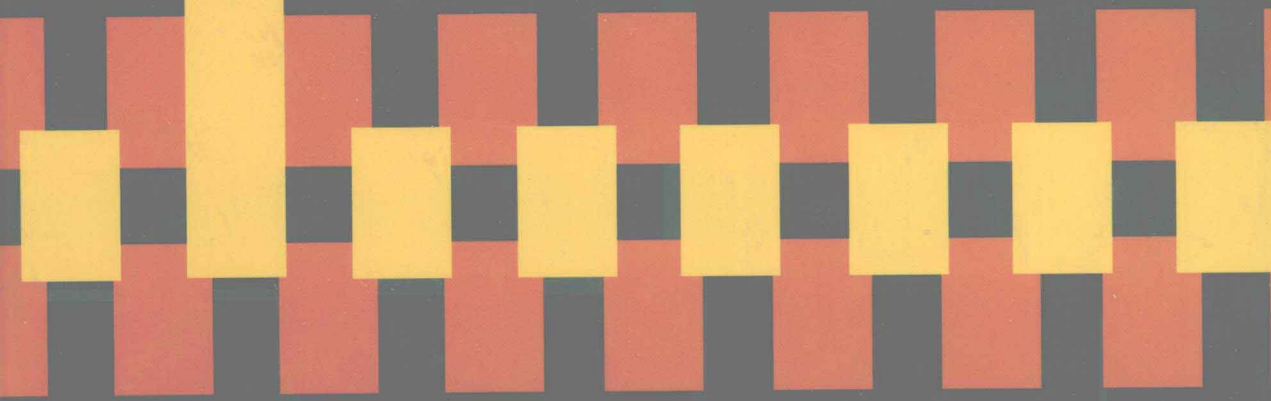
Algebraic Codes for Data Transmission

Richard E. Blahut



Algebraic Codes for Data Transmission

Richard E. Blahut

Henry Magnuski Professor in Electrical and Computer Engineering,
University of Illinois at Urbana – Champaign

CAMBRIDGE
UNIVERSITY PRESS

PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE
The Pitt Building, Trumpington Street, Cambridge, United Kingdom

CAMBRIDGE UNIVERSITY PRESS
The Edinburgh Building, Cambridge CB2 2RU, UK
40 West 20th Street, New York, NY 10011-4211, USA
477 Williamstown Road, Port Melbourne, VIC 3207, Australia
Ruiz de Alarcón 13, 28014 Madrid, Spain
Dock House, The Waterfront, Cape Town 8001, South Africa

http://www.cambridge.org

First published 2003

Printed in the United Kingdom at the University Press, Cambridge

Typefaces Times 10.5/14 pt and Helvetica Neue *System* LaTeX 2_ε [TB]

A catalogue record for this book is available from the British Library

ISBN 0 521 55374 1 hardback

Preface

This book is a second edition of my 1983 book *Theory and Practice of Error Control Codes*. Some chapters from that earlier book reappear here with minor changes. Most chapters, however, have been completely rewritten. Some old topics have been removed, and some new topics have been inserted.

During the two decades since the publication of that first edition, error-control codes have become commonplace in communications and storage equipment. Many such communication and storage devices, including the compact disk, that are now in general use could not exist, or would be much more primitive, if it were not for the subject matter covered by that first edition and repeated in this edition.

The second edition retains the original purpose of the first edition. It is a rigorous, introductory book to the subject of algebraic codes for data transmission. In fact, this phrase, "algebraic codes for data transmission," has been chosen as the title of the second edition because it reflects a more modern perspective on the subject.

Standing alongside the class of algebraic codes that is the subject of this book is another important class of codes, the class of nonalgebraic codes for data transmission. That rapidly developing branch of the subject, which is briefly treated in Chapter 11 of this edition, deserves a book of its own; this may soon appear now that the topic is reaching a more mature form.

