponding received symbols being weak or strong.

1. An error has occurred at the i^{th} transmission, i.e. $x_i \neq t_i$. In this case the transmitter tries to correct the error by repeating t_i. For example, consider path f^1 in Fig. 6.5, where without loss of generality we assume that $x_i = 0$. Thus, $z_0 = -(k'-1)$ or $-(k-1)$ depending on whether $y_i = 0$ or 1, respectively. According to the rules of the random walk, the following $k'-1$ (or $k-1$) correction bits bring the walk to the 0-state and the last correction bit gives an equivalent step into the correct

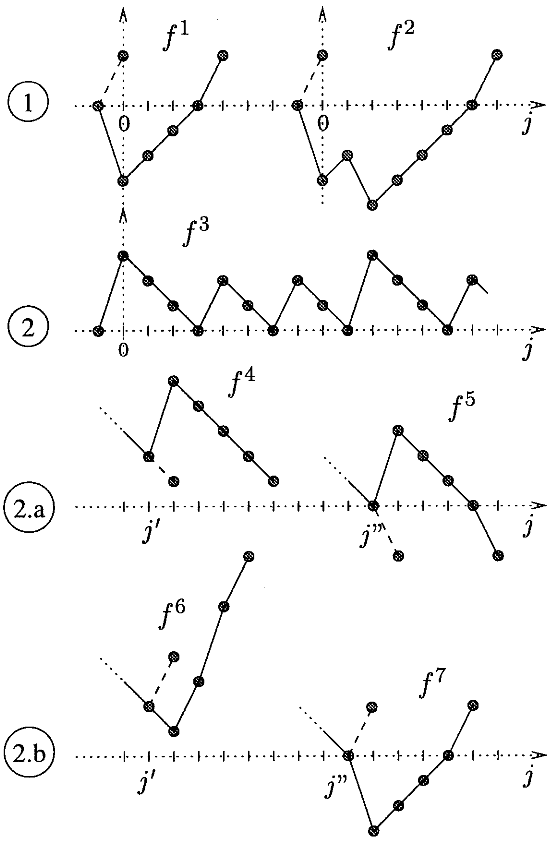


Figure 6.5: Examples for the states of the random walk in different cases.

region, which was initially intended for t_i, see the dashed line on path f^1. Note that every weak or strong error in the correction subsequence of t_i momentarily drifts the walk $k' - 1$ or $k - 1$ steps away from the 0-state and, accordingly, all drifts are pulled back by the corresponding correction bits, see path f^2 in Fig. 6.5.

2. The i^{th} transmission is received correctly, i.e. $x_i = t_i$. For example, assume that $x_i = t_i = 1$ and, therefore, $z_0 = k' - 1$ or $k - 1$ depending on whether $y_i = 0$ or 1, respectively. Note that if t_i is a precoded bit (or the last correction bit of a precoded bit resulting from case 1), it is not possible for any following precoded sequence to bring the random walk to the negative region. Because, in the worst case, the following

precoded sequence would be $0^{k'-1}$ (or 0^{k-1}) followed by any combination of $1_w 0_-^{k'-1}$ and $1_s 0_-^{k-1}$, see path f^3 in Fig. 6.5. If t_i is a correctly received middle correction bit (other than the last one), then $t_{i+1} = 1$ and there is no chance for the following precoded bits to bring the random walk below $z_0 = k' - 1$ (or $z_0 = k - 1$), where the previous statement certainly holds.

Now let us assume that some errors occur during the transmission of the following precoded (or repetition) bits.

2a. A $0 \to 1$ error during transmitting t_{i+j}, $j = 1 \ldots D$, sends the random walk $k' - 1$ units or $k - 1$ units, depending on corresponding y_{i+j}, away from the 0-state and sets $z_j = z_{j-1} + (k' - 1)$ or $z_{j-1} + (k - 1)$, see path f^4 in Fig. 6.5. Consequently, the following k' (or k) correction bits bring the walk back to $z_{j+k} = z_{j-1} - 1$ (exactly to the same place that t_{i+j} intended to, shown by a dashed line on path f^4). Note that paths like f^5 in Fig. 6.5 cannot occur as a consequence of the precoding rule when $x_i = t_i$.

2b. A $1 \to 0$ error occurs during transmitting t_{i+j}, $j = 1 \ldots D$. When $z_{j-1} > 0$, then $z_j = z_{j-1} - 1 \geq 0$ and the following k' (or k, if $y_{i+j} = 1$) transmissions of 1's send the walk away from the 0-state such that, in the worst case, $z_{j+k'} = z_{j-1} - 1 + k'(k' - 1) \geq z_{j-1} + (k' - 1)$, when the right-hand side is the equivalent step if the error did not occur ($k' \geq 2$). Most of the time, the path leads farther into the correct region, see path f^6 in Fig. 6.5. Note that this extra drift will not affect the correct estimation of t_i as the path remains in the correct region. If the $1 \to 0$ error occurs when $z_{j-1} = 0$, we have a similar case to that of path f^1 in Fig. 6.5 and that error also will eventually be corrected for sufficiently large decoding delays, see path f^7 in Fig. 6.5.

Remark 6.2 One of the nice features of a left-to-right estimator is its capability for decoding with a variable coding delay [66]. In the case of fixed coding delay, the estimate is determined after D steps of the random walk. However, with variable coding delay, the estimate is made as soon as the random walk enters one of the so-called absorbing states $z_j \geq r$ or $z_j \leq -s$ for the first time, i.e. for the smallest $j > 0$. The error exponent of the variable coding delay estimation is larger than that of the fixed coding delay estimation in MRFC schemes for binary symmetric and asymmetric channels [66]. For binary input SRFC schemes, we also expect a variable coding delay estimator to perform better than a fixed coding delay estimator. □

Right-to-left estimator

This estimator uses the block decoding method of MRFC schemes to estimate t_i. Introduced in [91], the right-to-left estimator can be used for all MRFC schemes (including our new SRFC scheme), with binary or nonbinary input alphabets. More specifically, the estimator moves from the end to the beginning of sequence $\mathbf{r}_i^{(i+D)}$ and replaces all forbidden subsequences with their last correction symbol. At the end, the type of the left-most remaining symbol is the estimate of t_i. The fact that we are only interested in estimating transmitted bit t_i simplifies the right-to-left decoder for use as an estimator.

A right-to-left estimator needs one register and one counter to keep the most recent symbol and its index, respectively, during the right to left scanning of sequence $\mathbf{r}_i^{(i+D)}$. The right-to-left estimating procedure of the SRFC scheme for the BIQO channel can be formulated as follows, where $\hat{T}_R$ and C_R represent the contents of the register and the counter, respectively.

Right-to-left estimator

1. Set $j := D$, $C_R := 1$ and $\hat{T}_R := x_{i+D}$.

2. Set $j := j - 1$ and
   if $x_{i+j} = \hat{T}_R$ then $C_R := C_R + 1$;
   else $C_R := C_R - (k-1)y_{i+j} - (k'-1)(1-y_{i+j})$.

3. If $C_R < 1$ then set $C_R := 1$ and $\hat{T}_R := x_{i+j}$.

4. If $j > 0$ then go to Step 2; otherwise stop.

At the end, the content of the resister is the estimation of t_i, i.e. $\hat{t}_i := \hat{T}_R$.

In the SRFC schemes for BIQO channels, we can show that the mentioned right-to-left and left-to-right estimators result in the same estimate. Let $\hat{t}_i^{l-t-r}$ and $\hat{t}_i^{r-t-l}$ denote the estimates of t_i by the left-to-right and right-to-left estimators, respectively. Remember that these estimations are based on sequence $\mathbf{r}_i^{(i+D)}$, i.e. x_{i+j} and y_{i+j}, $j = 0 \dots D$. Let $\mathcal{Z}_p$ and $\mathcal{Z}_n$ represent the set of positive and negative states in the state diagram of the left-to-right estimator, respectively (note that the 0-state does not belong to either of them). Let sequence $\mathbf{g}^A$, where $g_a \in \{\mathcal{Z}_p, \mathcal{Z}_n\}$, $a = 1, \dots, A$, denote the sequence of the regions that the random walk **enters** from the beginning to the end of the walk. Note that the random walk does not enter a new region if it reaches the 0-state in the current region.

Theorem 6.3 *In the SRFC scheme for BIQO channels, both right-to-left and left-to-right estimators always result in the same estimate. I.e. we have*

$$\hat{t}_i^{l-t-r} = \hat{t}_i^{r-t-l} = x_{i+j_0} \qquad \text{for } k > k' \geq 2 \text{ and } D \geq 1,$$

where $j_0 \in \{0, 1, \ldots, D\}$ is the index of the first step of the random walk of the left-to-right estimator into region g_A.

Proof : See Appendix C. Q.E.D.

A similar theorem is proven in [91] for the case of MRFC for BSC's where the random walk does not end up at the 0-state. That proof is simple and, with some small modifications, does also apply here. However, the presented proof in Appendix C is more general and also gives an insight to the problem of the left-to-right estimating for nonbinary signalling.

6.2.2 Channel-error remover

To explain the operation of the channel error remover, it is better to have another look at the transmitter. The transmitter repeats an input bit which is received erroneously k' or k times, depending on whether the error is weak or strong, respectively. From the estimator, the channel-error remover obtains estimate $\hat{\mathbf{t}}$ of transmitted sequence $\mathbf{t}$. By comparing $\hat{\mathbf{t}}$ with the received sequence $\mathbf{x}$, the channel error remover obtains the error locations, i.e. the i's where $\hat{t}_i \neq x_i$. Moreover, the corresponding y_i's, i.e. the received status bits at the error positions, determine whether the errors are weak or strong. Now working from left to right in $\mathbf{x}$ and $\mathbf{y}$, the channel-error remover removes weak or strong errors and the following $k'-1$ or $k-1$ **correctly** received correction bits, respectively. In this way, the last correction bit and its status remain on the position of the erroneously received precoded bit and its correction subsequence. Eventually, as long as no estimation error is made we obtain $\hat{\mathbf{v}} = \mathbf{v}$ and $\hat{\mathbf{s}} = \mathbf{s}$.

6.2.3 Estimation error probability

An event error occurs in recursive decoding when the estimator makes an erroneous estimation, i.e.

$$P_r(i) = Pr\{\hat{T}_i \neq t_i\}.$$

where $\hat{T}_i$ is the random variable denoting $\hat{t}_i$. As mentioned in [91], we recognize two different sorts of estimation errors for a fixed coding delay recursive decoder by looking closely at the different outcomes of random walks.

1. An estimation error occurs when a random walk of an erroneous reception enters the correct region *after* D transmissions (paths f^1 and f^2 illustrate this kind of error in Fig. 6.5. Let P_1 be the probability of this kind of estimation error and let

$$E_1 = \lim_{D\to\infty} \frac{-\log P_1}{D}.$$

Using the approach of [91, pp. 62], it can be shown that

$$E_1 = -\log(p'\ k'\ \gamma^{k'-1} + p\ k\ \gamma^{k-1}),$$

where γ is the solution of

$$q + q' = p'(k'-1)\gamma^{k'} + p\ (k-1)\gamma^{k}.$$

2. The other estimation error is caused by certain precoded sequences called flip sequences. The random walk of a flip sequence hugs the 0-state, see path f^3 in Fig. 6.5. A few transmission errors at the end of $r_i r_{i+1} \ldots r_{i+D}$ push the random walk to the wrong region.

In [91, pp. 63], it is conjectured that E_1 is the error exponent for the recursive decoding of MRFC schemes in the range where the corresponding repetition parameters give the maximum transmission rate. We expect the same conjecture to hold for the case of SRFC schemes.

Example 6.3 We simulated the recursive SRFC scheme with $k' = 2$ and $k = 8$ used on a BIQO channel which was obtained from quantizing the output of an AWGN channel, see Section 6.4. At arbitrarily chosen point $p_{bsc} = 0.05$, see the same section for the definition of p_{bsc}, no estimation error was observed during 10^9 estimations for a randomly produced sequence $\mathbf{v}$ when $D = 150$. □

Remark 6.3 Notice that if the M-block retransmission strategy of Chapter 4 is employed with this SRFC scheme, then it can be shown that E_1 is the error exponent of the resulting scheme as $M \to \infty$ and $k' \geq 3$. □

6.3 Precoding and inverse-precoding

In the simplest case, we can assume that each state of a FSM represents a source which produces a sequence of iid bits. Let α be the probability of producing the same precoded bit as the previous one in the state Q of the Markov source modelling the FSM, e.g. see Fig. 5.4. Then the corresponding source of state Q produces sequence $\mathbf{u}(Q)$ consisting of iid bits $u_i(Q)$, $i = 1, 2, \ldots$, with $Pr\{U_i(Q) = 0\} = \alpha$. During transmission, the precoder goes through the FSM and at each state changes its output bit if the corresponding source of the state produces 1. Knowing the initial state of the precoder and obtaining $\hat{\mathbf{v}} = \mathbf{v}$ and $\hat{\mathbf{s}} = \mathbf{s}$, the inverse-precoder can trace the state sequence of the precoder. Thus, the sequences of all sources can be extracted from $\hat{\mathbf{v}}$.

In a more practical case, the precoding problem can be seen as mapping an arbitrary data string onto a string of allowed channel bits, i.e. a mapping of message m onto precoded sequence $\mathbf{v}$, according to status sequence $\mathbf{s}$ which is, in turn, dictated by the BIQO channel during transmission. There are many methods to perform this mapping for MRFC schemes, see [91, Chapter 7]. In this dissertation we mention another method which is very interesting from the SRFC point of view. This precoding method uses an arithmetic decoder and an arithmetic encoder to precode and inverse precode, respectively. In this method, precoding and transmission can be done simultaneously (adaptive precoding). In fact, such a precoder operates pretty much similar to the method mentioned in the previous paragraph. In the following subsections, the idea of using arithmetic coding for precoding is explained.

6.3.1 Precoder

Assume that the message is a sequence of iid bits $w_1 w_2 \ldots$ with $p_W(0) = p_W(1) = 0.5$. If a binary point is put to the left of the message sequence of infinite length, the message may then be represented by a point in the interval $[0, 1)$ and it is called a **message point** and designated by $m \in [0, 1)$. We can say that the message point is uniformly distributed in the unit interval, i.e. $M \sim U[0, 1)$, where M is the random variable denoting message point m. Using an arithmetic decoder, we can map the most-significant bits (see Subsection 6.3.2) of this message point onto L symbols, where a given FSM (or corresponding Markov source model) governs the mapping procedure. Without loss of generality, we assume that the precoded symbols are binary and the FSM is that of either the SRFC scheme or the MRFC scheme for BSC's.

Assume that the Markov source of a precoder consists of states Q_ψ, $\psi = 1, \ldots, \Psi$, at which the probability of producing 0 is p_ψ, respectively. Let $z_l \in \{Q_1, \ldots, Q_\Psi\}$ denote the state of the Markov source, at which the l^{th}

precoded bit, i.e. v_l, is produced, $l = 1, \ldots, L$, see also Fig. 6.6. Let $\alpha_l = Pr\{V_l = 0\}$, where $\alpha_l = p_\psi$ if $z_l = Q_\psi$, $\psi = 1, \ldots, \Psi$.

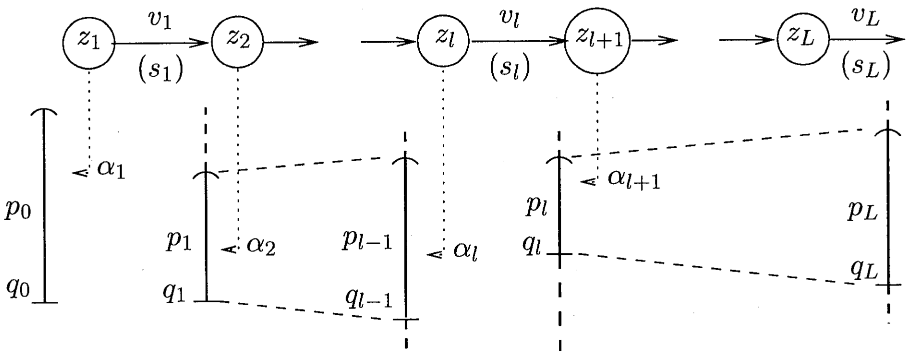


Figure 6.6: An illustration of the precoding procedure.

To begin the precoding process, assume that the precoder starts from state Q_ψ, $\psi \in \{1, \ldots, \Psi\}$, i.e. $z_1 = Q_\psi$. To produce the first precoded bit v_1, m is compared with α_1 portion of interval $[0, 1)$. Depending on whether or not m lies below α_1, $v_1 = 0$ or $v_1 = 1$, respectively. Since the message point m is uniformly distributed in interval $[0.1)$, the

$$Pr\{V_1 = 0|z_1\} = 1 - Pr\{V_1 = 1|z_1\} = \alpha_1.$$

Knowing v_1, we have

$$\begin{cases} M \sim U[0, \alpha_1) & \text{if} \quad v_1 = 0 \\ M \sim U[\alpha_1, 1) & \text{if} \quad v_1 = 1. \end{cases}$$

Let us assume $m < \alpha_1$, then $v_1 = 0$ and, therefore, M is uniformly distributed in interval $[0, \alpha_1)$. Knowing v_1 (and s_1 in the case of SRFC schemes) determines the second state of the FSM, i.e. z_2, and corresponding α_2. Depending on whether or not m lies below α_2 portion of interval $[0, \alpha_1)$, $v_2 = 0$ or $v_2 = 1$, respectively, where $Pr\{V_2 = 0|z_2\} = 1 - Pr\{V_2 = 1|z_2\} = \alpha_2$. Continuing this procedure, the algorithm successively narrows the interval of $[q_0, q_0 + p_0) = [0, 1)$ to $[q_l, q_l + p_l)$ when l precoded bits of $\mathbf{v}^l$ (and the corresponding status sequence of $\mathbf{s}^{(l-1)}$ in the case of SRFC) are produced. We can observe that

- the interval encompasses message point m, i.e. $q_l \leq m < q_l + p_l$,

- the message point is uniformly distributed in interval $[q_l, q_l + p_l)$ conditioned on $\mathbf{v}^l$ and z_1 (and on $\mathbf{s}^{(l-1)}$ in the case of SRFC) and
- the size of interval is equal to $p_l = \prod_{i=1}^{l} Pr\{V_i = v_i | \mathbf{v}^{(l-1)}, (\mathbf{s}^{(l-1)}), z_1\}$, which is the probability of sequence $\mathbf{v}^l$ produced by the Markov source.
- q_l represents the accumulative probability of all precoded sequences of length l which are lexicographically smaller than $\mathbf{v}^l$ given z_1 (and $\mathbf{s}^{(l-1)}$).

