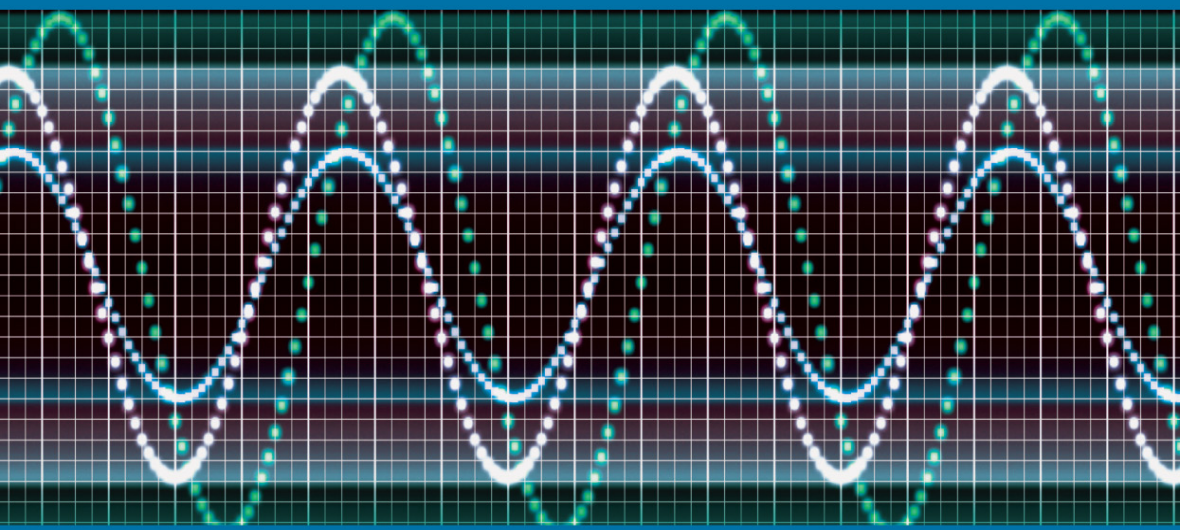
FOCUS

DIGITAL SIGNAL AND IMAGE PROCESSING SERIES



Nonlinear Digital Encoders for Data Communications

Călin Vlădeanu and Safwan El Assad

iSTE WILEY

Nonlinear Digital Encoders for Data Communications

FOCUS SERIES

Series Editor Francis Castanié

Nonlinear Digital Encoders for Data Communications

Călin Vlădeanu
Safwan El Assad

ISTE

WILEY

First published 2014 in Great Britain and the United States by ISTE Ltd and John Wiley & Sons, Inc.

ISTE Ltd
27-37 St George's Road
London SW19 4EU
UK

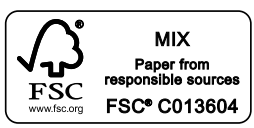
www.iste.co.uk

John Wiley & Sons, Inc.
111 River Street
Hoboken, NJ 07030
USA

www.wiley.com

Library of Congress Control Number: 2013956557

British Library Cataloguing-in-Publication Data
A CIP record for this book is available from the British Library
ISSN 2051-2481 (Print)
ISSN 2051-249X (Online)
ISBN 978-1-84821-649-5



Printed and bound in Great Britain by CPI Group (UK) Ltd., Croydon, Surrey CR0 4YY

Table of Contents

Preface

Several applications using nonlinear dynamical systems have been developed over the last few decades. Among these, the field of telecommunications has clearly benefited from these nonlinear blocks. Hence, the nonlinear blocks have proved to be suitable for implementing several telecommunications techniques, such as encryption, spectrum spreading and channel coding.

In this book, we present novel solutions for channel coding using nonlinear sequence generators. To the best of our knowledge, there are only a few works that attempt to propose the use of nonlinear functions for channel coding in telecommunications systems. In this context, this book aims to demonstrate that nonlinear encoders can be designed to provide good performances and to encourage researchers to investigate these approaches further.

This book contains many numerical examples that complete the description of the analyzed schemes. Also, some performance simulation results are provided. Some sections include presentations of the mathematical apparatus used throughout the book and some Matlab/Simulink scripts and schemes used to run the simulations.

We recommend this book to students, especially for Master's and PhD students who will easily understand the concepts presented in this book and can design and test these schemes by simulations.

We also consider that this book may be used by telecommunications engineers to complete their grounding in the field of signal processing for telecommunications, especially on non-conventional and nonlinear coded modulation techniques.

Călin VLĂDEANU
Safwan EL ASSAD
January 2014

Introduction

During the last few decades, several chaotic sequence generators have been investigated for secure and efficient digital communication systems. Because of their generators' sensitive dependence on the initial state, these sequences present pseudo-random properties and offer an enhanced security.

As is well known, chaotic sequence generators are nonlinear dynamical systems and their finite precision or quantized approximations affect the chaotic regimes. Hence, this problem can be overcome by developing digital generators. In [FRE 93], Frey proposed a chaotic digital infinite impulse response (IIR) filter for a secure communications system. The Frey filter contains a nonlinear function called the left-circulate (LCIRC) function, which provides the chaotic properties of the filter. The LCIRC function performs a bit left circulation over the N bits representation word. In the same paper, Frey showed that this nonlinear recursive filter possesses quasi-chaotic properties, both for autonomous and non-autonomous modes. In [WER 98], Werter improved this encoder in order to increase the randomness between the output sequence samples. The performances of a pulse amplitude modulation (PAM) communication system using the Frey encoder, with additive white Gaussian noise (AWGN), were analyzed in [AIS 96]. A modified state feedback decoder was proposed in [AIS 96] and performs better than the Frey inverse filter decoder, in terms of bit error rate (BER) at a high signal-to-noise ratio (SNR).

In all these previous works, the Frey encoder has been considered as a digital filter, operating over Galois field GF$\left(2^N\right)$ and was used to increase the security of the transmissions. During the last few decades, the nonlinear

functions have been used extensively in chaotic sequence generators to increase the security of communications systems.

Nowadays, channel-encoded transmissions are used in all systems. Several types of channel encoding methods have been proposed during the last few decades. Almost all coding methods known in the literature use linear functions. Barbulescu *et al*. made one of the first approaches regarding the possible use of the Frey encoder in a turbo-coded communication system [BAR 00]. Nevertheless, in [BAR 00], the authors only mentioned as an advantage the intrinsic randomness of the encoder, which eliminates the use of an interleaver from the typical turbo encoding/decoding schemes. Despite the promising idea, this book did not consider any information theory approach for performance evaluation and, above all, it did not prove the advantages of these turbo encoders. In [ZHO 01], Zhou *et al.* made a similar analysis and again, the paper lacks proof for the stated performance enhancement. Another more recent work [NG 08] addressed different methods for using chaotic sequence generators to enhance the coding gain or the security of several coded schemes. More recently, in [ESC 09], a turbo trellis-coded modulation (TTCM) scheme using digital chaotic encoders with binary inputs and chaotic outputs is proposed. This work and the references therein introduce a different family of nonlinear encoders than the encoders analyzed in this book. However, we consider that the work of Escribano *et al.* [ESC 09] presents many similarities to our work, especially in the encoders' design and performance analysis solutions.

In [VLA 09a], a different perspective was offered for the chaotic digital encoder proposed by Frey in [FRE 93]. Mainly, it was observed that this encoder with finite precision possesses a trellis that describes its deterministic functioning. This led to the possibility of improving the performances of the chaotic sequence transmission over a noisy channel by using sequence detection. In fact, this is the reason why this encoder should perform well in turbo coding schemes. To demonstrate this performance improvement as compared to the non-encoded system, a new TCM scheme was developed for obtaining better performance in the presence of noise. This method partially follows the rules proposed by Ungerboeck in [UNG 82] for defining optimum TCMs by proper set partitioning (SP). The main idea is to use a different word length in the output as compared to the input. In [CLE 06], Clevom *et al.* developed a method for separating a recursive systematic convolutional (RSC) encoder into subencoders with only a single delay element. This

equivalency makes a GF(2) RSC equivalent to a simpler RSC that works inside over a higher order field, while its input and output still work over GF(2). Even if a different problem was addressed in [CLE 06], this idea led to the fact that different representation word lengths can be used in the input and output, and a higher order field nonlinear encoder is equivalent to a linear GF(2) RSC encoder. Therefore, in [VLA 09a], it was demonstrated that the Frey encoder with finite precision (word length of N bits) presented in [FRE 93] is a recursive convolutional (RC), but non-systematic, encoder operating over GF$\left(2^N\right)$. In the same work [VLA 09a], a new method is proposed for enhancing the performances of the chaotic PAM-TCM transmission over a noisy channel.

A generalization of the optimum one-delay GF(4) encoder in [VLA 09a] was made, for any output word length N and for any possible encoding rate, in the cases of PAM-TCM [VLA 09b] and phase-shift keying TCM (PSK-TCM) [VLA 10b] transmission schemes. The development of optimum GF$\left(2^N\right)$ encoders for the quadrature amplitude modulation (QAM) TCM scheme is more difficult than in the case of PAM and PSK modulations, due to the larger constellations and non-uniform power per symbol. In [VLA 11b], a generalization of the optimum one-delay GF(4) encoder in [VLA 09a] is performed, for any output word length N and for any possible encoding rate in QAM-TCM schemes. These encoders follow the rules proposed by Ungerboeck [UNG 82] for defining optimum TCM by proper SP. However, all the previously mentioned encoders are non-systematic, making them unfeasible for TTCM schemes.

Two-dimensional (2D) TCM schemes using a different trellis optimization method for Frey encoder were proposed in [VLA 09c]. Hence, the filter scheme is modified to have an additional output, which transforms it into a rate $1/2$ encoder. The second output is derived in such a manner as to transmit a 2D TCM signal, following the Ungerboeck SP rules [UNG 82]. Despite these changes, the filter representation word length is not changed.

The coding gain was estimated theoretically for all the considered encoders, for different values of N. In fact, exact expressions of the minimum Euclidean distance were determined for all these TCM schemes. Frey encoders with small representation word length ($N = 1$ and $N = 2$) were considered for simulations. There are two reasons for simulating encoders with small word lengths. First, the trellis size increases exponentially with N;

second, the coding gain reduces when the word length increases. This last property is explained by the fact that the increase in the constellation size decreases the signal minimum Euclidean distance, more than the encoding can cope with. It is also noted that the linear encoders corresponding to the considered nonlinear encoders (obtained by eliminating the nonlinear block) do not have good trellises. For all these schemes, the simulated performances confirm the theoretically determined Euclidean distances.

The turbo coding scheme introduced by Berrou and Glavieux in their seminal paper [BER 96] allows communications systems' performances close to the Shannon limit, by concatenating in parallel RC encoders in the transmitter and using iterative decoding algorithms in the receiver. Turbo schemes were developed for the TCM schemes as well [OGI 01, ROB 98].

In [PAU 10a], the RC-LCIRC encoder from [VLA 10b] is adapted for, and introduced into, a parallel turbo PSK-TCM transmission scheme, and the performances of this scheme are analyzed in case of transmitting over a channel with AWGN. Similarly, in [PAU 10b], the same RC-LCIRC is introduced into a parallel turbo QAM. The QAM-TCM transmission scheme and the performances of this scheme are analyzed in case of transmitting over a channel with AWGN. The performances of the RC-LCIRC-TTCM schemes introduced in [PAU 10a] and [PAU 10b] were analyzed in [VLA 10a] assuming a transmission over a non-selective Rayleigh fading channel.

In [VLA 11a], an improved version of the RC-LCIRC encoder from [VLA 10b] is proposed. The main encoder improvement consists of making it systematic. In fact, this was the only disadvantage of the previous LCIRC encoders which were not fully compatible with the corresponding binary encoders. Further to this new advantage, the encoder designing process aimed to keep all previous advantages of the LCIRC encoders, such as optimum performances in terms of Euclidean distance, the reduced complexity in the memory usage (for any encoding rate, only one delay element is used) and a compact expression of the Euclidean distance for a specific modulation. The optimum SP method is used both for PSK and QAM-TCM schemes. The symbol error rate (SER) performances of these new schemes are analyzed in case of transmitting over a channel with AWGN. Corresponding binary encoders, i.e. with the same values for the encoding rate, the number of trellis states and the minimum Euclidean distance, are considered as a reference for SER comparison.

A family of nonlinear encoders for the TTCM scheme was analyzed in [VLA 11c]. The systematic encoders introduced in [VLA 11a] were adapted for a parallel TTCM transmission scheme. As compared to the TTCM scheme analyzed in [PAU 10b] operating at low coding rates due to the lack of puncturing, the work in [VLA 11c] introduces specific interleaving and puncturing methods for the TTCM scheme. Moreover, the conventional logarithmic maximum *a posteriori* probability (log-MAP) decoding algorithm is modified to operate in a symbol-by-symbol manner for punctured received sequences, following the method presented in [ROB 98] and [VUC 00]. The optimum SP method is used for PSK punctured TTCM schemes. The performances of this scheme are analyzed in case of transmitting over a channel with AWGN.

The work in [VLA 12] extends the performance analysis for these nonlinear TTCM schemes. The extrinsic information transfer (EXIT) chart is an important tool for visualizing the exchange of the extrinsic information between constituent decoders in a turbo receiver scheme [TEN 01]. Moreover, the EXIT charts are presented to underline the convergence behavior of these schemes. The EXIT chart was also applied to TTCM schemes to depict the decoding trajectory, allowing the prediction of BER waterfall and BER floor regions [CHE 04, KLI 06]. Two channel models were considered for the performance analysis, i.e. the AWGN noisy channel and the Rayleigh fading channel. Also, the considered multilevel modulation techniques were PSK and QAM. Therefore, the EXIT chart can be used as a tool in the design of TTCM schemes [NG 08].

The book is organized as follows. In Chapter 1, the nonlinear digital encoders are introduced and their main applications for telecommunications are presented and discussed. In section 1.1, the general nonlinear digital encoder scheme is presented, as it was introduced in the literature for secure communications applications. This structure generates quasi-chaotic sequences that are suitable for such secure communications applications. Here, an example of a nonlinear codec introduced by Frey is analyzed and simulations reveal its quasi-chaotic features. In section 1.2, the possible use of these structures in spread-spectrum applications is briefly discussed. A major problem arising from the use of discrete-time nonlinear systems in telecommunications is the sequence synchronization in the receiver part. This issue is addressed in section 1.3 for the particular case of the Henon discrete-time map. Here, an efficient method for sequence synchronization,

i.e. the dead-beat method, is presented and analyzed both theoretically and by means of simulations.

Chapter 2 is dedicated to the presentation of the Frey nonlinear encoder introduced in section 1.1. To be more specific, in section 2.1, the Frey encoder is mathematically described using the modulo-2^N operators. The properties of these operators are thoroughly analyzed in section 2.2. The demonstrations for some of the theorems given in section 2.2 are presented in the appendix. Using the operators introduced in section 2.2, the properties of the LCIRC function are demonstrated in section 2.3. Finally, taking advantage of the mathematical development from the previous sections, the Matlab/Simulink simulation block schemes for the Frey digital encoder, decoder, including the dead-beat synchronization method (described in section 1.3), are presented in section 2.4.

Chapter 3 presents the Frey encoder from a different perspective, i.e. as a convolutional encoder. In section 3.1, the original Frey encoder is analyzed as a convolutional encoder and it is shown that it is not efficient for channel coding. Two new PAM-TCM schemes for the Frey encoder are introduced in section 3.2. First, the trellis optimization method by reducing the input representation word length is presented for a particular case, and then it is generalized for any output word length value. Hence, a generalized optimum nonlinear RC encoder scheme is proposed and an expression is provided for the minimum Euclidean distance of these encoders in a PAM-TCM transmission. Also, the second PAM-TCM modified Frey scheme is presented, which optimizes the coding performances by increasing the number of outputs, without modifying the input and output word lengths. Using a similar input word length optimization, a generalized optimum GF$\left(2^N\right)$ RC encoder scheme is proposed and an expression is provided for the minimum Euclidean distance of these encoders for a PSK-TCM transmission in section 3.3 and for a QAM-TCM transmission in section 3.4. The simulated SER performance is presented in section 3.5 for the optimum PAM-TCM, PSK-TCM and QAM-TCM transmissions.

Chapter 4 analyzes the parallel TTCM schemes including convolutional LCIRC encoders. First, in section 4.1, the RC-LCIRC encoder introduced in Chapter 3 is used as a constituent encoder for non-punctured parallel PSK and QAM-TTCM schemes. A multilevel log-MAP algorithm is used for the iterative detection. The non-systematic nature of the RC-LCIRC encoder and

its effect on the TTCM scheme performances are pointed out. Therefore, section 4.2 presents the new generalized RSC-LCIRC encoder operating over Galois field GF$\left(2^N\right)$ and the optimum SP for two-dimensional TCM schemes. In section 4.3, a parallel TTCM transmission scheme using RSC-LCIRC component encoders with symbol puncturing is presented. A symbol-by-symbol log-MAP algorithm is used for the iterative detection. For the constituent RSC-LCIRC encoders, the minimum Euclidean distance, minimum effective length and product distance are estimated for the AWGN channel and the Rayleigh fading channel, respectively. The method used to plot the EXIT charts of these TTCM schemes is described in section 4.4. The simulated BER performance graphs and EXIT charts for the TTCM schemes introduced in previous sections are presented in section 4.5. First, the simulated BER performance over AWGN and Rayleigh fading channels is plotted for the 8-PSK-TTCM and 16-QAM-TTCM transmission schemes from section 4.1, using three types of mappings: Gray, natural and SP. The coding gains of these schemes using different mappings, as compared to the uncoded modulation and the non-iterative schemes, are derived from simulations. Next, the SER performances are plotted for optimum 8-PSK and 16-QAM RSC-LCIRC-TCM transmissions over an AWGN channel, using the encoder introduced in section 4.2. Finally, the simulated BER performances in AWGN and Rayleigh fading channels are presented and analyzed for the punctured 8-PSK and 16-QAM-TTCM transmission using three different interleaver sizes. Also, the EXIT charts are plotted to compare the convergence of different TTCM decoding schemes for the punctured 8-PSK and 16-QAM-TTCM transmissions.

1

Applications of Nonlinear Digital Encoders

1.1. Secure communications using nonlinear digital encoders

Recently, interest has been growing in the use of chaos for secure communications. Spread-spectrum communications are included in the category of secure communications, which also stand to benefit from this research.

Using analog chaotic systems for secure communications schemes presents some drawbacks for practical applications, since all show an intrinsic weak robustness, i.e. the unavoidable errors on the values of the circuit components, as well as the disturbances generated by the communication link, can heavily influence the synchronization process and can make it ineffective.

In general, three communication schemes have been proposed so far: chaotic switching where the information is binary and switches the transmitted signal between two attractors, chaotic masking where the information is simply added to the chaotic signal and chaotic modulation where the information is "modulated" on a chaotic carrier by means of an invertible nonlinear transformation. Researchers have shown experimental prototype implementations of secure communication systems utilizing chaos. They capitalize upon the fact that chaotic circuits taken from an appropriate class can be made to synchronize. Specifically, it has been shown [PEC 90] that if a chaotic system can be divided into subsystems with stable Lyapunov exponents, then it will asymptotically track a replica of itself. Kocarev *et al.* [KOC 92] and Cuomo and Oppenheim [CUO 93] demonstrate the fact that this tracking phenomenon is robust enough to allow locking to occur even in

the presence of a continuous perturbation. In particular, they add a small (12 dB down in [KOC 92] and 20 dB down in [CUO 93]) message signal to the chaotic signal produced by a first chaotic circuit taken from the class described above. This transmitted signal is then relayed to the receiver, where an identical chaotic circuit locks onto the dominant chaotic component. When a lock is achieved, the chaotic replica of the original chaotic signal is subtracted from the received composite signal, leaving the message signal.

A variation is studied in [CUO 93] and [PAR 92], where the message signal is a binary signal causing a parameter in the transmitting chaotic circuit to take on two possible values, thereby producing a modulated chaotic output. The receiver replica has the respective parameter fixed to one of the possible values in its counterpart. As a result, it tracks the transmitter anytime the binary input is one state and falls out of sync at times corresponding to the other input state. Lock and unlock conditions are easily detected, resulting in proper demodulation. An even more intriguing variation is proposed by Halle *et al.* [HAL 93], which is most closely related to the work presented in the following, although it is analog. In [HAL 93], Chua's circuit is driven by an external input that is a modulated version of the message signal. The modulation is achieved via a nonlinear operation with the chaotic output of the circuit. Then, this output is fed to a copy of the chaotic circuit at the receiver end, which synchronizes with the transmitter circuit and contains the inverse modulation function. It is demonstrated that this nonlinear modulation of the input with the chaotic signal produces a system whose encoded signal is quantitatively chaotic and that is sensitive to parameter mismatch between the receiver and the transmitter, two qualities that are highly desirable for use in secure communications applications. The benefit of the encoding (or modulation) schemes outlined above is that the transmitted signal is substantially chaotic to an observer. Hence, in general, it would bear little resemblance to the message signal, thereby producing a measure of security in the transmission of the data. While all these approaches are fascinating and surely warrant further investigation, they may suffer from serious drawbacks in practice. First, since two matched analog chaotic circuits are required at remote locations, in practice there can be serious problems with system performance unless a method of calibration is devised. On the other hand, the robustness of the tracking phenomena should allow some mismatch to occur while providing acceptable performance. However, this robustness could be exploited by an unintended listener diminishing the security of the system.

Another drawback to the additive approach [CUO 93, KOC 92], i.e. where the message signal is added to the chaotic signal, is that the signal-to-noise ratio of the received signal is directly degraded by the chaos-to-signal ratio (12 dB in [KOC 92] and 20 dB in [CUO 93]) relative to available channel signal-to-noise ratio. Furthermore, adaptive filtering techniques may allow an intruder to find the information in the chaotic noise, despite its overwhelming magnitude. Alternatively, in the binary modulation approach, it may be possible to detect the binary signal without locking onto the chaotic signal directly, thereby compromising security.

1.1.1. *The general nonlinear digital encoder scheme*

In this section, we present the secure communications system proposed by Frey [FRE 93] that has all the advantages of any other chaos-based system, but also has other advantages, which are not present in previously proposed schemes. First, the system is digital, which makes a perfect reconstruction in the receiver of the transmitted signal possible. Second, the system introduced by Frey produces transmitted signals with virtually no correlation to the input, while being substantially chaotic in appearance.

The block scheme of the general encoder proposed by Frey is shown in Figure 1.1. The scheme of the corresponding decoder is shown in Figure 1.2.

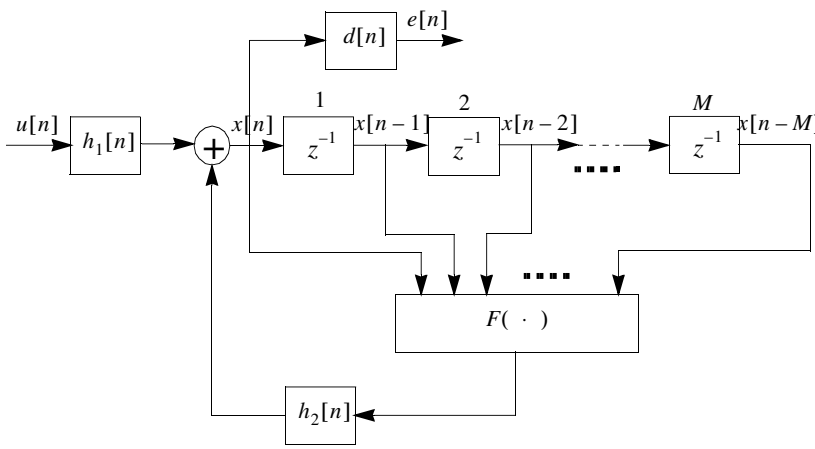


Figure 1.1. *The general digital encoder scheme*

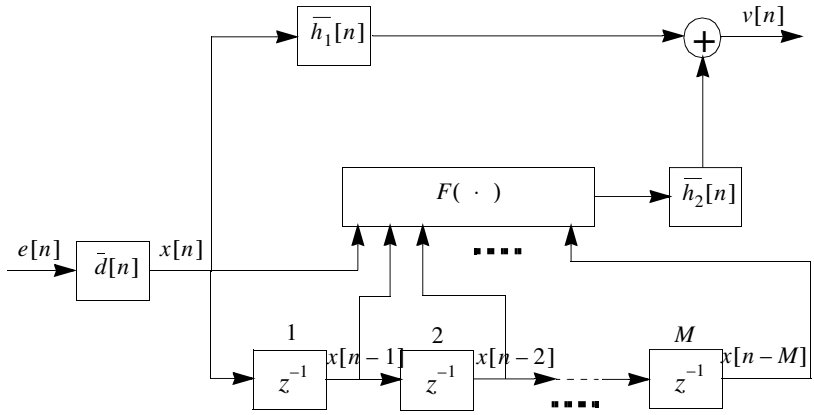


Figure 1.2. *The general digital decoder scheme*

The class of coders/decoders analyzed here are generally defined by the following equations. Hence, the encoder is defined by:

$$\begin{aligned} x[n] &= h_1[n] \bullet u[n] \oplus h_2[n] \bullet F\left(x[n], x[n-1], \ldots, x[n-M]\right) \\ e[n] &= d[n] \bullet x[n] \end{aligned} \quad [1.1]$$

The corresponding decoder is defined by:

$$\begin{aligned} x[n] &= \overline{d}[n] \bullet e[n] \\ v[n] &= \overline{h_1}[n] \bullet x[n] \oplus \overline{h_2}[n] \bullet F\left(x[n], x[n-1], \ldots, x[n-M]\right) \end{aligned} \quad [1.2]$$

where $u[n]$ is the input sequence (message signal), $x[n]$ is an internal signal, $e[n]$ is the encoded sequence to be transmitted to the receiver and $v[n]$ is the final response sequence (decoded message signal; ideally, $v[n]$ should be the same as $u[n]$). The impulse responses $h_1[n]$, $h_2[n]$ and $d[n]$ are generally infinite impulse response (IIR), but one or more can be finite impulse response (FIR) or trivial. The nonlinear map, $F(.)$, is a general nonlinear map suited to hardware implementation and its output signal depends on the internal signal $x[n]$ and on M delayed versions of $x[n]$. For all these signals, the discrete-time variable is denoted as n. The addition and convolution operators, denoted as $\bullet$ and $\oplus$, are assumed to be generally nonlinear operations due to overflow in the actual finite length adders used for hardware computation. Some of these operators will be presented in detail in Chapter 2.

The defining equations given by [1.1] allow for a very wide variety of systems. Note that the decoder must be a nonlinear filter that implements the inverse of the encoder for proper recovery of the input sequence, $u[n]$). This explains why $x[n]$ is assumed to be exactly recovered in the decoder as an internal variable. The $\overline{h_1}[n]$, $\overline{h_2}[n]$ and $\overline{d}[n]$ blocks in the decoder scheme are considered as complementary to those in the encoder scheme (in the sense of the convolution operator, i.e. $\overline{d}[n] \bullet d[n] = \delta[n]$, where $\delta[n]$ is the Dirac pulse). No general results, specific to the quasi-chaotic (QC) properties, on this class of systems is available in the literature. While Chua and Lin [CHU 88, CHU 90] have given a very clever construction to prove the existence of chaotic trajectories in second- and third-order filters, the discrete nature of the systems considered in this work, coupled with the complexity associated with non-autonomous responses, unfortunately seems to preclude any direct application of their method to the case discussed here. In an earlier work [EBE 69], Ebert *et al.* have also suggested a method intuitively similar to the method in [CHU 88, CHU 90], but their results are aimed at basically stable filters and the appearance of limit cycles. A mathematical analysis of the basic encoder of this chapter is presented in section 2.1, which draws upon the work in [CHU 88, CHU 90] and [EBE 69], but only qualitative results are available at this time. The results shown in this work hopefully make the pursuit of a complete theory more compelling.

1.1.2. *Quasi-chaotic sequence properties*

The presence of chaotic regimes in digital filters has been demonstrated by Chua and Lin [CHU 88, CHU 90]. In these works, it was shown that the overflow nonlinearity determines an otherwise linear digital filter to behave chaotically with certain initial conditions. The authors showed that the filters present the chaotic regimes, even in the more realistic situation where finite word length is considered [LIN 91]. This fact is important, since, technically, all finite precision digital filters must possess periodic autonomous responses, thereby precluding the possibility of chaos. Nevertheless, the results in [LIN 91] give credence to the idea that practical digital filters can be essentially chaotic.

In fact, the regimes analyzed in [CHU 88, CHU 90, LIN 91] and [FRE 93] are not truly chaotic, because the digital filters possessing them work in finite precision. In [FRE 93], Frey introduces for the first time in the literature a

definition for a "quasi-chaotic behavior" by proposing a set of conditions that must be met by the filter response. This set of conditions, which is consistent with the common observations about chaos, will be used as the criterion for whether a chaotic response has been generated. This definition of a QC digital filter is provided below.

DEFINITION 1.1.– *A QC digital filter is defined as a non-autonomous digital filter that possesses the following properties [FRE 93]:*

1) the zero input response has a wideband noiselike spectrum for almost all choices of initial condition. Under the same conditions, the autocorrelation function of the response is similar to an uncorrelated noise sequence;

2) the response of the filter to arbitrary inputs has a broadband noiselike spectrum for almost all choices of initial conditions. Under the same conditions, the autocorrelation function of the response is similar to an uncorrelated noise sequence;

3) the response of the filter to almost all arbitrary inputs is uncorrelated to the input for almost all choices of initial conditions;

4) the responses of the filter to the same input sequences are uncorrelated to one another for almost all choices of different initial conditions;

5) for almost all choices of input to two identical filters having different but arbitrarily close initial states, the states of the two filters will diverge.

The expression "almost all" used in properties 1–5 refers to the vast majority (e.g. greater than 90% or 95%). Also, the term "noiselike" implies noise similar to a filtered version of white Gaussian noise. The autocorrelation of the filter response has a maximum value in the origin, and very low values elsewhere, while the period of the autocorrelation function depends on the length of cycles in the filter solution trajectories. Also, the correlation (intercorrelation) between two different responses, generated with different initial conditions, takes low values as compared to the maximum in the autocorrelation.

The empirically defined properties 1–5 [FRE 93] are useful for testing the chaotic behavior for any digital filter. Moreover, these properties are important for guaranteeing the security of a communication system using the chaotic filters possessing them. Even these properties were determined

empirically, an important number of tests were performed by simulation to check the validity of the properties [FRE 93]. Therefore, the conclusion is that the QC properties do hold for even relatively simple filters. To verify the generality of these properties, we ran a significant number of tests to check them, following the same procedure as in [FRE 93]. The simulation results of such tests confirm the validity of the QC properties. These results are presented in section 1.1.4.

1.1.3. *An example of simple nonlinear digital encoder: the Frey chaotic encoder*

In [KAC 92], Frey and Kaczmarczyk proposed a nonlinear digital filter working in finite precision for data encoding. The digital Frey codec is shown in Figure 1.3. It can be easily seen that the structure in Figure 1.3 is a particular case of the encoder/decoder general scheme shown in Figures 1.1 and 1.2, where $\overline{d}[n] = d[n] = \delta[n]$, $h_1[n] = \overline{h_1}[n] = \delta[n]$, $M = 2$, $h_2[n] = -\overline{h_2}[n] = \delta[n]$ and $F\left(x[n], x[n-1], x[n-2]\right) = LCIRC\left(x[n-2]\right)$. So, the output signal is given by $e[n] = x[n]$.

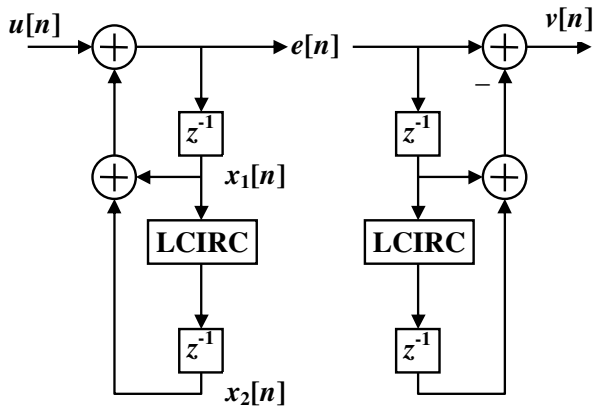


Figure 1.3. *Frey codec*

The encoder represented in the left side of Figure 1.3 is an IIR digital filter, while the decoder on the right side of Figure 1.3 is a nonlinear digital filter that implements the inverse of the encoder block. This inverse system is used for exact decoding of the original data in the absence of the noise [FRE 93]. The

first thing to note is that in real transmission systems, we always have noise and a more realistic study is necessary assuming the presence of the noise. It is also important to note that there is no internal feedback path in this decoder. Specifically, the decoder is an FIR nonlinear digital filter, which guarantees that an error in the communications channel, which can cause at most a finite burst error in the output $e[n]$.

Specifically, the system of Figure 1.3 follows these equations:

$$\begin{aligned} e[n] &= u[n] \oplus \{e[n-1] \oplus f\,(e[n-2])\} \\ v[n] &= e[n] \ominus \{e[n-1] \oplus f\,(e[n-2])\} \end{aligned} \qquad [1.3]$$

Let us assume that N denotes the word length used for binary representation of each sample. The encoder in Figure 1.3 is composed of two delay elements with a sample interval, two modulo-2^N adders and a nonlinear function called left-circulate (LCIRC), which is a function that is typically available as a basic accumulator operation in microprocessors. For further details on these modulo-2^N operators, see section 2.2, while the LCIRC function properties are presented in section 2.3. In [1.3], the function $f\,(.)$ is the LCIRC function. For each sample moment n, $u[n]$ represents the input data sample, $x_1[n]$ and $x_2[n]$ denote the delay elements' states and $e[n]$ is the output sample. The superscript U denotes that all the samples are represented in unsigned form, using N bits per word, i.e. $u^U[n]$, $e^U[n] \in \left[0, 2^N - 1\right]$.

The nonlinear LCIRC function is responsible for the QC behavior of this encoder. Considering the unsigned modulo-2^N operations, the LCIRC function is defined in the following.

DEFINITION 1.2.– *The LCIRC function consists of a multiplication by* 2 *plus the carry bit. The LCIRC function is defined in the* 2^N*-set, by the following equation [FRE 93]:*

$$y^U[n] = LCIRC\left(x^U[n]\right) = \left(2\dot{x^U}[n] + s[n]\right) mod 2^N \qquad [1.4]$$

In [1.4], $s[n]$ denotes the carry bit, which is given by:

$$s[n] = \begin{cases} 0 & \text{if} \quad 0 \le x^U[n] \le 2^{N-1} - 1 \\ 1 & \text{if} \quad 2^{N-1} \le x^U[n] \le 2^N - 1 \end{cases} \qquad [1.5]$$

where $x^U, y^U \in \left[0,\, 2^N - 1\right]$.

The LCIRC function defined by [1.4] and [1.5] is equivalent to the following operations performed over the binary word unsigned representation: denoting by N the word length used for binary representation of each sample, the LCIRC function performs a bit rotation by placing the most significant bit to the less significant bit and shifting the other $N-1$ bits one position to a higher significance. This is the reason why the function is called LCIRC. Analyzing the expression [1.5], we can note that besides the nonlinearity in the modulo-2^N multiplications and additions, the carry bit $s[n]$ also determines the nonlinearity of the LCIRC function, being related by a nonlinear inequality with the input sample value.

In [KAC 92] and [FRE 93], it was demonstrated that the structure shown in Figure 1.3, which is defined by [1.3]–[1.5], possesses the QC properties presented in section 1.1.2. This was achieved by running extensive simulations for a very large number of initial conditions. It was observed that with a constant input, the output was qualitatively chaotic for almost all sets of initial states in the delays. Chaos was qualitatively determined by verifying the QC properties introduced in section 1.1.2.

1.1.4. *Simulation results revealing the quasi-chaotic properties for the sequences generated using the Frey encoder*

In this section, the QC properties of the system as shown in Figure 1.3 are verified following the same procedure as in [FRE 93]. First, it was noted that the richness of the dynamics is directly related to the word length. As in [FRE 93], the simulation results are determined for an 8-bit word length, which represents a compromise between complexity and the computational burden of simulating the system over a wide variety of test conditions. To investigate QC property 1, the zero input response was observed for all possible choices of initial states. Almost all ($> 98\%$) states were part of cyclic responses of a period greater than 100. The longest cycle included $37,749$ states and there were six other distinct cycles covering more than $1,000$ states, namely $7,063$, $6,594$, $3,392$, $2,601$, $1,343$ and $1,260$. Between length 100 and $1,000$ were five cycles of length 116, two cycles of length 384 and distinct cycles of length 715, 554, 544, 400, 378, 242 and 156. The signal spectrum is determined using the discrete Fourier transform (DFT). The DFT

spectrum and the autocorrelation functions of the encoded signal, $e[n]$, corresponding to all cycles greater than length 100, were computed over a 256-point window. To avoid the large DC component in the sequences, in all computations performed, the integers were assumed to be in 2's complement form, yielding positive and negative integers with close-to-zero mean value.

For example, let us consider the cycle of $1,260$ samples (for the initial states $x_1[0] = 245$ and $x_2[0] = 93$) in the unsigned form. Computer simulations were run, using the Frey encoder from Figure 1.3, and the following parameters were determined for the above-mentioned cycle: the time domain representation of the encoded signal corresponding to a zero input is depicted in Figure 1.4, the DFT spectrum for the zero input response from Figure 1.4 is shown in Figure 1.5 and the autocorrelation function for the same signal is shown in Figure 1.6. Even a particular cycle was considered, these results are representative of all cases. Analyzing the autocorrelation values in Figure 1.6, it results that the signal is clearly noiselike.

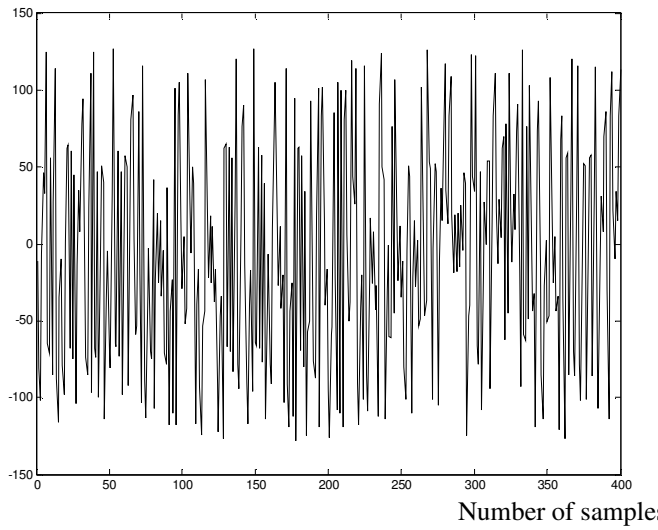


Figure 1.4. *The time domain representation of the encoded signal corresponding to a zero input*

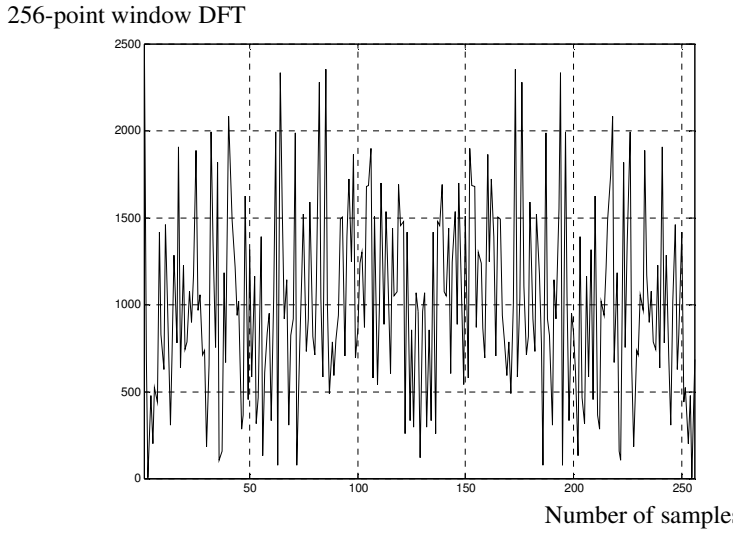


Figure 1.5. *DFT spectrum for a zero input response*

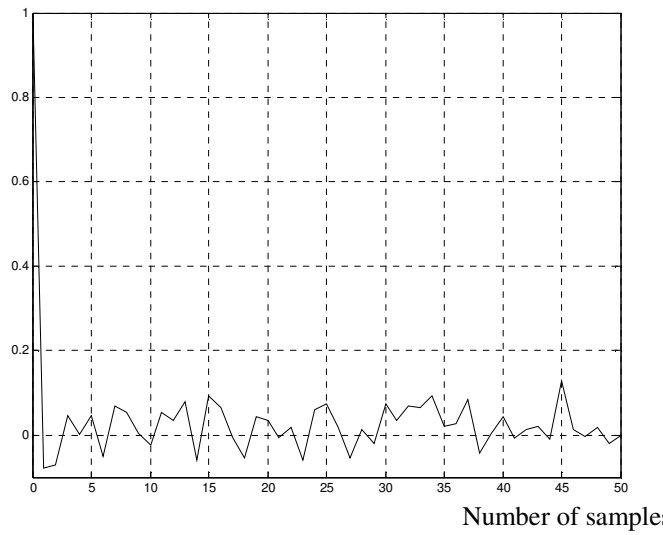


Figure 1.6. *Normalized autocorrelation function for a zero input response*

Moreover, the cross-correlation function between pairs of encoded signals, corresponding to different cycles, was also computed for a variety of cases. It was found that encoded signals corresponding to different length cycles were essentially uncorrelated. Signals having small (theoretically null) cross-correlation values are called *orthogonal*. Figure 1.7 shows a typical case where the encoded signals correspond to cycles of $1,260$ and $1,343$. The cycle of $1,343$ was generated considering the initial states $x_1[0] = 243$ and $x_2[0] = 246$ in the unsigned form.

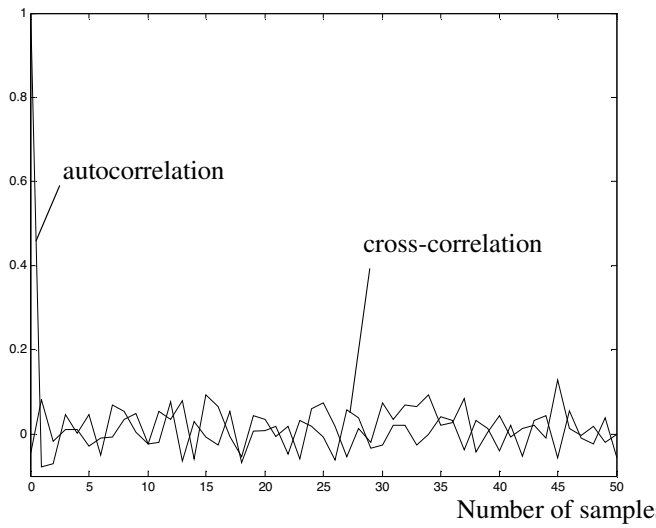


Figure 1.7. *Cross-correlation of encoded signals due to different states compared with the autocorrelation function for one of them*

Next, a very simple repetitive input sequence was applied to the encoder, namely $\{0, 100, 0, -100\}$ in the signed form or $\{0, 100, 0, 156\}$ in the unsigned form [FRE 93]. All possible initial states were again chosen and then the state of the system was sampled every fourth time. This procedure detects periodic responses. More than 96% of initial states give rise to responses having a period more than 50 times the length of the input sequence. The DFT spectrum and the autocorrelation function of the encoded signal were computed for a random sampling of cases. Figures 1.8–1.10 show typical parameters, corresponding to a cycle 513 times the length of the input period, generated for the initial states $x_1[0] = 200$ and $x_2[0] = 147$ in the unsigned form.

