TESE DE MESTRADO

TÉCNICAS DE DECISÃO SUAVE

Hélio Magalhãe.6 de. Olíve,ina.

3fo» *"

COORDENAÇÃO DO MESTRADO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ELETRÔNICA E SISTEMAS
UNIVERSIDADE FEDERAL DE PERNAMBUCO
CIDADE UNIVERSITÁRIA
RECIFE - BRASIL

- 1983 -

Aos meus Pais,

Nilson e Djanira.

AGRADECIMENTOS

Gostaria de expressar meus sinceros agradecimentos a todos que prestaram sua colaboração na elaboração deste trabalho.

A todos os professores do curso de Mestrado em Engenharia Elétrica, em especial aos professores Marco Wolfenson, Fernando Campeilo e Luiz Torres.

Aos colegas Marcos Antonio Martins, Clóvis P« Florêncio (entre outros), pelas suas contribuições.

A Ricardo Campello e Fernando Casanova pelo auxilio na parte computacional e sugestões valiosas.

A senhora Rosa Maria Alves Neves, pela cuidadosa datilografia dos originais manuscritos.

Ao professor Valdemar Cardoso da Rocha Júnior, por introduzir-me neste campo de pesquisa, por sua influência na minha filosofia, sua orientação, incentivo e amizade.

RESUMO

Nesta tese sao apresentados e analisados alguns recentes resultados obtidos na ãrea de códigos corretores de erros, relacionados com decodificaçao utilizando decisão suave. Tal procedimento é usado com o fim de evitar a degradação no desempenho de sistemas codificados que resulta quando na recepção uma quantização abrupta precede a decodificação.

No capítulo I ê apresentado um tratamento introdutório sobre códigos lineares, estudando códigos de bloco e códigos convolucionais. Em seguida, no capítulo II, é abordado o conceito de decisão suave e é apresentada uma maneira de utilizã-la, adaptando técnicas algébricas de decodificação por distância mínima a uma medida de distância que permite o uso de informação de verossimilhança. Alguns procedimentos subõtimos que utilizam decisão suave são também apresentados. No capítulo III são estudados dois algoritmos ótimos que fazem uso de tais técnicas na decodificação de códigos lineares em canais sem memória.O desempenho destes algoritmos é analisado através de simulação em computador digital, no capítulo' IV, para vários códigos. Os resultados obtidos são comparados com procedimentos que empregam decisão abrupta. Finalmente, no capítulo V, comentários gerais são feitos discutindo vantagens e restrições decorrentes do uso destas técnicas, assim como sugestões para investigações futuras.

ABSTRACT

In this thesis, some recent results obtained in the area of error correcting codes, related to soft decision decoding techniques, are presented and analyzed. Such procedure is used in order to avoid the degradation in the performance of encoded systems which results when in the reception a hard decision proceedes the decoding.

In chapter I an introductory treatment of linear codes is presented, studying block codes and convolutional codes. Then, in chapter II, the concept of soft decision is approached, and a way of using it is presented, adapting algebraic techniques of minimum distance decoding to a distance measure which permits the use of likelihood information. Some suboptimal procedures which use soft decision are also presented.

In chapter III two algorithms with optimal properties are studied, which make use of such techniques in the decoding of linear codes in memoryless channels. The performance of these algorithms is analyzed via digital computer simulation, in chapter IV, for several codes. The results obtained are compared with procedures which use hard decision. Finally, in chapter V, general comments are made discussing the advantages and restrictions implied by the use of these techniques, as well as suggestions for further investigations.

Í N D I C E

ÍNDICE DE FIGURAS

LISTA DE SÍMBOLOS E ABREVIATURAS

A Amplitude de um pulso binário

AByb+O Comportamento assintótico para baixa relação sinal-ruído

AB^k+oo Comportamento assintótico para alta relação sinal-ruído

ASK Modulação por chaveamento de amplitude

BCH Bose-Chaudhuri-Hocquenghem

BEC Canal binário com apagamento

BSC Canal binário simétrico

c Palavra código, ou sequência código

c* = n.'/i! (n-i) !

C Código de bloco

C Código dual do código linear C

d Distância mínima de um código de bloco

d_S Distância suave

D_(r,f)Distância generalizada de Forney
o – –

$D_н$(f,g)Distância de Hamming entre duas énuplas

-m Om' lm n-lm

f Palavra código de um código de bloco

fdP Função densidade de probabilidade

f(r|c) Função de verossimilhança de um canal

2 Palavra código de um código de bloco

g(i) Geradores de um código convolucional

g(i,j) Subgeradores de um código convolucional

g(s) Função geradora de momentos

g(x) Polinómio gerador de um código cíclico

[G]'00 Matriz geratriz de um código linear

GF(q)	Campo de Galois com q elementos
h(x)	Polinómio de verificação de paridade de um código cíclico
[H], *H*.	Matriz de verificação de paridade de um código linear
HF	Alta frequência
i i d	Independentes e identicamente distribuídas
I_k	Matriz identidade kxk
IQ( )	Função de Bessel modificada de ordem zero(primeira espécie)
J	Amplitude da zona nula; numero total de classes de confiabilidade
k	Números de dígitos de informação; profundidade em uma treliça
Lj	*log* razão-de verossimilhança na classe de confiabilidade c.
(m)	Bloco de mensagem
m(X)	Polinómio mensagem
n	Comprimento de um código de bloco
ndc	Nao decodificar corretamente
N	Potência do ruído
N(d)	Número de palavras código com peso d
NP	Neymann-Pearson
PQ,P^,...	Probabilidades a priori dos símbolos da fonte
P( ),Pr( )	Probabilidade de um evento
$p\{decj\|n_s, j_s\}$	Probabilidade de decodificar j_s dado que II_{n-k} foi observado e j_s foi transmitido
P^	Subconjunto de {0,l,...,q -1}
PLL	Phase lock loop
2^	Número de níveis de quantização
Q(x) = l//2nf exp(-t /2)dt	
r	Palavra recebida
rem	Resto de uma divisão

r(X)	Polinómio recebido
R	Taxa (eficiência) de um código corretor de erros
R_{gi}	Coeficiente de confiabilidade condicional de Kiefer
IR	Conjunto dos números reais
RQ	Espaço das possíveis regiões de quantização
R^	Espaço dos possíveis valores do £og verossimilhança
S_	Síndrome
So(k)	£-ésimo nó na profundidade k
S;s	Conjunto de índices; número de apagamentos
SNR	Relação sinal ruído
t	Capacidade de correção de um código
u,v	Énuplas sobre GF(q)
V"$_n$	Espaço vetorial das énuplas sobre GF(q)
W(u)	Peso da énupla u
X	Estimador de X
X	Énupla
(X),[x]	Matriz X
$(X)^T$,$[x]^T$	Transposta de [x]
[xj	Parte inteira de X
Z(e)	*iog* da função de verossimilhança f (r |c)
a	Probabilidade de erro tipo I
a-a $_{D}$ s	Peso de uma classe de confiabilidade
6	Probabilidade de erro tipo II
Y/Y_b	Relação sinal-ruído de um canal ruidoso
ô.	Delta de Kronecker
0 = exp^n/^/P)	
X	Constante
\i(s)	Função geradora cumulativa
n	Partição do espaço de observações
p = (1-0)/(1+0)	
o	Tensão eficaz do ruído

Razão de verossimilhança para a amostra r.

Θ_{ij} Razão de verossimilhança para a classe n_s dado

$\sum$ Somatório módulo-q

Soma módulo-2

t

$p($ Distribuição normal

Distribuição binomial

CAPÍTULO I

INTRODUÇÃO AOS CÓDIGOS LINEARES

1.1 - INTRODUÇÃO

Sistemas de comunicação são essencialmente usados para transmissão de informação, e em geral são constituídos por um transmissor, um meio de transmissão através do qual a informação é enviada e um receptor. Ao bloco transmissor está conectada a fonte de informação a ser transmitida - sinais de voz, sinais de TV, dados de saída de computadores, dados de telemetria de naves de exploração espacial ou dados de telemetria transmitidos de uma instalação operada automaticamente para uma estação central de controle.

Ã medida que os sinais atravessam o meio de transmissão (canal), eles são distorcidos; ruídos e interferências a eles se somam e se torna uma tarefa importante interpretar os sinais quando finalmente recebidos no extremo receptor.

O problema básico em um sistema de comunicação se resume em transmitir a informação através do canal ruidoso, de uma maneira confiável.

O problema da transmissão confiável de informação tem representado um desafio constante para engenheiros e pesquisadores' em comunicações. Aqui, a confiabilidade diz respeito *a.* imunidade da transmissão a ação do ruído e outras interferências, e não a indec_i

frabilidade de mensagens (criptografia).

Nas últimas décadas tem havido um grande crescimento na utilização de sistemas de comunicação digital, i.e., sistemas onde a mensagem já está na forma digital ou é convertida para este formato, como no caso dos sistemas PCM/TDM]33]. Um sistema de comunicação digital pode ser descrito como sendo constituído pelos seguintes elementos: Fonte •* Codificador •* Canal •* Decodificador •* Destinatário.

A fonte consiste da fonte original de informação e de um codificador fonte, enquanto que o canal consiste de um modulador, um meio de transmissão e um demodulador. As razões para o crescimento do emprego de sistemas de comunicação digital são bastante conhecidas | l l |. Para estes sistemas, uma abordagem eficaz para o problema da confiabilidade envolve o uso de códigos corretores de erros. Será mostrado que a introdução de símbolos adicionais (redundantes) ao fluxo de símbolos produzidos pela fonte (dígitos de informação) permitirá que se execute a deteção e correção de erros às custas, evidentemente, de um aumento na complexidade do sistema e de uma redução na taxa de transmissão de dados (ou de um aumento na banda passante exigida). A potencialidade máxima dos códigos corretores de erros foi estabelecida•em 1948 por C.E.Shannon j35j através do teorema de codificação para um canal ruidoso. Este teorema estabelece que todo canal tem uma capacidade máxima C e para qualquer taxa de transmissão R < C, existem códigos de taxa R, os quais, com decodificação de máxima verossimilhança, tem uma probabilidade de decodificação errônea arbitrariamente pequena. O teorema' de Shannon demonstrou a existência de códigos os quais resultam numa probabilidade de erro na decodificação arbitrariamente pequena, mas não indicou como construir tais códigos.

Devido a presença de sinais ruidosos no canal, como ante

riormente mencionado, podem ocorrer erros durante a transmissão. Estes erros podem ser esporádicos e independentes, neste caso são chamados erros aleatórios, ou podem ocorrer em "burst" de vários erros de cada vez, e então diz-se que o canal tem memória. O ultimo caso[1] não será objeto de estudo neste trabalho.

Para combater os possíveis erros que possam ocorrer durante uma transmissão de dados digitais em um canal de comunicação ruidoso, são empregados códigos corretores de erros, os quais podem ser classificados como códigos lineares ou não lineares. Códigos Lineares têm os seus dígitos redundantes calculados como soma módulo 2 dos dígitos de informação, enquanto que códigos não lineares empregam lógica não linear (tais como portas e, ou, não, etc , no caso binário). No que segue serão abordados os códigos lineares, por sua importância prática e por serem estudados na quase totalidade das publicações sobre códigos.

Entre os códigos lineares existem basicamente duas técnicas distintas de controle de erros para realização de uma transmissão confiável de dados digitais em canais ruidosos: Os códigos de bloco e os códigos de árvore. Dependendo da maneira como a redundância é adicionada aos dígitos de informação (mensagem), um destes dois tipos de código é gerado. Códigos para os quais a redundância em um bloco de dígitos verifica a ocorrência ou não de erros, apenas naquele bloco particular, são chamados códigos de bloco. Já os códigos onde a redundância em um bloco verifica a existência ou não de erros em mais de um bloco são chamados de códigos de arvore. A subclasse mais importante destes códigos é a dos códigos convolucionais, os quais são mais simples e de fácil implementação com relação a outros tipos de códigos de árvore.

Os códigos de bloco e convolucionais são competitivos em muitas situações. A escolha final de um deles depende de fatores como o formato dos dados, o retardo na decodificação, a complexida-

de do sistema necessário para alcançar uma determinada taxa de er - ros, etc. É dada ênfase neste trabalho aos códigos de bloco, muito' embora os algoritmos considerados sejam na sua maioria também apli- cáveis a códigos convolucionais.

Nos anos recentes tem havido um grande interesse em es- quemas de decodificação utilizando decisão suave, aplicáveis a estes dois tipos de códigos, com a finalidade de evitar a degradação no desempenho que resulta quando uma quantização abrupta símbolo-por - símbolo precede o decodificador. Em sistemas de comunicação via sa- tél te, por exemplo, uma diminuição de 1 dB na relação sinal/ruído' proprociona uma economia de milhões de dólares.

0 intuito desta tese é de apresentar, sugerir e anali - sar alguns importantes algoritmos de decodificação de códigos linea res que façam uso de tais técnicas. Neste capitulo são estudados de forma introdutória os códigos lineares, de modo que seja possível ' a compreensão das técnicas de decodificaçao probabilísticas aborda- das em capítulos posteriores.

1.2 - CÓDIGOS DE BLOCO

1.2.1 - códigos de Bloco Lineares

Os códigos de bloco lineares representam sem dúvida a parte mais bem desenvolvida da teoria dos códigos corretores de er - ros, fato que se deve a sua forte estrutura algébrica, que permite' o emprego de ferramentas matemáticas como a teoria das matrizes, e a teoria dos campos de Galois, por exemplo.

0 processo de codificação consiste em segmentar uma men sagem em blocos de k dígitos e acrescentar a cada bloco n-k dí- gitos redundantes. Estes n-k dígitos são determinados a partir ' dos k dígitos de mensagem e destinam-se a deteção e/ou corre cão de erros que possam vir a ocorrer durante a transmissão. Des-

te modo, o codificador opera de modo independente com cada bloco , e a redundância acrescentada serve apenas para detetar e/ou corrigir erros somente no bloco considerado.

Um fato importante a ser considerado no estudo de códigos de bloco, é que para um comprimento de bloco n e uma taxa k/n fixadas, o melhor código de bloco linear tem desempenho quase tão bom quanto o melhor código de bloco com os mesmos parâmetros [5]. A utilização de códigos lineares resulta em simplificação na implementação e na análise do desempenho, por isso são os estudados quase na totalidade das pesquisas na área.

Alguns códigos de bloco simples e bastante utilizados, tais como códigos de repetição, códigos de peso constante, códigos de arranjos, códigos de único dígito de paridade e códigos de Hamming estão descritos nas referências [5], [11].

Definição 1.2.1 - Um código de bloco linear $C(n,k,d)$ é um conjunto de q^k énuplas com elementos em GF(q), chamadas de palavras código, as quais diferem entre si em pelo menos d posições e formam um subespaço do espaço vetorial V'_n de todas as énuplas definido sobre GF(q).

Deve ser notado conforme definido acima, que o codificador se torna proibitivamente complexo para valores grandes de k, desde que devem ser armazenadas as q^k palavras código de n dígitos cada, em uma memória. Isto pode ser evitado lembrando que o código é um subespaço vetorial, como será visto a seguir.

1.2.2 - Capacidade de Correção

Dado um código de bloco C(n,k,d), é importante determinar qual a sua capacidade de correção, i.e., quantos erros ele é capaz de detetar/corrigir, em cada bloco transmitido. Será mostrado que esta capacidade de correção está relacionada com o parâmetro d.

Portanto, uma importante figura de mérito de um código é a sua distância mínima d entre duas palavras código, medida em termos de distância de Hamming abaixo definida.

Definição 1.2.2a - O peso de Hamming de uma énupla ***u,*** denotado por W(u) é definido como sendo o número de componentes não nulas de u.

Assim, por exemplo, se u = (1,0,2,0,0,5,0,1), então W(u) = 4.

Definição 1.2.2b - A distância de Hamming entre duas énuplas u e v denotada por $D_H(u,v)$, é definida como o número de componentes nas quais estas énuplas diferem.

Assim, se u = (1,0,2,0,1,2) e v = (1,0,1,0,2,2), a distância de Hamming entre elas é $D_H(u,v) = 2$.

