

Error-rate in Viterbi decoding of a duobinary signal in presence of noise and distortions: theory and simulation

Henri Mertens and Marc Van Droogenbroeck

Montefiore Institute, University of Liège, Belgium

Abstract. The Viterbi algorithm, presented in 1967, allows a maximum likelihood decoding of partial response codes. This study focuses on the duobinary code which is the first member of this family and has been specified for the digital part of television systems recommended by International Organizations. Up to now the error-rate, which is the main criterion of the performance, has been evaluated by simulation. Although there exist theoretical bounds, these bounds are not satisfactory for a channel such as broadcasting (by terrestrial transmitters, cable networks or satellite) which is strongly impaired by noise, and linear and non-linear distortions. Analytical methods, verified by simulation, are presented here in order to evaluate the theoretical and exact values of the error-rate, in the form of series of numerical integrations, for a transmission in baseband or in radio-frequency with quadriphase modulation (or AM/VSB for cable networks) and coherent demodulation, in presence of noise and several distortions. This methodology can be later extended to other partial response codes, to convolutional codes and their concatenations.

Keywords: Viterbi decoding, bit-error rate, duobinary code, error-rate, convolutional code, partial response signaling, digital modulation, interference, noise, synchronization error, non-linear distortion, echo

Contents

1	Introduction	5
2	Brief review of Viterbi's decoding algorithm	6
3	Duobinary code and partial response codes	8
3.1	General presentation	8
3.2	Duobinary code	8
3.3	Other partial response codes	9
3.4	Implementation of the Viterbi decoder	10
3.5	Some properties of the duobinary code	11
3.5.1	Duobinary eye diagram	11
3.5.2	Correlation of the duobinary signal	13
3.5.3	Probability of the duobinary symbols	15
4	Theoretical framework for duobinary coding and the Viterbi decoding	15
4.1	Formulation of the tests made by the Viterbi decoder	15
4.2	Alternative formulation of the tests	16
4.3	Duobinary error probability with low noise	17
4.4	Probability of a recognized sequence r for a transmitted sequence d	18
4.5	Duobinary error-rate	19
4.6	Error-rate on the survivors	21
4.7	Evolution of the errors on the survivors	22
4.7.1	Normal rules	24
4.7.2	Additional rules	24
4.8	Binary error-rate without precoding	26
4.9	Binary error-rate with precoding	27
4.10	Upper bound of the binary error-rate	28
4.11	Error statistics	32
5	Interferences	32
5.1	Correlated or uncorrelated interference	33
5.2	General theoretical method in absence of noise	34
5.3	General theoretical method in presence of interference and noise	35

5.4	Interference resulting from a phase error φ in a CQPRS demodulation . . .	36
5.4.1	Phase error without noise	36
5.4.2	Phase error with noise	38
5.5	Interference resulting from a synchronization error τ	38
5.5.1	Synchronization error without noise	38
5.5.2	Synchronization error with noise	41
5.6	Interference resulting from an echo in CQPRS	41
5.6.1	Echo without noise	41
5.6.2	Echo with noise	44
5.7	Interference resulting from an echo in AM/SSB	47
5.7.1	Echo without noise	47
5.7.2	Echo with noise	48
5.8	Example of a non-linear distortion	48
5.8.1	General	48
5.8.2	Non-linearity model	51
5.8.3	Conventional definition of the back-off	52
5.8.4	Note on band-pass filtering	53
5.8.5	Method of calculation	53
5.9	CQPRS modulation with Viterbi decoding, no precoding and unmatched filtering	54
5.10	COQPRS modulation with Viterbi decoding, no precoding and unmatched filtering	56
6	Conclusions	56
	List of acronyms	57

Notations

Main notations

The following main notations are used in this paper:

a_i	input binary bits -1 or +1
b_i	precoded bits -1 or +1
D	delay operator of one bit
d_i	duobinary transmitted symbols -1, 0 or +1
r_i	duobinary symbols recognized (correctly or not) by the decoder
n_i	noise samples
S_1, S_2	survivors corresponding to the states -1 and +1
L	length of a duobinary sequence
f_b, T	bit frequency and bit period
x_i or y_i	input level at the decoder
$x(t)$	input signal at the decoder
$y(t)$	interference at input of decoder
φ	phase error
τ	synchronization error or echo delay
β	echo relative amplitude
ψ	echo phase

Some special notations

\lesseqgtr	symbol for the following test: “<” OR “>”
$a_1(t), a_2(t)$	data signals modulating the two carriers in quadrature
$r_1(t), r_2(t)$	data signals after demodulation, at the decoder input
$x(t) = x_r(t) + j x_i(t)$	modulated signal <i>before</i> distortion
$z(t) = z_r(t) + j z_i(t)$	modulated signal <i>after</i> distortion
$\rho(t)$ and $\rho_d(t)$	envelopes of $x(t)$ and $z(t)$
$\theta(t)$ and $\theta(t) + \varphi(t)$	phases of $x(t)$ and $z(t)$
A_c	amplitude of the carrier
ω_c	carrier angular frequency
A_s	saturation level of the TWT
B_{off}	backoff (see the conventional definition)
C	carrier power in absence of non-linear distortion
N_0	one-sided power spectral density of the noise in RF
σ^2	noise variance at the decoder

1 Introduction

While already proposed in 1967, the rules of the Viterbi algorithm [24] are still used in a number of digital communication and broadcasting systems. Applications in these fields, as well as other applications such as magnetic recording, and even the more recent “turbo-codes” [2] include rules of the Viterbi algorithm for several forms of RF multiplexing, such as Orthogonal Frequency Division Multiplexing (OFDM) and several forms of modulation (QPSK, QPRS, 16 or 64-QAM, ...).

In communication systems, the Viterbi algorithm is commonly used for decoding partial response codes and convolutional codes and provides a large improvement over classical threshold decoding. The final error-rate for these systems are often obtained only by simulations.

The purpose of this paper is to present and to corroborate, by comparison with simulation, a methodology to derive the theoretical and exact value of the error-rate in the form of an analytical expression, in the case of a channel with strong impairments due to additive Gaussian noise but also due to important forms of linear and non-linear distortions. Such a channel is typically a broadcasting channel with data transmission by terrestrial transmitters, by cable networks or by satellite. For example severe causes of impairments are echoes with long delay and multipath propagation, echoes with short delay, and non-linear distortions in the power stage amplifier, respectively in terrestrial broadcasting, in cable networks, and in satellite transmission.

The paper is deliberately restricted to the study of the duobinary code which is the first member of the family of partial response codes and mainly because it has been already specified for the digital part of television systems recommended for satellite broadcasting by the EBU, CCIR and ITU (Mac/packet systems) [4, 5, 6, 8, 11, 12, 18]. However the presented methodology can be later extended to other partial response codes, to convolutional codes, and even to their concatenations.

This paper focuses on the evaluation of the final bit error-rate (BER) which, in data transmission, is the main criterion for the system performance. In the type of channel considered, it is generally assumed that the final quality should still be satisfactory with a BER of the order of 10^{-3} or even more, account being taken that, for example in teletext, the most sensitive bytes are again protected by an appropriate error-correcting code. This level of BER is unacceptable in applications such as magnetic recording where the BER should be of the order of 10^{-6} or even less.

Theoretical upper bounds of the BER with Viterbi decoding have already been given in the literature. In the case of decoding convolutional codes such bounds are given for example in [24] and [20, pp. 462-470]. For the case of decoding partial-response codes, similar bounds are described in [9, 13, 23], and by a different approach in [1]. The upper bound given in these two last references for the duobinary code will be examined in more details in Section 4.10. The same kind of bound for another system was developed by Tjhung *et al.* [21]; the system studied was narrow band FM where the binary input is first coded in

duobinary, then FM modulated, detected by discriminator and finally duobinary decoded by the Viterbi algorithm (note that for this system, the FM noise, including the “click” noise is no longer Gaussian and the Viterbi algorithm is no more optimum).

But, while all these bounds give a good approximation of the actual BER at high values of the signal to noise ratio (*i.e.* for very low BER), they depart from the exact values when the BER is low, or in other words in presence of strong noise or interference which is our main concern.

The paper is organized as follows. Section 2 briefly describes the key elements and terminology of the Viterbi algorithm. Then, we present the duobinary coding and propose a theoretical framework capable to provide performance figures of a baseband transmission system that combines a duobinary coding and the Viterbi decoding algorithm, respectively in Section 3 and Section 4. Finally we analyze the impact of several types of interferences or distortions with quadriphase modulation and coherent demodulation in Section 5, namely a phase error in coherent demodulation, a synchronization error, an echo in radio-frequency (with CQPRS or AM/SSB modulation taken as an approximation of AM/VSB), and the non-linear distortion in a model of a traveling-wave tube frequently used as the satellite power amplifier.

2 Brief review of Viterbi's decoding algorithm

The algorithm proposed by Viterbi in 1967 [24] is a rather general algorithm permitting to solve, with a limited number of calculations, a number of problems for searching the shortest path in a graph called a *trellis*.

Consider a linear system which can take m different *states* at successive instants $1, 2, \dots, p-1, p$. These states are the *nodes* of the trellis. The evolution of the system is represented by a trellis with m nodes. An example of the path followed in a trellis for the simple case $m = 2$, which corresponds to the duobinary code, is shown in Figure 1; the two possible states are noted s_1 and s_2 .

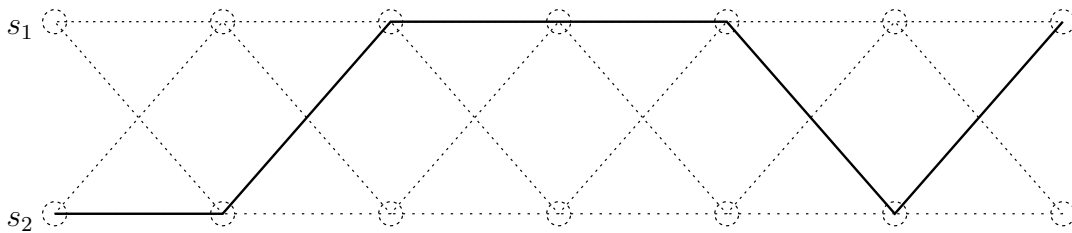


Fig. 1: Example of a path followed, from left to right, in a two nodes trellis. The dotted lines correspond to other possible state transitions.

The transition from a node of order $p - 1$ to a node of order p is a *branch* of the trellis. In

order to evaluate the length of the path followed in the trellis, it is necessary to define a *metric*, *i.e.* a rule giving the length of any branch.

Let us consider the m possible states at instant $p - 1$. To each of these states, there is one and only one path having the shortest metric from the origin. These m shortest paths are called the *survivors* of order $p - 1$. After this step, there are m^2 possible state-transitions from order $p - 1$ to order p , since from each node of order $p - 1$ there are m branches going to the m nodes of order p .

When the algorithm is applicable, the two essential following results have been demonstrated:

1. The shortest path from the origin to a node of order p completely contains the survivor of order $p - 1$ of this path, the metrics being additive, which is the case when the noise samples added to the signal are statistically independent. Note that this assumption is true for additive white Gaussian noise (AWGN).
2. All the survivors are converging to each other in the sense that the probability of non-convergence has the limit zero after an infinite number of steps; in practice however the convergence is obtained after a finite number of state-transitions called the *decoding constraint length* L_C which for most systems is typically 20 to 30 bits. In addition, in absence of impairment, the survivors converge to the system states.

Therefore, the calculation of the shortest path from the origin up to any node is made by recurrence. At order $p - 1$, the decoder keeps in memory the m survivors arriving at the nodes $p - 1$. It then computes the lengths of the m^2 transitions from order $p - 1$ to order p and adds the length of the m transitions starting from a node of order $p - 1$ to the corresponding survivor of order $p - 1$. The result is the lengths of the m possible paths from the origin to the nodes of order p but passing by a given node of order $p - 1$. Among these m paths only the shortest one is retained. This calculation is repeated for all nodes of order $p - 1$, thus forming the survivors of order p .

When applied to communication systems, the Viterbi algorithm no longer gives *hard decisions* (taken bit by bit) but *soft decisions* based on the whole followed path. In presence of white Gaussian noise it is then optimum in the sense of maximum a posteriori likelihood.

In the next section, we develop the duobinary signaling (with a brief presentation of other partial response codes). Then we present analytical methods for evaluating the theoretical BER in presence of noise and distortions with duobinary and Viterbi decoding.

3 Duobinary code and partial response codes

3.1 General presentation

A *partial response code* is a weighted addition of n successive input data bits such as the coded symbol d_i is given by

$$d_i = k(h_0 a_i + h_1 a_{i-1} + h_2 a_{i-2} + \dots + h_n a_{i-n}), \quad (1)$$

where k is a scale factor and the coefficients h define the code. If we introduce the *delay operator of one bit*, denoted D , the code may equivalently be represented by the polynomial $P(D)$ defined as

$$d_i = kP(D) = k(h_0 + h_1 D + h_2 D^2 + \dots + h_n D^n). \quad (2)$$

3.2 Duobinary code

Duobinary coding (also referred to as Class 1) was proposed as a mean to introduce some controlled amount of *Inter-symbol Interference* (ISI) from the adjacent bit rather than trying to eliminate it completely. If a_i are the binary input bits (0 or 1) and d_i the duobinary symbols (with the symmetrical values -1 , 0 , or $+1$) at time t_i , the duobinary symbols are defined by the transverse filter [14, 19]

$$d_i = \frac{a_i + a_{i-1}}{2} = \frac{1}{2}(1 + D). \quad (3)$$

If the binary signal has the form of short pulses $\delta(t)$, the duobinary signal can also be obtained by convolution with the impulse response of the *duobinary filter* which is

$$G(f) = \cos\left(\frac{\pi f}{f_b}\right), \quad (4)$$

f_b being the bit frequency. The spectrum of the duobinary signal is then cascaded with an ideal filter and strictly limited to the Nyquist band $f_b/2$.

The duobinary filter can either be *matched* (*i.e.* split in equal parts between the transmitter and the receiver) or *unmatched* (*i.e.* entirely done at the transmitter). Note that matched filtering introduces a correlation between the successive noise samples. If desired the binary input bits can be precoded into the bits b_i according to the modulo-2 addition

$$b_i = \hat{a}_i \oplus b_{i-1}, \quad (5)$$

where \hat{a}_i is the complement of a_i . But neither matched filtering nor precoding have a significant advantage for Viterbi decoding.

Transition arriving	Possible sequences of bits a_i	Duobinary symbol
-1	-1 -1	-1
	+1 -1	0
+1	-1 +1	0
	+1 +1	+1

Tab. 1: Possible state-transitions.

The states of the system s_1 and s_2 are (in symmetrical form) the bits -1 and $+1$ of the binary sequence to be coded. The state-transitions are given by the corresponding duobinary symbol (see Table 1).

If x is the decoder input level without noise and y the same level with noise, the metric adopted (which is optimal) is the quadratic difference between x and y , expressed as

$$\delta = (x - y)^2 = x^2 - 2xy + y^2. \quad (6)$$

Let us note $\gamma_{1,p-1}$ and $\gamma_{2,p-1}$ the lengths of the survivors of order $p-1$ and $\lambda_{1,p}$ and $\lambda_{2,p}$ the lengths of the possible paths arriving at nodes -1 and $+1$ at order p . These paths are formed by adding the length of the last branch to the survivors of order $p-1$. This is illustrated in Table 2.

Transition	Duobinary symbol	Metric of the last branch	Length of the paths
-1 -1	-1	$1 + 2y + y^2$	$\lambda_{1,p} = \gamma_{1,p-1} + 1 + 2y + y^2$
+1 -1	0	y^2	$\lambda_{1,p} = \gamma_{2,p-1} + y^2$
-1 +1	0	y^2	$\lambda_{2,p} = \gamma_{1,p-1} + y^2$
+1 +1	+1	$1 - 2y + y^2$	$\lambda_{2,p} = \gamma_{2,p-1} + 1 - 2y + y^2$

Tab. 2: Length of the possible paths at order p .

The first test is to search which of the two values of $\lambda_{1,p}$ and $\lambda_{2,p}$ is the smallest one. For a transition arriving at state -1 , this test can be written as

$$\gamma_{1,p-1} - \gamma_{2,p-1} \lesseqgtr -1 - 2y. \quad (7)$$

Note that, in this expression, the symbol \lesseqgtr denotes the test “< OR >”. If the response to this test is “<” the shortest path corresponds to the transition $-1 -1$ and the survivor S_1 of order p prolongs the survivor S_1 of order $p-1$. On the other hand if the response is “>” the survivor S_1 of order p prolongs the survivor S_2 of order $p-1$. A similar test is made on the transitions arriving at state $+1$, and so on.

3.3 Other partial response codes

Apart from the duobinary or Class 1 code, there are four other non-extended partial response codes defined as follows by their polynomial $P(D)$ [20]:

- Class 2: $P(D) = 1 + 2D + D^2$,
- Class 3: $P(D) = 2 + D - D^2$,
- Class 4: $P(D) = 1 - D$,
- Class 5: $P(D) = 1 + 2D^2 - D^4$.

Note that the Class 3 code may be of some interest for our application because of its good resistance to echoes in AM/VSB modulation [17].

Extended partial response codes have a polynomial of the form $P(D) = (1 - D)^m(1 + D)^n$, duobinary coding being a sub-case for which $m = 0$ and $n = 1$. These codes have been extensively studied in particular for application like data magnetic recording and disk storage, where as already mentioned, the error-rate should be very low (*i.e.* a high signal to noise ratio).