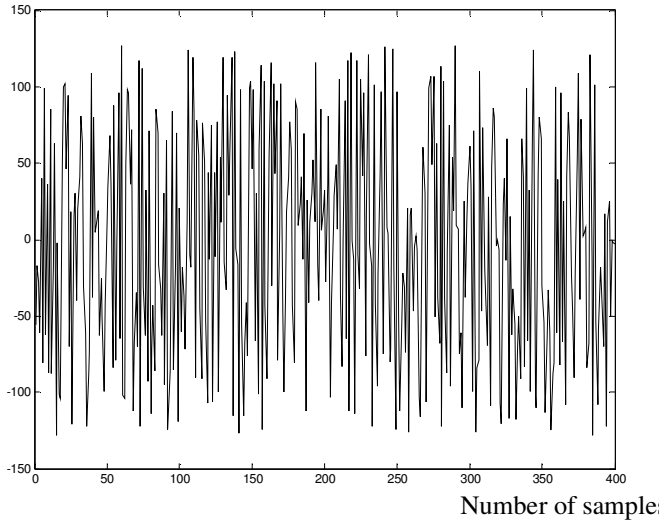


Figure 1.8. *The time domain representation of the encoded signal corresponding to a periodic input*

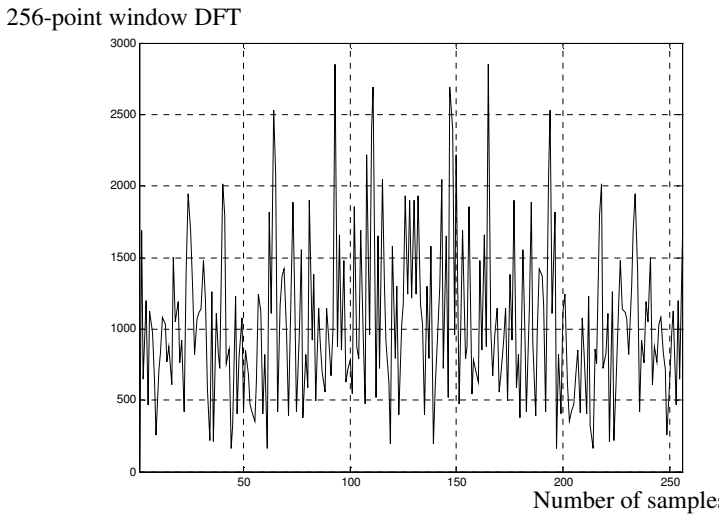


Figure 1.9. *DFT spectrum for encoded signal corresponding to a periodic input*

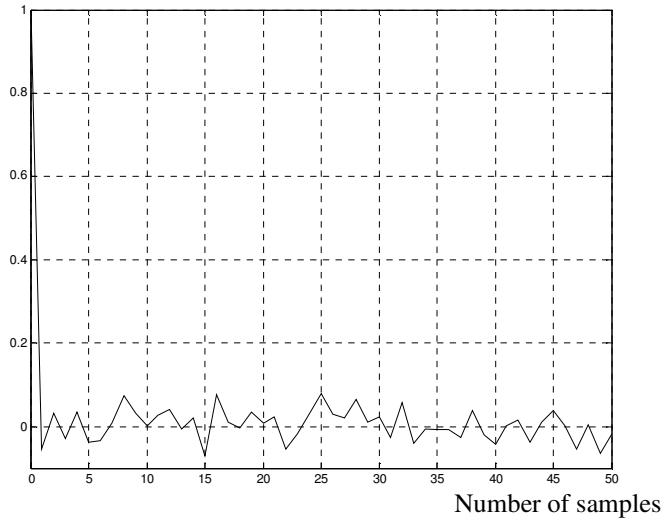


Figure 1.10. *Normalized autocorrelation function for encoded signal corresponding to a periodic input*

The cross-correlation function between the input and the encoded sequence and the autocorrelation function of the encoded sequence are represented comparatively in Figure 1.11. Note how little correlation there is between the two sequences. Figure 1.11 is representative of all the tests we tried.

In all the cases presented above, the input sequence is exactly reconstructed by the decoder, since it is the exact inverse system. An observation is worth making about the LCIRC function. The dynamics of the system were considerably less interesting when it was replaced by a simple modulo multiplication by 2. On the other hand, the dynamics were again comparable to those above when something other than 1 was added because of an overflow, such as 3 or 5. When the encoded response to two different input signals was compared, there was virtually no correlation in the tests conducted. While these tests were not exhaustive, they definitely support the conclusion that this encoder possesses the QC properties [FRE 93].

Finally, since the decoder has FIR, an error in the encoded data manifests itself as a burst error of length 3, equal to the length of the impulse response.

Another feature of this encoder is that it locks onto an encoded signal, regardless of its initial state. Hence, after a short burst error, the decoder correctly recovers the input sequence without having to be initialized. This may be the closest feature this encoding scheme has with to of [CUO 93, KOC 92, HAL 93], and [PAR 92] and would be a desirable feature in many applications for communications systems.

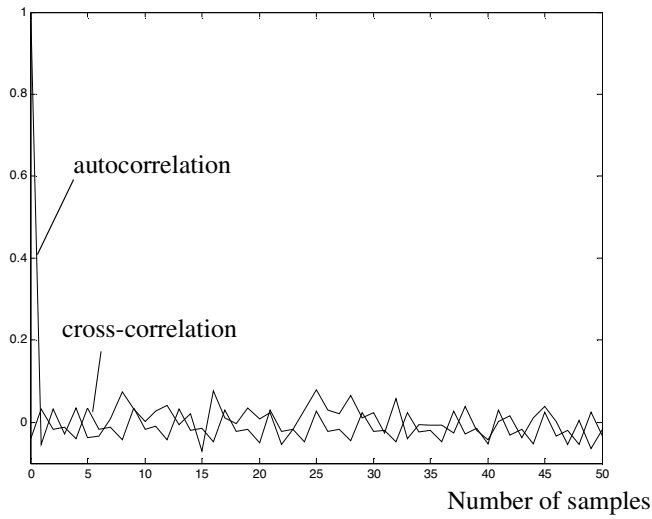


Figure 1.11. *Cross-correlation function of the encoded and input signals compared with the autocorrelation function of the encoded signal*

1.2. Chaotic spreading sequences for direct-sequence code division multiple access

The performances of direct-sequence code division multiple access (DS-CDMA) systems depend mainly on the correlation properties of the spreading sequences set [PUR 77a, PUR 77b, RAP 96, MAZ 97]. The use of low cross-correlation sets of sequences increases the bit error rate (BER) performances and the system capacity as well. Hence, it is imperative to design optimum spreading sequences sets that minimize the BER.

Most DS-CDMA systems presented to date have used binary pseudonoise (PN) maximal-length sequences generated by linear-feedback shift register

(LFSR) schemes. These maximal-length sequences include Gold sequences and Kasami sequences and proved to have quasi-orthogonality correlation properties. Even for minimum cross-correlation sequences, forming Gold and Kasami sets, the set dimension and the period of the sequences are limited by the LFSR polynomial degree. Moreover, these sequences present values for the cross-correlation function that depend on the generator polynomial degree [DIN 98]. Another drawback of these sequences is induced by the generator linearity, which increases the interception probability.

A new direct-sequence spreading method assumes the use of discrete-time nonlinear dynamical systems trajectories. These chaotic sequences present noiselike features that make them good for spreading in DS-CDMA systems. Unless some restrictions are specified, most chaotic systems generate almost perfect random sequences (non-periodic sequences) and their time evolution (their orbit) depends totally on the initial state of the system. So, a single system, described by its discrete chaotic map, can generate a very large number of distinct chaotic sequences, each sequence being uniquely specified by its initial value [FEE 00]. This dependency on the initial state and the nonlinear character of the discrete map make the DS-CDMA system using these sequences more secure. Despite their general process of generation, not all the chaotic sequences are good for spreading. Like in the case of binary PN spreading sequences, it is necessary to select the sets of chaotic PN sequences that present good correlation properties. Mazzini *et al.* [MAZ 97, ROV 00] proposed a new family of chaotic spreading sequences with optimum cross-correlation properties.

To define the desired features that an optimum set of spreading sequences must have, we first have to present some particular details about the DS-CDMA system. Each DS-CDMA user has a unique spreading sequence assigned to him, which has a higher data rate than the baseband signal. As a matter of fact, the baseband data signal is multiplied by the spreading signal and the resulting wideband signal acquires both the spectral and statistical properties of the spreading signal. Hence, all users transmit over the same wideband channel with a total overlapping in time [RAP 96]. This multiple-access interference (MAI) between different DS-CDMA user signals is demonstrated to depend on the correlation properties of the spreading sequences [PUR 77a, PUR 77b, RAP 96, MAZ 97].

Considering the fact that the DS-CDMA system performances depend mainly on the spreading sequences' properties, we can say that an ideal DS-CDMA system uses a set of optimum spreading sequences. Therefore, the optimum set of spreading sequences is defined in the following.

DEFINITION 1.3.– *The optimum set of spreading sequences for the DS-CDMA system is the set composed of sequences having the following properties:*

1) easy to generate, by relatively simple structures;

2) to fulfill the orthogonality condition (null cross-correlation) and to have a null mean value over an information bit period;

3) to minimize the possibility to reconstruct the whole sequence from a short fragment of it;

4) to allow an easy (and fast) sequence synchronization in the receiver;

5) to have the possibility of forming sets of sequences, which are as large as possible, having properties 1–4.

Now, let us discuss each of the properties given above. The first property assures a simple implementation of the sequence generator, and therefore, it also simplifies the implementation of the DS-CDMA transmitter and receiver. As mentioned above, the orthogonality condition specified in property 2 determines the reception performances for each DS-CDMA user. Hence, the smaller the cross-correlation values for the spreading sequences, the smaller the MAI term affecting the decision process in the receiver. The sequences' periodicity is a natural feature for all the sequences generated by a real autonomous system. Because of the numerical limitations given by the operation with finite word lengths, the number of distinct states that such a generator might visit is also limited to a maximum value. Therefore, the generated sequences are pseudo-random (almost random) and the orthogonality mentioned by property 2 depends on the period of the spreading sequences. Also, there is an interdependence between properties 2 and 4. In most practical solutions, the sequence periodicity is used for realizing the sequence synchronization in the receiver, mentioned by property 4. To be more specific, the receiver follows the periodicity of the maximum autocorrelation values in order to lock the phase of the receiver sequence. The security of the transmission scheme depends on the difficulty degree involved in the reconstruction of the spreading sequence by an eavesdropper user.

Therefore, property 3 helps to increase the transmission security, because the probability of undesired interception is reduced. Finally, property 5 refers to the DS-CDMA system capacity, defined as the number of users simultaneously accessing the system's resources. Considering that each user is assigned a unique spreading sequence, the set dimension determines the DS-CDMA system capacity.

It is important to further discuss the interdependencies of properties 1 and 5. For example, the use of sequences having a smaller period may ease both their generation (according to property 1) and synchronization (according to property 4), but it certainly decreases the dimension of the spreading sequences set (according to property 5). However, increasing the sequences' period increases their randomness (according to property 2), but decreases the synchronization performances (according to property 4). However, restricting the randomness properties from property 2, by introducing additional conditions for the sequences, may determine the system capacity decrease (according to property 5), etc.

Now, it is very important to observe some similarities between the QC properties introduced in section 1.1.2, definition 1.1, and the optimum spreading properties given in definition 1.3. The aim of this comparison is to prove that the nonlinear digital filters presented in section 1.1.3 can be used for DS-CDMA spreading. First, we can note that the nonlinear digital filter as shown in Figure 1.3, working in finite precision, possesses the QC properties even for a small word length. For example, all simulation results presented in section 1.1.4 were obtained for an 8-bit word length. Observing that the scheme as shown in Figure 1.3 presents a reduced operational complexity for a small word length, results that these filters fulfill the criterion 1 from definition 1.3. However, properties 1–5 from definition 1.1 represent requirements involved in the definition of the orthogonality between the sequences, and therefore, the nonlinear filters also fulfill criterion 2 from definition 1.3. Considering that the period of each sequence generated by the nonlinear digital filter in Figure 1.3 depends on the initial state of the filter, we can say that a proper selection of the initial state will allow us to select the sequences with a longer period, and therefore, criterion 3 from definition 1.3 is met. However, the security provided by the Frey nonlinear digital filter was already demonstrated in section 1.1.4. Moreover, the high peak of the autocorrelation function of the filter's response, provided by properties 1 and 2 from definition 1.1, will certainly improve the synchronization required by

property 4 from definition 1.3. Finally, the sensitive dependency on the initial states specified in property 5 from definition 1.1 provides the large set of sequences required by property 5 from definition 1.3.

In conclusion, the nonlinear digital filters introduced by Frey [FRE 93] and other similar structures working in finite precision can be used both for secure communications and for DS-CDMA spreading [ELA 06].

1.3. Sequence synchronization in discrete-time nonlinear systems

The digital chaotic communication schemes, where the chaotic signal generator is a discrete-time nonlinear map, allow a more reliable reconstruction of the transmitted information as compared to their analog counterparts, due to the increased protection against noise and other perturbations. Assuming a digital configuration with finite representation, the problem of parameter mismatching and transmission noise surely appears less critical. However, as mentioned in section 1.3, the chaotic signal synchronization from the receiver might pose some additional problems [PEC 90, KOC 92, WU 93]. Nevertheless, there are some special discrete-time maps, which can exhibit the appealing property of synchronizing in finite time. Such a property, called "dead-beat synchronization" as an analogy to the well-known performance of discrete-time control systems [FRA 88], is applied to a simple communication scheme as in [TES 94, DEA 95].

1.3.1. *An example of sequence synchronization using the inverse system*

Let us analyze the synchronization of a particular (but representative) discrete-time system, i.e. the Hénon map, which is a second-order well-known chaotic system, represented by the following equations:

$$\begin{aligned} x_1[n+1] &= 1 - \alpha x_1^2[n] + x_2[n] \\ x_2[n+1] &= \beta x_1[n] \end{aligned} \qquad [1.6]$$

where n denotes the sampled time variable.

It is known that the map exhibits a chaotic behavior in a large neighborhood of the parameter values $\alpha = 1.4$ and $\beta = 0.3$. In addition, the values for x_1 and x_2 are from the interval $(-2, 2)$. The trajectory of the Hénon map (represented as simple as $x_2 = f(x_1)$), given by equation [1.6] and computed for $1,000$ samples in the sequence, is depicted in Figure 1.12. The initial values considered here were $x_1 = 0.3$ and $x_2 = 0.8$.

The system given by [1.6] is assumed to be the transmitter and we select its output as:

$$y[n] = 1 - \alpha x_1^2[n] \qquad [1.7]$$

The receiver is then defined by:

$$\begin{aligned} \hat{x}_1[n+1] &= \hat{x}_2[n] + y[n] \\ \hat{x}_2[n+1] &= \beta \hat{x}_1[n] \end{aligned} \qquad [1.8]$$

The block scheme of the chaotic sequence generator corresponding to equations [1.6] and [1.7] is shown in Figure 1.13, and the block scheme of the receiver for the discrete-time synchronization method, corresponding to equations [1.7] and [1.8], is depicted in Figure 1.14. In fact, the generator is a nonlinear digital filter with a zero input signal.

From [1.6] to [1.8], the synchronization error $\Delta x_i[n] = \hat{x}_i[n] - x_i[n]$ is given by:

$$\begin{aligned} \Delta x_1[n+1] &= \Delta x_2[n] \\ \Delta x_2[n+1] &= \beta \Delta x_1[n] \end{aligned} \qquad [1.9]$$

so tending to zero for $|\beta| < 1$.

The convergence to zero of the synchronization errors $\Delta x_i[n] = \hat{x}_i[n] - x_i[n]$, $i \in \{1, 2\}$, can be easily seen if the equations in [1.9] are rewritten in a recurrent manner. Consider the initial errors $\Delta x_1[0]$ and $\Delta x_2[0]$. Then, the individual errors can be derived.

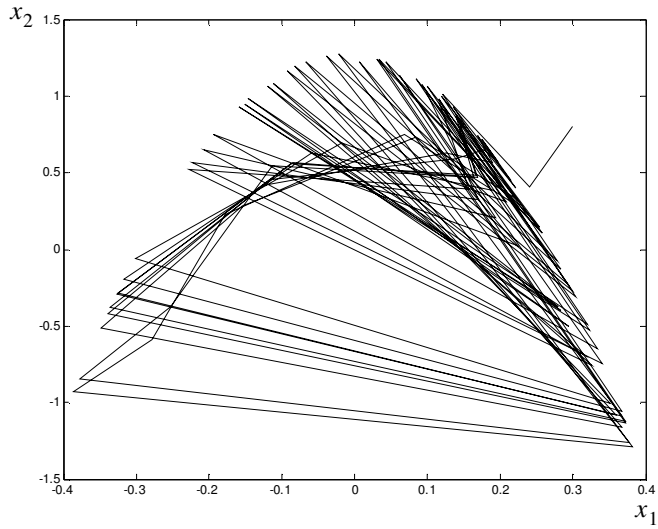


Figure 1.12. *A trajectory of the Hénon map*

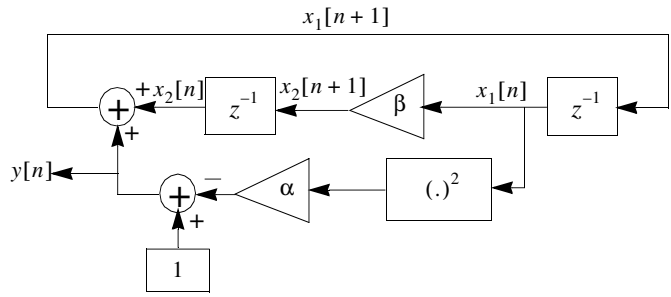


Figure 1.13. *The chaotic sequence generator corresponding to equations [1.6] for the discrete-time synchronization method*

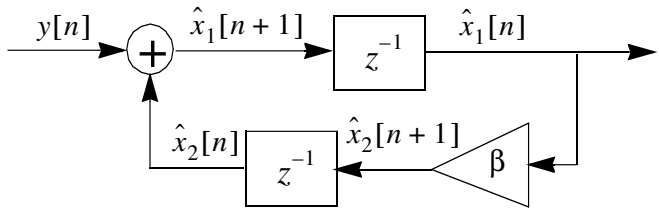


Figure 1.14. *The chaotic receiver corresponding to equations [1.7] and [1.8] for the discrete-time synchronization method*

The synchronization error for the first state variable $\Delta x_1[n]$ is given by:

$$\begin{aligned} \Delta x_1[n+1] &= \Delta x_2[n] = \beta \Delta x_1[n-1] = \beta \Delta x_2[n-2] = \\ &= \beta^2 \Delta x_1[n-3] = \beta^2 \Delta x_2[n-4] \\ &= \beta^j \Delta x_1\left[n-(2j-1)\right] = \beta^j \Delta x_2[n-2j] \\ & j \in \{0,1,2,\ldots\} \end{aligned} \quad [1.10]$$

If n is an odd value, $n = 2m-1$, then:

$$\begin{aligned} \Delta x_1[n+1] &= \beta^j \Delta x_1\left[n-(2j-1)\right] = \beta^{\frac{n+1}{2}} \Delta x_1\left(0\right) \\ & j \in \{0,1,2,\ldots\ \tfrac{n+1}{2}\} \end{aligned} \quad [1.11]$$

else, if n is an even value, $n = 2m$, then:

$$\begin{aligned} \Delta x_1[n+1] &= \beta^j \Delta x_1\left[n-(2j-1)\right] = \beta^j \Delta x_2\left[n-2j\right] = \\ &= \beta^{\frac{n}{2}} \Delta x_1[1] = \beta^{\frac{n}{2}} \Delta x_2[0] \\ & j \in \{0,1,2,\ldots\ \tfrac{n}{2}\} \end{aligned} \quad [1.12]$$

The synchronization error for the second state variable $\Delta x_2[n]$ is given by:

$$\begin{aligned} \Delta x_2[n+1] &= \beta \Delta x_1[n] = \beta \Delta x_2[n-1] = \\ &= \beta^2 \Delta x_1[n-2] = \beta^2 \Delta x_2[n-3] = \\ &= \beta^3 \Delta x_1[n-4] = \beta^j \Delta x_2\left[n-(2j-1)\right] = \\ &= \beta^{j+1} \Delta x_1[n-2j] \\ & j \in \{0,1,2,\ldots\} \end{aligned} \quad [1.13]$$

Now, if n is an odd value, $n = 2m-1$, then:

$$\begin{aligned} \Delta x_2[n+1] &= \beta^j \Delta x_2\left[n-(2j-1)\right] = \beta^{\frac{n+1}{2}} \Delta x_2[0] \\ & j \in \{0,1,2,\ldots\ \tfrac{n+1}{2}\} \end{aligned} \quad [1.14]$$

else, if n is an even value, $n = 2m$, then:

$$\begin{aligned} \Delta x_2[n+1] &= \beta^j \Delta x_2\left[n-(2j-1)\right] = \beta^{j+1} \Delta x_1[n-2j] = \\ &= \beta^{\frac{n}{2}} \Delta x_2[1] = \beta^{\frac{n}{2}+1} \Delta x_1[0] \\ & j \in \{0,1,2,\ldots\ \tfrac{n}{2}\} \end{aligned} \quad [1.15]$$

In conclusion, for every state, x_1 and x_2, the synchronization error converges to zero for $|\beta| < 1$, no matter what initial values they take, $\Delta x_1[0]$ and $\Delta x_2[0]$. Because of the β^n factors, when n increases (i.e. $n \to \infty$) and for $|\beta| < 1$, the synchronization error converges to zero ($\Delta x_i[n] \to 0$). The problem is that the synchronization time, until the transmitter and the receiver are synchronized, depends on the initial values for the synchronization errors $\Delta x_1[0]$ and $\Delta x_2[0]$. The obtained result has the same characteristics of those concerning several continuous time systems [WU 93, TES 94], but a more useful conclusion can now be derived for the studied map.

1.3.2. *The dead-beat synchronization method*

Consider again the Hénon map introduced in section 1.3.1, in equations [1.6], and choose $x_1[n]$ as the transmitted signal, i.e. instead of [1.7] assume:

$$y[n] = x_1[n] \qquad [1.16]$$

so that the receiver equations are now:

$$\begin{aligned} \hat{x}_1[n+1] &= \hat{x}_2[n] + 1 - y^2[n] \\ \hat{x}_2[n+1] &= \beta y[n] \end{aligned} \qquad [1.17]$$

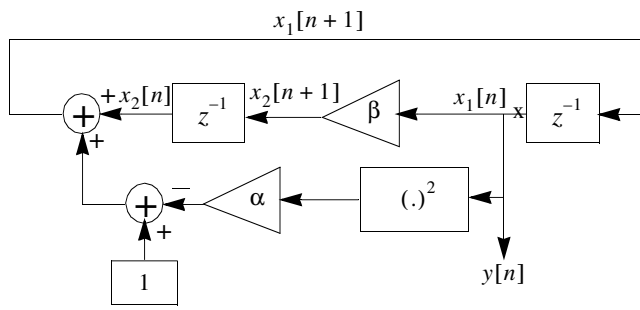


Figure 1.15. *The chaotic sequence generator corresponding to equations [1.6] and [1.16] for the dead-beat synchronization method*

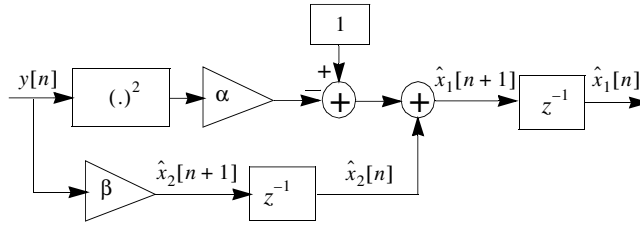


Figure 1.16. *The chaotic receiver corresponding to equations [1.16] and [1.17] for the dead-beat synchronization method*

The block scheme of the chaotic sequence generator corresponding to equations [1.6] and [1.16] is shown in Figure 1.15, and the block scheme of the receiver for the dead-beat synchronization method, corresponding to equations [1.16] and [1.17], is depicted in Figure 1.16 [TES 94, DEA 95]. In fact, the generator is the same nonlinear digital filter as shown in Figure 1.13, but having a different output. This scheme for the receiver is obtained from the block scheme of the generator shown in Figure 1.15, where a cut was made in the loop in order to obtain a open-loop version of the same scheme. This cut is marked with "$\times$" in Figure 1.15 at the output of the first delaying element. So, the input of the receiver in Figure 1.16 is on the left side of the cut, and the output is on the right side of the cut, after the delaying element. From [1.6], [1.16] and [1.17], the synchronization error dynamics results in:

$$\begin{aligned} \Delta x_1[n+1] &= \Delta x_2[n] \\ \Delta x_2[n+1] &= 0 \end{aligned} \qquad [1.18]$$

DEFINITION 1.4.– *Systems that achieve synchronization in a finite number of steps are called dead-beat synchronizing.*

This implies that the errors, independently from their initial values, *will reach exactly zero in two steps*. It is obligatory to note that the dead-beat synchronizing receiver obtains the synchronization after the first two samples from the transmitted sequence are memorized in the two delaying elements from the receiver. So, the number of delaying elements in the schemes will determine the duration of the synchronization cycle. This is true for the case when we have only a transmission delay in the channel between the transmitter and the receiver, and assuming no noise and no other perturbation. When the channel is in noise, the received samples will have different

amplitudes, so the signal at the output of the receiver will synchronize with the received sequence, instead of the transmitted sequence. This is why a reduction method for the noise effects is needed in the receiver to improve the synchronization. For this case with noise, the synchronization cycle will exceed the duration of two steps, as before. A method for reducing the noise effects in the dead-beat synchronizing receiver is presented in [DE 95].

1.3.3. *A communication scheme using the dead-beat synchronization*

The main feature of the dead-beat synchronizing systems is immediate. Due to the deterministic nature of chaotic motions, in fact, once dead-beat synchronization has been achieved, then the two systems will remain synchronized *regardless of the presence of the synchronization signal*. This property can therefore be applied in a new simple secure communication scheme presented in the following [DE 95]. Suppose that the dead-beat synchronization of the chaotic circuit is achieved in Q steps, that $y[n]$ is the chaotic output of the transmitter and $s[n]$ is the information to be sent. Choose a coding function $c\,(s,x)$, i.e. continuous and invertible. Then, the communication process is performed as follows:

– Split the information into strings of M samples each, $M \gg Q$

$$\{s[0], s[1], \ldots, s[M-1]\},\ \{s[M], \ldots, s[2M-1]\}, \ldots \qquad [1.19]$$

– Transmit the following sequence of alternate strings:

- $(A.1)$ $\{y[0], y[1], \ldots, y[Q-1]\}$ as a drive for the chaotic circuit of the receiver.

- $(B.1)$ $\{c\,(s[0], y[Q])\,, c\,(s[1], y[Q+1])\,, \ldots, c\,(s[M-1], y[Q+M-1])\}$ directly to the decoding process.

- $(A.2)$ $\{y[Q+M], \ldots, y[2Q+M-1]\}$ as a drive for the chaotic circuit of the receiver.

- $(B.2)$ $\{c\,(s[M], y[2Q+M])\,, c\,(s[M+1], y[2Q+M+1])\,, \ldots, c\,(s[2M-1], y[2Q+2M-1])\}$ directly to the decoding process.

- ...

as schematically shown in Figure 1.17.

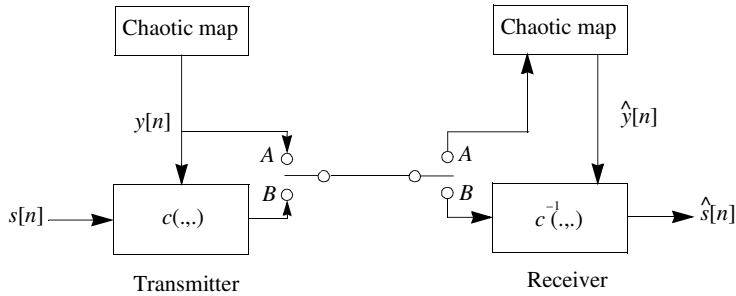


Figure 1.17. *Dead-beat communications scheme. Points A are connected in synchronization phase (step $(A.i)$), while points B are connected in the information transmission phase (step $(B.i)$)*

The receiver, for each couple $(A.i)$, $(B.i)$ of steps, will perform the following operations:

$(A.i)$ Synchronize the chaotic circuit so exploiting its Q-steps dead-beat synchronization property, i.e. after Q steps $y[n] = \hat{y}[n]$.

$(B.i)$ Decode the message by using the output samples $\hat{y}[n]$ of the receiver unforced chaotic circuit, and the knowledge of the inverse coding function c^{-1}, in the form:

$$\hat{s} = c^{-1}\left(c\left(s, y\right), \hat{y}\right) \qquad [1.20]$$

Observe that for a receiver defined by the structure and the parametric configuration of the transmitter (the "nominal" receiver), a perfect reconstruction of $s[n]$ would be guaranteed by a correct synchronization during the step $(A.1)$. Unfortunately, the occurrence of errors in the first synchronizing string would make the following information incomprehensible. The repetition of the synchronization string (step $(A.i)$) at the beginning of each information string (step $(B.i)$) is intended to overcome this problem. In such a way, the decoding can be incorrect for, at most, the M samples of the interested string and will start anew for the following ones. However, if the receiver is slightly different from the nominal one, the coding and the decoding signals $y[n]$ and $\hat{y}[n]$ will quickly become uncorrelated because of the very high sensitivity of chaotic motions to errors, both due to initial conditions and parametric mismatching.

It is important to note that after each synchronization step $(A.i)$, the open-loop receiver must be changed to a closed-loop scheme, such as the transmitter shown in Figure 1.15. During the information transmission step $(B.i)$, the receiver must work as a stand-alone, independent system, so that the transmitter does. Hence, when the switches in the transmitter and the receiver change their positions from A to B, at the same time, the receiver block must switch from the open-loop mode to the closed-loop mode, by re-establishing the connection cut and marked by "$\times$" as shown in Figure 1.15.

2

Presentation of the Frey Nonlinear Encoder as a Digital Filter

2.1. The mathematical analysis of the Frey encoder

In this section, the Frey encoder presented in section 1.1.3 is analyzed. This provides some insight into the properties observed earlier in sections 1.1.3 and 1.1.4. This analysis is possible because of the special properties of the modulo operator, as used in [FRE 93, CHU 88, CHU 90] and [EBE 69]. First, let us reconsider the encoder-defining equations [1.3], resulting from the scheme in Figure 1.3, where this time the modulo operator is explicitly written as follows:

$$\begin{array}{rcl} e[n] & = & mod\,(u[n] + mod[x_1[n] + x_2[n]]) \\ & = & mod\,(u[n] + x_1[n] + x_2[n]) \\ x_1[n] & = & e[n-1] \\ x_2[n] & = & LCIRC\,(x_1[n-1]) \\ & = & mod\,(2x_1[n-1] + s[n]) \\ s[n] & = & \begin{cases} 0 & \text{if} \quad x_1[n-1] < 2^{N-1} \\ 1 & \text{otherwise} \end{cases} \end{array} \qquad [2.1]$$

where N is the binary word length, $x_1[n] = x[n-1]$ and $x_2[n] = x[n-2]$ are the states, namely the outputs of the delays, and the modulo operator has the base given by 2^N. Also, the expression of the left-circulate (LCIRC) function, introduced in definition 1.2, has been used here. Combining these equations,

we have:

$$\begin{aligned} e[n] &= mod\,(u[n] + e[n-1] + mod\,(2e\,(n-2) + s[n])) \\ &= mod\,(u[n] + e[n-1] + 2e[n-2] + s[n]) \end{aligned} \quad [2.2]$$

Note that $s[n]$ plays the role of a noise source that is correlated in a nonlinear way to the response, $e[n]$.

In [PEN 01], Penaud thoroughly analyzed the recursive system used by Frey to generate the chaotic sequences. In fact, Penaud introduced a more general case of the coder/decoder scheme proposed by Frey in [FRE 93]. Hence, equations [2.1] and [2.2] are rewritten in a generalized form as follows:

$$e[n] = u[n] \oplus f\left\{\sum_{i=1}^{M} (K_i e\,[n - D_i] \oplus s\,[n])\right\} \quad [2.3]$$

where $f\,(x) = \begin{cases} x & \text{if } -2^{N-1} \leq x \leq 2^{N-1} - 1 \\ x \bmod \left(2^N\right) & \text{otherwise} \end{cases}$. In all the computations performed, the integer values for $e\,[n]$, $u\,[n]$, $s\,[n]$ and K_i were assumed to be represented in 2's complement form (C2), yielding positive and negative integers with a mean value close to zero. Also, all additions are modulo-2^N, where N is the binary equivalent word length. These operators are assumed to be generally nonlinear operations, due to overflow in the actual finite length adders used for hardware computation.

It can be easily seen that the basic Frey codec presented in section 1.1.3 (equation [1.3]) is a particular case of the system given by [2.3], considering $M = 2$, $K_1 = 1$, $K_2 = 2$, $D_1 = 1$ and $D_2 = 2$, respectively.

The detailed expressions for the definitions and properties of these operators are presented in section 2.2.

2.2. The definitions and properties of the unsigned and 2's complement signed sample operators

DEFINITION 2.1.– *An unsigned number, denoted as y^U, is a natural number represented with code words of N bits. From this point on, the superscript U will denote an unsigned number, which is a positive integer.*

If the following notation is considered, i.e. $y^U = x^U \mathrm{mod}\left(2^N\right)$, then $y^U \in \left[0,\, 2^N - 1\right]$, for any positive integer number x^U. If x^U is already a value less than $2^N - 1$, then we have the following expression: $y^U = x^U = x^U \mathrm{mod}\left(2^N\right) \in \left[0,\, 2^N - 1\right]$.

DEFINITION 2.2.– *An signed number, denoted as y^s, is an integer number, represented in 2's complement ($C2$) form with code words of N bits. From this point on, the superscript s will denote a signed $C2$ number, which is an integer.*

If the following notation is considered, i.e. $y^S = x^S \mathrm{mod}\left(2^N\right)$, then $y^S \in \left[-2^{N-1},\, 2^N - 1\right]$, for any signed integer number x^S. If x^S is already a value from the modulo $C2$ interval, i.e. $x^S \in \left[-2^{N-1},\, 2^N - 1\right]$, then y^S is computed as follows: $y^S = x^S = x^S \mathrm{mod}\left(2^N\right) \in \left[-2^{N-1},\, 2^{N-1} - 1\right]$. In the sequel, we will show how this signed number is expressed.

DEFINITION 2.3.– *The two sets of integer numbers presented above, $x^U \in \{0,\, 2^N - 1\}$ and $x^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right]$, will be referred to as the 2^N-set and the $\left[\mathrm{C2}, 2^N\right]$-set, respectively.*

DEFINITION 2.4.– *The number x^U is converted into its corresponding signed $C2$ value, x^S, using the following equation:*

$$\begin{aligned} \left(x^U\right)^S &= x^S \\ &= \begin{cases} x^U & \text{if} \quad 0 \leq x^U \leq 2^{N-1} - 1 \\ C_2\left(2^N - x^U\right) = x^U - 2^N & \text{if} \quad 2^{N-1} \leq x^U \leq 2^N - 1 \end{cases} \end{aligned} \quad [2.4]$$

where $C_2\left(x\right)$ is the 2's complement value for x represented with N bits.

In the following, this operation will be denoted as $\left(x^U\right)^S$ and is defined as the *conversion function from the unsigned natural representation to the signed 2's complement representation*: $\left(x^U\right)^S = x^S$, where $x^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right]$ and $x^U \in \left[0,\, 2^N - 1\right]$.

NOTE 2.1.– For the general case, when the unsigned natural number x^U does not necessarily take a value from the interval $x^U \in \left[0,\, 2^N - 1\right]$, the conversion from the unsigned form to the signed $C2$ form can be performed as follows:

$$y^S = x^S \mathrm{mod}\left(2^N\right) = \left(x^U \mathrm{mod}\left(2^N\right)\right)^S \quad [2.5]$$

In fact, a new operator was used, the signed modulo-2^N operator. This operation consists of computing the normal modulo-2^N value and then, a conversion to the signed $C2$ form is performed, as above. The above expression is used to estimate any signed number, as defined in definition 2.2.

DEFINITION 2.5.– *The inverse operation of* $\left(x^U\right)^S$*, which performs the conversion function from the signed 2's complement representation to the unsigned natural representation, is denoted by* $\left(x^S\right)^U$*. The number* x^S *is converted into its corresponding unsigned value,* x^U*, as follows:*

$$x^U = \begin{cases} x^S & \text{if} \quad 0 \leq x^S \leq 2^{N-1} - 1 \\ x^S + 2^N & \text{if} \quad -2^{N-1} \leq x^S < 0 \end{cases} \qquad [2.6]$$

where $\left(x^S\right)^U = x^U$*,* $x^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right]$ *and* $x^U \in \left[0,\, 2^N - 1\right]$*.*

DEFINITION 2.6.– *The addition operator of unsigned* x^U *numbers in the* 2^N*-set. Let us consider two unsigned numbers* $x^U, y^U \in \left[0,\, 2^N - 1\right]$*, then we define the addition operation in the* 2^N*-set as:*

$$\begin{array}{l} z^U = x^U \oplus y^U = \left(x^U + y^U\right) \bmod \left(2^N\right) \\ z^U \in \left[0,\, 2^N - 1\right] \end{array} \qquad [2.7]$$

NOTE 2.2.– Because of some properties of the modulo-2^N operation, we can derive some new expressions.

In the general case, for the expression $a \bmod b$, we have:

$$a \bmod b = r \qquad [2.8]$$

where r is the remainder of the division of a and b $\left(\frac{a}{b}\right)$:

$$a = bc + r \qquad [2.9]$$

and c is the quotient of $\left(\frac{a}{b}\right)$.

For the case discussed above, when we use the modulo-2^N operation for the unsigned 2^N-set numbers, we can write:

$$\left(x^U + y^U\right) \bmod \left(2^N\right) = r \Leftrightarrow x^U + y^U = 2^N c + r \qquad [2.10]$$

When $x^U, y^U \in \left[0,\, 2^N - 1\right]$, we have $0 \leq x^U + y^U \leq 2^{N+1} - 2$, and then the quote c takes one of the values $c \,\text{in}\{0, 1\}$. Taking this into account, the next expression can be derived:

$$x^U + y^U = \begin{cases} 2^N + \left(x^U + y^U\right) \bmod \left(2^N\right) & \text{if} \quad c = 1 \\ \left(x^U + y^U\right) \bmod \left(2^N\right) & \text{if} \quad c = 0 \end{cases} \qquad [2.11]$$

Equation [2.11] is equivalent to:

$$x^U + y^U = \begin{cases} 2^N + \left(x^U + y^U\right) \bmod \left(2^N\right) \\ \quad \text{if} \quad 2^N \leq x^U + y^U \leq 2^{N+1} - 2 \\ \left(x^U + y^U\right) \bmod \left(2^N\right) \\ \quad \text{if} \quad 0 \leq x^U + y^U \leq 2^N - 1 \end{cases} \qquad [2.12]$$

Now, from equations [2.11] and [2.12], the following expression for the addition operation in the 2^N-set is obtained:

$$\begin{array}{l} r = z^U = x^U \oplus y^U = \\ = \begin{cases} x^U + y^U - 2^N & \text{if} \quad 2^N \leq x^U + y^U \leq 2^{N+1} - 2 \\ x^U + y^U & \text{if} \quad 0 \leq x^U + y^U \leq 2^N - 1 \end{cases} \\ z^U \in \left[0,\, 2^N - 1\right] \end{array} \qquad [2.13]$$

NOTE 2.3.– In the following, we can note that the condition $0 \leq x^U + y^U \leq 2^{N+1} - 2$ is equivalent to $0 \leq 2^N c + r \leq 2^{N+1} - 2$. Therefore, considering equations [2.8] and [2.9], we have two possible cases:

1) If $c = 0 \Rightarrow 0 \leq r \leq 2^N - 1$, or

2) If $c = 1 \Rightarrow 0 \leq 2^N + r \leq 2^{N+1} - 2$.

So, considering both cases, the remainder r can take a value only from the reunion interval: $r = z^U \in \{[0,\ 2^N - 1] \cup [0,\ 2^N - 2]\} = [0,\ 2^N - 1]$. Hence, the condition $z^U \in [0,\ 2^N - 1]$ from [2.13] is obvious.

NOTE 2.4.– Taking into account the previous observation and equation [2.13], a general expression of the addition operation, for any number of terms, in the 2^N-set can be derived. Let us consider m unsigned values ${a_i}^U, i \in [1,\ m], {a_i}^U \in [0,\ 2^N - 1]$. In the appendix, it is demonstrated that the addition of all these values, in the 2^N-set, is given by the following equation:

$$
\begin{aligned}
b^U = {a_1}^U \oplus {a_2}^U \oplus {a_3}^U \oplus \ldots \oplus {a_m}^U = \left(\sum_{i=1}^{m} {a_i}^U\right) \bmod \left(2^N\right) = \\
\begin{cases}
\sum_{i=1}^{m} {a_i}^U - (m-1)\,2^N \\
\quad \text{if} \quad (m-1)\,2^N \le \sum_{i=1}^{m} {a_i}^U \le m2^N - m \\
\sum_{i=1}^{m} {a_i}^U - (m-2)\,2^N \\
\quad \text{if} \quad (m-2)\,2^N \le \sum_{i=1}^{m} {a_i}^U \le (m-1)\,2^N - 1 \\
\sum_{i=1}^{m} {a_i}^U - (m-3)\,2^N \\
\quad \text{if} \quad (m-3)\,2^N \le \sum_{i=1}^{m} {a_i}^U \le (m-2)\,2^N - 1 \\
\ldots\ldots\ldots\ldots\ldots \\
\sum_{i=1}^{m} {a_i}^U \quad \text{if} \quad 0 \le \sum_{i=1}^{m} {a_i}^U \le 2^N - 1
\end{cases} \\
b^U \in [0,\ 2^N - 1]
\end{aligned}
\qquad [2.14]
$$

DEFINITION 2.7.– *The addition operator of signed x^S numbers in the $[\mathrm{C2}, 2^N]$-set. Let us consider two signed $C2$ numbers $x^S, y^S \in [-2^{N-1},\ 2^{N-1} - 1]$, then we define the addition operation in the $[\mathrm{C2}, 2^N]$-set as:*

$$
\begin{aligned}
& z^S = x^S \oplus y^S = \left(x^U \oplus y^U\right)^S = \left[\left(x^U + y^U\right) \bmod \left(2^N\right)\right]^S = \left(z^U\right)^S \\
& z^S = \left(z^U\right)^S \in [-2^{N-1},\ 2^N - 1]
\end{aligned}
\qquad [2.15]
$$

THEOREM 2.1.– The addition operation in the $\left[\mathrm{C2}, 2^N\right]$-set is associative.

This associativity property can be expressed as follows:

$$\begin{array}{l} x^S \oplus y^S \oplus z^S = \left(x^S \oplus y^S\right) \oplus z^S = x^S \oplus \left(y^S \oplus z^S\right) \\ x^S, y^S, z^S \in \left[-2^{N-1},\, 2^N - 1\right] \end{array} \quad [2.16]$$

DEMONSTRATION 2.1.– According to definition 2.7 and [2.11], we can derive the following:

$$x^S \oplus y^S = \left(x^U \oplus y^U\right)^S \quad [2.17]$$

and then:

$$x^S \oplus y^S \oplus z^S = \left(x^U \oplus y^U \oplus z^U\right)^S \quad [2.18]$$

$q.e.d.$

NOTE 2.5.– If the addition operation in the 2^N-set is associative, then the addition operation in $\left[\mathrm{C2}, 2^N\right]$-set is also associative.

THEOREM 2.2.– The addition operation in the 2^N-set is associative:

$$\begin{array}{ll} w^U & = x^U \oplus y^U \oplus z^U = \left(x^U \oplus y^U\right) \oplus z^U = x^U \oplus \left(y^U \oplus z^U\right) = \\ & = \left(x^U + y^U + z^U\right) \bmod \left(2^N\right) \\ w^U & \in \left[0,\, 2^N - 1\right] \end{array} \quad [2.19]$$

The demonstration of theorem 2.2 is presented in the appendix.