Se os vetores considerados são binários, da definição de soma módulo 2 é fácil verificar que $D_H(u,v) = W(u(+)v)$, i.e., que a distância entre u e v é exatamente igual ao peso do vetor soma.

Dado um código linear C(n,k,d), calculando as distâncias entre todos os possíveis pares de palavras código, a menor distância obtida é chamada de distância mínima do código, denotada por d. Portanto,

$$d = \underset{u/v}{\text{Min}}\ D_H(u,v) \qquad u,v \in C \tag{1-1}$$

Se u e v são palavras código de um código linear, então u (+) v também o é, visto que o código é um subespaço vetorial. Segue-se deste fato que a distância mínima para um código linear binário é igual ao peso mínimo dentre as palavras código não nulas,

Agora, com relação à capacidade de deteção de um código de bloco C(n,k,d), torna-se trivial verificar que a ocorrência

de uma quantidade de erros menor ou igual a d-1 resulta em uma ' n k das 2 -2 ênuplas que não são palavras código, o que possibilita a deteção de erros. Portanto, um código com distância mínima d é capaz de detetar até (d-1) erros.

Pode ser facilmente mostrado 136| que na decodificação por máximo de verossimilhança com canal BSC, o decodificador estima a palavra transmitida como sendo a palavra código mais próxima* da palavra recebida no sentido de distância de Hamming. Se o código de bloco é usado para correção de erros aleatórios e tem distância mínima tal que 2t+2 > d > 2t+l, então este decodificador poderá corrigir todas as configurações de erros contendo t ou menos[1] erros que possam ter ocorrido I36 j.

Assim, um código com distância mínima d tem uma capacidade corretora de $t = \frac{d-1}{2}$ erros

Desta forma o problema de encontrar um código com uma' dada capacidade de correção, consiste basicamente em garantir uma dada distância mínima. A teoria da informação proporciona o estabelecimento de cotas superiores e inferiores para d, quando são fixados um comprimento n e uma taxa k/n, contudo não indica como' construir tais códigos |l5|•

1.2.3 - Matriz Geradora de um Código Linear

Para um subespaço C de V''_n constituindo um código[1] linear, é possível encontrar um conjunto de k énuplas linearmente independentes, as quais constituem uma base para C. Então, cada énupla u E C pode ser expressa como combinação linear dos vetores {v^, Y_2,...,y_k} da base, **ÍA,**

$$u \in C \Rightarrow u = m_1 V_1 (+)_q \text{ nu,} V_2 \copy_g \ldots \copy_q m \hat{} \qquad (i-_2)$$

ou seja,

$$: z\ C \Rightarrow u = \sum_{l=i}^{k} m_p V_{\text{»}} \ , \quad (V£)\ m_p \text{ e } GF(q) \qquad (1\text{-}3)$$

Um código linear C(n,k,d) constitui um subespaço de V com dimensão k, de modo que é possível representá-lo através' dos k vetores que constituem a base. Estes vetores (X ^ i podem ser arranjados como linhas de uma matriz kxn, denominada de matriz geratriz, como mostrado abaixo.

$$[G] = \left| \begin{array}{c} *1 \\ \\ . \end{array} \right| = \left| \begin{array}{ccc} {}^{\prime}ll & v_{12} & .. \\ {}^{v}21 & & \text{"} \\ \cdot & & \\ & \backslash 2 & \bullet\bullet \end{array} \right| \begin{array}{c} > \\ \\ \vdots \\ \cdot \\ kn \end{array} \qquad (1\text{-}4$$

Se é assumido um bloco de informação m = (m^,...,m^), então a palavra código correspondente é dada por u = m[G] ;

$$u = (m,,... \quad {}^{2l} \quad {}^{k} \quad = Z \quad {}^{1=1} \qquad (1\text{-}5$$

$$^{2} \quad {}^{k}$$

ou seja, a palavra código correspondente â mensagem m é uma combinação linear de linhas de [G] .

Assim, as linhas da matriz geradora [G] geram um código linear C(n,k,d), onde cada bloco de informação é codificado ' em uma palavra com n dígitos para transmissão. A razão R = k/n é chamada de taxa (ou eficiência) do código.

Desde que um código linear é completamente especificado pela matriz [G], o tamanho da memória requerida no codificador' é reduzido enormemente. O codificador pode armazenar somente as k linhas de [G], ao invés das q palavras código, se ele possuir ' elemento lógico capaz de fazer as combinações lineares das linhas '

armazenadas.

Exemplo 1.1 - Seja um código de bloco linear (5,3,2) com matriz geratriz dada por

$$[G] = \begin{vmatrix} 2^l \\ \\ V_3 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{vmatrix}$$

A palavra código correspondente à mensagem m = (1,0,1), por exemplo, é encontrada facilmente por

$$u = (1,0,1) \begin{matrix} {}^{2^l} \\ = \\ {}^{V_3} \end{matrix} (1,0,1,1,0).$$

É possível codificar cada bloco de informação em palavras código de tal maneira que os k primeiros dígitos da palavra código sejam exatamente os mesmos do bloco de informação, e os últimos n-k dígitos sejam os dígitos redundantes que são função dos dígitos de informação. Um código nesta forma é dito sistemático. A redundância é adicionada de maneira a proporcionar capacidade de proteção à mensagem, e o problema do codificador se reduz a determinar estes dígitos.

Um código sistemático pode ser descrito por uma matriz geradora [G] na forma

$$[G] = \begin{matrix} 1 & 0 & \cdots & 0 & P_{1,1} & \cdots & P_{1,\,n-k} \\ 0 & 1 & \cdots & 0 & P_{2,1} & \cdots & P_{2,\,n-k} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & P_{k,1} & \cdots & P_{k,\,n-k} \end{matrix} \qquad (1\text{-}6)$$

Onde $p_{ij} \in GF(q)$.

Isto pode ser representado na forma [G] = [I_k | P], onde I_k é uma matriz identidade kxk e [P] uma matriz kx(n-k) com elementos em GF(q).

Se é considerado um bloco de informação m = (m_1, ..., m_k), e uma matriz geradora na forma acima, a palavra código correspondente será dada por

$$u = (m_1, \ldots, m_k) \cdot [G] \qquad (1-7)$$

Efetuando a multiplicação, são obtidas as equações de verificação de paridade do código:

$$u_j = m_1 p_{1j} \oplus \ldots \oplus m_k p_{kj} \qquad (1-8)$$

Os k primeiros dígitos da palavra código são exatamente os dígitos de informação a serem transmitidos, enquanto que os últimos n-k dígitos são redundantes (ou de verificação de paridade), calculados como funções lineares dos dígitos de informação.

1.2.4 - A Matriz de Verificação de Paridade

Para cada matriz [G] kxn, existe uma matriz [H] (n-k)xk de tal forma que o espaço linha de [G] é ortogonal a [H], isto é, o produto interno de vetores do espaço linha de [G] com as linhas de [H] é nulo. Esta matriz é denominada de matriz de verificação de paridade e pode ser escrita sob a forma

$$[H] = [h_1 \; h_2 \ldots h_{n-k}]^T, \text{ onde } (\forall i)\ h_i \text{ é uma énupla} \qquad (1-9)$$

Desta forma, se [H] é o espaço nulo de [G], então se verifica a relação

$$[G] \cdot [H] = 0 \qquad (1\text{-}10)$$

Se o código está na forma sistemática [G] = [1^'P], então o espaço linha de [G] é o espaço nulo da matriz [H] dada por [H] = [-P^T jI_n_k]. (1-11)

Definição 1.2.4 - Se [H] é usada como matriz geratriz de um código de bloco, este código é dito ser o dual do código gerado pela matriz [G].

Seja u = (u,,...,u) um vetor qualquer do espaço linha de [G]. Então u [H] = 0_ , pois desde que [H] e o espaço nulo de [G], u.hjL = 0 para i=l,2,...,n-k.

Então o código linear gerado por [G] pode também ser descrito de outra maneira:

"u é palavra código se e somente se u[H]^T = 0^, onde[H] é a matriz de verificação de paridade do código linear gerado por [Gp.

Exemplo 1.2 - A matriz [H] para o código de bloco (5,3,2) introduzido no exemplo 1.1 pode ser determinada pela equação (1-11) resultando em

$$[H] = \begin{matrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{matrix}$$

A énupla u = (1,0,1,1,0) resulta em u[H] = 0.

Conclui-se, portanto ,que um código linear é unicamente' especificado pela matriz geradora ou pela matriz de verificação de paridade.

Na construção de um código linear, a matriz [P] deve' ser escolhida de modo que o código tenha as propriedades de corre - ção de erros desejadas.

Para um código com matriz geradora [G] = [1^- P], as equações de verificação de paridade podem também ser obtidas direta mente da matriz [H] = [-P^{T}- $I_{n-k.}$,] , como mostrado abaixo. Seja u = ÍUi|...fU$_{n}$) uma palavra código correspondente a mensagem ' m = (m, . . . ,m,) , o que implica em $_{u}$- =m . , i=l, 2, . . . , k .

T

Desde que u[H] = *0_*, segue-se que

^{u}k+j $^{=P}$ l j ^{u}l© ©^{P}kj^{u}k ' $^{i'e}$-
^{u}K$_{+}$j $^{=P}$ij^{m}l©- - -©Pkj^{m}k ^{para} J " ^{1} * ^{2} * (1 - 1 2)

que são as equações já encontradas em (1-8).

1.2.5 - Síndrome

Suponha que uma palavra código u de um código de bloco linear C(n,k,d) com matriz geradora [G] e matriz de verificação de paridade [H] é transmitida através de um canal ruidoso. No receptor, uma énupla r é recebida, a qual pode diferir de u devido ao ruído adicionado durante a transmissão. A tarefa do decodificador é recuperar u a partir de r.

T

Definição 1.2.5 - O vetor de n-k componentes S =£[H] e chamado de síndrome da énupla r.

Como visto na seção anterior, u é uma palavra código ^{1} se e somente se sua síndrome é nula.

Portanto, a síndrome de um vetor recebido na saída do ^{1} canal pode ser usada para deteção/correção de erros.

O processo de decodificação envolve uma decisão sobre ^{1} qual foi a palavra código transmitida. O arranjo padrão 124 | suge-

re uma maneira de como fazer isto. No espaço V , as 2^N énuplas são distribuídas em 2^k conjuntos distintos, de modo que cada um deles contenha apenas uma palavra código. Todos os elementos de um dado conjunto possuem a mesma síndrome, e dois conjuntos distintos são associados a síndromes diferentes. Desta forma é estabelecida uma correspondência um a um entre uma configuração de erros (coset leader) e uma síndrome. Neste caso, o decodificador corrige erros quando a configuração de erros do canal é um "coset leader". Estes fatos implicam em um procedimento geral para decodificação de códigos de bloco lineares, denominado de busca sistemática 130] •

1 . 2 . 6 - Códigos Cíclicos Binários

Uma considerável parte das pesquisas em códigos de bloco estão concentradas numa subclasse conhecida como códigos cíclicos, introduzidos por Prange 126 | em 1957 .

Estes códigos são também os mais importantes do ponto de vista de aplicações em engenharia. Tudo isto é devido a sua forte estrutura matemática que permite uma considerável simplificação na implementação dos sistemas. Os procedimentos de decodificação de códigos de blocos lineares são também aplicáveis a códigos cíclicos, todavia, propriedades algébricas decorrentes da estrutura cíclica permitem simplificações importantes, visto que a codificação e o cálculo da síndrome podem ser facilmente realizados empregando registros a deslocamento com conecções de realimentação. Sua estrutura inerentemente algébrica também torna possível encontrar vários métodos de decodificação simples e eficientes. Entre os procedimentos de decodificação mais importantes para este tipo de código, são conhecidos DECODIFICAÇÃO DE MEGGITT]36|, DECODIFICAÇÃO POR LÓGICA MAJORITÁRIA 128| e o método conhecido como "ERROR-TRAPPING" J24 | , os quais são de implementação simples e bastante atrativos.

Definição 1.2.6 - Um código de bloco é cíclico quando uma permuta - ção cíclica, aplicada a qualquer de suas palavras, resulta ainda em uma palavra código, ije., se v = (VQ,v^, . . .,'_]^ 6 uma palavra código, então v' = (v , v ~ , v , v . ,) também o é, consi-

derando os índices reduzidos módulo n.

É possível tratar as componentes do vetor v como coefi cientes de um polinómio de grau sempre menor ou igual a n-1, através de uma correspondência um a um:

$$v = (v_0, \ldots, v_{n-1}) \Leftrightarrow v(x) = v_0 + v_1 x + \ldots + v_{n-1} x^{n-1} \quad (1\text{-}13)$$

Fazendo uso de propriedades de campos finitos, prova-se' 136 I<3^i^ todas as palavras de um código cíclico (n,k,d) são múlti - pias de um polinómio g(X), de grau n-k, bem definido; e reciproca mente que todo polinómio de grau igual ou menor que n-1, e que seja divisível por gCX), ê uma palavra código.

O polinómio g(X) é chamado polinómio gerador do código cíclico e é fator de X^n+1, ie.,

$$g(X)h(X) = X^n+1, \quad \text{ou} \quad g(X)h(X) = 0 \mod X^n+1.$$

A representação de um código cíclico pode ser feita atra vés do seu polinómio gerador g(X), da mesma forma que a matriz [G] representa um código de bloco linear.

Conforme mencionado, as palavras código são múltiplas do polinómio g(X). Desta forma, os polinómios g(X), Xg(X),...,X g(X) são palavras código e como são linearmente independentes, podem ' ser usadas como base para formar a matriz geradora do código cíclico, como mostrado a seguir.

$$x^{k-1}g(x)$$

$$[G] = \quad x\,g(x)$$

$$g(x)$$

Para fins de codificação, a propriedade do deslocamento cíclico permite uma implementação sequencial da matriz [G] .

Representando a mensagem a ser codificada por um polinômio m(X) de grau menor ou igual a k-1, um codificador na forma sistemática gera a palavra código de acordo com

$$V(X) = C(X) + X^{n-k}\,m(X) \qquad (1\text{-}14)$$

onde C(X) é o resto da divisão de $X^{n-k}\,m(X)$ por g(X),

$$C(X) = \text{rem}\{X^{n-k}\,m(X)/g(X)\}. \qquad (1\text{-}15)$$

A implementação do codificador pode ser feita através de registro a deslocamento com n-k estágios, pois multiplicações e divisões de polinómios com coeficientes em GF(2) podem ser efetuadas facilmente com o auxílio de registradores a deslocamento |11|.

Um codificador com k estágios ao invés de n-k estágios pode também ser implementado]30| * levando em consideração o polinómio de verificação de paridade h(X).

A síndrome pode ser facilmente determinada, conforme será mostrado a seguir. De um modo geral, a palavra recebida pode ser representada pela expressão r(X) = V(X) + e(X), onde e(X) é um polinómio correspondente à configuração de erros introduzida no canal. A síndrome pode ser determinada cono o resto da divisão entre o polinómio recebido e o polinómio gerador, **ÍÊ.,**

$$S(X) = \text{rem}\{r(X)/g(X)\} = \text{rem}\{\frac{l}{g(X)} + \frac{1*1}{g(X)}\} \qquad (1\text{-}16)$$

Como $V(X) = m(X)\, g(X)$, segue-se que

$$S(X) = \text{rem}\ \{e(X)/g(X)\} \qquad (1\text{-}17)$$

Caso $S(X)$ seja zero, o decodificador aceita a palavra recebida como pertencente ao código. No caso contrário, i,e., $S(X) \neq 0$, considera que ocorreram erros durante a transmissão. Desta maneira' é fácil perceber a simplicidade de um circuito decodificador para [1] deteção de erros quando são empregados códigos cíclicos.