For example reference [22] describes techniques for the construction of “good” codes (with $n = 1$) by matching the trellis to the channel and using a “pre-coder” which can be a punctured convolutional code. Reference [16] is an attempt to simplify the Viterbi decoder by deleting the less probable branches in the trellis. Reference [1] is a systematic search of state-sequences for which the squared Euclidean distance from the decoded sequence has a given value. This paper will be analyzed later (see Section 4.10) where we give a theoretical upper bound for the final error-rate with duobinary coding.

The paper of Vityaev and Siegel [26] gives, by solving a series of problems in linear programming, upper and lower bounds of the metric difference between two paths of same length in the trellis but arriving at two different states. This consideration is essential for the practical implementation of the Viterbi decoder (see Section 3.4). Despite of its practical importance this aspect is not dealt with in this paper where we are not considering any implementation of a duobinary Viterbi decoder in the form of integrated circuits, some of them having already been developed by the industry.

3.4 Implementation of the Viterbi decoder

The results given in this paper were always obtained by software (*i.e.* a number of computer programs specially written). In hardware implementations, the received signal as well as the metrics and the survivors path should be, in discrete form, represented by binary numbers with a sufficient number of bits and a small enough quantizing interval, while assuring a sufficiently high speed and hardware simplicity.

Reference [23] gives an example of a CMOS implementation developed at the CCETT (France) of a Viterbi decoder for duobinary signals, with a discussion of the choices made for quantization. Risks of overflow also need to be addressed carefully. Bounds of metrics, as derived by Vityaev and Siegel [26], are useful to avoid overflows in the registers which could otherwise produce long error bursts.

The paper of Chang [7] presents another design of Viterbi decoder based on in-place state metric update and hybrid survivor path management. It describes the three main functional blocks of the decoder: a Branch Metric Unit which receives the noisy symbols and computes the corresponding branch metric, the Add-Compare-Select (ACS) unit which updates the metrics for all states by proper comparison (*i.e.* what in this paper we call the *tests* made by the decoder) and Trace-Back Unit which searches the best survivor path on the basis of accumulated decisions. The design of these units result in an efficient architecture for a Viterbi decoder specially in the cases of trellis with a large number of states and a large value of the constraint length, for example for decoding convolutional codes or concatenations of two codes.

The paper of Zand and Johns [27] describes an analog CMOS Viterbi detector for use on a 4-PAM duobinary signaling. The chip is an implementation of a reduced state sequence with pipelining and parallel processing and the application studied is optical links.

It is also known that implementations of Viterbi decoders have been developed by industry, even if the results have not been published as generally available papers.

3.5 Some properties of the duobinary code

3.5.1 Duobinary eye diagram

In order to compute the error-rate in presence of a synchronization error or of an echo, it is necessary to know the decoder input level at any time and not only at the nominal sampling instants. This can be given by the eye diagram. A typical duobinary eye diagram, obtained by simulation with a random sequence, is given in Figure 2. It is seen that there is no inter-symbol interference at the nominal sampling instants kT , where T denotes the duration of one bit.

A first approximation of the level of the diagram at any time t is obtained by superposing a large number of impulse responses of the duobinary filter, shifted by n bit periods, including all possible binary sequences of $+1$ and -1 up to length 4, and by developing the result in Fourier series. It is found that the level can take twenty analytical values with equal probabilities. For illustration, only the first four values are given hereafter, with a \pm sign:

$$x_1(t) = \pm 1, \quad (8)$$

$$x_2(t) = \pm \left(\frac{1}{3} + \frac{2}{3} \cos \left(\frac{2\pi t}{3T} \right) \right). \quad (9)$$

The eye diagram reconstructed by superposing these twenty forms is shown in Figure 3.

A second (and better) approximation is obtained by extending this procedure to sequences of -1 and $+1$ up to length 9. Development in Fourier series was not made for this case because the number of possible analytical forms becomes too large. Instead, all the possible

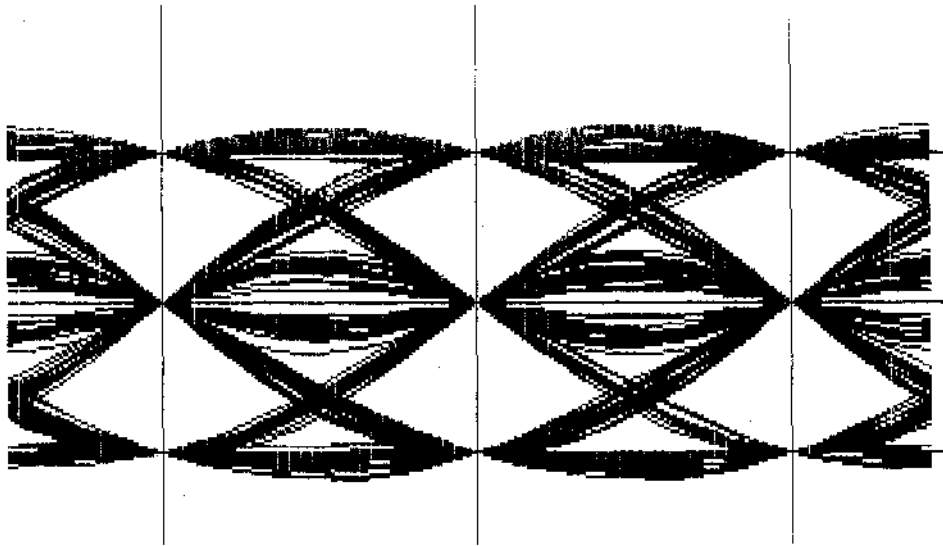


Fig. 2: Duobinary eye diagram (by simulation).

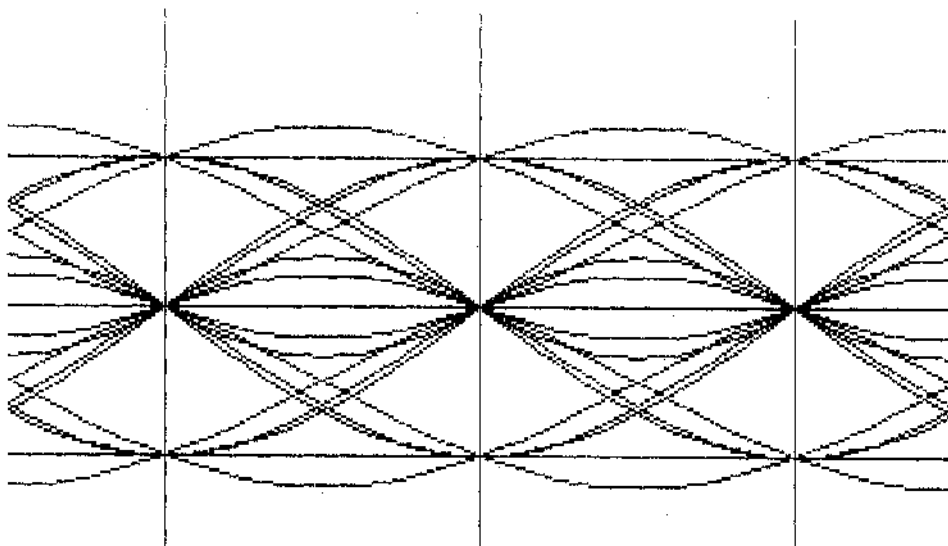


Fig. 3: Reconstructed duobinary eye diagram (first approximation).

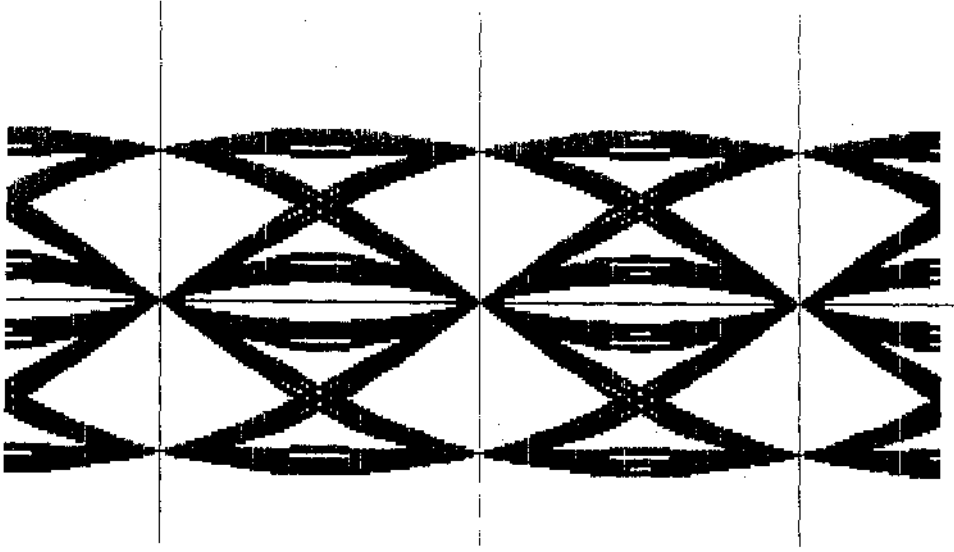


Fig. 4: Reconstructed duobinary eye diagram (second approximation).

levels of the eye diagram were stored in a computer file with a sampling frequency of 16 times the bit frequency. This second approximation is given in Figure 4.

For the case of AM/SSB modulation, the same method was used to obtain the eye diagram of the duobinary signal in quadrature by filtering all duobinary sequences obtained after the duobinary filter by the quadrature filter of transfer function

$$G_q(f) = e^{-j\frac{\pi}{2}} \text{sign}(f). \quad (10)$$

All the levels were again stored in a computer file and the corresponding eye diagram is given in Figure 5.

3.5.2 Correlation of the duobinary signal

The correlation coefficients between duobinary symbols separated by jT were computed for the following cases:

- D : autocorrelation of a duobinary signal,
- DQ : autocorrelation of a duobinary signal in quadrature,
- I : intercorrelation between D and DQ ,

with the same values for negative separation j , except a change of sign for the intercorrelation I . Results are drawn in Table 3.

These values are considered in Section 5 for the case of a correlated interference due to another duobinary sequence.

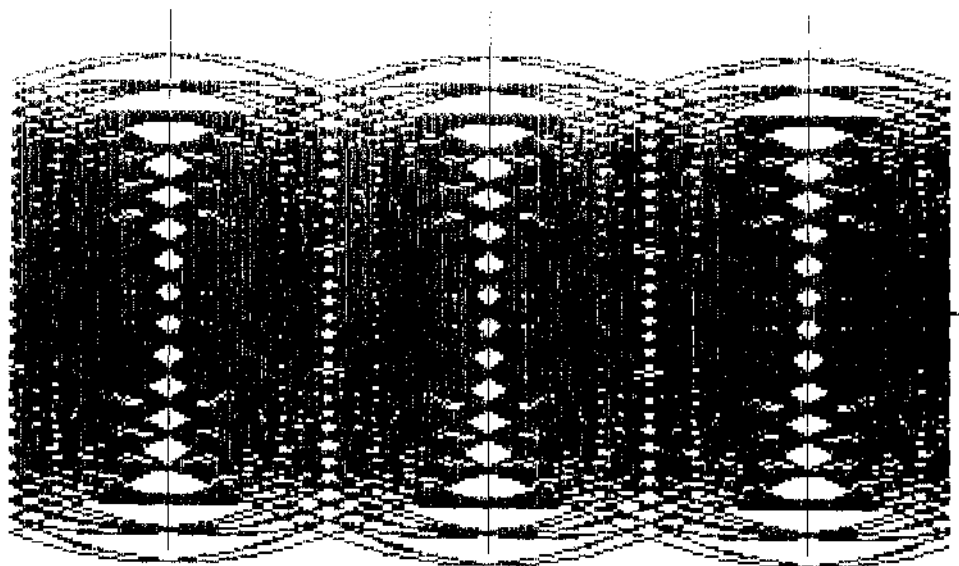


Fig. 5: Eye diagram of a duobinary signal in quadrature.

Separation j	D	DQ	I
6	$ \lt 10^{-2} $	-1.8×10^{-2}	1.3×10^{-2}
5	$ \lt 10^{-2} $	-8.0×10^{-2}	$ \lt 10^{-2} $
4	$ \lt 10^{-2} $	-0.135	-4.7×10^{-2}
3	$ \lt 10^{-2} $	-0.109	-0.105
2	$ \lt 10^{-2} $	-6.9×10^{-2}	-0.211
1	0.25	0.189	-0.312
0	0.5	0.440	$ \lt 10^{-2} $

Tab. 3: Duobinary correlation coefficients.

3.5.3 Probability of the duobinary symbols

If the binary input data -1 and 1 have the same probability, the probabilities of the duobinary symbols $-1, 0, 1$ are respectively $1/4, 1/2$ and $1/4$.

4 Theoretical framework for duobinary coding and the Viterbi decoding

A detailed description of the Viterbi decoding is given in references [3, 10, 15, 25]. The length of the possible paths at order p in the trellis was already given in Table 2 above.

4.1 Formulation of the tests made by the Viterbi decoder

In order to construct the survivors at order p , the decoder looks for the shortest path between the two values of $\lambda_{1,p}$ and $\lambda_{2,p}$. If the decoder keeps in memory the *renormalized metric* μ_p , which is the difference in length between the two survivors, we have

$$\mu_{p-1} = \gamma_{1,p-1} - \gamma_{2,p-1}, \quad (11)$$

$$\mu_p = \gamma_{1,p} - \gamma_{2,p}. \quad (12)$$

The tests on the path lengths are then

$$\mu_{p-1} + 2y \leq -1, \quad (13)$$

$$\mu_{p-1} + 2y \leq 1. \quad (14)$$

It can be shown that three different situations can occur with the following consequences:

1. If

$$\mu_{p-1} + 2y < -1 \quad (15)$$

then

$$\mu_p = 1 + 2y. \quad (16)$$

The decoder recognizes the duobinary symbol -1 and the survivors S_1 and S_2 of order p prolong the survivor S_1 at order $p - 1$; in other words the survivor S_1 is copied into the new survivors S_1 and S_2 .

2. If

$$-1 \leq \mu_{p-1} + 2y \leq 1 \quad (17)$$

then

$$\mu_p = -\mu_{p-1}. \quad (18)$$

The decoder recognizes the duobinary symbol 0 and the survivor S_1 at order p prolongs the survivor S_2 at order $p - 1$, while the survivor S_2 at order p prolongs the survivor S_1 , so that S_1 is copied in the new survivor S_2 and S_2 is copied in the new survivor S_1 . In other words, survivors are *crossed*.

3. If

$$\mu_{p-1} + 2y > 1 \quad (19)$$

then

$$\mu_p = 1 - 2y. \quad (20)$$

The decoder recognizes the duobinary symbol 1 and the survivors S_1 and S_2 at order p prolong the survivor S_2 of order $p - 1$; in other words S_2 is copied into the new survivors S_1 and S_2 .

After these tests -1 is added to S_1 and 1 is added to S_2 and the two survivors are truncated and shifted by one position. Since the two survivors are converging, decoding of the input bit a_i can be made after a limited number of tests, the *decoding constraint length* being typically 20 to 30 bits.

4.2 Alternative formulation of the tests

In the following, we consider additive noise. We may therefore rewrite y_p as

$$y_p = d_p + n_p. \quad (21)$$

Assume that, at order p ,

$$\mu_p + 2y_p = \mu_p + 2(d_p + n_p) < -1. \quad (22)$$

The next value for the normalized metric is

$$\mu_{p+1} = 1 + 2y_p = 1 + 2(d_p + n_p), \quad (23)$$

so that the new test is made on the quantity

$$\mu_{p+1} + 2y_{p+1} = 1 + 2(d_p + n_p) + 2(d_{p+1} + n_{p+1}). \quad (24)$$

If this test is again giving the answer < -1 , then

$$\mu_{p+2} = 1 + 2y_{p+1}, \quad (25)$$

and so on.

If the test (24) gives the answer > 1 , then

$$\mu_{p+2} = 1 - 2y_{p+1} = 1 - 2(d_{p+1} + n_{p+1}). \quad (26)$$

If finally the quantity (24) is between -1 and 1 ,

$$\mu_{p+2} = -\mu_{p+1} = -1 - 2(d_p + n_p), \quad (27)$$

and the test $p + 2$ is made on the quantity

$$\mu_{p+2} + 2y_{p+2} = -1 - 2(d_p + n_p) + 2(d_{p+2} + n_{p+2}), \quad (28)$$

where the duobinary symbols and the noise samples are d_p and d_{p+2} , and n_p and n_{p+2} , respectively. Therefore as long as the decoder recognizes duobinary symbols 0 , the test involves the transmitted symbol d_p and the noise sample n_p corresponding to the last test which gave the answer < -1 or > 1 .

With a slight change in the notations, the following rules can therefore be established.

1. Last test $(i - k)$ with answer > 1
Test i is made on the quantity

$$2(d_i + d_{i-k}) + 2(n_i + n_{i-k}) - 1 \quad \text{with } k \text{ odd} \quad (29)$$

$$2(d_i - d_{i-k}) + 2(n_i - n_{i-k}) + 1 \quad \text{with } k \text{ even} \quad (30)$$

2. Last test $(i - k)$ with answer < -1
Test i is made on the quantity

$$2(d_i + d_{i-k}) + 2(n_i + n_{i-k}) + 1 \quad \text{with } k \text{ odd} \quad (31)$$

$$2(d_i - d_{i-k}) + 2(n_i - n_{i-k}) - 1 \quad \text{with } k \text{ even} \quad (32)$$

4.3 Duobinary error probability with low noise

If there is no noise, the symbol r_i recognized by the decoder is always the transmitted symbol d_i .