THEOREM 2.3.– The addition operation in the $\left[\mathrm{C2}, 2^N\right]$-set is commutative. This commutativity property can be expressed as follows:

$$\begin{array}{l} x^S \oplus y^S = y^S \oplus x^S \\ x^S, y^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right] \end{array} \quad [2.20]$$

DEMONSTRATION 2.2.– According to equations [2.7] and [2.13] in definition 2.6 and taking into account the commutativity property of the addition operation for integer values, it is obvious that the addition operation in the 2^N-set is also commutative. Now, from equation [2.15] in definition 2.7, it results that the addition operation in the $\left[\mathrm{C}2, 2^N\right]$-set is also commutative.

$q.e.d.$

DEFINITION 2.8.– *The subtraction operator of unsigned numbers in the 2^N-set. Let us consider two unsigned numbers $y^U, z^U \in \left[0,\, 2^N - 1\right]$, then we define the subtraction operation in the 2^N-set as:*

$$\begin{array}{l} x^U = z^U \ominus y^U = \left(z^U - y^U\right) \bmod \left(2^N\right) \\ x^U, y^U, z^U \in \left[0,\, 2^N - 1\right] \end{array} \qquad [2.21]$$

THEOREM 2.4.– The subtraction operator of unsigned numbers in the 2^N-set is the inverse of the addition operator of unsigned numbers in the 2^N-set. In another words, if $z^U = x^U \oplus y^U$, then $x^U = z^U \ominus y^U$ and vice versa.

The demonstration of theorem 2.4 is presented in the appendix.

NOTE 2.6.– From equation [2.21] and similarly to equation [2.13], the following compact expression for the $\ominus$ operator is obtained:

$$\begin{array}{l} x^U = z^U \ominus y^U = \left\{ \begin{array}{ll} z^U - y^U + 2^N & \text{if} \quad -2^N + 1 \leq z^U - y^U \leq -1 \\ z^U - y^U & \text{if} \quad 0 \leq z^U - y^U \leq 2^N - 1 \end{array} \right. \\ x^U, y^U, z^U \in \left[0,\, 2^N - 1\right] \end{array} \qquad [2.22]$$

DEFINITION 2.9.– *The subtraction operator of signed x^S numbers in the $\left[\mathrm{C}2, 2^N\right]$-set. Similar to equation [2.15] in definition 2.7, we can define a subtraction operator of signed x^S numbers in the $\left[\mathrm{C}2, 2^N\right]$-set. Let us consider two signed $C2$ numbers $x^S, y^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right]$, then we define the subtraction operation in the $\left[\mathrm{C}2, 2^N\right]$-set as:*

$$\begin{array}{l} x^S = z^S \ominus y^S = \left(z^U \ominus y^U\right)^S = \left[\left(z^U - y^U\right) \bmod \left(2^N\right)\right]^S = \left(x^U\right)^S \\ x^S = \left(x^U\right)^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right] \end{array} \qquad [2.23]$$

NOTE 2.7.– The subtraction operator of signed x^S numbers in the $\left[\mathrm{C2}, 2^N\right]$-set was used to define the chaotic decoders in [FRE 93] and [PEN 01].

NOTE 2.8.– The subtraction operator represents the reverse operation for the addition operation in the $\left[\mathrm{C2}, 2^N\right]$-set.

DEFINITION 2.10.– *The multiplication operator of unsigned x^U numbers in the 2^N-set. Let us consider two unsigned numbers $x^U, y^U \in \left[0,\ 2^N - 1\right]$, then we define the multiplication operation in the 2^N-set as:*

$$\begin{array}{l} z^U = x^U \otimes y^U = \left(x^U \cdot y^U\right) \bmod \left(2^N\right) \\ x^U, y^U, z^U \in \left[0,\ 2^N - 1\right] \end{array} \qquad [2.24]$$

NOTE 2.9.– From equation [2.24] and similarly to [2.13], the following expression for the $\otimes$ operator is obtained:

$$z^U = x^U \otimes y^U$$

$$= \begin{cases} x^U \cdot y^U - \left(2^N - 2\right) 2^N \\ \quad \text{if} \quad \left(2^N - 2\right) 2^N \leq x^U \cdot y^U \leq \left(2^N - 2\right) 2^N + 1 \\ x^U \cdot y^U - \left(2^N - 3\right) 2^N \\ \quad \text{if} \quad \left(2^N - 3\right) 2^N \leq x^U \cdot y^U \leq \left(2^N - 2\right) 2^N - 1 \\ \ldots\ldots\ldots\ldots\ldots \\ x^U \cdot y^U - 2 \cdot 2^N \quad \text{if} \quad 2 \cdot 2^N \leq x^U \cdot y^U \leq 3 \cdot 2^N - 1 \\ x^U \cdot y^U - 2^N \quad \text{if} \quad 2^N \leq x^U \cdot y^U \leq 2 \cdot 2^N - 1 \\ x^U \cdot y^U \quad \text{if} \quad 0 \leq x^U \cdot y^U \leq 2^N - 1 \end{cases}$$

$$x^U, y^U, z^U \in \left[0,\ 2^N - 1\right] \qquad [2.25]$$

DEFINITION 2.11.– *The multiplication operator of signed numbers in the $\left[\mathrm{C2}, 2^N\right]$-set. Similar to equation [2.15] in definition 2.7, we can define a multiplication operator of signed x^S numbers in the $\left[\mathrm{C2}, 2^N\right]$-set. Let us consider two signed $C2$ numbers $x^S, y^S \in \left[-2^{N-1},\ 2^{N-1} - 1\right]$, then we*

define the multiplication operation in the $\left[\mathrm{C}2, 2^N\right]$*-set as:*

$$\begin{aligned} &z^S = x^S \otimes y^S = \left(x^U \otimes y^U\right)^S = \left[\left(x^U \cdot y^U\right) \bmod \left(2^N\right)\right]^S = \left(z^U\right)^S \\ &z^S = \left(z^U\right)^S \in \left[-2^{N-1},\, 2^{N-1} - 1\right] \end{aligned} \quad [2.26]$$

2.3. The properties of the LCIRC nonlinear function used in the Frey encoder scheme

In section 1.1.3, it was mentioned that Frey proposed a 2's complement representation form for all the computations in the codec in Figure 1.3, in order to determine a direct current (DC) component close to zero. To simplify the codec analysis, we will consider a conversion from the signed 2's complement form to the unsigned form and its reverse conversion. Therefore, the codec in Figure 1.3 that works entirely in signed 2's complement can be changed to an equivalent codec that first converts each signed input sample value $u^S[n]$ into an unsigned value $u^U[n]$, processes it in unsigned modulo-2^N form and then at the output, converts each unsigned input $e^U[n]$ sample value into a signed value $e^S[n]$. For this scheme equivalency, we used the operators and their properties introduced in section 2.2. This equivalent scheme was used for all simulations in sections 1.1.4 and 2.4. More details about this simulation scheme are also presented in section 2.4.

In the following, some properties of the LCIRC function are analyzed.

Using definition 2.6 for the addition operator of unsigned x^U numbers in the 2^N-set (equation [2.7]), expression [1.4] can be rewritten as:

$$\begin{aligned} &y^U = \left(2 \cdot x^U\right) \oplus s \\ &x^U, y^U \in \left[0,\, 2^N - 1\right] \end{aligned} \quad [2.27]$$

where the time variable n was eliminated for the sake of clarity.

NOTE 2.10.– The unsigned natural value for the product $2 \cdot x^U$ from equation [2.27] does not necessarily take a value from the interval $\left[0,\, 2^N - 1\right]$. Only the unsigned natural number x^U is from the interval $x^U \in \left[0,\, 2^N - 1\right]$. Also, the unsigned natural value s takes a value from two: 0 or 1, both being from the interval $\left[0,\, 2^N - 1\right]$. So, we can consider s as an unsigned number in the

2^N-set, $s = s^U$. In fact, s adds a value 1 to the product $2 \cdot x^U$ only if this product exceeds the 2^N-set interval's maximum limit. This feature confers the nonlinear property of this function.

THEOREM 2.5.– If unsigned number x^U from the 2^N-set is represented in a binary string, from the 2^N-field (2-base) form, as ${x_{(2)}}^U = (x_{N-1}, x_{N-2}, \ldots, x_1, x_0)$, where $\{x_i, i \in \{0, \ldots N-1\}\} = \{0, 1\}$, and the 10-base representation of $x_{(2)}$ is:

$$\begin{aligned} x^U &= x^U_{(10)} = x_{N-1} \cdot 2^{N-1} + x_{N-2} \cdot 2^{N-2} + \ldots + x_1 \cdot 2 + x_0 \\ &= \sum_{i=1}^{N-1} x_i \cdot 2^i, \end{aligned} \qquad [2.28]$$

then the LCIRC value of x^U is having the binary representation as:

$$\begin{aligned} y^U &= {y_{(10)}}^U = LCIRC\left(x^U\right) \\ &\Rightarrow {y_{(2)}}^U = (x_{N-2}, x_{N-3}, \ldots, x_1, x_0, x_{N-1}) \end{aligned} \qquad [2.29]$$

NOTE 2.11.– From [2.29], it is obvious that in the binary representation of $y_{(2)}$, the first bit, i.e. the most significant bit (MSB), from $x_{(2)}$ has changed its position to the least significant bit (LSB) position, and all the other bits in $x_{(2)}$ have been shifted one position to the left. This is the reason for calling this function the *LCIRC function*.

DEMONSTRATION 2.3.– Let us consider the following notations for the binary representations of y^U and x^U as ${y_{(2)}}^U = (y_{N-1}, y_{N-2}, \ldots, y_1, y_0)$, and ${x_{(2)}}^U = (x_{N-1}, x_{N-2}, \ldots, x_1, x_0)$, respectively. Then, regarding the value of the MSB in ${x_{(2)}}^U$, we have two possible cases:

Case (a) $x_{N-1} = 0$. In this case, the decimal value of x^U is within the interval $0 \leq {x_{(10)}}^U \leq 2^{N-1} - 1 \Rightarrow s = 0$ and then $0 \leq 2 \cdot {x_{(10)}}^U \leq 2^N - 2$. Now, the decimal value for y^U can be expressed as: ${y_{(10)}}^U = LCIRC\left({x_{(10)}}^U\right) = \left(2 \cdot {x_{(10)}}^U + s\right) \bmod \left(2^N\right) = 2{x_{(10)}}^U$. Using the binary representation for x^U, and considering $x_{N-1} = 0$, the next expression is derived:

$$\begin{aligned} {y_{(10)}}^U &= 2{x_{(10)}}^U = 2 \cdot \left(x_{N-1} \cdot 2^{N-1} + x_{N-2} \cdot 2^{N-2} + \ldots + x_1 \cdot 2 + x_0\right) \\ &= x_{N-2} \cdot 2^{N-1} + x_{N-3} \cdot 2^{N-2} + \ldots + x_1 \cdot 2^2 + x_0 \cdot 2 + 0 \cdot 2^0 \end{aligned} .$$

Considering the LSB as $x_{N-1} = 0$, the binary representation for y^U is given by:

$$y_{(2)}{}^U = (x_{N-2}, x_{N-3}, \ldots, x_1, x_0, x_{N-1}) \quad [2.30]$$

$q.e.d.$

Case (b) $x_{N-1} = 1$. In this case, the decimal value of x^U is within the interval: $2^{N-1} \leq x_{(10)}{}^U \leq 2^N - 1 \Rightarrow s = 1$ and then $2^N \leq 2 \cdot x_{(10)}{}^U \leq 2^{N+1} - 2$. The argument for the LCIRC function respects the condition: $2^N + 1 \leq 2 \cdot x_{(10)}{}^U + s \leq 2^{N+1} - 1$. Now, from the definition of the addition operation in the 2^N-set (equation [2.13]), the decimal value for y^U can be expressed as: $y_{(10)}{}^U = LCIRC\left(x_{(10)}{}^U\right) = \left(2 \cdot x_{(10)}{}^U + 1\right) \bmod \left(2^N\right) = 2x_{(10)}{}^U + 1 - 2^N$. Using the binary representation for x^U, and considering $x_{N-1} = 1$, the next expression is derived:

$$\begin{aligned} y_{(10)}{}^U &= 2x_{(10)}{}^U + 1 - 2^N \\ &= 2 \cdot \left(x_{N-1} \cdot 2^{N-1} + x_{N-2} \cdot 2^{N-2} + \ldots + x_1 \cdot 2 + x_0\right) + 1 - 2^N \\ &= 2^N + x_{N-2} \cdot 2^{N-1} + x_{N-3} \cdot 2^{N-2} + \ldots + x_1 \cdot 2^2 + x_0 \cdot 2 \\ &\quad +1 - 2^N \\ &= x_{N-2} \cdot 2^{N-1} + x_{N-3} \cdot 2^{N-2} + \ldots + x_1 \cdot 2^2 + x_0 \cdot 2 + 1 \cdot 2^0 \end{aligned} .$$

Considering the LSB as $x_{N-1} = 1$, the binary representation for y^U is given by:

$$y_{(2)}{}^U = (x_{N-2}, x_{N-3}, \ldots, x_1, x_0, x_{N-1}) \quad [2.31]$$

$q.e.d.$

Considering both the cases, a and b, theorem 2.5 is demonstrated.

The LCIRC shifting of one position to the left in the binary representation of the argument demonstrated in theorem 2.5 can be generalized for any number of bit rotations. This general LCIRC function is defined in the following.

DEFINITION 2.12.– *Considering a bit index denoted by k for the binary representations using N bits in a code word, the LCIRC function application*

for k times consecutively is defined as follows:

$$LCIRC^k\left(x^U\right) \triangleq \underbrace{LCIRC\left(LCIRC\left(\ldots LCIRC\left(x^U\right)\right)\right)}_{k \text{ times}} \qquad [2.32]$$

THEOREM 2.6.– The LCIRC function application for k times, defined in [2.32], determines a k bits rotation in the binary code word.

The demonstration of theorem 2.6 is similar to the demonstration given for theorem 2.5, considering the latter in each bit rotation from the k rotations.

NOTE 2.12.– As a result of theorem 2.6, when applying, for N times consecutively, the LCIRC function to an N bits word length unsigned value x^U, it results in the original argument value:

$$LCIRC^N\left(x^U\right) \triangleq \underbrace{LCIRC\left(LCIRC\left(\ldots LCIRC\left(x^U\right)\right)\right)}_{N \text{ times}} = x^U \qquad [2.33]$$

2.4. The simulation of the Frey sequence generator block in Simulink: some practical considerations

2.4.1. *The transmitter chaotic sequence generator*

Considering the properties of the operators defined in sections 2.2 and 2.3 for signed x^S numbers in the $\left[\mathrm{C2}, 2^N\right]$-set and using the block scheme of the basic encoder–decoder proposed by Frey in [FRE 93] and shown in Figure 1.3 (section 1.1.3), a chaotic sequence generator was implemented in Simulink.

The signal at the output of this generator is composed by the B_i quantified levels, represented with N bits word length and having the duration of T seconds, for each output sample. The extreme quantified levels, determined by the word length N, take the values:

$$L = -2^{N-1},\ L_{max} = 2^{N-1-1}.$$

To reduce the signal's power and not to make its amplitude depend on the number of levels (in fact, on N), all the levels are normalized by the maximum absolute value of the quantified levels, $L' = 2^{N-1}$. Then, the generated chaotic signal is given by:

$$e'^{S}(t) = \sum_{i=0}^{\infty} B_i p_T(t - iT) \qquad [2.34]$$

where $\frac{-2^{N-1}}{L''} \leq B_i \leq \frac{2^{N-1}-1}{L''}$, $L'' = 2^{N-1} \Rightarrow -1 \leq B_i \leq 1$

and $p_T(t) = \begin{cases} 1 & \text{if} \quad 0 \leq t \leq T \\ 0 & \text{otherwise} \end{cases}$

The input signal of the encoder will be used as a key signal to change the generator's dynamics. The block scheme used to simulate the chaotic signal generator is shown in Figure 2.1.

As discussed in section 1.1.3, this block scheme includes two delay elements and some modulo-2^N adders. To benefit from the modulo function defined for real, positive numbers, some conversions from the unsigned x^U number representation in the 2^N-set to signed x^S number representation in the $\left[\mathrm{C2}, 2^N\right]$-set, and the reverse conversion, were performed. These conversions follow equations [2.4] and [2.5], presented in section 2.2. Hence, the idea was to make the loop work in the unsigned manner inside, and to output the signal in a signed manner. The output signal $e'^{S}[n]$ is the normalized signed representation signal obtained from the internal unsigned representation sequence $e^U[n]$. This normalized signal will be used for transmission. The signal $e^U[n]$ is passed through two consecutive delay elements, in the unsigned internal representation, and added modulo-2^N according to the basic encoder scheme shown in Figure 1.3 (section 1.1.3). Before the normalization, the signal is converted to the signed form.

The block denoted as "carry bit function $s[n]$", represents a block that verifies the condition for adding the carry bit and performs this addition correspondingly, according to definition 1.2 of LCIRC, presented in section 1.1.3. The signal $s[n]$ generated like this was considered as a sequence of unsigned 2^N-set values and the addition operation with the double value of $e^U[n-2]$ was considered as a modulo-2^N addition. According to the

definitions and classifications of signed/unsigned representations in the 2^N-set, these modifications do not change the principle of the encoder presented in section 1.1.3 at all. The input key unsigned form signal $u^U[n-2]$ is added in a modulo fashion to the signal resulting from the previous additions, and after a conversion, to the signed representation, the obtained signal is fed to the output. The unsigned result is again placed at the input of the first delaying element and so the loop is closed. Some further details about the implementation in Simulink for each of these blocks will be presented in section 2.4.3.

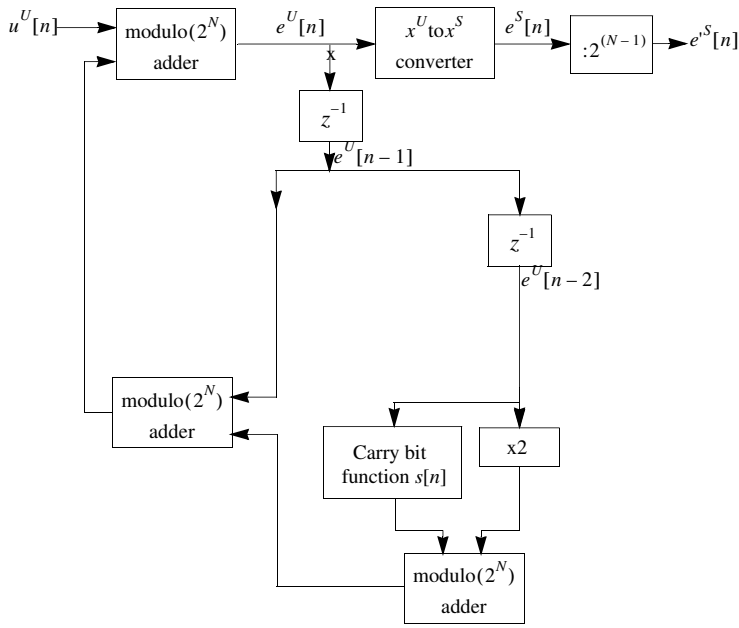


Figure 2.1. *The block scheme for the simulated chaotic signal generator*

2.4.2. *The receiver chaotic sequence generator and the dead-beat synchronization with the transmitter block*

As presented in section 1.3.2, the receiver chaotic signal generator, which presents the dead-beat synchronization feature, is an open-loop version of the transmitter generator. This means that the receiver's chaotic sequence generator has almost the same implementation scheme, as in the transmitter,

only that a link in the loop is cut, in order to obtain the open-loop version. It is known that the best way to open the loop is to make a cut before and after one of the two delay elements in the scheme shown in Figure 2.1 [DE 95]. The cut comes up with some other necessary modifications in the scheme, like the need to have the output of the transmitter generator, and both the input and the output of the receiver generator, virtually in the same point. This cut is marked with "$\times$" in Figure 2.1, at the input of the first delaying element. So, considering all the comments above, the block scheme of the receiver chaotic generator is shown in Figure 2.2. First, the received signal is increased in amplitude by the factor $L' = 2^{N-1}$, in order to have in the receiver the same values for the B_i quantified levels, as in the transmitter. The following operations are identical to those performed in the transmitter generator. The output signal $e'''^S[n]$ is a normalized signed representation signal, which in an error-free case is identical to the input signal $e'^S[n]$.

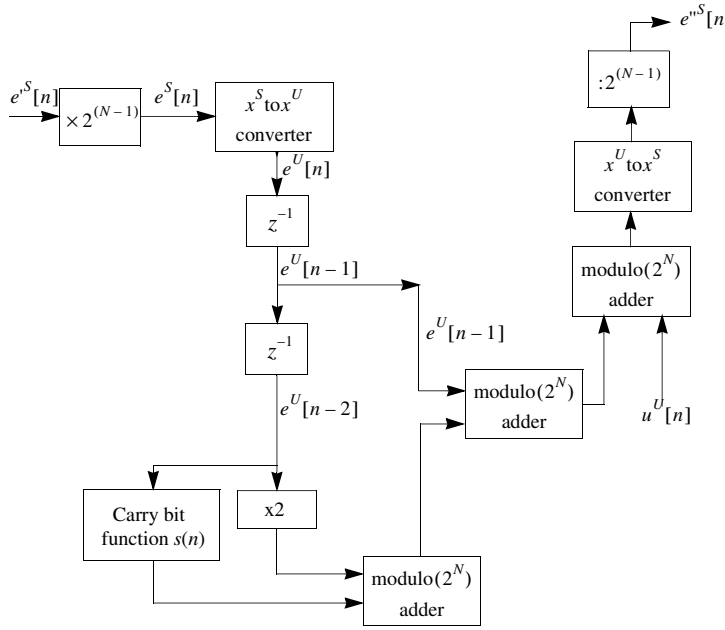


Figure 2.2. *The block scheme for the simulated chaotic signal receiver*

According to the dead-beat synchronization principle, the output signal $e'''^S[n]$ is synchronized to the transmitted chaotic signal, in a few steps (corresponding to two samples in the sequence) [DE 95].

For an error-free case, when the additive noise and transmission channel delay are ignored, the synchronized receiver sequence is identical to the transmitted sequence. For a more realistic case, when at least these two impairments must be considered, the dead-beat synchronization scheme is generating a signal with errors, where the error rate depends on the noise level. Any delay in the transmission channel is compensated by the auto-synchronizing dead-beat scheme.

Now, a very simple method for reducing the error rate became obvious. Because the chaotic sequence is a discrete multilevel sequence (the 2^N different B_i quantified levels), a multilevel threshold detection and decision can be used to recover the transmitted signal. So, before introducing the received signal into the receiver open-loop dead-beat synchronization block, a simple multilevel threshold detection and decision will decrease the symbol error rate (SER) in the received signal.

2.4.3. *The Simulink implementations for the blocks used in the Frey chaotic codec*

Both the transmitter and the receiver generators, presented in sections 2.4.1 and 2.4.2, use the modulo-2^N adder having the scheme shown in Figure 2.3. This block scheme simply follows equations [2.7] and [2.13] presented in section 2.2 with definition 2.6 for the addition operation in the 2^N-set.

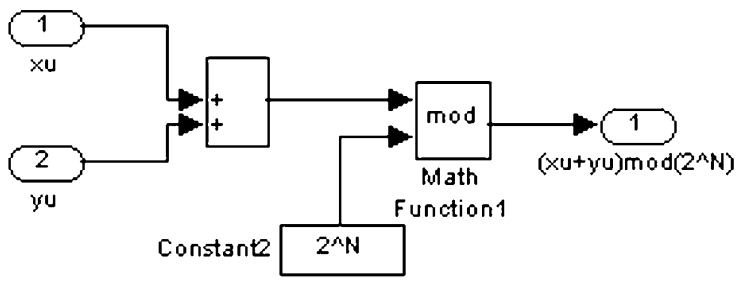


Figure 2.3. *The modulo-2^N adder block scheme*

The transmitter generator, presented in section 2.4.1, uses the principle of conversion from the unsigned x^U number representation in the 2^N-set to

signed x^S number representation in the $\left[\mathrm{C}2, 2^N\right]$-set. This conversion is made in accordance with [2.4], presented with definition 2.4, in section 2.2. The conversion scheme is shown in Figure 2.4.

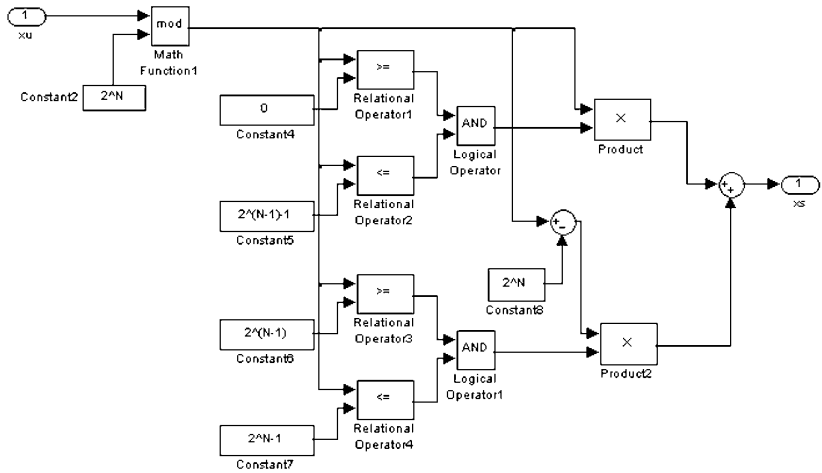


Figure 2.4. *The block scheme of the convertor from unsigned to signed representation*

The receiver generator, presented in section 2.4.2, uses the principle of conversion from the signed x^S number representation in the $\left[\mathrm{C}2, 2^N\right]$-set to the unsigned x^U number representation in the 2^N-set. This conversion is made in accordance with [2.6], presented with definition 2.5, in section 2.2. The conversion scheme is shown in Figure 2.5.

The modulo-2^N blocks at the input of the unsigned-to-signed convertor scheme (Figure 2.4) and at the output of the signed-to-unsigned convertor scheme (Figure 2.5) are used to keep the unsigned number within the definition interval $\left[0, 2^N - 1\right]$. The block denoted as "carry bit function $s[n]$", used in both transmitter and receiver generators, is in fact a block that verifies the condition for adding the carry bit and generates this bit sequence correspondingly, according to the LCIRC definition presented in sections 1.1.3 and 2.1. The signal $s[n]$ is generated in accordance with [1.5] and the block scheme used to simulate this function is shown in Figure 2.6.

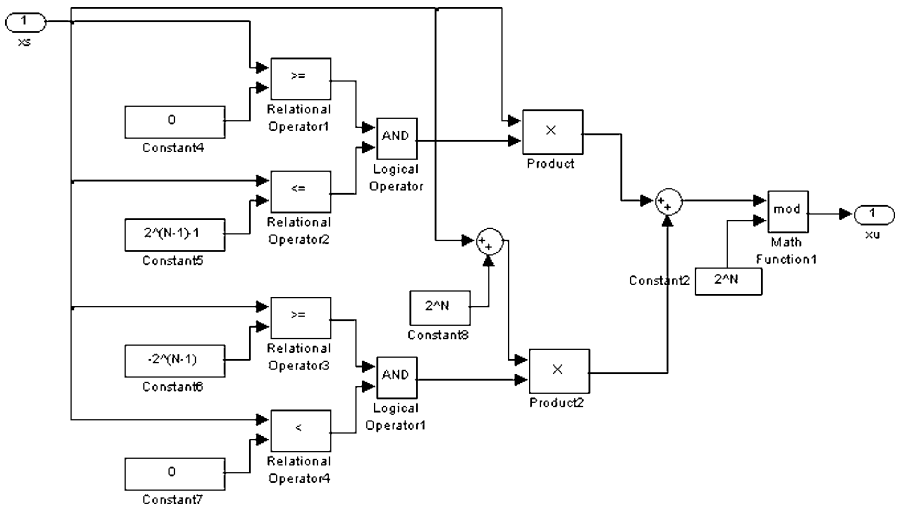


Figure 2.5. *The block scheme of the convertor from signed to unsigned representation*

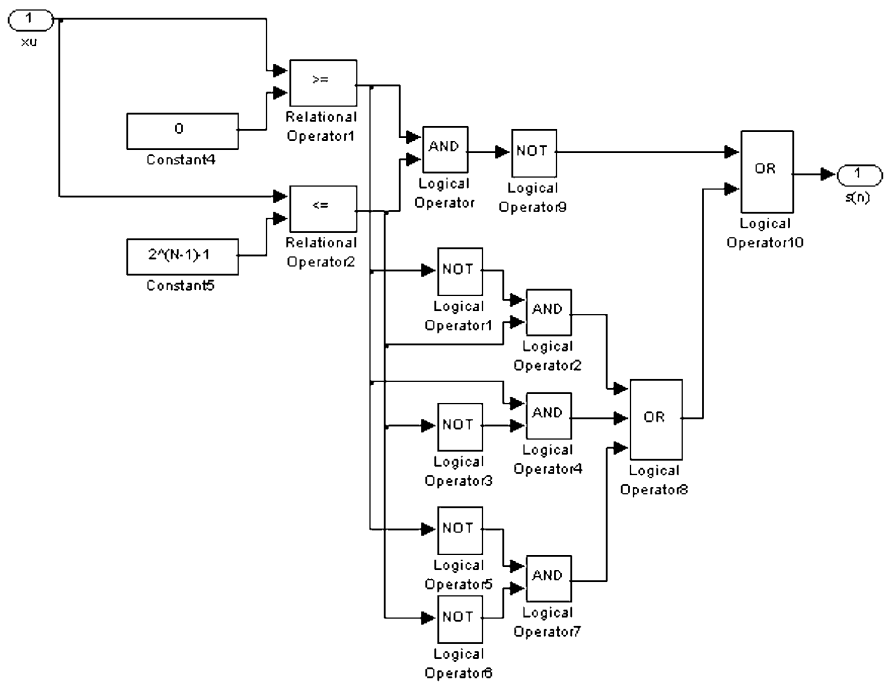


Figure 2.6. *The block scheme of carry bit function $s[n]$ generator*

3

Trellis-Coded Modulation Schemes Using Nonlinear Digital Encoders

3.1. The presentation of the Frey nonlinear encoder as a convolutional encoder

It is very interesting to have a different look over the encoder structure in Figure 1.3 (section 1.1.3). Let us consider the simplest case, when one bit for the sample representation is used, i.e. $N = 1$.

Also, let us denote the state of the encoder as the concatenation of the binary representations for the samples pair $\left(x_1^U[n], x_2^U[n]\right)$, considering the left one as the least significant bit. Then, the encoder may pass through four different states given the distinct values for the binary sample pairs $(0, 0)$, $(0, 1)$, $(1, 0)$ and $(1, 1)$.

Denoting these states by their decimal converted values, we can represent the trellis diagram in Figure 3.1 for the Frey encoder transitions from sample time interval n to the interval $n + 1$. As in the legend of Figure 3.1, transitions determined by the binary unsigned input $u^U[n]$ are represented differently. Hence, transitions determined by the unsigned input $u^U[n] = 0$ are represented with dashed lines, while transitions determined by the unsigned input $u^U[n] = 1$ are represented with continuous lines. The values on top of each transition are the corresponding signed output values $e^S[n] \in \{-1, 0\}$.

It can be easily noted that the minimum Euclidean distance of this code is obtained for the following pair of trellis paths (path 2 diverges from the state 0

and then converges back to state 0), presented as the sequences of states along the paths:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 0\ 0\ 0\ 0\\ &\text{Path 2 (trellis states)} : 0\ 1\ 2\ 0\end{aligned}$$

The Euclidean distance between these two paths, for the trellis in Figure 3.1, is $d_E^2 = 1 + 0 + 0 = 1$, which offers no coding gain over the non-encoded binary signed 2's complement binary pulse amplitude modulated (PAM) signal.

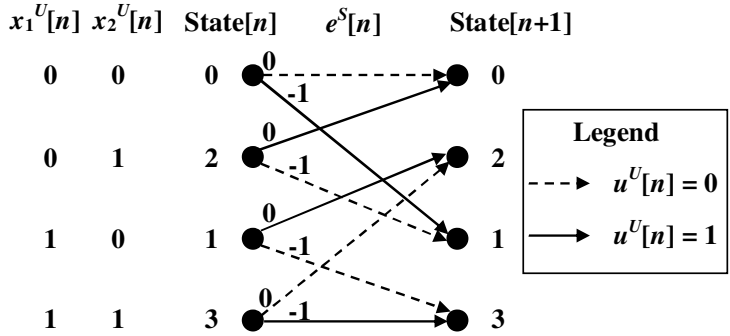


Figure 3.1. *Trellis for binary (N = 1) Frey encoder*

For comparison, we can also analyze a structure derived from the Frey encoder by eliminating the nonlinear left-circulate (LCIRC) function block in Figure 1.3. The resulting codec is the linear version of the Frey encoder, being composed of a linear infinite impulse response (IIR) filter as encoder and a linear IIR filter as decoder. Obviously, this codec offers almost no security considering the lack of the quasi-chaotic properties. The linear codec is shown in Figure 3.2.

It can be noted that for a binary representation ($N = 1$), the structure has an identical trellis to the Frey encoder trellis, shown in Figure 1.3. This is determined by the fact that the LCIRC function is the identity function when working over the binary field, i.e. $LCIRC\left(x^U[n]\right) = x^U[n]$, for $x^U[n] \in \{0, 1\}$. Hence, besides the lack of quasi-chaotic properties, the linear codec also offers no coding gain over the non-encoded binary signed 2's

complement PAM signal. The reason for considering this linear encoder is to demonstrate the superiority of the Frey nonlinear structure over its linear version for all considered examples.

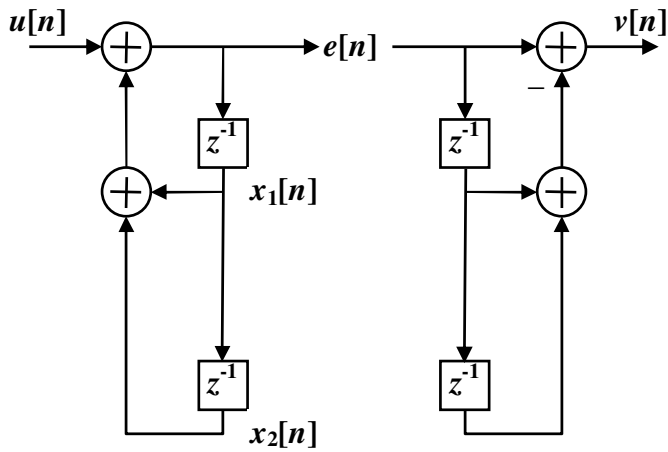


Figure 3.2. *Linear codec*

Now, let us consider a more complex case, when the word length is $N = 2$. Using the same notations as above, the trellis diagram in Figure 3.3. is obtained. This trellis has 16 states because the sample values determining the encoder states take four values each, i.e. $x_1^U[n], x_2^U[n] \in \{0, 1, 2, 3\}$. In Figure 3.3, four different lines are used for representing the transitions corresponding to the input sample $u^U[n]$. Each transition in Figure 3.3 is associated with a 2's complement signed output value $e^S[n] \in \{-2, -1, 0, 1\}$. For each originating state, the values in the box, from left to right, are associated with the transitions in the descending order.

It can be easily noted that the minimum Euclidean distance of this code is obtained for the following pair of trellis paths (path 2 diverges from the state 0 and then converges back to state 0), presented as the sequences of states along the paths:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 0\ 0\ 0\ 0\\ &\text{Path 2 (trellis states)} : 0\ 1\ 8\ 0\end{aligned}$$

Hence, despite the complex trellis represented in Figure 3.3, the Euclidean distance between these two paths is $d_E^2 = 1 + 0 + 0 = 1$, which offers no

coding gain over the non-encoded binary signed 2's complement PAM signal. In fact, the signed quaternary output signal $e^S[n]$ has a higher mean power than the binary signal. Therefore, in order to make a fair performance comparison, the power of the transmitted signed signal $e^S[n]$ has to be normalized, for each value of N.

Let us consider the signed 2's complement signal for any value of N. Hence, the signed signal takes equally probable values $x^S[n] \in \{-2^{N-1}, -2^{N-1}+1, \ldots, -1, 0, 1, \ldots, 2^{N-1}-1\}$. The mean and the variance of the signed representation signal take the following values:

$$\begin{aligned} &\overline{x^S[n]} = -\tfrac{1}{2} \\ &\sigma^2_{x^S[n]} = \overline{(x^S[n])^2} - \left(\overline{x^S[n]}\right)^2 = \tfrac{2^{2N-1}+1}{6} - \tfrac{1}{4} = \tfrac{\left(2^{2N}-1\right)}{12} \end{aligned} \qquad [3.1]$$

In the following, the coding performance of these trellises will be analyzed using Ungerboeck's rules [UNG 82], considering that all the signals are normalized by their actual standard deviation in [3.1]. It can be noted that the minimum distance pair of paths in the Frey encoder trellis differs in only three transitions, and for the last two transitions, the individual distance is null for any value of N. Considering the above normalization, the distance between the two paths in the first transition takes the minimum value from the signal set distances, i.e. $1/\sigma^2_{x^S[n]} = 12/\left(2^{2N}-1\right)$. Hence, the second rule of Ungerboeck is not fulfilled, i.e. transitions originating from the same state do not receive signals with a maximum distance. It can be noted that the output signal is equal (in the unsigned representation) to the least significant state N-tuple ($e^U[n] = x_1^U[n]$). Unfortunately, the last transitions, which are joining the same state, will receive identical signals from the same set, determining a null distance between them. This is in contradiction with the third rule of Ungerboeck [UNG 82]. In this case, the first rule is also never fulfilled, because the trellis is not symmetrical. Considering the fourth rule of Ungerboeck, it can be noted that there are no parallel transitions. In conclusion, the Frey encoder does not fulfill any of Ungerboeck's rules for an optimum trellis design. It must be noted that the fact that the second and the third rules are not fulfilled is not determined by the nonlinear LCIRC function, but because of the relation between the input and output of the Frey encoder and the first delay output $x_1[n]$. In fact, using any other nonlinear function instead, or even for the linear encoder, these two rules are also not fulfilled.

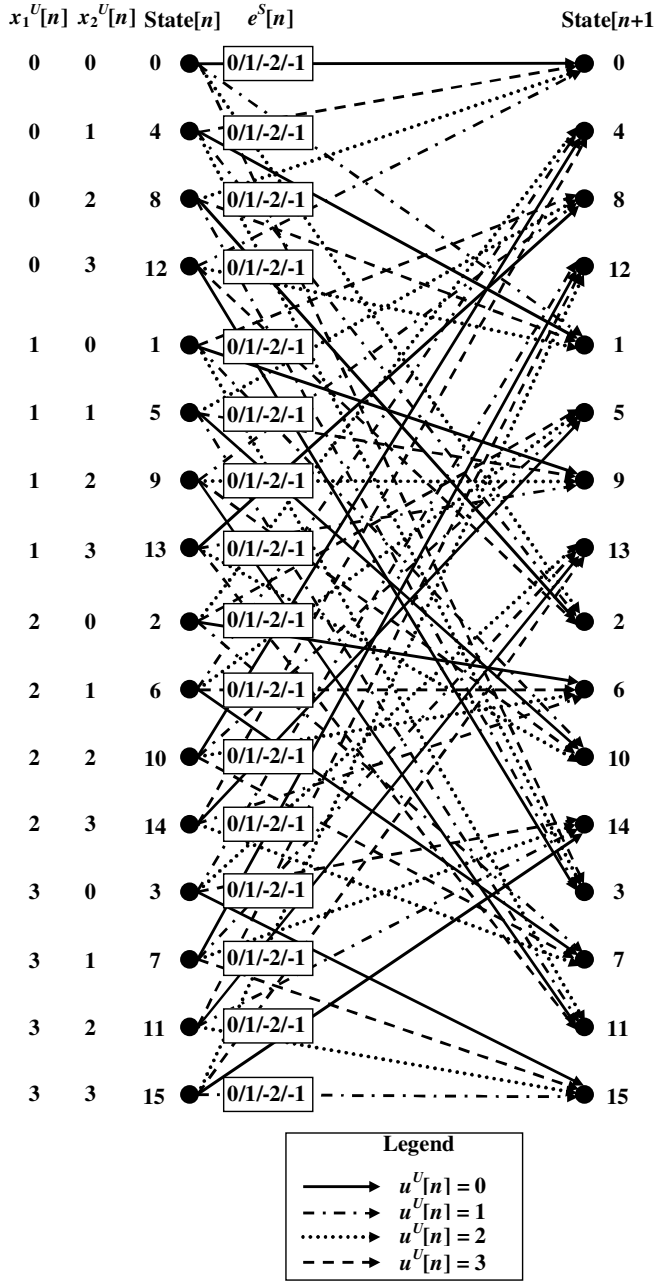


Figure 3.3. *Trellis for quaternary ($N = 2$) Frey encoder*

The binary signal trellis offers a distance of $d^2_{E,N=1,Frey} = 12/3 = 4$, while the quaternary signal trellis offers a distance of $d^2_{E,N=2,Frey} = 12/15 = 0.8$. Therefore, quaternary signal trellis of the Frey encoder performs worse than the binary one by $10log_{10}\left(d^2_{E,N=1,Frey}/d^2_{E,N=2,Frey}\right) = 10log_{10}(5) \approx 7\text{dB}$.

The linear version of the Frey encoder can be analyzed for $N = 2$. The trellis for the linear encoder is shown in Figure 3.4.

Following the same normalization procedure as above, we can compute the Euclidean distance for the linear encoder with $N = 2$, having the trellis in Figure 3.4. The minimum Euclidean distance of this code is obtained for the following pair of trellis paths presented as the sequences of states along the paths:

$$\text{Path}\,1\ (\text{trellis states}) : 0\ 0\ 0\ 0$$
$$\text{Path}\,2\ (\text{trellis states}) : 0\ 1\ 4\ 0$$

The minimum Euclidean distance for this linear encoder with $N = 2$ is: $d^2_{E,N=2,linear} = \left(\frac{1}{\sqrt{5/4}}\right)^2 + 0 + 0 = \frac{4}{5} = 0.8$, which is equal to that of the quaternary signal trellis of the Frey encoder. This means that both structures work identically and worse than the binary one.

Section 3.2 presents a method for improving the Frey encoder coding performances.

3.2. Frey encoder trellis design optimization methods for pulse amplitude – trellis-coded modulation (TCM) schemes

As it was demonstrated in section 3.1, the Frey nonlinear encoder has a trellis that does not offer any coding gain over the linear encoder and over any usual coded system with the same complexity.

To obtain certain coding gains, we developed two distinct approaches. For both cases, the goal is to design the trellis following Ungerboeck's rules of proper set partitioning [UNG 82]. For the sake of simplicity and ease of comparison with the Frey's filter, we will study only the PAM signals, represented in 2's complement form.