This book is a companion to my more advanced book *Algebraic Codes on Lines, Planes, and Curves*. Although both books deal with algebraic codes, they are written to be read independently, and by different audiences. Consequently, there is some overlap in the material, which is necessary so that each book stands alone. I regard the two books as belonging to the general field of *informatics*, an emerging collection of topics that is not quite mathematics and not quite engineering, but topics that form the intellectual bedrock for the information technology that is now under rapid development.

The preparation of the second edition has benefited greatly from the comments, advice, and criticism of many people over many years. These comments came from the reviewers of the first edition, from some readers of the first edition, and from Dr.

Irina Grusko who kindly and capably translated that book into Russian. Critical remarks that helped this second edition came from Professor Dilip Sarwate, Dr. G. David Forney, Professor Joseph A. O'Sullivan, Dr. Weishi Feng, Professor William Weeks IV, Dr. Gottfried Ungerboeck, Professor Ralf Koetter, Professor Steve McLaughlin, Dr. Dakshi Agrawal, and Professor Alon Orlitsky. The quality of the presentation has much to do with the editing and composition skills of Mrs Helen Metzinger and Mrs Francie Bridges. And, as always, Barbara made it possible.

Urbana, Illinois

"Words alone are nothing."

– MOTTO OF THE ROYAL SOCIETY

Contents

1 Introduction

A profusion and variety of communication systems, which carry massive amounts of digital data between terminals and data users of many kinds, exist today. Alongside these communication systems are many different magnetic tape storage systems, and magnetic and optical disk storage systems. The received signal in any communication or recording system is always contaminated by thermal noise and, in practice, may also be contaminated by various kinds of defects, nongaussian noise, burst noise, interference, fading, dispersion, cross talk, and packet loss. The communication system or storage system must transmit its data with very high reliability in the presence of these channel impairments. Bit error rates as small as one bit error in 10^{12} bits (or even smaller) are routinely specified.

Primitive communication and storage systems may seek to keep bit error rates small by the simple expedient of transmitting high signal power or by repeating the message. These simplistic techniques may be adequate if the required bit error rate is not too stringent, or if the data rate is low, and if errors are caused by noise rather than by defects or interference. Such systems, however, buy performance with the least expendable resources: Power and bandwidth.

In contrast, modern communication and storage systems obtain high performance via the use of elaborate message structures with complex cross-checks built into the waveform. The advantage of these modern communication waveforms is that high data rates can be reliably transmitted while keeping the transmitted power and spectral bandwidth small. This advantage is offset by the need for sophisticated computations in the receiver (and in the transmitter) to recover the message. Such computations, however, are now regarded as affordable by using modern electronic technology. For example, current telephone-line data modems use microprocessors in the demodulator with well over 500 machine cycles of computation per received data bit. Clearly, with this amount of computation in the modem, the waveforms may have a very sophisticated structure, allowing each individual bit to be deeply buried in the waveform. In some systems it may be impossible to specify where a particular user bit resides in the channel waveform; the entire message is modulated into the channel waveform as a package, and an individual bit appears in a diffuse but recoverable way.

The data-transmission codes described in this book are codes used for the prevention of error. The phrase "prevention of error" has a positive tone that conveys the true role such codes have in modern systems. The more neutral term, "error-control code," is also suitable. The older and widespread term, "error-correcting code," is used as well, but suffers from the fact that it has a negative connotation. It implies that the code is used only to correct an unforeseen deficiency in the communication system whereas, in modern practice, the code is an integral part of any high-performance communication or storage system. Furthermore, in many applications, the code is so tightly integrated with the demodulation that the point within the system where the errors occur and are corrected is really not visible to any external observer. It is a better description to say that the errors are prevented because the preliminary estimates of the data bits within the receiver are accompanied by extra information that cross-checks these data bits. In this sense, the errors never really happen because they are eliminated when the preliminary estimate of the datastream is replaced by the final estimate of the datastream that is given to the user.