If we consider a very low level of the noise, the duobinary error-rate is not zero. Suppose $d_i = 1$ is the transmitted symbol. The test i , as given by (29) to (32), provides a value close to 3 which is well separated from the limit 1 and there is no error. If $d_i = -1$, test i gives a value close to -3 and again there is no error. If finally $d_i = 0$, test i gives a value close to 1 or to -1 and there is therefore a probability equal to $1/2$ that the decoder recognizes 1 (or -1) instead of 0 .

The mean error-rate can be found by computing the error probability on the last 0 of the duobinary sequences:

- 10 and -10 where the probability of the symbols before the last 0 is $1/4$,

- 100 and -100 where the probability of the symbols before the last 0 is $1/8$,

and so on.

It can be shown that, with the method of numerical integrations described later in Sections 4.4 and 4.5, the duobinary error-rate with low noise at length k is in fact $1/k$. By summing the terms $1/k$ for $k = 2, 3, 4, \dots$, weighted by the probabilities of the symbols before the last 0, we see that the mean duobinary error probability is given by

$$P_d(\text{low noise}) = \frac{1}{2} \frac{1}{4} + \frac{1}{3} \frac{1}{8} + \frac{1}{4} \frac{1}{16} + \dots, \quad (33)$$

$$P_d(\text{low noise}) = \sum_{k=2}^{\infty} \frac{1}{k 2^k} = 0.193. \quad (34)$$

This value was confirmed by simulation. In the next section, we examine the effect for any level of noise.

4.4 Probability of a recognized sequence r for a transmitted sequence d

For any level of the noise, if d_i are the transmitted duobinary symbols and r_i the symbols decoded (correctly or not), the tests (29) to (32) where d_i and r_i are replaced by their values, take the form

$$n_i < a \mp n_{i-k}, \quad (35)$$

$$n_i > b \mp n_{i-k}, \quad (36)$$

the limits a and b depending on d_i , d_{i-k} , r_i and r_{i-k} .

As an example, let us consider the transmitted sequence

$$d_i = 1000, \quad (37)$$

and the recognized sequence

$$r_i = 1(-1)00. \quad (38)$$

(Note that we will always start the sequences d_i and r_i with 1 or -1 in order to start the calculation with a test $i - k$ of known value).

The tests made by the decoder can be written in this case

$$n_1 < -1 - n_2, \quad (39)$$

$$n_3 < -n_2 \quad \text{and} \quad n_3 > -1 - n_2, \quad (40)$$

$$n_4 < 1 + n_2 \quad \text{and} \quad n_4 > n_2. \quad (41)$$

The problem is to compute the probability of a system of simultaneous inequalities like (39) to (41), where the n_i are independent samples of Gaussian noise with variance σ^2 . However note that the same noise samples (in this case n_2) can appear in several inequalities.

Let $g(n)$ be the Gaussian probability density distribution

$$g(n_i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{n_i^2}{2\sigma^2}}. \quad (42)$$

The probability P_r of (39) to (41) is given by

$$P_r = \int_{-\infty}^{+\infty} g(n_2) \int_{-\infty}^{-1-n_2} g(n_1) \int_{-1-n_2}^{-n_2} g(n_3) \int_{n_2}^{1+n_2} g(n_4) dn_2 dn_1 dn_3 dn_4, \quad (43)$$

which for a sequence of length L is, in general, an L -uple integral.

However, if a particular noise sample (in this case n_2) is common to several successive integrals, the calculation is simplified by introducing the Q function

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{u^2}{2}} du, \quad (44)$$

and (43) reduces to the simple integral

$$P_r = \int_{-\infty}^{+\infty} \left[1 - Q\left(\frac{-1-n_2}{\sigma}\right) \right] \left\{ \left[1 - Q\left(\frac{-n_2}{\sigma}\right) \right] - \left[1 - Q\left(\frac{-1-n_2}{\sigma}\right) \right] \right\} \\ \left\{ \left[1 - Q\left(\frac{1+n_2}{\sigma}\right) \right] - \left[1 - Q\left(\frac{n_2}{\sigma}\right) \right] \right\} dn_2. \quad (45)$$

Such a simplification is not always possible, and there are cases of length $L = 4$ where P_r is given by a double integral, of length 5 by a triple integral, and so on.

4.5 Duobinary error-rate

The duobinary error-rate due to an incorrect recognition of 0 is obtained by computing the error probability on the last 0 of the following duobinary sequences (of which the total probability of the symbols before the last 0 is $1/2 + 1/4 + 1/8 + \dots = 1$):

- $L = 2$: 10 and -10 ,
- $L = 3$: 100 and -100 ,
- $L = 4$: 1000 and -1000 ,

and so on. Sequences such as 110 or $-1 - 10$ should be excluded because we always start with a test on 1 or -1 followed by zeros.

The total error probability on 0 is then

$$P_{d0} = \frac{1}{2}p_{02} + \frac{1}{4}p_{03} + \frac{1}{8}p_{04} + \dots, \quad (46)$$

which is a sum weighted by the probabilities of the corresponding sequences.

Similarly, the duobinary error-rate due to a false recognition of 1 (or -1) is obtained by computing the error probability on the last 1 (or -1) of the following duobinary sequences:

- $L = 2$: 11,
- $L = 3$: -101 ,
- $L = 4$: 1001,

and so on.

The corresponding error-rate is

$$P_{d1} = P_{d-1} = \frac{1}{2}p_{12} + \frac{1}{4}p_{13} + \frac{1}{8}p_{14} + \dots \quad (47)$$

The error probabilities on 0 and 1 of all these sequences, up to length $L = 5$, were computed by the expressions of the previous Sections, using the software “Mathematica” for integration.

For the length $L = 6, 7, 8$, the theoretical calculation involves quadruple, quintuple and sextuple integrals, and the computation time at each step is multiplied by a significant factor. A data base of all possible duobinary errors was therefore filled in by simulation, not of the Viterbi decoding itself, but on the tests on the noise samples expressed by a system of inequalities such as (39) to (41).

Because the terms of the series (46) and (47) are decreasing, the contribution of the lengths larger than 8 is negligible and the dominant terms of the final results were computed analytically.

Since the error probabilities on 1 and -1 are equal (due to the symmetry of the code), the final mean duobinary error-rate is

$$P_d = \frac{1}{2}(P_{d0} + P_{d1}). \quad (48)$$

This theoretical result and values obtained by simulation are drawn on Figure 6 as a function of the signal-to-noise ratio S/N (in baseband with unmatched filtering). As shown by this Figure, the present theory is well confirmed by simulation.

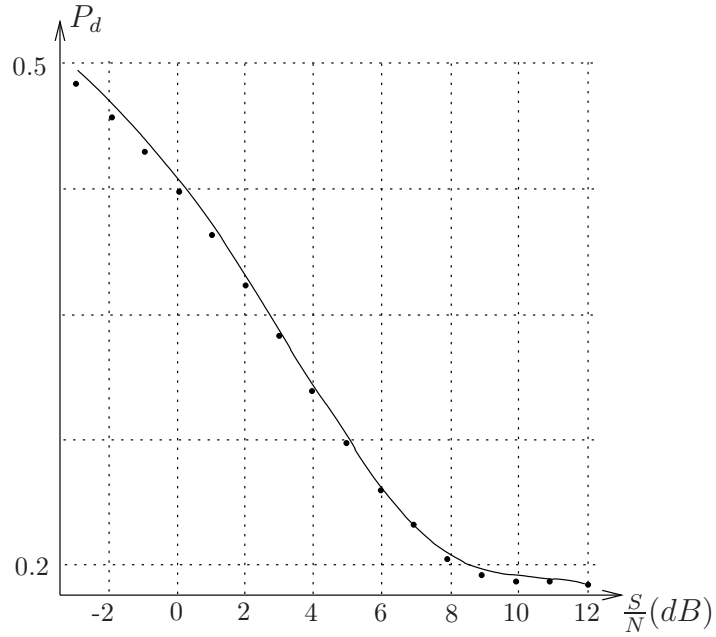


Fig. 6: Duobinary error-rate as a function of S/N : theoretical curve and points of simulation.

4.6 Error-rate on the survivors

Let us call a *simple error* an error at one position on one of the survivors and a *double error* an error which affects both survivors at the same position.

The computation of the error-rate on the survivors is complex but we can use the following rules that we have established:

- If the duobinary transmitted symbol 0 is recognized as -1 , there are $n+1$ consecutive simple errors on the survivor S_1 , n being the number of times that a symbol 0 is recognized correctly before the duobinary error.
- If a transmitted 0 is recognized as 1, there are similarly $n+1$ consecutive simple errors on S_2 .
- If a transmitted -1 is recognized as 0, there are $n+1$ consecutive simple errors on S_1 .
- If a transmitted 1 is recognized as 0, there are $n+1$ consecutive simple errors on S_2 .
- If a transmitted 1 (or -1) is recognized as -1 (or 1), there are $n+1$ consecutive double errors on S_1 and S_2 .

If $P_0 = 1/2$ is the probability of symbol 0 and P_{d0} the duobinary error probability on 0, the probability of one simple error on S_1 (or S_2) due to a duobinary error on 0 is given by

$$P_{s,1}(0) = [1 - P_0(1 - P_{d0})]P_0P_{d0}, \quad (49)$$

the term within brackets representing the probability of having anything other than a 0 correctly recognized before the duobinary error.

The probability to have k consecutive simple errors is then

$$P_{s,k}(0) = [1 - P_0(1 - P_{d0})]P_0^{k-1}(1 - P_{d0})^{k-1}P_0P_{d0}. \quad (50)$$

If $P_1 = 1/4$ is the probability of symbol 1 and P_{d10} the probability of a duobinary error on a symbol 1 recognized as 0, the probabilities of a single simple error on S_1 or S_2 or of k consecutive simple errors are

$$P_{s,1}(1) = [1 - P_0(1 - P_{d0})]P_1P_{d10}, \quad (51)$$

$$P_{s,k}(1) = [1 - P_0(1 - P_{d0})]P_0^{k-1}P_1(1 - P_{d0})^{k-1}P_{d10}, \quad (52)$$

and for the double errors we have

$$P_{sd,1}(1) = [1 - P_0(1 - P_{d0})]P_1P_{d1-1}, \quad (53)$$

$$P_{sd,k}(1) = [1 - P_0(1 - P_{d0})]P_0^{k-1}P_1(1 - P_{d0})^{k-1}P_{d1-1}, \quad (54)$$

where P_{d1-1} is the probability that 1 (or -1) is recognized as -1 (or 1). The mean rate of simple and double errors on the survivors were computed with these expressions and compared to values obtained by simulation. The results are given in Figures 7 and 8.

4.7 Evolution of the errors on the survivors

The necessary and sufficient condition for a binary error to occur is a double error on the survivors, because such an error will never be eliminated if S_1 or S_2 are *recopied* or *crossed* (see Section 4.1). Fortunately during the process of survivors convergence, most of the simple errors will be eliminated, while some of these simple errors are transformed into double errors. Indeed, let us suppose that a single simple error occurs at step i on S_1 and that, after the error, the decoder recognizes $j - 1$ symbols 0. Then the survivors are crossed $j - 1$ times. At step $i + 1$ the error is transported at S_2 , at step $i + 2$ again at S_1 , and so on. If now at step $i + j$, the decoder recognizes a symbol -1 , the survivor S_1 of step $i + j - 1$ is copied into S_1 and S_2 . If j is odd, then the simple error becomes a double error, while the error is eliminated if j is even.

Generalization of this reasoning gives the following rules for the conditions of transformation of a simple error into a double error and thus into a binary error

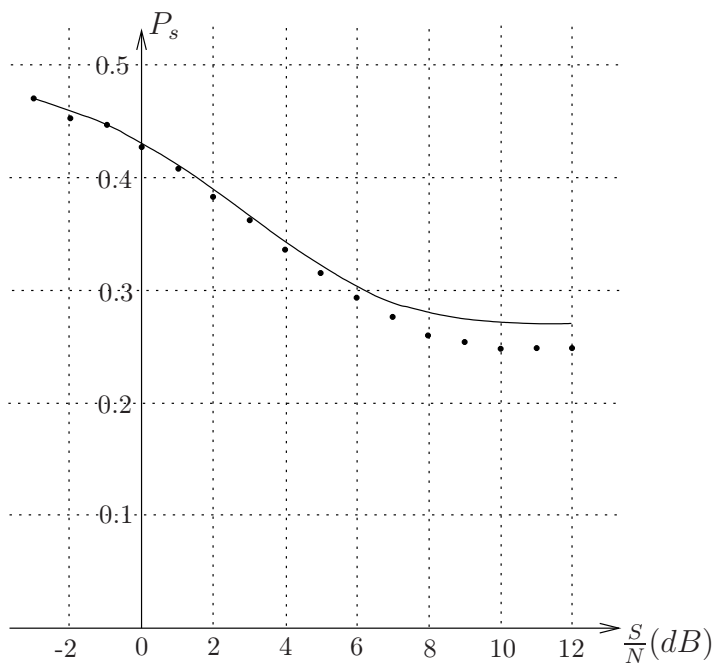


Fig. 7: Simple error-rate on the survivors as a function of S/N : theoretical curve and points of simulation.

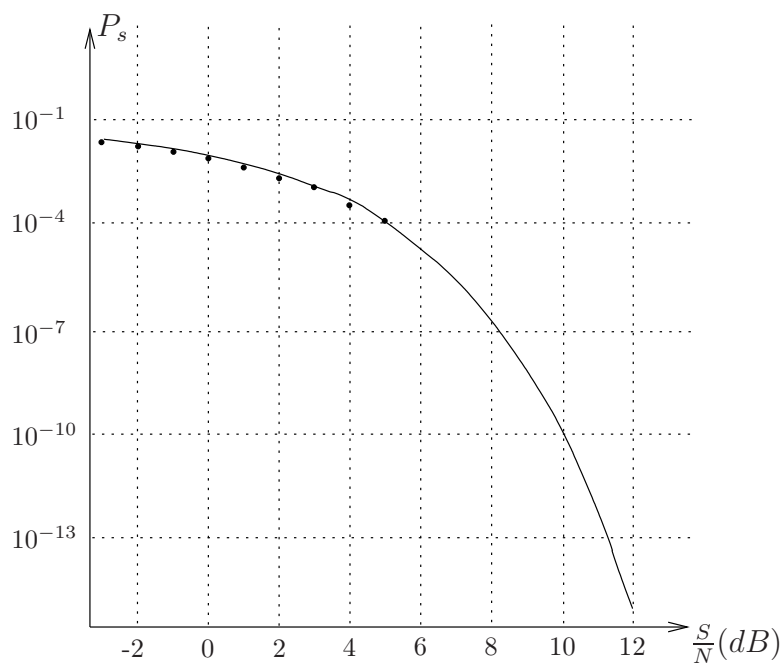


Fig. 8: Double error-rate on the survivors as a function of S/N : theoretical curve and points of simulation.

4.7.1 Normal rules

1. Simple error at step i on S_1 becomes a double error if

- test $i + j$ gives < -1 with j odd.
- test $i + j$ gives > 1 with j even.

2. Simple error at step i on S_2 becomes a double error if

- test $i + j$ gives < -1 with j even.
- test $i + j$ gives > 1 with j odd.

4.7.2 Additional rules

The rules given above are only valid if there are no new errors on the survivors between test i (where there is an initial error) and test $i + j$ (where the initial error is eliminated or transformed into a binary error). This is, in fact, the case when the binary error-rate is low.

If the signal is affected by a strong noise or distortion, the probability of new errors between i and $i + j$ is no longer negligible and the behavior of the decoder becomes more complex. Some additional rules for the evolution of the survivors errors must then be introduced. The most frequent of them are given below.