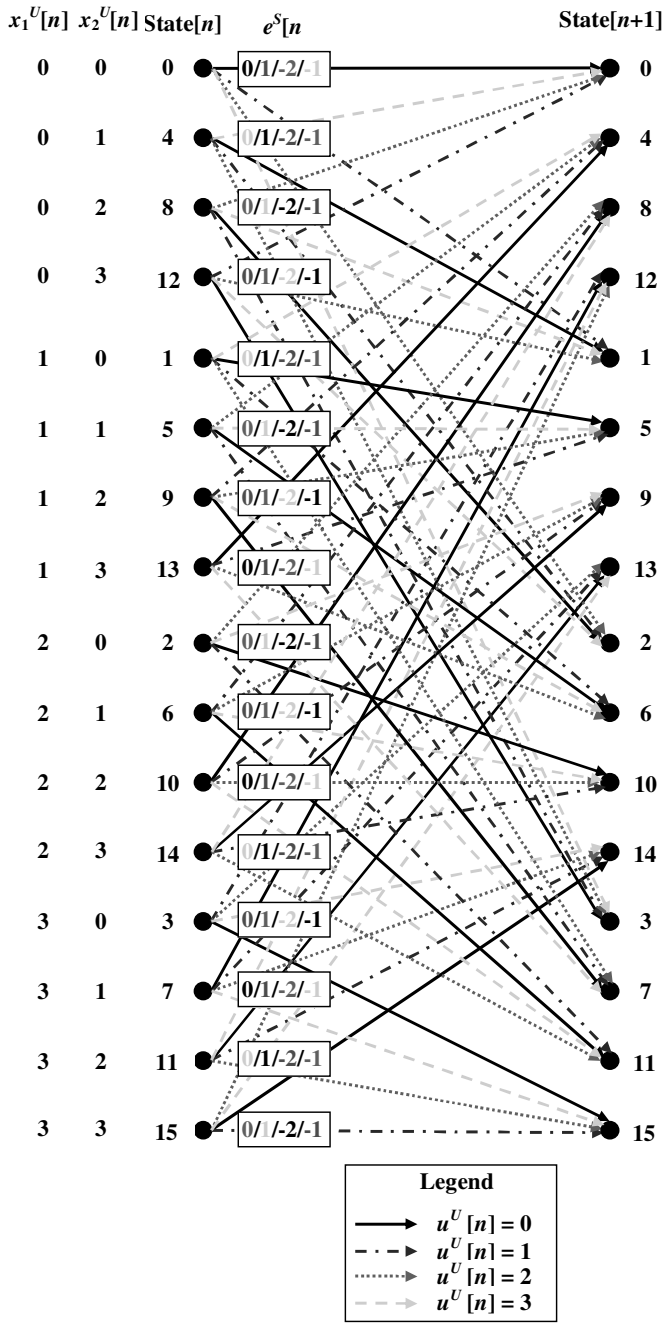


Figure 3.4. *Trellis for quaternary ($N = 2$) linear encoder*

In the first case, in order to fulfill the second Ungerboeck's rule, i.e. to have transitions originating from the same state associated with signals having a larger minimum distance, the input signal set is reduced, which is equivalent to an input word length decrease as compared to the output word length. We have to mention that in this case, the structure of the filter is not modified, and we only need different representation code words in the input and output. Therefore, the coding rate is decreased. This input word length decrease method for proper signal set design is presented in section 3.2.1. Then, in order to increase the coding gain, a proper set partitioning for the PAM signal is needed.

To prove the coding features of these nonlinear structures, the equivalency between a modified $GF(4)$ Frey encoder and an optimum conventional linear $GF(2)$ recursive and systematic convolutional (RSC) encoder is demonstrated in section 3.2.2. Even this equivalence is demonstrated for a particular case; these results can be generalized. Therefore, in section 3.2.3, a generalized block scheme for optimum PAM-TCM encoder is provided.

The second trellis optimization method is introduced in section 3.2.4. This method assumes keeping the same representation word length throughout the whole structure and modifying the encoder in order to have two outputs instead of one. Hence, an encoder having an encoding rate of $1/2$ is obtained. Also, the signal will be transmitted in the quadrature amplitude shift keying (QASK) form with 2's complement representation for both branches. Here, the coding gain is increased also by proper signal set partitioning.

3.2.1. *Increasing the coding gain by reducing the representation code word length in the input*

Now, let us analyze in depth the encoders introduced in section 3.1. Each of these schemes has a trellis that denotes a convolutional encoder similarity. For these schemes, which are in fact finite precision digital filters, we can even define an encoding rate as the ratio between the input word length N_{in} and the output word length N_{out}:

$$R_N \triangleq \frac{N_{in}}{N_{out}} \qquad [3.2]$$

Using [3.2], it results that all the codes in section 3.1 have an encoding rate equal to 1 because the input, the internal processing and the output word

lengths are equal. This is the reason why they perform this bad against noise. Hence, an obvious conclusion is that 1 can enhance the noise protection of such structures only by reducing the encoding rate. This is easily done by reducing the input word length as compared to the output 1. It is obvious that this can be done only for $N_{out} \geq 2$. For a PAM transmission, the input word length determines the spectral efficiency, i.e. N_{in} b/s/Hz. Therefore, reducing the input word length also determines a spectral efficiency decrease.

Let us consider a simple case, which assumes a decrease in the input word length from 2 bits per sample to $N_{in} = 1$ for the quaternary output ($N_{out} = N = 2$) Frey encoder. As a matter of fact, the input signal takes unsigned values from the binary set $u^U[n] \in \{0, 1\}$, while the output and the internal signals take quaternary unsigned values from the set $e^U[n] \in \{0, 1, 2, 3\}$. The resulting trellis for this rate-$1/2$ convolutional Frey encoder is shown in Figure 3.5. The changes from Figure 3.3 to Figure 3.5 consist of the elimination of the transitions determined by the input unsigned values $u^U[n] \in \{2, 3\}$ in Figure 3.3.

Let us compute the minimum Euclidean distance for two distinct paths starting in the 0 state and ending in the same 0 state, of the code in Figure 3.5, as follows:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 0\ 0\ 0\ 0\ 0\ 0\ 0\\ &\text{Path 2 (trellis states)} : 0\ 1\ 9\ 8\ 3\ 12\ 0\end{aligned}$$

The distance takes a value of: $d^2_{E,R=1/2,Frey,u^U[n]\in\{0,1\}} = \left(\frac{1}{\sqrt{5/4}}\right)^2 + \left(\frac{1}{\sqrt{5/4}}\right)^2 + 0 + \left(\frac{-1}{\sqrt{5/4}}\right)^2 + 0 + 0 = \frac{12}{5} = 2.4$, which is three times larger than the distance in trellis of the original quaternary Frey encoder.

Unfortunately, the value of the minimum distance of this trellis takes other values (some of these values are even smaller, i.e. 0.8, 1.6 and 7.2). The actual coding gain of this trellis, having different values for the Euclidean distances between paths, depending on the selected states, is difficult to estimate. The problem resides in the fact that the minimum distance depends on the initial state; so, this is not a uniform (symmetrical) trellis.

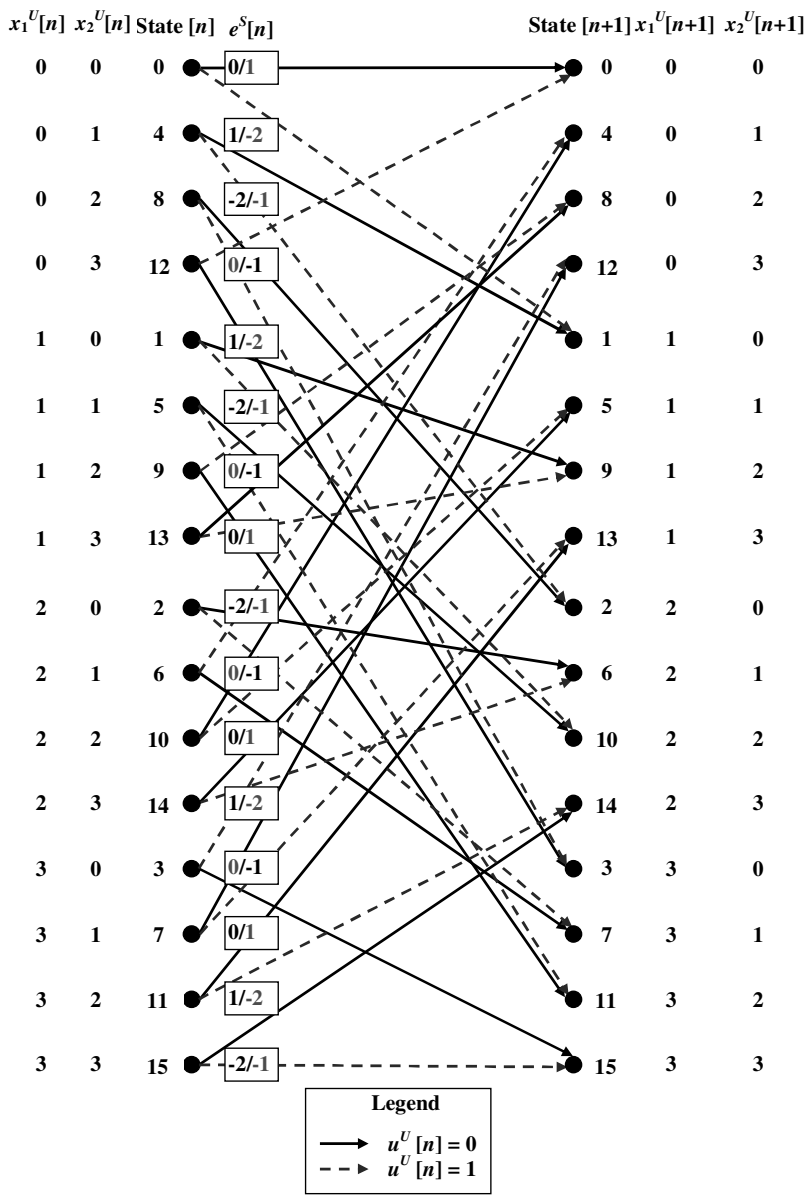


Figure 3.5. *Trellis for rate-1/2 non-optimum convolutional Frey encoder ($N_{in} = 1$, $N_{out} = 2$), $u^U[n] \in \{0, 1\}$*

However, this gain over the original quaternary Frey encoder is determined by the scarcity increase in the trellis structure, determined by the elimination of half the number of possible transitions. Therefore, this increases the minimum distance between different paths. As it is shown later, this is not the largest gain that can be achieved by the encoding rate reducing in these schemes, because the set partitioning optimization rules were not considered.

In the following, the best performance case is determined for a rate-$1/2$ Frey encoder, with the input word length of $N_{in} = 1$ and quaternary output ($N_{out} = N = 2$). For the best Frey encoder with these parameters, the input signal takes unsigned values from the binary set $u^U[n] \in \{0, 2\}$, while the output and the internal signals take quaternary unsigned values from the set $e^U[n] \in \{0, 1, 2, 3\}$. The resulting trellis for the best rate-$1/2$ Frey encoder is obtained by eliminating the transitions determined by the input unsigned values $u^U[n] \in \{1, 3\}$ from Figure 3.3. The resulting trellis for this best rate-$1/2$ convolutional Frey encoder is shown in Figure 3.6.

For the sake of clarity, let us consider the following pair of two paths, both starting in state 3 and ending in state 6:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 3\ 13\ 8\ 0\ 0\ 2\ 6\\ &\text{Path 2 (trellis states)} : 3\ 15\ 12\ 3\ 13\ 10\ 6\end{aligned}$$

The Euclidean distance between the above pair of paths of the code is determined as

$$\begin{aligned}d^2_{E,R=1/2,Frey,u^U[n]\in\{0,2\}} &= \left(\frac{2}{\sqrt{5/4}}\right)^2 + 0 + \left(\frac{1}{\sqrt{5/4}}\right)^2\\ &\quad + \left(\frac{-1}{\sqrt{5/4}}\right)^2 + 0 + 0 = 6/\left(\sqrt{5/4}\right)^2 = 4.8.\end{aligned}$$

In fact, this is the minimum value of the Euclidean distance for the code in Figure 3.6.

We noted that starting from any state, there is a unique pair of paths in the trellis ending in any state (from all 16) after six transitions, having the same minimum distance. Hence, we can say that this is a uniform (symmetrical) trellis. In fact, this gain is explained by the proper set partitioning of the

output signal set. The selected transitions in the trellis ($u^U[n] \in \{0, 2\}$) will determine the output signal pairs, for the transitions starting from the same state, to have the maximum distance between them. In the 2's complement representation, the output pairs of maximum distance are $(1, -1)$ and $(0, -2)$. The Euclidean distance increases at least by a factor of 4 as compared to the distance of the original quaternary Frey encoder. Hence, the second rule of Ungerboeck is fulfilled, i.e. transitions originating from the same state have signals assigned from sets with a maximum distance between the elements. In addition, we can note that the output signal is equal (in the unsigned representation) to the least significant state tuple ($e^U[n] = x_1^U[n]$). For example, all transitions having the $e^S[n] = -2$ output signal assigned will converge in a state having $x_1^U[n] = 2$. Unfortunately, the transitions joining the same state will receive identical signals from the same set. This is in contradiction with the third rule of Ungerboeck. We can note that these encoders do not fulfill this rule by *de facto*. To end the analysis of this structure following the rules of Ungerboeck, we have to mention that the first rule is never fulfilled in the reduced input code word approach, because the trellis is no longer symmetrical (see Figure 3.6). This fact also reduces the coding gain of the structure. Considering the fourth rule of Ungerboeck, we must note that we cannot have parallel transitions in this structure because the encoder is not systematic (we do not have unfiltered samples) at the output.

The best rate-$1/2$ ($N_{in} = 1, N_{out} = 2$) Frey encoder performs better than the quaternary Frey encoder, having a coding gain of $10log_{10}\left(d^2_{E,R=1/2,Frey,u^U[n]\in\{0,2\}}/d^2_{E,N=2,Frey}\right) = 10log_{10}(6) \approx 7.78\text{dB}$, which outperforms the signed binary output signal trellis by $10log_{10}(4.8/4) \approx 0.79\text{dB}$.

As in section 3.1, let us consider the linear rate-$1/2$ convolutional linear encoder ($N_{in} = 1, N_{out} = 2$), following the same set partitioning optimization procedure as for the similar Frey encoder, i.e. the set of selected inputs is $u^U[n] \in \{0, 2\}$. In this case, the trellis is less efficient than that of the Frey encoder, having a smaller minimum Euclidean distance. The resulting trellis for the rate-$1/2$ linear encoder with input unsigned samples values $u^U[n] \in \{0, 2\}$ is shown in Figure 3.7.

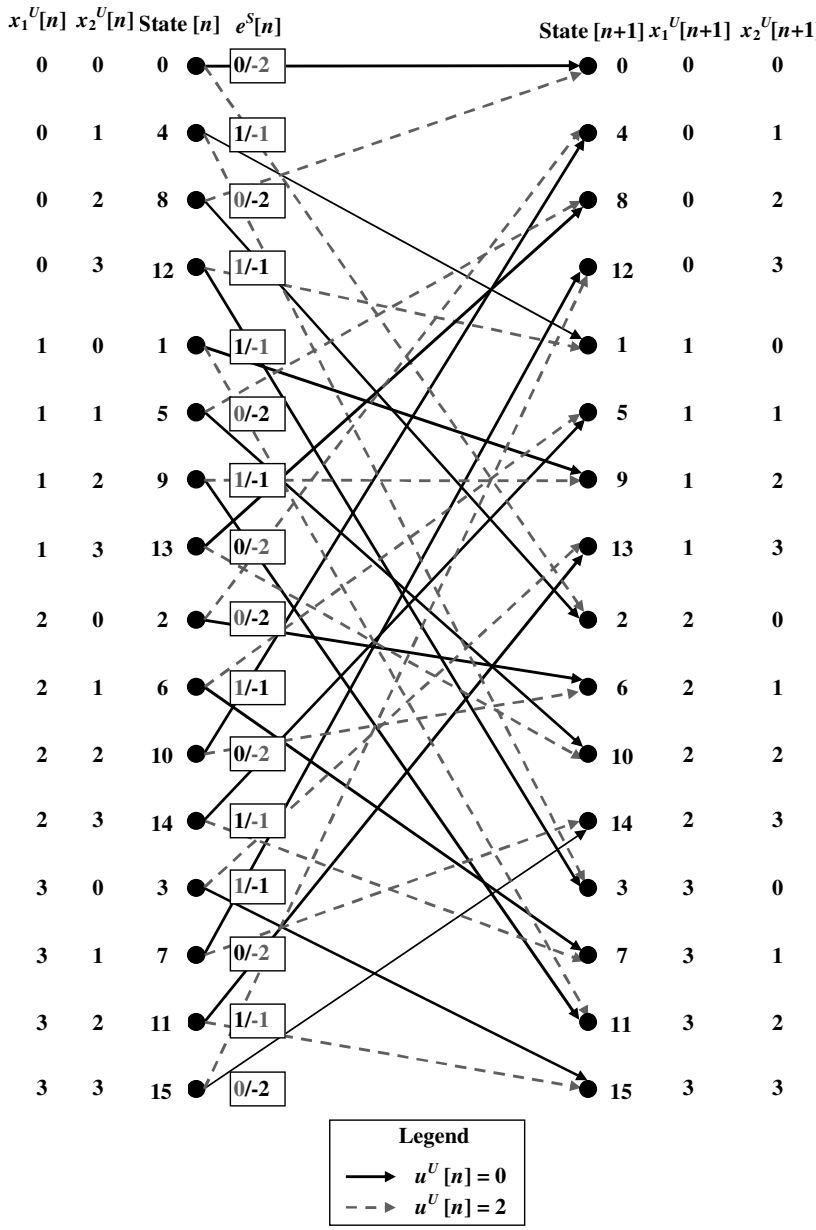


Figure 3.6. *Trellis for rate-1/2 optimum convolutional Frey encoder ($N_{in} = 1$, $N_{out} = 2$), $u^U[n] \in \{0, 2\}$*

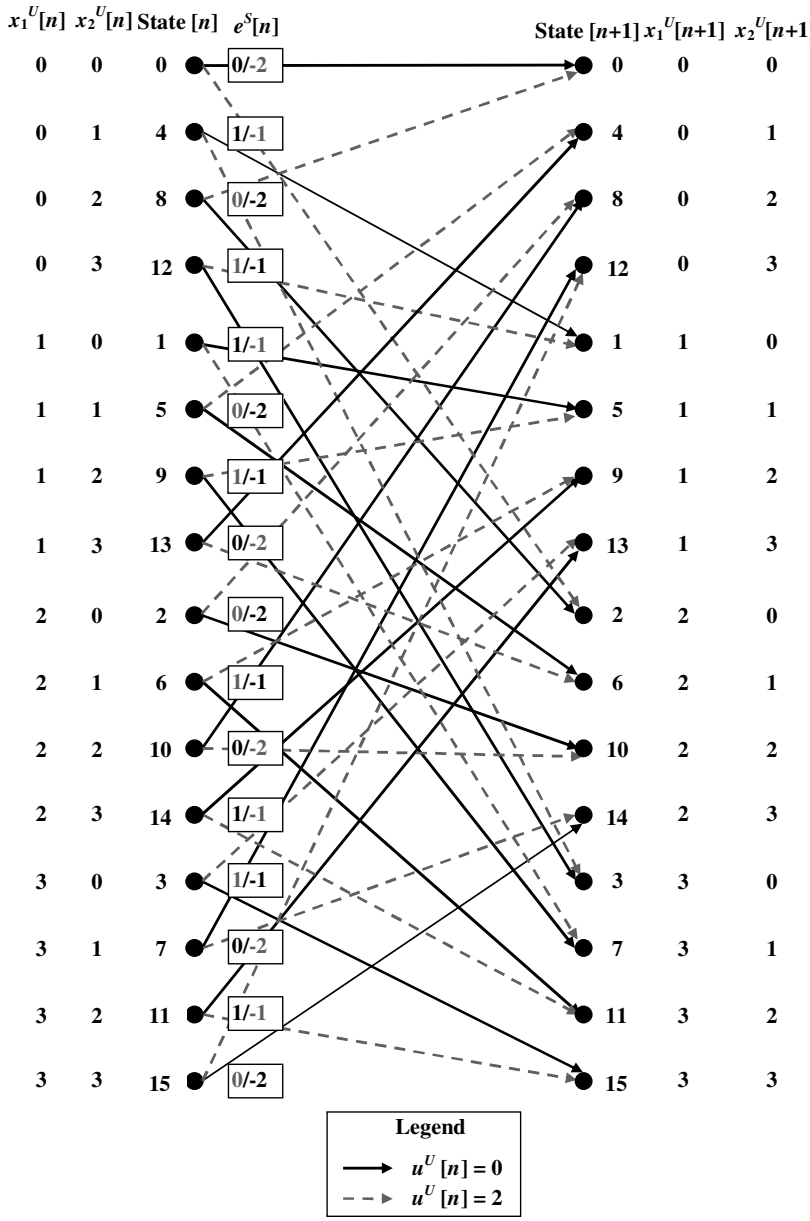


Figure 3.7. *Trellis for rate-1/2 convolutional linear encoder ($N_{in} = 1$, $N_{out} = 2$), $u^U[n] \in \{0, 2\}$*

Comparing the trellises in Figures 3.6 and 3.7, we can easily see that the absence of the LCIRC block affects transitions (changes the next state) originating from states having $x_1^U[n] \in \{1, 2\}$ (initial states placed in the middle) and it does not change the output values $e^S[n]$ associated with a specific initial state. This is explained by the fact that the LCIRC function changes only the value of $x_2^U[n+1]$ without changing the other variables (see the linear encoder scheme in Figure 3.2).

For example, let us consider the following pair of two paths for the trellis in Figure 3.7, having the minimum Euclidean distance, both starting in state 3 and ending in state 1:

$$\text{Path 1 (trellis states)} : 3\ 15\ 12\ 1$$
$$\text{Path 2 (trellis states)} : 3\ 13\ 4\ 1$$

The Euclidean distance between these paths is: $d^2_{E,R=1/2,linear,u^U[n]\in\{0,2\}} = \left(\frac{-2}{\sqrt{5/4}}\right)^2 + \left(\frac{2}{\sqrt{5/4}}\right)^2 + 0 = 4/\left(\sqrt{5/4}\right)^2 = 3.2$, which is four times larger than the distance in the trellis of the original quaternary linear encoder. The same result is obtained for the input set $u^U[n] \in \{1, 3\}$. However, the rate-$1/2$ convolutional linear encoder in Figure 3.7 performs worse than the corresponding rate-$1/2$ convolutional Frey encoder, having an encoding loss of $10log_{10}\left(d^2_{E,R=1/2,Frey,u^U[n]\in\{0,2\}} \,/\, d^2_{E,R=1/2,linear,u^U[n]\in\{0,2\}}\right) = 10log_{10}(6/4) \approx 1.76$ dB. When compared to the signed binary output signal trellis, we also have a loss of $10log_{10}(4/3.2) \approx 0.97$ dB. Using the inverse system proposed by Frey for the sample decoding [FRE 93], the same performance is noted as for the rate 1, $N = 2$ encoder.

We also studied the larger word length encoders, but the results were not satisfactory. The complexity increase in the trellis (the number of states) is not really exploited in the Frey encoder case. Actually, we can generalize the Frey encoder trellis' parameters for the scheme in Figure 1.3 (section 1.1.3) for any pair of values for the input and output representation code words, i.e. $1 < N_{in} \leq N_{out} = N$.

Now, let us consider a rate-$1/2$ convolutional Frey encoder, having $N_{in} = 2$ bits per sample in the input and a 16 levels output signal ($N_{out} = 4$). Using

the above-mentioned set partitioning method, we select the quaternary set for the input signal, i.e., $u^U[n] \in \{0, 4, 8, 12\}$, while the output and the internal signals take 16 unsigned values from the set $e^U[n] \in \{0, 1, 2, \ldots, 15\}$. The resulting trellis for the rate-$1/2$ convolutional Frey encoder has 256 states, and its graphical analysis is too difficult. However, we can compute the minimum Euclidean distance of the trellis as for the previous cases. Let us consider the following pair of paths, both starting in state 1 and ending in state 105:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 1\ 33\ 35\ 105\\ &\text{Path 2 (trellis states)} : 1\ 37\ 163\ 105\end{aligned}$$

This trellis offers a minimum distance, also determined between the above considered two paths, with the value: $d^2_{E,R=1/2,Frey,u^U[n]\in\{0,4,8,12\}} = \left(\frac{-4}{\sqrt{85/4}}\right)^2 + 0 + 0 = 16/\left(\sqrt{85/4}\right)^2 \approx 0.753$. The same result is obtained, for example, when the input set is either $u^U[n] \in \{1, 5, 9, 13\}$, $u^U[n] \in \{2, 6, 10, 14\}$ or $u^U[n] \in \{3, 7, 11, 15\}$.

We can note that rate-$1/2$ convolutional Frey encoder ($N_{in} = 2$, $N_{out} = 4$) is performing worse than the rate-$1/2$ convolutional Frey encoder ($N_{in} = 1$, $N_{out} = 2$), the last one having an encoding gain of $10log_{10}\left(d^2_{E,R=1/2,Frey,u^U[n]\in\{0,2\}}/d^2_{E,R=1/2,Frey,u^U[n]\in\{0,4,8,12\}}\right) = 10log_{10}\left(\frac{4.8}{64/85}\right) \approx 8.05$ dB .

The conclusion here is that, even if a good signal set partitioning was used, for larger sizes of representation word length, the output signal set increases exponentially with $2N_{out}$, which reduces the minimum distance of the $2^{N_{out}}$-PAM signal symbols accordingly. The Frey encoder trellis cannot overcome this reduction in the protection against noise. As noted in the previous example, even if the distance between the paths is 16 times larger than the distance in the non-encoded signal, the exponential decrease in the later symbol distance also reduces the whole trellis minimum Euclidean distance. Also, this happens due to the lack of symmetry in the trellis and null distance over the transitions converging into the same state.

We can conclude here that the Frey encoder cannot offer better performances for higher word lengths in the output, and therefore the binary and optimum quaternary cases remain the best encoders.

Using the same set partitioning, the encoding gain of 6 dB is obtained for any value of N as presented in Table 3.1. For $N = 2$, the minimum distance is increased by a factor of 6, while for $N \geq 2$ the factor is 4.

N	$d^2_{E,N,Frey} = 12/(2^{2N}-1)$	$d^2_{E,R=1/2,N,Frey}/d^2_{E,N,Frey}$
1	4	–
2	0.8	6
3	0.1905	4
4	0.0471	4
5	0.0117	4
6	0.0029	4

Table 3.1. *Coding gains as function of N for rate-*$1/2$ *Frey encoders*

All the above notes were made considering the same value for N as a reference. A more fair comparison assumes the spectral efficiency criterion. As it is presented in Table 3.1, the rate-$1/2$ Frey encoder using an output word length of N and the rate 1 Frey encoder using an word length of $N-1$ offer the same spectral efficiency, i.e. $(N-1)$ b/s/Hz. For $N \geq 3$, the encoding gain of the first encoder as compared to the second encoder is given by:

$$10log_{10}\left(\frac{d^2_{E,R=1/2,N,Frey}}{d^2_{E,N-1,Frey}}\right) = 10log_{10}\left[\frac{2^{2N}-4}{2^{2N}-1}\right] < 0 \qquad [3.3]$$

Hence, for $N \geq 3$, the rate-$1/2$ Frey encoder performs worse than the rate-1 Frey encoder for the same spectral efficiency. For $N \to \infty$, the coding loss in [3.3], rapidly increases to 0, meaning that these two encoders perform almost the same for large values of N.

In fact, only for $N = 2$, a coding gain is obtained, i.e. $10log_{10}\left[\left(3\cdot 2^{2N-1}-6\right) \,/\, \left(2^{2N}-1\right)\right]\big|_{N=2} = 10log_{10}(6/5) \approx 0.79$ dB. Therefore, using the Viterbi decoder, a coding gain of approximately 0.79 dB is expected, for the same spectral efficiency PAM transmission, i.e. 1/b/s/Hz. However, this value is significantly lower than the coding gain offered by the optimum $GF(2)$ 16 states encoder [UNG 82].

As a conclusion, from the spectral efficiency point of view, only the quaternary rate-$1/2$ Frey encoder ($N_{out} = N = 2$) offers an encoding gain.

Increasing the word length for the rate-$1/2$ Frey encoder determines a coding performance decrease as compared with the corresponding rate-1 Frey encoder. The reason for this drawback is that the complexity increase in the trellis is not really exploited by the modified Frey encoders with larger word lengths.

Another drawback of the word length increase is the trellis complexity itself because the number of trellis states grows exponentially with the output word length, i.e. 2^{2N}, while the number of transitions originating from and ending in the same state grows exponentially with the input word length, i.e. $2^{2N_{in}}$. All these drawbacks are caused by the encoder structure because the Frey encoder was used, without performing any other changes.

3.2.2. *Equivalence between a nonlinear and a linear encoder in a particular case*

As demonstrated in sections 3.1 and 3.2.1, in order to obtain significant coding rates, the Frey encoder scheme has to be significantly changed. In fact, the block that determines the coding gains of all presented schemes over their linear versions is the LCIRC block. In this section, the potential of the nonlinear LCIRC function is shown for designing efficient encoders. Therefore, following the trellis optimization presented in section 3.2.1, a simple nonlinear encoder operating over $GF(4)$ was developed, which has a binary input. It is demonstrated that this encoder performs identically to an optimum rate-$1/2$ binary field RSC convolutional encoder. Both encoders offer maximum coding gain for 1 b/s/Hz [UNG 82].

Let us consider a first-order nonlinear digital filter working over $GF(4)$, which is a simplified version of the Frey encoder. This scheme is shown in Figure 3.8. Here, the time variable is neglected, and we introduce an exponent "$c1$" denoting the first encoder from the proposed analysis. All the values are represented in the unsigned form. The trellis for the encoder in Figure 3.8 follows all Ungerboeck's rules.

THEOREM 3.1.– The encoder in Figure 3.8 performs identically with the rate-$1/2$ binary field RSC convolutional encoder in Figure 3.9.

For the sake of clarity, the time variable notation is also neglected and an exponent "$c2$" is introduced, denoting the second encoder to be analyzed.

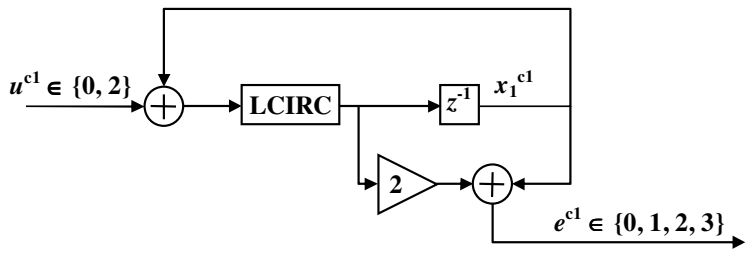


Figure 3.8. *Rate-1/2 GF(4) nonlinear encoder for 1 b/s/Hz*

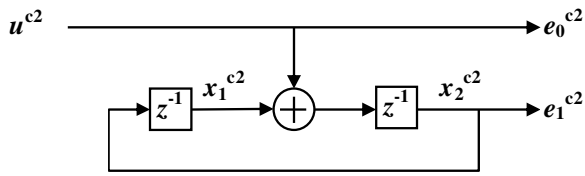


Figure 3.9. *Rate-1/2 GF(2) RSC linear encoder for 1 b/s/Hz*

Let us make the following notations and assumptions:

$$\begin{array}{l} u^{c1} = 2u^{c2} = \left(u^{c2};0\right)_2 ; \left(x_1^{c1}\right)_4 = \left(x_1^{c2};x_2^{c2}\right)_2 \\ \left(e^{c1}\right)_4 = \left(e_0^{c2};e_1^{c2}\right)_2 \end{array} \quad [3.4]$$

where $(.)_N$ denotes the N-base numerical representation, having the most significant element on the leftmost position.

Using the relations from [3.4] with the $GF(2)$ and $GF(4)$ operators properties, it can be easily demonstrated that these two encoders are equivalent, having the same trellis. Here, the Euclidean distance is $d^2_{E,R=1/2,opt.,u^U[n]\in\{0,2\}} = 9/\left(\sqrt{5/4}\right)^2 = 7.2$. Hence, this rate-$1/2$ code for 1b/s/Hz PAM transmission offers a coding gain of $10log_{10}\left(3/2\right) = 1.76$ dB over the rate-$1/2$ Frey encoder in section 3.2.1.

DEMONSTRATION 3.1.– Analyzing the scheme in Figure 3.8, we can write the following relations between the input, output and state variables:

$$\begin{array}{l} x_1^{c1}\left[n\right] = LCIRC\left(u^{c1}\left[n-1\right] \oplus_4 x_1^{c1}\left[n-1\right]\right) \\ e^{c1}\left[n\right] = 2 \cdot LCIRC\left(u^{c1}\left[n\right] \oplus_4 x_1^{c1}\left[n\right]\right) \oplus_4 x_1^{c1}\left[n\right] \end{array} \quad [3.5]$$

where " $\cdot$ " denotes the simple product (algebraic), the LCIRC function is expressed as in definition 1.2 (section 1.1.3) with the properties presented in section 2.3, and the "$\oplus_M$" operator is the modulo-M adder given by definition 2.6 (section 2.2).

From the scheme in Figure 3.9, we have the following relations between the input, output and state variables:

$$\begin{array}{l} x_1^{c2}[n] = x_2^{c2}[n-1] \\ x_2^{c2}[n] = x_1^{c2}[n-1] \oplus_2 u^{c2}[n-1] \\ e_0^{c2}[n] = u^{c2}[n] \\ e_1^{c2}[n] = x_2^{c2}[n] \end{array} \qquad [3.6]$$

Introducing [3.4] and [3.6] in [3.5], the latter can be rewritten as:

$$\begin{array}{l} \left(x_1^{c2};\, x_2^{c2}\right)_2[n] = \\ LCIRC\{\left(u^{c2}[n-1] \oplus_2 x_1^{c2}[n-1] \oplus_2 \left(0 \cdot x_2^{c2}[n-1]\right);\right. \\ \left.\left(0 \oplus_2 x_2^{c2}[n-1]\right)\right)_2\} = \\ \left(x_2^{c2}[n-1];\, u^{c2}[n-1] \oplus_2 x_1^{c2}[n-1]\right)_2 \Rightarrow \\ \qquad \left\{ \begin{array}{l} x_1^{c2}[n] = x_2^{c2}[n-1] \\ x_2^{c2}[n] = u^{c2}[n-1] \oplus_2 x_1^{c2}[n-1] \end{array} \right. \\ \left(e_0^{c2};\, e_1^{c2}\right)_2[n] = \\ 2 \cdot LCIRC\{\left(u^{c2}[n] \oplus_2 x_1^{c2}[n] \oplus_2 \left(0 \cdot x_2^{c2}[n]\right);\right. \\ \left.\left(0 \oplus_2 x_2^{c2}[n]\right)\right)_2\} \oplus_4 \left(x_1^{c2};\, x_2^{c2}\right)_2[n] = \\ \left(x_2^{c2}[n];\, u^{c2}[n] \oplus_2 x_1^{c2}[n]\right)_2 \oplus_4 \left(x_2^{c2}[n];\right. \\ \left. u^{c2}[n] \oplus_2 x_1^{c2}[n]\right)_2 \oplus_4 \left(x_1^{c2};\, x_2^{c2}\right)_2[n] = \\ \left(x_2^{c2}[n] \oplus_2 x_2^{c2}[n] \oplus_2 \left(\left(u^{c2}[n] \oplus_2 x_1^{c2}[n]\right) \cdot\right.\right. \\ \left.\left.\left(u^{c2}[n] \oplus_2 x_1^{c2}[n]\right)\right);\right. \\ \left. u^{c2}[n] \oplus_2 x_1^{c2}[n] \oplus_2 u^{c2}[n] \oplus_2 x_1^{c2}[n]\right)_2 \oplus_4 \\ \oplus_4 \left(x_1^{c2};\, x_2^{c2}\right)_2[n] = \\ \left(\left(u^{c2}[n] \oplus_2 x_1^{c2}[n]\right) \cdot \left(u^{c2}[n] \oplus_2 x_1^{c2}[n]\right) \oplus_2 x_1^{c2}[n] \oplus_2\right. \\ \left.\left(0 \cdot x_2^{c2}[n]\right);\, \left(0 \oplus_2 x_2^{c2}[n-1]\right)\right)_2 \Rightarrow \\ \qquad \left\{ \begin{array}{l} e_0^{c2}[n] = u^{c2}[n] \\ e_1^{c2}[n] = x_2^{c2}[n] \end{array} \right. \end{array} \qquad [3.7]$$

In [3.7], the relations given in [3.8] between the "$\oplus_4$", "$\oplus_2$" and "$\cdot$" operators were used. If $a \oplus_4 b = c$, where $a, b, c \in \{0, 1, 2, 3\}$ and denoting

$a = (a_1; a_2)_2$, $b = (b_1; b_2)_2]$ and $c = (c_1; c_2)_2$, with $a_1, a_2, b_1, b_2, c_1, c_2 \in \{0, 1\}$, the following relations result in:

$$\begin{cases} c_1 = a_1 \oplus_2 b_1 \oplus_2 (a_2 \cdot b_2) \\ c_2 = a_2 \oplus_2 b_2 \end{cases} \quad [3.8]$$

In conclusion, the equations on right side of [3.7] demonstrate the equivalency between the encoders in Figures 3.8 and 3.9.

$q.e.d.$

3.2.3. *Generalized optimum encoder for a PAM-TCM transmission*

Let us consider an RSC encoder working over $GF(4)$ using the LCIRC function. This scheme is shown in Figure 3.10. Here, all the values are represented in the unsigned form. Let us assume that N denotes the word length used for binary representation of each sample. This encoder is composed of one delay element with a sample interval, two modulo-2^N adders, one modulo-2^N multiplier by a constant factor 2 and an LCIRC block. For each moment n, $u[n]$ represents the input data sample, $x_1[n]$ denotes the delay output or the encoder current state and $e[n]$ is the output sample.

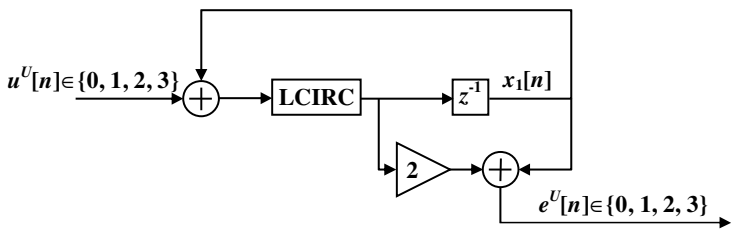


Figure 3.10. *Rate-1 GF(4) nonlinear encoder for 2 b/s/Hz*

The encoding rate for the encoder in Figure 3.10 is the ratio between the input word length N_{in} and the output word length $N = N_{out}$ [VLA 09a], i.e. $R = 1$, because $N_{in} = N_{out} = 2$.

The trellis for the encoder in Figure 3.10 is shown in Figure 3.11 and does not follow Ungerboeck's rules [UNG 82, VLA 09a]. This trellis has four

states because the sample determining the encoder state takes four values, i.e. $x_1^U[n] \in \{0, 1, 2, 3\}$. In Figure 3.11, four different lines are used for representing the transitions corresponding to the input sample $u^U[n]$. Each transition in Figure 3.11 is associated with an unsigned output value $e^U[n] \in \{0, 1, 2, 3\}$. For each originating state, the values in the box, from left to right, are associated with the transitions in the descending order.

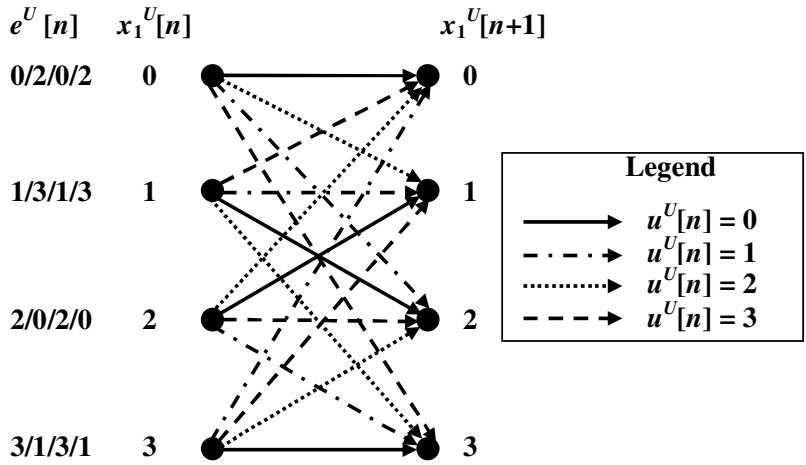


Figure 3.11. *Trellis for rate-1 GF(4) nonlinear encoder (2 b/s/Hz)*

In [FRE 93], Frey proposed a 2's complement representation form for all the computations in the scheme. For ease of comparison, the 2's complement representation will be considered for all PAM signals [VLA 09a, VLA 09b]. As in the previous sections, in the following, it will be considered for the performance analysis that all the signals are normalized by their actual standard deviation in [3.1]. In addition, the actual coding gain of the signed quaternary output signal over the signed binary output signal is estimated. Considering the above mentioned above-mentioned normalization, the quaternary signal trellis in Figure 3.11 presents a minimum Euclidean distance of $d^2_{E,R=1,N=2} = \left(1/\sqrt{5/4}\right)^2 = 0.8$, offering no coding gain over the non-encoded binary signed 2's complement PAM signal.

Next, a different trellis design method is presented, which again shows the potential of the nonlinear LCIRC function. Therefore, following the trellis optimization presented in [VLA 09a], a simple nonlinear encoder operating

over $GF(4)$ was developed, which has a binary input. It is demonstrated that this encoder performs identically to an optimum rate-$1/2$ binary field RSC convolutional encoder. Both encoders offer maximum coding gain for 1 b/s/Hz [UNG 82, VLA 09a]. The scheme of the rate-$1/2$ optimum $GF(4)$ encoder is shown in Figure 3.12. In fact, this scheme has already been shown in Figure 3.8, but is reproduced here for an easier understanding of the new concept. However, in Figure 3.12, the time variable is neglected and all the values are represented in the unsigned form.

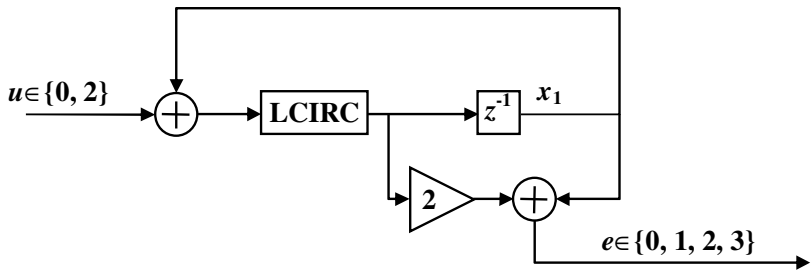


Figure 3.12. *Rate-1/2 optimum GF(4) nonlinear encoder for 1 b/s/Hz*

The trellis for the encoder in Figure 3.12 is shown in Figure 3.13 and follows all Ungerboeck's rules. It can be noted that the trellis in Figure 3.13 was obtained from the trellis in Figure 3.11 by eliminating transitions corresponding to the input values $u^U[n] \in \{1, 3\}$, thus doubling the distance between the transitions originating from a state and between transitions joining a state. Nevertheless, the code rate is reduced to $1/2$, and consequently, the spectral efficiency is reduced to 1 b/s/Hz. As was already shown in section 3.2.2, the Euclidean distance is $d^2_{E,R=1/2,opt.,u^U[n]\in\{0,2\}} = 9/\left(\sqrt{5/4}\right)^2 = 7.2$. Hence, this rate-$1/2$ code for 1 b/s/Hz PAM transmission offers a coding gain of $10log_{10}\,(7.2/0.8) = 9.54\,\mathrm{dB}$ over the rate-1 encoder shown in Figure 3.10.

Following the same design procedures as in Figure 3.12, we can design optimum RSC encoders using the LCIRC function, for any output word length N. In fact, for a fixed output word length N, an optimum RSC encoder will be determined for each input word length $N_{in} \in \{1, 2, \ldots, N-1\}$, for which the encoding rate is $R = \{1/N, 2/N, \ldots, (N-1)/N\}$.

The general block scheme for a rate-N_{in}/N optimum encoder, with $N_{in} \in \{1, 2, \ldots, N-1\}$, using one delay element and the LCIRC function, is shown in Figure 3.14. The notation $\mathrm{LCIRC}^{N_{in}}$ represents the LCIRC function application for N times consecutively, as was defined in section 2.3 (definition 2.12). Both adders and the multiplier are modulo-2^N operators.