Since the Markov sources of the precoders are unifilar, the state sequence of the Markov source is governed by the deterministic rule of the FCM, i.e. $z_{l+1} = g(z_l, v_l)$ in the FSM of MRFC schemes for BSC's (and $z_{l+1} = g(z_l, v_l, s_l)$ in the case of SRFC schemes). Having z_{l+1} and depending on whether or not m lies below α_{l+1} portion of interval $[q_l, q_l+p_l)$, $v_{l+1} = 0$ or $v_{l+1} = 1$, respectively.

6.3.2 Inverse-precoder

Conversely, after having L precoded bits, we can recompute the boundaries of interval $[q_L, q_L + p_L)$. Every precoded bit v_l , $l = 1, \ldots, L$, can determine which of $[q_{l-1}, q_{l-1} + \alpha_l \cdot p_{l-1})$ or $[q_{l-1} + \alpha_l \cdot p_{l-1}, q_{l-1} + p_{l-1})$ intervals must be taken as interval $[q_l, q_l + p_l)$. Due to the Markov source being unifilar, the state sequence of the Markov source and the sequence of α^L are also known to the inverse-precoder.

At the end, we can say that interval $[q_L, q_L + p_L)$, which represents all message points $q_L \leq m < q_L + p_L$, becomes known to the receiver. Since p_L is a very small probability value, $\mathbf{v}^L$ and therefore interval $[q_L, q_L + p_L)$ uniquely represents the most-significant bits of messages $q_L \leq m < q_L + p_L$. Let L' be the number of these most-significant bits. We can then claim that L' information bits are transmitted to the receiver (provided that $\hat{\mathbf{v}}^L = \mathbf{v}^L$ and $\hat{\mathbf{s}}^L = \mathbf{s}^L$).

6.3.3 Arithmetic coder

The mentioned (Elias) procedure can be implemented by an arithmetic coder which approximates q_l and p_l by $\tilde{q}_l$ and $\tilde{p}_l$, respectively, $l = 1, \ldots, L$. An arithmetic coder uses finite-precision arithmetic where a fixed number of bits are available to represent numbers during calculation. In this way, a linear time encoding is created without any serious impact on precoding efficiency. Two implementations of arithmetic coding can be found in [5, Chapter 5] and [33][5]. Some techniques involved in implementing the latter method as

[5]In [38, 83] variations of arithmetic coding are presented which allow a fixed rate (pre-) coding for constrained channels. Generally, the resulting rates are not optimum.

a precoder and inverse-precoder are outlined in the following. For a detailed description of the arithmetic coding algorithm, the interested reader is referred to Ref. [33].

Since $\tilde{p}_l$ is a very small number representing the width of the interval corresponding to the probability of precoded sequence $\mathbf{v}^l$, it contains a lot of leading 0's. A normalization procedure is involved to limit the precision of P_R, representing the least significant bits of $\tilde{p}_l$, to $w+1$ bits. Therefore,

$$\tilde{p}_l = 2^{-(d_l+w+1)} P_R,$$

where d_l represents the number of leading 0's of $\tilde{p}_l$ such that $2^w \leq P_R < 2^{(w+1)}$. To represent $\tilde{q}_l$, however, a register of $(d_l + w + 1)$ bits is necessary in theory. The binary expansion of $\tilde{q}_l$ can be shown as

$$(j_1, j_2, \cdots, j_{(a_l-1)}, 0, \overbrace{1, 1, \ldots, 1}^{b_l \text{ ones}}, j_{(a_l+b_l+1)}, j_{(a_l+b_l+2)}, \cdots, j_{(a_l+b_l+w+1)}).$$

where $a_l + b_l = d_l$. This is divided into four parts from right to left: $w+1$ bits participate in the calculation of the new interval, the preceding run of b_l 1's, the preceding 0 and $(a_l - 1)$ remainder bits. In the calculation of the lower bound of the new interval, i.e. $\tilde{q}_{(l+1)}$, a carry of 1 might be created which would propagate along the run of b_l 1's and increment $j_{a_l} = 0$ to 1, and there no further carries can occur. Therefore, the $(a_l - 1)$ bits of $\tilde{q}_l$ are immune from alteration and can be dispensed with after being used to select the first $(a_l - 1)$ bits of the message point. As a result,

- a $(w + 1)$ bit register and a counter are needed to hold the rightmost part of $\tilde{q}_l$ and the length of preceding run of 1's, respectively.

- These $(a_l - 1)$ dispensed bits take part neither in the precoding nor in the inverse-precoding procedures. Therefore, considering the precoding and inverse-precoding as a whole, we can symbolically say that these $(a_l - 1)$ information bits are conveyed to the receiver side after having produced l precoded bits.

When the lower boundary $\tilde{q}_L$ is obtained, the minimum number of conveyed message bits is $L' = a_L - 1$. In this way, the *termination* problem of this precoding method can be resolved. For the next block, the new message point starts from the bit after the $(L')^{\text{th}}$ message bit of the current message point, see also Rem. 6.6.

Remark 6.4 Note that an arithmetic decoder is used for precoding and an arithmetic encoder is used for inverse-precoding. The arithmetic decoder presented in [33] does not need to calculate $\tilde{q}_l$ because it only keeps track of $m - \tilde{q}_l$. Used as a precoder, the arithmetic decoder must also calculate $\tilde{q}_l$ in order for the transmitter to know the value of L', see also Rem. 6.6. □

Remark 6.5 A MRFC scheme using the arithmetic coder mentioned above might not be able to efficiently transmit certain sequences of message bits. For these message bit sequences, b_L is a large number and, therefore, $L' = a_L - 1$ is very small with respect to the size of the precoded block L. This problem can be prevented when there is not any long run of 1's or 0's in the sequence of a message point. To illustrate this, let us consider the message points which have a large b_l, $l = 1, \ldots, L$. Such ill message points can be bounded as

$$\begin{aligned}(j_1, j_2, \ldots, j_{(a_l-1)}, 0, \overbrace{1, 1, \ldots, 1}^{b_l \text{ ones}}, j_{(a_l+b_l+1)}, \ldots, j_{(a_l+b_l+w+1)}) \leq m \\ < (j_1, j_2, \ldots, j_{(a_l-1)}, 1, \overbrace{0, 0, \ldots, 0}^{b_l \text{ zeroes}}, j'_{(a_l+b_l+1)}, \ldots, j'_{(a_l+b_l+w+1)}),\end{aligned}$$

where b_l is a large number. As a simple remedy to this problem, one can remove all message points in the intervals of above form, i.e. where q_l has more than b_m runs of 1's. If a redundant 0 or 1 is inserted after b_m consecutive 1's or 0's, respectively, in the sequence of the message point m, then another message point m' can be obtained without a run of more than b_m 1's or 0's.

Since b_m can be a rather large number (about a few tens to 100), a simple method known as regular grammar [58, 57, 91] can be used to map message sequences of m into m' and vice versa. For doing this, the state diagram of Fig. 6.7 is used. Starting from an initial state, every change or no change in the bit sequence corresponding to message point m determines a transition along the dashed or solid edges, respectively. In state Q_{b_m}, a change is always imposed, which is shown by a dotted edge in the state diagram of Fig. 6.7.

From another point of view, the state diagram of Fig. 6.7 can be seen as a Markov source having state transition probabilities 0.5 for states $Q_1, \ldots, Q_{b_{m-1}}$. Therefore, the corresponding arithmetic coding is to simply put an input bit into the output in states $Q_1, \ldots, Q_{b_{m-1}}$ and to insert or delete an appropriate redundant bit in state Q_{b_m} at the transmitter or receiver, respectively. The entropy rate of this Markov source can be calculated to give the rate of the mentioned procedure for inserting a bit after every b_m consecutive 1's or 0's. On the other hand, we can roughly argue that the probability of having b_m consecutive 1's or 0's in the message sequence is $2 \cdot (\frac{1}{2})^{b_m}$, which makes the

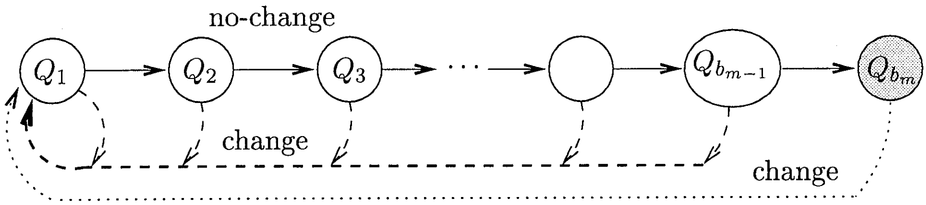


Figure 6.7: State digram used for inserting or deleting bits.

probability of inserting a redundant bit very small if b_m is chosen large enough. Therefore, practically, we do not expect any rate loss when b_m is about a few tens or 100. □

Remark 6.6 In the precoding method mentioned in this section, the length of the sequence of message bits, i.e. L', is variable in the block coding. This causes error propagation in the sequence of the message bits if a block is decoded erroneously, i.e. if $\hat{\mathbf{v}}^L \neq \mathbf{v}^L$ for such a block (in the case of SRFC schemes if $\hat{\mathbf{v}}^L \neq \mathbf{v}^L$ or $\hat{\mathbf{s}}^L \neq \mathbf{s}^L$). To remedy this problem, a system designer can consider one or a combination of the following measures.

- By employing a block retransmission strategy, see Chapter 4, the block error probability can for practical purposes be eliminated.
- The transmitter also decodes every received block (known to the transmitter via the feedback channel) and obtains the L' of the receiver. This parameter is the same for both transmitter and receiver if $\hat{\mathbf{v}}^L = \mathbf{v}^L$ (in the case of SRFC if $\hat{\mathbf{v}}^L = \mathbf{v}^L$ and $\hat{\mathbf{s}}^L = \mathbf{s}^L$). In case of a block error, the transmitter starts to transmit the next message bit corresponding to the L' of the receiver. In this way, error propagation in the sequence of massage bits can be limited. □

Remark 6.7 Precoding for the recursive MRFC or SRFC schemes can be performed similarly, assuming that L is a quite large number, e.g. $L = 10^6$. Note that all procedures involved in a recursive MRFC scheme, i.e. the redundant bit insertion, arithmetic coding and channel error removing, are left-to-right operations, in accordance with the definition of a recursive coding method. □

6.3.4 Simulation results

The precoding technique is implemented in the precoding of the MRFC scheme for BSC's with $k = 6$. Using the maxentropic Markov source model of the

FSM, see Appendix A, message bits are mapped onto precoded bits and the original message bits are retrieved from the precoded bits. Fig. 6.8 shows the simulation results of the precoding rate in terms of L, the size of the precoded blocks. The precoding and inverse-precoding procedures are conducted 10^9 to 10^7 times, depending on the value of L, and the average precoding rate is calculated for a given L. As L increases, the precoding rate approaches the entropy rate of the corresponding Markov source.

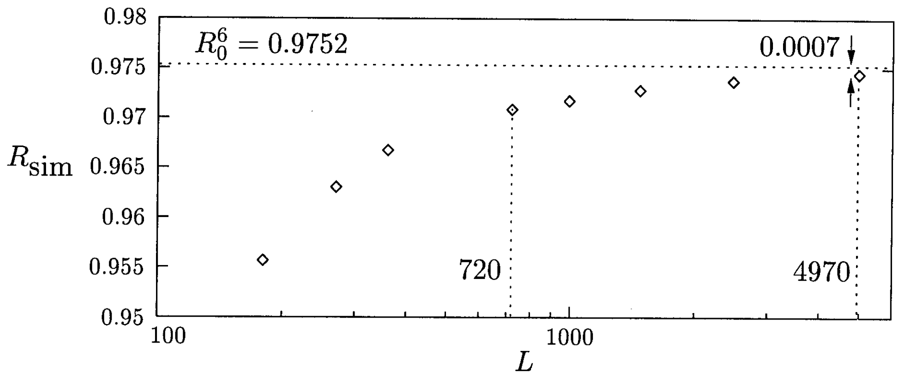


Figure 6.8: Simulation results of the precoding (MRFC scheme for BSC's with $k = 6$).

At small block lengths, the inefficiency of the precoding method is partly due to the overhead in the implemented arithmetic coding algorithm, where the loss is proportional to the inverse block length. Another reason is that the Markov source representing the FSM has not yet reached its asymptotic state probabilities, [25, pp. 60]. Fig. 6.9 shows the simulation results of the precoding rate for the SRFC scheme mentioned in Example 5.3 with repetition parameters $k' = 3$, $k = 7$ and the Markov source parameters given in Table 5.1. Comparing the simulations results of Figs. 6.8 and 6.9 shows that the rates of convergence to the asymptotic value are very similar in both cases.

The behaviour of Markov chains can be analyzed by using the theory of types, see for example [17, 3]. Consider an irreducible Markov source with stationary probabilities q_{ij} on the edges e_{ij}, i.e. $Pr\{$ transition along $e_{ij}\}$ $= q_{ij}$, where e_{ij} represents the edge from state Q_i to Q_j, of a finite directed graph Φ. For $L \geq 1$ we can consider the empirical type

$$\hat{\mathbf{p}}^L(e_{ij}) = \frac{1}{L}\sum_{l=1}^{L} \delta_{ij}(l).$$

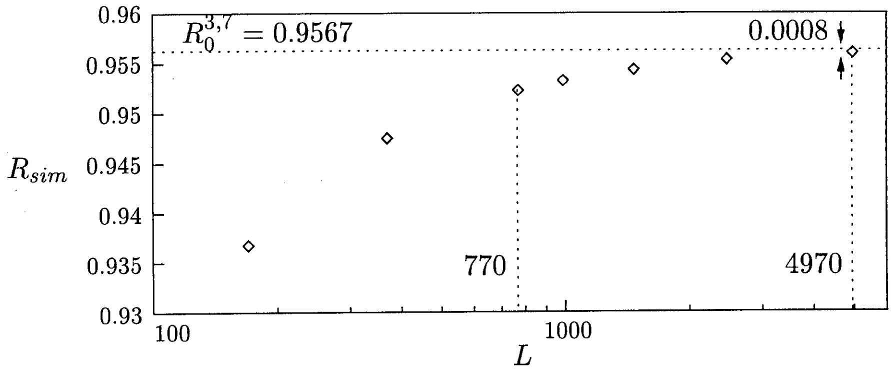


Figure 6.9: Simulation results of the precoding (SRFC scheme with $k' = 3$,$k = 7$).

where $\delta_{ij}(l)$ is 1 if the l^{th} transition of the source is along edge e_{ij} and it is 0 otherwise. Thus $\hat{\mathbf{p}}^L$ is the empirical distribution of the first L transitions. The true distribution of the Markov source in denoted by $\mathbf{p}^L$ where $\mathbf{p}^L(e_{ij}) = Pr\{e_{ij}\} = q_{ij}$ for all edges of graph Φ. If the Markov chain is irreducible, then $\hat{\mathbf{p}}^L$ will converge almost surely to the stationary distribution on the set of edges, by the strong law of large numbers [3]. Likewise, the entropy rate of the corresponding sequence will converge to the entropy rate of the Markov source.

6.4 Examples and extensions

6.4.1 SRFC scheme for quantized AWGN channels

Assume x and $-x$ are inputs to an AWGN channel with noise deviation $\sigma = \sqrt{\frac{N_o}{2}}$. If the output of the channel is quantized on three threshold levels of $-\rho$, 0 and ρ, then we obtain a BIQO channel with output symbols $1_s, 1_w, 0_w$ and 0_s. For this channel Fig. 6.10 shows the transmission rate of the SRFC scheme with $k' = 3$ and $k = 10$ as well as the capacity of the BIQO channel and that of the BEC (as it is defined in Example 5.1) resulting from quantizing the output of the AWGN channel in three and two levels, respectively.

Fig. 6.11 shows the same capacities together with the transmission rate of the SRFC scheme with $k' = 2$ and $k = 8$. All these graphs are depicted in terms of p_{bsc}, the cross-over probability of the BSC obtained from binary quantization of the output of the AWGN channel, at different signal to noise

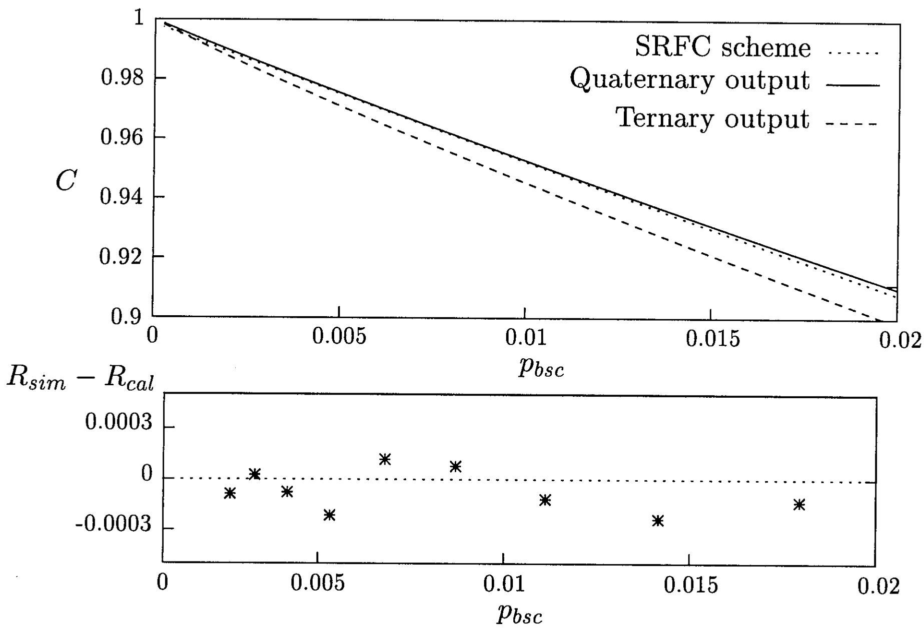


Figure 6.10: The rate of SRFC scheme with $k' = 3$, $k = 10$, for a binary-input quantized-output AWGN channel.

ratios $X^2 = (\frac{x}{\sigma})^2$. In these graphs, those values of threshold $T = \frac{\rho}{\sigma}$ that maximize the corresponding capacity of the BEC or the BIQO channel are considered at a given X, where the rates of the schemes are calculated by maximizing (5.16) with respect to T and the transition probabilities of the corresponding Markov sources. Shown in Fig. 6.10 and Fig. 6.11 are also the simulation results which closely match the theory. According to these calculations, SRFC schemes with $k' = 2$ give higher throughput for such BIQO channels.