1. *Simple errors at i followed by simple errors at $i + m$ with $m < j$.*

Let $IS1(i)$ and $IS2(i)$ be the numbers of simple errors occurring on survivors S_1 and S_2 at test i . We have:

If	Then everything is as if
$IS1(i) > 0, IS2(i + m) > 0, r(i) = -1, m \text{ odd } < j$	$IS1(i) = 0$
$IS2(i) > 0, IS1(i + m) > 0, r(i) = 1, m \text{ odd } < j$	$IS2(i) = 0$
$IS1(i) > 0, IS1(i + m) > 0, r(i) = -1, m \text{ even } < j$	$IS1(i) = 0$
$IS2(i) > 0, IS2(i + m) > 0, r(i) = 1, m \text{ even } < j$	$IS2(i) = 0$

2. *Simple errors at i followed by double errors at $i + m$ with $m = j$.*

Let in addition $ID(i)$ be the number of double errors on the survivors at test i . We have:

If	Then everything is as if
$IS1(i) > 0, ID(i+m) > 0,$ $r(i) = -1, r(i+m) = -1, m \text{ odd} = j$	$IS1(i) = 0$
$IS2(i) > 0, ID(i+m) > 0,$ $r(i) = 1, r(i+m) = 1, m \text{ odd} = j$	$IS2(i) = 0$
$IS1(i) > 0, ID(i+m) > 0,$ $r(i) = -1, r(i+m) = 1, m \text{ even} = j$	$IS1(i) = 0$
$IS2(i) > 0, ID(i+m) > 0,$ $r(i) = 1, r(i+m) = -1, m \text{ even} = j$	$IS2(i) = 0$

3. *Non transformed errors: simple errors at i followed by simple errors at $i+m$ with $m = j$.*

Let now $IS1N(i)$ and $IS2N(i)$ be the numbers of simple errors on survivors S_1 and S_2 that would be transformed in double errors at test j according to the normal rules but that are in fact not transformed. Then

If	Then
$IS1(i) > 0, IS1(i+m) > 0,$ $r(i) = -1, r(i+m) = -1, m \text{ odd} = j$	$IS1N(i) = IS1(i)$
$IS2(i) > 0, IS2(i+m) > 0,$ $r(i) = 1, r(i+m) = 1, m \text{ odd} = j$	$IS1N(i) = IS1(i)$
$IS1(i) > 0, IS2(i+m) > 0,$ $r(i) = -1, r(i+m) = 1, m \text{ even} = j$	$IS1N(i) = IS1(i)$
$IS2(i) > 0, IS1(i+m) > 0,$ $r(i) = 1, r(i+m) = -1, m \text{ even} = j$	$IS1N(i) = IS1(i)$

4. *Evolution of non transformed errors.*

Let $i+j+j_1$ be the first test after $i+j$ where the symbol recognized by the decoder is -1 or 1 . Then if $IB(i)$ is the number of *binary* errors due to duobinary errors at test i :

If	Then also
$IS1N(i) > 0, j \text{ even}, j_1 \text{ even}, r(i+j+j_1) = 1$	$IB(i) = IS1N(i)$
$IS1N(i) > 0, j \text{ even}, j_1 \text{ odd}, r(i+j+j_1) = -1$	$IB(i) = IS1N(i)$
$IS1N(i) > 0, j \text{ odd}, j_1 \text{ even}, r(i+j+j_1) = -1$	$IB(i) = IS1N(i)$
$IS1N(i) > 0, j \text{ odd}, j_1 \text{ odd}, r(i+j+j_1) = 1$	$IB(i) = IS1N(i)$
$IS2N(i) > 0, j \text{ even}, j_1 \text{ even}, r(i+j+j_1) = -1$	$IB(i) = IS2N(i)$
$IS2N(i) > 0, j \text{ even}, j_1 \text{ odd}, r(i+j+j_1) = 1$	$IB(i) = IS2N(i)$
$IS2N(i) > 0, j \text{ odd}, j_1 \text{ even}, r(i+j+j_1) = 1$	$IB(i) = IS2N(i)$
$IS2N(i) > 0, j \text{ odd}, j_1 \text{ odd}, r(i+j+j_1) = -1$	$IB(i) = IS2N(i)$

If	Then also
$IS1N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ even}, j_1 \text{ even}, r(i + j + j_1) = -1$	$IB(i) = IS1N(i)$
$IS1N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ even}, j_1 \text{ odd}, r(i + j + j_1) = 1$	$IB(i) = IS1N(i)$
$IS1N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ odd}, j_1 \text{ even}, r(i + j + j_1) = 1$	$IB(i) = IS1N(i)$
$IS1N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ odd}, j_1 \text{ odd}, r(i + j + j_1) = -1$	$IB(i) = IS1N(i)$
$IS2N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ even}, j_1 \text{ even}, r(i + j + j_1) = 1$	$IB(i) = IS1N(i)$
$IS2N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ even}, j_1 \text{ odd}, r(i + j + j_1) = -1$	$IB(i) = IS1N(i)$
$IS2N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ odd}, j_1 \text{ even}, r(i + j + j_1) = -1$	$IB(i) = IS1N(i)$
$IS2N(i) > 0, ID(i + j + j_1) > 0,$ $j \text{ odd}, j_1 \text{ odd}, r(i + j + j_1) = 1$	$IB(i) = IS1N(i)$

Although the *normal* and additional rules cover most cases, there are still a few rare cases where the correct answer is not found. A more exact method will be described later, when considering distortions. In practice one can use the normal rules only when the error-rate is low, use the normal and additional rules when the error-rate is higher, but refer to a more exact (and more lengthy) calculation described later if the noise is very high and especially in presence of a strong and correlated interference.

4.8 Binary error-rate without precoding

With all the previous rules, the binary error-rate without precoding (or, with precoding, the error-rate on the precoded sequence) can be computed by the following algorithm:

- Consider all the possible duobinary transmitted sequences starting and ending with a symbol 1 or -1 of length $L = 3, 4, 5, \dots$
- Consider, for each transmitted sequence, all the possible recognized sequences and compute the duobinary error probability, by writing the system of inequalities corresponding to the duobinary tests and solving this system by numerical integrations.
- Count the number of simple and double errors on the survivors and then the number of simple errors transformed into double errors, by applying the normal and the additional rules.

The total probability of binary errors is then the sum of the computed duobinary error probabilities multiplied by the number of final double errors and weighted by the probabilities of the considered sequences. Fortunately the number of sequences introducing binary

errors is limited, which means that the majority of duobinary errors are eliminated during the convergence of the survivors.

The above algorithm was followed to obtain the binary error-rate. It was found that it is not necessary to consider sequences of length greater than 6, their contribution to the final result being negligible. The theoretical curve of the binary error-rate as a function of S/N is given in Figure 9 together with the results of simulation.

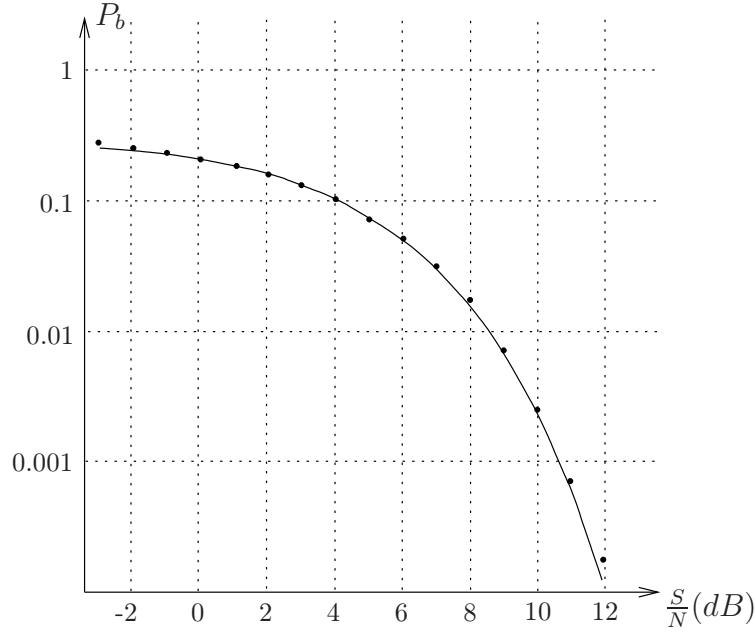


Fig. 9: Binary error-rate as a function of S/N without precoding: theoretical curve and points of simulation.

4.9 Binary error-rate with precoding

In order to obtain the binary error-rate with precoding, we have to note that a single binary error on the precoded sequence produces two binary errors on the input sequence, but that k consecutive errors on the precoded sequence only produce two errors too on the input sequence. The calculation procedure is then similar to the procedure described in the previous Section with two changes.

First we assume that each final double error on the survivors gives two binary errors on the input sequence, but this leads to an overestimated number of binary errors. Let q be the probability of a double error on the survivors. Consecutive double errors can occur 2, 3, ... times with probabilities q^2, q^3, \dots . It is easy to show that we must subtract, from q , the number of errors in excess which is

$$2q^2 + 4q^3 + 6q^4 + \dots, \quad (55)$$

so that the binary error probability becomes

$$q_p = 2 \left[q - 2q^2(1 + 2q + 3q^2 + 4q^3 + \dots) \right], \quad (56)$$

which is equal to

$$q_p = 2 \left[q - \frac{2q^2}{(1-q)^2} \right]. \quad (57)$$

The mean binary error-rate is finally computed by a weighted summation of the q_p terms corresponding to the different duobinary sequences. Theoretical values of the binary error-rate were computed as just described and compared with results obtained by simulation. All these values have been drawn in Figure 10.

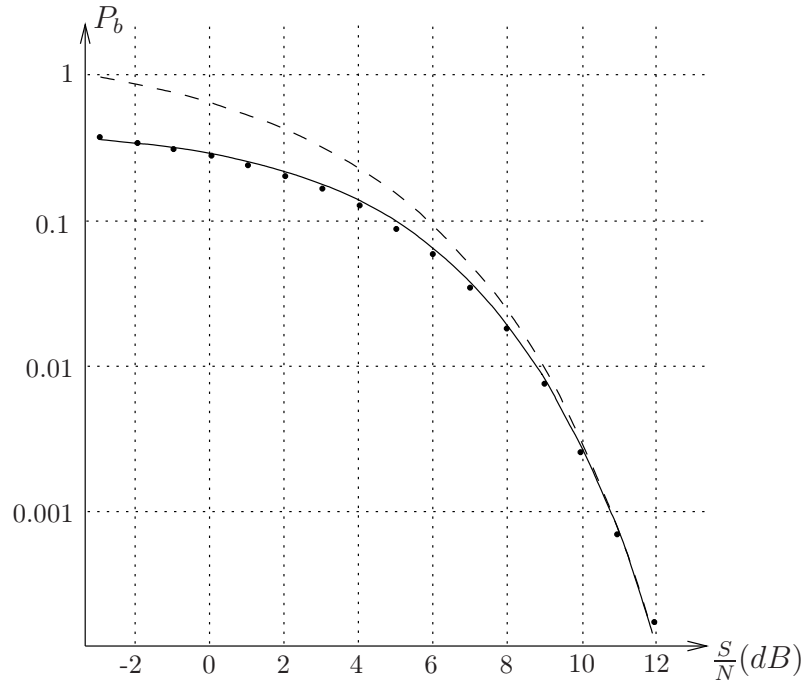


Fig. 10: Comparison of methods giving the binary error-rate as a function of S/N with precoding: theoretical curve (full line), simulations (points), and theoretical upper bound as established by Allard and detailed in Section 4.10 (dashed line).

This figure also shows the results of the theoretical upper bound of the BER developed by Allard in [23] which is summarized in the next Section.

4.10 Upper bound of the binary error-rate

The EBU document [23] reproduces a study made by Allard at the CCETT (France) where an upper bound of the BER is derived for the duobinary code. As this document is

not generally available, a summary of the theoretical calculations of the bound is given hereafter.

Let us denote by s_k the path correctly followed in the trellis (*i.e.* the sequence of the binary input states), by s'_k the path recognized by the decoder, by x_k and x'_k the duobinary sequences transmitted and recognized, and by y_k the received sequence impaired by the noise of variance σ^2 .

It is assumed that the path recognized is in general identical to the correct path but that it can diverge from the later immediately after time t and reconverge immediately before time $t + r$ (otherwise the code would be subject to “catastrophic error propagation” which has no practical interest). Such an event of divergence followed by reconvergence of the paths is denoted by ϵ and is called as a *closed error event*, by opposition to an *open error event* where there is no reconvergence of the paths until infinity and therefore catastrophic propagation.

The squared Euclidean distance between the transmitted and recognized duobinary sequences is defined by

$$D^2(\epsilon) = \sum_{k=t+1}^{k=t+r-1} (x_k - x'_k)^2, \quad (58)$$

which is shown to be an even integer larger or equal to 2.

It is then assumed that the recognized sequence x'_k is, among all the possible sequences x_i , the closest to the received sequence y_k . In other words, x'_k is at a lower distance of y_k than any other possible sequence and in particular the correct sequence x_k .

Since the decision distance is $D(\epsilon)/2$, the probability of an closed error event ϵ is bounded by

$$P(\epsilon) \leq Q\left(\frac{D(\epsilon)}{2\sigma}\right). \quad (59)$$

If considering only confusion between adjacent duobinary symbols, thus neglecting the less probable confusion between -1 and $+1$, the squared Euclidean distance (58) reaches its minimum value equal to 2.

It remains to count the number of duobinary sequences that are at the squared Euclidean distance 2 from the transmitted sequence x_k . If the path s'_k diverges from the correct path after instant t and reconverges just before instant $t + r$, the duobinary symbols x'_{t+1} and x'_{t+r-1} are different from x_{t+1} and x_{t+r-1} , which is sufficient to reach the minimum squared distance 2. This is only possible if the state sequence s_k takes alternate values 0 and 1 (or -1 and $+1$), the adversary sequence s'_k taking the opposite values.

Therefore there is only one path diverging from the correct path from $t + 1$ to $t + r - 1$ with the corresponding duobinary sequences x_k and x'_k being at the minimum squared Euclidean distance 2. The probability of existence of this path is $(1/2)^{r-2}$. It follows that, for the whole set \mathcal{E} of ϵ events, the probability $P(\mathcal{E})$ is, when making $D(\epsilon)$ in (59) equal to its minimum value which is $\sqrt{2}$,

$$P(\mathcal{E}) \leq \sum_{r=2}^{\infty} \left(\frac{1}{2}\right)^{r-2} Q\left(\frac{\sqrt{2}}{2\sigma}\right) = 2Q\left(\frac{1}{\sigma\sqrt{2}}\right). \quad (60)$$

Without precoding an ϵ event produces $r-1$ binary errors, while with precoding it produces 2 binary errors.

Finally the upper bound of binary error-rate is:

- without precoding,

$$P_b \leq \sum_{r=2}^{\infty} \frac{r-1}{2^{r-2}} Q\left(\frac{1}{\sigma\sqrt{2}}\right) = 4Q\left(\frac{1}{\sigma\sqrt{2}}\right), \quad (61)$$

- and with precoding,

$$P_b \leq \sum_{r=2}^{\infty} \frac{2}{2^{r-2}} Q\left(\frac{1}{\sigma\sqrt{2}}\right) = 4Q\left(\frac{1}{\sigma\sqrt{2}}\right). \quad (62)$$

Both (identical) expressions, developed by Allard, give a good approximation of the BER for high values of S/N but overestimate the error probability for low S/N ; this is clearly shown in the calculations by bounding the probability of an ϵ event in equation (59) and by neglecting the errors produced by confusion between duobinary symbols -1 and $+1$.

As both equations (61) and (62) provide the same result, it has been thought that the BER is the same with and without precoding. Although the error-rate becomes closer and closer to the value without precoding for high S/N , our results show that the error-rate is always higher without precoding. As seen in Figure 10, equations (61) and (62) clearly give a good approximation for the low noise channels (S/N high) but they become insufficient for channels impaired by strong noise.

The paper by Altekari *et al.* [1] presents a different approach based on the partial response codes defined by a polynomial of the form

$$h(D) = (1 - D)^m(1 + D)^n. \quad (63)$$

In this paper, an input error sequence is defined as

$$\varepsilon_s(D) = s(D) - s'(D), \quad (64)$$

where s and s' are respectively the correct states and the states recognized by the Viterbi decoder in binary numbers 0 and 1. An output error sequence is then

$$\varepsilon_y(D) = h(D) \varepsilon_s(D). \quad (65)$$

The performance of the system is largely dictated by input error sequences that result in an output with small squared Euclidean distance

$$\|\varepsilon_y(D)\|^2 = \sum_k \varepsilon_{y,k}^2. \quad (66)$$

The paper describes two algorithms, on the basis of a so-called *error state diagram*, used to derive all the input error sequences corresponding to a closed error event (where the paths s and s' diverge after time t and reconverge at time $t+r$) and to an open error event (where the paths do not reconverge). The input error sequences corresponding to a given squared Euclidean distance are finally listed for several codes, *i.e.* for different values of m and n in (63).

For $m = 0$ and $n = 1$, the code considered in the paper is the duobinary code which is the subject of our study. The first input error sequence listed for a closed error event and the minimum squared Euclidean distance equal to 2 is

$$(0), 1, (-1, 1, -1, 1, \dots) 0. \quad (67)$$

Recalling that this is the difference $s(D) - s'(D)$, we can obtain s and s' separately according to the following table

$s - s'$	s	s'
0	0	0
-1	0	-1
1	1	0
0	1	1

Therefore the input error sequence given corresponds to the following states:

$$s = 1, (0, 1, 0, 1, \dots) \quad (68)$$

$$s' = 0, (1, 0, 1, 0, \dots). \quad (69)$$

This is similar to the conclusion of Allard since during the interval of paths divergence, the correct state sequence s takes alternate values 0 and 1 while the adversary sequence s' takes the opposite values. Moreover if the input error sequence is duobinary coded in order to obtain the output error sequence, we have

$$1(0, 0, 0, 0, \dots)1, \quad (70)$$

which shows that the total squared Euclidean distance is 2.

Altekar *et al.* [1] give another input error event with a squared Euclidean distance 2 which leads to the same conclusion so that, when considering only the minimum squared Euclidean distance, the bounds of the binary error-rate given by (61) and (62) are confirmed. The same paper also gives other input error sequences with squared Euclidean distances of 6 and 10; these sequences could avoid neglecting confusion between duobinary symbols -1 and $+1$ which will make the previous bounds tighter.