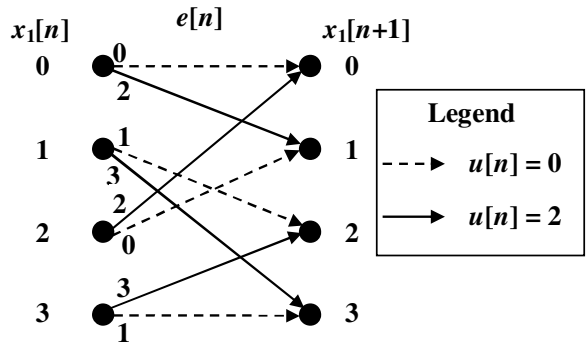


Figure 3.13. *Trellis for rate-1/2 optimum GF(4) nonlinear encoder (1 b/s/Hz)*

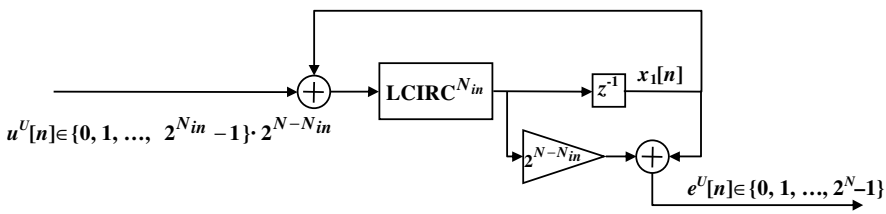


Figure 3.14. *Rate N_{in}/N optimum GF(2^N) nonlinear encoder for N_{in} b/s/Hz*

The trellis complexity of the codes generated with the scheme in Figure 3.14 increases with the word length, because the number of trellis states grows exponentially with the output word length, i.e. 2^{2N}, while the number of transitions originating from and ending in the same state grows exponentially with the input word length, i.e. $2^{2N_{in}}$.

It can be demonstrated that the minimum Euclidean distance for the encoder in Figure 3.14 has the following expression for the PAM-TCM signal:

$$d^2_{E,R=N_{in}/N} = \begin{cases} \dfrac{2\left(2^{N-N_{in}}\right)^2 + \sum\limits_{i=0}^{N-N_{in}-1}\left(2^i\right)^2}{\sigma^2_{xS[n]}} \\ \quad \text{for} \quad N_{in} \in \left\{1, 2, \ldots, \frac{N}{2}-1, \frac{N}{2}+1, \ldots, N-1\right\} \\ \dfrac{2\left(2^{N-N_{in}}\right)^2+1}{\sigma^2_{xS[n]}} \\ \quad \text{for} \quad N_{in} = \frac{N}{2} \end{cases} \qquad [3.9]$$

Introducing the variance expression from [3.1] in [3.9], the latter can be rewritten as:

$$d^2_{E,R=N_{in}/N} = \begin{cases} 4\dfrac{7\cdot 2^{2(N-N_{in})}-1}{2^{2N}-1} \\ \quad \text{for} \quad N_{in} \in \{1, 2, \ldots, \frac{N}{2}-1, \frac{N}{2}+1, \ldots, N-1\} \\ 12\dfrac{2^{2(N-N_{in})+1}+1}{2^{2N}-1} \\ \quad \text{for} \quad N_{in} = \frac{N}{2} \end{cases} \qquad [3.10]$$

For example, let us consider the optimum encoders for the output word length equal to 4, i.e. $N = 4$. The input word length may take three values $N_{in} \in \{1, 2, 3\}$, and the corresponding encoding rates are $R \in \{1/4, 1/2, 3/4\}$. For the rate-$1/4$ encoder, the scheme in Figure 3.14 is set with all the values corresponding to $N_{in} = 1$. From [3.10], it follows that the minimum distance of this code is $d^2_{E,R=1/4,opt.,u^U[n]\in\{0,8\}} = 149 \cdot 12/255 = 7.0118$, having a coding gain of $10log_{10}\left(d^2_{E,R=1/4,opt.,u^U[n]\in\{0,8\}}/d^2_{E,R=1,N=4}\right) = 10log_{10}(149/2) \approx 18.72\,\text{dB}$ over the optimum quaternary ($N = 4$) rate-1 nonlinear encoder. For the rate-$1/2$ encoder ($N_{in} = 2$), the minimum distance is $d^2_{E,R=1/2,opt.,u^U[n]\in\{0,4,8,12\}} = 33 \cdot 12/255 = 1.5529$, having a coding gain of $10log_{10}(33/2) \approx 12.17\,\text{dB}$ over the optimum quaternary ($N = 4$) rate-1 nonlinear encoder. Finally, for the rate-$3/4$ encoder ($N_{in} = 3$), the minimum distance of this code is $d^2_{E,R=3/4,opt.,u^U[n]\in\{0,2,4,6,8,10,12,14\}} = 9 \cdot 12/255 = 0.4235$, having a coding gain of $10log_{10}(9/2) \approx 6.53\,\text{dB}$

over the optimum quaternary ($N = 4$) rate-1 nonlinear encoder. The rate-1 optimum encoder is obtained for $N_{in} = N$, for any value of N, considering that $LCIRC^0\left(x^U\right) = LCIRC^N\left(x^U\right) = x^U$ assumes no bit circulation. This rate-1 optimum encoder offers a minimum distance of $d^2_{E,R=1,opt.} = 2/\sigma^2_{x^S[n]} = 24/\left(2^{2N} - 1\right)$. In Table 3.2, a few values of the minimum distances of the encoder in Figure 3.14 for different values of N_{in} and N are presented. The resulted coding rates are presented in the third column. Analyzing the values in Table 3.2, it can be noted that the minimum distance of a code decreases when its coding rate increases, for any value of N. This fact is well known, i.e. the code performances decrease as the rate increases. Unfortunately, these performances are related to the spectral efficiency of these PAM transmissions. For the codes presented in Table 3.2, having the encoder structure in Figure 3.14, the spectral efficiency for the PAM transmission is equal to the input word length N_{in}. Hence, the code performances' increase is paid for by a spectral efficiency decrease. To perform a more relevant comparison between these codes, the minimum distance of each code is represented in Figure 3.15 as a function of both related variables: coding rate R and spectral efficiency N_{in}.

N	N_{in}	R	d^2_E
1	1	1	8
2	1	1/2	7.2
2	2	1	1.6
3	1	1/3	7.0476
3	2	2/3	1.7143
3	3	1	0.3810
4	1	1/4	7.0118
4	2	1/2	1.5529
4	3	3/4	0.4235
4	4	1	0.0941

Table 3.2. *Minimum PAM-TCM distances as function of N and N_{in} for optimum $GF(2^N)$ nonlinear RSC encoders*

Analyzing the results in Figure 3.15, it can be noted that the minimum distance for a fixed spectral efficiency N_{in} tends rapidly to a limit value when

N increases. From [3.10] follows:

$$d^2_{E,R=N_{in}/N,N\to\infty} = \begin{cases} \frac{7}{2^{2N_{in}-2}} & \text{for} \quad N_{in} \in \{1,2,\ldots,\frac{N}{2}-1,\frac{N}{2}+1,\ldots,N-1\} \\ \frac{24}{2^{2N_{in}}} & \text{for} \quad N_{in}=\frac{N}{2} \end{cases} \quad [3.11]$$

The four possible optimum encoders presented before for the output word length $N = 4$, with the corresponding encoding rates $R \in \{1/4, 1/2, 3/4\}$, are shown in Figure 3.15 with cross markers.

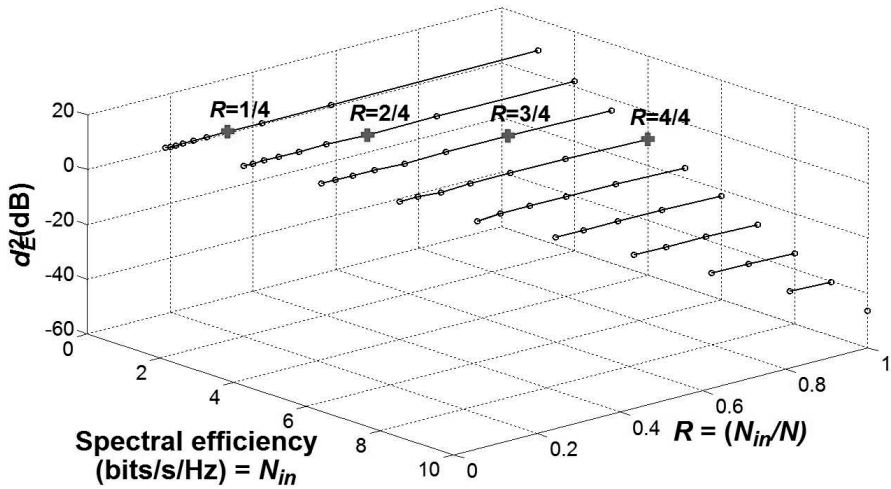


Figure 3.15. *Minimum distance as a function of the code rate R and spectral efficiency for PAM-TCM with optimum GF(2^N) nonlinear encoders*

3.2.4. *Increasing the coding gain by increasing the number of outputs*

As demonstrated in section 3.1, the Frey nonlinear encoder has a trellis that does not offer any coding gain over the linear encoder and over any usual coded system with the same complexity. To obtain certain coding gains, a TCM scheme was developed, which follows Ungerboeck's rules of proper set partitioning [UNG 82]. For the sake of simplicity and ease of comparison with the Frey encoder, only the 2's complement PAM signals were considered.

First, to reduce the coding rate, the Frey encoder scheme is modified by adding new outputs. Hence, in order to obtain an encoding rate of $1/2$, a new

output is added to the original Frey encoder. The two-dimensional (2D) signal is transmitted in the QASK modulation form, obtained from the combination of two PAM representations, one for each dimension. Here, the coding gain is increased by proper signal set partitioning. It must be noted that the same representation word length is kept throughout the whole structure. In the following, the Frey encoder in Figure 1.3 (section 1.1.3), having the same word length for the input as for the output, i.e. $N_{in} = N_{out} \overset{\text{not}}{=} N$, will be considered.

For the encoders presented in section 3.1, the encoding rate can be defined as the ratio between the number of input lines L_{in} and the number of output lines L_{out} in the encoder scheme:

$$R_L \triangleq \frac{L_{in}}{L_{out}} \qquad [3.12]$$

Using [3.12], it follows that all the codes in section 3.1 have an encoding rate equal to 1 because the number of input lines is equal to the number of output lines, both being equal to 1. The encoding rate can be decreased by increasing the number of output lines, for example to $L_{out} \leq 2$. In this case, we have a number of L_{out} output samples as a response to a single input sample. In fact, the signal at the output becomes a 2D one.

Therefore, considering the same 2's complement representation of the samples, we have a QASK signal in the output of the filter, having a square constellation. Hence, the size of the constellation is 2^{2N}. For a PAM transmission, the input word length determines the spectral efficiency, i.e. N_{in} b/s/Hz.

Let us consider the simplest case, which assumes the Frey encoder in Figure 1.3 (section 1.1.3), having only 1 bit word length. As a matter of fact, the input and the output signals take unsigned values from the binary set $u^U[n], e^U[n] \in \{0, 1\}$. Let us denote the two signed output signals by $e_1^S[n]$ and $e_2^S[n]$.

The 4-QASK constellation, determined by the 2's complement values in both outputs, with the best signal set partitioning, is represented in Figure 3.16. Again, the power normalization by the 2's complement signal set mean power is used as presented in section 3.1, equation [3.1]. However, this

time the normalization is performed by two times the mean variance in [3.1], due to the complex nature of the output QASK signal. The signal elements are denoted by S_0, S_1, S_2 and S_3, having the coordinates as represented in Figure 3.16. The minimum distance subsets are denoted by D_0 and D_1, having a distance equal to 4. We can see that we have already an improvement in the coding performance, because at the same mean power, the minimum distance per 2D signal subset is two times larger than that for a one-dimensional (1D) signal subset, as presented in section 3.1. This determines an increase in the coding gain by 3 dB. Also, this makes this second method more interesting than the first method (presented in section 3.2.1) regarding the Frey encoder performances. However, the method presented here changes the structure of the encoder, as will be presented later.

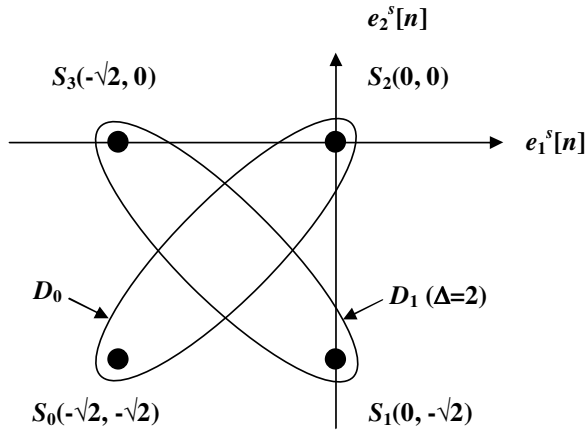


Figure 3.16. *4-QASK-normalized 2's complement constellation and optimum subset partitioning for binary (N = 1) and two outputs Frey encoder*

The resulted trellis for the rate-$1/2$ convolutional Frey encoder used for 4-QASK modulation is shown in Figure 3.17. The trellis in Figure 3.17 is identical to the trellis in Figure 3.1, only that the transitions here are assigned the signal elements from the constellation in Figure 3.16, using the rule of set partitioning as proposed by Ungerboeck for the states originating from the same state and entering the same state. Fortunately, we eliminated the disadvantage of having the same signal element for the transitions entering the same state, as compared to all the schemes in section 3.2.1. This fact also increases the minimum Euclidean distance of the trellis by 3 dB.

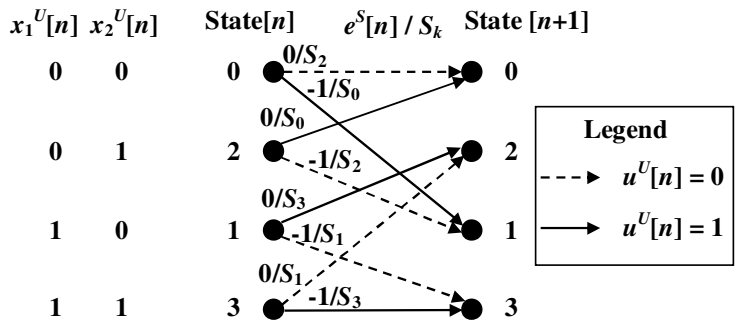


Figure 3.17. *Trellis for binary (N = 1) Frey encoder with two outputs, for 4-QASK modulation*

To introduce the set partitioning in Figure 3.16, we have to modify the scheme of the Frey encoder. Using the same notations as in Figure 1.3 (section 1.1.3), the expressions of the signed output signals $e_1^S[n]$ and $e_2^S[n]$ used to determine this set partitioning can be easily derived as given by:

$$\begin{aligned} e_1^S[n] &= u^S[n] \\ e_2^S[n] &= \{\left(u^U[n] + x_1^U[n]\right) \bmod 2\}^S \end{aligned} \qquad [3.13]$$

The resulting modified rate-$1/2$ convolutional Frey encoder used for 4-QASK modulation is depicted in Figure 3.18.

Let us determine the minimum Euclidean distance for the trellis in Figure 3.17, considering the following pair of paths, both starting in state 3 and ending in state 2:

$$\begin{aligned} &\text{Path 1 (trellis states)} : 3\ 2\ 1\ 2 \\ &\text{Path 2 (trellis states)} : 3\ 3\ 3\ 2 \end{aligned}$$

For each of these two paths, the output signal elements are as follows:

$$\begin{aligned} &\text{Path 1 (trellis states)} : \mathrm{S}_1\ \mathrm{S}_2\ \mathrm{S}_3 \\ &\text{Path 2 (trellis states)} : \mathrm{S}_3\ \mathrm{S}_3\ \mathrm{S}_1 \end{aligned}$$

The distance between paths that is also the minimum distance of the trellis in Figure 3.17 is determined as follows:

$d^2_{E,R=1/2,modified\,Frey,N=1,4-QASK} = (2)^2 + (\sqrt{2})^2 + (2)^2 = 10$, which is 2.5 times larger than the distance in trellis of the original binary Frey encoder.

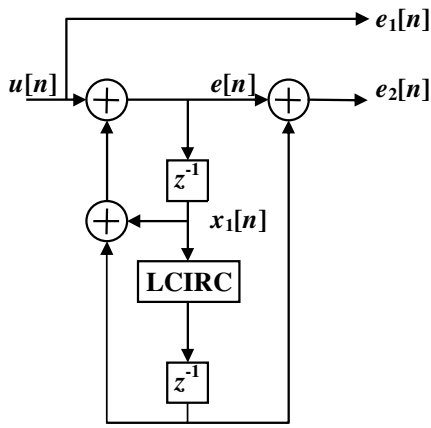


Figure 3.18. *Modified rate-1/2 convolutional binary (N = 1) Frey filter for 4-QASK modulation*

We can conclude that the rate-$1/2$ convolutional binary ($N = 1$) Frey filter for 4-QASK modulation is performing better than the simple binary Frey encoder, having an encoding gain of $10log_{10}\left(d^2_{E,R=1/2,modified\,Frey,N=1,4-QASK}/d^2_{E,N=1,Frey}\right) = 10log_{10}\,(10/4) \approx 4$ dB.

Now, let us consider a more complex case, which assumes the Frey encoder in Figure 1.3 from section 1.1.3, having two bits per sample in the representation code word ($N = 2$). As a matter of fact, the input and the output signals take unsigned values from the binary set $u^U[n], e^U[n] \in \{0, 1, 2, 3\}$. Again, the two signed output signals are denoted by $e^S_1[n]$ and $e^S_2[n]$.

The normalized power 16-QASK constellation, determined by the 2's complement values in both outputs, with the best signal set partitioning is represented in Figure 3.19.

Considering that the input of the encoder takes four possible values, it follows that we have a trellis with four transitions originating and joining

each possible state from the 16 states. Each group of four such transitions must be associated with a minimum distance subset from the signal set. Hence, we will split the whole 16-point constellation into four subsets of four signal states each, i.e. S_0, S_1, S_2 and S_3. These subsets are selected to have the maximum distance in each subset, i.e. $\Delta_1 = 2\Delta_0 = 4/\sqrt{10}$, and are assigned starting from the lower left corner and continuing by a counter-clockwise rotation.

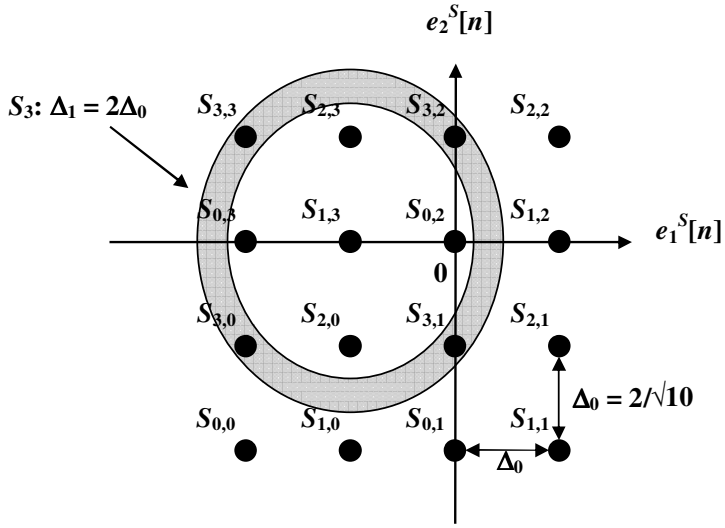


Figure 3.19. *16-QASK-normalized 2's complement constellation and optimum subset partitioning for quaternary (N = 2) Frey encoder with two outputs*

Each point in a subset S_i, denoted by $S_{i,j}$, is assigned following the same rule as for the subsets for all $i, j \in \{0, 1, 2, 3\}$. Hence, the trellis for the rate-$1/2$ convolutional Frey encoder used for 16-QASK modulation is shown in Figure 3.20. The trellis in Figure 3.20 is identical to the trellis in Figure 3.3, only that the transitions here are assigned the signal subsets in Figure 3.19, using the rule of set partitioning as proposed by Ungerboeck. Therefore, we eliminated the disadvantage of having the same signal element for the transitions entering the same state, as compared to all the schemes in section 3.2.1. Also, this modification increases the minimum Euclidean distance of the trellis by 3 dB.

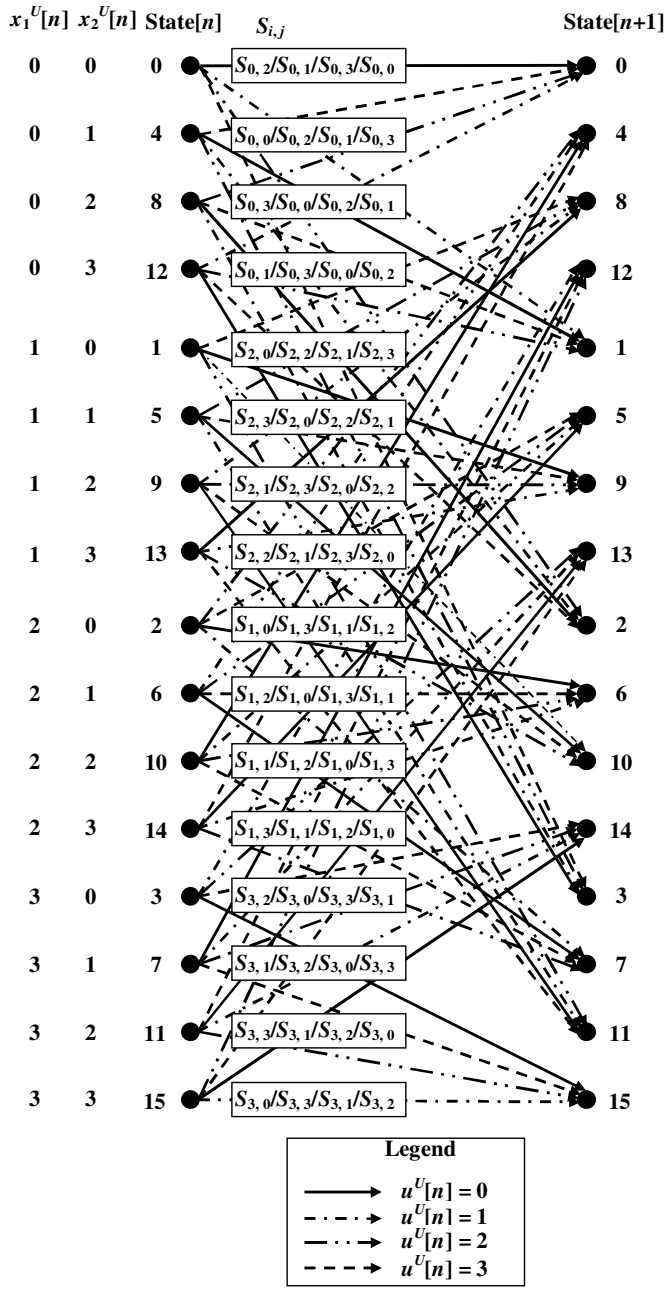


Figure 3.20. *Trellis for quaternary ($N = 2$) Frey encoder with two outputs, for 16-QASK modulation*

As for the binary case, in order to introduce the set partitioning in Figure 3.19, we have to modify the scheme of the Frey encoder. In fact, a subset optimum mapping has to be added. Using the same notations as in Figure 1.3, we can derive the expressions of the signed output signals $e_1^S[n]$ and $e_2^S[n]$, used to determine this set partitioning, as given by:

$$\begin{array}{l} e_1^S[n] = \{\left[F_1\left(u^U[n]\right) + F_2\left(\left(3 \cdot x_1^U[n]\right) \bmod 2^2\right)\right] \bmod 2^2\}^S \\ e_2^S[n] = \{\left(2 \cdot u^U[n] + x_1^U[n]\right) \bmod 2^2\}^S \end{array} \quad [3.14]$$

where the functions $F_1(.)$ and $F_2(.)$ are given by:

$$\begin{array}{l} F_1\left(x^U[n]\right) = \left\{ \begin{array}{lll} 0 & \text{if} & 0 \leq x^U[n] \leq 2^{N-1} - 1 \\ 2 & \text{if} & 2^{N-1} \leq x^U[n] \leq 2^N - 1 \end{array} \right. \\ F_2\left(x^U[n]\right) = \left\{ \begin{array}{lll} 0 & \text{if} & 0 \leq x^U[n] \leq 2^{N-1} - 1 \\ 1 & \text{if} & 2^{N-1} \leq x^U[n] \leq 2^N - 1 \end{array} \right. \end{array} \quad [3.15]$$

The resulting modified rate-$1/2$ convolutional Frey encoder used for 16-QASK modulation is depicted in Figure 3.21.

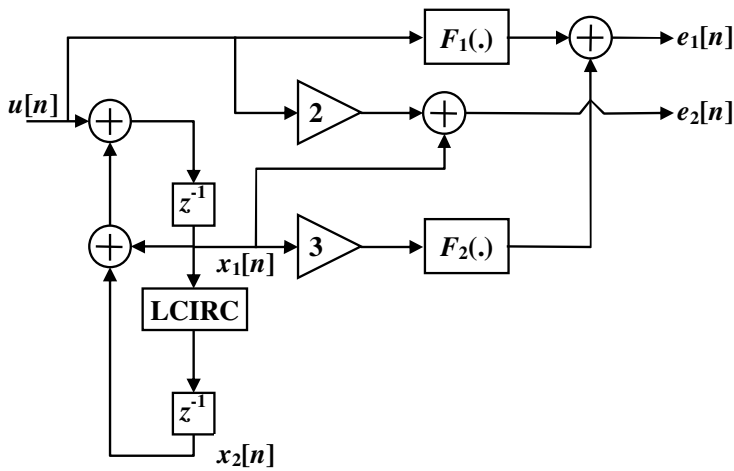


Figure 3.21. *Modified rate-1/2 convolutional quaternary (N = 2) Frey encoder for 16-QASK modulation*

We estimated the minimum Euclidean distance for the trellis in Figure 3.20, and we found two different values, each depending on the initial state in the trellis. The following two examples will present these two cases.

First, let us consider the following pair of paths, both starting in state 0 and ending in state 10:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 0\ 2\ 5\ 10\\ &\text{Path 2 (trellis states)} : 0\ 3\ 13\ 10\end{aligned}$$

For these two paths, the output signal elements are the following:

$$\begin{aligned}&\text{Path 1 (trellis states)} : \mathrm{S}_{0,3}\ \mathrm{S}_{1,3}\ \mathrm{S}_{2,1}\\ &\text{Path 2 (trellis states)} : \mathrm{S}_{0,0}\ \mathrm{S}_{3,0}\ \mathrm{S}_{2,2}\end{aligned}$$

The distance between these two paths is: $d^2_{E,R=1/2,modified\,Frey,N=2,16-QASK} = (2\Delta_0)^2 + \left(\Delta_0\sqrt{2}\right)^2 + (2\Delta_0)^2 = 10\Delta_0^2 = 4$, which is five times larger than the distance in trellis of the original quaternary Frey encoder.

The second value of the Euclidean distance can be determined for pairs of paths as:

$$\begin{aligned}&\text{Path 1 (trellis states)} : 1\ 11\ 14\ 4\\ &\text{Path 2 (trellis states)} : 1\ 8\ 2\ 4\end{aligned}$$

For these two paths, the output signal elements are the following:

$$\begin{aligned}&\text{Path 1 (trellis states)} : \mathrm{S}_{2,3}\ \mathrm{S}_{3,2}\ \mathrm{S}_{1,3}\\ &\text{Path 2 (trellis states)} : \mathrm{S}_{2,0}\ \mathrm{S}_{0,2}\ \mathrm{S}_{1,0}\end{aligned}$$

The distance between these two paths is: $d^2_{E,R=1/2,modified\,Frey,N=2,16-QASK} = (2\Delta_0)^2 + (\Delta_0)^2 + (2\Delta_0)^2 = 9\Delta_0^2 = 3.6$, which is 4.5 times larger than the distance in trellis of the original quaternary Frey encoder.

In conclusion, the trellis in Figure 3.20 offers an encoding gain of at least $10log_{10}\left(d^2_{E,R=1/2,modified\,Frey,N=2,16-QASK}/d^2_{E,N=2,Frey}\right) = 10log_{10}(3.6/0.8) \approx 6.5\text{dB}$ over the simple quaternary Frey encoder.

The TCM design method presented earlier was also applied for Frey encoders with larger word lengths, but the complexity of such TCM schemes increases dramatically. The number of trellis states grows exponentially with the input word length, i.e. 2^{2N}. However, this trellis design method can be generalized for any word length, determining in all cases significant encoding gains over the non-encoded system.

3.3. Optimum nonlinear encoders for phase shift keying – TCM schemes

3.3.1. *Rate-*$1/2$ *optimum encoder for a QPSK-TCM transmission*

In this section, the potential of the nonlinear LCIRC function is shown, for designing efficient encoders. Therefore, following the trellis optimization presented in [VLA 09a] and [VLA 09c], a simple nonlinear encoder operating over $GF(4)$ was developed, which has a binary input. It is demonstrated that this encoder performs identically to an optimum rate-$1/2$ binary field RSC convolutional encoder. Both encoders offer maximum coding gain for 1 b/s/Hz [UNG 82, VLA 09a]. The scheme of the rate-$1/2$ optimum $GF(4)$ encoder was introduced in Figure 3.12. The same notations as in section 3.2.3 are used. The trellis for the encoder in Figure 3.12 is shown in Figure 3.13.

Mapping an unsigned output symbol value $e^U[n]$ into an instant carrier phase value $\varphi^e[n]$ over the n-th sample interval, a 2^N levels PSK-TCM scheme is obtained. A simple phase mapping is given by:

$$\varphi^e[n] = e^U[n] \cdot \frac{2\pi}{2^N},\ e^U[n] \in \{0, 1, \ldots 2^N - 1\} \qquad [3.16]$$

The phase constellation for the QPSK scheme using the mapping in [3.16] is represented in Figure 3.22.

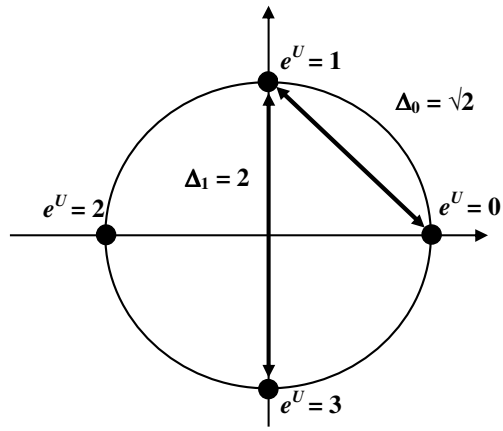


Figure 3.22. *Phase constellation for QPSK-TCM*

For the M-PSK signal, we can write the following expressions of the Euclidean distances between the constellation points in the ascending order:

$$\Delta_k = 2 \cdot sin\left[\frac{(k+1)\cdot\pi}{M}\right], \quad k \in \{0, 1, \ldots log_2(M) - 1\} \qquad [3.17]$$

where M denotes the number of phase levels.

Considering the mapping in [3.16] and the distances' expressions in [3.17], it follows that the QPSK-TCM signal trellis in Figure 3.13 presents a minimum Euclidean distance of $d^2_{E,R=1/2,opt.,u^U[n]\in\{0,2\},QPSK} = 2 \cdot \Delta_1^2 + \Delta_0^2 = 10$ for a spectral efficiency of 1 b/s/Hz. Hence, this rate-$1/2$ code for 1 b/s/Hz QPSK-TCM transmission offers a coding gain of $10log_{10}\,(2.5) \approx 4\,\mathrm{dB}$ over the rate-1 QPSK-TCM in transmission using the encoder in Figure 1.3 (section 1.1.3).

3.3.2. *Generalized optimum encoder for a PSK-TCM transmission*

Following the same design procedures as in section 3.3.1, we can design optimum recursive convolutional (RC) encoders using the LCIRC function, for any output word length N. In fact, for a fixed output word length N, an optimum RC encoder will be determined for each input word length $N_{in} \in \{1, 2, \ldots, N-1\}$, for which the encoding rate is

$R = \{1/N, 2/N, \ldots, (N-1)/N\}$. The general block scheme for a rate-N_{in}/N optimum PSK-TCM encoder, with $N_{in} \in \{1, 2, \ldots, N-1\}$ using one delay element and the LCIRC function is the same as for the PAM signal as shown in Figure 3.14. For the sake of conciseness, the same notations and definitions given in section 3.2.3 are used here.

It can be demonstrated that the minimum Euclidean distance for the encoder in Figure 3.14 has the following expression for the PSK-TCM signal:

$$d^2_{E,R=N_{in}/N,2^N-PSK} = \begin{cases} 2\left(\Delta_{2^{N-N_{in}}-1}\right)^2 + \sum_{i=0}^{2^{N-N_{in}}-2} (\Delta_i)^2 \\ \quad \text{for} \quad N_{in} \in \{1, 2, \ldots, \frac{N}{2}-1, \frac{N}{2}+1, \ldots, N-1\} \\ 2\left(\Delta_{2^{N-N_{in}}-1}\right)^2 + (\Delta_0)^2 \\ \quad \text{for} \quad N_{in} = \frac{N}{2} \end{cases} \quad [3.18]$$

For example, let us consider the optimum encoders for the output word length equal to 3, i.e. $N = 3$. The input word length may take two values $N_{in} \in \{1, 2\}$, and the corresponding encoding rates are $R \in \{1/3, 2/3\}$. For the rate-$1/3$ encoder, the scheme in Figure 3.14 is set with all the values corresponding to $N_{in} = 1$. From [3.17] and [3.18] it follows that the minimum distance of this code is $d^2_{E,R=1/3,opt.,8-PSK,u^U[n]\in\{0,4\}} = 14$, having a coding gain of $10log_{10}\left(d^2_{E,R=1/3,opt.,8-PSK,u^U[n]\in\{0,4\}}/d^2_{E,R=1,N=3,opt.,8-PSK}\right) = 10log_{10}\left(14\ /1.1716\right) \approx 10.77\,\text{dB}$ over the optimum 8-PSK ($N = 3$) rate-1 encoder. For the rate-$2/3$ encoder ($N_{in} = 2$), the minimum distance of the code is $d^2_{E,R=2/3,opt.,8-PSK,u^U[n]\in\{0,2,4,6\}} = 4 + 4 \cdot sin^2\left(\pi/8\right) \approx 4.5858$, having a coding gain of approximately $5.93\,\text{dB}$ over the optimum 8-PSK ($N = 3$) rate-1 nonlinear encoder. The rate-1 optimum encoder is obtained for $N_{in} = N$, for any value of N, considering that $LCIRC^0\left(x^U\right) = LCIRC^N\left(x^U\right) = x^U$. This rate-1 optimum encoder offers a minimum distance of $d^2_{E,R=1,opt.,N=3,8-PSK} = 8 \cdot sin^2\left(\pi/8\right) \approx 1.1716$.

In Table 3.3, a few values of the minimum distances of the encoder in Figure 3.14 for different values of N_{in} and N and for different PSK modulations are presented. The resulting coding rates are presented in the

third column. Analyzing the values in Table 3.3, it can be noted that the minimum distance of a code decreases when its coding rate increases, for any value of N. The interpretation of these results is similar to the results presented in Table 3.2 from section 3.2.3. For the codes presented in Table 3.3, having the encoder structure in Figure 3.14, the spectral efficiency for the PSK transmission is equal to the input word length N_{in}. Again, the code performances' increase is paid for by a spectral efficiency decrease.

N	N_{in}	R	d_E^2
1	1	1	8
2	1	1/2	10
2	2	1	4
3	1	1/3	14
3	2	2/3	$4 + 4 \cdot sin^2(\pi/8) \approx 4.5858$
3	3	1	$8 \cdot sin^2(\pi/8) \approx 1.1716$

Table 3.3. *Minimum PSK-TCM distances as function of N and N_{in} for optimum $GF(2^N)$ RC-LCIRC encoders*

It can be easily noted that all the rate-$(N-1)/N$, for any N value, the optimum RC-LCIRC encoders offer the same minimum distance as the corresponding binary optimum encoders determined by Ungerboeck in [UNG 82]. For example, the above mentioned rate-$2/3$ encoder ($N_{in} = 2$) has the minimum Euclidean distance of 4.5858, determining an asymptotic coding gain of 3.6 dB. Using the expressions in [3.18], we can easily plot the asymptotic gain of optimum rate-$(N-1)/N$ LCIRC encoder as a function of the spectral efficiency $N-1$. The results are shown in Figure 3.23.

The asymptotic coding gain is estimated using the following expression [UNG 82]:

$$G\,[dB] \triangleq 10 \cdot log_{10}\left(\frac{d^2_{E,R=(N-1)/N,2^N-PSK}}{\Delta^2_{0,\,2^{(N-1)}-PSK}}\right) \qquad [3.19]$$

where $d^2_{E,R=(N-1)/N,2^N-PSK}$ denotes the minimum Euclidean distance of the rate $(N-1)/N$ LCIRC TCM scheme using a PSK modulation with 2^N phase levels, and $\Delta^2_{0,\,2^{(N-1)}-PSK}$ represents the minimum Euclidean distance in the

$2^{(N-1)}$-PSK non-coded signal constellation. Hence, the asymptotic coded gain specifies the gain of the coded scheme when doubling the signal constellation size over the non-coded signal. As shown in Figure 3.23, the asymptotic gain decays rapidly to a limit value when the number of signal levels increases. This limit specifies the maximum number of signal levels to be selected.

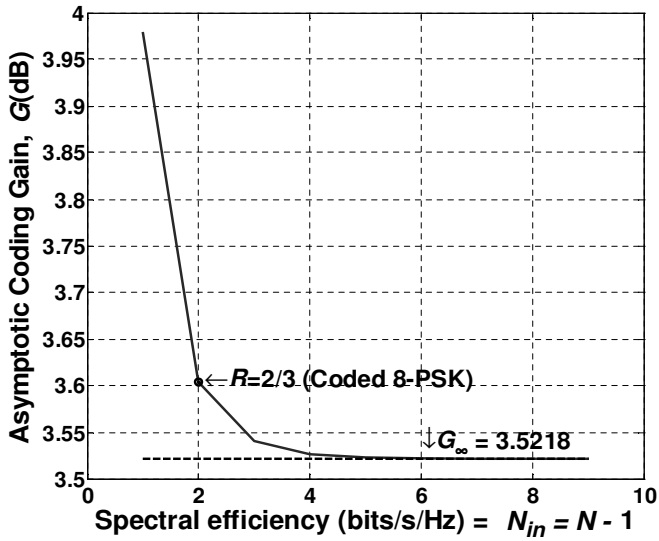


Figure 3.23. *Asymptotic gain as a function of the spectral efficiency for rate $(N-1)/N$ optimum GF(2^N) recursive convolutional LCIRC encoders*

However, the $GF(2^N)$ optimum RC-LCIRC encoders are less complex than the corresponding binary encoders. The memory size of the binary encoders increases logarithmically with the number of states in the trellis, while the $GF(2^N)$ optimum RC-LCIRC encoders include only one delay element, irrespective of the trellis complexity. As another advantage of these encoders, we can also mention the Euclidean distance compact expression as a function of N_{in} and N.

All TCM schemes presented earlier used PSK modulation. Even if PSK is used in practice only for small spectral efficiencies, i.e. up to 3 b/s/Hz, optimum RC-LCIRC encoders can be designed for any spectral efficiency value, using the scheme in Figure 3.14 with minimum distances given by [3.17] and [3.18].

3.4. Optimum nonlinear encoders for quadrature amplitude modulation – TCM schemes

Following the same design procedures as in section 3.3.2, we can design optimum RC encoders for quadrature amplitude modulation TCM (QAM-TCM) schemes using the LCIRC function having any value for the output word length N. Let us consider the general block scheme for a rate-N_{in}/N optimum QAM-TCM encoder, with $N_{in} \in \{1, 2, \ldots, N-1\}$ using one delay element and the LCIRC function, which is shown in Figure 3.14. For the sake of conciseness, all notations and definitions given in section 3.2.3 are kept here.

It can be demonstrated that the minimum Euclidean distance for the encoder in Figure 3.14 has the following expression for the QAM-TCM signal:

$$d^2_{E,R=N_{in}/N,2^N-QAM} = \begin{cases} \left(2^{N-N_{in}+1} + \sum_{i=0}^{N-N_{in}-1} 2^i\right) \cdot \Delta^2_{0,\,2^N-QAM} \\ \quad \text{for} \quad N_{in} \in \{1, 2, \ldots, \frac{N}{2}-1, \frac{N}{2}+1, \ldots, N-1\} \\ \left(2^{N-N_{in}+1}+1\right) \cdot \Delta^2_{0,\,2^N-QAM} \\ \quad \text{for} \quad N_{in} = \frac{N}{2} \end{cases} \qquad [3.20]$$

For example, let us consider the optimum encoders for the output word length equal to 4, i.e. $N = 4$. The input word length may take three values $N_{in} \in \{1, 2, 3\}$, and the corresponding encoding rates are $R \in \{1/4, 1/2, 3/4\}$. For the rate-$1/4$ encoder, the scheme in Figure 3.14 is set with all the values corresponding to $N_{in} = 1$. From [3.20], it follows that the minimum distance of this code is $d^2_{E,R=1/4,opt.,16-QAM,u^U[n]\in\{0,8\}} = 9.2$, determining a coding gain of $10log_{10}\left(d^2_{E,R=1/4,opt.,16-QAM,u^U[n]\in\{0,8\}} / d^2_{E,R=1,N=4,opt.,16-QAM}\right) = 10log_{10}(9.2/0.8) \approx 10.6\,\text{dB}$ over the optimum 16-QAM ($N = 4$) rate-1 encoder. For the rate-$2/4$ encoder ($N_{in} = 2$), the minimum distance of the code is $d^2_{E,R=1/2,opt.,16-QAM,u^U[n]\in\{0,4,8,12\}} = 3.6$, having a coding gain of approximately $6.53\,\text{dB}$ over the optimum 16-QAM ($N = 4$) rate-1 nonlinear encoder. For the rate-$3/4$ encoder ($N_{in} = 3$), the minimum distance of the code is $d^2_{E,R=3/4,opt.,16-QAM,u^U[n]\in\{0,2,4,6,8,10,12,14\}} = 2$, having a coding gain of approximately $3.97\,\text{dB}$ over the optimum 16-QAM ($N = 4$) rate-1

nonlinear encoder. The rate-1 optimum encoder is obtained for $N_{in} = N$, for any value of N, considering that $LCIRC^0\left(x^U\right) = LCIRC^N\left(x^U\right) = x^U$. This rate-1 optimum encoder offers a minimum distance of $d^2_{E,R=1,opt.,N=4,16-QAM} = 0.8$.

N	N_{in}	R	d^2_E
1	1	1	8
2	1	1/2	10
2	2	1	4
4	1	1/4	9.2
4	2	1/2	3.6
4	3	3/4	2
4	4	1	0.8
6	1	1/6	≈ 9
6	2	1/3	≈ 4.47
6	5	5/6	≈ 0.48
6	6	1	≈ 0.19

Table 3.4. *Minimum QAM-TCM distances as function of N and N_{in} for optimum $GF(2^N)$ RC-LCIRC encoders*

Table 3.4 presents a few values of the minimum distances of the encoder in Figure 3.14 for different values of N_{in} and N and for different QAM modulations. The resulting coding rates are presented in the third column. The interpretation of these results is similar to the results presented in Table 3.2 from section 3.2.3 and in Table 3.3 from section 3.3.2. For the codes presented in Table 3.4, having the encoder structure in Figure 3.14, the spectral efficiency for the QAM transmission is equal to the input word length N_{in}. Again, the code performances increase is paid for by a spectral efficiency decrease.

As in previous sections, it can be noted that all the rate-$(N-1)/N$, for any N value, the optimum RC-LCIRC encoders offer the same minimum distance as the corresponding binary optimum encoders determined by Ungerboeck in [UNG 82].