1.1 The discrete communication channel

A communication system connects a data source to a data user through a channel. Microwave links, coaxial cables, telephone circuits, and even magnetic and optical disks are examples of channels. A discrete communication channel may transmit binary symbols, or symbols in an alphabet of size 2^m, or even symbols in an alphabet of size q where q is not a power of 2. Indeed, digital communication theory teaches that discrete channels using a larger symbol alphabet are usually more energy efficient than channels that use a binary alphabet.

The designer of the communication system develops devices that prepare the codestream for the input to the discrete channel and process the output of the discrete channel to recover the user's datastream. Although user data may originate as a sequence of bits, within the communication system it is often treated as a sequence of symbols. A symbol may consist of eight bits; then it is called a *byte*. In other cases, a communication system may be designed around a symbol of r bits for some value of r other than eight; the symbol then is called an r-bit symbol. The choice of symbol structure within the communication system is transparent to the user because the datastream is reformatted at the input and output of the communication system.

A *datastream* is a sequence of data symbols, which could be bits, bytes, or other symbols at the input of an encoder. A *codestream* is a sequence of channel symbols, which could be bits, bytes, or other symbols at the output of an encoder. The user perceives that the datastream is being sent through the channel, but what is actually sent is the codestream.

The encoder maps the datastream into the codestream. Codes are of two types: block codes and tree codes. The distinction between them is based on the way that data

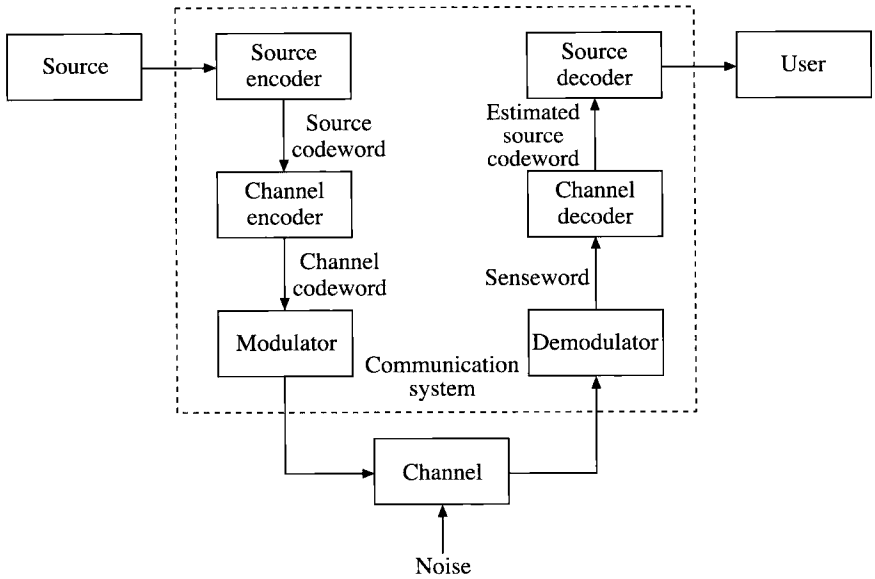


Figure 1.1. Block diagram of a digital communication system

memory is used in the encoder. For constructing the codestream, additional structure is defined on the datastream by segmenting it into pieces called *datawords* or *dataframes*. Likewise, the codestream is segmented into pieces called *codewords* or *codeframes*. The codewords or codeframes are serially concatenated to form the codestream.

It is traditional to partition the major functions of the digital communication system as in the block diagram of Figure 1.1. Data, which enters the communication system from the data source, is first processed by a source encoder designed to represent the source data more compactly. This interim representation is a sequence of symbols called the *source codestream*. The source codestream becomes the input datastream to the channel encoder, which transforms the sequence of symbols into another sequence called the *channel codestream*. The channel codestream is a new, longer sequence that has more redundancy than the source codestream. Each symbol in the channel codestream might be represented by a bit, or perhaps by a group of bits. Next, the modulator converts each symbol of the channel codestream into a corresponding symbol from a finite set of symbols known as the channel alphabet. This sequence of analog symbols from the channel alphabet is transmitted through the channel.