4.11 Error statistics

The duobinary errors are practically independent. On the other hand, the binary errors are certainly not independent since we have seen that a single duobinary error produces $n + 1$ consecutive survivors errors, n being the number of symbols 0 correctly recognized by the decoder before the duobinary error. One must therefore expect that a Viterbi decoder, with and without precoding, will give errors in bursts. With precoding, one single binary error as well as a burst of consecutive errors on the precoded sequence produces two final binary errors. It is then expected that with precoding the bursts are somewhat less frequent than without precoding.

To quantify this behavior we have computed by simulation the probabilities of 2, 3 and 4 consecutive binary errors. If p is the probability of binary errors and if these errors were independent, the probability of k consecutive errors would be p^k , while the actual probabilities of k consecutive errors are higher. Let p_k denote the probability of k consecutive errors. Table 4 gives, in rounded figures, the ratio p_k/p^k for $k = 2, 3, 4$.

S/N (dB)	Without precoding			With precoding		
	$k = 2$	$k = 3$	$k = 4$	$k = 2$	$k = 3$	$k = 4$
0	1.5	2.5	4	1.5	1.5	2.5
6	10	65	5	5	5	25
9	65	4×10^3	525	33	30	730

Tab. 4: Ratio p_k/p^k of the actual probability of k consecutive binary errors to the same probability for independent errors, with $k = 2, 3, 4$.

It is seen that binary errors occur in longer and longer bursts when S/N increases. Short bursts are however less frequent with precoding.

5 Interferences

This section is devoted to the study of the effect of several types of interferences and distortions on the performance of a Viterbi decoder.

In the following, we will define as an *additive interference* (or simply as an *interference*), anything which is linearly added to the signal, except noise. In general, interference is the result of a linear distortion in the transmission chain, but it can also come from *crosstalk* between different channels.

If $d(t)$ is the duobinary signal (*i.e.* the duobinary coded version of the data sequence) and if there is an interference $y(t)$, the input level without noise of the Viterbi decoder is

$$x(t) = d(t) + y(t), \quad (71)$$

while, in presence of noise, it is

$$x(t) = d(t) + y(t) + n(t). \quad (72)$$

5.1 Correlated or uncorrelated interference

As shown in Section 4.11, in presence of white noise, of which the successive samples are independent, the duobinary errors are also quasi-independent while the binary errors occur in bursts.

If the samples of the interference are uncorrelated, error-rates can be computed as done for the case of pure noise, simply by replacing noise by the sum of noise plus interference. The calculation method is again to define, by application of the duobinary tests, a system of simultaneous inequalities where n_i becomes $n_i + y_i$, and to derive the corresponding probability by numerical integrations, noting that the values of the interference will appear in the limits of integrations.

In most cases however, the interference samples are correlated and the *duobinary* errors will also occur in bursts. For example, let us consider the case where the wanted duobinary signal $d_1(t)$ is interfered by another sequence $d_2(t)$ which is:

- a duobinary sequence, independent of d_1 , or
- a ternary sequence taking the same levels $-1, 0, 1$ as a duobinary sequence with the same probabilities but without correlation.

In absence of noise, the probabilities of 2, 3 and 4 consecutive duobinary errors, obtained by simulation, are given in Table 5.

Cases	2 errors	3 errors	4 errors
$d_2 =$ correlated duobinary sequence	0.194	0.0757	0.0366
$d_2 =$ uncorrelated ternary sequence	0.123	0.0409	0.0146

Tab. 5: Probabilities of 2, 3 and 4 consecutive duobinary errors for an interference $y = d_2$.

It is seen that the probabilities of consecutive duobinary errors are significantly increased by the correlation of the interference, with two consequences:

- the Viterbi decoding is no longer optimal because the metrics in the trellis are no more strictly additive. However Viterbi decoding still gives a significant improvement with reference to threshold decoding.
- the theoretical error-rates can no longer be computed by the method of numerical integration of a system of inequalities. However this method is still valid as a first approximation and gives good results when the interference is low and is not strongly correlated.

5.2 General theoretical method in absence of noise

A general algorithm which should give the exact theoretical error-rates, involves the following steps:

1. Consider the duobinary initial transmitted sequences of length 3, 4, 5, ..., starting and ending by a symbol 1 or -1 .

For example

$$L = 3 \quad 10 - 1, -101, 111, -1 - 1 - 1$$

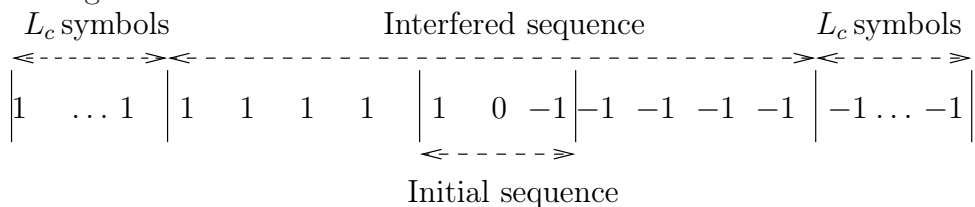
with probability of the intermediate symbol $1/8$

$$L = 4 \quad 1001, -100 - 1, 10 - 1 - 1, -1011$$

with probability of the intermediate symbols $1/16$

etc. The probabilities $1/8, 1/16, \dots$ quoted for the intermediate symbols come from the fact that these duobinary symbols are the coded form of binary sequences with length 4 for $L = 3$, of length 5 for $L = 4$, and so on. Since the input bits 1 and -1 have the same probability, the probability of a binary sequence of length 4 is $1/8$, of length 5 is $1/16, \dots$ Observe that for any length L , there are only four duobinary sequences starting and ending by 1 or -1 , so that the total probability of intermediate symbols in all given duobinary sequences is 1. Therefore all possible cases are considered.

2. In the case of noise (or uncorrelated interference), interference is applied to all symbols of these sequences but it was assumed that the first 1 (or -1) and the last 1 (or -1) are correctly recognized. Here we cannot avoid the case when there is a duobinary error on these symbols, as well as on the foregoing and the following symbols. We will then prolong the initial sequence to the left and to the right by known symbols 1 (or -1) in order to have an interfered sequence of length L_t . We will again prolong these sequences to the left and to the right by L_c symbols 1 (or -1) L_c being the decoding constraint length of the decoder. The total length of the transmitted sequence then becomes $L_t + 2L_c$ as illustrated hereafter for the first initial sequences of length 3.



3. Apply successively all possible values of the interference to the L_t symbols of the interfered sequence, but leave the first and the last L_c symbols free of interference.

4. Apply the duobinary tests given by (29) to (32) to the whole sequence to obtain the successive duobinary symbols r_i recognized by decoder. The presence of the first L_c non interfered symbols allows a correct initialization of this process, while the last L_c symbols allows complete termination of the decoding which is made with a delay of L_c symbols or bits.
5. Store and count all the cases where there is a duobinary error on the last but one symbol of the initial sequence. If there is no error on this symbol, iterate directly to the next value of the interference.
6. From the recognized duobinary symbols, reconstruct the survivors by using the rules of *copying* or *crossing* the survivors given in Section 4.
7. Go to the next step of Viterbi decoding and follow the evolution of the survivors.
8. Steps 3 to 7 above should be made with the actual values of the interference and also with no interference at all; in this second case there will be no duobinary nor binary errors, so that it is possible to construct two matrices (for the survivors S_1 and S_2) with interference and two similar matrices for S_1 and S_2 without interference and thus error free.
9. Compare the survivors matrices with and without errors and detect, then count the binary errors after L_c steps of the decoding (*i.e.* after convergence of the survivors).

The binary error-rate is finally the sum of all binary errors detected weighted by the probabilities of the values considered for the interference and also weighted by the probabilities of the intermediate symbols in the initial duobinary transmitted sequence.

In principle, this method should provide the exact answer if

- the length L of the initial sequences is 3, 4, \dots , ∞ .
- the length L_t of the interfered sequence is infinite.
- the constraint length (assuring complete convergence of the survivors) is infinite. In practice some truncation of this triple infinity is obviously needed and there should be some compromise between the precision required and the available memory as well as the time of computation. We adopted the figures $L = 3, 4, \dots, 8$, $L_t = 11$, $L_c = 10$, which give a good *second approximation* of the error-rates (our first approximation being the method of numerical integration).

5.3 General theoretical method in presence of interference and noise

The computation algorithm is the same as in the preceding Section, except that, at Step 3, it is necessary to combine each possible value of the interference with each possible value of the noise.

Suppose that with $L_t = 11$, we have $2^{11} = 2048$ values for the interfering sequences. If we sample the Gaussian distribution of noise into 100 values and store their probabilities, we will end up with 2048^{100} possible values of the total perturbation interference + noise. Such a number of cases is unmanageable. The only practical way is to replace all values or noise by a sufficiently high number of random sequences of Gaussian noise. We decided to generate 1000 noise sequences, which, as will be shown, gives the correct order of magnitude, but the final theoretical curves have then to be slightly smoothed.

For the same reason, when the number of values of the interference is too large, we will limit the number of simulations to 1000 random sequences of interference.

We will now consider the cases where the interference is the result of:

1. a phase error φ in a coherent demodulator of CQPRS modulation,
2. a synchronization error τ in the receiver, and
3. an echo of relative amplitude β , of phase ψ and of delay τ introduced in the radio-frequency path in CQPRS or AM/SSB modulations.

5.4 Interference resulting from a phase error φ in a CQPRS demodulation

5.4.1 Phase error without noise

If $d_1(t)$ and $d_2(t)$ are the two duobinary sequences modulating the carriers in quadrature and if there is a phase error in the first demodulator, the input level of the Viterbi decoder is

$$x(t) = d_1(t) \cos \varphi - d_2(t) \sin \varphi. \quad (73)$$

According to the definition (71), the interference is given by

$$y(t) = -d_1(t)(1 - \cos \varphi) - d_2(t) \sin \varphi. \quad (74)$$

There is therefore a mutual interference between the two independent duobinary sequences d_1 and d_2 vanishing for $\varphi = 0$.

For a given length L , there are 2^L possible values for the *interfering* sequence d_2 , given by duobinary coding of 2^L input bits ± 1 . Since the input bits are assumed to be independent and have the same probability, all the duobinary sequence d_2 have the same probability of $1/(2^L)$. For the case of $L_t = 11$, considered in our second approximation there are $2^{11} = 2048$ interfering sequences d_2 .

As explained above, a first approximation of the error-rates is obtained by numerical integration with the transmitted sequences d_1 used in Section 4.5 and all the 2^L values of d_2 .

A second theoretical approximation is obtained by the general method described in Section 5.2. Results of both approximations are given in Figure 11 for the binary error-rate, together with the results of simulation, for φ varying from 0 to $\pi/2$. The case of a phase error φ in CQPRS demodulation, without noise, with duobinary coding, and with Viterbi decoding (unmatched filtering), was chosen for Figure 11.

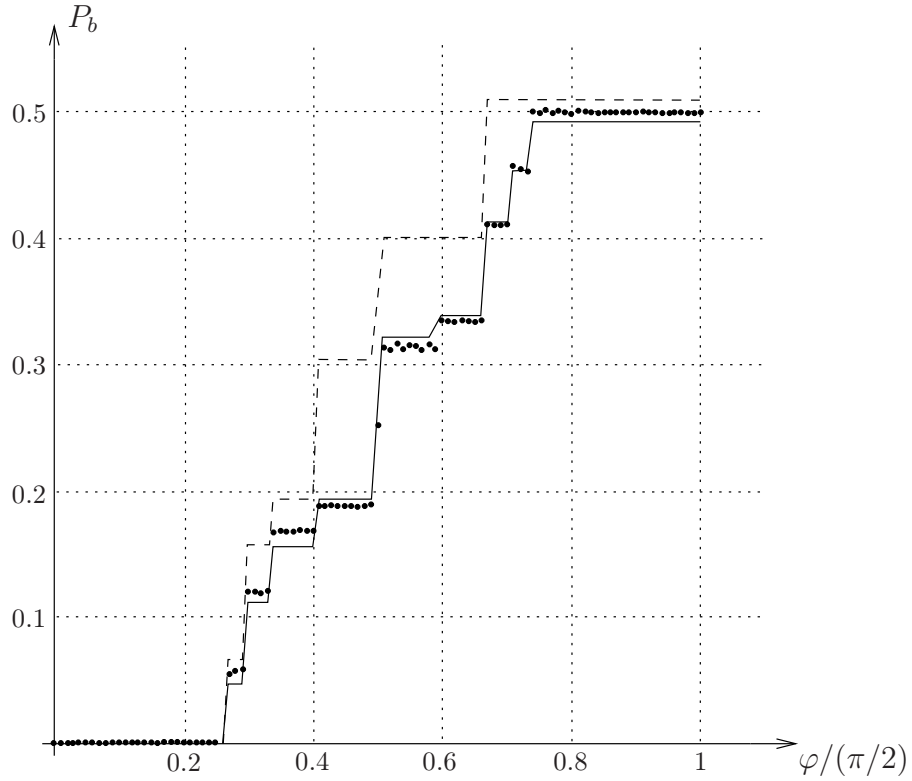


Fig. 11: Comparison of methods giving the binary error-rate with a phase error φ in CQPRS without noise: first approximation (dashed line), second approximation (full line), and simulations (points).

It can be seen that the first approximation is reasonably accurate for the duobinary error-rate and the survivors error-rate, even for large φ . For the binary error-rate we need the second approximation which takes account of the correlation of d_1 and d_2 by considering longer interfered sequences.

All the error-rates exhibit sudden variations for specific values of φ and then remain constant up to another *threshold* of variation. In particular the binary error-rate is strictly zero for $\varphi \leq 0.27 \pi/2$. These thresholds of variation correspond to limits in the duobinary tests. For example, test (29) can here be written as

$$(d_{1,i} + d_{1,i-k}) \cos \varphi - (d_{2,i} + d_{2,i-k}) \sin \varphi > 1 \quad \text{or} \quad < 0. \quad (75)$$

The limit 1 of this test is reached if

$$\varphi \approx 0.27 \frac{\pi}{2} \quad \text{or} \quad \cos \varphi - \sin \varphi = \frac{1}{2}, \quad (76)$$

and with $d_{1,i-k} = d_{1,i} = d_{2,i-k} = d_{2,i} = 1$.

Similarly the other thresholds of φ are found to be

$$\varphi \approx 0.30 \frac{\pi}{2} \quad \text{or} \quad \cos \varphi - 2 \sin \varphi = 0 \quad (77)$$

$$\varphi \approx 0.34 \frac{\pi}{2} \quad \text{or} \quad \sin \varphi = \frac{1}{2} \quad (78)$$

$$\varphi \approx 0.41 \frac{\pi}{2} \quad \text{or} \quad \cos \varphi + 2 \sin \varphi = 1 \quad (79)$$

$$\varphi \approx 0.5 \frac{\pi}{2} \quad \text{or} \quad \cos \varphi - \sin \varphi = 0 \quad (80)$$

$$\varphi \approx 0.6 \frac{\pi}{2} \quad \text{or} \quad -\cos \varphi + \sin \varphi = 1 \quad (81)$$

$$\varphi \approx 0.71 \frac{\pi}{2} \quad \text{or} \quad 2 \cos \varphi - \sin \varphi = 0 \quad (82)$$

$$\phi \approx 0.74 \frac{\pi}{2} \quad \text{or} \quad \cos \varphi - \sin \varphi = \frac{1}{2}. \quad (83)$$

5.4.2 Phase error with noise

Two examples of theoretical binary error-rates, compared with simulation are given in Figures 12 and 13, as functions of the ratio C/N_0 (carrier power to RF noise spectral density in CQPRS). Figure 12 assumes a phase error of $0.1 \pi/2$ (*i.e.* 9 degrees) while Figure 13 assumes a phase error of $0.28 \pi/2$, which is already above the first threshold of variation. Again, theoretical values obtained by the second approximation described in Section 5.3 are very close to the simulation results.

5.5 Interference resulting from a synchronization error τ

5.5.1 Synchronization error without noise

If, due to an improper synchronization recovery, the duobinary signal is sampled at times $kT + \tau$ instead of kT , the signal level at decoder input is

$$x(kT + \tau) = d(kT + \tau), \quad (84)$$

and there is an interference

$$y(kT + \tau) = -d(kT) + d(kT + \tau). \quad (85)$$

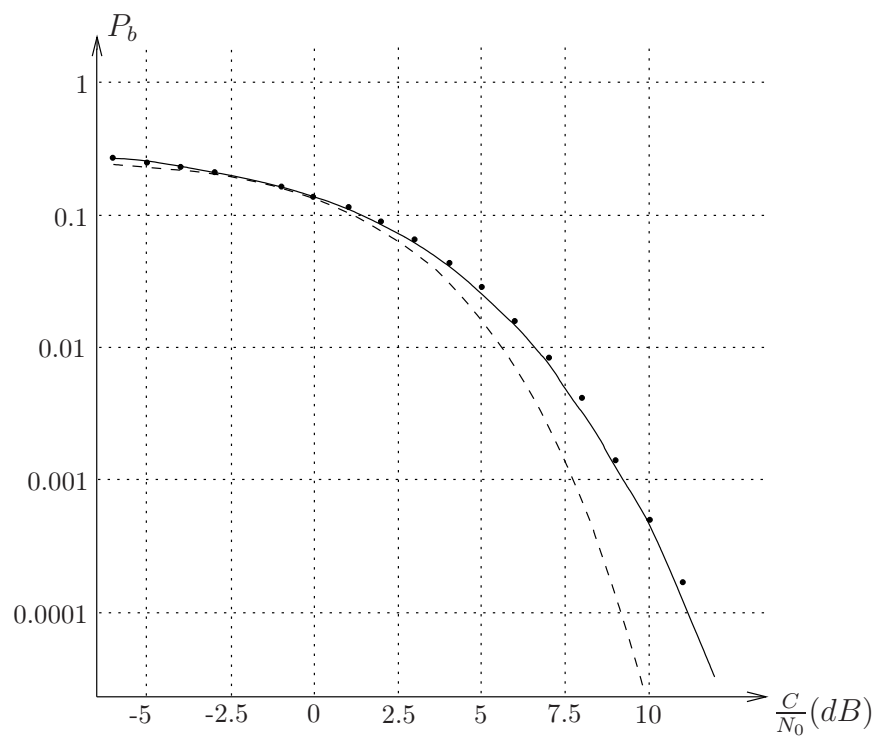


Fig. 12: Comparison of methods giving the binary error-rate with a phase error $\varphi = 0.1 \pi/2$ in CQPRS with noise as a function of C/N_0 : first approximation (dashed line), second approximation (full line), and simulations (points).