All TCM schemes presented in this section used QAM modulation. Even if our examples consider QAM constellations with a maximum number of 64

points (i.e. $N = 6$), in practice, very large spectral efficiencies are reached, i.e. up to 10 b/s/Hz (even more). Even for these very large constellations, optimum RC-LCIRC encoders can be designed using the scheme in Figure 3.14 with minimum distances given by [3.20]. In fact, optimum RC-LCIRC encoders can be designed for any spectral efficiency and for any encoding rate values.

3.5. Performance analysis of TCM data communications using modified nonlinear digital encoders: simulation results

All trellis-coded PAM schemes presented in sections 3.1, 3.2.1 and 3.2.2 were considered for simulations. The SER performances for these encoding schemes were analyzed in the presence of additive white Gaussian noise (AWGN). The estimated SER is represented in Figure 3.24 as a function of the SNR, expressed in decibels. All rate-1 nonlinear encoders offer no coding gain over their corresponding linear encoders for any representation word length (i.e. $N = 1$ and $N = 2$ in Figure 3.24), and the binary system performs better than the quaternary system by more than 7 dB. In both cases, there is no coding gain over the non-encoded transmissions, i.e. both the Viterbi decoder and the inverse filter decoder perform the same. The quaternary output rate-$1/2$ encoder presented in section 3.2.1 performs better than the binary rate-1 encoder by more than 0.7 dB, both working for 1 b/s/Hz. The same quaternary output rate-$1/2$ encoder outperforms its linear version by approximately 1.8 dB. The optimum rate-$1/2$ first-order encoder presented in section 3.2.2 has a coding gain of 1.8 dB over the binary rate-1 encoder and a coding gain of more than 3 dB over its linear version. All the simulation results shown in Figure 3.24 are consistent with the theoretical coding gains of these encoders.

Next, all TCM PAM schemes presented in section 3.1, and the 2D TCM QASK schemes introduced in section 3.2.4, were considered for simulations. The SER performances for these encoding schemes were analyzed in the presence of AWGN. The estimated SER is represented in Figure 3.25 as a function of the SNR, expressed in dB. All rate-1 nonlinear encoders offer no coding gain over their corresponding linear encoders for any representation word length (i.e. $N = 1$ and $N = 2$ in Figure 3.25), and the binary system performs better than the quaternary system by more than 7 dB. In both cases, there is no coding gain over the non-encoded transmissions, i.e. both the Viterbi decoder and the inverse filter decoder perform the same.

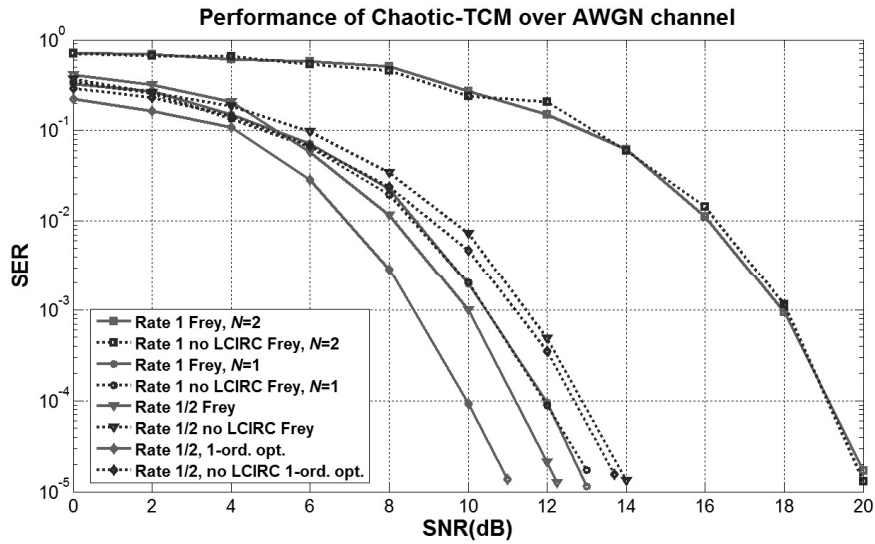


Figure 3.24. *SER performances for nonlinear and linear encoders*

The binary ($N = 1$) 4-QASK TCM scheme presented in section 3.2.4 performs better than the binary PAM TCM scheme in section 3.1 by 4 dB, both working for 1 b/s/Hz. On the other hand, the quaternary ($N = 2$) 16-QASK TCM scheme presented in section 3.2.4 performs better than the quaternary PAM TCM scheme in section 3.1 by more than 6 dB, both working for 2 b/s/Hz. All the simulation results shown in Figure 3.25 are consistent with the theoretical coding gains of these encoders.

The PSK-TCM schemes presented in section 3.3.2 using all optimum encoders in Table 3.3 were also considered for simulations. The SER performances for these encoding schemes using multilevel PSK signals and Viterbi decoding were analyzed in the presence of AWGN. The SER is shown in Figure 3.26 as a function of the SNR.

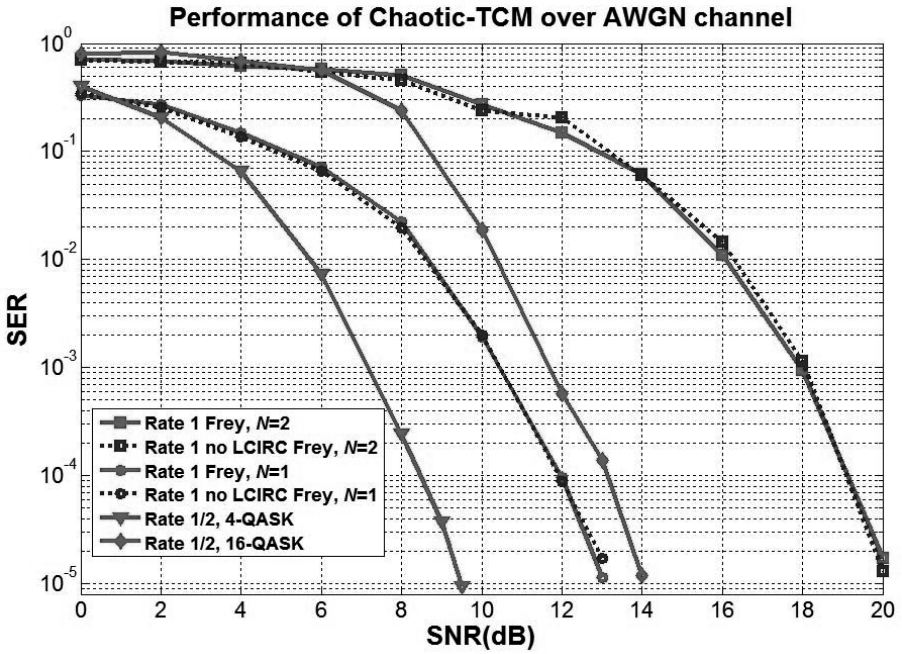


Figure 3.25. *SER performances for PAM and QASK TCM schemes*

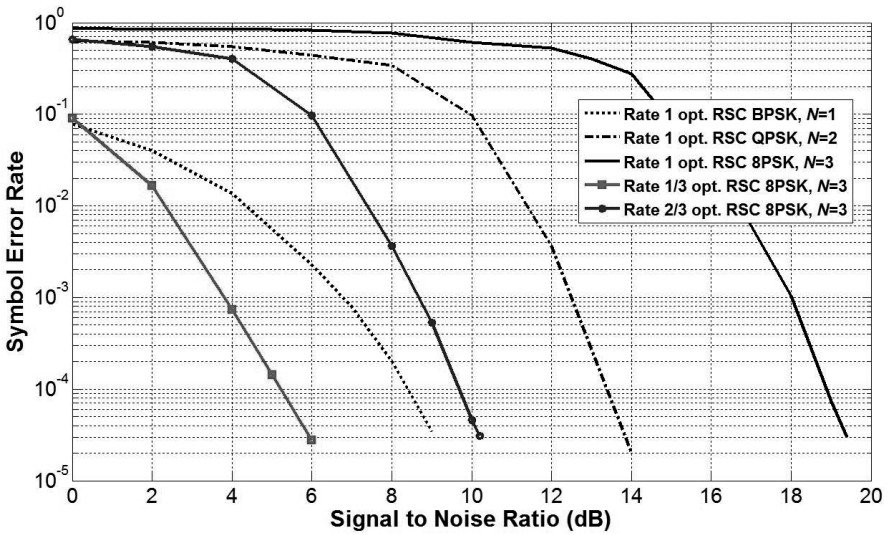


Figure 3.26. *SER performances for PSK-TCM schemes using optimum $GF(2^N)$ nonlinear RC encoders*

The PSK-TCM schemes using rate-1 optimum nonlinear RC encoders for the same spectral efficiencies as both optimum encoder PSK-TCM schemes for $N = 3$, were considered for comparison. For example, the rate-$1/3$ encoder for $N = 3$ has the same spectral efficiency as the rate-1 encoder for $N = 1$, i.e. 1 b/s/Hz, and the rate-$2/3$ encoder for $N = 3$ and the rate-1 encoder for $N = 2$ have an efficiency of 2 b/s/Hz. These cases are considered in Figure 3.26.

Analyzing the SER curves, it can be noted that the rate-$1/3$ encoder for $N = 3$ performs better than the rate-1 encoder for $N = 1$ by more than 3 dB (instead of a gain of approximately 2.43 dB, in theory; see Table 3.3), and the rate-$2/3$ encoder for $N = 3$ performs better than the rate-1 encoder for $N = 2$ by more than 3.8 dB (approximately 0.6 dB, in theory). The average multiplicity of error events with the minimum distance in [3.18], for optimum $GF(2^N)$ RC encoders, is smaller than multiplicity of minimum distance error events for the rate-1 encoders, for all encoders with $N_{in} < N$. This is the reason why the simulation results in Figure 3.26 show larger coding gains between these two encoders for a given spectral efficiency.

Finally, the QAM-TCM schemes presented in section 3.4 using all optimum encoders in Table 3.4 were considered for simulations. The SER performances for these encoding schemes using multilevel QAM signals and Viterbi decoding were analyzed in the presence of AWGN. The SER is shown in Figure 3.27 as a function of the SNR.

The QAM-TCM schemes using rate-1 optimum nonlinear RC encoders for the same spectral efficiencies as both optimum encoder PSK-TCM schemes for $N = 4$, were considered for comparison. For example, the rate-$1/4$ encoder for $N = 4$ has the same spectral efficiency as the rate-1 encoder for $N = 1$, i.e. 1 b/s/Hz, and the rate-$2/4$ encoder for $N = 4$ and the rate-1 encoder for $N = 2$ have an efficiency of 2 b/s/Hz. These cases are considered in Figure 3.27.

Analyzing the SER curves, it can be noted that the rate-$1/4$ encoder for $N = 4$ performs better than the rate-1 encoder for $N = 1$ by more than 1 dB and the rate $2/4$ encoder for $N = 4$ performs almost the same as the rate-1 encoder for $N = 2$. The simulation results are consistent with the theoretical results from Table 3.4. However, the average multiplicity of error events with the minimum distance in [3.20], for optimum $GF(2^N)$ RC encoders, is smaller than multiplicity of minimum distance error events for the rate-1 encoders, for all encoders with $N_{in} < N$. This is the reason why the simulation results in

Figure 3.27 show larger coding gains between these two encoders for a given spectral efficiency.

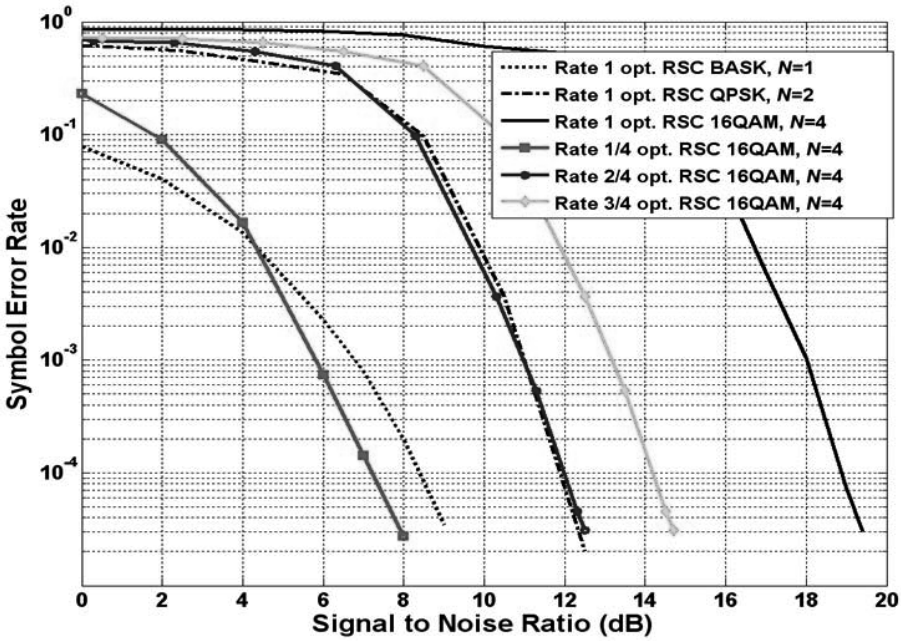


Figure 3.27. *SER performances for QAM-TCM schemes using optimum $GF(2^N)$ nonlinear RC encoders*

4

Parallel Turbo Trellis-Coded Modulation Schemes Using Nonlinear Digital Encoders

4.1. Recursive convolutional-left circulate (RC-LCIRC) encoder in a turbo trellis-coded modulation (TTCM) scheme

Figure 4.1 shows the parallel turbo trellis-coded modulation (TTCM) encoder for 2^N-phase shift keying (PSK) modulation. The information sequence and the N_{in} bits blockwise interleaved sequence are fed into component encoders recursive convolutional-left circulate 1 (RC-LCIRC1) and RC-LCIRC2 of rate N_{in}/N, shown in Figure 3.14, and mapped into 2^N-PSK or 2^N-quadrature amplitude modulation (QAM) symbol sequence ($x[n]$). The non-systematic nature of the RC-LCIRC encoder does not permit the parity bits puncturing as in traditional TTCM schemes. Hence, the overall coding rate for the scheme in Figure 4.1 is $N_{in}/(2 \cdot N)$. The 2^N-PSK symbol sequence is transmitted over a noisy channel. The received signal over the nth symbol interval is given by:

$$y[n] = x[n] + w[n] \qquad [4.1]$$

where $w[n]$ is an additive white Gaussian noise (AWGN) sequence and $x[n]$ denotes the 2^N-PSK symbol sequence mapped from the encoder output sequence $\{e_1^U[n], e_2^U[n]\}$; here, $\{e^U[n]\}$ denotes the output of the component RC-LCIRC encoder, as presented in section 3.2.3.

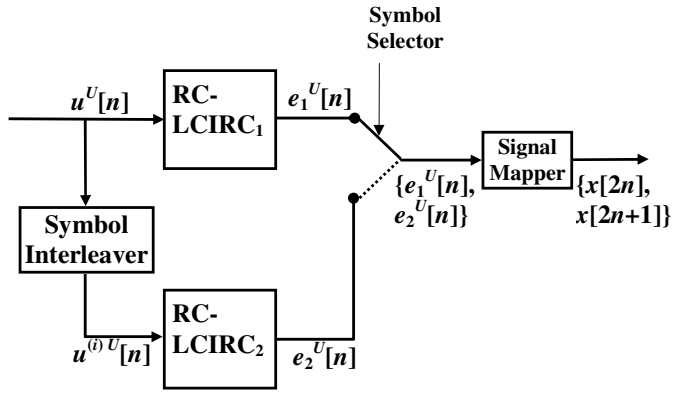


Figure 4.1. *Turbo-TCM transmitter with RC-LCIRC encoders*

The receiver structure, shown in Figure 4.2, has two components that use a multilevel version of the logarithmic maximum *a posteriori* probability (log-MAP) algorithm. The odd symbols from the received sequence are fed into first component decoder that corresponds to the RC-LCIRC1 encoder in order to compute the *a posteriori* log-likelihood ratio (LLR) per transmitted bit, denoted as L_{ap}, as follows [BER 96]:

$$L_{ap}\left(b_{t,m}[n]|\mathbf{y}[n]\right) = ln\frac{P\left(b_{t,m}[n]=1|\mathbf{y}[n]\right)}{P\left(b_{t,m}[n]=0|\mathbf{y}[n]\right)} \qquad [4.2]$$

where $b_{t,m}[n]$ is the mth bit from tth encoder input information word denoted as $\mathbf{b_t}[\mathbf{n}] \leftrightarrow u^U[n]$, transmitted over the nth symbol interval, with $m = 1, ..., 2^{N_{in}}$, and $\mathbf{y}[n]$ is the received symbol vector.

Using Bayes' theorem under the assumption of statistically independent bits, the joint probabilities can be split into products [BER 96]:

$$L_{ap}\left(b_{t,m}[n]|\mathbf{y}[n]\right) = ln\frac{\sum\limits_{b_{t,m}[n]\in\mathbf{b}_m^{(1)}} p(\mathbf{y}[n]|x[n])\cdot P\left(x[n]|\mathbf{b}_m^{(1)}\right)\cdot \prod\limits_{m=1}^{N_{in}} P(b_{t,m}[n]=1)}{\sum\limits_{b_{t,m}[n]\in\mathbf{b}_m^{(0)}} p(\mathbf{y}[n]|x[n])\cdot P\left(x[n]|\mathbf{b}_m^{(0)}\right)\cdot \prod\limits_{m=1}^{N_{in}} P(b_{t,m}[n]=0)} \qquad [4.3]$$

where $\mathbf{b}_m^{(0)}$ and $\mathbf{b}_m^{(1)}$ are the sets of $2^{N_{in}-1}$ words of input bits with the mth position bit $b_m = 0$ or $b_m = 1$, respectively. The third term in [4.3] represents the *a priori* bit knowledge fed by the other decoder $L_a\left(b_{t,m}[n]\right)$. The first two terms in both sums from denominator and nominator of [4.3] represent the symbol probability that depends on *a priori* bit values and trellis encoder constraints, which are used in transition metric computation of multilevel log-MAP algorithms [OGI 01]. Each log-MAP block in Figure 4.2 evaluates [4.3] recursively using the Jacobian logarithm [OGI 01]:

$$ln\left(e^a + e^b\right) = max(a,b) + ln\left(1 + e^{-|a-b|}\right) \qquad [4.4]$$

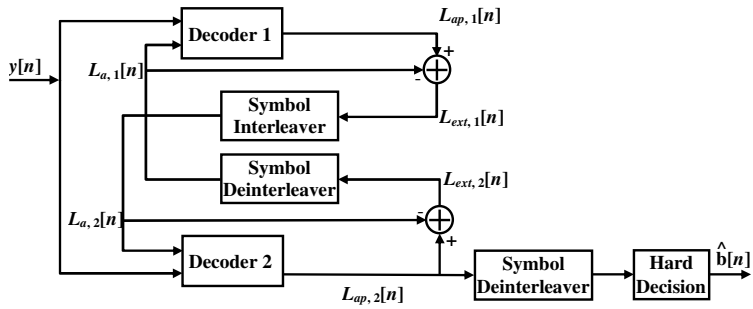


Figure 4.2. *Iterative turbo-TCM receiver*

Let us consider the same TTCM transmission scheme, shown in Figures 4.1 and 4.2, where a 2^N-QAM modulation is used instead of the 2^N-PSK modulation. In this case, equation [4.3] is evaluated iteratively as:

$$\begin{aligned} &L_{ap}\left(b_{t,m}[n]|\mathbf{y}[n]\right) = \\ &\max_{b_{t,m}[n]\in\mathbf{b}_m^{(1)}}{}^{*}\left(p\left(\mathbf{y}[n]|x[n]\right)\cdot P\left(x[n]|\mathbf{b}_m^{(1)}\right)\cdot\prod_{m=1}^{N_{in}}P\left(b_{t,m}[n]=1\right)\right) \\ &-\max_{b_{t,m}[n]\in\mathbf{b}_t^{(0)}}{}^{*}\left(p\left(\mathbf{y}[n]|x[n]\right)\cdot P\left(x[n]|\mathbf{b}_t^{(0)}\right)\cdot\prod_{m=1}^{N_{in}}P\left(b_{t,m}[n]=0\right)\right) \end{aligned} \qquad [4.5]$$

using an eight entries approximation table of the so-called Jacobian logarithm, given by [OGI 01]:

$$\max{}^{*}\left(a,b\right) = \ln\left(e^a + e^b\right) = \max\left(a,b\right) + \ln\left(1 + e^{-|a-b|}\right) \qquad [4.6]$$

Next, for any modulation scheme, the iterative decoder works in the same manner. Hence, the extrinsic information L_{ext} is calculated by subtracting the *a priori* LLR $L_a(b_{t,m}[n])$ from $L_{ap}(b_{t,m}[n])$. The extrinsic information shows the increment of the decoded symbol reliability. The extrinsic information sequence from the first log-MAP decoder (corresponding to the component encoder RC-LCIRC1) is interleaved and fed into the second component decoder as an *a priori* value, $L_{a,2}[n]$. It corresponds to the component encoder RC-LCIRC2. At the same time, the even order received symbols sequence is also fed into the second decoder and then this decoder calculates the extrinsic information, $L_{ap,2}[n]$. This extrinsic information sequence is deinterleaved and fed back into the first component decoder as an *a priori* value, $L_{a,1}[n]$, thus ending each iteration. The role of the deinterleaver is to rearrange the sequence of extrinsic information in the order corresponding to the received information from the first decoder component input. At the last iteration, a final decoded bit is obtained from the sign of $L_{ap,1}[n]$.

4.2. New recursive and systematic convolutional nonlinear encoders for parallel TTCM schemes

In this section, a new family of rate-1 recursive and systematic convolutional (RSC) encoders operating over Galois field GF$\left(2^N\right)$ and their use for TCM schemes are presented. In the following, all notations are kept the same as they were in section 3.2.3, and will be in all the subsequent sections.

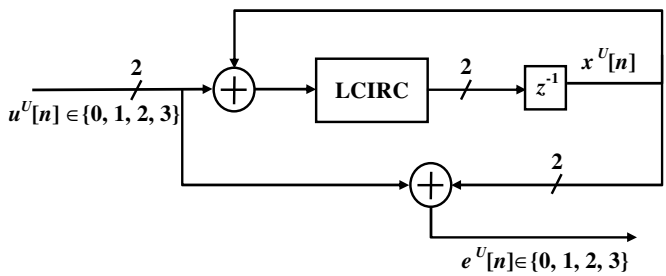


Figure 4.3. *Rate* 1 *GF*(4) *nonlinear RSC-LCIRC encoder for* 2 *b/s/Hz*

Let us consider an RSC encoder working over GF(4) using the LCIRC function. This scheme is shown in Figure 4.3. In this particular case, we have

a word length of two bits per sample, i.e. $N = 2$. This encoder is composed of one delay element with a sample interval, two modulo-$\left(2^N\right)$ adders, and a LCIRC block.

For each moment n, $u^U[n]$ represents the input data sample, $x^U[n]$ denotes the delay output or the encoder current state and $e^U[n]$ is the output sample. The superscript U denotes that all the samples are represented in unsigned form, using either N_{in} or N bits per word, i.e. $u^U[n] \in \left[0, 2^{N_{in}} - 1\right]$, $e^U[n] \in \left[0, 2^N - 1\right]$. The encoding rate for the encoder in Figure 4.3 is the ratio between the input word length N_{in} and the output word length $N_{out} = N$, i.e. $R = N_{in}/N = 1$, because $N_{in} = N = 2$ [VLA 09a].

The trellis for the encoder in Figure 4.3 is shown in Figure 4.4 and does not follow the Ungerboeck rules [UNG 82, VLA 09a]. This trellis has four states because the sample determining the encoder state takes four values, i.e. $x^U[n] \in \{0, 1, 2, 3\}$.

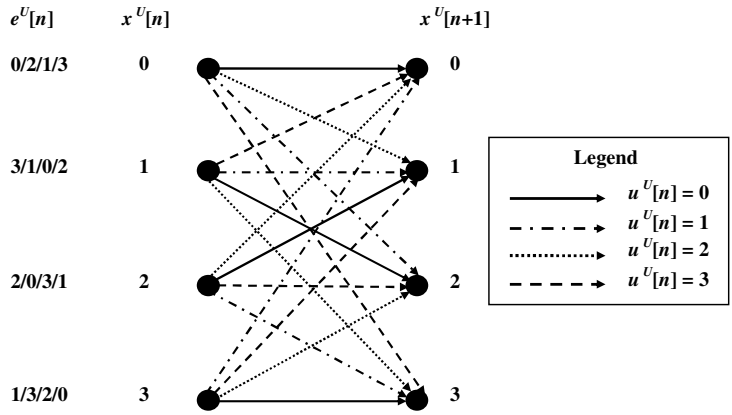


Figure 4.4. *Trellis for rate* 1 *GF*(4) *nonlinear RSC-LCIRC encoder (*2 *b/s/Hz)*

In Figure 4.4, four different lines are used for representing the transitions corresponding to the input sample $u^U[n]$. Each transition is associated with an unsigned output value $e^U[n] \in \{0, 1, 2, 3\}$. For each originating state, the values of $e^U[n]$, from left to right, are associated with the transitions in the descending order. Mapping an unsigned output symbol value $e^U[n]$ into a PSK or QAM symbol value is performed using the set partitioning (SP) over the

nth sample interval as in [UNG 82]. Therefore, a 2^N levels TCM scheme is obtained.

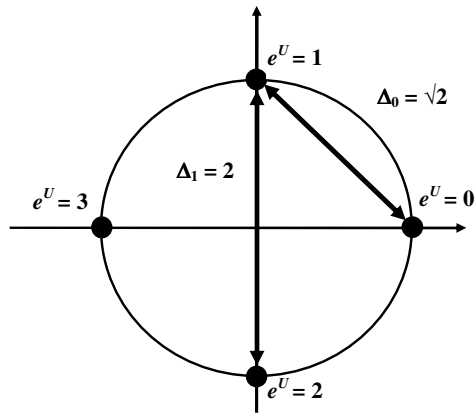


Figure 4.5. *Signal constellation for QPSK/4-QAM-TCM*

The signal constellation for the quaternary phase-shift keying (QPSK)/4-QAM-TCM scheme using the encoder in Figure 4.3 and the SP mapping is shown in Figure 4.5. Considering the SP mapping for the constellation in Figure 4.5, it follows that the QPSK-TCM signal trellis in Figure 4.4 presents a minimum Euclidean distance of $d^2_{E,N=2,R=1,\mathrm{QPSK}} = 2 \cdot \triangle_0^2 = \triangle_1^2 = 4$, offering no coding gain over the non-encoded binary PSK (BPSK) signal.

Next, we will develop a new family of optimum RSC-LCIRC encoders. First, we present the scheme optimization method applied to the particular case of $N = 2$, as shown in Figure 4.3. The resulted encoder shows a significant coding gain over the initial encoder. Second, the scheme optimization method is generalized, for any values of N and encoding rate $R = N_{in}/N$. The novelty of these encoders compared to the encoders introduced in [VLA 10b] and presented in section 3.2.3 consists of the design of a systematic output. This is an imperative requirement for TTCM schemes.

Following the trellis optimization presented in [VLA 09b] and [VLA 10b], a simple nonlinear encoder operating over GF(4) was developed, which has a binary input. Compared to the scheme in Figure 4.3, two blocks are added: a multiplier by 2 and a modulo-2 parity extractor. The modulo-2 block extracts

the least significant bit, denoted by $p[n]$, from the encoder current state value, $x^U[n]$. It is demonstrated that this encoder performs identically to an optimum rate $1/2$ binary field RSC convolutional encoder. Both encoders offer maximum coding gain for 1 b/s/Hz [UNG 82]. The scheme of the rate $1/2$ optimum GF(4) encoder is shown in Figure 4.6.

The trellis for the encoder in Figure 4.6 is shown in Figure 4.7 and follows all the Ungerboeck rules. Considering the SP mapping, the QPSK-TCM signal trellis in Figure 4.7 shows a minimum Euclidean distance of $d^2_{E,N=2,R=1/2,\mathrm{QPSK}} = 2 \cdot \triangle_1^2 + \triangle_0^2 = 5 \cdot \triangle_0^2 = 10$ for a spectral efficiency of 1 b/s/Hz. Hence, this rate $1/2$ code for 1 b/s/Hz QPSK-TCM transmission offers a coding gain of $10 \cdot \log_{10}(2.5) = 4$ dB over the rate 1 QPSK-TCM in Figure 4.3.

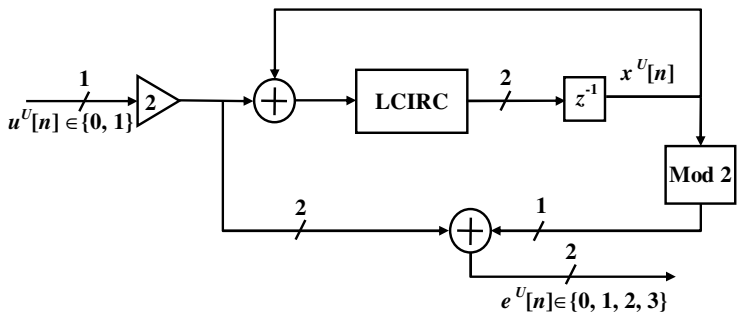


Figure 4.6. *Rate $1/2$ GF(4) nonlinear RSC-LCIRC encoder for 1 b/s/Hz*

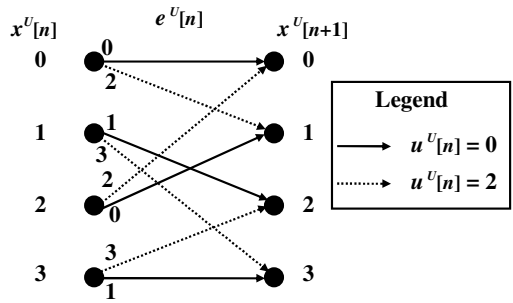


Figure 4.7. *Trellis for rate $1/2$ optimum GF(4) RSC-LCIRC encoder (1 b/s/Hz)*

Now, we aim to generalize the RSC-LCIRC encoder from Figure 4.6 operating over Galois field GF$\left(2^N\right)$ and their use for optimum TCM schemes.

Optimum encoders using the LCIRC function for TCM schemes were introduced in [VLA 10b]. However, despite being characterized by optimum Euclidean distances, these RC encoders are non-systematic. Therefore, the coding performances of these non-systematic encoders are not optimum when used in turbo schemes. Next, we will introduce a new encoder operating over Galois field GF$\left(2^N\right)$, using the LCIRC function, which is systematic, i.e. the encoder output value specifies explicitly the input value.

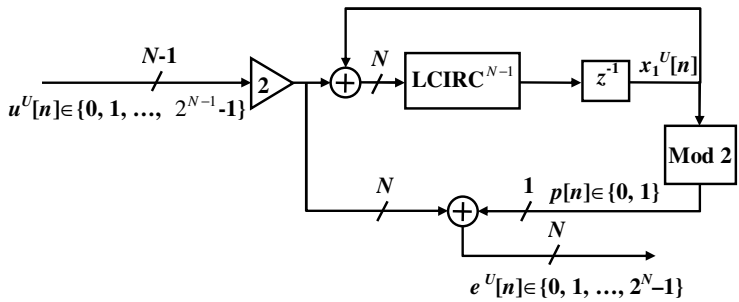


Figure 4.8. *Rate $(N-1)/N$ optimum GF(2^N) RSC-LCIRC encoder*

The block scheme for a rate $(N-1)/N$ optimum RSC-LCIRC encoder, using one delay element and the LCIRC function, is shown in Figure 4.8. For each moment n, $u[n]$ represents the input data sample, $x_1[n]$ denotes the delay output or the encoder current state and $e[n]$ is the output sample. The superscript U denotes that all the samples are represented in unsigned N bits word length, i.e. $u^U[n] \in \left[0, 2^{N-1}-1\right]$, $e^U[n] \in \left[0, 2^N-1\right]$. The encoding rate for the encoder in Figure 4.8 is the ratio between the input word length $N-1$ and the output word length N, i.e. $R = (N-1)/N$ [VLA 09a]. $LCIRC^{N-1}$ represents the LCIRC function application for $N-1$ times consecutively. Both adders and the multiplier are modulo-2^N operators. The modulo-2 block extracts the least significant bit, denoted by $p[n]$, from the encoder current state value, $x_1[n]$. $p[n]$ is the parity bit for the systematic rate $(N-1)/N$ encoder.

It is important to demonstrate that the encoder is systematic, i.e. the encoder output binary representation code word $e^U[n]$ includes the representation code

word of the encoder input $u^U[n]$. Hence, the output is obtained by shifting the $N-1$ bits of the input representation code word by one position to a higher significance and adding the parity bit $p[n]$ to the least significant position. The one position shifting presented above is equivalent to a multiplication by 2 in the GF$\left(2^N\right)$ field. Therefore, the encoder output value $e^U[n]$ is given by the following GF$\left(2^N\right)$ equation:

$$e^U[n] = 2 \cdot u^U[n] + p[n] = 2 \cdot u^U[n] + x^U[n] \bmod 2 \qquad [4.7]$$

We must note that for $N_{in} = N-1$, the binary representation code word of the systematic output includes all input bits, having the highest significance, and the last significant bit, which was left empty after the multiplication by 2, consists of the parity bit $p[n]$.

The trellis complexity of the codes generated with the scheme in Figure 4.8 increases with the word length N, because the number of trellis states grows exponentially with the output word length, i.e. 2^N, while the number of transitions originating from and ending in the same state grows exponentially with the input word length, i.e. $2^{N_{in}}$.

To introduce the RSC-LCIRC encoder shown in Figure 4.8 in a punctured TTCM scheme, it is important first to determine the optimum SP for the modulated signal. The SP for the punctured TTCM scheme, which optimizes the initialization of the *a priori* information for the first decoder, during the first iteration, was introduced in [ROB 98] and has two features. First, the SP follows the Ungerboeck optimum SP rules from [UNG 82], and second, the constellation points associated with the same group of $N-1$ systematic information bits, i.e. to the same input symbol $u^U[n]$, but differing in the least significant bit, i.e. the parity bit $p[n]$, should be placed at the minimum distance in the set, $\Delta_{0,2^N-\text{ary modulation}}$. Following these two requirements, the optimum SP rules for 8-PSK and 16-QAM are depicted in Figures 4.9 and 4.10, respectively. The first feature maximizes the minimum Euclidean distance of the component TCM code, while the second feature minimizes the distance between elements of the subsets associated with identical systematic bits, denoted by ovals in Figures 4.9 and 4.10, for the global punctured TTCM code.

It can be easily demonstrated that the minimum Euclidean distance for the 2^N-ary TCM encoder shown in Figure 4.8, using the optimum SP

constellations, has the expression given in [4.8], where Δ_i denotes the ith Euclidean distance in the ascending order of the natural index i, for the 2^N-ary PSK and QAM constellations.

$$d^2_{E,R=(N_{in})/N} = \begin{cases} 2 \cdot \left(\Delta_{2^{N-N_{in}}-1}\right)^2 + \sum_{i=0}^{2^{N-N_{in}}-2} \left(\Delta_i\right)^2, \\ \quad \text{for } 2^N - \text{PSK}, N_{in} \neq \frac{N}{2} \\ 2 \cdot \left(\Delta_{2^{N-N_{in}}-1}\right)^2 + \left(\Delta_0\right)^2, \\ \quad \text{for } 2^N - \text{PSK}, N_{in} = \frac{N}{2} \\ \left(2^{N-N_{in}+1} + \sum_{i=0}^{N-N_{in}-1} \left(2^i\right)\right) \cdot \Delta_0^2, \\ \quad \text{for } 2^N - \text{QAM}, N_{in} \neq \frac{N}{2} \\ \left(2^{N-N_{in}+1} + 1\right) \cdot \Delta_0^2, \\ \quad \text{for } 2^N - \text{QAM}, N_{in} = \frac{N}{2} \end{cases} \quad [4.8]$$

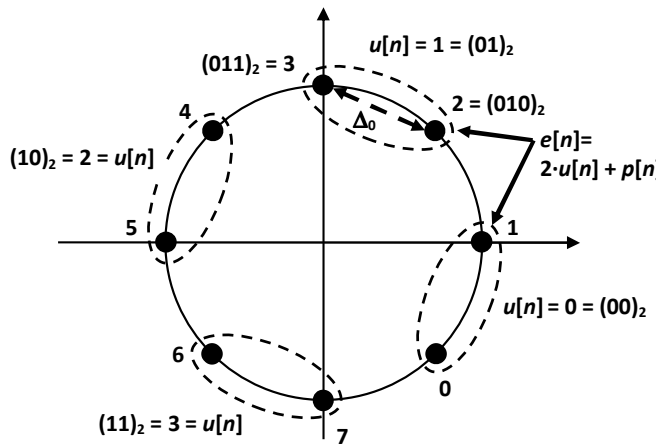


Figure 4.9. *Set partitioning for punctured 8-PSK TTCM*

Using the constellations in Figures 4.9 and 4.10, the minimum Euclidean distance given in [4.8] can be rewritten in a more compact form as follows:

$$d^2_{E,R=(N-1)/N} = \begin{cases} 2\Delta^2_{1,2^N-\text{PSK}} + \Delta^2_{0,2^N-\text{PSK}}, \text{for PSK} \\ 5\Delta^2_{0,2^N-\text{QAM}}, \text{for QAM} \end{cases} \quad [4.9]$$

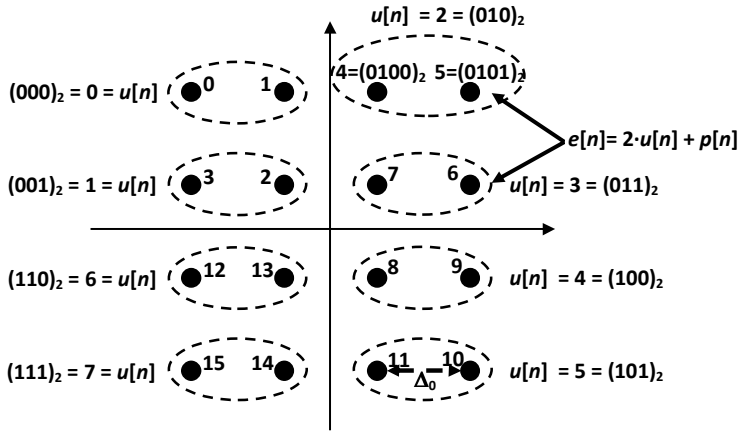


Figure 4.10. *Set partitioning for punctured 16-QAM TTCM*

In Table 4.1, some values of the minimum distance of the TCM encoder in Figure 4.8 are presented for different values of N and for the PSK and QAM constellations, respectively. The associated coding rates are presented in the second column. It can be easily seen from [4.8] that, for all the rates $(N-1)/N$, for any N value, the RSC-LCIRC encoders have the same minimum distances as the corresponding binary optimum encoders [ROB 98, VUC 00]. When transmitting over a Rayleigh non-selective fading channel, the TCM system performances depend mainly on two parameters, i.e. the minimum effective length and the minimum product distance [SCH 89]. We call the minimum effective length l_m the length of the shortest path pair of encoder output values $(x[n],\ x'^{[n]})$. Among these paths of length l_m, there is one which has the smallest product distance $d_p^2 = \Pi_{n=1,x[n]\neq x'[n]}^{l_m} \mid x[n] - x'[n] \mid^2$. The values of these parameters are presented in the last two columns in Table 4.1, for the same RSC-LCIRC codes. Again, the RSC-LCIRC encoders have the same values for minimum effective length and product distance, as their binary counterparts [SCH 89]. However, the GF$\left(2^N\right)$ RSC-LCIRC encoders are less complex than the corresponding binary encoders in terms of memory usage. The memory size of the binary encoders increases logarithmically with the number of states in the trellis, while the GF$\left(2^N\right)$ RSC-LCIRC encoders include only one delay element, no matter what the trellis complexity is. As another advantage of these encoders, we can also mention the Euclidean distance compact expression [4.8] as a function of N.

N	R	Modulation	d_E^2	l_m	d_p^2
2	1/2	QPSK, 4-QAM	10	3	32
3	2/3	8-PSK	≈ 4.5858	2	8
4	3/4	16-PSK	≈ 1.3238	2	≈ 1.1716
4	3/4	16-QAM	2	2	1.28
5	4/5	32-QAM, cross	1	2	0.32
6	5/6	64-QAM	≈ 0.4762	2	≈ 0.0725

Table 4.1. *Minimum 2^N-ary TCM distances as function of N for optimum GF(2^N) RSC-LCIRC encoders*

Hence, it was demonstrated that optimum RSC encoders over GF$\left(2^N\right)$ can be designed using the LCIRC function. A generalized one-delay GF$\left(2^N\right)$ RSC-LCIRC encoder scheme was defined, for any possible encoding rate. A general expression is found for the minimum Euclidean distances of PSK-TTCM and QAM-TCM schemes using these optimum encoders. As an advantage of this generalized encoder, we can mention its reduced complexity and the systematic output feature. Hence, using only one delay element and multiple bit circulations, we designed encoders which have complex trellises and large Euclidean distances. As an advantage of this generalized encoder, we can also mention the compact expressions for the minimum Euclidean distances for PSK and QAM TCM encoders, as a function of the symbol representation word length N. Also, the scheme of this encoder shows structural universality, i.e. the coding performances are controlled only by N. In addition, it was shown that LCIRC-based encoders offer at least the same performances as conventional binary encoders for the rate-$(N-1)/N$ schemes. Considering the systematic property of the RSC-LCIRC encoders presented above, we also aim to analyze the TTCM scheme over GF$\left(2^N\right)$ using puncturing for encoding rate increase. This scheme will be presented in the following section.

4.3. Punctured TTCM transmissions using recursive systematic convolutional nonlinear encoders

Figure 4.11 shows the TTCM transmitter for 2^N-ary PSK modulation. The information 2^{N-1}-ary symbol sequence $u[n]$ and its blockwise interleaved version $u^{(i)}[n]$ are fed into two identical component encoders RSC-LCIRC$_1$

and RSC-LCIRC$_2$ of rate $(N-1)/N$. The encoders' outputs are selected alternatively and mapped into 2^N-ary modulated symbol sequence $x[n]$. The output of the bottom encoder is deinterleaved according to the inverse operation of the interleaver. This ensures that at the input of the symbol selector, the $N-1$ information bits from the 2^{N-1}-ary input symbol, partly defining the encoded 2^N-ary symbols of both the upper and lower input, are identical [ROB 98]. Therefore, if the selector is switched on a symbol base, the mapper output is a punctured version of the two encoded sequences and the $N-1$ information bits appear only once, mapped in a single transmitted symbol selected either from $e_1[n]$ sequence or from $e_2[n]$ sequence.

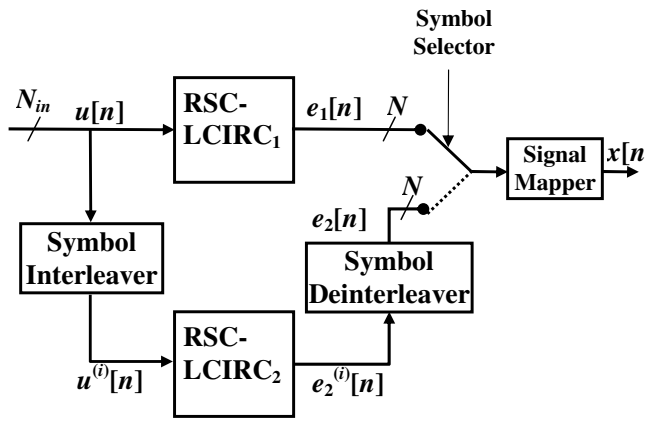


Figure 4.11. *TTCM transmitter with RSC-LCIRC encoders and symbol puncturing*

Nevertheless, the remaining parity bit carried by the transmitted symbol is taken alternatively from the upper and lower encoders. Hence, the overall coding rate for the scheme in Figure 4.11 is $(N-1)/N$. The 2^N-levels modulated symbol sequence is transmitted over a noisy and non-selective fading channel. The received signal over the nth symbol interval is given by:

$$y[n] = h[n]x[n] + w[n] \qquad [4.10]$$

where $w[n]$ is an AWGN sequence with $E\left[|w[n]|^2\right] = N_0$ and $x[n]$ denotes the 2^N-levels symbol value mapped from the encoders output sequences $(e_1[n], e_2[n])$ by puncturing over the nth symbol interval. The coefficient $h[n]$

is the path gain from the transmitting antenna to the receiving antenna, having a Rayleigh distribution. The path gains are modeled as the absolute part of samples of independent complex Gaussian random variables with variance 0.5 per real dimension. The wireless channel is assumed to be quasistatic, i.e. the path gain value is constant over a group of symbol intervals and varies from one group of symbols to another.