1.2.7 - Códigos de Hamming e Códigos B.C.H

Os códigos de Hamming, introduzidos em 1950 llól, foram os primeiros códigos não triviais propostos para corrigir erros. São códigos lineares que corrigem um erro por bloco e têm distância mínima igual a 3. Para cada inteiro $c>l$, existe um código de Hamming com os seguintes parâmetros:

$$n = 2^{c}-1 \qquad k = 2^{\circ}-l-c \qquad e \qquad d = 3.$$

A condição $n = 2^{C}-1$ é escolhida de modo a assegurar ' disponibilidade de redundância suficiente para verificar a ocorrência de um erro por palavra, pois o número de síndromes não nulas [1] $2^{C}-1$ resulta exatamente igual ao número de posições n em que o erro pode estar localizado. Isto significa que os códigos de Hamming são códigos perfeitos conforme a definição abaixo.

Definição 1.2.7 - Um código de bloco (n,k,d) corretor de t erros e definido em GF(q) é perfeito se e somente se

$$\sum_{i=0}^{t} (q-1)^{i} \binom{n}{i} = q^{n-k}$$

Com excessão dos códigos de Hamming, o código de Golay (23,12,7) com t=3 e o código ternário (11,6,7) com t=3, não existem outros códigos perfeitos não triviais.

É interessante observar que os códigos de Hamming são também códigos cíclicos e que a taxa (eficiência) é dada por $R = 1 - \frac{c}{2^c - 1}$, consequentemente são códigos cuja eficiência tende a 1 assintóticamente. Os códigos (7,4,3), (15,11,3) e (31,26,3) serão utilizados nas análises realizadas nos capítulos seguintes.

Com relação aos códigos BCH, estes códigos foram criados independentemente por Hocquenghen (1959) e por Bose-Chandhuri (1960); são códigos cíclicos, e representam a classe mais importante de códigos de bloco com algoritmos algébricos de decodificação Para quaisquer inteiros positivos m e t ($t<2^m-1$) existe um código BCH com os seguintes parâmetros:

$$n = 2^m - 1 \quad n-k < mt \quad e \quad d > 2t + 1.$$

O teorema fundamental dos códigos BCH é a seguir enunciado.

Teorema 1.1 - O código BCH cujo polinómio gerador tem d-1 raízes consecutivas $a, a^2, a^3, \ldots, a^{d-1}$ tem distância mínima pelo menos d.

Para um tratamento completo destes códigos o leitor deve consultar as referências | 3 ! e 126l -

1.3 - CÓDIGOS CONVOLUCIONAIS

1.3.1 - Considerações Gerais

Na classe de códigos denominada de códigos convolucionais, a redundância adicionada a cada bloco é função de mais de um bloco de informação, de modo que estes não são tratados de forma in

dependente como nos códigos de bloco. Estes códigos foram primeiramente introduzidos |26 | por P.Elias (1955), e podem ser classificados como fixos ou variantes no tempo. Entretanto, a grande maioria' dos códigos convolucionais empregados é invariante no tempo, de forma que apenas estes serão abordados aqui.

Na decodificação de códigos convolucionais, vários sub-blocos sucessivos recebidos são processados. Idealmente, toda sequência transmitida deveria ser empregada na decodificação, o que não é feito devido a limitações práticas e por introduzir um retardo muito grande na decodificação. Todavia, a degradação resultante' desta simplificação é inteiramente aceitável.

Num sentido restrito, códigos convolucionais podem ser vistos como uma classe especial de códigos de bloco lineares, porém, uma observação mais cuidadosa revela que a estrutura convolucional proporciona a um código linear propriedades superiores que melhoram o desempenho.

A estrutura de um código convolucional pode ser exibida por meio de diversas representações, visto que um codificador convolucional fixo pode ser considerado como uma máquina sequencial de estados finitos, invariante no tempo.

1.3.2 - 0 Codificador Convolucional

Na entrada do codificador é alimentado um bloco com k símbolos de informação (elementos de GF(q)), e um bloco com $n>k$ símbolos código é gerado e aparece na saída do referido codificador. Os símbolos de saída são combinações lineares em GF(q) dos símbolos de entrada dos $N-l$ blocos precedentes. Deste modo, os n dígitos de saída dependem não somente dos k dígitos de mensagem de um mesmo sub-bloco , mas também dos $N-l$ sub-blocos de mensagem anteriores.

Um código assim gerado é dito ser um código convolucional (n,k,N), com comprimento de restrição no codificador de N [1] blocos. A taxa assintótica para este código é R = k/n, e em geral, k e n são inteiros pequenos.

A formulação matricial para estes códigos é descrita a seguir:

Inicialmente é considerado um conjunto de k.(n-k) vetores com N dígitos binários cada, denominados SUBGERADORES:

$$(i,j) = (g_0(ifJ) > g^{\wedge}\text{íi/J}\} \qquad (1\text{-}18)$$

Para i = l, 2, . . . , k e j= l,2,...,n-k.

A partir deste conjunto, forma-se uma matriz com k linhãs,

$$g\ (D = [i_{\iota}\ P_0\ o\ p\ o\ P_2 . . . o\ P_{N-x}\ 0\ f^{\wedge 8} i^{*}] \qquad (1\text{-}19)$$

g (k)

Onde 1^ tem dimensão kxk

0 tem dimensão kxk

P^ tem dimensão kx(n-k), 0<£<N-1

$$[Pp] = \begin{matrix} g^{\wedge}(l,l)\ g_{\varepsilon}(l,2) \ldots g_{\varepsilon}(l,n\text{-}k) \\ g^{\wedge}(k,l)\ g_{\varepsilon}(k,2) \ldots g_{\varepsilon}(k,n\text{-}k) \end{matrix} \qquad (1\text{-}20)$$

Assim, cada linha g (i) é um vetor semi-infinito com
CO
componentes não nulas confinadas aos primeiros nN dígitos.

O vetor g(i) contendo somente as primeiras nN posições do vetor g^íi) é dito um GERADOR.

Um código convolucional fixo linear pode ser representado por uma matriz geradora sob a forma:

$$G = \begin{bmatrix} P_0 & P_X & 0\ldots0 & P_{N-1} & \text{zeros} \\ 0 & P_{N-l} & \text{zeros} & & \\ & 0 & P_{N-l} & \text{zeros} & \end{bmatrix} \qquad (1\text{-}21)$$

Alguns exemplos apresentados esclarecem como determinar a matriz G^ a partir dos subgeradores.

Exemplo 1.3 - Considere um código convolucional sistemático (2,1,4) o qual é gerado de acordo com o subgerador g(l,l) = (1 1 0 1).

O gerador é g(l) = (1 1 0 1 0 0 0 1) ea matriz geradora i determinada como sendo

$$G = \begin{bmatrix} 1\,1 & 0\,1 & 0\,0 & 0\,1 & & \\ & 1\,1 & 0\,1 & 0\,0 & 0\,1 & \\ & & 1\,1 & 0\,1 & 0\,0 & 0\,1 \end{bmatrix}$$

Portanto, para uma mensagem a ser codificada m = (1 £ 0^ *1* l....), a sequência codificada é encontrada por £ = m G^ , i,e., C = (11 01 00 10 10 . . .).

Exemplo 1.4 - Considere agora o código convolucional (4,3,3) com os seguintes subgeradores:

$$g(l,l) = (1\ 1\ 1\ 1)$$
$$g(2,l) = (1\ 0\ \ 1)$$
$$g(3,l) = (1\ 1\ 0\ )$$

Os geradores são determinados como sendo:

g(1) = (1 0 0 1 Q 0 0 1 0 0 0 1)

g(2) = (0 1 0 1 0 0 0 0 0 0 0 1)

g(3) = (0 0 1 1 0 0 0 1 0 0 0 0)

e a matriz geradora será dada por

```
        l o o n    o o o : i    o o o : i
              :            :            :
                                                    zeros
        o í o : i  o o o:o   o o o : i
                              0 0 0.0
G  =    o o 1:1    o o o : i
                   10011     o o o : i    0 0 0 : i
                       .          .          .     zeros
                   o i o : i  oo o:o   o o o : i

                   001:1     o o o i i    o o o :  0
```

Deve ser observado que $P_0^T = (1\ 1\ 1)$, $P_1^T = (1\ 0\ 1)$ e $P_2 = (1\ 1\ 0)$.

Um código convolucional (n,k,N) pode também ser especificado pela sua matriz de verificação de paridade H^, tal como um código de bloco. Novamente e verificada a relação $0^11^T = 0,$ donde segue-se que c é uma sequencia código se e somente se $c^T = \underline{0}$.

Para um código convolucional sistemático, ou seja, com matriz geradora da forma descrita em (1-21), a matriz de verificação de paridade é dada por

```
      | I
      |   n-k
      |   0
      |»5
H  =  |p^T    0          0        P_0^T   n-k
      | .             .  .                                   (1-22)
      |               .  .
      | N-1  0      P^T_2          P^T_1°  P_0^T     V k
               P_N-1  0    P^T_N-2
                                                     n-k
```

Exemplo 1.5 - Como um exemplo, a matriz de verificação de paridade para o código (4,3,3) descrito no exemplo 1.4 é determinada abaixo.

$$H = \begin{matrix} 1\,1\,1\,1 & & & \\ 1\,0\;\;1\,0 & 1\,1\,1\,1 & & \\ 1\,1\,0\;\;0 & 1\,0\;\;1\,0 & 1\,1\,1\,1 & \\ & 1\,1\,0\;\;0 & 1\,Q\,1\,0 & 1\,1\,1\,1 \end{matrix}$$

00

1.3.3 - Outra Representação da Matriz Geradora

É possível também representar a matriz G_{to} numa outra forma que facilita o projeto dos circuitos codificadores, como será mostrado adiante.

Para um código convolucional (n,k,N), a matriz G^ pode ser escrita sob a forma:

$$G = \begin{matrix} g_0 & g_x & g_2 & \ldots \\ {}^{g}o^{g}\,i & & & 'N\text{-}l \\ 9_0\;{}^{g}i & & & 'N\text{-}l \end{matrix} \qquad (1\text{-}23)$$

4

00

Onde gj / J ~ 0,1,... N-l são matrizes kxn.

Se o código é sistemático, comparando (1-21) e (1-23) , é possível observar que existem relações da forma

$$9_0 = I_k\ P_0\ '{}^{g}l = {}^{o}{}^{p}l\ '{}^{g}2 = {}^{o}{}^{p}2'\text{--}'\ '{}^{g}N\text{-}l = {}^{o}{}^{p}N\text{-}1\ \bullet$$

A matriz de verificação de paridade também pode ser es-

crita na forma abaixo.

$$H = [h_2 \quad h \quad h_Q] \tag{1-24}$$

N - IV:

$$\begin{matrix} 4 \\ 00 \end{matrix}$$

Onde $h_0 = p£ I_{n\_k}$, 0, $h_2 = P^T_2$ O. . . . $h_{.N}$ - i = P N - l °-

Os k geradores podem ser facilmente obtidos a partir' da relação

$$\begin{matrix} gCD \\ \\ g(k) \end{matrix} \quad \hat{0}^g \; l \; * \; * \; *^g N - l \tag{1-25}$$

Os exemplos apresentados abaixo mostram como esta repre sentação é refletida na implementação do circuito codificador.

Exemplo 1.6 - Um circuito codificador para o código convolucional' (2,1,3) especificado através de ^2 abaixO/é mostrado na figura 1.1.

g_0 = (L i)

g_1 = (1,0) => gerador g(l) = (1 1 10 11)

g_2 = (1.1)

Figura 1.1 - CODIFICADOR PARA O CÓDIGO CONVOLUCIONAL (2,1,3).

Exemplo 1.7 - Um circuito codificador para o código convolucional' não sistemático (3,2,2) especificado através das matrizes e g^' abaixo, está apresentado na figura 1.2.

í i i

9₀

0 1 0 g(l) = (1 1 1 1 0 1)

=> geradores

1 0 1 g(2) = (0 1 0 1 1 0)

1 1 0

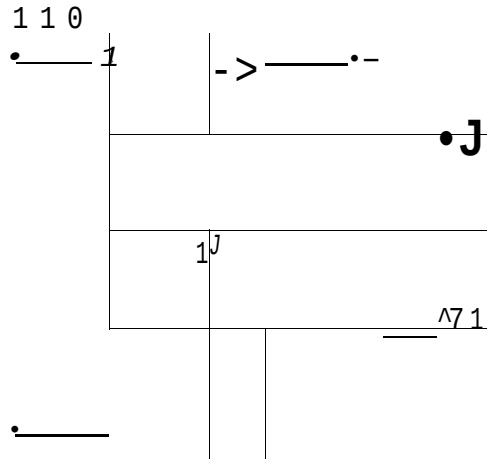


Figura 1.2 - CODIFICADOR PARA O CÓDIGO CONVOLUCIONAL (3,2,2).

Portanto, para implementação do codificador de um código convolucional (n,k,N) são necessários k.(N-l) estágios de registro a deslocamento com ligações determinadas pelos geradores.

1.3.4 - Diagrama em Arvore, Treliça e Diagrama de Estado

Uma das maneiras de representar a sequência de saída resultante da codificação de uma dada sequência de entrada para um código convolucional fixo, é por meio de uma representação da árvore' associada ao codificador.

Os bits de entrada do codificador são indicados pelo caminho seguido na arvore, enquanto que os bits de saída são indicados nos ramos. Um zero de entrada é representado pelo ramo superior de uma bifurcação, enquanto que o um é representado pelo ramo inferior.

O diagrama em árvore para um código convolucional não sistemático (4,1,3) é apresentado na figura 1.3 abaixo.

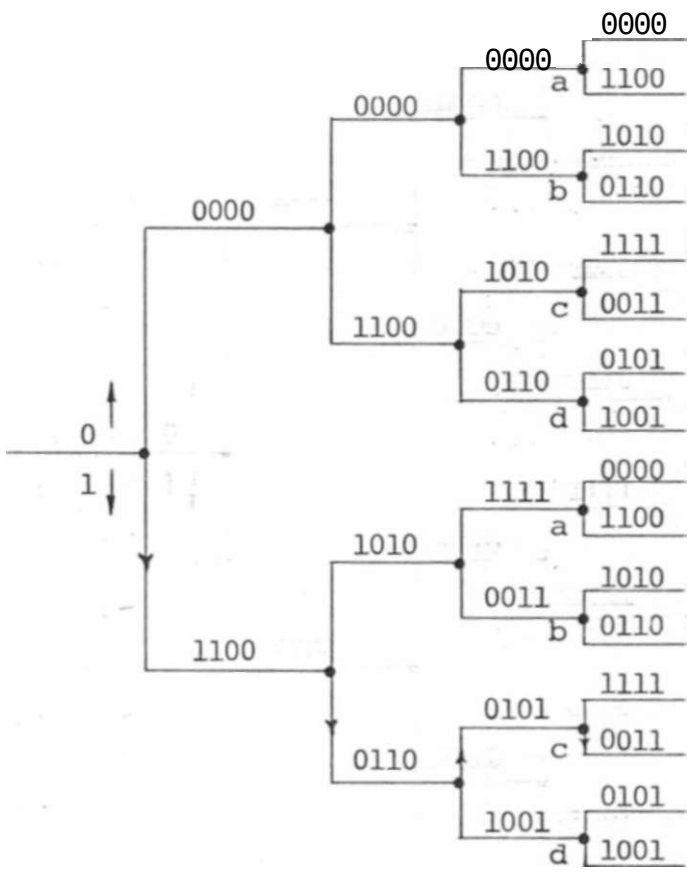


Figura.1.3 - DIAGRAMA EM ARVORE PARA O CÕDIGO CONVOLUCIONAL (4,1,3).

Deste modo, a sequência 1 1 0 1 0 ... aplicada na entrada do codificador produz uma sequência de saída [1]

1 1 0 0 0 1 1 0 0 1 0 1 0 0 1 1 0 1 1 0 ...

Observando-se o diagrama de arvore, nota-se que sua estrutura torna-se repetitiva após um certo número de ramos, o que está relacionado com o fato do codificador convolucional ser uma máquina com estados finitos. Isto torna possível fazer uma modificação que consiste em juntar os nos indicados por uma mesma letra, fazendo com que o diagrama em árvore evolua para um diagrama de treliça.

A treliça para o código (4,1,3) representado na figura 1.3 é facilmente obtida, resultando no diagrama mostrado abaixo.