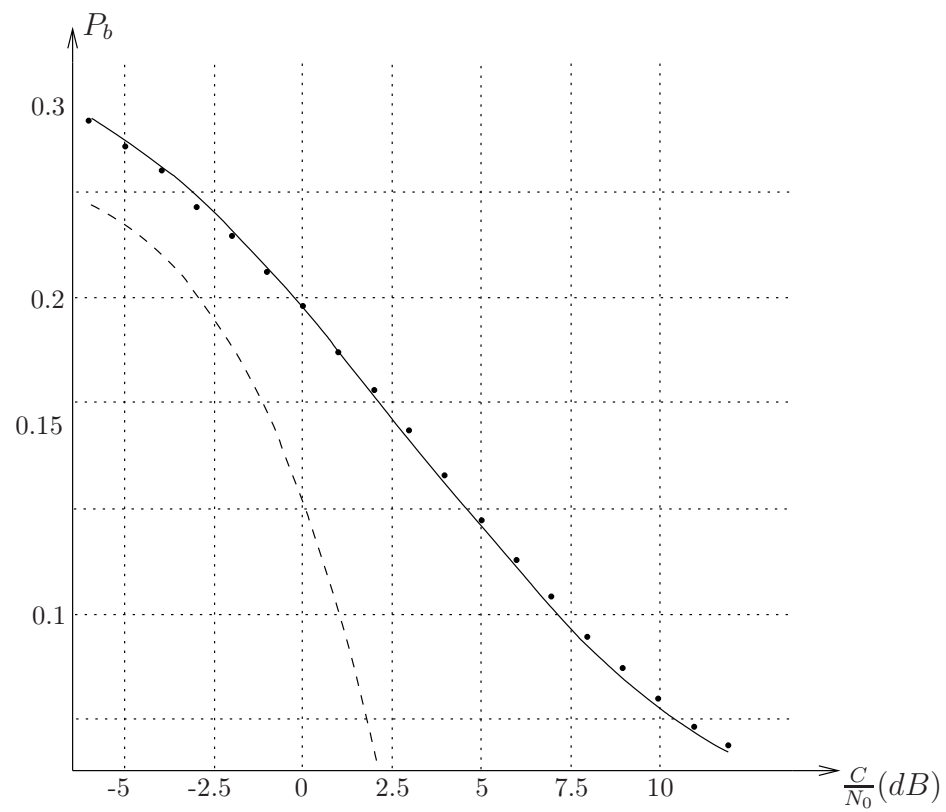


Fig. 13: Comparison of methods giving the binary error-rate with a phase error $\varphi = 0.28 \pi/2$ in CQPRS with noise as a function of C/N_0 : first approximation (dashed line), second approximation (full line), and simulations (points).

The calculation starts with the evaluation of the level of $x(kT + \tau)$. This problem has been solved by constructing a file of all possible levels to obtain the second approximation of the eye diagram (see Section 3.5.1). Depending on the transmitted sequence (including the duobinary symbols before and after $d(kT)$), the level x is simply read in the file at the appropriate address.

The error-rates can then be computed by the general algorithm of Section 5.2, but some care must be taken here with regards to the sign of τ . On the one hand, examination of the eye diagram shows a complete symmetry for a positive or negative value of τ . On the other hand, in theoretical calculation, one must remember that the duobinary errors on the *critical* symbol (last but one symbol of the initial transmitted sequence of length L) depend on duobinary symbols transmitted *before* the critical symbol, as expressed by the terms d_{i-k} in the duobinary tests. Therefore the theoretical calculation should be made with $\tau < 0$ but the result can be applied, by symmetry, for $\tau > 0$.

In a more detailed discussion of this problem, a distinction should be made between the duobinary errors on a symbol 0 and on a symbol 1 (or -1). It can also be shown by a discussion of the duobinary tests similar to that explained by equation (75) for a phase error, that the binary error-rate, without noise, is strictly zero for any value of $|\tau| < T/2$, *i.e.* half of the bit period. One can then already predict that Viterbi decoding will ensure a good robustness against synchronization errors.

The theoretical binary error-rate compared with simulation is given in Figure 14, where a bit period T is divided in 16 parts. It is seen that this error-rate is zero until $\tau = 8T/16$ and then jumps to $1/2$.

5.5.2 Synchronization error with noise

For a small value of τ (up to $2T/16$), the first approximation by numerical integration gives the correct order of magnitude of the binary error-rate. For larger τ , it is necessary to use the general algorithm of Section 5.3. One example of theoretical binary error-rates is given in Figure 15 for $\tau = 4T/16$. It can also be shown that for $\tau = 2T/16$ (*i.e.* $T/8$) the binary error-rate is not very much increased with reference to $\tau = 0$.

5.6 Interference resulting from an echo in CQPRS

5.6.1 Echo without noise

If $d_1(t)$ and $d_2(t)$ are again the two duobinary sequences modulating the two carriers in CQPRS and if, on the radio-frequency path there is an echo of relative amplitude β , of delay τ and of phase ψ , the input level at the Viterbi decoder is

$$x(t) = d_1(t) + \beta [d_1(t - \tau) \cos \psi - d_2(t - \tau) \sin \psi], \quad (86)$$

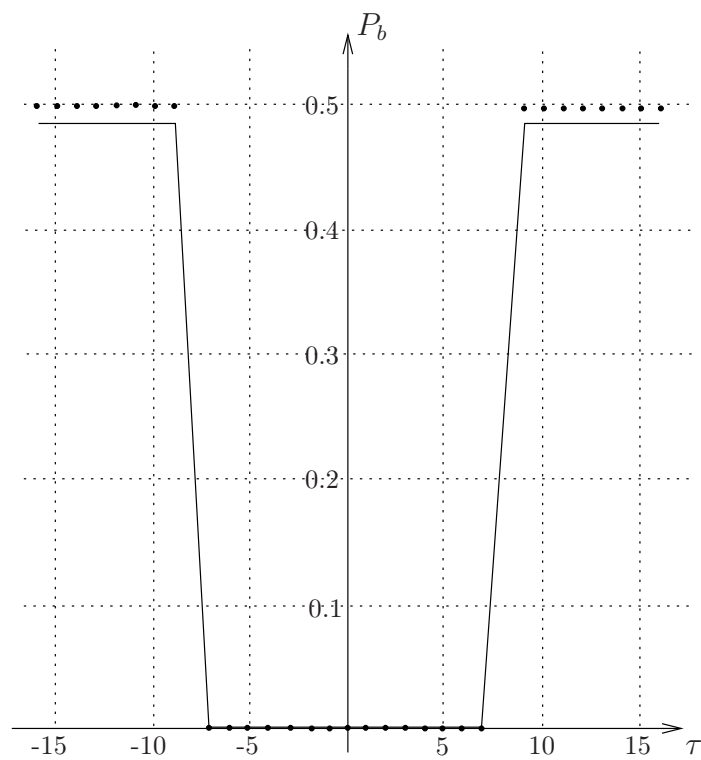


Fig. 14: Binary error-rate with a synchronization error τ in baseband and without noise: theoretical curve and points of simulation. Variation of τ : from $-15T/16$ to $15T/16$.

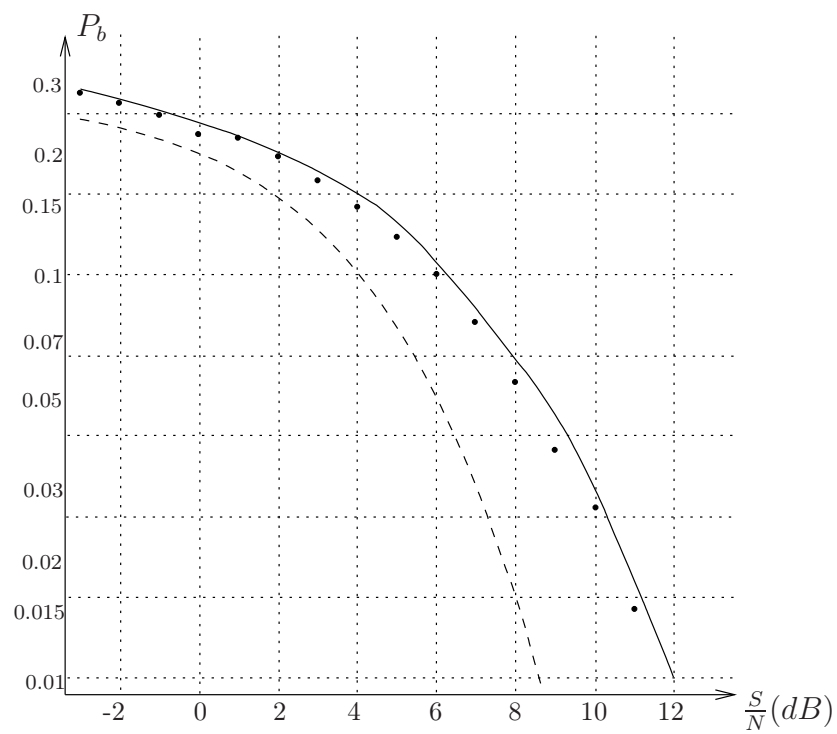


Fig. 15: Comparison of methods giving the binary error-rate with a synchronization error $\tau = 4T/16$ in baseband and with noise as a function of S/N : theory without synchronization error (dashed line), second theoretical approximation (full line), and simulations (points).

$d_1(t)$ being considered as the wanted signal. The interference is simply the echo signal

$$y(t) = \beta [d_1(t - \tau) \cos \psi - d_2(t - \tau) \sin \psi]. \quad (87)$$

The first duobinary test (29) is written as

$$d_{1,i} + d_{1,i-k} + y_i + y_{i-k} > 0 \quad \text{OR} \quad < 1, \quad (88)$$

with similar expressions for the three other tests. If β is zero or very small, there should be no binary errors at all, but as β becomes larger and larger there could be binary errors due to interference, even without noise. In order to find the minimum value m of β which gives binary errors, let us see when the test (75) reaches the limit 1. If τ is a multiple of T , this condition is fulfilled with $d_{1,i}(0) = 1$, $d_{1,i-k}(0) = 1$, $d_{1,i}(-kT) = -1$, $d_{1,i-k}(-kT) = -1$ and becomes

$$2 - 2\beta = 1, \quad (89)$$

so that

$$\beta_m = \frac{1}{2}. \quad (90)$$

If now τ is still a multiple of T but $\psi = \pi/4$ (or an odd multiple of $\pi/4$), we have

$$2 - 2\beta\sqrt{2} = 1 \quad (91)$$

$$\beta_m = \frac{1}{2\sqrt{2}} = 0.3535. \quad (92)$$

Finally, if we maximize the interference with respect to τ and ψ , we must have $\tau = (2k+1)T$ and $\psi = (2k+1)\pi/4$ and then, by taking the values of $d_1(t - \tau)$ and $d_2(t - \tau)$ in the file of levels (described in Section 3.5.1), we obtain

$$\beta_m = \frac{1}{2.414\sqrt{2}} = 0.293 \approx -10.7 \text{ dB}. \quad (93)$$

Therefore, in practice, there will be no binary errors at all (in absence of noise) if the echo is lower than -10.7 dB. This is confirmed by simulations.

5.6.2 Echo with noise

In the presence of noise, the error-rates are function of four parameters, namely the C/N_0 ratio and the echo parameters β , τ , ψ . Let us analyze this multidimensional surface.

Figure 16 is a cross-section of this surface corresponding to $\beta = 0.31$ and $C/N_0 = 12$ dB obtained by simulation. It is seen that the binary error-rate is maximum for very short echo delays ($\tau \leq 2T$) and for phases between π and $3\pi/2$. This could be expected because the first term of interference (87) is then strongly correlated with the direct signal. For longer delays ($\tau \geq 3T$), the cross-sections become practically flat and the bit-error rate

depends only slightly on τ and ψ . However if C/N_0 is high, or in other words if the effect of interference is bigger than the effect of noise, the binary error-rate tends to become a periodical function of τ , with period T with small variations, and also a periodical function of ψ , with period $\pi/4$ and somewhat bigger variations, the critical phases being odd multiples of $\pi/4$. This is clearly shown in Figure 16.

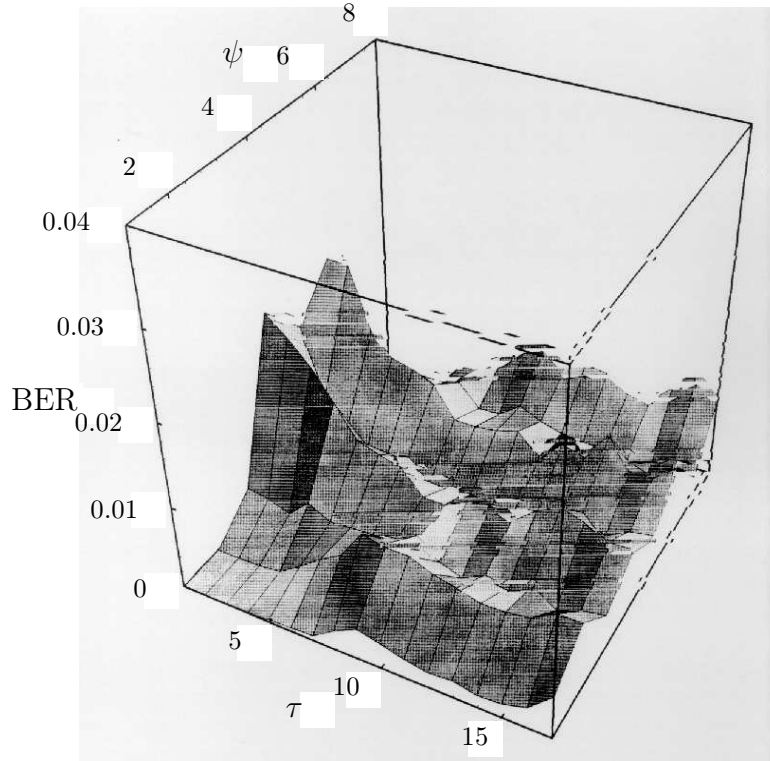


Fig. 16: Binary error-rate with an echo in CQPRS modulation as a function of echo delay τ and echo phase ψ : echo delay $\tau = 0$ to $4T$ i.e. $\tau = (x - 1)T/4$, echo phase $\psi = 0$ to $7\pi/4$ i.e. $\psi = (y - 1)\pi/4$, and echo amplitude $\beta = 0.31$, $C/N_0 = 12$ dB.

A first approximation of the theoretical binary error-rate can again be obtained by numerical integrations. The second approximation, with the general method of Section 5.3, gives better results, but as the error-rate does not depend very much on τ and ψ (for τ large), it is possible to define an *average* binary error-rate depending only on C/N_0 and β . Choosing for ψ the value $\pi/4$ makes this average slightly pessimistic. The results of the theoretical calculation (second approximation) is given in Figure 17 for $\beta = 0$ (no echo), 0.10, 0.15, 0.20, 0.25 and 0.31. This single Figure thus immediately gives the order of magnitude of the error-rate for a given echo amplitude, as a function of C/N_0 .

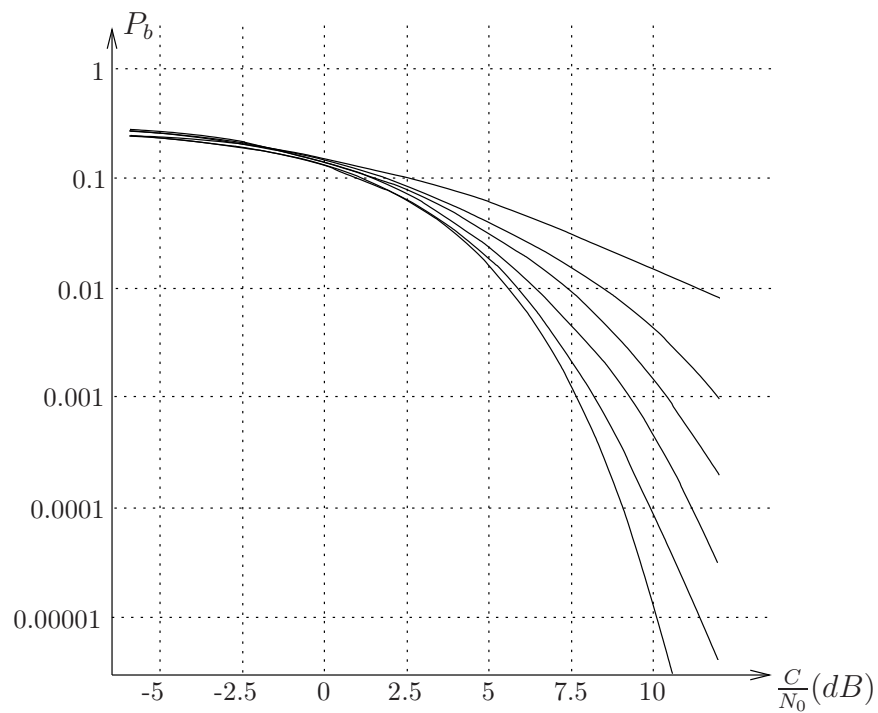


Fig. 17: Average binary error-rate with an echo in CQPRS modulation as a function of C/N_0 and echo amplitude β : $\beta = 0, 0.10, 0.15, 0.20, 0.25, 0.30$ (in order of increasing error-rate), echo phase $\psi = \pi/4$, and echo delay $\tau = \text{long and multiple of } T$.