6.4.2 Extensions to binary-input five-output channels

From a symmetrical channel with an even number of outputs, the SRFC scheme can easily be extended to a symmetrical channel with an odd number of outputs. For example, consider a 2-input 5-output channel which can be obtained by adding an output e to the BIQO channel such that $Pr\{Y = e|X = 0\} = Pr\{Y = e|X = 1\} = p_e$, where X and Y are random variables denoting the input and output of the channel, respectively. In terms of channel model described in Subsection 5.1.3, output e corresponds to

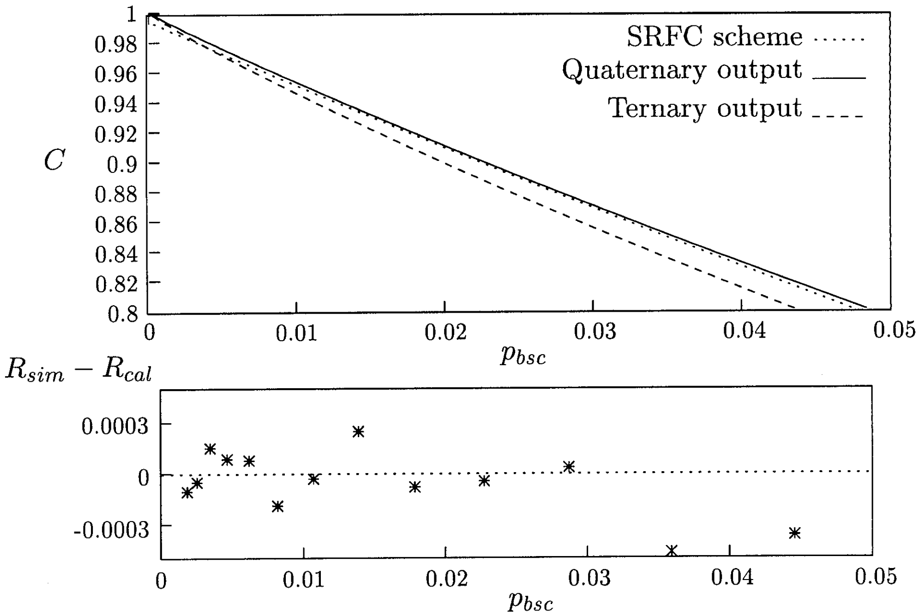


Figure 6.11: The rate of SRFC scheme with $k' = 2$, $k = 8$, for a binary-input quantized-output AWGN channel.

the third channel state in which a BSC with crossover probability 0.5 is chosen. Since output e gives an equal amount of information about the transmitted bit being 0 or 1 (when both inputs are equiprobable), output e is considered an erasure output. Here we can use the SRFC scheme of the BIQO channel, only the transmitter has to repeat every transmission which is received as e. In the receiver, the decoder simply ignores all e receptions. Then, the asymptotic transmission rate can be written as

$$R = R_0^{k',k}(q, q')(1 - p_e - k'\ p' - k\ p).$$

6.4.3 Extensions to some channels with memory

It is possible to apply the idea of adaptive precoding, involved in the SRFC scheme, to block interference channels [39] or a class of finite state channels [42, 27] with side information at the receiver. These channels model the fading channels and the corresponding channel side information is often provided by an automatic gain control device that informs the receiver which component channel was active in the last transmissions [39].

Block interference channels

Consider compound channel K consisting of two component BSC's with cross over probabilities p_s and p_w, respectively, $p_s \ll p_w$. Let us assume that in every block of $m \geq 1$ uses of channel K, random variable S determines the state of the channel according to an iid distribution over $\{s, w\}$, i.e. $Pr\{S = s\} = 1 - Pr\{S = w\} = \varphi$. Both sender and receiver know when these blocks begin and end, i.e. block synchronization is assumed. Channel K is referred to as a block interference channel in [39].

Furthermore, assume that side information is available to the receiver (after every transmission). In every block of transmissions, channel K with side information at the receiver can be viewed as a DMC with input alphabet $\{0,1\}^m$ and output alphabet $\{0,1\}^m \times \{s, w\}$ and capacity per channel use is $C = \varphi C_s + (1-\varphi)C_w$, where C_s and C_w are the capacities of the component channels, see [39, (3.7)]. Therefore, C is also the capacity of the channel with feedback in accordance with the results of [2].

Note that when $m = 1$, the resulting channel is a BIQO channel. Therefore, the SRFC method can also be applied to the block interference channel with side information at the receiver. The analysis of the transmission rate for this channels with $m > 1$ is not as straightforward as it is for the BIQO channel, nevertheless, it is expected that the SRFC schemes perform better for this channel with $m \gg 1$. This is due to the fact that the state of this channel with $m \gg 1$ does not change as fast as it does in the BIQO channel. Therefore, there is not that much of information lost in the correction subsequences as mentioned in Subsection 5.5.3 for the BIQO channel. If $p_s = p_k$ and $p_w = p_{k'}$, where p_k and $p_{k'}$ are the solutions of (2.1) for repetition parameters k and k', respectively, then the channel capacity can be closely approached for channels with sufficiently large m's.

Gilbert-Elliot channel

Let us assume that the states of a BIQO channel are determined by a stationary binary Markov process, shown in Fig. 6.12, i.e.

$$Pr\{S_i = w|S_{i-1} = s\} = 1 - Pr\{S_i = s|S_{i-1} = s\} = \varphi_w$$

and

$$Pr\{S_i = s|S_{i-1} = w\} = 1 - Pr\{S_i = w|S_{i-1} = w\} = \varphi_s,$$

then the resulting channel is a Gilbert-Elliot channel. The capacities of this channel with and without the channel side information at the receiver[6] are

[6] We assume that every state of the channel is known to the receiver immediately after receiving the corresponding output.

given in [42], see also [9] for sequential coding for the Gilbert-Elliot channel. The former is $C = \varphi C_s + (1-\varphi)C_w$, where C_s and C_w are the capacities of the component channels. According to [2], C is also the capacity of the channel with feedback.

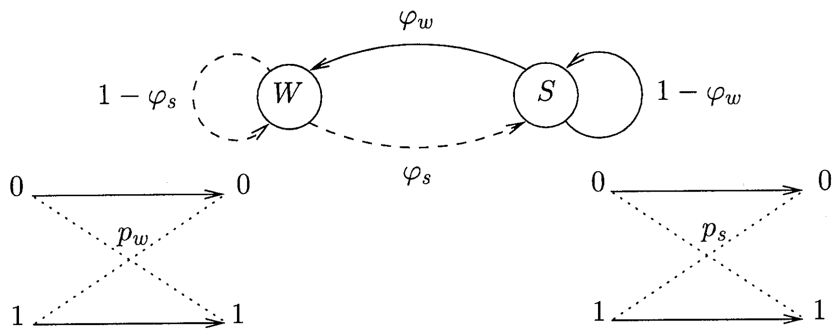


Figure 6.12: Gilbert-Elliot channel model.

If $\varphi_s = 1 - \varphi_w$, then the resulting channel is a BIQO channel. For small values of φ_s and φ_w, the channel persists on staying in its current state and it performs similar to a block interference channel with a large m. For the Gilbert-Elliot channel with channel side information at the receiver, the adaptive coding method of SRFC can be applied and the channel capacity can closely be approached when φ_s and φ_w are sufficiently small, similar to the case for block interference channels with sufficiently large m's.

6.5 Conclusions

In Chapters 5 and 6, MRFC schemes were extended to a class of soft-output DMC's which, for example, can be obtained from quantizing the output of a continuous channel. Unlike in previous MRFC schemes, the precoding of the new scheme is dependent of the channel output. However, the main advantage of the MRFC scheme, i.e. their simple decoding methods, is preserved in the new schemes. We explained block and recursive coding methods of the new scheme for a BIQO channel, which are able to decode the corresponding SRFC schemes with repetition parameters $k > k' > 2$ and $k > k' \geq 2$, respectively. Having low decoding complexity, the recursive decoding method also seems more interesting than the block decoding method because it allows schemes with $k' = 2$, which give maximum transmission rate for quantized AWGN channels. A precoding method is also outlined which is able to precode adaptively to the channel output as is required in the SRFC scheme.

Chapter 7

Applications, extensions and conclusions

At the beginning of this chapter, we shall mention some general areas where the results of information feedback coding can be applied. This is followed by outlining a few applications of the methods developed in this dissertation. Finally, some suggestions are made for the extension of the current results and for future research.

7.1 Applications

It appears that the information feedback coding which considers the presence of a delayless and error-free feedback channel is not realistically applicable to most communication channels. Nevertheless, it has drawn the attention of many researchers and information theorists. Initially, exploring such a topic is interesting from the theoretical point of view. Berlekamp in the introduction of his doctoral thesis [6] mentions that the study of (noiseless) feedback coding provides an interesting comparison with normal one-way coding. Between these two extremes, one might obtain bounds on the performance of coding schemes with partial or noisy feedback.

Furthermore, some of the results on information feedback coding can be applied to other scientific disciplines. For example, in [6] Berlekamp proposes that his results can also be used in the area of statistical design of experiments, a mathematical subject which is isomorphic to the theory of error correcting codes. This discipline deals with equalizing the anomalous effects expected in the results of a series of noisy experiments (often chemical, medical or psychological experiments, see also [91, Chapter 8] and references therein). In these cases, it is possible to modify the subsequent experiment according to the result of the previous experiments, i.e. to make use of the delayless and noiseless feedback channel. Another discipline where the result of information feedback coding can be applied is economic markets. In this view, prices in the market contain some information. For example, consider a trade session consisting of N auctions taking place at consecutive time intervals. At every auction, the price of an asset is determined by the amount of the trading in that auction and in the previous auctions. A market agent who is somehow informed about the final price, can observe the price of the asset in the current and previous intervals (collecting feedback information). Based on this information, he can adjust the amount of the asset that he offers in the next auction to maximize his overall profit. Ref. [46] explores the idea in detail, see also [45] or its overview in [91, Chapter 8].

Finally, some technical applications are proposed for information feedback schemes even with a noiseless feedback channel. Writing information into memories with known defects, e.g. into read only memories, satellite communications and army scouts are some examples of these applications, see [91, Chapter 8] for an overview. One can also think of some areas where future technology might allow us to use information feedback schemes. For example, in magnetic recording channels, the forward transmission is to write data onto the magnetic device using a writing head. To recover the data, a reading head is used, which can also serve as a noiseless feedback channel during writing process. Currently, due to the practical limitations of the writing and reading heads, it is impossible to write and immediately read the data. Improving these devices might open the door for applying information feedback schemes to this particular case. Nevertheless, having a delayed noiseless feedback channel may be feasible within current technology if the writing and the reading heads are located far enough from each other (see Subsection 7.1.2 for more detail on delayed feedback information).

In FCM, a feedback coding method introduced in Chapter 3 for AWGN channels, it is not necessary for the feedback channel to be noiseless. Therefore, one can think of many applications of the method, some of which are mentioned in the following subsection.

7.1.1 Applications of FCM

As mentioned in Chapter 3, FCM is a bandwidth efficient modulation scheme which can be used in band limited channels with high SNR's. As an alternative for TCM, FCM can be used in most communication channels where there exists an idle feedback link. For example, FCM can be employed in the modems which are used, for example, in telephone networks. Nowadays, all these point to point connections are via two-way channels [43, pp. 74] (one in each direction), where data often flows in one direction only.

There is currently a growing demand to transfer high speed digital data to households. Instead of installing optical fibers, the high speed data transmission links over distances of up to 4000 meters in digital subscriber loops and up to 100 meters in digital local area networks can be implemented on the existing Unshielded Twisted Pair (UTP) copper cables [82]. The UTP copper cables have been used to connect subscribers to central offices in the telephone networks. However, these loops are capable of operating in a much wider bandwidth and transferring a large amount of information. High bit-rate Digital Subscriber Line (HDSL) and Asymmetric Digital Subscriber Line (ADSL) technologies have been proposed for these high speed digital data transmissions. The HDSL offers a solution for bidirectional transmissions and ADSL architectures are designed to transmit video on demand signals via subscriber loops. In ADSL, the backward channel is used for control commands and, therefore, the capacity of the backward channel is less than that of the forward channel.

Various baseband and passband modulation techniques are being considered for HDSL and ADSL systems. For example, this modulation can be carried out by a single-tone modulation scheme such as QAM, where the transmission power is concentrated in one frequency band. Alternatively, a multi-tone modulation scheme, see also Subsection 7.2.4, can be used to optimize the power spectrum over more than one (disjoint) frequency band. I.e. the transmission channel is partitioned into a multitude of subchannels, each with its own associated carrier [30]. In single-tone (or multi-tone) modulation, every transmitted signal (or every transmitted signal from each subchannel) is a signal point from a 2-dimensional constellation. And because a feedback link is already available in the architecture of the HDSL and ADSL systems, there is a great potential for using the FCM method. Remember that in this way it is possible to gain about 5.5 dB in the SNR, i.e. to transmit about four times as many messages per 2-dimensional signal, with a small amount of coding effort.

7.1.2 Round trip delay

Round trip delay is an important issue to be considered in applying an information feedback scheme to a feedback network. The round trip delay, denoted by t_d, is the time needed at the transmitter to send a signal, i.e. to send a channel input, and receive the corresponding feedback signal. The propagation time of the signal over the forward and backward channels as well as the processing time at the receiver, needed for demodulation and retransmission of the channel output, account to t_d. For example, over a d meter subscriber line the wave propagation time is $t_p = \frac{d}{\nu}$, ν being the speed of wave in the twisted copper cables. In general, it is difficult to calculate ν precisely, however, we can approximate it to the speed of light. Hence, there is approximately a round trip delay of $t_d \approx 2t_p$ seconds for receiving the feedback signal (assuming that the processing time at the receiver is negligible).

If the channel bandwidth is W, then time interval between two consecutive transmissions T is almost $\frac{1}{2W}$. If $t_d < T$, then the next channel input can be determined according to the rules of the feedback coding scheme (assuming that the processing time at the transmitter is negligible). If $t_d > T$, then one can use the *interleaving* technique. For interleaving, N feedback coding schemes are used in parallel, where N is the smallest integer number such that $t_d < NT$. More clearly, every N consecutive transmissions constitute a time slot. The n^{th} symbol of every time slot corresponds to the n^{th} scheme, $n = 1, \ldots, N$. In the j^{th} time slot, $j = 1, 2, \ldots$, the j^{th} symbols of the schemes are transmitted in the order of $n = 1, \ldots, N$, and respectively after N transmissions, the transmitter receives the j^{th} feedback symbol of the n^{th} scheme. Therefore, the transmitter can accordingly determine the next transmit symbol of each scheme in the $(j+1)^{\text{st}}$ time slot.

- Using the interleaving technique increases the storage size of the decoder by a factor of N. However, the computational complexity of the decoder per symbol remains intact. Note that the storage size is very small in recursive MRFC systems, thus the amount of extra memory is still not considerable.

- As mentioned before the duration of a symbol transmission, i.e. T, is inversely proportional to the channel bandwidth W. Therefore, a way to reduce N is to increase T, i.e. to decrease the channel bandwidth. On the other hand, the bandwidth W in wideband channels is practically divided into a few smaller bandwidths to exploit the SNR characteristic of the channel at different frequencies. The resulting method, in which a channel is divided into some parallel subchannels of smaller bandwidths, is called multi-tone modulation, see also Subsection 7.2.4. Since the duration of signals in each subchannel increases in this way, the transmitter has more time for observing the feedback symbols.

The round trip delay is a design issue and very much depends on system parameters. To illustrate the point, consider the following example.

Example 7.1 In a 4000 meter subscriber line, we can approximately have $\frac{1}{t_d} \approx 37500$ signal transmissions per second per subchannel without having the round trip delay problem. For a higher signal transmission rate, the interleaving technique can be used in conjunction with FCM. For a data transmission rate of 1.5 mega bits per second and a constellation of 64 signal points, we need to interleave 7 FCM schemes in the single-tone case. In a multi-tone modulation method with more than 7 subchannels we do not need to interleave at all. □

7.2 Further extensions

Some aspects of the methods presented in this dissertation can be extended further. The following subsections provide an overview for such possible extensions, which can be pursued in future research.

7.2.1 Reducing the SNR_{fb} in FCM

In FCM we assumed that the signal points within the constellation of the feedback channel are sufficiently apart to have a feedback symbol error probability of at most 10^{-14}. Under the condition that $\mathrm{SNR}_{fb} \leq \mathrm{SNR}$, the mentioned error probability is maintained when the number of coordinate points in the constellation of the forward channel is more than 5, see also Rem. 3.4. Now the question is whether it is possible to relax this condition or, in other words, to require lower SNR_{fb}. To reduce SNR_{fb}, the minimum distance γ_{fb} must be reduced which, in turn, increases the signal error probability in the feedback channel. If this error probability is increased from 10^{-14} to, for example, 10^{-2} (i.e. γ_{fb} is decreased about 3 times), then SNR_{fb} is reduced about 9.5 dB. A signal error probability of 10^{-2} in the feedback channel is not tolerable for the FCM and, therefore, the feedback signals have to be encoded.

In [61], Schalkwijk proposed a coding strategy for Binary Duplex Channels (BDC's) which can be seen as the solution to the above problem when the constellation of the forward channel has two signal points (the same as that of the feedback channel). More precisely, a BDC is a combination of two BSC's with the same crossover probability p, one being the forward channel and the other being the backward channel, see Fig. 7.1. A MRFC scheme is used in the forward direction and every received bit (or every noise bit in the forward channel) is estimated at the transmitter by using a code on the feedback information.