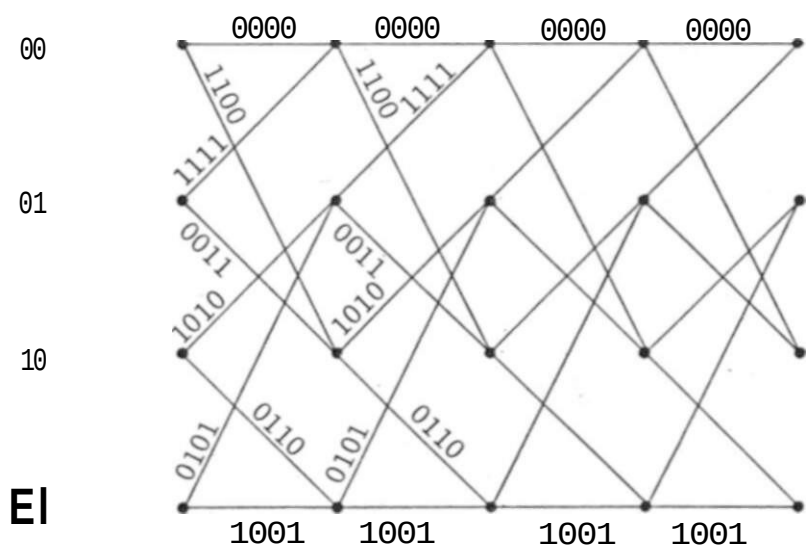


Figura 1.4 - DIAGRAMA DE TRELIÇA PARA CÓDIGO CONVOLUCIONAL (4,1,3).

Pode ainda ser observado que é possível fazer uma representação em diagrama de estado de um codificador convolucional. A partir da treliça é possível construir este diagrama, que representa perfeitamente o codificador. O diagrama para o exemplo em questão é mostrado na figura 1.5.

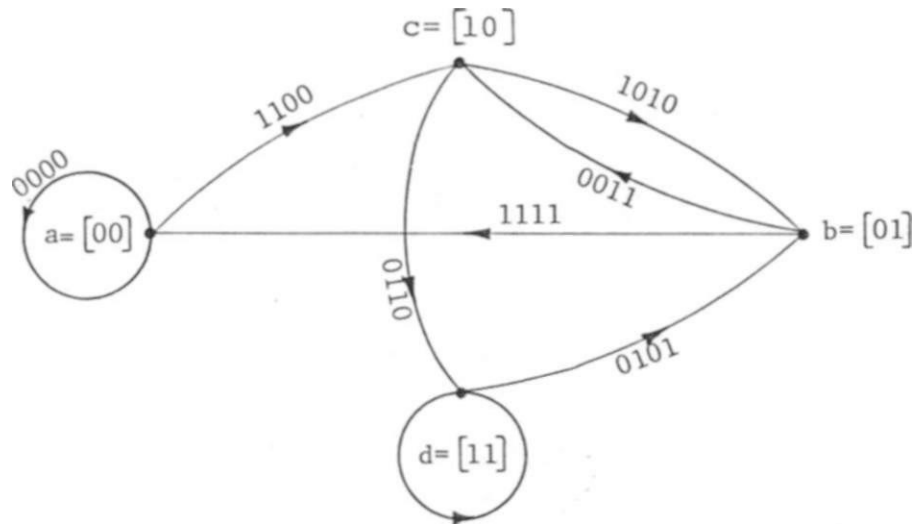


Figura 1.5 - DIAGRAMA DE ESTADO PARA O CÓDIGO CONVOLUCIONAL (4,1,3).

Utilizando este diagrama é possível a determinação do conjunto de pesos dos caminhos do código de uma forma relativamente simples [37].

A determinação da distância livre para um código convolucional, a qual é obtida a partir do conjunto de distâncias entre* o caminho todo zero e os demais caminhos que dele divergem, também' pode ser feita através do diagrama de estado [30] -

1 . 4 - DECODIFICAÇÃO DE CÓDIGOS CONVOLUCIONAIS

Vários procedimentos são usuais na decodificação de códigos convolucionais. Entre as técnicas mais comumente utilizadas[1] podem ser citadas a DECODIFICAÇÃO SEQUENCIAL [40], (algoritmo de FANO/WOZENCRAFT) e o ALGORÍTMO DE VITERBI [37], entre outras.

Para alguns códigos, a decodificação também pode ser realizada utilizando LÓGICA MAJORITÁRIA [36].

O leitor interessado nestes procedimentos deve se diri-

gir ã literatura apropriada |i|, |28|, |40|.

Ainda com relação a decodificação de códigos convolucionais, serão apresentados no capítulo III, duas maneiras de utilizar técnicas de decisão suave. Inicialmente, um decodificador será apresentado, o qual utiliza decisão suave e é ótimo no sentido de que minimiza a probabilidade de erro por bit transmitido. É também conhecido que o algoritmo de Viterbi para decodificação de códigos convolucionais maximiza a probabilidade a posteriori das sequências código em canais sem memória.

CAPÍTULO I I

DECODIFICAÇÃO USANDO DECISÃO SUAVE

2.1 - INTRODUÇÃO

Como já discutido no capítulo anterior, sabe-se que na prática,empregam-se largamente algoritmos de decodificação basea - dos em propriedades estruturais de códigos algébricos, o que permite frequentemente uma implementação relativamente simples em termos de circuitos digitais. Deve ser observado, entretanto, que estes procedimentos apresentam uma séria deficiência, uma vez que assumem um modelo de canal no qual em cada período de tempo um dos q possíveis sinais do alfabeto é transmitido e um deles ê recebido.

Este modelo implica em um receptor que efetua uma decisão abrupta, baseada no sinal recebido, escolhendo das q letras do alfabeto do transmissor aquela que foi a mais provavelmente [1] transmitida. Tal receptor descarta,portanto,toda informação acerca' da confiabilidade desta escolha, bem como acerca das probabilidades das outras letras que não a escolhida. Este fato torna bastante cia ro que o desempenho dos sistemas codificados pode ser melhorado pelo uso da informação probabilística associada ã palavra recebida.

Numa recepção que utiliza decisão abrupta, esta informação é perdida na quantização que precede a decodificação. Um receptor que emprega decisão suave é aquele que tenta reter toda (°u parte) da informação probabilística para uso apropriado pelo decodificador.

Em geral, os códigos corretores de erros empregados em receptores práticos são códigos lineares binários. A grande parte dos procedimentos existentes para decodificação deste tipo de código assumem a existência de canais com saída binária, ou seja, empregam um quantizador antes da decodificação para extrair a informação na forma digital. A maioria dos detetores existentes constam básicamente de um demodulador analógico, seguido por um decodificador digital que opera sobre os dígitos binários produzidos pelo demodulador. O bloco demodulador atua como um quantizador que apresenta um limiar de decisão, resultando em duas saídas distintas. Isto significa que de acordo com o valor da amostra que é colhida (acima ou abaixo do limiar de referência),um dígito binário correspondente (1 ou 0) é entregue ao decodificador. É desta forma que a informação probabilística, potencialmente ütil, entregue ao demodulador é inteiramente destruída, fato que reduz substancialmente a eficiência do sistema de comunicação. Um processo de deteção (demodulação e decodificação) com decisão suave utiliza a informação de natureza probabilística associada à mensagem recebida, de modo a obter uma melhor estimativa da mesma. A idéia é fornecer ao decodificador informação sobre a confiabilidade dos dígitos recebidos, evitando assim a degradação que resulta quando se utiliza antes da decodificação uma quantização de apenas duas regiões.

A maioria da perda de informação pode ser recuperada se a saída do demodulador é quantizada em 2^ regiões, 2^ ^ em cada lado do limiar de referência.Então,para cada dígito de entrada modulado,o circuito demodulador fornece um dígito de decisão (indicando se a saída está acima ou abaixo do limiar),e Q-l dígitos de confiabilidade(indicando quão distante a saída está do limiar de decisão). Os Q dígitos de decisão suave constituem então a saída total,ao invés de somente um dígito,como no caso de decisão abrupta.

Na figura 2.1 é mostrado um exemplo para Q=2.

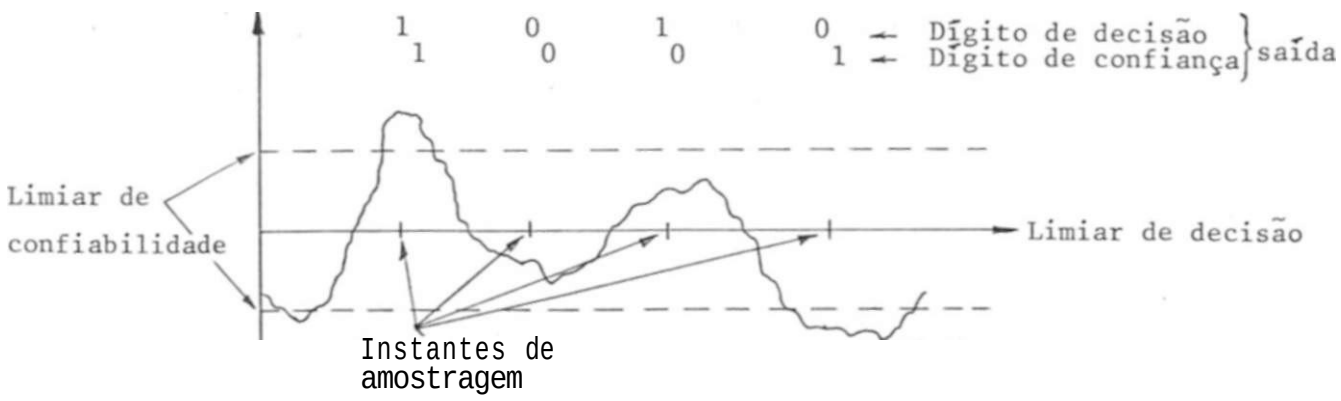


Figura 2.1 - DECISÃO SUAVE - QUANTIZAÇÃO EM 4 REGIÕES DE CONFIABILIDADE.

A mais simples forma de decisão suave é o apagamento, ou deteção por zona nula. Para um estudo mais detalhado, as referências] 4 I e I5 I são apropriadas. Neste método, a região próxima (imediatamente acima e imediatamente abaixo) do limiar de decisão é chamada de zona nula. Saídas caindo dentro desta região são entregues ao decodificador rotuladas como apagamento E, como observado no exemplo da figura 2.2.

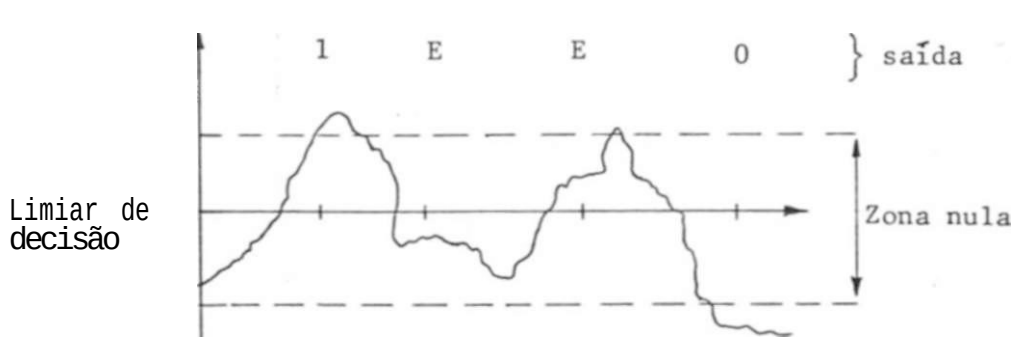


Figura 2.2 - DETEÇÃO POR ZONA NULA.

Quando o decodificador utiliza decisão abrupta, é assumido um modelo de canal BSC, enquanto que o uso de apagamento resulta em um canal BEC. Os algoritmos algébricos de decodificação podem ser modificados para manipularem eficientemente os apagamentos.

O decodificador utilizando deteção por zona nula tem agora algum conhecimento de onde os erros provavelmente ocorreram, e pode decodificar de acordo com este fato, ou seja, os d-1 erros detetados são localizados através da informação de confiabilidade.

Desta maneira, a potência de correção de erros de um código pode ser aproximadamente dobrada, desde que um código com distância mínima d pode corrigir d-1 apagamentos, mas somente $\frac{d-1}{2}$ erros.

A utilização de mais de duas regiões de quantização é uma tentativa de diminuir a perda de informação que resulta quando uma decisão abrupta é realizada.

Portanto, o detetor ótimo, i,e., aquele que no processo de decisão retém toda a informação contida no sinal recebido, seria aquele que entregaria ao decodificador o valor exato das amostras colhidas. Mais tarde, no capítulo III, estes detetores serão abordados.

Uma maneira de se utilizar os dígitos de confiabilidade no processo de decodificação é através do conceito de distância suave.

A distância suave d_s entre duas palavras código pode ser calculada de acordo com o procedimento descrito abaixo:

1) - invertendo os dígitos de confiança da palavra se seu dígito de informação correspondente é zero.

2) - Somando módulo 2 as duas palavras modificadas de acordo com o

passo 1.

3) - Dividindo a sequência resultante em n subsequências de Q [1] dígitos cada.

4) - Interpretando cada ama das subsequências de Q dígitos como um número binário e convertendo para decimal.

5) - Somando os números obtidos em 4, resultando em um decimal to - tal.

Assim, d$_g$ entre 000 111 e 011 011 é obtida por

0 1 1 I 1 1 1
0 0 0 : 0 0 0
0 l l i l l l
3 + 7 = 10

O uso de distância suave no processo de decodificação ' será ilustrado a seguir.

No exemplo abaixo é assumido Q = 3 e um código com exatamente duas palavras, 0 ou 1, para o qual d=l e R=l.

A figura 2.3 mostra as possíveis saídas obtidas após o demodulador.

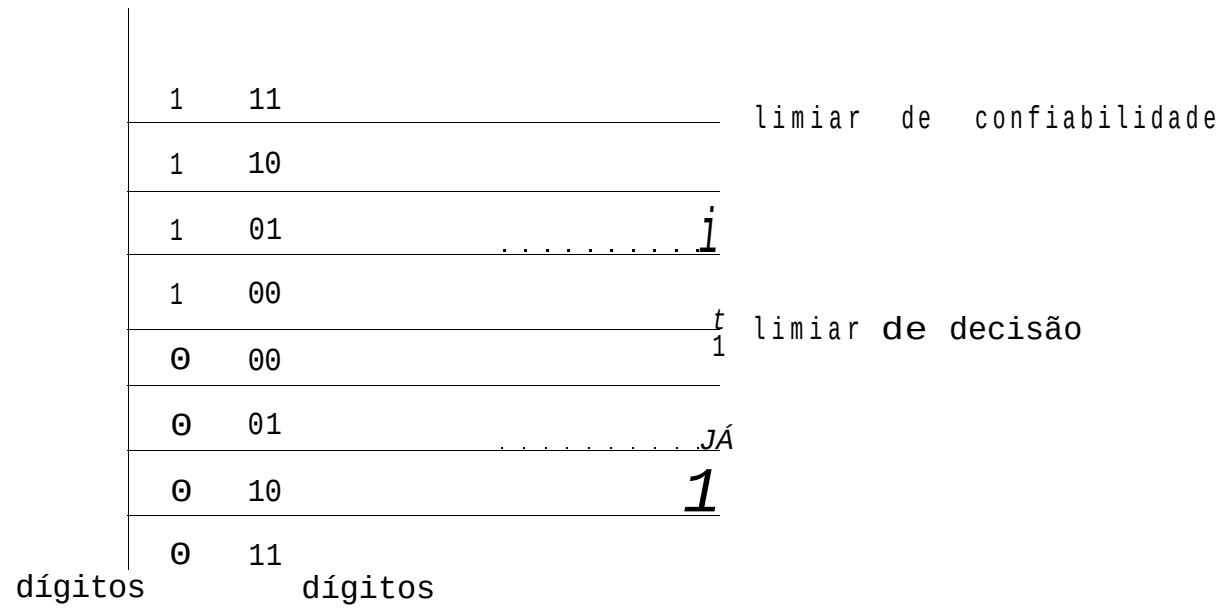


Figura 2.3 - SAÍDAS DO DEMODULADOR EMPREGANDO DECISÃO SUAVE.