5.7 Interference resulting from an echo in AM/SSB

5.7.1 Echo without noise

As continuous amplitude modulation with vestigial sideband (AM/VSB) is used for distribution of broadcasting signals over cable networks, we have considered AM/SSB for the transmission of a digital duobinary signal, as an approximation to AM/VSB.

If p and q are the amplitude of the lower and upper sidebands (p and q are functions of frequency), the input level at the decoder in presence of an echo is

$$x(t) = (p + q) [d(t) + \beta d(t - \tau) \cos \psi] + (p + q) [\beta d_q(t - \tau) \sin \psi], \quad (94)$$

where $d_q(t)$ is the signal in quadrature with $d(t)$.

The extreme cases are double sideband (AM/DSB) where $p = q = 1/2$ at all frequencies and single sideband (AM/SSB) where $p = 1$ and $q = 0$ (or $p = 0$ and $q = 1$). We will examine the case of AM/SSB which is close to AM/VSB with a simpler expression since

$$x(t) = d(t) + \beta d(t - \tau) \cos \psi + \beta d_q(t - \tau) \sin \psi. \quad (95)$$

We have explained in Section 3.5.1 that all possible levels of the signal $d_q(t)$ in quadrature with a duobinary signal $d(t)$ were also stored in a computer file. The procedure (reading the level at the appropriate address) can then be used for $d_q(t)$ as well as for $d(t)$.

The interference is given by

$$y(t) = \beta d(t - \tau) \cos \psi + \beta d_q(t - \tau) \sin \psi. \quad (96)$$

The first duobinary test (29) takes the form

$$d_i(kT) + d_{i-k}(kT) + \beta \cos \psi [d_i(kT - \tau) + d_{i-k}(kT - \tau)] + \beta \sin \psi [d_{i,q}(kT - \tau) + d_{i-k,q}(kT - \tau)] < 0 \quad \text{OR} \quad > 1. \quad (97)$$

The minimum value β_m of β which gives binary errors in absence of noise is found by the same method as used for a CQPRS echo, by expressing the condition where the test (97) reaches the limit 1.

If $\psi = 0$ or $\psi = \pi$, and if τ is a multiple of T , we obtain

$$\beta_m = \frac{1}{2}. \quad (98)$$

When ψ is still equal to 0 or π , but if we maximize the first term $d(kT - \tau)$ of the interference with respect to the delay τ , the file of the levels of d indicates that $\tau = (2k + 1)T/2$ with a level of 1.207, so that

$$\beta_m = 0.4142. \quad (99)$$

If now $\psi = \pi/2$ or $3\pi/2$, the interference is only due to d_q which has a maximum amplitude of 1.6677 for a delay τ multiple of T , so that

$$\beta_m = 0.2981. \quad (100)$$

For other cases, such as $\psi = (2k + 1)\pi/4$, the values of β_m are between (99) and (100). Therefore, in rounded figures, there will never be binary errors in absence of noise if

$$\beta < 0.2981 \approx -10.5 \text{ dB}. \quad (101)$$

5.7.2 Echo with noise

In the case of an echo with noise, the binary error-rate is a multidimensional surface function of four parameters, namely the S/N ratio and the echo characteristics β , τ , ψ .

Figure 18, obtained by simulation, is a cross-section of this surface corresponding to $\beta = 0.31$ and $S/N = 12$ dB. It is seen that the error-rate is very high for short delays ($\tau \leq 6T$), where not only $d(t - \tau)$ but also the quadrature component $d_q(t - \tau)$ are heavily intercorrelated with the direct signal. The critical phases are between π and $3\pi/2$. For longer delays, the cut becomes approximately flat and it is again possible to define an *average* binary error-rate, preferably with the most critical phase which is $3\pi/2$.

As was done for the echo in CQPRS, a first theoretical approximation of the binary error-rate can be obtained by numerical integrations, especially for low interference. A second theoretical approximation is obtained by the general method of Section 5.3 and is valid even for strong echoes.

Finally, Figure 19 gives the *average* binary error-rates for $\beta = 0, 0.1, 0.15, 0.20, 0.25$, and 0.31 .

As far as echoes with short delays are concerned (consider for example a system with a bit-rate of 10 Mbit/s), a delay of T corresponds to a path difference of 30 meters which is not unlikely on a cable network. Care must therefore be taken to prevent echoes with short delays in cable distribution.

5.8 Example of a non-linear distortion

5.8.1 General

The previous methodology can be extended to the case of a non-linear distortion of known characteristics. For the discussion, we have selected the example of a traveling-wave tube (TWT) having typical AM/AM and AM/PM characteristics. The TWT is considered as the high power amplifier for QPRS modulation with coherent demodulation, but for comparison we have also studied the case of the off-set modulation COQPRS.

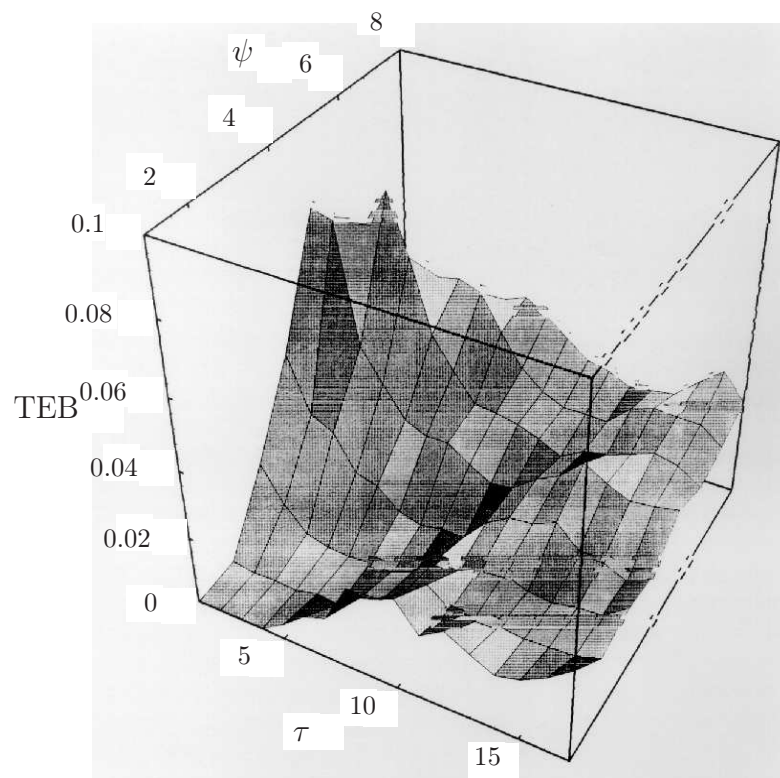


Fig. 18: Binary error-rate with an echo in AM/SSB modulation as a function of echo delay τ and echo phase ψ : echo delay $\tau = 0$ to $4T$ *i.e.* $\tau = (x - 1)T/4$, echo phase $\psi = 0$ to $7\pi/4$ *i.e.* $\psi = (y - 1)\pi/4$, and echo amplitude $\beta = 0.31$, $S/N = 12$ dB.

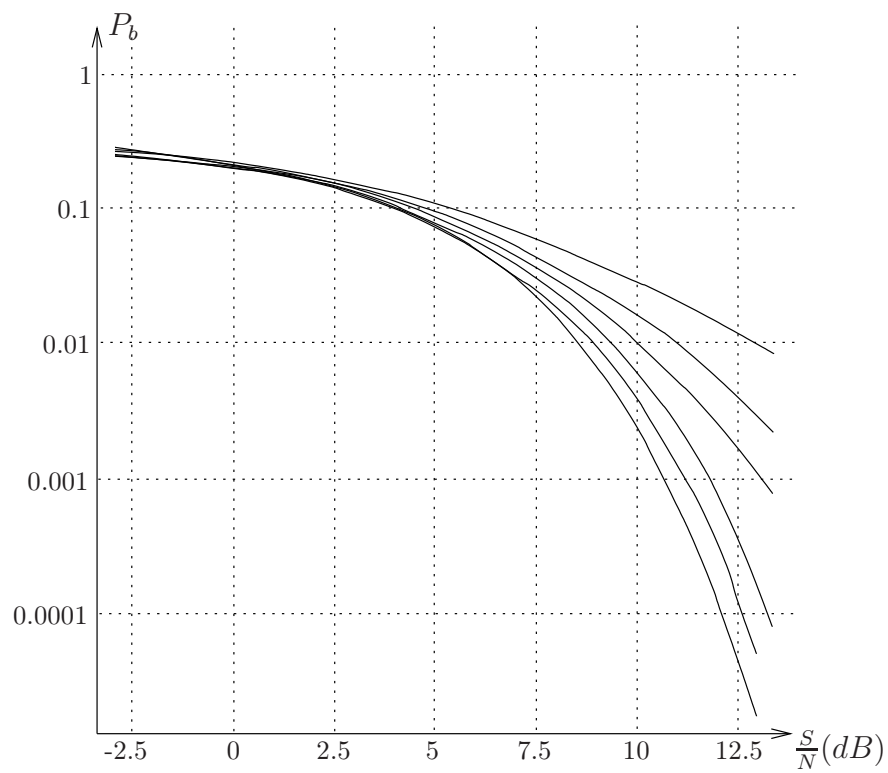


Fig. 19: Average binary error-rate with an echo in AM/SSB modulation as a function of S/N and echo amplitude β : $\beta = 0, 0.10, 0.15, 0.20, 0.25, 0.30$ (in order of increasing error-rate), echo phase $\psi = 3\pi/2$, echo delay $\tau =$ long and multiple of T .

5.8.2 Non-linearity model

In front of the non-linear device, the modulated signal can be written as

$$x(t) = \rho(t) e^{j\theta(t)}, \quad (102)$$

while after passing through the non-linear device, it becomes

$$z(t) = \rho_d(t) e^{j[\theta(t)+\varphi(t)]}. \quad (103)$$

When adopting the so-called ‘‘Saleh memoryless model’’, the AM/AM characteristic is

$$\rho_d(t) = A_s^2 \frac{\rho(t)}{\rho^2(t) + A_s^2}, \quad (104)$$

and the AM/PM characteristic is

$$\varphi(t) = \frac{\pi}{3} \frac{\rho^2(t)}{\rho^2(t) + A_s^2}, \quad (105)$$

where A_s is the TWT saturation level.

In the following, we select the value $A_s = 3$ which corresponds to a significant but not exaggerated distortion for a carrier amplitude $A_c = 1$. The AM/AM and AM/PM characteristics with $A_s = 3$ are reproduced in Figures 20 and 21.

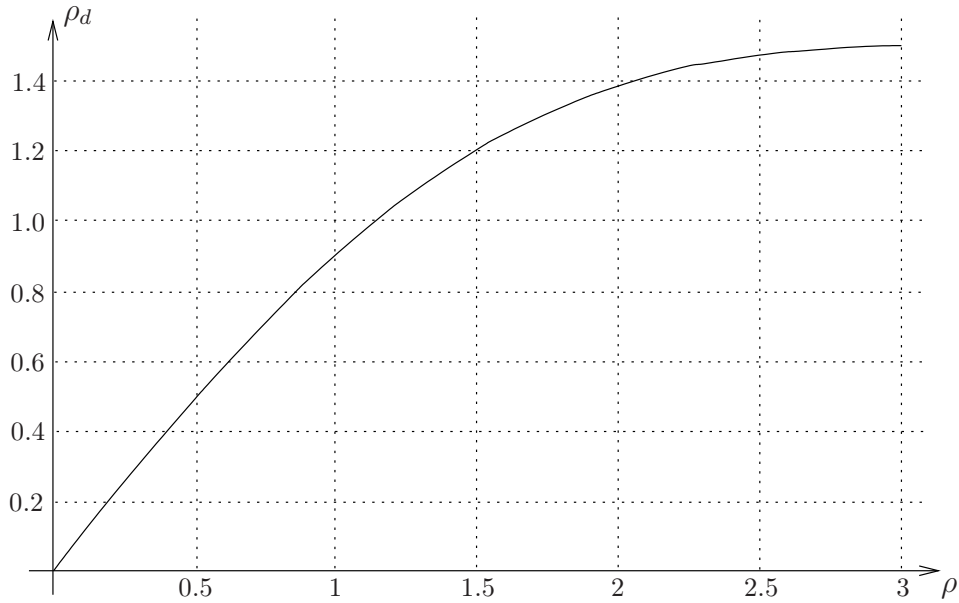


Fig. 20: AM/AM characteristics of the traveling-wave tube: saturation level $A_s = 3$.

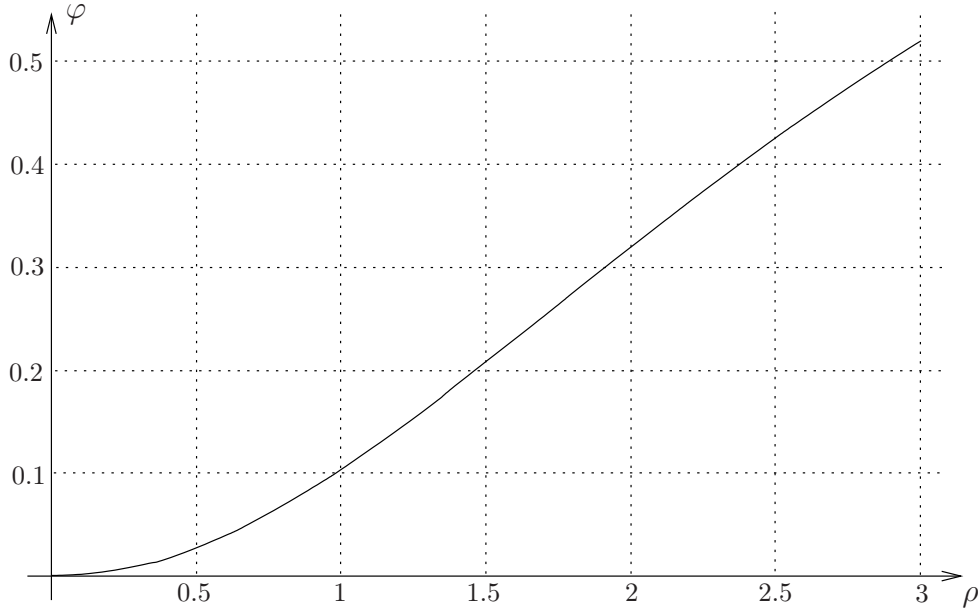


Fig. 21: AM/PM characteristics of the traveling-wave tube: saturation level $A_s = 3$.

5.8.3 Conventional definition of the back-off

The saturation level A_s is measured at the *input* of the TWT. According to the AM/AM characteristic of equation (104) and Figure 20, the maximum *output* amplitude is $A_s/2$ and the maximum *output* power is $A_s^2/4$. The usual definition of the backoff B_{off} is the ratio of the maximum output power to the actual output power

$$B_{\text{off}} = \frac{P_{\text{out max}}}{P_{\text{out}}} = \frac{A_s^2}{4 P_{\text{out}}}, \quad (106)$$

P_{out} being the mean value of $z^2(t)$.

In order to facilitate the calculations, we will adopt a slightly different definition. Since we consider only 4-state modulations, we note that the maximum output power on one channel, for example the channel corresponding to the real part of $x(t)$ and $z(t)$, is $A_s^2/8$. Then if C is the output power that would be obtained on one channel in the *absence of distortion* (*i.e.* with a *linear* TWT of power gain equal to 1 and a carrier amplitude $A_c = 1$), we define the backoff as the ratio

$$B_{\text{off}} = \frac{P_{\text{out max (one channel)}}}{C} = \frac{A_s^2}{8C}. \quad (107)$$

We will compute the error-ratio without distortion (linear operation) and with distortion (non-linear operation) as a function of the same parameter C/N_0 and assess the penalty due to the distortion in terms of C/N_0 .

In our tests, we have used the following numerical values of the parameters for CQPRS and unmatched filtering

$$C = \frac{A_c^2}{4} = 0.25, \quad (108)$$

$$B_{\text{off}} = 6.53 \text{ dB}, \quad (109)$$

and a noise variance σ^2 computed by

$$\frac{1}{\sigma^2} = 2 \frac{C}{N_0} T_s. \quad (110)$$

5.8.4 Note on band-pass filtering

In principle a band-pass filter must be inserted at the input and another band-pass filter at the output of the TWT. However, for CQPRS, there are no spectral components outside the RF band

$$B = f_b, \quad (111)$$

so that the first band-pass filter is useless.