Let us represent a binary sequence $b_1 b_2 b_3 \ldots$ by a formal power series $\mathbf{b}(\alpha) = b_1 + b_2\alpha + b_3\alpha^2 + \ldots$. Both transmitted and received sequences are passed through a convolutional scrambler with characteristic equation $\mathbf{c}(\alpha)$ to have $\mathbf{x}(\alpha)\mathbf{c}(\alpha)$ and $\mathbf{y}(\alpha)\mathbf{c}(\alpha)$, respectively. The scrambler with $\mathbf{c}(\alpha) =$

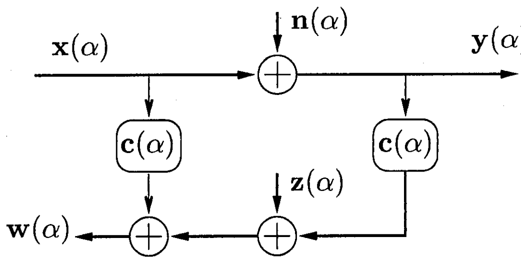


Figure 7.1: Schalkwijk's BDC.

$c_1 + c_2\alpha + \ldots + c_{\nu+1}\alpha^\nu$, where $c_i \in \{0,1\}$, $i = 1, \ldots (\nu+1)$, can be seen as a ν stage shift register. The input of the i^{th} register, $i = 1, \ldots, \nu$, and the output of the ν^{th} register are added modulo 2 if $c_i = 1$, $i = 1, \ldots, (\nu+1)$, respectively. The transmitter adds (modulo 2) $\mathbf{x}(\alpha)\mathbf{c}(\alpha)$ to the feedback sequence to have

$$\begin{aligned}\mathbf{w}(\alpha) &= \mathbf{x}(\alpha)\mathbf{c}(\alpha) + \mathbf{z}(\alpha) + (\mathbf{x}(\alpha) + \mathbf{n}(\alpha))\,\mathbf{c}(\alpha) \\ &= \mathbf{z}(\alpha) + \mathbf{n}(\alpha)\mathbf{c}(\alpha),\end{aligned}$$

where a sequence is obtained independently of the transmitted sequence. From $\mathbf{w}(\alpha)$ one can obtain the estimate of $\alpha^{(D-1)}\mathbf{n}(\alpha)$ using a 2^ν-state rate-$\frac{1}{2}$ systematic convolutional decoder [61]. A coding delay of about $N_s = 5(\nu+1)$ transmissions is necessary to estimate every noise signal in the forward direction. Therefore, an interleaving of N_s independent MRFC schemes is needed in this method.

We can also use the same idea in FCM and reduce the SNR_{fb}. As a result, the complexity of the encoder increases due to the estimation procedure, however, the decoding complexity increases slightly, mainly as a result of increasing memory size. We can say that complexity is transfered to the transmitter side by this method. Note that the round trip delay adds extra delay on top of the estimation delay. In this way, there is a possibility for an asymmetric dialog (two-way communication), where in the feedback direction the rate of the true information, i.e. not the feedback information related to the received symbols, is lower than that in the forward direction.

7.2.2 Phase shifting in FCM

A two-dimensional transmitted signal might be demodulated with a phase offset φ. This would cause problems for the soft-decision decoder of the

TCM scheme which operates on the signal sequence $a_i \exp\varphi$ instead of a_i, $i = 1, 2, \ldots$, where a_i is a complex-valued number. The phase offset could be caused, for example, by disturbances of the carrier phase of the received signal, which the phase-tracing scheme of the receiver cannot track instantly. In TCM, rotational invariance codes are designed for a group of phase offsets that leave the two-dimensional constellation invariant. For example, in multilevel QAM the group consists of phase offsets of 0° , 90°, 180° and 270°. Rotational invariance results when rotated sequences are also valid code sequences. This yields transparency of the transmitted information under such phase offsets.

In FCM, the feedback information can be closely investigated at the transmitter to detect any phase offset at the receiver. Thereafter, appropriate measures can be taken to adapt the following transmissions to the current phase of the receiver (as long as the phase offset does not occur in the backward direction). For simultaneous phase offsets in both directions some other measures must be devised (such as regularly transmitting some synchronization data).

7.2.3 Burst errors in FCM

In most practical channels with memory, errors tend to occur in bursts. In block MRFC coding schemes, on the other hand, the positions of errors in a block are irrelevant and the block is decoded correctly as long as the number of errors remains below a certain threshold, see Theorem 2.1. Therefore, block FCM is expected to operate well in channels with burst errors, where the length of blocks are long enough to accommodate a few runs of errors and correct them all.

7.2.4 Multi-band modulation

A general linear Gaussian channel is a waveform channel with input signal $x(t)$, a channel impulse response $h(t)$ and additive Gaussian noise $n(t)$. The received signal is

$$y(t) = x(t) * h(t) + n(t),$$

where $*$ denotes convolution. The frequency response of $h(t)$ is denoted by $H(f)$ and the one-sided power spectral density (psd) of the noise is denoted by $N(f)$. The transmitted signal $x(t)$ is subjected to power constraint

$$\int_{f\geq 0} P_x(f)df \leq P, \tag{7.1}$$

where $P_x(f)$ is one-sided psd of $x(t)$. The optimum $P_x(f)$ maximizes the mutual information between channel input and output subject to the power constraint (7.1) and $P_x(f) \geq 0$. The water filling argument [25] shows that the optimum input psd is

$$P_x^o(f) = \begin{cases} K - \frac{N(f)}{|H(f)|^2}, & f \in B; \\ 0, & \text{otherwise}, \end{cases}$$

where $B = \{f : P_x^o(f) > 0\}$ is called the capacity achieving band and K is a constant chosen such that (7.1) is satisfied with equality, see Fig. 7.2. The capacity of the channel is

$$C = \int_B \log\left(1 + \frac{P_x^o(f)|H(f)|^2}{N(f)}\right) df,$$

where $x(t)$ is a Gaussian process with psd $P_x^o(f)$, see [26].

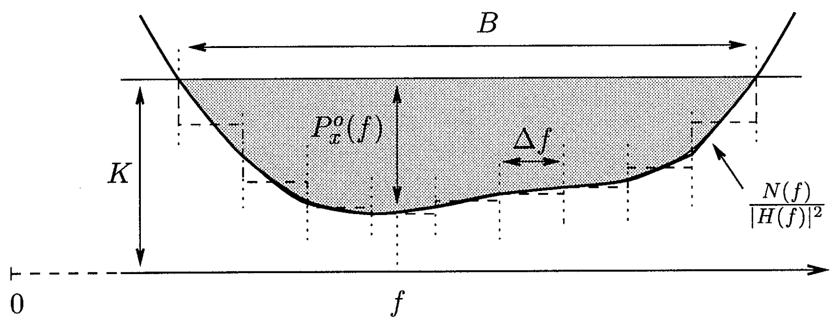


Figure 7.2: An illustration of the water filling approach.

One way of approaching the capacity of this channel can be deducted from the water filling argument. The capacity achieving band B can be divided into disjoint subbands of small enough width Δf over which $P_x^o(f)$ is constant. Each subchannel can be treated as an ideal AWGN channel [26] with bandwidth Δf, where the allocated transmit power to each subchannel centered at f is approximately $P_x^o(f)\Delta f$ and the corresponding subband capacity is

$$C(f) = \Delta f \log(1 + \frac{P_x^o(f)|H(f)|^2}{N(f)}).$$

Multicarrier transmission methods can be used for the resulting channel. Usually modulation and demodulation are carried out in the frequency domain and the corresponding systems are referred to as Orthogonal Frequency Devision Multiplexing (OFDM) or discrete multi-tone systems. This is a form of frequency devision multiplexing in which symbols are transmitted in individual subchannels using modulation methods such as QAM. Current application of the method include digital audio broadcasting and ADSL. Another multicarrier transmission method uses Wavelet transform and the resulting method became known as discrete Wavelet multi-tone modulation.

OFDM based systems can potentially be a place to apply the FCM scheme and to obtain a coding gain of about 5.5 dB in each subchannel. Having subchannels eases the round trip delay restriction where a delay of $\frac{1}{\Delta f}$ can be long enough to have the feedback information available just before the next transmission. In xDSL (ADSL, HDSL ...) technologies, the subchannels are usually high SNR channels and feedback information can constitute a small part of the backward transmissions, which can allow a two-way flow of information.

7.2.5 Low-rate feedback schemes for DMC's

In information feedback coding, every received output symbol is fed back to the transmitter via a noiseless (and maybe delayless) channel. In FCM, however, only the partition to which the received symbol belongs is fed back to the transmitter. The *low-rate feedback* term is used here to refer to such coding methods where the amount of feedback information is lower than that in the conventional information feedback schemes for the same forward channel.

The idea of low-rate feedback coding used in the FCM was first employed in DMC's, see [75, 77] and for a more comprehensive coverage see [74]. Consider a BSC where for each frame of f forward transmissions one is allowed to complete a group of g (with $g \ll f$) noiseless feedback transmissions. Using set partitioning in the frame of f bits, a forward code affects the rather simple intra-partition decoding, while inter-partition decoding is accomplished by a MRFC scheme. In FCM the signal points within partitions are transmitted without coding, however, in these low-rate feedback coding schemes the signal points in partitions are encoded, i.e. intra-partition coding, by a simple convolutional code.

In [75], a scheme with $(f, g) = (2, 1)$ is designed using a 2-state convolutional code for intra-partition coding. The transmission rate of the resulting scheme is well above 0.5. Similarly, in [75] schemes up to $(f, g) = (9, 1)$ are designed where at most a 4-state convolutional code is used. As g decreases relative to f, only a slight degradation in rate performance is observed. In the forward direction, a total transmission rate of 0.44 bits per channel bit is obtained by the scheme with $(f, g) = (9, 1)$. The error performance of all these schemes is comparable to that of a 64-state rate-0.5 convolutional code. The decoding complexity of the resulting scheme, however, is much lower than that of a one-way code with similar performance. The extensions of these methods can be suitable for use in channels with low SNR's, where low-rate one-way codes are used.

7.3 Open problems

Some open problems, arisen or encountered in the course of this research, are reviewed in this section to be considered for future research.

7.3.1 Low-rate noisy feedback coding

In information feedback, it is assumed that for every transmission in the forward direction there is a noiseless (and delayless) transmission in the backward direction to feed back the corresponding received symbol. As seen in Chapter 1, these schemes are generally simple and outperform one-way codes. For most applications, however, having a noiseless feedback channel is not feasible.

In this dissertation we introduced a feedback scheme working with a noisy feedback channel. The assumption here is that a (noisy) feedback channel is at transmitter's disposal. To deal with the noise in the feedback channel, the rate of feedback information is reduced by feeding back only partial information regarding every received symbol. The *low-rate noiseless* feedback information is transmitted at full rate over the noisy feedback channel, i.e. some redundancy (or power) is added to the (low amount of) feedback information and the result is transmitted over the feedback channel. This scheme still reduces decoding complexity considerably with respect to that of one-way coding schemes.

Another step forward in this direction would be to consider **low-rate noisy** feedback schemes where not only the amount of feedback information is small but also it is transmitted over a noisy feedback channel, i.e. where adding redundancy or increasing transmission power is not allowed. Assume that there is a single connection between two points, which can be used in either direction by time sharing. For transmission in one direction, a transmission block can be divided into two subblocks: a forward subblock and a backward subblock. The object is to design feedback schemes that transmit information in the forward subblock and then feed back the partial (low-rate) feedback information in the backward subblock. Such a scheme can be an alternative solution to be used in most cases where one-way codes are in use nowadays, see also [60]. It would be an interesting topic for future research to design such low-rate noisy feedback schemes and/or to compare their performance and complexity to those of one-way codes.

7.3.2 A capacity achieving SRFC scheme

In Subsection 5.5.3 we explained that a SRFC scheme cannot achieve the capacity of a BIQO channel because the status information of correction symbols is ignored and only the number of correction symbols, i.e. the type of the received correction symbols, is taken into account.

Another interesting topic for future research is to investigate whether it is possible for any MRFC based scheme to have a transmission rate that coincides with the capacity on a soft-output DMC. We would like to draw the attention of the interested reader to Veugen's conditions that must hold when the capacity of a DMC is achieved by a MRFC scheme. These necessary and sufficient conditions are given by (2.21) and we rewrite them as follows.

$$\frac{p_{xy}}{\pi_y}\left(\frac{p_{xx}}{\pi_x}\right)^{(k_{xy}-1)} = 1, \qquad \text{where } x \in \mathcal{X}, y \in \mathcal{Y}.$$

Let us see whether these conditions can hold for a BIQO channel if a hypothetical repetition scheme takes into account the status of the received correction symbols. In this view, a strong error such as $0 \to 1_s$ is compensated[1] by $(k-i)$ strong zeroes and k'_i weak zeroes, where k'_i is an integer chosen appropriately and $i = 1, \ldots, k$. In this scenario, the conditions analogous to (2.21) turn out to be

$$\frac{p}{\frac{p+q}{2}}\left(\frac{q'}{\frac{p'+q'}{2}}\right)^{k'_i}\left(\frac{q}{\frac{p+q}{2}}\right)^{(k-i)} = 1, \qquad i = 1, \ldots, k \text{ and } k'_i \geq 1.$$

A similar set of conditions can also be written for a weak error. To satisfy these conditions the effect of a strong correction symbol must be the same as those of k'' weak correction symbols, i.e.

$$\left(\frac{q'}{\frac{p'+q'}{2}}\right)^{k''} = \frac{q}{\frac{p+q}{2}}, \qquad k'' > 1.$$

This assumption is impossible in practice, because there is no guarantee that if a strong reception does not occur in a correction subsequence, then k'' weak

[1]For example in MRFC schemes for BSC's, a $0 \to 1$ error is compensated after receiving subsequence $10^{(k-1)}$. The last 0 corrects the error.

receptions occur instead. Therefore, it seems impossible for such a repetition scheme to take into account every chip of information embedded in the outputs of a soft-output channel and at the same time to fulfil conditions analogous to (2.21).

7.3.3 Capacity of the precoder

The FSM of SRFC is not a deterministic constrained channel in the sense that Shannon introduced in [67]. I.e. it is not possible to enumerate the number of constrained sequences before transmission because the status sequence is not known beforehand. Therefore, the entropy rate of the Markov source modelling the FSM is used to calculate the exponential growth rate of the number of such constrained sequences.

In Appendix A, it is shown that the maximum entropy rate of the Markov source modelling the FSM of MRFC schemes is equal to the capacity of the corresponding constrained channel. Analogously, we maximized the entropy rate of the Markov source modelling the FSM of the SRFC schemes given that $Pr\{S_j = s\} = \theta$, see (5.11). It is interesting to show that this *conditionally* maximum entropy rate is also the capacity of such constrained channels. An approach would be to use the concept of the *types* for the Markov sources that are not operating at the maximum entropy rate of the corresponding FSM, see [3]. There are only a polynomial number of Markov types and therefore of all the Markov type classes that satisfy the state constraints, at least one of them has the same exponent as the total number of sequences that satisfy the state constraints.

7.3.4 Extension to channels with ISI

A simple model of a channel with InterSymbol Interference (ISI) is shown in Fig. 7.3. The state entered at any instant is the same as the input at that instant. In state 0, the crossover probability of 0 to 1 is much less than that of 1 to 0. This is the other way around in state 1. In other words, the probability of an erroneous transmission in each state is higher if the input bit is different from the previous input bit. Since in MRFC schemes the transmitted correction bits are always the same, it seems promising to investigate the performance of these schemes on ISI channels.

7.4 Concluding remarks

Feedback channels potentially exist in many communication systems. Using feedback information can, at least, reduce the (de-) coding complexity. Assum-

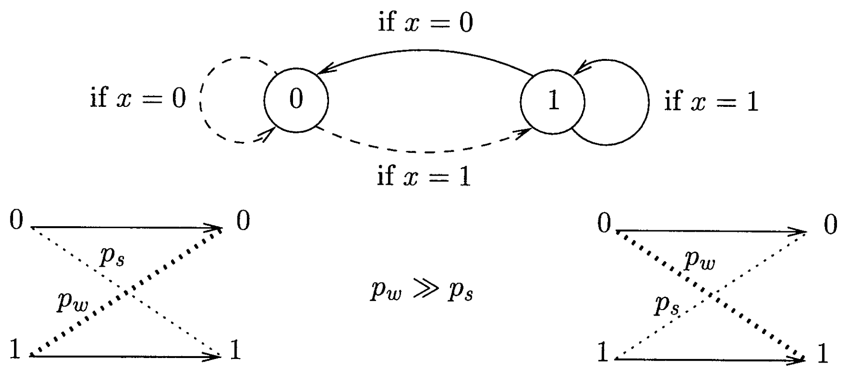


Figure 7.3: A simple model of a channel with ISI.

ing feedback channels to be noiseless is not realistic in most communication systems with feedback and it has been a major obstacle on the way of using feedback coding schemes.

In this research, a step is taken towards dealing with the noise in the feedback channel. As a result the FCM is devised for communication networks where the forward and backward channels are characterized as bandlimited AWGN channels with high SNR's. Even though this class of channels covers many practical cases, the ideas involved in FCM can be applied to other channel models as well. It is shown that a 5.5 dB improvement in the SNR is feasible by FCM, which is in the range of the effective coding gain of the most complex TCM with a 256 or 512 state trellis code. The decoding complexity of the resulting scheme is considerably less than that of the conventional TCM scheme. In this contribution it is also shown how to use signal shaping techniques in the FCM case. With a very simple signal shaping code an extra shaping gain of 1 dB can be obtained on top of the 5.5 dB coding gain. Moreover, a simple precoding method is proposed for the precoding of MRFC schemes by using an arithmetic coding algorithm. The precoding rate of the method, investigated by running simulations, is very close to the optimum value. MRFC schemes are the backbone of the designed methods. Two block retransmission strategies are devised to improve the performance of block MRFC. A MRFC scheme is also designed and evaluated for soft-output DMC's.