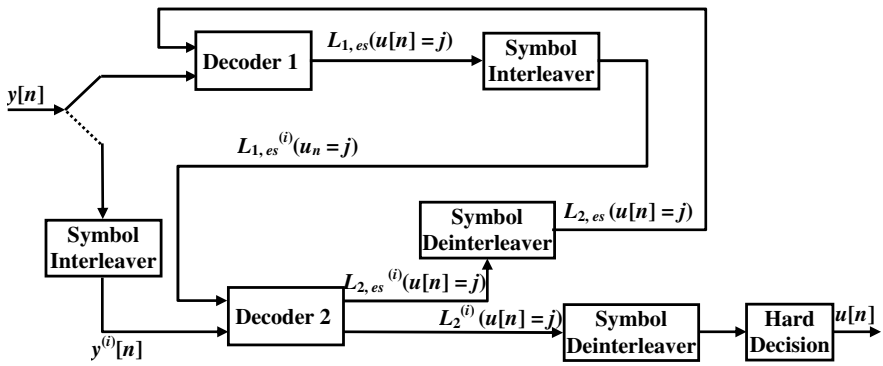


Figure 4.12. *TTCM receiver using symbol log-MAP decoders and puncturing*

The receiver structure, as shown in Figure 4.12, has two component decoders that use the symbol-by-symbol log-MAP algorithm introduced in [ROB 98]. We assume that the receiver has perfect side information on the path gains ($h[n]$). The decoding process is similar to the binary turbo decoding in [BER 96], except that the symbol probability is used as the extrinsic information rather than the bit probability [ROB 98, VUC 00]. The log-MAP decoder computes the LLR for each group of information bits transmitted at the nth symbol interval $u[n]$, embedded in the 2^{N-1}-ary input symbol taking one of the integer values $j \in \{0, 1, ..., 2^{N-1} - 1\}$ as [VUC 00]:

$$\begin{aligned} L\left(u[n]=j\right) &= \log \frac{P\left(u[n]=j|\mathbf{y}\right)}{P\left(u[n]=0|\mathbf{y}\right)} \\ &= \log \frac{\sum_{(l',l)\in B_n^j} \alpha_{n-1}\left(l'\right)\gamma_n^j\left(l',l\right)\beta_n\left(l\right)}{\sum_{(l',l)\in B_n^0} \alpha_{n-1}\left(l'\right)\gamma_n^0\left(l',l\right)\beta_n\left(l\right)} \end{aligned} \quad [4.11]$$

where $\mathbf{y}$ is the received signal vector, B_{nj} represents the set of trellis transitions at the nth symbol interval determined by an input symbol $u[n] = j$, denoted

as $(S_{n-1} = l' \rightarrow S_n = l)$ where S_n is the trellis state at moment n, and the probabilities $\alpha_n(l)$, $\beta_n(l)$ and $\gamma_n(l', l)$, denoting the forward, backward and the transition metrics, are computed recursively as in [VUC 00]. The symbol-by-symbol log-MAP decoder operates on an L symbols block basis. Hence, in all equations the symbol time variable n takes values between 1 and L. The receiver transition metric is given by:

$$\gamma_n^j\left(l', l\right) = \begin{cases} \frac{P(u[n]=j)}{P(u[n]=0)} \exp\left(-\frac{|y[n]-x[n]|^2}{2\sigma^2}\right), \text{ if } (l', l) \in B_n^j \\ 0, \quad \text{otherwise} \end{cases} \quad [4.12]$$

The transition metric in [4.12] is normalized over all input symbol values as follows:

$$\gamma_n\left(l', l\right) = \log \sum_{j=0}^{2^{N-1}-1} \gamma_n^j\left(l', l\right) \quad [4.13]$$

The first term in [4.12] denotes the *a priori* information for the transmitted input symbol j. The number of trellis states for each component RSC-LCIRC encoder is 2^N. The forward normalized metric is estimated as follows:

$$\alpha_n(l) = \log \sum_{l'=0}^{2^N-1} \exp\left[\alpha_{n-1}\left(l'\right) + \gamma_n\left(l', l\right)\right] \quad [4.14]$$

The recurrence in [4.14] is initialized with:

$$\alpha_0(0) = 0; \quad \alpha_0(l)|_{l\neq 0} = -\infty \quad [4.15]$$

The backward normalized metric is estimated as:

$$\beta_n(l) = \log \sum_{l'=0}^{2^N-1} \exp\left[\beta_{n+1}\left(l'\right) + \gamma_{n+1}\left(l', l\right)\right] \quad [4.16]$$

and the recurrence in [4.16] is initialized with:

$$\beta_L(0) = 0; \quad \beta_L(l)|_{l\neq 0} = -\infty \qquad [4.17]$$

The input symbol j with the largest LLR in [4.11] is chosen as the hard decision output.

In contrast to the binary turbo codes, in the TTCM case, we cannot separate the influence of the information and parity-check components within one symbol. The systematic information and the extrinsic information are not independent. Thus, both systematic and extrinsic information will be exchanged between the two component decoders. The joint extrinsic and systematic information of the first log-MAP decoder, denoted as $L_{1,es}(u[n] = j)$, is computed as:

$$L_{1,es}\left(u[n] = j\right) = L_1\left(u[n] = j\right) - \log \frac{P\left(u[n] = j\right)}{P\left(u[n] = 0\right)} \qquad [4.18]$$

The last term in [4.18] represents the *a priori* information symbol knowledge fed by the other decoder.

The joint extrinsic and systematic information $L_{1,es}\left(u[n] = j\right)$ is used as the estimate of the *a priori* LLR at the next decoding stage. After interleaving, this term is denoted as $L^{(i)}_{1,es}\left(u[n] = j\right)$. The joint extrinsic and systematic information of the second decoder is given by:

$$L_{2,es}(u[n] = j) = L_2(u[n] = j) - L^{(i)}_{1,es}(u[n] = j) \qquad [4.19]$$

In the next iteration, the *a priori* term in [4.18] is replaced by the deinterleaved joint extrinsic and systematic information from the second decoding stage, denoted as $L^{(i)}_{2,es}\left(u[n] = j\right)$.

It must be stated that for the symbol-by-symbol log-MAP decoding, each component decoder should avoid using the same systematic information twice in every iterative decoding step. In the TTCM scheme, each decoder alternately receives the noisy output of its own encoder and that of the other encoder. As mentioned before, the parity bit in every second received symbol

was generated by the other encoder, due to the symbol puncturing in the transmitter. The decoder ignores this symbol by setting the branch metric to zero. The only input at this decoding step consists of the *a priori* component obtained from the other decoder, which contains the systematic information. All the mentioned LLRs and the relations between them are represented in the TTCM receiver scheme in Figure 4.12.

The iterative metric computation presented above assumes that the *a priori* LLR is already available. Nevertheless, in the first iteration, the *a priori* LLR for the first decoder is unavailable. Considering the symbol mapping shown in Figure 4.9, the *a priori* information for the first decoder regarding the punctured symbols, i.e. the input symbols encoded by the second encoder and transmitted in the even positions n, is determined partially by the systematic information input symbol $u[n]$ and also by the unknown parity bit $p[n] \in \{0, 1\}$ generated by the second encoder (see equation [4.7]). Using the mixed Bayes' rule, the *a priori* probability is given by [ROB 98]:

$$P\left(u[n]=j\right) = \text{const} \cdot \sum_{k\in\{0,1\}} p(y[n]|u[n]=j,\, p[n]=k) = \\ \text{const} \cdot \sum_{k\in\{0,1\}} \exp\left(-\frac{|y[n]-h[n]x^{j,\,k}[n]|^2}{2\sigma^2}\right) \qquad [4.20]$$

where $const$ is a constant value, assuming that all the values of $u[n]$ and $p[n]$ are equally probable, and the 2^N-ary modulated symbol $x[n]$, transmitted by the second encoder, is given by $x^{j,k}[n] = 2 \cdot u[n] + p[n] = 2 \cdot j + k$. The *a priori* LLR is computed by normalizing the values in [4.20] by their sum estimated for all the values of $j \in \{0, 1, ..., 2^{N-1} - 1\}$. If the first decoder operates over the odd symbols $x[n]$, the *a priori* LLR is initialized with equally probable values assuming that $\Pr\left(u[n]=j\right) = 1/2^{N-1}$.

It was shown that the proposed optimum RSC-LCIRC encoder can be used as a component encoder in TTCM schemes with punctured parity check bits (symbols). Also, due to the inner non-binary operation of the RSC-LCIRC encoder, the symbol-by-symbol log-MAP algorithm proves to be suitable for iterative decoding. The nonlinear LCIRC function leads to low complexity encoders, while the systematic property attains good performances in punctured schemes.

4.4. Extrinsic information transfer (EXIT) charts analysis for TTCM schemes using nonlinear RSC encoders

A very important tool for the iterative decoding performances analysis consists of the extrinsic information transfer (EXIT) chart, which describes the extrinsic mutual information exchange between constituent decoders. A complexity efficient method for generating the symbol-based EXIT charts from symbol-based *a posteriori* probabilities (APPs) was proposed in [KLI 06]. The expression for the average extrinsic information $I_E(u)$, estimated at the output of the decoder for the input symbol vector u, is given by [KLI 06]:

$$I_{E,D}(u) = N-1 + \frac{1}{L}\sum_{n=1}^{L}\mathbf{E}\left[\sum_{i=1}^{2^{N-1}} e_D\left(u^{(i)}[n]\right)\cdot\log_2\left(e_D\left(u^{(i)}[n]\right)\right)\right] \quad [4.21]$$

where L is the number of information symbols in the decoded block, $N-1$ is the number of information bits per input symbol, $u^{(i)}[n]$ is the presumed transmitted information symbol at time instant n for $i \in \{1, 2, ..., 2^{N-1}\}$ and $e(\cdot)$ is the extrinsic probability. The expectation $\mathbf{E}[\cdot]$ can be approximated by simple time-averaging of the extrinsic probabilities of the information symbol. We propose to approximate the extrinsic probability as the normalized joint extrinsic and systematic information of the log-MAP decoder, from equations [4.18] and [4.19]:

$$e_D\left(u^{(i)}[n]\right) \approx \frac{\exp\left(L_{D,es}\left(u[n]=i\right)\right)}{\sum_{i=1}^{2^{N-1}}\exp\left(L_{D,es}\left(u[n]=i\right)\right)} \quad [4.22]$$

In equations [4.21] and [4.22], D denotes the decoder number, i.e. $D \in \{1, 2\}$. However, computing an exact value of the extrinsic probability requires that the systematic and parity parts of the channel observation variables are independent. Thus, equation [4.22] represents only an approximation of the true extrinsic information. The average *a priori* information $I_A(u)$ is computed in a similar manner. The EXIT chart is obtained by representing, on the same diagram, the decoder 1 transfer characteristic, i.e. $I_{E,1} = T(I_{A,1})$ and the decoder 2 transfer characteristic,

i.e. $I_{E,2} = T(I_{A,2})$, for a constant E_b/N_0 value. The axes of the decoder 2 transfer characteristic are swapped.

4.5. Performance analysis of TTCM data communications using nonlinear digital encoders: simulation results

The TTCM scheme proposed in section 4.1 was tested for 8-PSK and 16-QAM modulations by simulation over an AWGN channel, using a two-bitwise block interleaver of length 31×31. Both component encoders in the TTCM scheme in Figure 4.1 are identical rate-$2/3$ RC-LCIRC encoders for 8-PSK and rate-$3/4$ for 16-QAM, respectively. Hence, the overall coding rate is $1/3$ for 8-PSK modulation and $3/8$ for 16-QAM modulation, respectively. The TCM employed the 8-PSK modulation with three mapping rules: Gray, natural and SP [BER 96] and the 16-QAM modulation with two mapping rules: Gray and anti-Gray, where symbols at minimum Euclidean distance differ in one bit or in maximum number of bits, respectively [SCH 03].

The TTCM scheme performances are estimated by simulation as bit error rate (BER) versus E_b/N_0, where E_b is the signal energy per one information bit and N_0 is one-sided power spectral density of the background noise.

Figure 4.13 shows the BER performances of TTCM using Gray labeling. At third iteration, we observe a 7 dB gain as compared with non-coding scheme, for BER = 4×10^{-5}. This scheme provides a 5 dB gain over non-iterative scheme employing a symbol phase detector in the receiver and 3.5 dB improvement over three iterations.

The BER performances using natural labeling over the first three iterations are shown in Figure 4.14. Further iterations do not improve decoder/detector performances. This scheme offers about 8 dB gain versus encoded 8-PSK modulation and 2 dB gain as compared with the non-iterative scheme.

BER performances for SP labeling are shown in Figure 4.15. This scheme attained the best threshold from all three mapping rules. It needs 3 dB and eight iterations to perform at BER = 3×10^{-5}. It provides a 10 dB gain versus uncoded modulation and 3.7 dB gain compared with the non-iterative scheme.

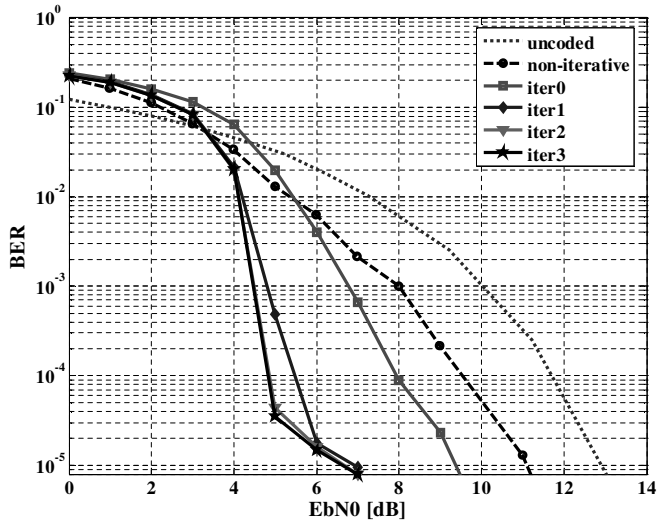


Figure 4.13. *BER for 8-PSK turbo-TCM with RC-LCIRC encoders and Gray mapping*

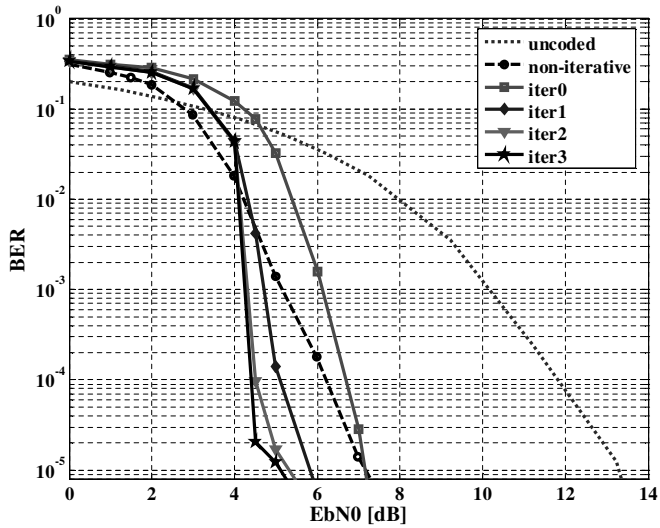


Figure 4.14. *BER for 8-PSK turbo-TCM with RC-LCIRC encoders and natural mapping*

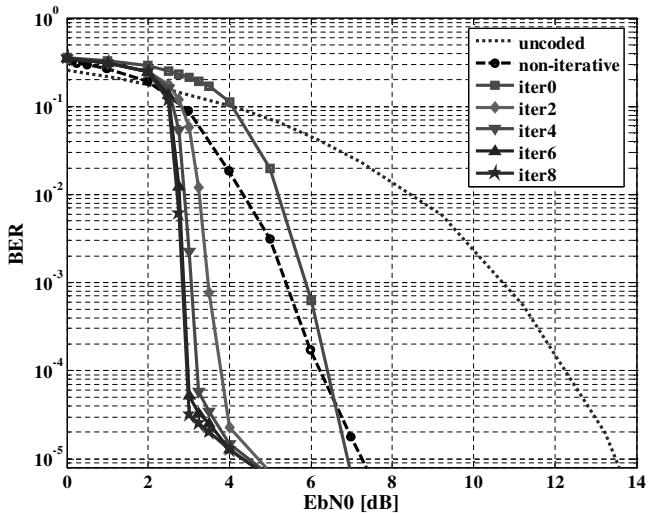


Figure 4.15. *BER for 8-PSK turbo-TCM with RC-LCIRC encoders and set partitioning mapping*

Figure 4.16 shows the BER performances of TTCM using 16-QAM, Gray labeling. At the fifth iteration, we observe a 7 dB gain compared with the non-coding scheme, for BER = 4×10^{-5}. This scheme provides a 5.5 dB gain over non-iterative scheme and 4 dB improvement through five iterations.

In Figure 4.17, the BER performances of 16-QAM, anti-Gray labeling over first five iterations are depicted. Further iterations do not improve decoder/detector performances. This scheme offers about 8 dB gain versus encoded 16-QAM modulation and 4.25 dB gain compared with the non-iterative scheme.

The increased throughput of 16-QAM modulation has a penalty of 1 dB and 2 dB for Gray and anti-Gray mapping, respectively, compared to 8-PSK modulation.

The next two figures depict the same transmission scenarios, but for the case of a non-selective Rayleigh fading channel. Figure 4.18 shows the BER performances of TTCM using 8-PSK modulation over the fading channel. It needs 3 dB and four iterations to perform at BER under 10^{-5}. It provides a 6.5 dB gain versus non-encoded modulation and 4.5 dB gain compared with the non-iterative scheme.

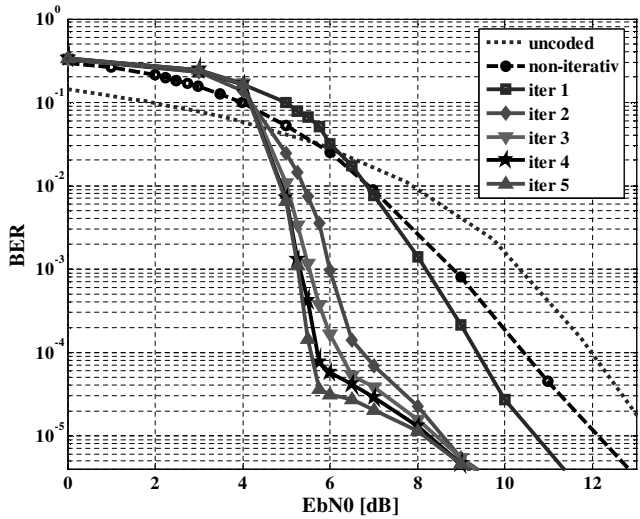


Figure 4.16. *BER for 16-QAM turbo-TCM with RC-LCIRC encoders and Gray mapping*

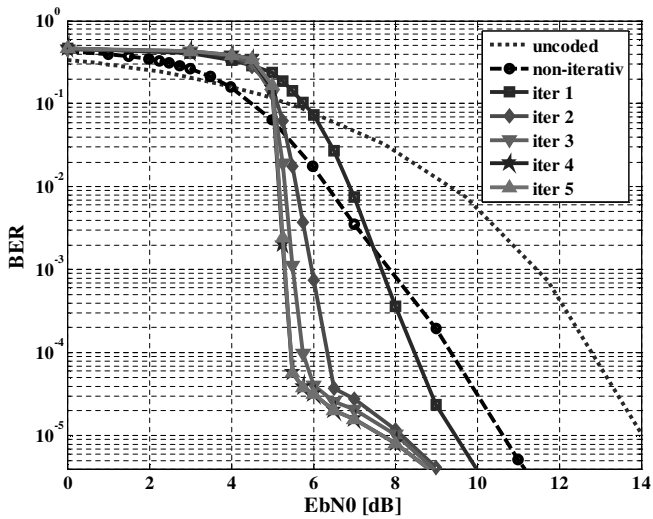


Figure 4.17. *BER for 16-QAM turbo-TCM with RC-LCIRC encoders and anti-Gray mapping*

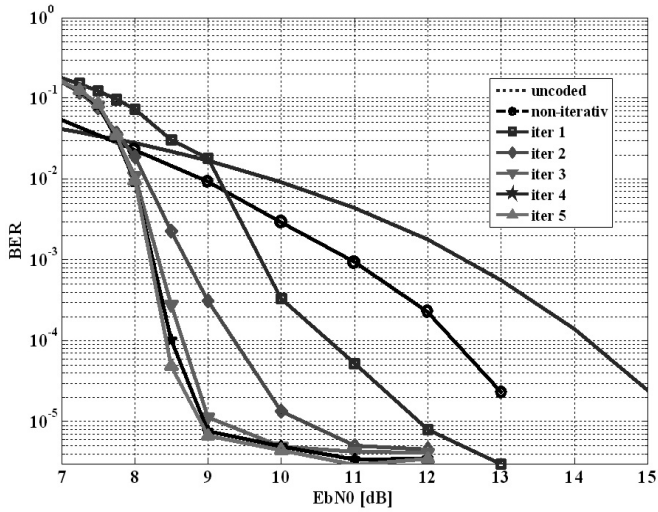


Figure 4.18. *BER for 8-PSK turbo-TCM with RC-LCIRC encoders and anti-Gray mapping over non-selective fading channel*

In Figure 4.19, the BER performances of 16-QAM transmission over the fading channel are depicted. This scheme offers about 6 dB gain versus non-encoded 16-QAM modulation and 4 dB gain compared with the non-iterative scheme.

We can note that the BER floor is approximately 10^{-5} for all simulations. This is due to low constraint length of the block interleaver, i.e. 961 symbols and is relatively independent of the modulation type and mapping rule.

The TCM scheme using the RSC-LCIRC encoders presented in section 4.2 was tested for rate $1/2$ QPSK, rate $2/3$ 8-PSK and rate $3/4$ 16-QAM by means of simulations over an AWGN channel. The modulations use the optimum SP in the transmitter and the Viterbi sequence detection with hardware decisions in the receiver. The simulation results are represented as symbol error rate (SER) performances versus the signal-to-noise ratio SNR expressed in decibels, as depicted in Figure 4.20.

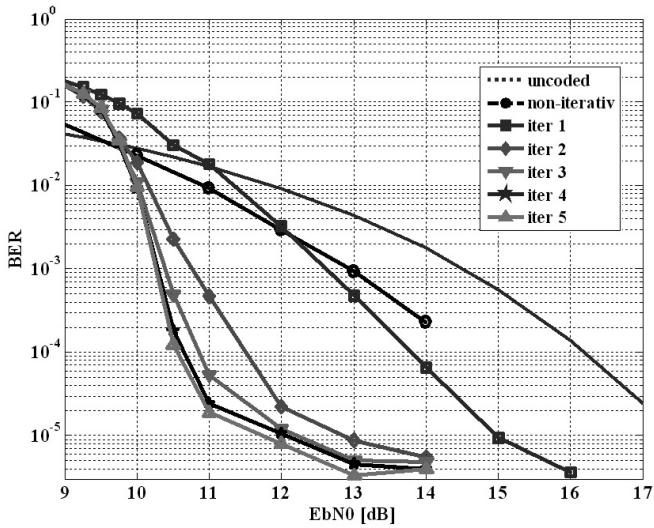


Figure 4.19. *BER for 16-QAM turbo-TCM with RC-LCIRC encoders and anti-Gray mapping over non-selective fading channel*

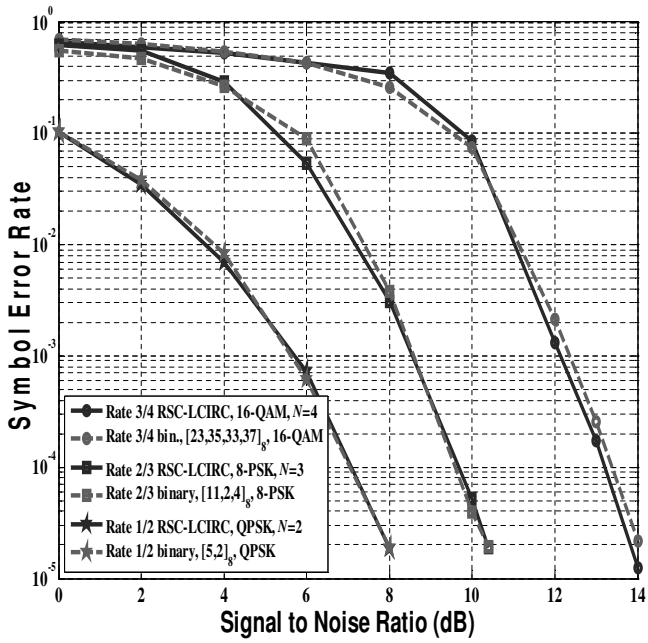


Figure 4.20. *SER performance for TCM schemes using optimum GF$\left(2^N\right)$ nonlinear RSC-LCIRC encoders*

As references, we considered for the same simulation scenario the optimum binary encoders with identical encoding rates values, $1/2$, $2/3$ and $3/4$, having the same number of states as the RSC-LCIRC encoders above, i.e. 4, 8 and 16, respectively. Their associated generator polynomials in ascending order and represented in octal notation are: $[05, 02]$, $[11, 02, 04]$ and $[23, 35, 33, 37]$ [UNG 82].

The simulation results show a perfect match of the SER performances between the optimum RSC-LCIRC encoders and their binary counterparts, for any encoding rate. This is explained by the fact that both encoders, RSC-LCIRC and its corresponding binary encoder, have the same Euclidean distance value. The common minimum Euclidean distance is presented in Table 4.1. Hence, in simulation, the QPSK-TCM transmission using a rate $1/2$ encoder has a coding gain, over the 8-PSK-TCM transmission using a rate $2/3$ encoder, of about 2.5 dB. According to the distances presented in Table 4.1, the expected theoretical coding gain is approximately $10 \cdot \log_{10}(10/4.5858) \approx 3.386\,\mathrm{dB}$. Here, the difference between the simulation and theoretical results is explained by the different average multiplicity of error events for rate $1/2$ encoders and rate $2/3$ encoders. However, it is interesting to note that for the same rate value, the RSC-LCIRC and the binary encoders have the same average multiplicity. Similarly, in simulation, the 8-PSK-TCM transmission using a rate $2/3$ encoder has a coding gain, over the 16-QAM-TCM transmission using a rate $3/4$ encoder, of approximately 3.5 dB. According to the distances presented in Table 4.1, the expected theoretical coding gain is approximately $10 \cdot \log_{10}(4.5858/2) \approx 3.6\,\mathrm{dB}$.

In the following, the TTCM scheme presented in section 4.3 using the RSC-LCIRC encoders presented in section 4.2 was tested for 8-PSK and 16-QAM by means of simulations over an AWGN non-faded channel and over an AWGN and non-selective fading channel, respectively. Both component encoders in the TTCM scheme are identical rate-$2/3$ RSC-LCIRC encoders for 8-PSK and rate-$3/4$ for 16-QAM. The modulation uses the optimum SP for the punctured TTCM scheme as presented in section 4.2. The symbol interleavers used for simulations are pseudo-random and operate independently on even and odd positions [ROB 98]. The symbol-by-symbol log-MAP decoding algorithm is used in the receiver. The following

simulation results are represented as BER performances versus E_b/N_0, where E_b is the signal energy per bit and N_0 is the one-sided power spectral density of the AWGN noise. The interleaver block includes $1,024$ symbols and the number of decoding iterations is eight. For all cases, further iterations do not significantly improve the BER performances. Figure 4.21 shows the BER performances for TTCM transmission, using 8-PSK with bandwidth efficiency of 2 b/s/Hz, over the non-faded channel (i.e. $h_n = 1$ in equation [4.10]). As a reference, we considered for the same simulation scenario the corresponding rate-$2/3$ optimum binary encoder with eight states, determined in [ROB 98] for 8-PSK TTCM with the generator polynomials, represented in octal notation $[11, 02, 04]$. The simulation results show a perfect match of the BER performances between the rate-$2/3$ RSC-LCIRC encoder and its binary counterpart. This is explained by the fact that both encoders have the same trellis. The common minimum Euclidean distance is presented in Table 4.1. Eight iterations are enough to reach a BER of 3×10^{-5} for $E_b/N_0 = 4.15$ dB. In Figure 4.22, the BER performances of the 16-QAM transmission over the non-fading channel, with bandwidth efficiency of 3 b/s/Hz, are depicted. The rate-$3/4$ optimum binary encoder with 16 states, considered as reference, was determined in [VUC 00] for 16-QAM TTCM with the generator polynomials $[23, 35, 33, 37]$. The results in Figure 4.22 show two important features. First, the BER convergence for 16-QAM TTCM is faster than for 8-PSK case, and second, the RSC-LCIRC encoder outperforms its binary counterpart at high E_b/N_0 values. Hence, the binary encoder gains less than 0.1 dB below an $E_b/N_0 = 5.8$ dB threshold, when compared to the corresponding RSC-LCIRC encoder. Above this threshold, the BER curves flatten and the RSC-LCIRC encoder outperforms its binary counterpart by almost one order of BER. For both RSC-LCIRC and binary encoders, eight iterations are enough to reach a BER of approximately 1.7×10^{-5} for the threshold value of $E_b/N_0 = 5.8$ dB. In Figure 4.23, BER performances after eight iterations, for the same modulation schemes transmitting over the non-fading channel, are depicted when the interleaver size is varied. Therefore, the interleaver block consecutively includes 512, $1,024$ and $2,048$ symbols. When the interleaver size is doubled, a coding gain of approximatively 0.4 dB is obtained for the 8-PSK modulation, while for 16-QAM modulation the coding gain varies between 0.2 and 0.5 dB.

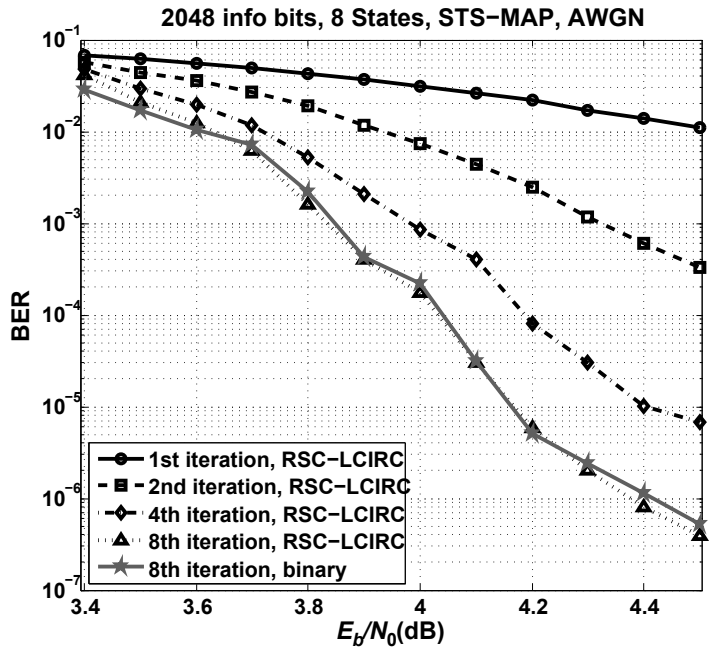


Figure 4.21. *BER performance for punctured 8-PSK-TTCM scheme using RSC-LCIRC encoders over an AWGN channel*

Figures 4.24 and 4.25 depict the performances of the same transmission scenarios but for the case of a non-selective Rayleigh fading channel. Figure 4.24 shows the BER performances of TTCM using 8-PSK modulation over the fading channel. As a reference, the same binary eight states encoder, as shown in Figure 4.21, was considered. Also, for the sake of clarity, the BER performances for the binary non-iterative 64 states 8-PSK TCM transmission, over the same fading channel, and using the symbol log-MAP detection, were considered. In the first iteration, the BER values for both binary- and LCIRC-encoded TTCM schemes overlap and take higher values than in the case of 64 states 8-PSK TCM. This is explained by the smaller number of trellis states in the first iteration, i.e. eight states for the TTCM constituent encoders, compared to 64 states for the 8-PSK TCM case. It can be seen in Figure 4.24 that the BER floor is reached after four iterations, for both TTCM schemes. Also, the curves for the fourth and the eighth iterations overlap for E_b/N_0 higher than 10 dB. Hence, it needs 9 dB and four iterations to perform at BER under 10^{-4} and it provides a coding gain larger than 5 dB

compared with the non-iterative 64 states 8-PSK TCM scheme. This is consistent with the results presented in [HAN 02]. Nevertheless, the RSC-LCIRC encoder outperforms its binary counterpart at E_b/N_0 values higher than 9 dB, by more than 1 dB. For the 8-PSK scheme, it is very important to note that in the case of the non-selective fading channel, the RSC-LCIRC encoder outperforms its binary counterpart in terms of BER (see Figure 4.24), while both have the same performances in the case of the non-faded AWGN channel (see Figure 4.21).

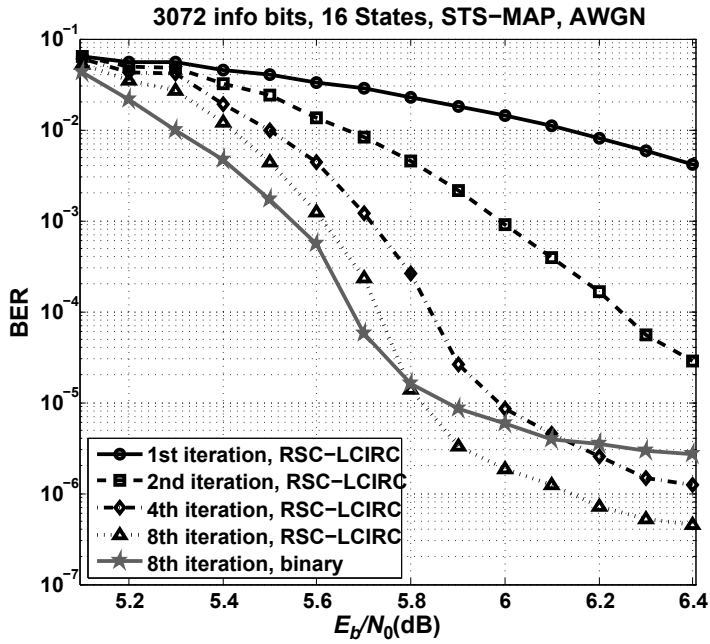


Figure 4.22. *BER performance for punctured 16-QAM-TTCM scheme using RSC-LCIRC encoders over an AWGN channel*

In Figure 4.25, the BER performances of 16-QAM transmission over the non-selective fading channel are depicted. As a reference, the same binary 16 states encoder was considered, as shown in Figure 4.22. In the first iteration, the BER values for both, binary- and LCIRC-encoded TTCM schemes, overlap. However, it can be seen in Figure 4.25 that starting with the second iteration, the BER for the binary-encoded TTCM scheme takes slightly smaller values. For the binary-encoded TTCM scheme, the receiver needs a

value of E_b/N_0 higher than 10.7 dB and eight iterations to perform at BER under 10^{-4} [HAN 02], while the RSC-LCIRC-TTCM receiver needs less than 10.8 dB to reach the same BER. Also, it is interesting to note that above $E_b/N_0 = 11.5$ dB the BER curves tend to floor and the binary encoder scheme provides a slightly lower floor, with a gain less than 0.1 dB. For the 16-QAM scheme, in the case of the non-selective fading channel, both the RSC-LCIRC encoder and its binary counterpart perform almost identically in terms of BER (see Figure 4.25 with the note regarding the BER floor), while RSC-LCIRC outperforms its binary counterpart in the case of the non-faded AWGN channel, for high E_b/N_0 (see Figure 4.22).

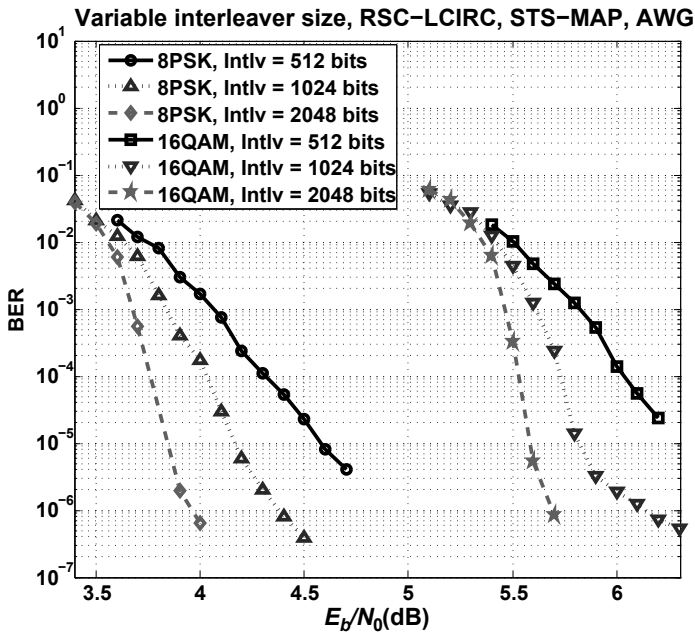


Figure 4.23. *BER performance for punctured RSC-LCIRC-TTCM scheme over an AWGN channel with variable interleaver size*

As a conclusion of BER performances analysis, we can state that considering only the simple TCM scheme performances meters (such as minimum Euclidean distance for AWGN channels or minimum effective length and product distance for Rayleigh fading channels) is not sufficient for the constituent encoder selection.

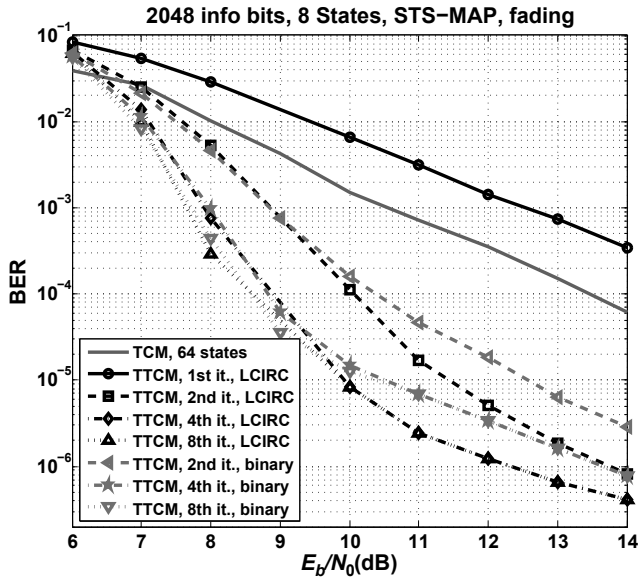


Figure 4.24. *BER performance for punctured 8-PSK-TTCM scheme using RSC-LCIRC encoders over a fading channel*

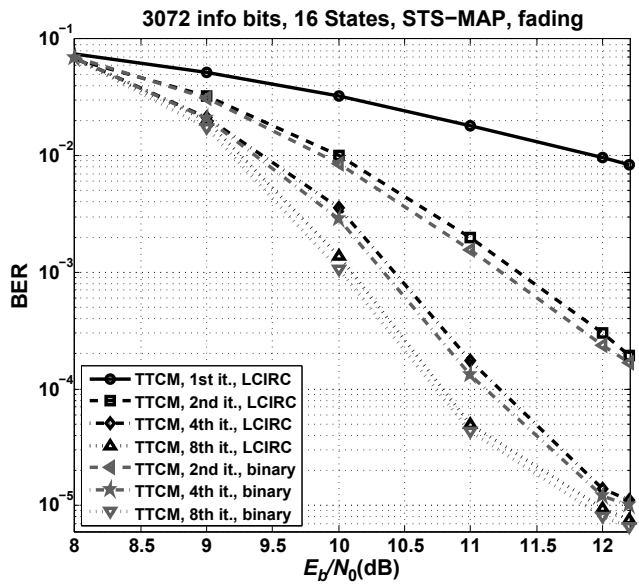


Figure 4.25. *BER performance for punctured 16-QAM-TTCM scheme using RSC-LCIRC encoders over a fading channel*

In the following, EXIT charts are contrived for 8-PSK and 16-QAM schemes presented above, using a simulation procedure described in [KLI 06]. These EXIT charts are relevant for the TTCM decoder convergence analysis, revealing important features, such as BER turbo cliff and BER floor regions. The average extrinsic information $I_E(u)$ and the average *a priori* information $I_A(u)$ are estimated using equation [4.21], assuming that the extrinsic probability is approximated with [4.22]. Figure 4.26 shows the extrinsic information transfer characteristic of a 8-PSK-RSC-LCIRC-TTCM decoder for a fading channel, assuming a variable E_b/N_0. The same scenario as in Figure 4.24 was considered: a non-selective Rayleigh fading channel, LCIRC-encoded 8-PSK TTCM scheme and a blocklength of 2^{14} symbols. Analyzing the curves in Figure 4.26, we can easily note that above $E_b/N_0 = 6.4$ dB, the average decoding trajectory obtained from real simulations shows convergence. In Figure 4.27(a), the EXIT chart for $E_b/N_0 = 6.5$ dB is depicted. This EXIT chart plots the *bottleneck region* with the decoding trajectory just managing to pass through a narrow tunnel, which corresponds to the BER waterfall region in Figure 4.24. In Figure 4.27(a), the convergence is almost reached after 30 iterations. The EXIT chart obtained under the same assumptions, for the corresponding binary encoder, is shown in Figure 4.27(b). It is clear that for the binary case decoder, the trajectory gets stuck, compared to the LCIRC decoder, for $E_b/N_0 = 6.5$ dB, assuming the same number of iterations. In Figure 4.27(c), the EXIT chart corresponding to the *wide-open region* is depicted. The scenario is identical to the previous one, but for $E_b/N_0 = 13$ dB. This region is related to the BER floor region in Figure 4.24. The trajectories for both binary and LCIRC decoders are depicted in Figure 4.27(c). Again, the LCIRC decoder outperforms its binary counterpart due to the wider opening of the EXIT chart. In this case, the convergence is almost reached after three iterations, a result that is also consistent with the one in Figure 4.24.

Figure 4.28 shows the extrinsic information transfer characteristic of a 16-QAM-RSC-LCIRC-TTCM decoder for a non-fading channel, assuming a variable E_b/N_0. The same scenario as in Figure 4.22 was considered, with a blocklength of 2^{14} symbols. Analyzing the curves in Figure 4.28, we can easily note that above $E_b/N_0 = 5.4$ dB, the average decoding trajectory shows convergence. In Figure 4.29(a), the EXIT chart for $E_b/N_0 = 5.4$ dB is depicted. This EXIT chart plots the *bottleneck region*, which corresponds to the BER waterfall region in Figure 4.22. In Figure 4.29(a), the convergence is

almost reached after 10 iterations. The EXIT chart obtained under the same assumptions, for the corresponding binary encoder, is shown in Figure 4.29(b). It is clear that for the binary case decoder, the trajectory is better than that of the LCIRC decoder. In fact, the binary decoder needs fewer iterations (only seven) to converge and the EXIT diagram is wider. In Figure 4.29(c), the EXIT chart for $E_b/N_0 = 6$ dB, corresponding to the *wide-open region*, is depicted. In this case, the LCIRC decoder outperforms its binary counterpart, due to the wider opening of the EXIT chart. Moreover, the convergence is almost reached after four iterations. As a conclusion, the LCIRC outperforms the binary case for $E_b/N_0 > 5.8$ dB, while the opposite happens for lower values of E_b/N_0. All these results are consistent with the BER results in Figure 4.22.