In all cases, the non-linearity introduces a certain amount of spectrum spreading. It can however be shown that, with the parameters used for our tests, the components outside the band B are very small (about 1% relative to the centre of the band). We will therefore neglect also the second band-pass filtering.

5.8.5 Method of calculation

For each particular modulation, the first step is to write the series of values taken by the real and imaginary parts $x_r(t)$ and $x_i(t)$ of the modulated signal at the sampling instants kT which are multiples of the symbol period T if $t = 0$ corresponds to the middle of one symbol. These values are then weighted with their respective probabilities.

The second step is to compute the values of the envelope $\rho(kT)$ and the phase $\theta(kT)$ at the sampling instants, before distortion. The next step is to apply equations (104) and (105) to obtain the envelope $\rho_d(kT)$ and the phase $\theta(kT) + \varphi(kT)$, after distortion.

The final step is to compute the real part $z_r(kT)$ of the distorted signal by

$$z_r(kT) = \rho_d(kT) \cos[\theta(kT) + \varphi(kT)]. \quad (112)$$

The difference

$$y_r(kT) = z_r(kT) - x_r(kT) \quad (113)$$

can then be considered as an additive interference and we can derive the binary error-rate on the *real* part of the channel by using the methods given above. Since there is a symmetry

between the *real* part of the channel $z_r(t)$ and the *imaginary* part of the channel $z_i(t)$, the error-rate is the same on both channels. If needed, $z_i(kT)$ can be computed by

$$z_i(kT) = \rho_d(kT) \sin [\theta(kT) + \varphi(kT)] , \quad (114)$$

with an interference

$$y_i(kT) = z_i(kT) - x_i(kT) . \quad (115)$$

5.9 CQPRS modulation with Viterbi decoding, no precoding and unmatched filtering

We have already seen that the Viterbi decoding, although no longer optimal in presence of interference, still gives a good improvement over threshold decoding. This is also the case in presence of a non-linear distortion.

Two general methods were presented to compute the binary error-rate. Both methods involve:

1. the calculations of the series of transmitted duobinary sequences of length $L = 3, 4, 5, \dots$, starting and finishing by a duobinary symbol 1 or -1 ,
2. the application of the duobinary tests in order to detect duobinary errors on the last but one symbol and,
3. computation of the consequential binary errors.

In the first method, this is done by numerical integrations to obtain the probability of a system of simultaneous inequalities which contains the successive noise samples.

This method does not take account of the correlation of the interference and thus gives only a first approximation of the final error-rate, if, as it is the case in CQPRS, the interference is coming from a independent duobinary sequence. Note that the values of interference (or distortion) here appear in the limits of the integrations.

We will only present here the results with the second method which gives a better approximation. In this second method, the transmitted sequence is prolonged by known symbols and, after the duobinary tests, the survivors are reconstructed so that the binary errors are detected by following their evolution. Account is taken here of the correlation of interference.

We again consider all duobinary transmitted sequences of length $L = 3, \dots, 8$ starting and finishing by 1 or -1 , but these sequences are prolonged on each side by known symbols 1 or -1 up to length $L_t = 11$. For each transmitted sequence, we apply all possible interfering sequences (*i.e.* $2^{11} = 2048$) to the 11 symbols L_t as well as a sufficient number or random noise sequences. In addition, on each side of the $L_t = 11$ symbols, the sequence

is prolonged by L_c known symbols 1 or -1 , but without adding noise or interference. This will allow the correct initialization and termination of the series of duobinary tests. All the duobinary symbols are then correctly recognized and the survivors can be reconstructed by the *recopying* rule at each step of the decoding.

The evolution of the survivors is then followed, once with noise and distortion and once without noise or distortion. The comparison of the two series of survivors gives the correct number of binary errors in the sense that we count the number of binary errors due to duobinary errors on the last but one symbol of the initial transmitted sequence. In each case, the interference $y(i, j)$ is computed according to a relation similar to (113) in which z_r is written $z_r(i, j)$ where i is the iteration on the wanted sequence and j the iteration on all possible duobinary interfering sequences.

As explained previously, this method is exact, provided that the lengths L , L_t and L_c are infinite and that an infinite number of noise sequences is used. In practice truncation is necessary to limit the computation time, so that the method gives what we call a second approximation.

Our results are drawn in Figure 22 (the saturation level A_s was chosen equal to 3). This figure shows a good agreement between theory and simulation.

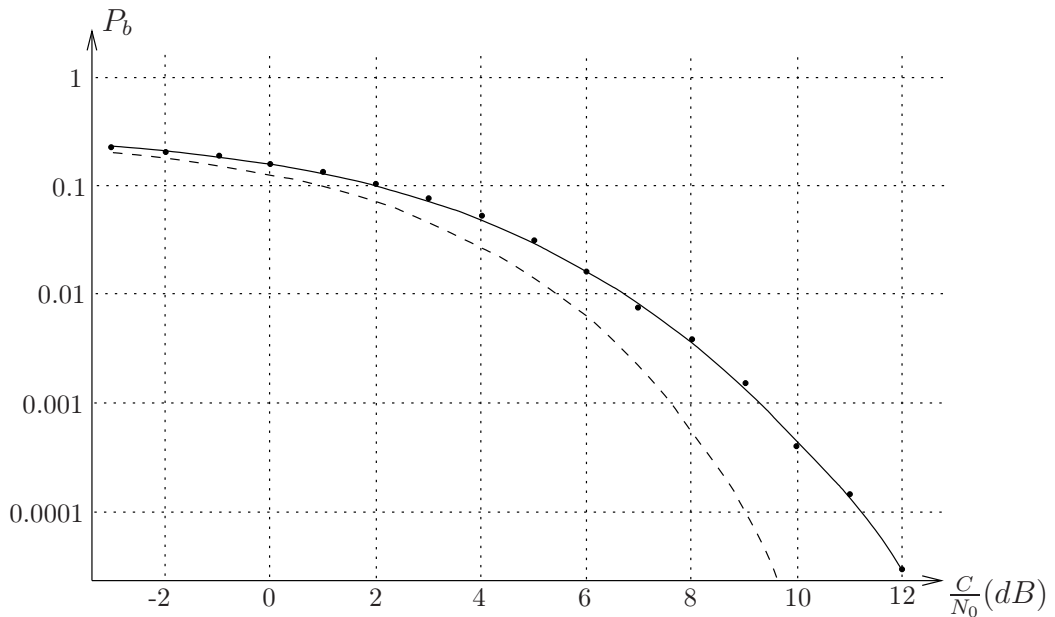


Fig. 22: Comparison of methods giving the binary error-rate for CQPRS with a non-linear distortion in a traveling-wave tube, as a function of C/N_0 : theory without distortion (dashed line), second theoretical approximation by recalculation of the survivors (full line), and simulations (points).

5.10 COQPRS modulation with Viterbi decoding, no precoding and unmatched filtering

For this modulation, where one of the two duobinary streams is off-set by $T_s/2$, we will only use the method described as the second approximation by reconstruction of the survivors. The calculation is similar to that of the previous section, except that $x_i(kT_s)$ can now take a number of values corresponding to the possible level of the duobinary eye diagram at half of a symbol period from the sampling instant. These values depend on the whole duobinary sequence, which is here the interfering sequence.

We have used the reconstructed eye diagram where all the levels were computed (by summation of a large number of impulse responses) and stored in a computer file, for all duobinary sequences of length 9.

To obtain the value of $x_i(kT_s)$ for a given interfering sequence, one has to search the sequence of length 9 identical on 9 symbols to the interfering sequence, the symbol of order i of this sequence corresponding to the middle symbol (of order 5) of the sequence of length 9. The level of the eye diagram at $T_s/2$ is then read in the file, at the appropriate address, and divided by $\sqrt{2}$ to obtain $x_i(kT_s)$. The values of the duobinary symbols recognized by the decoder $z_r(kT_s)$, of the interferences $y(i, j)$, the survivors with and without distortion and finally the binary errors are then obtained as for CQPRS, but with a larger number of calculations and a longer computation time.

The results are given in Figure 23 where it is seen that there is, again, good agreement between theory and simulation.

In terms of C/N_0 , COQPRS seems to be slightly better than CQPRS since the degradation due to the non-linear distortion is 1.7 dB for the first, and 2.1 dB for the second one at a BER of 10^{-4} . Moreover for these two modulations, Viterbi decoding still provides an improvement of about 1.5 to 2 dB over threshold decoding.

6 Conclusions

The duobinary code, which is the subject of the present study, is used in a number of communications and broadcasting systems. The binary error-rate (BER) is normally evaluated by simulation with pseudo-random binary sequences as input data. In all cases, decoding of the duobinary code with the Viterbi algorithm provides a significant improvement with respect to the simple threshold decoding.

With Viterbi decoding, the theoretical value of the BER is often not known, specially for channels strongly impaired by noise and linear or non-linear distortions. In the previous Sections, several numerical methods usable on a simple computer were presented for the calculation of the theoretical BER, in presence of noise and typical linear or non-linear distortions. In all cases, the results obtained were in good agreement with those of simulation.

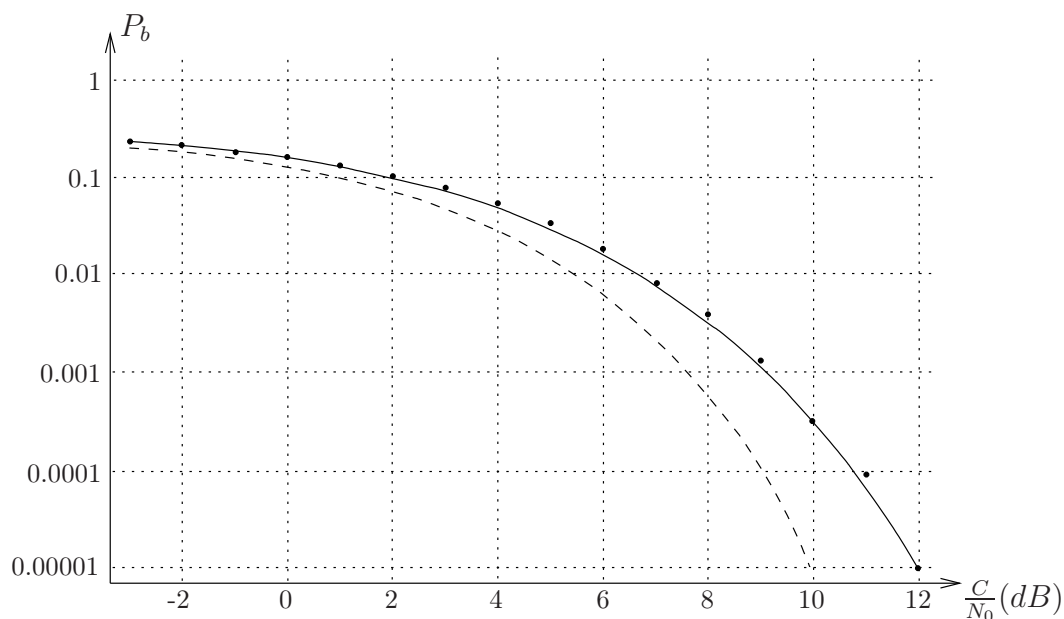


Fig. 23: Comparison of methods giving the binary error-rate for COQPRS with a non-linear distortion in a traveling-wave tube, as a function of C/N_0 : theory without distortion (dashed line), second theoretical approximation by recalculation of the survivors (full line), and simulations (points).

Special note

This paper is published in memory of Professor Henri Mertens. Henri Mertens was Associate Professor Emeritus at the University of Liège (Liège), Belgium, and also former Deputy Director of the Technical Center of the European Broadcasting Union (EBU), Brussels, Belgium.

List of acronyms

ACS	Add-Compare-Select
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
CCIR	Comité consultatif international pour la radio (now ITU-R)
CMOS	Complementary Metal-Oxide Semiconductor
CQPRS	Continuous Quadrature Partial Response Signaling modulation
EBU	European Broadcasting Union

FM	Frquency Modulation
ISI	Inter-Symbol Interference
ITU	International Telecommunication Union
OFDM	Orthogonal Frequency Division Multiplexing
PAM	Phase-Amplitde modulation
QAM	Quadrature Amplitude Modulation
QPRS	Quadrature Partial-Response Signaling
QPSK	Quadrature Phase Shift Keying
RF	Radio Frequency
SSB	Single-Sideband modulation
TWT	Traveling-Wave Tube
VSF	Vestigial Sideband modulation

References

- [1] Shirish A. Altekar, Mganus Berggren, Bruce E. Moision, Paul H. Siegel, and Jack K. Wolf. Error-event characterization on partial-response channels. *IEEE Transactions on Information Theory*, 45(1):241–247, January 1999. URL: <https://doi.org/10.1109/18.746796>.
- [2] Claude Berrou and Alain Glavieux. *Turbo codes: general principles and applications*, pages 215–226. Elsevier, 1994.
- [3] Martin Bossert. *Channel Coding for Telecommunications*. John Wiley & Sons, 1999.
- [4] CCIR. Recommendation 650-1. Television standards for satellite broadcasting in the channels defined by the WARC BS-77 and the RARC SAT-83, 1990.
- [5] CCIR. Report 1073 (mainly relates to vision signals), 1990.
- [6] CCIR. Report 632 (mainly relates to sound and data signals), 1990.
- [7] Yun-Nan Chang. An efficient in-place VLSI architecture for Viterbi algorithm. *Journal of VLSI Signal Processing*, 33:317–324, March 2003. URL: <https://doi.org/10.1023/a:1022246815354>.
- [8] EBU. Tech. doc. 3268-E. Specification of the systems of the MAC/packet family, 1986.

-
- [9] G. David Forney. Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference. *IEEE Transactions on Information Theory*, 18(3):363–378, May 1972. URL: <https://doi.org/10.1109/TIT.1972.1054829>.
- [10] G. David Forney. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, March 1973. URL: <https://doi.org/10.1109/PROC.1973.9030>.
- [11] ITU. BO-787. MAC/packet based system for HDTV broadcasting-satellite services, 1992.
- [12] ITU. BO-1074-1. Satellite transmission of multiplexed analogue component (MAC) vision signals, 2003.
- [13] Hisashi Kobayashi. Correlative level coding and maximum-likelihood decoding. *IEEE Transactions on Information Theory*, 17(5):586–594, September 1971. URL: <https://doi.org/10.1109/TIT.1971.1054689>.
- [14] Ernest Kretzmer. Generalization of a technique for binary data communication. *IEEE Transactions on Communications*, 14(1):67–68, February 1966. URL: <https://doi.org/10.1109/TCOM.1966.1089288>.
- [15] Adam Lender. The duobinary technique for high-speed data transmission. *IEEE Transactions on Communication and Electronics*, 82(2):214–218, May 1963. URL: <https://doi.org/10.1109/TCE.1963.6373379>.
- [16] Michael. Leung, Borivoje Nikolic, Leo Ki-Chun Fu, and Taehyun Jeon. Reduced complexity sequence detection for high-order partial response channels. *IEEE Journal on Selected Areas in Communications*, 19(4):649–661, April 2001. URL: <https://doi.org/10.1109/49.920173>.
- [17] Henri Mertens. Echo sensitivity of partial-response codes in amplitude modulation with vestigial sideband. *EBU Review Technical*, 237/238, 1989.
- [18] CCIR Publication. Specifications of transmission systems for the broadcasting-satellite service, 1988.
- [19] Bernard Sklar. *Digital communications: fundamentals and applications*. Prentice Hall, 1988.
- [20] James J. Spilker. *Digital Communication by Satellite*. Prentice Hall, 1977.
- [21] Tjeng T. Tjhung, K. J. Tan, and L. K. Ho. Error performance analysis for narrow-band duobinary FM with discriminator detection and soft decision decoding. *IEEE Transactions on Communications*, 37(11):1222–1228, November 1989. URL: <https://doi.org/10.1109/26.46517>.

-
- [22] Bartolomeu F. Uchiôa Filho, Mark A. Herro, and Daniel J. Costello. A multilevel approach to constructing trellis-matched codes for binary-input partial-response channels. *IEEE Transactions on Information Theory*, 45(7):2582–2591, November 1999. URL: <https://doi.org/10.1109/18.796410>.
- [23] UER. Doc. GTR5 101. Démodulation d’un signal duobinaire selon l’algorithme de Viterbi. Original work by M. Allard. Démodulation d’un signal duobinaire selon l’algorithme de Viterbi, Document of the CCETT, France, MDD/138/88/MN, March 1986.
- [24] Andrew Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, April 1967. URL: <https://doi.org/10.1109/TIT.1967.1054010>.
- [25] Andrew Viterbi. Convolutional codes and their performance in communication systems. *IEEE Transactions on Communication Technology*, 19(5):751–772, October 1971. URL: <https://doi.org/10.1109/TCOM.1971.1090700>.
- [26] Andrei Vityaev and Paul H. Siegel. On Viterbi detector path metric differences. *IEEE Transactions on Communications*, 46(12):1549–1554, December 1998. URL: <https://doi.org/10.1109/26.737388>.
- [27] Bahram Zand and David A. Johns. High-speed CMOS analog Viterbi detector for 4-PAM partial-response signaling. *IEEE Journal of Solid-State Circuits*, 37(7):895–903, July 2002. URL: <https://doi.org/10.1109/JSSC.2002.1015688>.