Reducing the complexity is an important issue of concern in designing versatile communication systems. Demand for such systems is growing in the era of the information super highway, where in some areas such as multi-media applications the number of consumers is very large. Therefore, we believe that feedback coding will become an important design option to be considered in implementing such complex systems and it cannot simply be ignored by skeptic traditionalists.

Appendix A

Precoding rate

In this appendix we outline two general approaches to calculate the capacity of a constrained channel corresponding to the precoder of a MRFC scheme[1]. A precoder can be seen as a state machine consisting of finite number of states or as a Finite State Machine (FSM) for short. At each state, the FSM produces a symbol (a number of symbols) from $\mathcal{X}$ according to the precoding rules. Given repetition parameters k_{xy} $(x, y \in \mathcal{X})$, $yx^{k_{xy}}$ are forbidden subsequences of the precoder, which determine the precoding rules. We adopt a FSM in which every two states are connected by at most one single edge. In a transition from one state to another, the FSM produces one symbol corresponding to that transition, see the FSM on the left-hand side of Fig. 2.4.

Assume that the FSM consists of Ψ states denoted by $Q_1, Q_2, \dots, Q_\Psi$. The number of states depends on $|\mathcal{X}|$, i.e. the size of the alphabet $\mathcal{X}$, and the values of repetition parameter k_{xy}'s. At every state, the FSM can produce one symbol from $\mathcal{X}$ as long as the produced symbol does not cause any constrained or forbidden subsequence. For example, if reaching state Q by the FSM means that the last symbols of the precoded sequence are $\dots yxx$, then the precoder can produce symbol x if $k_{xy} - 1 > 2$ and all $z \in \mathcal{X} - \{x\}$ if $k_{zx} > 1$. Each produced symbol changes the state of the FSM to a different state, see Fig. A.1. In summary, all constrained sequences that end up at given state Q must be restricted by identical constraints from *producing the next symbol* point of view.

Remark A.1 For a sequence ended up at state Q, the state must identify

- the last symbol of the sequence,

[1]Note that maximizing the precoding rate does not necessarily maximizes the transmission rate of the MRFC schemes for asymmetric channels, see [91, pp. 67]. In MRFC schemes for strict-sense and wide-sense symmetric channels, however, maximizing the precoding rate leads to maximizing the transmission rate.

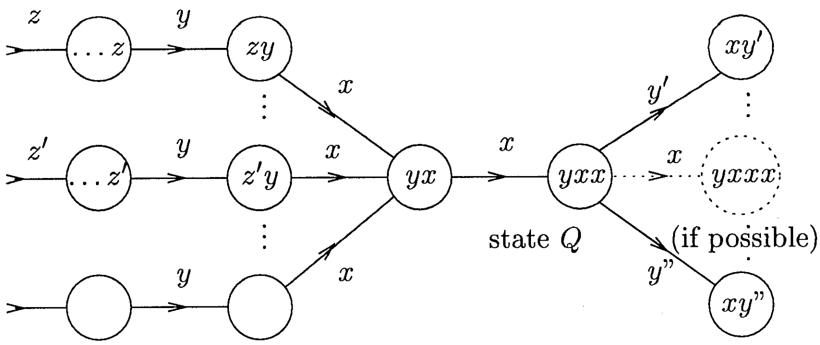


Figure A.1: The state structure of the FSM

- the number of identical symbols at the end of the sequence,
- the previous symbol (non identical) to the last symbol.

The first factor is essential when the next symbol of the sequence changes with respect to the last symbol of the sequence, and next two factors are essential when the next symbol of the sequence does not change.

For example, sequences $\ldots z, yxx$ and $\ldots z', yxx$ must end up in the same state Q, see Fig. A.1 because as long as preventing the occurrence of a forbidden subsequence is concerned, producing the next symbol is independent from z and z' for the sequences. Therefore, state Q represents all sequences which end up in yxx. This reflects the Markovian nature of the FSM of the precoder. The FSM can be described by a $\Psi \times \Psi$ connection matrix $\mathbf{D_c} = [d_{ij}]$, for which the entries are

$$d_{ij} = \begin{cases} 1 & \text{if there is a connection from state } Q_i \text{ to state } Q_j, \\ 0 & \text{if there is no connection from state } Q_i \text{ to state } Q_j, \end{cases}$$

for $i, j = 1, \ldots, \Psi$.

Example A.1 Consider the FSM depicted on the left-hand side of Fig. 2.4, where the states are indexed in the order of $Q_0^1, Q_0^2, Q_0^3, Q_0^4, Q_1^1, Q_1^2, Q_1^3$ and Q_1^4. The connection matrix is

$$\mathbf{D_c} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad \square$$

For L outputs of such channel, one can have $M(L)$ distinct constrained output sequences. Shannon in [67] defined the capacity of such a noiseless constrained channel as

$$C = \lim_{L \to \infty} \frac{\log M(L)}{L}, \tag{A.1}$$

or the net growth rate of the logarithm of the number of distinct constrained sequences with the number of constrained symbols. From Def. 1.11, C can be seen as the entropy per output letter of the constrained channel (with the assumption that all sequences are equiprobable). To compute C, we outline two procedures and show that their results are the same.

The first way to compute C is to enumerate the number of constrained sequences of length L. Shannon in his original paper [67] used an enumeration method for a more general case. Shannon's method enumerates the number of constrained sequences ending up in a state. Here we outline the enumeration method presented in [31] where the number of constrained sequences starting from a state is enumerated. Let $M_i(L)$ be the number of distinct constrained sequences of length L, which are emerging from state Q_i, $i = 1 \ldots \Psi$. Since every edge emerging from state Q_i corresponds to one precoded symbol, we have

$$M_i(L) = \sum_{j=1}^{\Psi} d_{ij} M_j(L-1). \tag{A.2}$$

Assuming a general solution $M_i(L) = a_i \lambda^L$, where a_i is a constant independent of L, $i = 1 \ldots \Psi$. From (A.2) we have

$$a_i \lambda = \sum_{j=1}^{\Psi} d_{ij} a_j. \tag{A.3}$$

Writing the corresponding equations for $i = 1 \ldots \Psi$, the following set of equations is obtained.

$$\lambda \mathbf{a} = \mathbf{D_c a}, \tag{A.4}$$

where vector $\mathbf{a}^T = (a_1, a_2, \ldots, a_\Psi)^2$. The nontrivial solution of (A.4) is obtained when

$$\det(\mathbf{D_c} - \lambda \mathbf{I}) = 0, \tag{A.5}$$

where $\mathbf{I}$ is the $\Psi \times \Psi$ identity matrix. Let λ_0 be the largest solution of (A.5) or the largest eigenvalue of matrix $\mathbf{D_c}$. The capacity can be written from (A.1) and (A.2) as

$$C = \lim_{L \to \infty} \frac{\log(M_i(L))}{L} = \lim_{L \to \infty} \frac{\log(a_i \lambda_0^L)}{L} = \log \lambda_0.$$

For the second approach, the FSM diagram can also be seen as a Markov source which consists of states $Q_1, Q_1, \ldots, Q_\Psi$. Let random variable S_l denote the state of the Markov source at instant $l = 1, 2, \ldots$. Moreover, assume that q_{ij} represents the transition probability from state i to state j, i.e.

$$q_{ij} = Pr\{S_l = Q_j | S_{l-1} = Q_i, \mathbf{S}^{(l-2)} = \mathbf{s}^{(l-2)}\} = Pr\{S_l = Q_j | S_{l-1} = Q_i\}.$$

The $\Psi \times \Psi$ matrix $\mathbf{Q}$ with entries q_{ij} is called the *state transition matrix.* Let q_j be the steady (stationary) state probability of being in state Q_j, $j = 1 \ldots \Psi$. The state distribution vector $\mathbf{q}^T = (q_1, q_2, \ldots, q_\Psi)$ is the solution of

$$\mathbf{Q}^T \mathbf{q} = \mathbf{q}.$$

For most practical Markov sources (including ours)[3] there is only one limiting state distribution vector. In other words, the steady state distribution of state probabilities converges to $\mathbf{q}$ regardless of initial distribution. During every transition from one state to another, the Markov source produces an output symbol, or in other words, a precoded symbol. For example, the solid and dotted edges in Fig. 2.4 correspond to producing 1 and 0, respectively. Therefore, the Markov source in state Q_i is like a memoryless source with entropy

$$H_i = -\sum_{j=1}^{\Psi} q_{ij} \log q_{ij}.$$

[2]Upper-script T denotes the transpose of a matrix or a vector.

[3]The state set of these Markov sources is irreducible and ergodic, see [25, pp. 65].

For a unifilar Markov source the entropy rate is obtained from

$$H_M = \sum_{i=1}^{\Psi} q_i H_i = -\sum_{i=1}^{\Psi}\sum_{j=1}^{\Psi} q_i \; q_{ij} \log q_{ij}. \tag{A.6}$$

Let sequence $s_1 s_2 \ldots$ and sequence $x_1 x_2 \ldots$ be particular state and output sequences of the Markov source, respectively. At instant l, the source which is in state s_l, produces output x_l, $l = 1, 2, \ldots$.

Definition A.1 (Unifilar Markov source) *In a Markov source, if the current state s_l and the current output x_l uniquely determines the next state s_{l+1}, then the Markov source is unifilar.*

Fig. A.2 schematically shows the selection mechanism of the next state of a unifilar Markov source by deterministic function $g(\cdot, \cdot)$. In other words, the state sequence of a Markov source can be uniquely determined by the output sequence of the source and its initial state. It is easy to see that the Markov source corresponding to the precoder of a MRFC scheme is unifilar. The second way to calculate the capacity of a constraint channel is by maximizing the entropy rate of the Markov source corresponding to the FSM with respect to state transition probabilities. A Markov source with a set of transition probabilities that maximizes its entropy rate is called *maxentropic* [31].

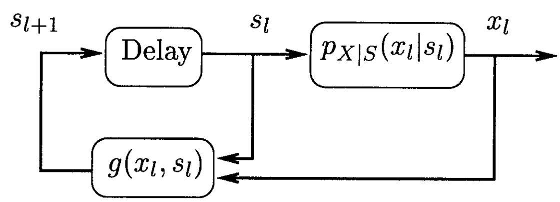


Figure A.2: Determination of the next state in a unifilar Markov source.

Theorem A.1 *Given the connection matrix $\mathbf{D_c}$ of the FSM of MRFC schemes and its largest eigenvalue λ_0, the maximum entropy rate of the corresponding unifilar Markov source satisfies*

$$\max_{q_{ij}} H_M = \log \lambda_0 = C.$$

Proof : The proof of [31] is presented here with slight changes.

1. There isn't any set of state transition probabilities, or stochastic matrix $\mathbf{Q}$, for which $H_M > \log \lambda_0$. Because then $C = \log \lambda_0$ cannot be the capacity of the channel.

2. Now we look for a set of state transition probabilities which yields $H_M = \log \lambda_0$ and, therefore, maximizes the entropy rate due to point 1. Using λ_0 in (A.3), we can write the i^{th} characteristic equation as

$$\sum_{j=1}^{\Psi} \frac{d_{ij} a_j}{\lambda_0 a_i} = 1$$

We claim that choosing $q_{ij} = \frac{d_{ij} a_j}{\lambda_0 a_i}$ for $i, j = 1 \ldots \Psi$ yields $H_M = \log \lambda_0$. Note that the resulting state transition matrix is stochastic because the components of the eigenvector $\mathbf{a}$ are positive, see [88, pp. 30], and hence q_{ij}'s are nonnegative and $\sum_{j=1}^{\Psi} q_{ij} = 1$. From (A.6), the entropy rate can be written as

$$\begin{aligned}
\sum_{i=1}^{\Psi} q_i H_i &= -\sum_{i=1}^{\Psi} q_i \sum_{j=1}^{\Psi} q_{ij} \log q_{ij} \\
&\overset{(\alpha)}{=} -\sum_{i=1}^{\Psi} \sum_{j=1}^{\Psi} q_i q_{ij} (\log a_j - \log a_i - \log \lambda_0) \\
&= -\sum_{j=1}^{\Psi} \sum_{i=1}^{\Psi} q_i q_{ij} \log a_j + \sum_{i=1}^{\Psi} \sum_{j=1}^{\Psi} q_i q_{ij} \log a_i \\
&\quad + \log \lambda_0 \sum_{i=1}^{\Psi} \sum_{j=1}^{\Psi} q_i q_{ij} \\
&= -\sum_{j=1}^{\Psi} q_j \log a_j + \sum_{i=1}^{\Psi} q_i \log a_i + \log \lambda_0 \\
&= \log \lambda_0
\end{aligned}$$

Note that in (α) term d_{ij} is not written because d_{ij} is either 0 or 1. When $d_{ij} = 0$, the corresponding $q_{ij} = 0$ which disappears in the definition of the entropy by the convention that $0 \log 0 = 0$.

Q.E.D.

Appendix B

Rate of discrete-input AWGN channels

In a vector AWGN channel, when the transmitted vector $\mathbf{x}^D = (x_1, x_2, \dots, x_D)$ can be any signal point in the D-dimensional Euclidean space $\mathbb{R}^D$, then capacity is achieved with independent and normally distributed input coordinates having mean value 0 and variance $\sigma^2 = \frac{E}{D}$, i.e. $X_d \sim N(0, \frac{E}{D})$ for $d = 1, \dots, D$, where E is the average transmitted energy per signal. Now we are interested in finding the capacity (more precisely, average mutual information) of AWGN channels in which X_d is a discrete random variable that can independently take one of the L equiprobable values from

$$\mathcal{X}' = \{\pm\frac{\gamma}{2}, \pm\frac{3\gamma}{2}, \dots, \pm\frac{(L-1)\gamma}{2}\} \qquad \text{(L even)},$$

where γ is the minimum Euclidean distance between two signal coordinate points. Therefore, the set of input symbols (signal constellation) of the vector channel is $\mathcal{X} = \mathcal{X}'^D$, e.g. see Fig. 3.1. Extension of the well-known formula for the capacity of a DMC, (1.3) and (1.5), to the case of a continuous-valued output channel [86] yields

$$C = \max_{p_\mathbf{X}} \sum_{j=1}^{|\mathcal{X}|} p_\mathbf{X}(\mathbf{x}_j^D) \int f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D) \log\left(\frac{f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D)}{\sum_{k=1}^{|\mathcal{X}|} p_\mathbf{X}(\mathbf{x}_k^D) f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_k^D)}\right) d\mathbf{y}^D, \tag{B.1}$$

where $p_\mathbf{X}(\mathbf{x}_j^D)$, $j = 1, \dots, |\mathcal{X}|$, denotes the a-priori probability distribution associated with input symbol $\mathbf{x}_j^D$ and

$$\begin{aligned} f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D) &\stackrel{(\alpha_1)}{=} \prod_{d=1}^{D} f_{Y|X}(y_d|x_{jd}) \\ &\stackrel{(\alpha_2)}{=} \frac{1}{(\sqrt{2\pi\frac{N_o}{2}})^D} e^{\sum_{d=1}^{D} -\frac{(y_d-x_{jd})^2}{N_o}} \\ &= \frac{1}{(\pi N_o)^{\frac{D}{2}}} e^{-\frac{|\mathbf{y}^D-\mathbf{x}_j^D|^2}{N_o}} \\ &= P_N(\mathbf{y}^D-\mathbf{x}_j^D) = P_N(\mathbf{n}^D), \end{aligned} \tag{B.2}$$

in which (α_1) follows from the fact that the D parallel subchannels are independent. Since random variable $Y_d = X_d + N_d$ and $N_d \sim N(0, \frac{N_o}{2})$, for the d^{th} coordinate we have

$$f_{Y|X}(y_d|x_{jd}) = N(x_{jd}, \frac{N_o}{2}),$$

which explains (α_2). In vector representation, $\mathbf{Y}^D = \mathbf{X}^D + \mathbf{N}^D$, therefore

$$f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D) = N(\mathbf{x}_j^D, \Delta),$$

where $\Delta = \frac{N_o}{2}\mathbf{I}$ is the covariance matrix of the AWGN vector and $\mathbf{I}$ is the $D \times D$ identity matrix. With the further assumption that the signal points at the input of the modulator are equiprobable, i.e.

$$p_{\mathbf{X}}(\mathbf{x}_j^D) = \frac{1}{|\mathcal{X}|}, \qquad j = 1, \ldots, |\mathcal{X}|,$$

the maximization over the a-priori probability distribution of signal points can be dropped. Therefore, equation (B.1) can be written in the form

$$\begin{aligned}
C' &= \frac{1}{|\mathcal{X}|}\sum_{j=1}^{|\mathcal{X}|}\int f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D)\log(\frac{f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D)}{\frac{1}{|\mathcal{X}|}\sum_{k=1}^{|\mathcal{X}|} f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_k^D)})d\mathbf{y}^D \\
&= \log|\mathcal{X}| + \frac{1}{|\mathcal{X}|}\sum_{j=1}^{|\mathcal{X}|}\int f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D)\log(\frac{f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_j^D)}{\sum_{k=1}^{|\mathcal{X}|} f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}^D|\mathbf{x}_k^D)})d\mathbf{y}^D \\
&\overset{(\alpha)}{=} \log|\mathcal{X}| + \frac{1}{|\mathcal{X}|}\sum_{j=1}^{|\mathcal{X}|}\int P_N(\mathbf{y}^D-\mathbf{x}_j^D)\log(\frac{P_N(\mathbf{y}^D-\mathbf{x}_j^D)}{\sum_{k=1}^{|\mathcal{X}|} P_N(\mathbf{y}^D-\mathbf{x}_k^D)})d\mathbf{y}^D \\
&= \log|\mathcal{X}| + \frac{1}{|\mathcal{X}|}\sum_{j=1}^{|\mathcal{X}|}\int P_N(\mathbf{n}^D)\log(\frac{P_N(\mathbf{n}^D)}{\sum_{k=1}^{|\mathcal{X}|} P_N(\mathbf{n}^D+\mathbf{x}_j^D-\mathbf{x}_k^D)})d\mathbf{n}^D \\
&\overset{(\beta)}{=} \log|\mathcal{X}| - \frac{1}{|\mathcal{X}|}\sum_{j=1}^{|\mathcal{X}|}\int P_N(\mathbf{n}^D)\log(\sum_{k=1}^{|\mathcal{X}|} e^{-\frac{|\mathbf{n}^D+\mathbf{x}_j^D-\mathbf{x}_k^D|^2-|\mathbf{n}^D|^2}{N_o}})d\mathbf{n}^D \\
&= \log|\mathcal{X}| - \frac{1}{|\mathcal{X}|}\sum_{j=1}^{|\mathcal{X}|}\overline{\log(\sum_{k=1}^{|\mathcal{X}|} e^{-\frac{|\mathbf{N}^D+\mathbf{x}_j^D-\mathbf{x}_k^D|^2-|\mathbf{N}^D|^2}{N_o}})}. \qquad \text{(B.3)}
\end{aligned}$$

In (α) we used the notation of (B.2) and in (β) we have integration replaced by expectation over the normally distributed vector of

$$\mathbf{N}^D = \mathbf{Y}^D - \mathbf{x}_j^D \sim N(\mathbf{0}^D, \Delta).$$

Using a Gaussian random number generator, C' can be evaluated from (B.3) by the following procedure. For each signal vector of $\mathbf{x}_j^D$ produce n_{try} times normally distributed vector $\mathbf{n}^D$, such that its d^{th} coordinate $N_d \sim N(0, \frac{N_o}{2})$, $d = 1 \dots D$. For each produced vector $\mathbf{n}^D$ calculate

$$\log(\sum_{k=1}^{|\mathcal{X}|} e^{-\frac{|\mathbf{n}^D+\mathbf{x}_j^D-\mathbf{x}_k^D|^2-|\mathbf{n}^D|^2}{N_o}}).$$

Then take the average of these amounts, by adding them up and dividing by n_{try}, in order to use in (B.3) for the j^{th} term.