Because the channel is subject to various types of noise, distortion, and interference, the channel output differs from the channel input. The demodulator may convert the received channel output signal into a sequence of the symbols of the channel codestream. Then each demodulated symbol is a best estimate of that code symbol, though the demodulator may make some errors because of channel noise. The demodulated sequence of symbols is called the *senseword* or the *received word*. Because of errors, the symbols of the senseword do not always match those of the channel codestream. The channel decoder uses the redundancy in the channel codestream to correct the errors

in the received word and then produces an estimate of the user datastream. If all errors are corrected, the estimated user datastream matches the original user datastream. The source decoder performs the inverse operation of the source encoder and delivers its output datastream to the user.

Alternatively, some functions of the demodulator may be moved into the channel decoder in order to improve performance. Then the demodulator need not make hard decisions on individual code symbols but may give the channel decoder something closer to the raw channel data.

This book deals only with the design of the channel encoder and decoder, a subject known as the subject of *error-control codes*, or *data-transmission codes*, or perhaps, *error-prevention codes*. The emphasis is on the algebraic aspects of the subject; the interplay between algebraic codes and modulation is treated only lightly. The data compression or data compaction functions performed by the source encoder and source decoder are not discussed within this book, nor are the modulator and the demodulator. The channel encoder and the channel decoder will be referred to herein simply as the encoder and the decoder, respectively.

1.2 The history of data-transmission codes

The history of data-transmission codes began in 1948 with the publication of a famous paper by Claude Shannon. Shannon showed that associated with any communication channel or storage channel is a number C (measured in bits per second), called the *capacity* of the channel, which has the following significance. Whenever the information transmission rate R (in bits per second) required of a communication or storage system is less than C then, by using a data-transmission code, it is possible to design a communication system for the channel whose probability of output error is as small as desired. In fact, an important conclusion from Shannon's theory of information is that it is wasteful to make the raw error rate from an uncoded modulator–demodulator too good; it is cheaper and ultimately more effective to use a powerful data-transmission code.

Shannon, however, did not tell us how to find suitable codes; his contribution was to prove that they exist and to define their role. Throughout the 1950s, much effort was devoted to finding explicit constructions for classes of codes that would produce the promised arbitrarily small probability of error, but progress was meager. In the 1960s, for the most part, there was less obsession with this ambitious goal; rather, coding research began to settle down to a prolonged attack along two main avenues.

The first avenue has a strong algebraic flavor and is concerned primarily with block codes. The first block codes were introduced in 1950 when Hamming described a class of single-error-correcting block codes. Shortly thereafter Muller (1954) described a class of multiple-error-correcting codes and Reed (1954) gave a decoding algorithm for them. The Hamming codes and the Reed–Muller codes were disappointingly weak

compared with the far stronger codes promised by Shannon. Despite diligent research, no better class of codes was found until the end of the decade. During this period, codes of short blocklength were found, but without any general theory. The major advances came when Bose and Ray-Chaudhuri (1960) and Hocquenghem (1959) found a large class of multiple-error-correcting codes (the BCH codes), and Reed and Solomon (1960) and, independently, Arimoto (1961) found a related class of codes for nonbinary channels. Although these remain among the most important classes of codes, the theory of the subject since that time has been greatly strengthened, and new codes continue to be discovered.

The discovery of BCH codes led to a search for practical methods of designing the hardware or software to implement the encoder and decoder. The first good algorithm was found by Peterson (1960). Later, a powerful algorithm for decoding was discovered by Berlekamp (1968) and Massey (1969), and its implementation became practical as new digital technology became available. Now many varieties of algorithms are available to fit different codes and different applications.