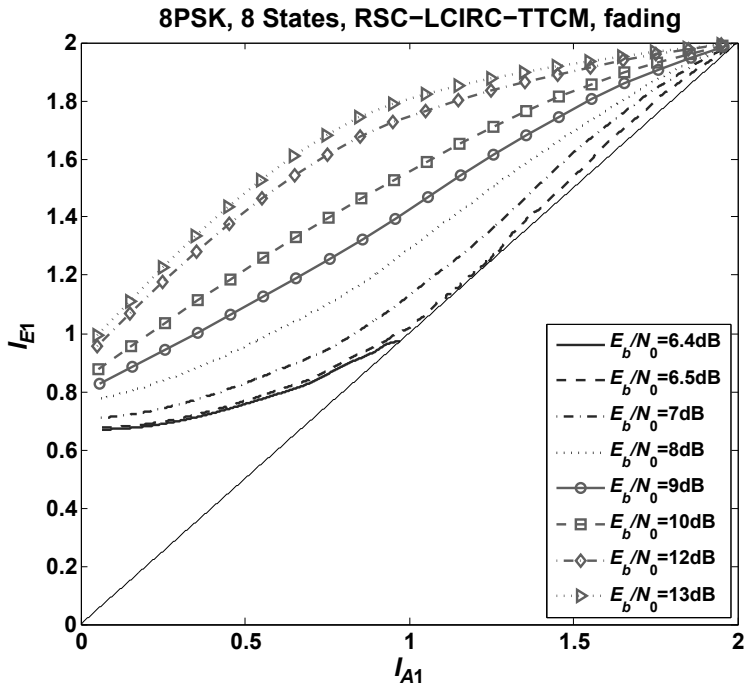
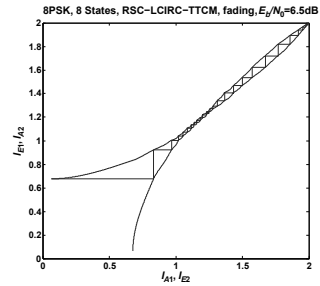
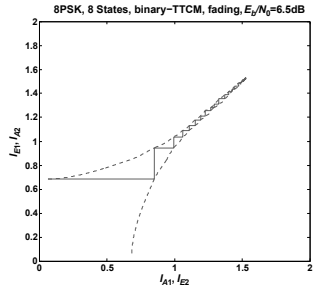


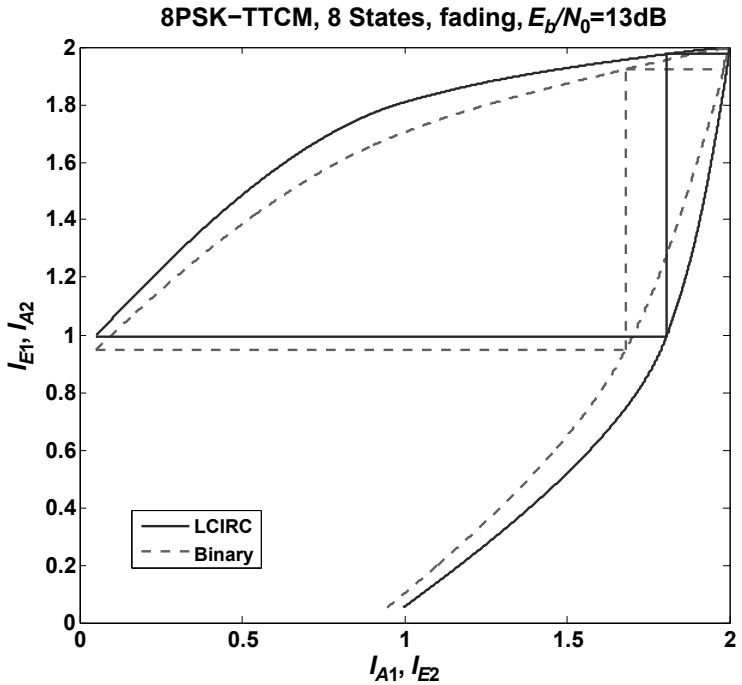
Figure 4.26. *Extrinsic information transfer characteristic of a 8-PSK-RSC-LCIRC-TTCM decoder for a fading channel*



a)



b)



c)

Figure 4.27. *EXIT charts for 8-PSK-TTCM over a fading channel. a) LCIRC,* $E_b/N_0 = 6.5$ *dB; b) binary,* $E_b/N_0 = 6.5$ *dB; c)* $E_b/N_0 = 13$ *dB*

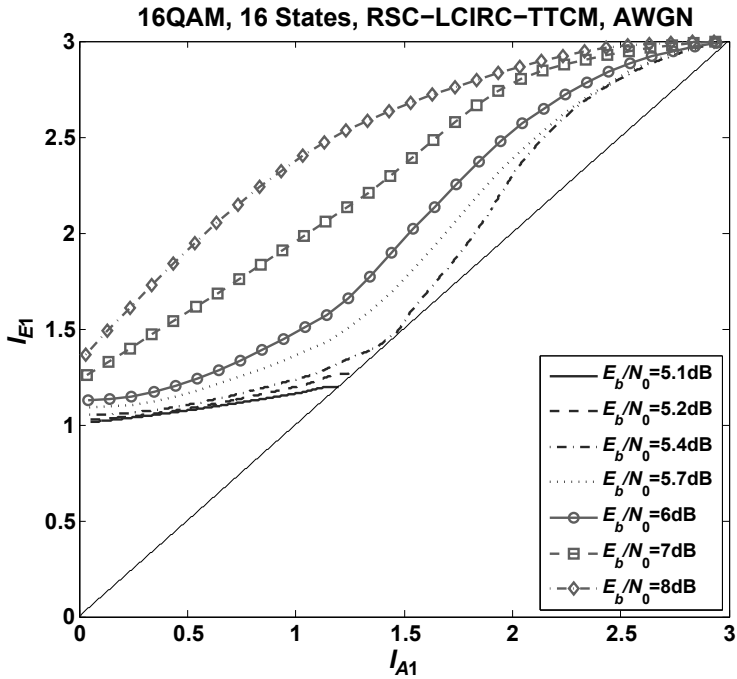
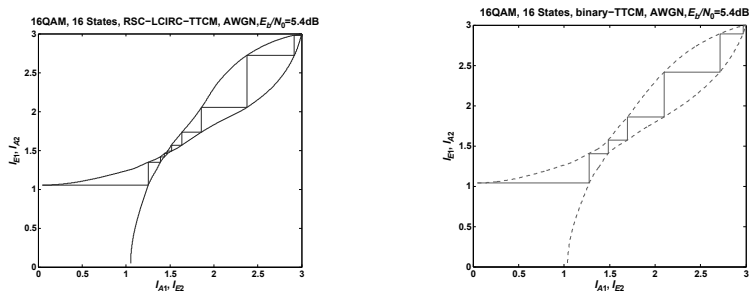
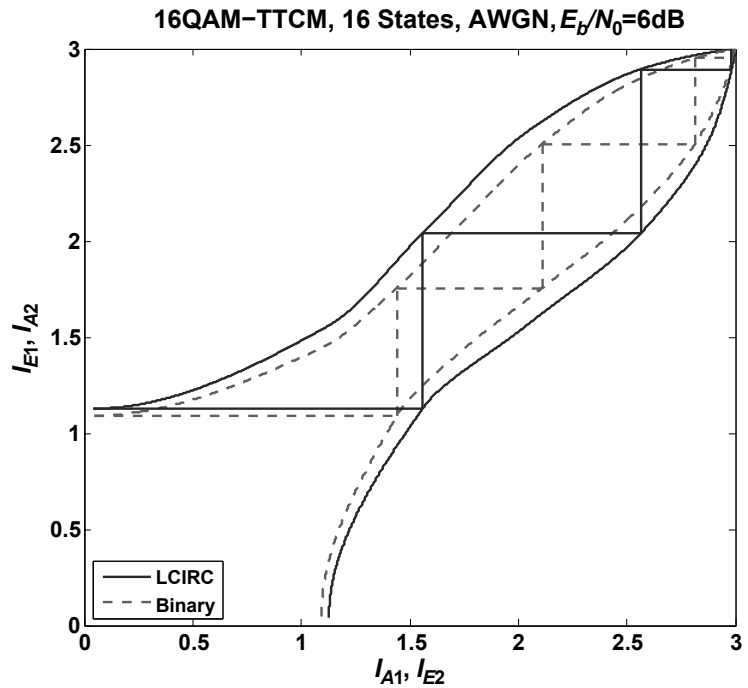


Figure 4.28. *Extrinsic information transfer characteristic of a 16-QAM-RSC-LCIRC-TTCM decoder for a non-fading channel*

For all EXIT charts mentioned above, a high initial value of the *a priori* information is noted for the first decoder. This is explained by the good approximation provided by equation [4.20], in the first iteration initialization.



a) b)



c)

Figure 4.29. *EXIT charts for 16-QAM-TTCM over a non-fading channel. a) LCIRC,* $E_b/N_0 = 5.4$ *dB; b) binary,* $E_b/N_0 = 5.4$ *dB; c)* $E_b/N_0 = 6$ *dB*

Appendix

Demonstrations for the Properties of the Unsigned and Signed Finite Precision Operators

A1.1. The demonstration of theorem 2.2

In section 2.2, the theorem 2.2 was enunciated with [2.19]. Therefore, the addition operation in the 2^N-set is associative, as expressed by:

$$\begin{aligned} w^U &= x^U \oplus y^U \oplus z^U = \left(x^U \oplus y^U\right) \oplus z^U = x^U \oplus \left(y^U \oplus z^U\right) = \\ &= \left(x^U + y^U + z^U\right) \bmod \left(2^N\right) \\ w^U &\in \left[0,\, 2^N - 1\right] \end{aligned} \qquad [A1.1]$$

DEMONSTRATION A1.1.– According to definition 2.6 given in section 2.2, the addition operation for two unsigned numbers, in the 2^N-set, is given by equation [2.13]:

$$v^U = x^U \oplus y^U = \begin{cases} x^U + y^U - 2^N & \text{if} \quad 2^N \leq x^U + y^U \leq 2^{N+1} - 2 \\ x^U + y^U & \text{if} \quad 0 \leq x^U + y^U \leq 2^N - 1 \end{cases} \qquad [A1.2]$$

$$v^U \in \left[0,\, 2^N - 1\right]$$

Let us consider the first term in [A1.1], ${w_1}^U = \left(x^U \oplus y^U\right) \oplus z^U = v^U \oplus z^U$, and using equation [A1.2], it results in the following expression:

$$
\begin{aligned}
&{w_1}^U = v^U \oplus z^U = \left(z^U + v^U\right) \bmod \left(2^N\right) \\
&= \begin{cases} z^U + v^U - 2^N & \text{if} \quad 2^N \le z^U + v^U \le 2^{N+1} - 2 \\ z^U + v^U & \text{if} \quad 0 \le z^U + v^U \le 2^N - 1 \end{cases} \\
&= \begin{cases}
(x^U + \ y^U - 2^N) + z^U - 2^N \text{ if } \left(2^N \le x^U + y^U - 2^N + z^U \le \right. \\
\qquad \left. \le 2^{N+1} - 2\right) \cap \\
\qquad \cap \left[\left(2^N \le x^U + y^U \le 2^{N+1} - 2\right) \cup \right. \\
\qquad \left. \cup \left(0 \le z^U \le 2^N - 1\right)\right] \\
(x^U + \ y^U - 2^N) + z^U \text{ if } \left(0 \le x^U + y^U - 2^N + z^U \le \right. \\
\qquad \left. \le 2^N - 1\right) \cap \\
\qquad \cap \left[\left(2^N \le x^U + y^U \le 2^{N+1} - 2\right) \cup \right. \\
\qquad \left. \cup \left(0 \le z^U \le 2^N - 1\right)\right] \\
(x^U + \ y^U) + z^U - 2^N \text{ if } \left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 2\right) \cap \\
\qquad \cap \left[\left(0 \le x^U + y^U \le 2^N - 1\right) \cup \right. \\
\qquad \left. \cup \left(0 \le z^U \le 2^N - 1\right)\right] \\
(x^U + \ y^U) + z^U \text{ if } \left(0 \le x^U + y^U + z^U \le 2^N - 1\right) \cap \\
\qquad \cap \left[\left(0 \le x^U + y^U \le 2^N - 1\right) \cup \right. \\
\qquad \left. \cup \left(0 \le z^U \le 2^N - 1\right)\right]
\end{cases}
\end{aligned}
\qquad \text{[A1.3]}
$$

which is equivalent to:

$$
{w_1}^U = \left\{ \begin{array}{ll}
x^U + & y^U + z^U \text{ if } \big(0 \le x^U + y^U + z^U \le 2^N - 1\big) \cap \\
& \cap \big(0 \le x^U + y^U + z^U \le 2^{N+1} - 2\big) = \\
& = \big(0 \le x^U + y^U + z^U \le 2^N - 1\big) \\
x^U + & y^U + z^U - 2^N \text{ if } \big[\big(2^N \le x^U + y^U + z^U \le \\
& \le 2^{N+1} - 2\big) \cap \\
& \cap \big(0 \le x^U + y^U + z^U \le 2^{N+1} - 2\big)\big] \cup = \\
& \cup \big[\big(0 \le x^U + y^U + z^U - 2^N \le 2^N - 1\big) \cap \\
& \cap \big(2^N \le x^U + y^U + z^U \le 2^N \cdot 3 - 3\big)\big] = \\
& = \big(2^N \le x^U + y^U + z^U \le 2^{N+1} - 2\big) \cup \\
& \cup \big(2^N \le x^U + y^U + z^U \le 2^{N+1} - 1\big) = \\
& = \big(2^N \le x^U + y^U + z^U \le 2^{N+1} - 1\big) \\
x^U + & y^U + z^U - 2^N \cdot 2 \text{ if } \big(2^N \le x^U + y^U + z^U - 2^N \le \\
& \le 2^{N+1} - 2\big) \cap \\
& \cap \big(2^N \le x^U + y^U + z^U \le 2^{N+1} + 2^N - 1 - 2 = \\
& = 2^N \cdot 3 - 3\big) = \\
& = \big(2^{N+1} \le x^U + y^U + z^U \le 2^N \cdot 3 - 2\big) \cap \\
& \cap \big(2^N \le x^U + y^U + z^U \le 2^N \cdot 3 - 3\big) = \\
& = \big(2^{N+1} \le x^U + y^U + z^U \le 2^N \cdot 3 - 3\big)
\end{array} \right. \qquad \text{[A1.4]}
$$

So, we have:

$$
{w_1}^U = \left\{ \begin{array}{ll}
x^U + & y^U + z^U \\
& \text{if } \ 0 \le x^U + y^U + z^U \le 2^N - 1 \\
x^U + & y^U + z^U - 2^N \\
& \text{if } \ 2^N \le x^U + y^U + z^U \le 2^{N+1} - 1 \\
x^U + & y^U + z^U - 2 \cdot 2^N \\
& \text{if } \ 2 \cdot 2^N \le x^U + y^U + z^U \le 2^N \cdot 3 - 3
\end{array} \right. \qquad \text{[A1.5]}
$$

Now, let us consider the other term in [A1.1], ${w_2}^U = x^U \oplus \left(y^U \oplus z^U\right) = x^U \oplus t^U$, where t^U is determined according to definition 2.6 from section 2.2, for the addition operation of two unsigned numbers, in the 2^N-set, given by [2.13]:

$$t^U = y^U \oplus z^U = \begin{cases} y^U + z^U - 2^N & \text{if} \quad 2^N \le y^U + z^U \le 2^{N+1} - 2 \\ y^U + z^U & \text{if} \quad 0 \le y^U + z^U \le 2^N - 1 \end{cases} \qquad \text{[A1.6]}$$
$$t^U \in \left[0,\, 2^N - 1\right]$$

Using equation [A1.6], we have:

$$\begin{aligned} {w_2}^U &= x^U \oplus t^U = \left(x^U + t^U\right) \bmod \left(2^N\right) \\ &= \begin{cases} x^U + t^U - 2^N & \text{if} \quad 2^N \le x^U + t^U \le 2^{N+1} - 2 \\ x^U + t^U & \text{if} \quad 0 \le x^U + t^U \le 2^N - 1 \end{cases} \\ &= \begin{cases} x^U + & \left(y^U + z^U - 2^N\right) - 2^N \\ & \text{if} \quad \left(2^N \le x^U + y^U + z^U - 2^N \le 2^{N+1} - 2\right) \cap \\ & \cap \left[\left(2^N \le y^U + z^U \le 2^{N+1} - 2\right) \cup \right. \\ & \left. \cup \left(0 \le x^U \le 2^N - 1\right)\right] \\ x^U + & \left(y^U + z^U\right) - 2^N \\ & \text{if} \quad \left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 2\right) \cap \\ & \cap \left[\left(0 \le y^U + z^U \le 2^N - 1\right) \cup \right. \\ & \left. \cup \left(0 \le x^U \le 2^N - 1\right)\right] \\ x^U + & \left(y^U + z^U - 2^N\right) \\ & \text{if} \quad \left(0 \le x^U + y^U + z^U - 2^N \le 2^N - 1\right) \cap \\ & \cap \left[\left(2^N \le y^U + z^U \le 2^{N+1} - 2\right) \cup \right. \\ & \left. \cup \left(0 \le x^U \le 2^N - 1\right)\right] \\ x^U + & \left(y^U + z^U\right) \\ & \text{if} \quad \left(0 \le x^U + y^U + z^U \le 2^N - 1\right) \cap \\ & \cap \left[\left(0 \le y^U + z^U \le 2^N - 1\right) \cup \right. \\ & \left. \left(0 \le x^U \le 2^N - 1\right)\right] \end{cases} \end{aligned} \qquad \text{[A1.7]}$$

which is equivalent to:

$$
w_2{}^U = \begin{cases}
x^U+ \quad y^U + z^U \\
\qquad \text{if} \quad \left(0 \le x^U + y^U + z^U \le 2^N - 1\right) \cap \\
\qquad \cap \left(0 \le x^U + y^U + z^U \le 2^{N+1} - 2\right) = \\
\qquad = \left(0 \le x^U + y^U + z^U \le 2^N - 1\right) \\
x^U+ \quad y^U + z^U - 2^N \\
\qquad \text{if} \quad \left[\left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 2\right) \cap \right. \\
\qquad \left. \cap \left(0 \le x^U + y^U + z^U \le 2^{N+1} - 2\right)\right] \cup = \\
\qquad \cup \left[\left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 1\right) \cap \right. \\
\qquad \left. \cap \left(2^N \le x^U + y^U + z^U \le 2^N \cdot 3 - 3\right)\right] = \\
\qquad = \left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 2\right) \cup \\
\qquad \cup \left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 1\right) = \\
\qquad = \left(2^N \le x^U + y^U + z^U \le 2^{N+1} - 1\right) \\
x^U+ \quad y^U + z^U - 2^N \cdot 2 \\
\qquad \text{if} \quad \left(2^N \le x^U + y^U + z^U - 2^N \le 2^{N+1} - 2\right) \cap \\
\qquad \cap \left(2^N \le x^U + y^U + z^U \le 2^{N+1} + 2^N - 1 - 2 = \right. \\
\qquad \left. = 2^N \cdot 3 - 3\right) = \\
\qquad = \left(2^{N+1} \le x^U + y^U + z^U \le 2^N \cdot 3 - 2\right) \cap \\
\qquad \cap \left(2^N \le x^U + y^U + z^U \le 2^N \cdot 3 - 3\right) = \\
\qquad = \left(2^{N+1} \le x^U + y^U + z^U \le 2^N \cdot 3 - 3\right)
\end{cases}
\qquad \text{[A1.8]}
$$

Therefore, it results in the following expression:

$$
w_2{}^U = \begin{cases}
x^U+ \quad y^U + z^U \\
\qquad \text{if} \quad 0 \le x^U + y^U + z^U \le 2^N - 1 \\
x^U+ \quad y^U + z^U - 2^N \\
\qquad \text{if} \quad 2^N \le x^U + y^U + z^U \le 2^{N+1} - 1 \\
x^U+ \quad y^U + z^U - 2 \cdot 2^N \\
\qquad \text{if} \quad 2 \cdot 2^N \le x^U + y^U + z^U \le 2^N \cdot 3 - 3
\end{cases}
\qquad \text{[A1.9]}
$$

NOTE A1.1.– It is obvious from equations [A1.5] and [A1.9] that ${w_1}^U = {w_2}^U = w^U$. *q.e.d.*

NOTE A1.2.– From equations [A1.5] and [A1.9] we can derive another property for the addition operation; the associativity property can be extended for any number of terms in the sum, not just for three. Section A1.2 refers to this property.

A1.2. The general expression of the addition operation in the 2^N-set

Let us consider m unsigned values ${a_i}^U, i \in [1,\, m], {a_i}^U \in \left[0,\, 2^N - 1\right]$. The induction method is used to prove equation [2.14], where this sum term, ${b_m}^U$, denotes the general term in the induction step m, or the sum of m unsigned terms.

$$
{b_m}^U = {a_1}^U \oplus {a_2}^U \oplus {a_3}^U \oplus \ldots \oplus {a_m}^U = \left(\sum_{i=1}^{m} {a_i}^U\right) \bmod \left(2^N\right) =
$$

$$
\begin{cases}
\sum_{i=1}^{m} {a_i}^U - (m-1)\, 2^N \\
\quad \text{if} \quad (m-1)\, 2^N \leq \sum_{i=1}^{m} {a_i}^U \leq m2^N - m \\
\sum_{i=1}^{m} {a_i}^U - (m-2)\, 2^N \\
\quad \text{if} \quad (m-2)\, 2^N \leq \sum_{i=1}^{m} {a_i}^U \leq (m-1)\, 2^N - 1 \\
\sum_{i=1}^{m} {a_i}^U - (m-3)\, 2^N \\
\quad \text{if} \quad (m-3)\, 2^N \leq \sum_{i=1}^{m} {a_i}^U \leq (m-2)\, 2^N - 1 \\
\ldots\ldots\ldots\ldots\ldots \\
\sum_{i=1}^{m} {a_i}^U \quad \text{if} \quad 0 \leq \sum_{i=1}^{m} {a_i}^U \leq 2^N - 1
\end{cases}
\qquad \text{[A1.10]}
$$

$$
{b_m}^U, {a_i}^U \in \left[0,\, 2^N - 1\right]
$$

Step 1: according to [2.13] and [A1.2], the sum of two unsigned numbers, in the 2^N-set, is given by:

$$\begin{aligned} & {b_2}^U = {a_1}^U \oplus {a_2}^U = \\ & = \begin{cases} {a_1}^U + {a_2}^U - 2^N & \text{if} \quad 2^N \le {a_1}^U + {a_2}^U \le 2^{N+1} - 2 \\ {a_1}^U + {a_2}^U & \text{if} \quad 0 \le {a_1}^U + {a_2}^U \le 2^N - 1 \end{cases} \\ & {b_3}^U, {a_1}^U, {a_2}^U \in \left[0,\, 2^N - 1\right] \end{aligned} \qquad [A1.11]$$

Step 2: according to [2.19], [A1.1], [A1.5], and [A1.9], the sum of three unsigned numbers, in the 2^N-set, is given by:

$$\begin{aligned} & {b_3}^U = \begin{cases} {a_1}^U + \quad {a_2}^U + {a_3}^U \\ \qquad\quad \text{if} \quad 0 \le {a_1}^U + {a_2}^U + {a_3}^U \le 2^N - 1 \\ {a_1}^U + \quad {a_2}^U + {a_3}^U - 2^N \\ \qquad\quad \text{if} \quad 2^N \le {a_1}^U + {a_2}^U + {a_3}^U \le 2^{N+1} - 1 \\ {a_1}^U + \quad {a_2}^U + {a_3}^U - 2 \cdot 2^N \\ \qquad\quad \text{if} \quad 2 \cdot 2^N \le {a_1}^U + {a_2}^U + {a_3}^U \le 2^N \cdot 3 - 3 \end{cases} \\ & {b_3}^U, {a_1}^U, {a_2}^U, {a_3}^U \in \left[0,\, 2^N - 1\right] \end{aligned} \qquad [A1.12]$$

Step 3: now, we have to prove that the $(m+1)$-order sum term, ${b_{m+1}}^U$, is derived from the m-order sum term, ${b_m}^U$, as

$$\begin{aligned} & {b_{m+1}}^U = {b_m}^U \oplus {a_{m+1}}^U \\ & {b_m}^U, {b_{m+1}}^U, {a_{m+1}}^U \in \left[0,\, 2^N - 1\right] \end{aligned} \qquad [A1.13]$$

where the term ${b_m}^U$ is given by equation [A1.10].

Then, the term in [A1.13] can be expressed as:

$$
\begin{aligned}
&{b_{m+1}}^U = \\
&= {b_m}^U \oplus {a_{m+1}}^U = \left({a_1}^U \oplus {a_2}^U \oplus {a_3}^U \oplus \ldots \oplus {a_m}^U\right) \oplus {a_{m+1}}^U = \\
&= \left[\left(\sum_{i=1}^{m} {a_i}^U\right) \bmod \left(2^N\right) + {a_{m+1}}^U\right] \bmod \left(2^N\right) \\
&{b_m}^U, {b_{m+1}}^U, {a_i}^U \in \left[0,\, 2^N - 1\right]
\end{aligned}
\qquad \text{[A1.14]}
$$

Using the same principle as in section A1.1, for demonstrating equations [A1.5] and [A1.9], from [A1.4] and [A1.8], to intersect and reunite definition intervals, we can easily obtain:

$$
\begin{aligned}
&{b_{m+1}}^U = {a_1}^U \oplus {a_2}^U \oplus {a_3}^U \oplus \ldots \oplus {a_{m+1}}^U = \\
&= \left(\sum_{i=1}^{m+1} {a_i}^U\right) \bmod \left(2^N\right) = \\
&\begin{cases}
\sum_{i=1}^{m+1} {a_i}^U - (m)\, 2^N \\
\quad \text{if} \quad (m)\, 2^N \le \sum_{i=1}^{m+1} {a_i}^U \le (m+1)\, 2^N - (m+1) \\
\sum_{i=1}^{m+1} {a_i}^U - (m-1)\, 2^N \\
\quad \text{if} \quad (m-1)\, 2^N \le \sum_{i=1}^{m+1} {a_i}^U \le m 2^N - 1 \\
\sum_{i=1}^{m+1} {a_i}^U - (m-2)\, 2^N \\
\quad \text{if} \quad (m-2)\, 2^N \le \sum_{i=1}^{m+1} {a_i}^U \le (m-1)\, 2^N - 1 \\
\ldots\ldots\ldots\ldots\ldots \\
\sum_{i=1}^{m+1} {a_i}^U \quad \text{if} \quad 0 \le \sum_{i=1}^{m+1} {a_i}^U \le 2^N - 1
\end{cases} \\
&{b_{m+1}}^U, {a_i}^U \in \left[0,\, 2^N - 1\right]
\end{aligned}
\qquad \text{[A1.15]}
$$

A1.3. The subtraction as the inverse of addition in the 2^N-set

In section 2.2, theorem 2.4 was enunciated. According to this theorem, the subtraction operator of unsigned numbers in the 2^N-set is the inverse of the addition operator of unsigned numbers in the 2^N-set. In another words, if $z^U = x^U \oplus y^U$, then $x^U = z^U \ominus y^U$ and vice versa.

DEMONSTRATION A1.2.– *Step 1*: the demonstration of the direct implication, $z^U = x^U \oplus y^U \Rightarrow x^U = z^U \ominus y^U$. According to definition 2.6 given in section 2.2, the addition operation for two unsigned numbers, in the 2^N-set, is given by equation [2.13] (also see equation [A1.2]):

$$z^U = x^U \oplus y^U = \begin{cases} x^U + y^U - 2^N & \text{if} \quad 2^N \leq x^U + y^U \leq 2^{N+1} - 2 \\ x^U + y^U & \text{if} \quad 0 \leq x^U + y^U \leq 2^N - 1 \end{cases} \qquad \text{[A1.16]}$$
$$z^U \in \left[0,\, 2^N - 1\right]$$

On the other hand, according to the first definition interval in [A1.16], we have:

$$\begin{aligned} &z^U = x^U + y^U - 2^N \quad \text{if} \quad 2^N \leq x^U + y^U \leq 2^{N+1} - 2 \\ &z^U \in \left[0,\, 2^N - 1\right] \end{aligned} \qquad \text{[A1.17]}$$

So, in this case, expression [A1.17] can be rewritten as:

$$\begin{aligned} &x^U = z^U - y^U + 2^N \\ \text{if} \quad &\left(2^N \leq z^U + 2^N = x^U + y^U \leq 2^{N+1} - 2\right) = \\ &= \left(0 \leq z^U \leq 2^N - 2\right) \end{aligned} \qquad \text{[A1.18]}$$

But $0 \leq y^U \leq 2^N - 1$, and considering this condition in [A1.18] we obtain the following result:

$$-2^N + 1 \leq z^U - y^U \leq 2^N - 2 \qquad \text{[A1.19]}$$

We also have the unsigned number x^U in [A1.18], and then the next condition is fulfilled:

$$0 \leq x^U = z^U - y^U + 2^N \leq 2^N - 1 \Rightarrow -2^N \leq z^U - y^U \leq -1 \qquad \text{[A1.20]}$$

Now, considering both conditions for the difference value, from [A1.19] and [A1.20] the following intersection interval results:

$$\begin{aligned} x^U \quad &= \quad z^U - y^U + 2^N \\ &\text{if} \quad \left(-2^N + 1 \leq z^U - y^U \leq 2^N - 2\right) \cap \\ &\cap \left(-2^N \leq z^U - y^U \leq -1\right) = \left(-2^N + 1 \leq z^U - y^U \leq -1\right) \end{aligned} \qquad \text{[A1.21]}$$

On the other hand, according to the second definition interval in [A1.16], we have:

$$\begin{aligned} &z^U = x^U + y^U \quad \text{if} \quad 0 \leq x^U + y^U \leq 2^N - 1 \\ &z^U \in \left[0,\, 2^N - 1\right] \end{aligned} \qquad \text{[A1.22]}$$

So, in this case, expression [A1.22] can be rewritten as

$$\begin{aligned} x^U = z^U - y^U \quad \text{if} \quad &\left(0 \leq z^U = x^U + y^U \leq 2^N - 1\right) = \\ &= \left(0 \leq z^U \leq 2^N - 1\right) \end{aligned} \qquad \text{[A1.23]}$$

But $0 \leq y^U \leq 2^N - 1$, and considering this condition in [A1.23] we obtain the following result:

$$-2^N + 1 \leq z^U - y^U \leq 2^N - 1 \qquad \text{[A1.24]}$$

We also have the unsigned number x^U in [A1.23], and then the next condition is fulfilled:

$$0 \leq x^U = z^U - y^U \leq 2^N - 1 \Rightarrow 0 \leq z^U - y^U \leq 2^N - 1 \qquad \text{[A1.25]}$$

Now, considering both conditions for the difference value, from [A1.24] and [A1.25] the following intersection interval results:

$$\begin{aligned} x^U &= z^U - y^U \\ &\text{if} \quad \left(-2^N + 1 \leq z^U - y^U \leq 2^N - 1\right) \cap \\ &\cap \left(0 \leq z^U - y^U \leq 2^N - 1\right) = \left(0 \leq z^U - y^U \leq 2^N - 1\right) \end{aligned} \qquad \text{[A1.26]}$$

CONCLUSION A1.1.– From equations [A1.21] and [A1.26] the following compact expression for the $\ominus$ operator is obtained:

$$x^U = z^U \ominus y^U = \begin{cases} z^U - y^U + 2^N & \text{if} \quad -2^N + 1 \leq z^U - y^U \leq -1 \\ z^U - y^U & \text{if} \quad 0 \leq z^U - y^U \leq 2^N - 1 \end{cases} \qquad \text{[A1.27]}$$

$$x^U, y^U, z^U \in \left[0,\, 2^N - 1\right]$$

q.e.d.

Step 2: the demonstration of the reverse implication, $x^U = z^U \ominus y^U \Rightarrow z^U = x^U \oplus y^U$, is obviously similar to the direct implication presented in the previous step.

Bibliography

[AIS 96] AISLAM T., EDWARDS J., "Secure communications using chaotic digital encoding", *Electronics Letters*, vol. 32, no. 3, pp. 190–191, 1996.

[BAR 00] BARBULESCU S., GUIDI A., PIETROBON S., "Chaotic turbo codes", *Proceedings of the IEEE International Symposium on Information Theory*, Sorrento, Italy, p. 123, June 2000.

[BER 96] BERROU C., GLAVIEUX A., "Near optimum error correcting coding and decoding: turbo–codes", *IEEE Transactions on Communications*, vol. 44, no. 10, pp. 1261–1271, 1996.

[CHE 04] CHEN H., HAIMOVICH A., "EXIT charts for turbo trellis-coded modulation", *IEEE Communications Letters,*, vol. 8, no. 11, pp. 668–670, 2004.

[CHU 88] CHUA L., LIN T., "Chaos in digital filters", *IEEE Transactions on Circuits and Systems*, vol. 35, no. 6, pp. 648–658, 1988.

[CHU 90] CHUA L.O., LIN T., "Chaos and fractals from third-order digital filters", *International Journal of Circuit Theory and Applications*, vol. 18, no. 3, pp. 241–255, 1990.

[CLE 06] CLEVOM T., SCHOTSCH B., SCHMALEN L., *et al.*, "Separation of recursive convolutional codes into sub-codes using Galois field arithmetic", *IEEE 40th Annual Conference on Information Sciences and Systems*, Princeton, NJ, pp. 245–245, 22–24 March 2006.

[CUO 93] CUOMO K.M., OPPENHEIM A.V., "Chaotic signals and systems for communications", *IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP' 93)*, vol. 3, Minneapolis, MN, pp. 137–140, 1993.

[DE 95] DE ANGELI A., GENESIO R., TESI A., "Dead-beat chaos synchronization in discrete time systems", *IEEE Transactions on Circuits and Systems I; Fundamental Theory and Applications*, vol. 42, no. 1, pp. 54–56, 1995.

[DIN 98] DINAN E., JABBARI B., "Spreading codes for direct sequence CDMA and wideband CDMA cellular networks", *IEEE Communications Magazine*, vol. 36, no. 9, pp. 48–54, 1998.

[EBE 69] EBERT P.M., MAZO J.E., TAYLOR M.G., "Overflow oscillations in digital filters", *Bell System Technical Journal*, vol. 48, no. 9, pp. 2999–3020, 1969.

[ELA 06] EL ASSAD S., VLĂDEANU C., "Digital chaotic codec for DS-CDMA communications systems", *Lebanese Science Journal*, vol. 7, no. 2, pp. 55–71, 2006.

[ESC 09] ESCRIBANO F., KOZIC S., LOPEZ L., *et al.*, "Turbo-like structures for chaos encoding and decoding", *IEEE Transactions on Communications*, vol. 57, no. 3, pp. 597–601, 2009.

[FEE 00] FEELY O., "Nonlinear dynamics of discrete-time electronic systems", *IEEE Circuits and Systems Society Newsletter*, vol. 11, no. 1, pp. 1–12, 2000.

[FRA 88] FRANKLIN G.F., POWELL J.D., EMAMI-NAEINI A., *Feedback Control of Dynamic Systems*, Addison-Wesley, Reading, MA, 1988.

[FRE 93] FREY D., "Chaotic digital encoding: an approach to secure communication", *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 40, no. 10, pp. 660–666, 1993.

[HAL 93] HALLE K.S., WU C.W., ITOH M., *et al.*, "Spread spectrum communication through modulation of chaos", *International Journal of Bifurcation and Chaos*, vol. 3, no. 2, pp. 469–477, 1993.

[HAN 02] HANZO L., LIEW T., YEAP B., *Turbo Coding, Turbo Equalisation, and Space-Time Coding for Transmission over Fading Channels*, John Wiley & Sons, 2002.

[KAC 92] KACZMARCZYK F., Nonlinear Codec in the Digital Domain, Master Thesis, Lehigh University, Bethlehem, PA, 1992.

[KLI 06] KLIEWER J., NG S.X., HANZO L., "Efficient computation of EXIT functions for nonbinary iterative decoding", *IEEE Transactions on Communications*, vol. 54, no. 12, pp. 2133–2136, 2006.

[KOC 92] KOCAREV L., HALLE K.S., ECKERT K., *et al.*, "Experimental demonstration of secure communications via chaotic synchronization", *International Journal of Bifurcation and Chaos*, vol. 2, no. 3, pp. 709–713, 1992.

[LIN 91] LIN T., CHUA L., "On chaos of digital filters in the real world", *IEEE Transactions on Circuits and Systems*, vol. 38, no. 5, pp. 557–558, 1991.

[MAZ 97] MAZZINI G., SETTI G., ROVATTI R., "Chaotic complex spreading sequences for asynchronous DS-CDMA – part I: system modeling and results", *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 10, pp. 937–947, 1997.

[NG 08] NG S.X., ALAMRI O., LI Y., *et al.*, "Near-capacity turbo trellis coded modulation design based on EXIT charts and union bounds", *IEEE Transactions on Communications*, vol. 56, no. 12, pp. 2030–2039, 2008.

[OGI 01] OGIWARA H., MIZUTOME A., KOIKE K., "Performance evaluation of parallel concatenated trellis-coded modulation", *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, vol. E84-A, no. 10, pp. 2410–2417, 2001.

[PAR 92] PARLITZ U., CHUA L.O., KOCAREV L., *et al.*, "Transmission of digital signals by chaotic synchronization", *International Journal of Bifurcation and Chaos*, vol. 2, no. 4, pp. 973–977, 1992.

[PAU 10a] PAUN A., VLĂDEANU C., MARGHESCU I., *et al.*, "New recursive convolutional GF(2^N) encoders for parallel turbo-TCM schemes", *6th Advanced International Conference on Telecommunications (AICT)*, Barcelona, Spain, pp. 182–186, 9–15 May 2010.

[PAU 10b] PAUN A., VLĂDEANU C., MARGHESCU I., *et al.*, "On the QAM parallel turbo-TCM schemes using recursive convolutional GF(2^N) encoders", *Proceedings of the 18th European Signal Conference (EUSIPCO'10)*, Aalborg, Denmark, pp. 1414–1418, 23–27 August 2010.

[PEC 90] PECORA L.M., CARROLL T.L., "Synchronization in chaotic systems", *Physical Review Letters*, vol. 64, no. 2, pp. 821–824, 1990.

[PEN 01] PENAUD S., Etude des potentialités du chaos pour les systèmes de télécommunications. Evaluation des performances de systèmes à accès multiples à répartition par les codes (CDMA) utilisant des séquences d'étalement chaotiques, Doctorate Thesis, University of Limoges, France, 2001.

[PUR 77a] PURSLEY M., "Performance evaluation for phase-coded spread-spectrum multiple-access communication – part I: system analysis", *IEEE Transactions on Communications*, vol. 25, no. 8, pp. 795–799, 1977.

[PUR 77b] PURSLEY M., SARWATE D., "Performance evaluation for phase-coded spread-spectrum multiple-access communication – part II: code sequence analysis", *IEEE Transactions on Communications*, vol. 25, no. 8, pp. 800–803, 1977.

[RAP 96] RAPPAPORT T.S., *Wireless Communications – Principles and Practice*, Prentice-Hall, 1996.

[ROB 98] ROBERTSON P., WORZ T., "Bandwidth-efficient turbo trellis-coded modulation using punctured component codes", *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 206–218, 1998.

[ROV 00] ROVATTI R., MAZZINI G., SETTI G., "A tensor approach to higher order expectations of quantized chaotic trajectories – part I: general theory and specialization to piecewise-affine markov maps", *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 11, pp. 1571–1583, 2000.

[SCH 89] SCHLEGEL C., COSTELLO D., "Bandwidth efficient coding for fading channels: code construction and performance analysis", *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 9, pp. 1356–1368, 1989.

[SCH 03] SCHRECKENBACH F., GORTZ N., HAGENAUER J., *et al.*, "Optimized symbol mappings for bit-interleaved coded modulation with iterative decoding", *IEEE Global Telecommunications Conference (GLOBECOM '03)*, vol. 6, San Francisco, pp. 3316–3320, 1–5 December 2003.

[TEN 01] TEN BRINK S., "Convergence behavior of iteratively decoded parallel concatenated codes", *IEEE Transactions on Communications*, vol. 49, no. 10, pp. 1727–1737, 2001.

[TES 94] TESI A., DE ANGELI A., GENESIO R., "On the system decomposition for synchronizing chaos", *International Journal of Bifurcation and Chaos*, vol. 4, no. 6, pp. 1675–1685, 1994.

[UNG 82] UNGERBOECK G., "Channel coding with multilevel/phase signals", *IEEE Transactions on Information Theory*, vol. IT-28, no. 1, pp. 55–67, 1982.

[VLA 09a] VLĂDEANU C., EL ASSAD S., CARLACH J.-C., *et al.*, "Improved Frey chaotic digital encoder for trellis-coded modulation", *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, no. 6, pp. 509–513, 2009.

[VLA 09b] VLĂDEANU C., EL ASSAD S., CARLACH J.-C., *et al.*, "Optimum PAM-TCM schemes using left-circulate function over GF(2^N)", *International Symposium on Signals, Circuits and Systems (ISSCS 09)*, Iasi, Romania, pp. 267–270, 9–10 July 2009.

[VLA 09c] VLĂDEANU C., EL ASSAD S., CARLACH J.-C., *et al.*, "Chaotic digital encoding for 2D trellis-coded modulation", *5th Advanced International Conference on Telecommunications (AICT' 09)*, Venice, Italy, pp. 152–157, 24–28 May 2009.

[VLA 10a] VLĂDEANU C., PAUN A., LUCACIU R., *et al.*, "Parallel turbo-TCM schemes using recursive convolutional GF(2^N) encoders over frequency non-selective fading channel", *9th International Symposium on Electronics and Telecommunications (ISETC '10)*, Timisoara, Romania, pp. 285–288, 11–12 November 2010.

[VLA 10b] VLĂDEANU C., EL ASSAD S., CARLACH J.-C., *et al.*, "Recursive GF(2^N) encoders using left-circulate function for optimum PSK-TCM schemes", *Signal Processing*, vol. 90, no. 9, pp. 2708–2713, 2010.

[VLA 11a] VLĂDEANU C., EL ASSAD S., "Designing optimum 2D-TCM schemes using new systematic convolutional encoders over GF(2^N)", *10th International Symposium on Signals, Circuits and Systems (ISSCS '11)*, Iasi, Romania, pp. 479–483, 30 June–1 July 2011.

[VLA 11b] VLĂDEANU C., EL ASSAD S., "Optimum QAM-TCM schemes using left-circulate function over GF(2^N)", *IEEE 7th Advanced International Conference on Telecommunication (AICT '11)*, St. Maarten, the Netherlands, pp. 112–116, 20–25 March 2011.

[VLA 11c] VLĂDEANU C., EL ASSAD S., "Punctured 8-PSK turbo-TCM transmissions using recursive systematic convolutional GF(2^N) encoders", *Proceedings of the 19th European Signal Conference (EUSIPCO '11)*, Barcelona, Spain, pp. 111–115, 29 August–2 September 2011.

[VLA 12] VLĂDEANU C., MARTIAN A., EL ASSAD S., "EXIT Charts analysis for turbo-TCM schemes using non-binary RSC encoders", *IEEE 8th Advanced International Conference on Telecommunication (AICT '12)*, Stuttgart, Germany, pp. 150–155, 27 May–1 June 2012.

[VUC 00] VUCETIC B., YUAN J., *Turbo Codes: Principles and Applications*, Springer, 2000.

[WER 98] WERTER M., "An improved chaotic digital encoder", *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 2, pp. 227–229, 1998.

[WU 93] WU C.W., CHUA L.O., "A simple way to synchronize chaotic systems with application to secure communications systems", *International Journal of Bifurcation and Chaos*, vol. 3, no. 6, pp. 1619–1627, 1993.

[ZHO 01] ZHOU X., LIU J., SONG W., *et al.*, "Chaotic turbo codes in secure communication", *International Conference on Trends in Communications (EUROCON '01)*, vol. 1, Bratislava, Slovakia, pp. 199–201, 5–7 July 2001.

Index