Appendix C

Proofs to Theorems 6.2 and 6.3

C.1 Proof to Theorem 6.2

The proof of Theorem 6.2 closely follows those of [91, Theorems 2.1 and 2.2] for previously existing MRFC schemes. To show that uncompleted correction subsequences in the tail part do not cause error propagation to the precoded data part, we first assume that the length of the precoded sequence is 1.

Lemma C.1 *Let $L = 1$. If $k'e_w + ke_s < N$ and $k \geq k' > 2$, then we have $\hat{v}_1 = v_1$ and $\hat{s}_1 = s_1$ when using the tail appending method of the SRFC scheme for the BIQO channel.*

Proof : Let $\bar{a}$ denote the complement of bit $a \in \{0, 1\}$, where $\bar{0} = 1$ and $\bar{1} = 0$. Let $e_{tot} = e_w + e_s$ be the total number of channel errors in the block of N transmissions. If $e_{tot} = 0$ then no error has occurred and, consequently, there is no forbidden subsequence in $\mathbf{r}^N$. Therefore, $\hat{v}_1 = v_1$ and $\hat{s}_1 = s_1$.

Now assume that $e_{tot} > 0$ errors have occurred (e_w weak errors and e_s strong errors). We use the following argument on the number of forbidden subsequences in the received block.

1. There are no forbidden subsequences in the received block. In this case, there must not be more than $k' - 1$ and $k - 1$ correct receptions after every weak and strong error, respectively. Then the first bit is received correctly. If not, an error has to occur at the first transmission and the maximum resulting block length equals $N_{max} = e_w + (k' - 1)e_w + e_s + (k - 1)e_s = k'e_w + ke_s$. This value of N_{max} violates the assumption that $k'e_w + ke_s < N$. Hence, the first bit is received correctly and since there

	Removed symbols				
$\dots\ j-1$	j	$j+1$	$j+2$	$j+k'-1$	$j+k'\ \dots$
$\dots\ t_{j-1}$ $\dots\ (x_y)_{j-1}$	a $\bar{a}_w$	a a_-	$a\ \dots$ $a_-\ \dots$	a a_-	$t_{j+k'}\ \dots$ $a\ \dots$
$\dots\ t_{j-1}$ $\dots\ (x_y)_{j-1}$	$\bar{a}$ $\bar{a}_w$	a a_-	$\bar{a}\ \dots$ $a_-\ \dots$	$\bar{a}$ a_-	$\bar{a}\ \dots$ $a\ \dots$
$\dots\ t_{j-1}$ $\dots\ (x_y)_{j-1}$	$\bar{a}$ $\bar{a}_w$	$\bar{a}$ a_-	$\bar{a}\ \dots$ $a_-\ \dots$	$\bar{a}$ a_-	$\bar{a}\ \dots$ $a\ \dots$

Table C.1: Illustrations of the transmitted and received sequences before removing the right-most forbidden subsequence, considered in the different steps of the proof of Lemma C.1. Note that $(x_y)_j$ designates r_j.

is no forbidden subsequence in the received block, $\hat{v}_1 = x_1 = v_1$ and $\hat{s}_1 = y_1 = s_1$.

2. There are some (at least one) forbidden subsequences in the received block. Let j be the position of the first bit of the right-most forbidden subsequence. "Right-most forbidden subsequence" means the closest one to the end of the block.

   - The right-most forbidden subsequence is that of a weak error. Then $x_j \dots x_{j+k'} = \bar{a}a^{k'}$ and $y_j, y_{j+1} \dots y_{j+k'-1}, y_{j+k'} = w, - \dots -, *$, where $a \in \{0,1\}$, $*$ and $- \in \{w, s\}$. By replacing the forbidden subsequence with its last symbol in $\mathbf{r}^N$, i.e. by replacing the corresponding bits in $\mathbf{x}^N$ and $\mathbf{y}^N$, we obtain sequences

$$\mathbf{x}'^{N'} = x'_1 \dots x'_{j-1}, x'_j, x'_{j+1}, \dots x'_{N'} = x_1 \dots x_{j-1}, a, x_{j+k'+1} \dots x_N$$
$$\mathbf{y}'^{N'} = y'_1 \dots y'_{j-1}, y'_j, y'_{j+1} \dots y'_{N'} = y_1 \dots y_{j-1}, *, y_{j+k'+1} \dots y_N,$$

   where $N' = N - k'$. There are the following possibilities.

     * $x_j \neq t_j$. Then a weak error has occurred at the j^{th} transmission and the other k' symbols are the correct receptions of the correcting bits, see the top section of Table C.1. Therefore, the mentioned replacement reduces the number of weak errors by one, i.e. $e'_w = e_w - 1$ and $e'_s = e_s$. The necessary condition still holds for the new $\mathbf{x}'^{N'}$ and $\mathbf{y}'^{N'}$, i.e.

$$k'e'_w + ke'_s = k'e_w - k' + ke_s < N - k' = N'.$$

* $x_j = t_j = \bar{a}$. Here two cases are possible.
   - ◇ $x_{j+1} = t_{j+1} = a$ which means that all transmissions from the $(j+2)^{\text{nd}}$ to the $(j+k')^{\text{th}}$ are received erroneously, see the middle section of Table C.1. Because if $t_j t_{j+1} = x_j x_{j+1} = \bar{a}a$ then due to tail condition $t_{j+2} \neq t_{j+1} = a = x_{j+2}$. Therefore, in order to correct the error at the $(j+2)^{\text{nd}}$ position, $t_{j+3} = t_{j+2} = \bar{a}$ which is erroneously received as a. This situation continues until the $(j+k')^{\text{th}}$ transmission. The mentioned replacement removes errors from the $(j+2)^{\text{nd}}$ to the $(j+k'-1)^{\text{st}}$ transmission ($k'-2$ errors) and reduces the number of weak and strong errors to e'_w and e'_s, respectively. As we can see, in the worst case where all these $k'-2$ errors are weak errors, the necessary condition still holds, i.e.

$$\begin{aligned}\max(k'e'_w + ke'_s) &= k'(e_w - (k'-2)) + ke_s \\ &= k'e_w + ke_s - k'(k'-2) \\ &< N - k'(k'-2) \\ &\leq N - k' = N', \end{aligned} \tag{C.1}$$

   as long as

$$k'(k'-2) \geq k' \qquad \text{or} \qquad k' \geq 3.$$

   - ◇ $x_{j+1} \neq t_{j+1} = \bar{a}$ which means that all transmissions from the $(j+1)^{\text{st}}$ to the $(j+k')^{\text{th}}$ are received erroneously, see the bottom section of Table C.1. Because $t_{j+2} = t_{j+1} = \bar{a}$ in order to correct the error at the $(j+1)^{\text{st}}$ transmission, but $x_{j+2} = a$, and this situation continues until the $(j+k')^{\text{th}}$ transmission. The mentioned replacement removes errors from the $(j+1)^{\text{st}}$ to the $(j+k'-1)^{\text{st}}$ transmission ($k'-1$ errors) and reduces the number of weak and strong errors to e'_w and e'_s, respectively. In the worst case where all these $k'-1$ errors are weak errors and when $k' > 2$, the necessary condition still holds, i.e.

$$\begin{aligned}\max(k'e'_w + ke'_s) &= k'(e_w - (k'-1)) + ke_s \\ &= k'e_w + ke_s - k'(k'-1) \\ &< N - k' = N'. \end{aligned}$$

  - The right most forbidden subsequence is that of a strong error. Here the argument is the same as that in the case of the forbidden subsequence of a weak error. In the worst case, a requirement similar to inequality (C.1) also holds because

$$k \geq k' > 2 \qquad \Rightarrow \qquad k'(k-2) \geq k.$$

3. After removing the right most forbidden subsequence, the resulting $\mathbf{r}'^{N'}$, i.e. $\mathbf{x}'^{N'}$ and $\mathbf{y}'^{N'}$, can still be seen as the outcomes of a SRFC scheme. This is obvious in the case of $x_j \neq t_j$. In $x_j = t_j$ case, we have $t'_j = t_j = \bar{a} = t_{j+k'}$, so both transmission scenarios (corresponding to after and before the replacement) behave similarly up to the j^{th} transmission.

   Since $x'_j = x_{j+k'} = a$ and $y'_j = y_{j+k'}$, we can imagine that an $\bar{a} \rightarrow a$ error (weak or strong, depending on $y_{j+k'}$) has occurred at the j^{th} position of block $\mathbf{r}'^{N'}$. This error is the same as the one occurred at the $(j+k')^{\text{th}}$ position of $\mathbf{r}^N$, resulting in $t_{j+k'} = \ldots = t_N = \bar{a}$ (otherwise there would be a forbidden subsequence on the right side of the j^{th} position of $\mathbf{r}^N$). Similarly to $\mathbf{t}^N_{j+k'}$ and the corresponding $\mathbf{r}^N_{j+k'}$, $\mathbf{t}'^{N'}_j$ and $\mathbf{r}'^{N'}_j$ represent a SRFC scenario, in which the transmitter has unsuccessfully tried to correct the $\bar{a} \rightarrow a$ error from the $(j+1)^{\text{st}}$ to the $(N')^{\text{th}}$ transmission, i.e. $t'_j = \ldots = t'_{N'} = \bar{a}$ and $r'_j, \ldots, r'_{N'} = a_*, r_{j+k'+1}, \ldots, r_N$.

Finally, if step 2 is repeated for all forbidden subsequences, then we will eventually arrive in step 1 which proves the Lemma C.1. Q.E.D.

Proof of Theorem 6.2 : Let $e_{v,w}$ and $e_{v,s}$ denote the number of weak and strong errors during the transmission of precoded sequence $\mathbf{v}^L$ and let $e_{t,w}$ and $e_{t,s}$ denote the number of weak and strong errors during the transmission of the tail sequence. Hence, $e_w = e_{v,w} + e_{t,w}$ and $e_s = e_{v,s} + e_{t,s}$. Denote $n = L + k'e_{v,w} + ke_{v,s}$. So the transmission of the last bit of precoded sequence $\mathbf{v}^L$ (or its last correction bit) is ended at the n^{th} transmission, therefore, the n^{th} transmission is a correct transmission, i.e. $x_n = t_n = v_L$. Moreover, the first bit of the tail is $z_{n+1} = t_{n+1} = \bar{v}_L$. Since we have $k'e_{t,w} + ke_{t,s} < N - n = N''$, the first transmitted bit of the tail, i.e. $t_{n+1} = z_{n+1}$, will correctly be decoded by block decoding method according to Lemma 1. Similarly, $t_n = v_L$ will be decoded correctly by this method since $k'e_{t,w} + ke_{t,s} < N - n < N'' + 1$. The fact that both x_n, x_{n+1} and y_n, y_{n+1} are decoded correctly to $v_L, \bar{v}_L$ and s_L, s_{L+1} (the status bit corresponding to the first bit of the tail) by the tail appending method, guarantees that no error propagation will occur from the tail part to the precoded data part, due to possible unfinished correction subsequences in the tail part. To illustrate this, let $\mathbf{x}'^{N'}$ denote the result of the right-to-left decoder when it has reached the n^{th} symbol in block $\mathbf{r}^N$. Then in

$$\dots, x'_{n-2}, x'_{n-1}, x'_n, x'_{n+1}, x'_{n+2}, \dots = \dots, x_{n-2}, x_{n-1}, v_L, \bar{v}_L, x'_{n+2}, \dots$$

the $\bar{v}_L$ cannot be a part of any forbidden subsequence when the right-to-left decoder moves on position $j < n$. Therefore, the first L bits of $\mathbf{x}^N$ and $\mathbf{y}^N$ (the precoded data part) will be decoded correctly by the algorithm. Q.E.D.

C.2 Proof to Theorem 6.3

The proofs of the theorem and the corresponding lemmas comprise the following steps.

- Determining $\hat{t}_i^{l-t-r}$ based on the random walk of sequence $\mathbf{r}_i^{(i+D)}$.
- Performing the right-to-left estimation on $\mathbf{r}_i^{(i+D)}$. After every replacement of the right-most forbidden subsequence, the estimate of the random walk of the resulting sequence is checked to be the same[1] as $\hat{t}_i^{l-t-r}$.
- Determining $\hat{t}_i^{r-t-l}$ when all forbidden subsequences in $\mathbf{r}_i^{(i+D)}$ are replaced.

Lemma C.2 *In the left-to-right estimator, if the random walk of sequence $\mathbf{r}_i^{(i+D)}$ starts from region $g_1 \in \{\mathcal{Z}_p, \mathcal{Z}_n\}$ and never enters the other region, then we have*

$$\hat{t}_i^{l-t-r} = \hat{t}_i^{r-t-l} = x_i \qquad \textit{for } k > k' \geq 2 \textit{ and } D \geq 1.$$

Proof : Without loss of generality assume that $x_i = 1$ and, therefore, $g_1 = \mathcal{Z}_p$. Since the random walk does not enter the region of negative states, $\hat{t}_i^{l-t-r} = x_i = 1$, see the upper part of Fig. C.1. For the result of the right-to-left estimator consider the following cases.

1. Assume that there are no forbidden subsequences in $\mathbf{r}_i^{(i+D)}$. Then, $\hat{t}_i^{r-t-l} = x_i = \hat{t}_i^{l-t-r}$.

[1] Then, at the end it will be enough to show that the results of both estimators are the same for the sequence obtained from removing all forbidden subsequences of $\mathbf{r}_i^{(i+D)}$ by the right-to-left estimator.

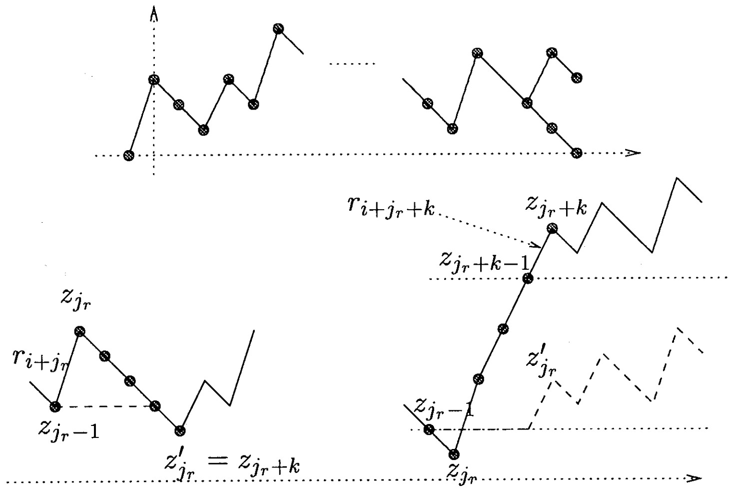


Figure C.1: Some paths illustrating the states of the random walk in the proof of Lemma 1 (dashed paths are the resulting random walks after removing a forbidden subsequence).

Remark C.1 Note that if there isn't any forbidden subsequence in $\mathbf{r}_i^{(i+D)}$ and $x_i = 1$, then the corresponding random walk, starting from state $z \geq 0$, does not go below state z according to the rules of the random walk. Similarly, if $x_i = 0$, then the corresponding random walk, starting from state $z \leq 0$, does not go above state z.

2. Assume that there are some forbidden subsequences in $\mathbf{r}_i^{(i+D)}$. Let $\mathbf{r}'^{D'}$ denote the sequence obtained from replacing the right-most forbidden subsequence of $\mathbf{r}_i^{(i+D)}$ with its last symbol. We have to show that the results of the left-to-right estimator are the same when it is applied to sequences $\mathbf{r}_i^{(i+D)}$ and $\mathbf{r}'^{D'}$, respectively. Without loss of generality, assume that the right-most forbidden subsequence is that of a strong error, then $D' = D - k$. In other words, we have to show that the random walk of the resulting sequence $\mathbf{r}'^{D'}$ does not enter region $\bar{g}_1 = \mathcal{Z}_n$.

   Let z_j and z'_j denote the states of the random walks after the j^{th} step corresponding to sequences $\mathbf{r}_i^{(i+D)}$ and $\mathbf{r}'^{D'}$, respectively. Let $r_{i+j_r} \ldots r_{i+j_r+k}$ be the right-most forbidden subsequence of $\mathbf{r}_i^{(i+D)}$. Obviously, $z'_j = z_j$, $j = 1, \ldots, (j_r - 1)$. If the right-most forbidden subsequence is $1_s 0_-^k$, then $z_{j_r-1} = z_{j_r-1+k}$, see left-lower part of Fig. C.1. In this case $z'_j = z_{j+k}$,

$j = j_r, \ldots, D'$, and the position of the end point of the random walk will not change by the replacement.