É admitido que a palavra 1 é transmitida, e que devido* ao ruído do canal, 0 0 0 é a saída do demodulador. As duas possib_i lidades de plena confiança para a saída do demodulador são 0 1 1 e 1 1 1, de maneira que a distância de decisão suave d_g/ para cada caso é

$$d_s(0\ 0\ 0\ ,\ 0\ 1\ 1) = \mathbf{3}$$

$$\mathbf{d_s\ (0\ 0\ 0\ ,\ 1\ 1\ 1) = 4.}$$

Desta maneira, 0 será selecionado incorretamente como[1] tendo sido a palavra transmitida, quando é empregado um decodificador por distância suave mínima.

Agora, será considerado um novo exemplo no qual se assume as palavras código como sendo 0 0 e 1 1, resultando em um código com d=2 e R=0,5. É admitido que a palavra 11 é transmitida, mas 0 0 0 1 1 1 é recebida na saída do demodulador, ou seja, o ruído afetou o primeiro dígito da palavra código.

As duas saídas de plena confiança são agora [1] 0 1 1 0 1 1 e 1 1 1 1 1 1, de modo que

$$d_s(0\ 0\ 0\ 1\ 1\ 1,\ 0\ 1\ 1\ 0\ 1\ 1) = 10$$

$$d_s(0\ 0\ 0\ 1\ 1\ 1,\ 1\ 1\ 1\ 1\ 1\ 1) = \mathbf{4.}$$

Portanto, 1 1 será selecionada corretamente, e um único erro foi corrigido, embora o código possua d=2.

Este tipo de deteção com decisão suave é particularmente aplicável â decodificação por distância mínima (máximo de verossimilhança) para códigos de bloco, e também para códigos convolucionais utilizando o algoritmo de Viterbi ou decodificação sequencial.

2.2- ALGUNS PROCEDIMENTOS SUBÕTIMOS

Nesta seção são descritos algoritmos de decodificação[1] de códigos de bloco que empregam decisão suave e que são subótimos, no sentido que o ganho teórico máximo possível não é atingido."Em[1] alguns casos isto se deve ao fato do uso de um numero finito de regiões de quantização, de maneira a viabilizar a implementação prática.

Tais procedimentos, embora subótimos, representam uma melhoria considerável no desempenho com relação a sistemas que utilizam decisão abrupta.

2.2.1 - Decisão Suave na Decodificação de Códigos de um Onico Digito de Paridade

O procedimento descrito a seguir é devido a Farrell e Kalligeros |l2 |# onde se utilizou como exemplo um código de bloco' (8,7,2) e um detetor com decisão suave quantizado em 8 regiões, cujos níveis são +0,25V , +0,5V, +0,75V para sinais de +ÍV.

O decodificador deve operar de acordo com a seguinte ' sequencia:

1) - avaliar os dígitos de decisão k-^k,^.. .k^k-yC e seus correspondentes "bytes" de confiabilidade.
2) - Recalcular o dígito de paridade *c*-^ a partir dos dígitos de informação e compará-lo com o valor recebido c^.

a) Se c-^ estiver correto, ê assumido que não ocorreram erros, e a palavra é liberada.

b) Em caso contrário, inverter o dígito de mais baixa confiabilidade, de modo a corrigir o erro isolado mais provável, e liberar então a palavra.

A curva de desempenho, obtida por simulação em computa

dor, é mostrada a seguir na figura 2.4 abaixo.

Para relações sinal/ruído maiores que -2,5dB observa-se um ganho em relação ao sistema sem codificação, o qual é superior ' aquele obtido por um código de Hamming de mesmo comprimento e/ou [1] eficiência.

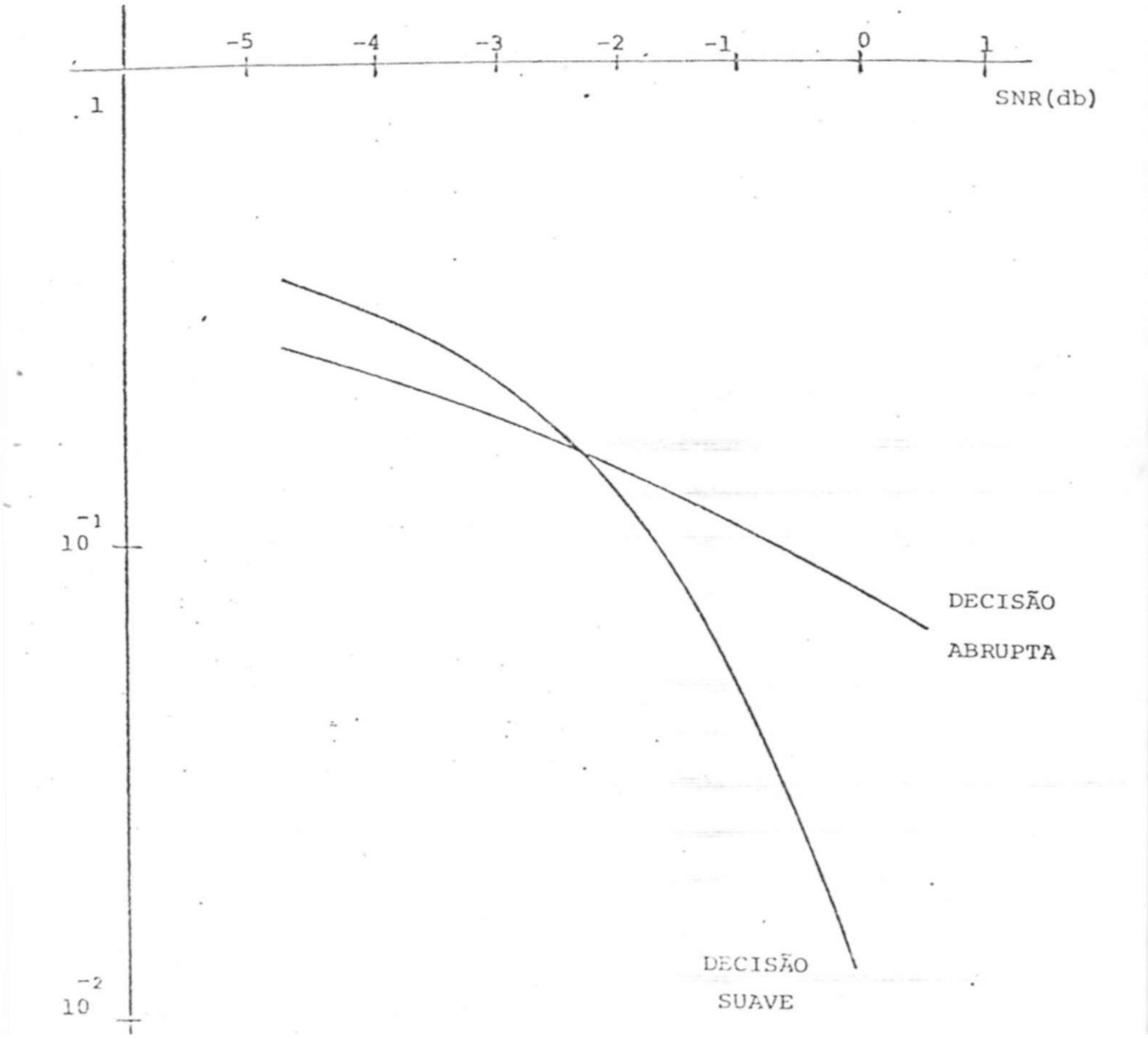


Figura 2.4 - CURVAS DE DESEMPENHO DO ALGORÍTMO DE FARRELL E KALLIGEROS.

2.2.2 - Decodificação por Distância Suave Mínima

Decodificação por distância mínima usando decisão suave foi empregada em um sistema estudado por Farrell e Munday (1976) [112]. É basicamente um sistema de comunicação em HF sem canal de retorno. As características do sistema são:

1) - baixa taxa de dados, aproximadamente 50 bits/seg.
2) - Um código produto [361] é empregado, com n = 15x15 = 225 e d = 3x3 = 9 codificado e decodificado em cascata.
3) - Intercalamento de bits para combater a mistura de erros aleatórios e "burst" do canal de HF.
4) - "Sequence inversion keying" [11] de aproximadamente 100 bits / seg, para espalhar o espectro num canal de HF com 3kHz. Para tal, uma sequência - m de 15 dígitos foi empregada.
5) - Modulação ASK, com deteção coerente empregando PLL.
6) - Correlação dos dígitos da sequência m para determinar se a sequência está invertida ou não (dígito de decisão) e o grau de correlação na presença de erros do canal (dígitos de confiança); o que é equivalente a um esquema de demodulação por decisão suave com Q=4.
7) - Decodificação por distância de decisão suave mínima das linhas e colunas componentes do código produto, por seleção alternada adaptativa das linhas e colunas.

O código empregado é um código produto, como mostrado na figura 2.5.

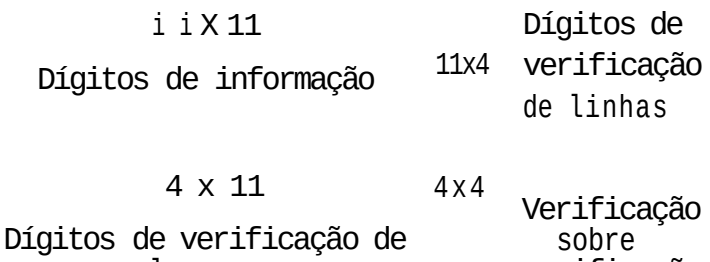


Figura 2.5 - CÓDIGO PRODUTO.

O desempenho do sistema foi muito bom em testes de ruído

Com simulação digital de erros, uma taxa de erros de 2x10 ^ nos dígitos da sequência - m foi reduzida para 4xyL0 ^ nos dígitos de decisão na saída do correlador, e ainda reduzida para cerca de 4x10 ^ depois da decodificação por decisão s^lave.

A figura 2.6 mostra as curvas para testes com ruído branco.

-4 -

O ganho do código em P_g = 10 e aproximadamente 12.3dB, enquanto que o ganho teórico máximo ê cerca de 14.7dB. A diferença não é grande tendo em vista que esta degradação possibilita a realização prática.

2.2.3 - O Algoritmo de Harrison

Pode ser mostrado 140 | que o processo de decisão abrupta num canal gaussiano resulta numa degradação no desempenho de aproximadamente 2dB comparado com o receptor ótimo. Segue-se que , para este canal, o máximo de melhoria a ser esperada por introduzir decisão suave no sistema, é equivalente a esta degradação. Harrison

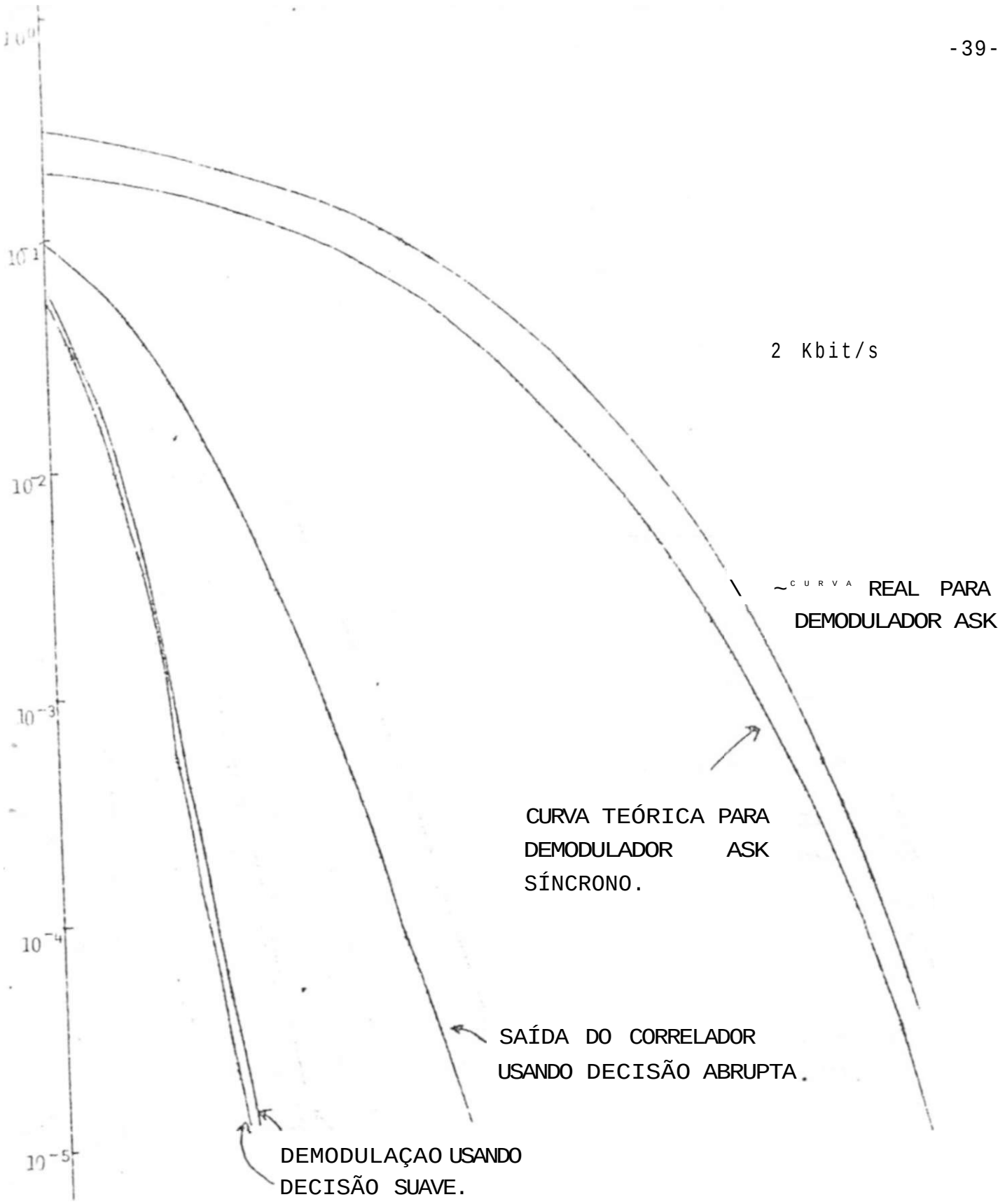


Figura 2.6 - CURVAS DE DESEMPENHO NA DECODIFICAÇÃO POR DISTÂNCIA SUAVE MÍNIMA.

|l7| propôs um algoritmo de decodificação empregando decisão suave' que permite um ganho de aproximadamente ldB, o que é um bom resulta do tendo em vista a relativa simplicidade de implementação do mesmo. Neste esquema, a palavra recebida consiste da informação básica da decisão abrupta com "marcas" de confiabilidade (indicando quais as prováveis fontes de erro). Esta informação de confiabilidade dispo- nível ao decodificador indica simplesmente quais os bits na palavra código que são os mais prováveis de terem sido afetados por erros.

Utilizou-se um código de Hamming (7,4,3) como exemplo [1] para aplicação desta técnica. Desde que a complexidade do decodifi- cador cresce rapidamente com o aumento do numero de níveis de quan- tização, um esquema com 4 níveis ê descrito.