The second avenue of coding research has a more probabilistic flavor. Early research was concerned with estimating the error probability for the best family of block codes despite the fact that the best codes were not known. Associated with these studies were attempts to understand encoding and decoding from a probabilistic point of view, and these attempts led to the notion of sequential decoding. Sequential decoding required the introduction of a class of nonblock codes of indefinite length, which can be represented by a tree and can be decoded by algorithms for searching the tree. The most useful tree codes are highly structured codes called *convolutional codes*. These codes can be generated by a linear shift-register circuit that performs a convolution operation on the data sequence. Convolutional codes were successfully decoded by sequential decoding algorithms in the late 1950s. It is intriguing that the Viterbi algorithm, a much simpler algorithm for decoding them, was not developed until 1967. The Viterbi algorithm gained widespread popularity for convolutional codes of modest complexity, but it is impractical for stronger convolutional codes.

During the 1970s, these two avenues of research began to draw together in some ways and to diverge further in others. Development of the algebraic theory of convolutional codes was begun by Massey and Forney, who brought new insights to the subject of convolutional codes. In the theory of block codes, schemes were proposed to construct good codes of long blocklength. Concatenated codes were introduced by Forney (1966), and Justesen used the idea of a concatenated code to devise a completely constructive class of long block codes with good performance. Meanwhile, Goppa (1970) defined a class of codes that is sure to contain good codes, though without saying how to identify the good ones.

The 1980s saw encoders and decoders appear frequently in newly designed digital communication systems and digital storage systems. A visible example is the compact disk, which uses a simple Reed–Solomon code for correcting double byte errors. Reed–Solomon codes also appear frequently in many magnetic tape drives and network

modems, and now in digital video disks. In other applications, such as telephone-line modems, the role of algebraic codes has been displaced by euclidean-space codes, such as the trellis-coded modulation of Ungerboeck (1982). The success of these methods led to further work on the design of nonalgebraic codes based on euclidean distance. The decade closed with widespread applications of data-transmission codes. Meanwhile, mathematicians took the search for good codes based on the Hamming distance into the subject of algebraic geometry and there started a new wave of theoretical progress that continues to grow.

The 1990s saw a further blurring of the walls between coding, signal processing, and digital communications. The development of the notion of turbo decoding and the accompanying codes of Berrou (1993) can be seen as the central event of this period. This work did as much for communications over the wideband channel as Ungerboeck's work did the previous decade for communications over the bandlimited channel. Practical iterative algorithms, such as the "two-way algorithm," for soft-decision decoding of large binary codes are now available to achieve the performance promised by Shannon. The Ungerboeck codes and the Berrou codes, together with their euclidean-space decoding algorithms, have created a body of techniques, still in rapid development, that lie midway between the subjects of modulation theory and of data transmission codes. Further advances toward the codes promised by Shannon are awaited.

This decade also saw the development of algorithms for hard-decision decoding of large nonbinary block codes defined on algebraic curves. Decoders for the codes known as hermitian codes are now available and these codes may soon appear in commercial products. At the same time, the roots of the subject are growing even deeper into the rich soil of mathematics.

1.3 Applications

Because the development of data-transmission codes was motivated primarily by problems in communications, much of the terminology of the subject has been drawn from the subject of communication theory. These codes, however, have many other applications. Codes are used to protect data in computer memories and on digital tapes and disks, and to protect against circuit malfunction or noise in digital logic circuits.

Applications to communication problems are diversified. Binary messages are commonly transmitted between computer terminals, in communication networks, between aircraft, and from spacecraft. Codes can be used to achieve reliable communication even when the received signal power is close to the thermal noise power. And, as the electromagnetic spectrum becomes ever more crowded with man-made signals, data-transmission codes will become even more important because they permit communication links to function reliably in the presence of interference. In military applications, it often is essential to employ a data-transmission code to protect against intentional enemy interference.