Assume that the right-most forbidden subsequence is $0_s 1_-^k$. In this case, $z_{j_r-1} > 0$, otherwise the random walk of $\mathbf{r}_i^{(i+D)}$ had entered a negative state, see right-lower part of Fig. C.1. The sequence $r_{i+j_r+k} \ldots r_{i+D}$ is without any forbidden subsequence, therefore, $z_j \geq z_{j_r+k-1}$, $j = (j_r + k), \ldots, D$, according to Rem. C.1. Therefore, after the replacement (i.e. deleting states between z_{j_r-1} and z_{j_r+k}), the random walk will not go below $z'_{j_r-1} = z_{j_r-1} > 0$ and, consequently, none of z'_j, $j = 1, \ldots, D'$, will be in the region of negative states. Finally, the result of the left-to-right estimator does not change after removing the right-most forbidden subsequence.

If we repeat step 2 for all forbidden subsequences, we will arrive in step 1 which proves the lemma. Q.E.D.

Lemma C.3 *In the left-to-right estimator, if the random walk of sequence* $\mathbf{r}_i^{(i+D)}$ *starts from region* $g_1 \in \{\mathcal{Z}_p, \mathcal{Z}_n\}$ *and enters the other region at the* j_0^{th} *step of the walk, without returning back to region* g_1*, then we have*

$$\hat{t}_i^{l-t-r} = \hat{t}_i^{r-t-l} = x_{i+j_0} \qquad \text{for } k > k' \geq 2 \text{ and } D \geq 1.$$

Proof : Without loss of generality assume that $x_i = 1$. Since the random walk does not return to the region of positive states, $\hat{t}_i^{l-t-r} = x_{i+j_0} = 0$, see Fig. C.2. If the right-to-left decoding is applied to sequence $r_{i+j_0} \ldots r_{i+D}$, then we obtain $r'_{i+j_0} \ldots r'_{i+D'}$ which does not have any forbidden subsequences and $D' \leq D$. According to Lemma 1, $x'_{i+j_0} = x_{i+j_0} = 0$.

The path of the left-to-right estimator for $\mathbf{r}'^{D'} = r_i \ldots r_{i+j_0-1}, r'_{i+j_0} \ldots r'_{i+D'}$ is the same as that of $\mathbf{r}_i^{(i+D)}$ in region $g_1 = \mathcal{Z}_p$, it enters region $g_2 = \mathcal{Z}_n$ at the j_0^{th} step and does not return back to region $\mathcal{Z}_p$ (because sequence $r'_{i+j_0} \ldots r'_{i+D'}$ does not have any forbidden subsequence, see Rem. C.1).

Firstly, we claim that there is at least one forbidden subsequence in $\mathbf{r}'^{D'}$ starting at position $j < j_0$, otherwise, the random walk corresponding to $\mathbf{r}'^{D'}$ would not enter the region of negative states.

Secondly, the right-most forbidden subsequence in $\mathbf{r}'^{D'}$ is either $1_- 0_-^{k'}$ or $1_- 0_-^k$. Otherwise, for example, the random walk of $0_- 1_-^{k'}$, which is located above the 0-state, would end up in a state at least $k'(k'-1)$. Since there

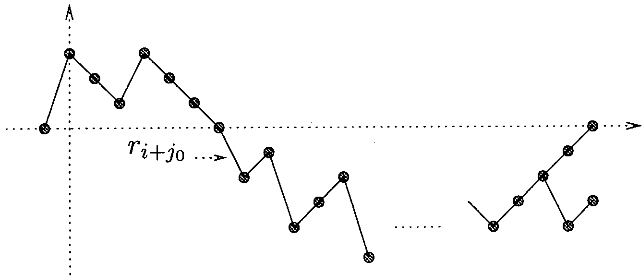


Figure C.2: A path for the random walk in the proof of Lemma C.3.

is no further forbidden subsequence in $\mathbf{r}'^{D'}$ at the right-hand side of the last 1_- of the $0_-1_-^{k'}$, the random walk should not come below state $k'(k'-1)$, in accordance with Rem. C.1. This is a contradiction and, therefore, the right-most forbidden subsequence in $\mathbf{r}'^{D'}$ is either $1_-0_-^{k'}$ or $1_-0_-^{k}$.

Let z' and z'' denote the states of the random walk before 1_- and the last 0_- of the right-most forbidden sequence $1_-0_-^{k'}$ (or $1_-0_-^{k}$). Since $z', z'' \in \mathcal{Z}_p$ or are zero, we have $z' = z''$. The random walk of $\mathbf{r}''^{(D'-k')}$ (or $\mathbf{r}''^{(D'-k)}$), obtained from replacing the right-most $1_-0_-^{k'}$ (or $1_-0_-^{k}$) in $\mathbf{r}'^{D'}$ with its last correction symbol, has a walk similar to that of $\mathbf{r}'^{D'}$ except it enters into the region of negative states at its $(j_0-k')^{\text{th}}$ step (or $(j_0-k)^{\text{th}}$ step) because $z' = z''$. If we continue in this way, all symbols before the $r'_{i+j_0} \cdots r'_{i+D'}$ sequence will disappear and the right-to-left decoding results in $r'_{i+j_0} \cdots r'_{i+D'}$. Finally, the right-to-left estimate will be $x'_{i+j_0} = x_{i+j_0}$ according to Lemma C.2. Note that if the left-to-right estimation method is applied to any sequence resulting from a forbidden subsequence replacement mentioned here, then the left-to-right estimate is always the same as $\hat{t}_i^{l-t-r} = x_{i+j_0} = 0$. Q.E.D.

Proof of Theorem 6.3 : Let $g_1 \ldots g_A$ be the sequence of regions in which the random walk of $\mathbf{r}_i^{(i+D)}$ enters, see Fig. C.3 for an illustration. According to the definition of j_0, region g_A is entered at the j_0^{th} step, therefore, $\hat{t}_i^{l-t-r} = x_{i+j_0}$.

First the right-to-left estimator removes the forbidden subsequences in the last region of sequence $\mathbf{r}_i^{(i+D)}$. The random walk of the resulting sequence, $\mathbf{r}'^{D'}$, still enters region g_A with x_{i+j_0}, according to Lemma 1. Then, according to Lemma 2, the right-to-left estimator continues to remove all symbols of $\mathbf{r}'^{D'}$ lying in region g_{A-1} and yields sequence $\mathbf{r}''^{D''}$. This sequence consists of two parts: sequence $\mathbf{r}''_1$ representing those symbols of $\mathbf{r}_i^{(i+D)}$, whose random walk is located in regions $g_1 \ldots g_{A-2}$ and sequence $\mathbf{r}''_2$ representing those symbols of $\mathbf{r}_i^{(i+D)}$, whose random walk is located in region g_A and from which all forbidden subsequences are replaced by the right-to-left estimator.

The type bit of the received symbol at which region $g_{(A-2)}$ is entered is $x_{(i+j_1)} = x_{(i+j_0)}$, see Fig C.3, because $g_{A-2} = g_A$ which also implies that both regions have the same rules (see Fig. 6.4) for the random walk. Therefore, not only sequence $\mathbf{r}''_2$ is concatenated with sequence $\mathbf{r}''_1$, but also its random walk can be seen as the continuation of that of $\mathbf{r}''_1$ in region g_{A-2} (i.e. without entering region $\bar{g}_{A-2}$). Now we can assume that the sequence of regions of the path of sequence $\mathbf{r}''^{D''}$ is $g_1, \ldots, g_{(A-2)}$. This implies that the random walk of $\mathbf{r}''^{D''}$ gives the estimate of $x_{i+j_1} = x_{i+j_0}$.

Repeating the same argument for $\mathbf{r}''^{D''}$, either g_1 will be removed or $g_1 = g_A$ at the end of the right-to-left estimation procedure. In either case, $\hat{t}_i^{r-t-l} = x_{i+j_0}$. Q.E.D.

Remark C.2 Using this proof, we can also illustrate the bottle neck of a left-to-right algorithm for nonbinary input MRFC schemes. Let us assume that, similarly to that in the binary input schemes, the left-to-right algorithm for nonbinary input schemes uses different walking rules in $|\mathcal{X}|$ regions, where $\mathcal{X}$ is the input alphabet of the channel. In such a nonbinary input case, when $g_A \neq g_{(A-2)}$, it is not possible to say that the path of symbols in g_A is a continuation of that in region $g_{(A-2)}$. After removing the symbols in region g_{A-1} by the right-to-left estimator, it is very much possible that the corresponding symbols of sequence $\mathbf{r}''_2$ drive the path out of region $g_{(A-2)}$ and alter the estimate of the left-to-right estimator for $\mathbf{r}''^{D''}$ with respect to that for $\mathbf{r}_i^{(i+D)}$. □

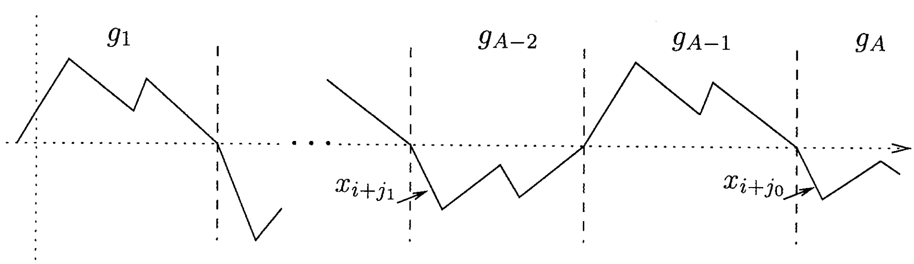


Figure C.3: An illustrative random walk for the proof of Theorem 6.3.

Bibliography

[1] R. Ahlswede. A constructive proof of the coding theorem for discrete memoryless channels with feedback. In *Proc. 6^{th} Prague Conf. on Inform. Theory, Statistical Decision Functions and Random Processes*, pp. 39–59, 1971.

[2] F. Alajaji. Feedback does not increase the capacity of discrete channels with additive noise. *IEEE Trans. Inform. Theory*, 41(2), pp. 546–549, March 1995.

[3] P.H. Algoet and B.H. Marcus. Large deviation theorems for empirical types of Markov chains constrained to thin sets. *IEEE Trans. Inform. Theory*, 38(4), pp. 1276–1291, July 1992.

[4] D.W. Becker. *Multiple-repetition feedback coding.* PhD thesis, University of California, 1973.

[5] T.C. Bell, J.G. Cleary, and I.H. Witten. *Text Compression.* Prentice Hall, 1990.

[6] E.R. Berlekamp. *Block coding with noiseless feedback.* PhD thesis, MIT, Cambridge, MA, 1964.

[7] R.E. Blahut. *Digital Transmission of Information.* Addison Wesley 1990.

[8] P. Bloemen. *Discrete coding schemes for two-way channels.* PhD thesis, Eindhoven University of Technology, 1996.

[9] G. Bratt. *Sequential decoding for the Gilbert-Elliott channel, strategy and analysis.* PhD thesis, Lund University, June 1994.

[10] M.V. Burnashev. Data transmission over a discrete channel with feedback, random transmission time. *Problemy Peredachi Informatsii*, 12(4), pp. 10–30, 1976.

[11] M.V. Burnashev. Reliability function of a binary symmetrical channel with feedback. *Problemy Peredachi Informatsii*, 24(1), pp. 3–10, 1988.

[12] S. Butman. A general formulation of linear feedback communication systems with solutions. *IEEE Trans. Inform. Theory*, 15(3), pp. 392–400, May 1969.

[13] A.R. Calderbank. Bandwidth efficient communication. EIDMA mini-course, Eindhoven University of Technology, March 1996.

[14] R. Campbell. *Linear feedback codes.* PhD thesis, Department of Applied Physics, Harvard University, Cambridge, MA, 1967.

[15] T.M. Cover. Enumerative source encoding. *IEEE Trans. Inform. Theory*, 19(1), pp. 73–77, January 1973.

[16] T.M. Cover and J.A. Thomas. *Elements of Information Theory.* Wiley, New York, 1991.

[17] I. Csiszár, T.M. Cover, and B. Choi. Conditional limit theorems under Markov conditions. *IEEE Trans. Inform. Theory*, 33(4), pp. 788–801, July 1987.

[18] A.J. de Bart. Coding and shaping for channels with intersymbol interference. Master's thesis, Eindhoven University of technology, 1995.

[19] R.L. Dobrushin. Information transmission in a channel with feedback. *Theory of Prob. and its Appl.*, 3(4), pp. 367–383, December 1958.

[20] A. Drukarev and JR. D.J. Costello. Hybrid ARQ error control using sequentional decoding. *IEEE Trans. Inform. Theory*, 29(4), pp. 521–535, July 1983.

[21] P. Elias. Coding for noisy channels. *IRE Conv. Rec.*, 3(4), 37–46, 1955.

[22] R.J.F. Fang. Lower bounds on reliability functions of variable-length nonsystematic convolutional codes for channels with noiseless feedback. *IEEE Trans. Inform. Theory*, 17(2), pp. 161–171, March 1971.

[23] G.D. Forney, Jr. Exponential error bounds for erasure, list and decision feedback schemes. *IEEE Trans. Inform. Theory*, 14(2), pp. 206–220, March 1968.

[24] N.T. Gaarder and J.K. Wolf. The capacity region of a multiple access discrete memoryless channel can increase with feedback. *IEEE Trans. Inform. Theory*, 21(1), pp. 100–102, January 1975.

[25] R.G. Gallager. *Information Theory and Reliable Communication.* Wiley, New York, 1968.

[26] Jr. G.D. Forney and G. Ungerboeck. Modulation and coding for linear Gaussian channels. *IEEE Trans. Inform. Theory*, 44(6), pp. 2384–2415, October 1998.

[27] A.J. Goldsmith and P. Varaiya. Capacity, mutual information and coding for finite-state Markov channels. *IEEE Trans. Inform. Theory*, 42, pp. 868–886, May 1996.

[28] P.E. Green. Feedback communication systems. In E. J. Baghdady, editor, *Lectures on Communication System Theory*, chapter 14. McGraw-Hill, New York, 1961.

[29] M. Horstein. Sequential transmission using noiseless feedback. *IEEE Trans. Inform. Theory*, 9, pp. 136–143, July 1963.

[30] G.H. Im. Bandwidth efficient digital transmission over unshielded twisted pair wiring. *IEEE J. Select. Areas Commun.*, 13(9), pp. 1643–1655, December 1995.

[31] K.A. Schouhamer Immink. *Coding Techniques for Digital Recorders.* Prentice Hall, 1991.

[32] F. Jelinek. *Coding for discrete memoryless two-way channels.* PhD thesis, MIT, Cambridge, 1962.

[33] C.B. Jones. An efficient coding system for long source sequences. *IEEE Trans. Inform. Theory*, 27(3), pp. 280–291, May 1981.

[34] R.L. Kashyap. Feedback coding schemes for an additive noise channel with a noisy feedback link. *IEEE Trans. Inform. Theory*, 14(3), pp. 471–480, May 1968.

[35] A.J. Kramer. *Analysis of communication schemes using an intermittent feedback link.* PhD thesis, Stanford University, 1967.

[36] A.J. Kramer. Improving communication reliability by use of an intermittent feedback channel. *IEEE Trans. Inform. Theory*, 15(1), pp. 52–60, January 1969.

[37] B.D. Kudryashov. Message transmission over a discrete channel with noiseless feedback. *Problemy Peredachi Informatsii*, 15(1), pp. 3–13, 1979.

[38] G.N.N. Martin, Jr. G.G. Langdon, and S.J.P. Todd. Arithmetic codes for constrained channels. *IBM J. Res. Develop.*, 27(2), pp. 94–106, March 1983.

[39] R.J. McEliece and W.E. Stark. Channels with block interference. *IEEE Trans. Inform. Theory*, 30(1), pp. 44–53, January 1983.

[40] H.B. Meeuwissen. *Information theoretical aspects of two-way communication.* PhD thesis, Eindhoven University of Technology, 1998.

[41] J.J. Metzner. *Reliable Data Communications.* Academic Press, 1998.

[42] M. Mushkin and I. Bar-David. Capacity and coding for the Gilbert-Elliott channels. *IEEE Trans. Inform. Theory*, 35, pp. 1277–1290, November 1989.

[43] A.M. Noll. *Introduction to Telephones & Telephone Systems.* Artech House, second edition, 1991.

[44] J.K. Omura. Optimum linear transmission of analog data for channels with feedback. *IEEE Trans. Inform. Theory*, 14(1), pp. 38–43, January 1968.

[45] W.D. O'Neill. An application of Shannon's coding theorem to information transmission in economic markets. *Information Sciences*, 41, pp. 171–185, 1987.

[46] W.D. O'Neill. Profit maximizing and price distortion minimizing codes for a channel model of an asset market. *IEEE Trans. Inform. Theory*, 41(6), pp. 2009–2014, November 1995.

[47] J.M. Ooi. *A framework for low-complexity communication over channels with feedback.* PhD thesis, MIT, Cambridge, MA, 1997.

[48] J.M. Ooi and G.W. Wornell. Fast iterative coding for feedback channels. In *Proc. IEEE Int. Symp. Inform. Theory, Ulm, Germany*, page 133, July 1997.

[49] J.M. Ooi and G.W. Wornell. Fast iterative coding for feedback channels. *IEEE Trans. Inform. Theory*, 44(7), pp. 2960–2976, November 1998.

[50] L.H. Ozarow. *Coding and capacity for additive white Gaussian noise multi-user channels with feedback.* PhD thesis, MIT, Cambridge, 1979.

[51] J.E. Savage. A bound on the reliability of block coding with feedback. *Bell Syst. Tech. J.*, 45, pp. 967–977, 1966.

[52] J.E. Savage. Complexity of decoders: I-classes of decoding rules. *IEEE Trans. Inform. Theory*, 15(6), pp. 689–695, November 1969.

[53] J.E. Savage. The complexity of decoders-part 2: computational work and decoding time. *IEEE Trans. Inform. Theory*, 17(1), pp. 77–85, 1971.

[54] J.P.M. Schalkwijk. A coding scheme for additive noise channels with feedback part 2: band-limited signals. *IEEE Trans. Inform. Theory*, 12(2), pp. 183–189, April 1966.