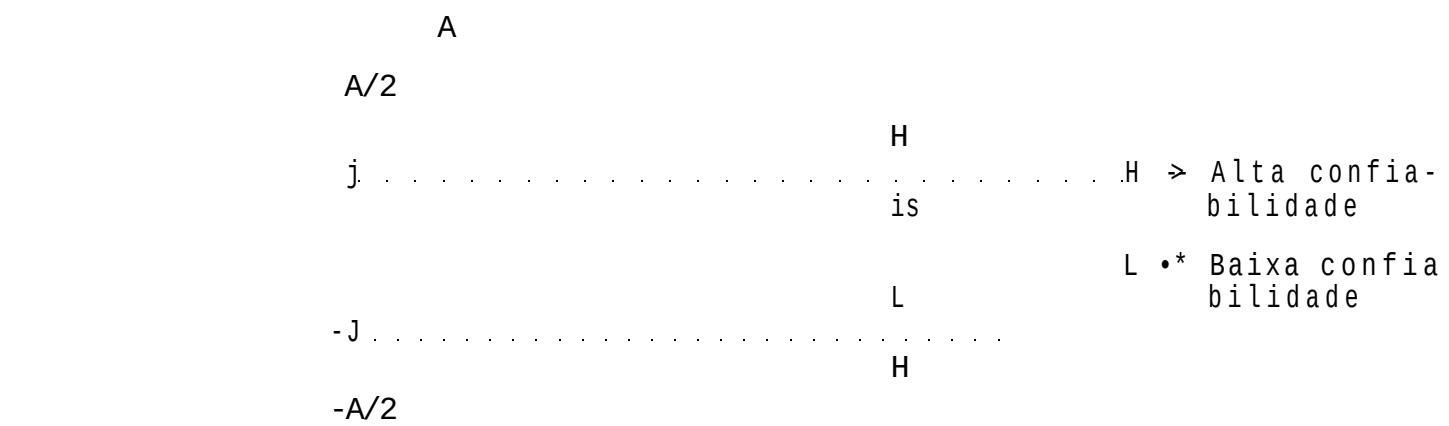


Figura 2.7 - REGIÕES DE CONFIABILIDADE.

A posição dos limiares relativos a confiabilidade é mu_i to importante nas considerações de informação de confiabilidade. Se a tensão J ê muito pequena, poucos bits recebidos serão de baixa' confiabilidade, transportando pouca informação ao decodificador. Em contrapartida, se J *ê* uma tensão elevada, praticamente todos os bits recebidos serão de igual (baixa) confiabilidade, o que é equi- valente a decisão abrupta.

Existe uma posição ótima, para cada código corretor, a qual maximiza a informação de confiabilidade entregue ao decodifica

dor. As curvas de desempenho obtidas para 3 limiares de decisão diferentes são apresentadas na figura 2.8, onde é observada a existência deste valor ótimo para J entre 40% e 50% da amplitude dos pulsos binários transmitidos.

O procedimento proposto consiste basicamente dos seguintes passos:

1) - Se a palavra recebida possui 0,1 ou mais que 2 dígitos com baixa confiabilidade, o decodificador atua desprezando a informação probabilística, ie., empregando decisão abrupta.

2) - Se a palavra recebida tem 2 dígitos com baixa confiabilidade, então a síndrome s_ ê calculada:

   a) Se s = 0, então a palavra é suposta correta e é liberada.

   b) Se s indica um erro em uma posição de baixa confiabilidade, então a mesma é corrigida, e a palavra liberada.

   c) Se s indica um erro em uma posição de alta confiabilidade, então dois dígitos de baixa confiabilidade são invertidos e a síndrome é recalculada. Se o novo valor é s = 0, é considerado que dois erros foram corrigidos, e em caso contrário, ocorreu um erro numa posição de alta confiabilidade e o decodificador deve operar normalmente.

Desta maneira o desempenho do decodificador com decisão suave deve ser pelo menos tão bom quanto o receptor equivalente empregando decisão abrupta, para todos os valores de relação sinal / ruído.

2.3- DECDDIFICAÇÃO POR DECISÃO SUAVE PARA CÓDIGOS DE BLOCO USANDO UMA TRELIÇA

Será descrito um algoritmo introduzido em 1978 por [1] Jack Wolf 1381 *o* qual faz uso de decisão suave na decodificação por

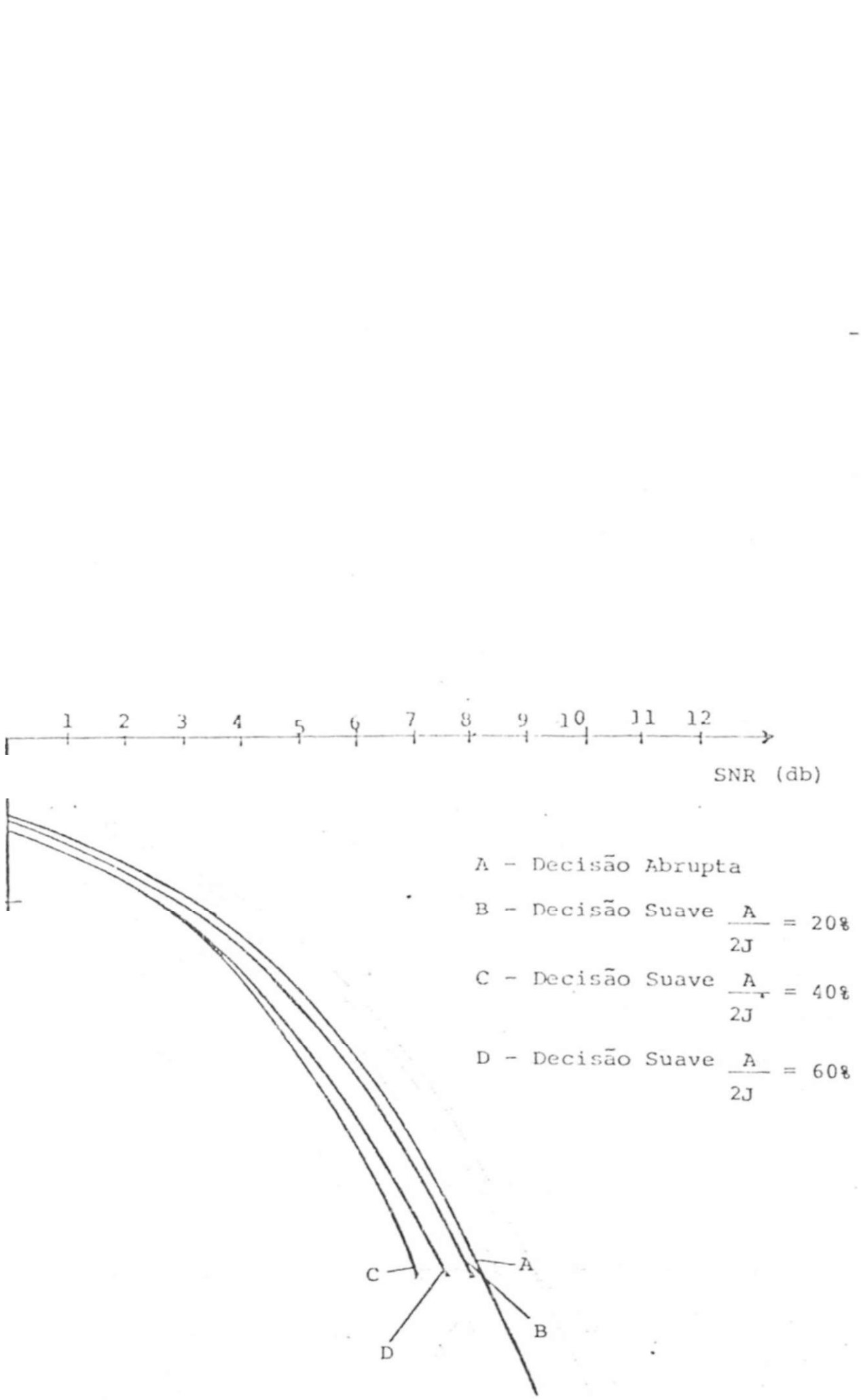


Figura 2.3 - ALGORÎTMO DE HARRISON - CURVAS DE DESEMPENHO.

máxima verossimilhança de qualquer código de bloco linear (n,k,d)sobre GF(q) , e que pode ser implementado usando uma treliça tendo não mais que q^{n-k} estados.

2.3.1 - Considerações Gerais

O algoritmo apresentado faz uso da técnica de decisão suave, o que está associado ao fato de que emprega números reais (e.g., a saída analógica de filtros casados para os sinais) relacionados com cada símbolo correspondente da palavra código.

Será mostrado que a decodificação de códigos de bloco usando uma treliça é de grande interesse na decodificação de códigos com alta eficiência, visto que a complexidade da treliça é limitada superiormente por uma função do número de símbolos de paridade. Também, algumas simplificações resultam quando são considerados códigos lineares particulares, tais como códigos cíclicos ou códigos' produto [38] .

2.3.2 - Treliça Associada a um Código de Bloco Linear

Seja a matriz de verificação de paridade [H] de um código de bloco linear (n,k,d) em GF(q) considerada sob a forma $[H] = [h_1, h_2, \ldots, h_n]$, onde cada vetor coluna h_i possui (n-k) coordenadas em GF(q), ou seja, $h_i^T = (h_{i1}, h_{i2}, \ldots, h_{i,n-k})$ $i = 1, 2, \ldots, n$.

Considere um diagrama de árvore como uma coleção de nós (estados) interligados entre si por ramos unidirecionais. Cada estado a uma dada "profundidade" k é indicado por uma (n-k)-upla $S^{(k)}$ com elementos em GF(q). As linhas são traçadas da profundidade k para a profundidade k+1, considerando os nós $\{S^{(k)}\}$ a uma dada profundidade gerados como indicado no diagrama da figura 2.9.

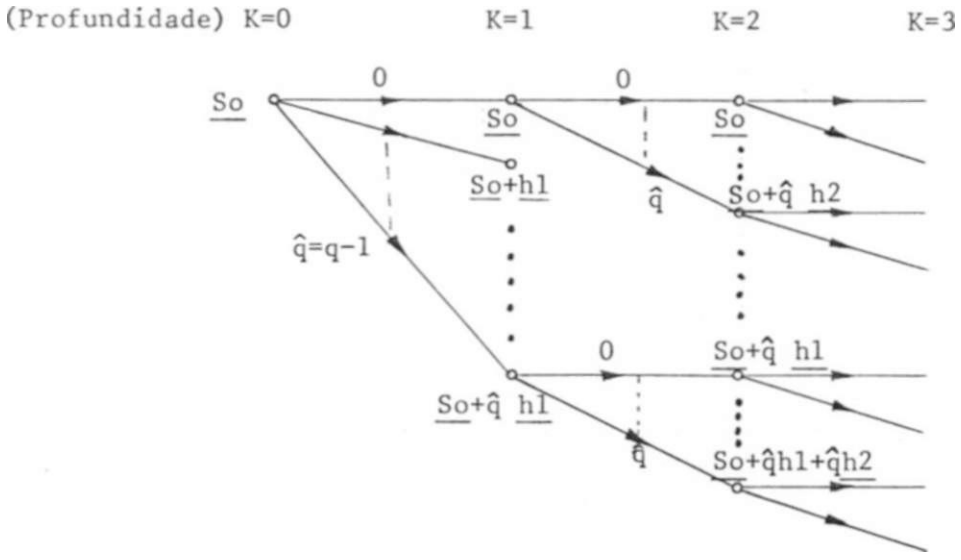


Figura 2.9 - DIAGRAMA DE ÁRVORE PARA UM CÓDIGO DE BLOCO.

Deste modo é possível verificar que a uma dada profundidade k existem os seguintes nós:

§0

$s_0 \oplus a_j \; h_t$ í < *i* < k

â₀ © ª j i ^ i © ª j f íif ¹ S ¹ * ¹ *Z* *

etc

De uma maneira resumida, é possível escrever que a uma dada profundidade k existem os nós:

$$S^{(k)} = S_{(0)} \oplus \sum_{i=1}^{k} h_i \quad \text{com} \quad a_i \text{ e } GF(q) \qquad (2\text{-}1)$$

Cada estado é identificado por uma (n-k)-upla em GF(q), de modo que o número de estados em qualquer profundidade não pode exceder q^{n-k}, o número de (n-k)-uplas distintas com elementos em

GF(q). Deste fato segue-se que é possível fazer uma modificação na figura 2.9, que consiste em juntar os nos indicados pela mesma (n-k)-upla, resultando em um diagrama de treliça.

Definição 2.3.2 - Uma treliça associada a um código de bloco em GF(q) é uma coleção particular de nós (ou estados) interligados por ramos unidirecionais, e agrupados em conjuntos indexados por um parâmetros k, k = 0,1,2,...,n.

Um nó indexado por um dado valor de k é dito estar a uma profundidade k. As linhas serão traçadas entre certos pares de nós a uma profundidade k e a uma profundidade k+1, para k = 0,1,2,...,n-1; com a direção do ramo da profundidade k para a profundidade k+1. Para qualquer profundidade k existirão no máximo q^{n-k} nós, os quais serão identificados por (n-k)-uplas, $S^{(k)}_i$ com elementos em GF(q) para certos valores de i. Todas as q^{n-k} (n-k)-uplas sao ordenadas de 0 a $q^{n-k}-1$, com 0 referindo-se a (n-k)-upla toda zero. $S^{(k)}_i$ é interpretada como a i-êsima (n-k)-upla desta lista ordenada. Desde que nem todas as (n-kj-uplas podem corresponder aos nós a uma profundidade k, deve ser considerado um conjunto P^{n-k}_i, subconjunto dos inteiros $\{0,1,2,...,q^{n-k}-1\}$, correspondendo a essas (n-k)-uplas que estão relacionadas a nós a uma profundidade k.

2.3.3 - Construção da Treliça

A treliça associada a um código possui n+1 profundidades, desde k=0 a k=n, e possui a cada profundidade q^{n-k} destas dos possíveis. Cada nó é caracterizado pela profundidade k e pelo estado $i=0,1,2,...,q^{n-k}-1$; de modo que cada nó origina q outros. Cada nó em qualquer profundidade k é obtido como combinação linear dos vetores h- correspondentes a profundidades menores que

k, de acordo com a equação (2-1). É importante notar que S^(k) não é univocamente definido, pois diferentes caminhos (no máximo q) podem levar a um mesmo estado.

O número de nos que são definidos a uma profundidade k ê dado por q^{LI}, onde LI é o número de vetores coluna $\{h^\}^{k}$ li ~ nearmente independentes.

A construção da treliça associada a um código de bloco' é feita de acordo com o procedimento descrito a seguir:

1) - Na profundidade k=0, a treliça contém somente um nó, chamado' SQ(0), a (n-k)-upla toda nula.

2) - Para cada k = 0,1,...,n-1, a coleção de nós na profundidade ' k+1 é obtida a partir da coleção de nós na profundidade k pela fórmula:

$$S(k+1) = S_{\pm}(k)\ 0\ a_{£}\ h_{k+1}\ ieP_{k},\ a^eGF(q) \qquad (2-2)$$

Para cada *i ^ v '* as li-nnas de conexão são traçadas entre o nó (k) e os q nós na profundidade $k+l_f$ gerados empregando a equação (2-2) acima. Cada linha é determinada pelo valor part_icular de que gerou S^(k+1) a partir de S^Ck),

3) - Expurgar quaisquer nós que não tenham um caminho para o estado todo zero na profundidade k=n, e retirar todas as linhas traçadas para estes nós expurgados.

Há uma correspondência biunlvoca entre cada palavra código e a sequência de $^{a}j_{c}$'^{s}' sobre qualquer caminho, do nó todo zero na profundidade 0 para o todo zero na profundidade n. Existem $^{1}_{k}$ q caminhos distintos através da treliça, e cada um deles corresponde a uma palavra código. Assim, a treliça fornece um método compacto de catalogar todas as palavras código; cada palavra código ' distinta é representada por um caminho distinto na treliça.

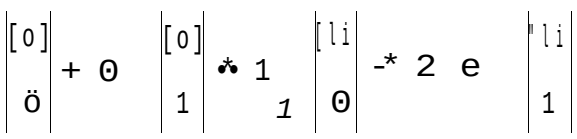
Exemplo 2.1 - O procedimento descrito é melhor ilustrado neste exempio, onde é considerado o código binário (5,3,2) introduzido no

exemplo 1.1, cuja matriz de verificação de paridade foi determinada no exemplo 1.2 como sendo

$$[H] = \begin{matrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{matrix}$$

Neste caso, n=5, k=3 e q=2, de modo que $q^{n-k}=4$ P_k C{0,1,2,3}.