[55] J.P.M. Schalkwijk. Center-of-gravity information feedback. *IEEE Trans. Inform. Theory*, 14(2), pp. 324–331, March 1968.

[56] J.P.M. Schalkwijk. Recent developments in feedback communication. *Proc. of the IEEE*, 57(7), pp. 1242–1249, July 1969.

[57] J.P.M. Schalkwijk. A formal language approach to two-way communication. Technical Report R 743, Philips research laboratories, 1970. Reprinted from: Philips research reports 25, 430-436, december 1970.

[58] J.P.M. Schalkwijk. A class of simple and optimal strategies for block coding on the binary symmetric channel with noiseless feedback. *IEEE Trans. Inform. Theory*, 17(3), pp. 283–287, May 1971.

[59] J.P.M. Schalkwijk. An algorithm for source coding. *IEEE Trans. Inform. Theory*, 18(3), pp. 395–399, May 1972.

[60] J.P.M. Schalkwijk. Coding for a computer network. Technical Report 74-E-52, Eindhoven University of Technology, October 1974.

[61] J.P.M. Schalkwijk. A coding scheme for duplex channels. *IEEE Trans. Commun.*, 22(9), pp. 1369–1373, September 1974.

[62] J.P.M. Schalkwijk. The binary multiplying channel - a coding scheme that operates beyond the Shannon inner bound. *IEEE Trans. Inform. Theory*, 28(1), pp. 107–110, January 1982.

[63] J.P.M. Schalkwijk. Communication principles. Syllabus TUE, 1989.

[64] J.P.M. Schalkwijk and M.E. Barron. Sequential signalling under a peak power constraint. *IEEE Trans. Inform. Theory*, 17(3), 278–282, 1971.

[65] J.P.M. Schalkwijk and T. Kailath. A coding scheme for additive noise channels with feedback-part 1: no bandwidth constraint. *IEEE Trans. Inform. Theory*, 12(2), pp. 172–182, April 1966.

[66] J.P.M. Schalkwijk and K.A. Post. On the error probability for a class of binary recursive feedback strategies. *IEEE Trans. Inform. Theory*, 19(4), pp. 498–511, July 1973.

[67] C.E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(3), pp. 379–423, July 1948.

[68] C.E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27, pp. 623–656, October 1948.

[69] C.E. Shannon. The zero error capacity of a noisy channel. *IRE Trans. Inform. Theory*, 2, pp. 8–19, September 1956.

[70] C.E. Shannon. Some geometrical results in channel capacity. *Nachrichtentechnische Zeit*, 10, 1957.

[71] C.E. Shannon. Channels with side information at the transmitter. *IBM J. Res. and Develop.*, 2, pp. 289–293, October 1958.

[72] C.E. Shannon. Two-way communication channels. In Jerzy Neyman, editor, *Proceedings of the 4th Berkeley symposium on Mathematical Statistics and Probability*, pages 611–644, Berkeley, 1961.

[73] C.E. Shannon. Probability of error for optimal codes in a Gaussian channel. *The Bell System Technical Journal*, 38, pp. 611–656, May 1969.

[74] M. Shoae Bargh. Low rate noiseless feedback coding strategies. Technical report, Eindhoven University of Technology, February 1997.

[75] J. P. M. Schalkwijk and M. Shoae Bargh. Coding for channels with low rate noiseless feedback. In *Proc. 17th Symp. Inform. Theory Benelux*, pp. 121–127, May 1996.

[76] M. Shoae Bargh and J. P. M. Schalkwijk. A generalization of multiple repetition feedback coding schemes to soft output discrete memoryless channels. In *Proc. IEEE Int. Symp. Inform. Theory, MIT, USA*, 1998.

[77] M. Shoae Bargh and J.P.M. Schalkwijk. Coding for channels with low rate noiseless feedback. In *Proc. of IEEE Int. Symp. Inform. Theory, Ulm, Germany*, pp. 129, July 1997.

[78] M. Shoae Bargh and J.P.M. Schalkwijk. Feedback coded modulation. In *Proc. 18th Symp. Inform. Theory Benelux*, pp. 41–48, May 1997.

[79] M. Shoae Bargh and J.P.M. Schalkwijk. A block retransmission strategy for multiple repetition feedback coding schemes. In *Proc. 19th Symp. Inform. Theory Benelux*, pp. 91–98, May 1998.

[80] M. Shoae Bargh and J.P.M. Schalkwijk. Repetition feedback coding schemes for binary-input soft-output discrete memoryless channels with recursive decoding. In *Proc. 19th Symp. Inform. Theory Benelux*, pp. 201–208, May 1998.

[81] M. Shoae Bargh and J.P.M. Schalkwijk. On error correction in information feedback schemes. In *Proc. 20th Symp. Inform. Theory Benelux*, pp. 173–180, May 1999.

[82] W.E. Stephens, H. Samueli, and G. Cherubini. Copper wire access technologies for high performance networks. *IEEE J. Select. Areas Commun.*, 13(9), pp. 1538–1539, December 1995.

[83] S.J.P. Todd, Jr. G.G. Langdon, and G.N.N. Martin. A general fixed rate arithmetic coding method for constrained channels. *IBM J. Res. and Develop.*, 27(2), pp. 107–115, March 1983.

[84] G.L. Turin. Signal design for sequential detection systems with feedback. *IEEE Trans. Inform. Theory*, 11, pp. 401–408, July 1965.

[85] G. Ungerboeck. Trellis-coded modulation with redundant signal sets-part 2: state of the art. *IEEE Commun. Mag.*, 25(2), pp. 12–21, 1987.

[86] G. Ungerboeck. Channel coding with multilevel/phase signals. *IEEE Trans. Inform. Theory*, 28(1), pp. 55–67, January 1982.

[87] W.M.C.J. van Overveld. *On the capacity region for deterministic two-way channels and write-unidirectional memories.* PhD thesis, Eindhoven University of technology, 1990.

[88] R.S. Varga. *Matrix Iterative Analysis.* Prentice Hall, 1962.

[89] T. Veugen. Proof of the median paths. Unpublished, private commun.

[90] T. Veugen. A simple class of capacity achieving strategies for discrete memoryless channels with feedback. *IEEE Trans. Inform. Theory*, 42(6), pp. 2221–2227, November 1996.

[91] T. Veugen. *Multiple repetition coding for channels with feedback.* PhD thesis, Eindhoven University of Technology, 1997.

[92] A.J. Viterbi. The effect of sequential decision feedback on communication over the Gaussian channel. *Inform. and control*, 8, pp. 80–92, February 1965.

[93] E.J. Weldon, Jr. An upper bound on the probability of a word error for block codes used with the memoryless binary symmetric channel with noiseless information feedback. Memorandum No. 38763-15, Bell telephone laboratories, August 1962.

[94] F.M.J. Willems and J.J. Wuijts. A pragmatic approach to shaped coded modulation. In *IEEE* 1^{st} *Symp. on Communications and Vehicular Technology in the Benelux*, October 1993.

[95] J. Wolfowitz. *Coding Theorems of Information Theory*. Springer-Verlag, Berlin, 1978.

[96] J.M. Wozencraft and I.M. Jacobs. *Principles of Communication Engineering*. Wiley, New York, 1965.

[97] J.J. Wuijts. De constructie van een energiebegrensde kanaalcode met behulp van enumeratieve technieken (stageverslag). Eindhoven University of Technology, October 1991.

[98] A.D. Wyner. On the Schalkwijk-Kailath coding scheme with a peak energy constraint. *IEEE Trans. Inform. Theory*, 14(1), pp. 129–134, January 1968.

[99] H. Yamamoto and K. Itoh. Asymptotic performance of a modified Schalkwijk-Barron scheme for channels with noiseless feedback. *IEEE Trans. Inform. Theory*, 25(6), pp. 729–733, November 1979.

[100] H. Yamamoto and K. Itoh. Viterbi decoding algorithm for convolutional codes with repeat request. *IEEE Trans. Inform. Theory*, 26(5), pp. 540–547, September 1980.

[101] K.Sh. Zigangirov. Information transmission over a binary symmetrical channel with noiseless feedback (random transmission time). *Problemy Peredachi Informatsii*, 4(3), pp. 38–47, 1968.

[102] K.Sh. Zigangirov. Upper bounds for the error probability for channels with feedback. *Problemy Peredachi Informatsii*, 6(2), pp. 87–92, 1970.

[103] K.Sh. Zigangirov. Number of correctable errors for transmission over a binary symmetrical channel with feedback. *Problemy Peredachi Informatsii*, 12(2), pp. 3–19, 1976.

[104] K.Sh. Zigangirov. Recurrent transmission through the binary symmetric channel with feedback. *Problems of Control and Information Theory*, 6(4), pp. 189–205, 1977. English translation at pages 12–25.

Samenvatting

In communicatiesystemen is er vaak sprake van een tweewegkanaal. Er is dan naast de verbinding van de zender naar de ontvanger, het voorwaartse kanaal, ook een verbinding van de ontvanger naar de zender, het feedback-kanaal. Een telefoonverbinding is een voorbeeld van zo'n tweewegkanaal. Als de ene kant spreekt, luistert de andere kant, en het feedback-kanaal blijft ongebruikt. Feedback-strategieën gebruiken dit onbenutte feedback-kanaal tijdens hun coderingsproces. Daardoor kunnen deze strategieën, in vergelijking met strategieën die geen gebruik maken van dit feedback-kanaal, over het algemeen met een lagere complexiteit van de decoder over het voorwaartse kanaal communiceren, of met een hogere betrouwbaarheid, of met een hogere capaciteit, of met een combinatie van deze voordelen. Wij beschouwen hier informatie feedback-strategieën waarbij de zender elk ontvangen symbool, zonder vertraging, via een ruisvrij feedback-kanaal ziet. Dit geeft de zender de mogelijkheid elke opgetreden fout te corrigeren. 'Multiple Repetition Feedback Coding' (MRFC) strategieën vormen een klasse van eenvoudige en efficiënte informatie feedback-strategieën voor discrete geheugenloze kanalen. Deze strategieën vormen de kern van de twee onderzoeksrichtingen beschreven in deze dissertatie. In het eerste deel worden deze strategieën toegepast op kanalen waarbij het feedback-kanaal niet ruisvrij is, en in het tweede deel worden deze strategieën, voor de situatie met een ruisvrij feedback-kanaal, verbeterd, en toegepast op de klasse van discrete geheugenloze 'soft-output'-kanalen.

Een belangrijke tekortkoming van bestaande informatie feedback-strategieën is dat ze aannemen dat het feedback-kanaal ruisvrij is. In de praktijk is dit echter niet altijd een realistische aanname. Voor deze situatie hebben wij in het eerste deel van deze dissertatie feedback-strategieën ontworpen voor een ruisvrij feedback-kanaal met een lagere transmissiesnelheid. Doordat hier het aantal transmissies op het feedback-kanaal kleiner is dan op het voorwaartse kanaal, is het mogelijk de feedback-informatie te beschermen tegen de ruis op het kanaal. In het bijzonder is er een bandbreedte-beperkte modulatie-methode ontworpen met behulp van MRFC strategieën voor additieve witte Gaussische ruiskanalen waarbij zowel het voorwaartse kanaal als het feedback-kanaal dezelfde signaal-ruisverhouding hebben. De hieruit voortkomende methode geeft een verbetering van 5,5 dB ten opzichte van ongecodeerde transmissies. In deze methode is de complexiteit van de decoder, in vergelijking met de complexiteit van de decoder van 'trellis coded modulation', aanzienlijk verlaagd. Omdat we één bit per dimensie over het feedback-kanaal moeten versturen, kan het feedback-kanaal in principe als foutvrij beschouwd worden voor kanalen met gematigde signaal-ruisverhoudingen. De theoretisch berekende verbetering van deze methode wordt ondersteund door de resultaten van simulaties.

In het tweede deel van dit proefschrift, worden de MRFC strategieën verbeterd en uitgebreid voor de klasse van 'soft-output'-kanalen. Er zijn twee typen MRFC strategieën: bloksgewijze MRFC strategieën en recursieve MRFC strategieën. Ten eerste zijn er twee nieuwe bloksgewijze MRFC hertransmissie strategieën ontworpen met betere prestaties dan bestaande bloksgewijze MRFC strategieën. Eén van deze bloksgewijze MRFC strategieën geeft op asymmetrische discrete geheugenloze kanalen zelfs betere prestaties dan bestaande bloksgewijze en recursieve MRFC strategieën. Ten tweede zijn de MRFC strategieën aangepast voor 'soft-output'-kanalen. Een discreet geheugenloos 'soft-output'-kanaal kan bijvoorbeeld verkregen worden door het quantiseren van de outputs van een continu kanaal met discrete inputs. Deze 'soft-output'-kanalen worden gedefinieerd met behulp van een specifieke klasse van 'compound'-kanalen. Voor deze klasse van 'compound'-kanalen is bewezen dat het gebruik van feedback of vertraagde 'side-information' de capaciteit van het kanaal niet kan verhogen. Deze aanpassing van de MRFC strategieën wordt verduidelijkt aan de hand van een discreet geheugenloos kanaal met twee inputs en vier outputs. De nieuwe strategie resulteert in een transmissiesnelheid die heel dicht bij de kanaalcapaciteit ligt, mits een geschikte Markov bron als precoder wordt gebruikt. Een MRFC strategie bestaat uit een repetitie-deel en een precoder-deel. Tot dusver is hier alleen het repetitie-deel beschreven, maar de aangepaste MRFC strategie voor 'soft-output'-kanalen kan geen gebruik maken van bestaande precoders. Daarom is er, tenslotte, een precoder ontworpen en geëvalueerd die ook voor deze aangepaste MRFC strategie gebruikt kan worden.

Biography

Mortaza Shoae Bargh was born on April 17, 1966 in Tabriz, Iran. After finishing high school in 1984, he went to Isfahan where he obtained the BSc degree in Electronic Engineering from Isfahan University of Technology in 1989. From 1989 to 1992, he carried out his military service and then he worked in a power distribution company for one year.

In January 1993, Mortaza came to The Netherlands in order to continue his education. He received the Master of Electronic Engineering degree with distinction from the Eindhoven International Institute in the summer of 1994, and next he also obtained the "ingenieur" degree in Electrical Engineering from the Eindhoven University of Technology. His master's thesis is concerned with the Wavelet transform and its applications. In February 1995, he started his postgraduate studies in the designer course of Information and Communication Technology and two years later he obtained the Master of Technological Design degree from the "Stan Ackermans Instituut", Eindhoven University of Technology. Since February 1997, Mortaza has been working towards a doctorate in the field of Information and Communication Theory at the Radiocommunication Group of the faculty of Electrical Engineering, Eindhoven University of Technology, of which the results are presented in this thesis.

Stellingen

behorende bij het proefschrift

Coding Strategies for Channels with Feedback

door

Mortaza Shoae Bargh

1. In the era of the information super high-way, the applications of versatile and advanced communication systems are rapidly expanding. The advantages of feedback coding schemes over one-way coding schemes, in particular their lower decoding complexity, will make the feedback coding schemes an important design option for implementing such complex systems. *This thesis, Chapter 7.*

2. Regarding his information feedback coding method, James Ooi in [1, pp.25] says that:

    > "This framework shows the problem of channel coding with feedback to be strongly related to the problem of source coding."

    In the precoding stage of the MRFC schemes we again see a relation between channel coding with feedback and source coding.

[1] J.M. Ooi. *A framework for low-complexity communication over channels with feedback.* PhD thesis, Massachusetts Institute of Technology, 1997.

3. In most practical channels with memory, errors tend to occur in bursts. On the other hand, the length of the block in MRFC schemes can easily be made large enough to accommodate a few runs of burst errors. Therefore, FCM with block MRFC can operate better in channels with burst errors than TCM. *This thesis, Chapter 7.*

4. In MRFC for a BSC with crossover probability p, the rule of $k = - < \log p >$[1] is proposed in [1] for choosing the repetition parameter in order to maximize the transmission rate. This seems intuitively a reasonable choice because a transmission error, which occurs with probability p, needs $\log \frac{1}{p}$ bits (or transmissions) to be commuted to the decoder [1].

    One must note that the precoding of input data is also needed to differentiate between correct receptions and erroneous ones, i.e. to produce an acknowledgement/rejection mechanism. Thus, a combination of precoding redundancy and repetitions is the cost paid to correct the channel errors in MRFC. *This thesis, Chapters 1 and 3.*

[1] T. Veugen. *Multiple repetition coding for channels with feedback.* PhD thesis, Eindhoven University of Technology, 1997.

5. In [1], the choice of the repetition parameters in a MRFC scheme is based on maximizing the transmission rate. Another criterion to choose the

[1] Operator $< z >$ gives the closest integer to real number z.

repetition parameters in a block MRFC scheme could be to maximize the transmission rate for a given correctable error fraction of the block. These two approaches, suitable for the asymptotic and non-asymptotic cases, respectively, may lead to different outcomes.

[1] T. Veugen. *Multiple repetition coding for channels with feedback.* PhD thesis, Eindhoven University of Technology, 1997.

6. Free media is a necessity in a democratic society. Nevertheless, the public should still approach the news with a critical attitude to avoid hearing only the loudest voice.

7. What amazes me most is the extent of the similarities among people of different cultural, ethnic and geographical backgrounds.

8. One way to get to know other cultures is to travel into other countries, not just to other countries.

9. "A perfume is all about the delicacy of its fragrance, not about the brand name of its producer."

   A rough translation of a quotation from **Saadi**, Iranian poet, 13$^{\text{th}}$ century AD.

   If Saadi wanted to say this today, he would probably use "and" instead of "not"!

10. *"Spend time with wine by a stream,*

    And let sorrows away stream.

    My life, like a rose, is but few days;

    Youthful and joyous live this dream."

    Hafiz, Iranian poet, 14$^{\text{th}}$ century AD. Translated by Shahriar Shahriari, http://www.zbnet.com/hafiz/rubaiyat/.

11. " ... what is intuitive is a purely subjective matter on which opinions may easily differ widely."

    Jacob Wolfowitz, Coding Theorems of Information Theory, pp. 14, Springer-Verlag, 1978.