Os nós são associados aos números inteiros na forma indicada:



A treliça para este código é apresentada na figura 2.11,

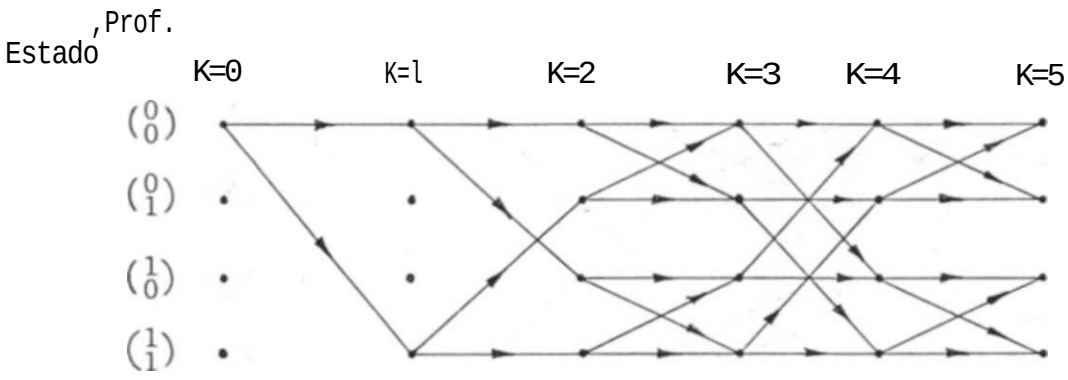


Figura 2.10 - TRELIÇA NÃO EXPURGADA PARA O CÓDIGO (5,3,2).

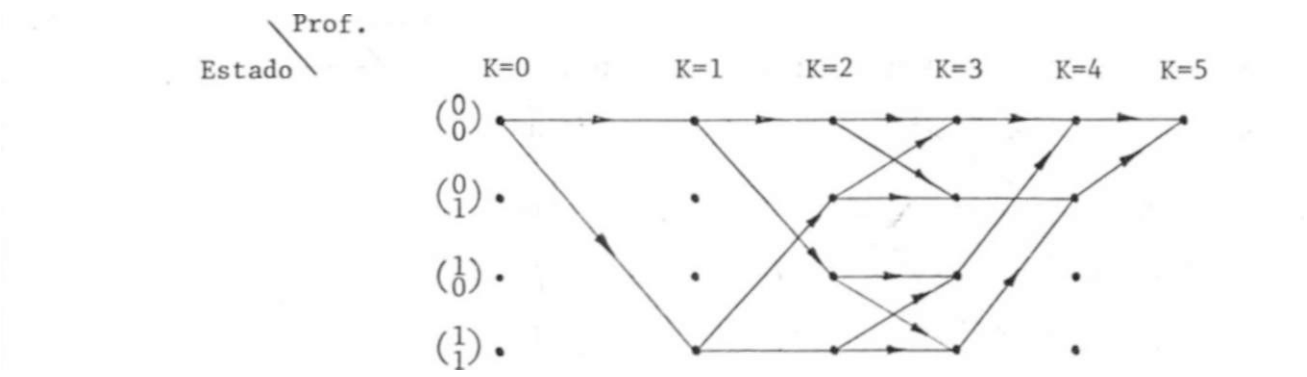


Figura 2.11 - TRELIÇA PARA O CÓDIGO DE BLOCO (5,3,2) (EXPURGADA).

A referência |25| apresenta alguns algoritmos, os quais permitem uma redução na complexidade da busca em uma treliça associada a um código de bloco. Entretanto, é evidente que deve existir um compromisso entre a complexidade e eficiência dos sistemas que empregam tais algoritmos, o qual é analisado através de simulação, na referência citada. Alguns destes procedimentos serão apresentados na seção 2.6.

2.3.4 - Decodificador de Viterbi usando a Treliça

A decodificação é executada baseada na énupla recebida r, com componentes reais. É assumido que não há interferência inter simbólica, de modo que a j-ésima componente de r depende somente da j-ésima componente da palavra transmitida c.

Também é assumido que a contribuição do ruído em cada uma destas componentes é descrita por variáveis aleatórias estatisticamente independentes N., com fdp $f_{N_i}(\ )$ $i = 1,2,\ldots,n$. Então o logaritmo da função de verossimilhança, dado a palavra código transmitida, é da forma

$$\ln f_M(r \mid c) = \sum_{i=1}^{n} \ln f_{N_i}(r_i - c_i) = \sum_{i=1}^{n} Z_i(r_i, c_i) = Z(c) \qquad (2\text{-}3)$$

Para uma dada sequência recebida r, é óbvio que uma decodificação com máxima probabilidade a posteriori é aquela que encontra a palavra código c que maximiza Z(c). Seria emprego da força bruta verificar por exaustão, i.e., tentando todas as q possíveis palavras código.

A seguir é descrito um algoritmo recursivo pelo qual muitas palavras código podem ser descartadas no processo de decod^

ficação quando na determinação da palavra código que maximiza $Z(c)$.

A cada nó $S^{ik})$ na profundidade k, deve ser associado um número real $V(S^{(k)})$ de acordo com as seguintes regras:

1) - Para a profundidade k=0, faça $V(S_{Q}(0)) = 0$.

2) - V£ e $^{p}k+l'^{forme}$ $V(S^{(k+l)})$ a partir de $VÍS^{k))}$ da seguinte maneira:

$$V(S_{\_\varepsilon}(k+l)) = \max_{a_j \in GF(q)} \{V(S_{(k)}) + Z_{k+1}(r_{k+1}, a_j)\} \qquad (2\text{-}4)$$

com $^{i \in P_k}$ j^{i} onde $P^{} \subset P^{}$ é um subconjunto de Índices tal que, para algum a de GF(q) tem-se

$$S_{\varepsilon}(k+1) = \delta_{ik)} \oplus a h_{k+1}$$

3) - Retenha somente aquele caminho para o qual a fórmula (2-4) forneça o máximo.

4) - Em k=n, a sequência de α_i's obtida simplesmente ligando o caminho do estado todo zero na profundidade k=0 até o estado todo zero na profundidade k=n, corresponde ã palavra código c que maximiza $Z(c)$.

2.4 - DECISÃO SUAVE NA DECODIFICAÇÃO QUANDO A FONTE TEM DISTRIBUIÇÃO DE PROBABILIDADE DESCONHECIDA

Será apresentado um procedimento que faz uso de coeficientes probabilísticos condicionais de confiança como medida de confiabilidade, resultando em uma regra de decodificação de máxima verossimilhança condicional.

2.4.1 - Noções Preliminares

Considere um sistema de comunicação no qual os símbolos[1] da fonte binária tem uma distribuição de probabilidade desconhecida' e são transmitidos em blocos de n dígitos através de um canal sem

memória, caracterizado por uma função de verossimilhança fíy^lj), ' onde j e {0,1}, y c R.

Uma vez a palavra binária transmitida e uma sequência [1] Y = (yi'***'Y_n^ recebida, o uso do método de Neymann-Pearson clássico l131 para cada y^ com um nível de confiabilidade prescrito , resulta em um dígito binário x^. A decisão x^ não considera qua_l quer medida de conclusividade relativa a uma observação particular' y^; pois diferentes Y i ' s podem ser decodificados em um mesmo x^, com as mesmas probabilidades de erro tipo I e tipo II, podendo cada uma das decisões possuir diferentes graus de conclusividade.

A maioria dos procedimentos de decodificação que empregam decisão suave, tratam de casos quando a distribuição da fonte é conhecida, de modo que é possível ser usada a probabilidade a posteriori como medida de confiabilidade. Contudo, é conhecido que em vários problemas práticos esse não é o caso. Wolfenson e Rocha 139 | propuseram fazer uso de coeficientes de confiança condicional de ' Kiefer | 2l|» os quais têm interpretação frequentista, como uma med_ida de confiabilidade.

2.4.2 - Decodificação usando Confiança Condicional

O uso do lema fundamental de Neymann-Pearson permite de codificar com probabilidades de erro prescritas, cada componente da palavra recebida y_, resultando num vetor x. Como discutido, este procedimento não expressa nenhuma conclusividade levando em conta ' uma observação y^ particular.

Denotando por a e 3 as probabilidades de erro tipo i e I I , a conclusão provida pelo método é simplesmente que cada y^ deve ser decodificado em x^ com probabilidades de erro a e 3 Kiefer |21| propôs um esquema de atribuir uma confiabilidade a cada decisão *x-* em termos de um coeficiente de confiança condicional.

Considere uma partição II do espaço de observações R, onde H = {PI > para j *c* {0,1}, s e S (S um conjunto de índices) é também definido Ü^{s} através da relação $II^{s}=IIQ\ U\ 1\ ^{\wedge}$.

Observando um dado c R, é procurado o elemento ^{1} II_{j}^{S} c II no qual caiu, e assim $y\_^{\wedge}$ ē decodificado em j com ^{1} confiança condicional* dada por $R = P\{dec_{j}|n^{s0}_{r}j/«$

É assumido ainda que n é escolhida de tal maneira que

$$P\{dec\ 0|n^{s°},0\} = P\{dec\ l|n^{s°},l\} \qquad (2\text{-}6)$$

R sendo uma probabilidade, pode ser interpretada em termos de frequência relativa. Repetindo o experimento independentemente um número muito grande de vezes de modo que se tenha um grande número' de ocorrências de II , R_{s0} fornece aproximadamente a fração das sO vezes que II ocorreu e nas quais foi tomada uma decisão correta. Isto fornece, portanto, uma maneira objetiva de atribuir uma medida de confiabilidade a cada decisão NP. Assumindo que $\underline{y} = (y^{\wedge}« \bullet \#y)$ si foi recebido e que seja verificado y. e H para cada ^{1} i = l,2,...,n, Segue-se que cada y. pode ser decodificado pelo ' procedimento convencional NP com a e $, e a medida de confiabilida de da decisão é dada por R • - A questão agora é como combinar as confiabilidades R_{si}., i = 1,2,...,n de modo a atribuir uma medida' de confiabilidade a sequência decodificada $x = (x\text{-}^{\wedge}/\text{-}..,x_{n})«$ Se x é a sequência decodificada, levando em consideração que o canal é sem memória, é possível escrever

$$P\{dec\ x\ in\ I_{i}.,\ i=l,\ n,\ x\} = \prod_{i=l}^{n} P\{dec\ x_{i}.|JI_{i}.,\ x_{i}.\}$$

$$= \prod_{i=l}^{n} R_{si}. = R(x) \qquad (2\text{-}7)$$

* Veja lista de símbolos e abreviaturas.

Desta maneira é associada uma confiabilidade dada por $R(x) = \prod_{i=1}^{n} R_{s_i}$ à sequência inteira decodificada.

2.4.3 - Exemplo: Aplicação a um Canal com Ruído Branco Gaussiano

O método de decodificação apresentado é uma solução de compromisso entre minimizar a probabilidade de erro por dígito e escolher a palavra código mais provável. É possível visualizá-lo como um procedimento em dois passos, onde primeiramente o critério NP é usado, com probabilidades de erro prescritas, e então a estrutura do código é empregada juntamente com a medida de confiabilidade de modo a encontrar a palavra código mais provável, condicionada à observação.

Nesta aplicação será considerado um canal perturbado por ruído branco gaussiano com média nula e variância unitária. A fonte bipolar transmite +1V ou -IV correspondendo aos símbolos 1 e 0, respectivamente.

Também é assumido iguais probabilidades de erros a = 8 a decisão NP é $y_1 > 0$ com a = 3 = Q(l) =0,16.

A partição II é escolhida de modo a permitir a máxima variabilidade nos coeficientes de confiança. No exemplo, S é escolhido como o conjunto dos possíveis valores de R_S, i.e., II^S será indexado de modo que R_s = s. Portanto,

$$R = s = P\text{ídec}\{j|y,j\} = \frac{P\{dec\ j'Vlj\}}{P\{y|j\}} \qquad (2\text{-}8)$$

ou seja,

$$R = s = \frac{l}{1 + \frac{P\text{ídec}(l\text{-}j,y|j)}{P\text{Ídec}(j,y|j)}} \qquad (2\text{-}9)$$

Assim, $s = \frac{1}{1 + e^{-2y}}$, e n^s é escolhida como

$$n^s = \{* \ 1/2 \ In \ (s^{-1} - l) \quad para \quad s > 1/2\}. \qquad (2\text{-}10)$$

Suponha, por exemplo, que $y_ = (-0.2, -0.7, -0.7, 0.2, 1.'1)$ de forma que o procedimento NP bit a bit resulta em $x^{=}(0,0,0,1,1)$, com coeficientes condicionais (0.6,0.8,0.8,0.6,0.9) e com confiabilidade R(x) = 0,207.

Observe que devido a escolha de II, +0.2V pode ser decodificado em 1 ou 0 com medida de confiança 0.6 ou 0.4, respectivamente. Supondo que o código (5,3,2) descrito no exemplo 1.1 é usado no contexto acima, é possível fazer uso do diagrama de treliça associado (veja exemplo 2.1, figura 2.11) para encontrar a palavra códi. go de maior confiabilidade.

Os passos da decodificação estão esquematizados abaixo.

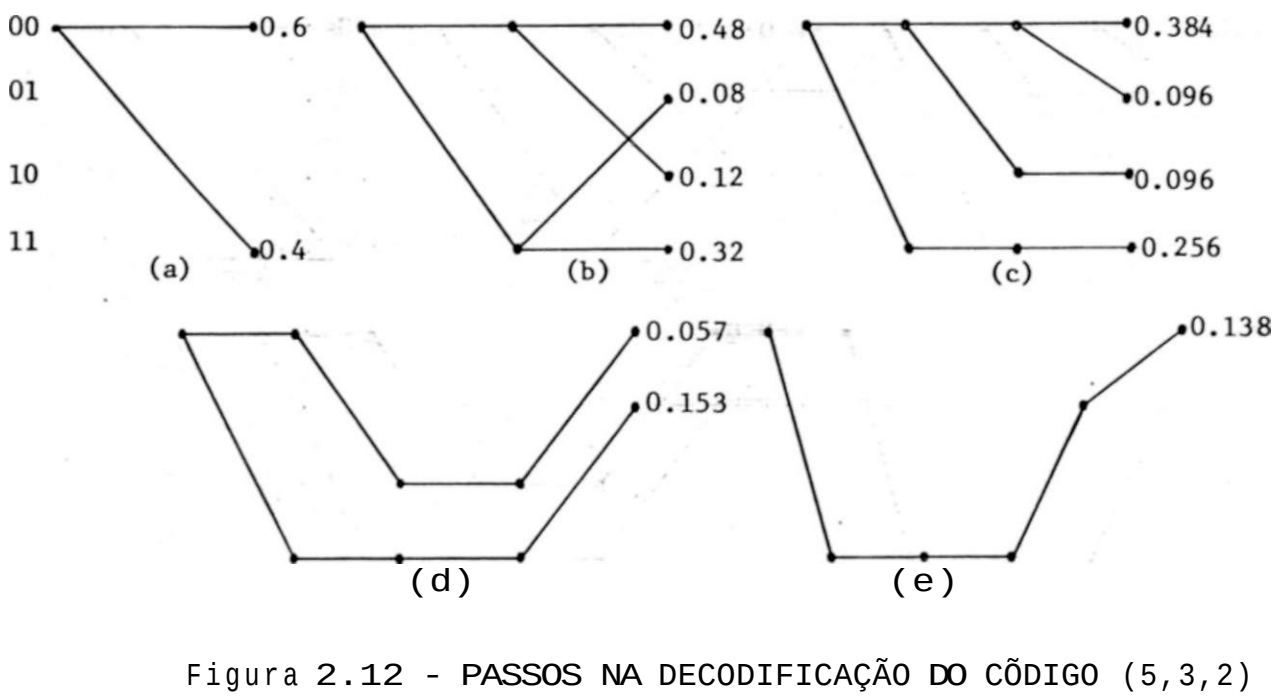


Figura 2.12 - PASSOS NA DECODIFICAÇÃO DO CÕDIGO (5,3,2) USANDO MÉTODO DE WOLFENSON-ROCHA.

Observe que um erro foi corrigido no primeiro dígito ' (R -, =0.6), pois, uma decisão abrupta estima a palavra transmitida

como (0,0,0,1,1) com R(x) = Q.2Q7, enquanto que a saída do decodificador estima (1,0,0,1,1) com R(x) = 0.138. Deve também ser observado que a posição alterada foi a de menor confiabilidade, e que uma mudança na quarta posição (R_s4 = 0.6) não é permitida pela estrutura do código.

2.5 - DISTÂNCIA GENERALIZADA

Será mostrado que distância generalizada constitui uma extensão da idéia de apagamento discutida anteriormente, a qual permite uma maior flexibilidade no uso da informação de confiabilidade. O emprego de distância generalizada permite que algoritmos algébricos de decodificação possam ser prontamente adaptados para fazerem[1] uso da decisão suave. Desta forma, algumas características de decodificação probabilística são introduzidas sem sacrificar a atratividade dos procedimentos algébricos.

Serão analisados dois algoritmos para decodificar por distância mínima generalizada, sendo o segundo procedimento descrito, proposto pelo autor. Para o primeiro procedimento, introduzido por Forney [14] em 1966, foram também desenvolvidas cotas exponenciais para a probabilidade de não decodificar corretamente.

2.5.1 - Medidas de Distância

Os algoritmos de decodificação são formulados, geralmente, em termos da escolha da palavra-cõdigo mais próxima da palavra recebida, onde a "proximidade" é definida por alguma medida de distância. Duas importantes medidas de distância são apresentadas nesta seção. A distância de Hamming, já introduzida na definição 1.2.2b, será reapresentada de uma maneira mais formal.

Definição 2,5.1a - A distância de Hamming $D_H(f,3)$ entre duas palavras $f^e 2^n$ é simplesmente o número de posições em que elas diferem

iA, $D_H(f,g) = I\ d_H(f_{i/}g_i)$

Onde

$$V^f i\ '\ 9\ i > \quad \begin{cases} 0 & \text{se } f_\pm = g_\pm \\ 1 & \text{se } f_\pm\ i\text{-}\ g_\pm \end{cases}$$

Suponha que o receptor decide, para cada amostra recebida, qual das q letras foi a mais provavelmente enviada, uma escolha [1] que corresponde a um elemento do GF(q) rotulado por r^. Esta escolha é então associada a uma classe de confiabilidade c_. de acordo[1] com a certeza que se tem da mesma ter sido correta. A cada classe c.. são associados dois parâmetros 3_{CD} e 3_{eu} tais que $0<3_{CD}<3_{CD-e}<1$ • Contudo, será mostrado que somente a quantidade $a_* = 3_{D\ eD} - 3_{CD\ f}$ • chamada de peso da classe c_{*}, é significante. Vê-se claramente que [1] (Vj) é verificada a relação $0 < a_. < 1$.

Definição 2.5.1b - Distância generalizada D^E/f.) entre a palavra' recebida r e a palavra código f é definida como sendo

$$D_{(r,f)} = \sum_{i=l}^{n} d_{G\ 1\ 1}(r_.,f_.)$$

Onde

$$d_{\circ}(r_.,f_.) = \begin{cases} 3_{CD} & \text{se } r_{.1} = f_{.1}, r_{.i} \text{ e } c_D \\ 3_{eD} & \text{se } r_{.1} \neq f_{.1}, r_{.i} \text{ e } c_D \end{cases}$$

Deve ser observado que esta distância não constitui uma métrica | 2 |, enquanto que D^ representa uma métrica.

Em geral, as classes para as quais a=l serão referidas como classes de plena confiança, e aquelas para as quais a=0 como' classes não confiáveis. Valores intermediários de correspondem' a níveis intermediários de confiabilidade. Também serão ordenados os números das classes em ordem decrescente com relação ao seu peso, de

maneira que se j < k, então $\alpha_j > a_k$.

2.5.2 - Decodificação por Distância Mínima Generalizada I

Algumas das propriedades da distância generalizada serão discutidas a seguir, as quais permitem o estabelecimento de um decodificador usando distância mínima generalizada. Cotas exponenciais de erros para o algoritmo estabelecido serão também desenvolvidas.

2.5.2.1 - Propriedades da Distância Generalizada

Inicialmente suponha que durante a transmissão de uma palavra f de um código de bloco (n,k,d), o numero de letras recebidas corretamente ($r_i = f_i$) e postas na classe c_i é n_{i3}, e o número das recebidas incorretamente ($r_i \neq f_i$) e postas nesta classe c_j é n_{eu}..

Teorema 2.1 - Se f é enviada e n_{cj} e n_{ej} são tais que se verifica $\sum_{j=1}^{J}(l-a_j)n_{cj} + (l+a_j)n_{ej} < d$, então $D_b(r,f) < D_G(r,g)$ para todas palavras código $3 \neq f$.

Prova - Pela definição de distância generalizada segue-se que

$$D_\_(r,f) = \sum_{i=1}^{n} n_{i3} + \sum_{i=1}^{n} n_{i1} \qquad (2\text{-}11)$$

Para qualquer palavra código $g \neq f$ serão considerados três conjuntos de índices, de acordo com

s_0 se $f_i = g_i$

s_j se $f_i \neq g_i$, $r_i = f_i$ com $r_i \in c_3$.

s_e se $f_i \neq g_i$, $r_i \neq f_i$ com $r_i \in C_D$

0 numero de letras s_i e s_{i+1} satisfazem as relações $s_i \leq n_i$ e $s_{i+1} \leq n_{i+1}$. Então:

$$d_G(r,g) > 0 = d_H(<3_i f_i>) \quad i \in s_0 \qquad (2\text{-}12)$$

$$d_G(r,i \wedge i) = e_j = W_l * -^{1} + {}^{3}e_j \quad i \leq s_i \qquad (2\text{-}13)$$

$$d_G(r,i,g,i) > {}^{3}c_j = W_l * "^{1} + {}^{6}c_j \quad i \in s_{i+1} \qquad (2\text{-}14)$$

14) acima, obtém-se:

$$s_i \cdot <i\text{-}e_{cj}> s_{i+1}$$

$$d - z_j (1-B_i)n_i + (1-6_i)n_{i+1} \qquad (2\text{-}15)$$

Utilizando a hipótese $d > \pounds(1-3_1+3_1)n_1 + (1-3_2+3_2)n_2 J.$ para enfraquecer a desigualdade (2-15), tem-se

$$D_G(r,fl) > E\{n_{cj} \cdot i_i + n_{i+1} e_j\} = D_G M$$

C.Q.D

É óbvio que se o critério de distância mínima generalizada for utilizado como critério de decodificação, não serão cometidos erros nos casos onde n_i e n_{i+1} são tais que a desigualdade do teorema 2.1 é satisfeita.

No caso especial em que hã somente uma classe $a_1 = 1$, $n_{e'l}$ é o número total t de erros em r, de maneira que o teorema torna-se $D_G(r,f) < D_G(r,g)$ $Vg \neq f$ se $2t < d$. Se for permitida a inclusão de urna classe de apagamentos $o_1 = 0$, cujo número total é $s = n_{c2} + n_{e2}$, o teorema pode ser reescrito como

$$D_r(I,i) < D_r(\pounds,2) \; V S + i \text{ se } 2t + s < d$$

Estas sao condições familiares que asseguram a nao ocor rência de erros quando for utilizada a decodificação por distância' mínima.

Definição 2.5.2a - É dito que a palavra código está "dentro da distância mínima da palavra recebida" quando a desigualdade do teorema 2.1 é aparentemente satisfeita.

O teorema mostra que há no máximo uma palavra código [1] dentro da distância mínima de qualquer palavra recebida. Assim, se o decodificador encontra alguma palavra código dentro da distância mínima da palavra recebida, ele pode imediatamente anunciar que esta e a palavra decodificada.

2.5.2.2 - O Algoritmo de Forney

I

Forney Jr.|14| propôs um algoritmo prático que usa distância minima generalizada como critério de decodificação, e que funciona se há uma palavra dentro da distância mínima da palavra recebida. Aqui é assumida a existência de algoritmos de decodificação que são capazes de manipular com apagamentos e erros, e que operam' se o número s de apagamentos e o número t de erros são tais que $2t + s < d$.

Suponha, para este propósito, que as letras recebidas [1] em algumas classes são consideradas como apagamentos, e as restantes como completamente confiáveis. Denotando por

E = { j l c . é não confiável}, e R = E ={j c. e de plena ' confiança}

é tentado decodificar a palavra recebida pelo algoritmo de erros-e-apagamentos, o qual estará apto para decodificar se

!

$$2\ l\ n_{j}\ .\ +\ I\ (n_{j}\ .\ +\ n_{j}\ .)\ <\ d. \qquad (2\text{-}16)$$
$$j\in R \Rightarrow j\in E \quad \wedge$$

Se isto acontece, verifica-se, então, se a palavra código obtida está dentro da distância mínima da palavra recebida. Caso ela esteja, então foi determinada a única palavra dentro da distância mínima.

Não há uma única atribuição provisória de apagamentos para a qual este método tem sucesso. Contudo, o teorema a seguir e seu corolário mostram que um pequeno número de tais tentativas apresentam êxito em encontrar a única palavra código dentro da distância mínima, se existe uma. Como anteriormente, as classes são consideradas ordenadas de acordo com ordem decrescente de confiabilidade.

Seja J o número total de classes, e um vetor de J dimensões a definido como $a = (a_1, \cdots a_J)$. São também definidos os conjuntos $R_b = \{j | j < b\}$ e $E_b = \{ j \mid j > b+1\}$ com $0 < b < J$.

É considerado que a_{-b}, um vetor J-dimensional com l's nas primeiras b posições e zeros nas demais, representa uma atribuição provisória correspondente a $R =$ e $E = E_1$. A idéia do teorema 2.2 abaixo é que ot está dentro de uma envoltória convexa, cujos pontos extremos são $\pm k_s$ enquanto a expressão f(ct) a ser definida é uma função linear de ot, que deve assumir valores mínimos em algum ponto extremo, ie., para uma das atribuições provisórias ct_J.

Teorema 2.2 - Se $\sum_{j=1}^{J}(1-a_j)n_j + (1+a_j)n_j \geq 1 < d$ e $a_j \neq a_k$ para $j < k$, existe algum inteiro b tal que a desigualdade

$$2\sum_{j=1}^{b} n_j + \sum_{j=b+1}^{J} (n_j + n_j) < d \text{ é satisfeita.}$$

Prova - Inicialmente e definida uma função do vetor ot.

$$f(a) = z\wedge(1-a_j)\wedge + U+C\wedge n\wedge\} < \qquad (2\text{-}17)$$

a qual satisfaz as seguintes propriedades:

i - f é uma função linear do vetor J-dimensional a

ii - note que $f(a_b) = \sum_{j=1}^{b} n_j + \sum_{j=b+1}^{J} (n_j + n_j)$.

O teorema é provado supondo que (Vb) $f(a_b) > d$, $0 < b < J$, e exibindo uma contradição.

Para isto, sejam $X_0 = 1 -$

$\lambda_b = a_b - a_{b+1}$, $1 \leq b < J-1$

$\lambda_J = a_J$

É facilmente verificado que $0 < \lambda_b \leq 1$ para $b = 0,1,\ldots,J$ e que $\sum_{b=0}^{J} \lambda_b = 1$.

Ora, $a = \sum_{b=0}^{J} \lambda_b a_b$, de modo que é possível escrever

$$f(a) = f\left(\sum_{b=0}^{J} \lambda_b a_b\right) = \sum_{b=0}^{J} \lambda_b f(a_b) > d \sum_{b=0}^{J} \lambda_b = d$$

CONTRADIÇÃO 2

Portanto, conclui-se que $f(a_b) \leq d$ para pelo menos um valor de b, $0 < b < J$. C.Q.D.

Este teorema prova que se há alguma palavra código dentro da distância mínima da palavra recebida, então deve haver alguma atribuição provisória, a qual permite que o decodificador de erros-e-apagamentos tenha sucesso em decodificar de acordo com o critério de distância mínima generalizada. Mas um decodificador de erros-e-apagamentos tem êxito se e somente se existem aparentemente

Nenhum erro e d - 1 apagamentos

Um erro e d - 3 apagamentos

- t_0 erros e $d - 2t_0 - 1$ apagamentos, onde $t_0 = \frac{d-1}{2}$

<u>Corolário</u> - Bastam $t_0 + 1 \leq \frac{d+1}{2}$ tentativas de possíveis atribuições provisórias para que se tenha sucesso em decodificar qualquer palavra recebida que está dentro da distância mínima pelo critério

de distância mínima generalizada, nao importando quantas classes de confiabilidade existam.

Segue-se, portanto, que o número máximo de tais processos é somente proporcional a d. Ademais, muitos dos processos (taj. vez todos) podem ter êxito, de modo que o número médio de proces - sos pode ser apreciavelmentemenor que o máximo.

2.5.2.3 - Cotas (exponenciais) de Erros para o Algoritmo de Forney

Para o algoritmo proposto por Forney |l4 | existem desenvolvidas cotas "apertadas" para a probabilidade de não decodificar' corretamente, onde "não decodificar corretamente" significa decodificação incorreta ou incapacidade de decodificar. Este último evento ocorre quando não há palavra código dentro da distância mínima ' da palavra recebida.

Considere uma variável aleatória *y^,* a qual para cada letra transmitida assume valores:

$$y_i = \begin{cases} 1-a_D & \text{se a letra é recebida corretamente e posta em } c_j \\ 1+a_D & \text{se a letra é recebida incorretamente e posta em } c_j \end{cases}$$

Assumindo um canal sem memória, estas variáveis aleatórias $\{y_i\}_1^n$ resultam iid.

Sejam $p_{c_j} = pr\{y_i = (1-a_j)\}$ e $p_{e_j} = pr\{y_i = (1+a_j)\}$ (2-18)

A função geradora de momentos |s g(s) para a variável aleatória y^ é dada por

$$g(s) = E\{e^{sy_i}\} = \sum p_{c_j} e^{s(1-cx_j)} + p_{e_j} e^{s(1+a_j)} \quad (2\text{-}19)$$

e a função geradora semi-invariante | 8 | correspondente, expressa por u(s) = £ng(s).

Ja e sabido que nao haverá erros na decodificaçao se n_{CD}. e n_p sao tais que a desigualdade do teorema 2.1 é satisfeita. o que é o mesmo que requerer que $\sum_{i=1}^{n} y_i < d$. Deste fato segue-se que

$$pr(ndc) = pr\{\sum_{i=1}^{n} y_i \geq d\} \qquad (2\text{-}20).$$

A cota de Chernoff | 8 | ê uma cota exponencialmente apertada para a probabilidade de que a soma de variáveis aleatórias iid exceda um certo número; no caso, a cota resulta em

$$pr(ndc) < \exp - n\,|s\delta - u(s)| \qquad (2\text{-}21)$$

válida Vs >' 0, onde 6 = d/n.

É possível tornar a cota mais apertada maximizando o expoente pela escolha de s e dos ou's:

$$E(\hat{o}) = \max_{S,C_{ij}} |s\delta - u(s)| \qquad (2\text{-}22)$$

Primeiro, a maximização será feita sobre ct_{ij}; e desde que y(s) = £ng(s), isto e realizado minimizando g(s).

$$g(s) = -s\,p\,e^{s(l-a_j)} + s\,p\,e^{s(l+a_j)} = 0 \qquad (2\text{-}23)$$

Os et_j's ótimos são então determinados a partir de (2-23) acima, lembrando da restrição Q < < 1, o que resulta em

$$a_j = \begin{cases} 0 & \text{se } L_j \leq 0 \\ L_j/2s & \text{se } 0 < L_j < 2s \\ 1 & \text{se } L_j \geq 2s \end{cases} \qquad (2\text{-}24)$$

onde $L_{\hat{}} = In \frac{p \cdot}{p_{ej}} = In \frac{prír.\ correta]\ r.\ E\ c\ \}}{pr\{r\hat{}incorreta\ |r_i\ e\ c'.\}}$

Assim, a atribuição ótima dos pesos envolve apagamento' de qualquer recepção para a qual a probabilidade da escolha correta p seja menor que 1/2 ; considera plenamente confiável qualquer recepção para a qual p seja maior que um limiar (que depende de S), e para valores intermediários assume um valor proporcional ao £og da razão de verossimilhança *In* p/(l-p). Portanto, exceto pelas limitações nos extremos, a decodificação por distância mínima generalizada é um método que utiliza o £og da razão de verossimilhança em esquemas de decodificação algébrica.

Suponha definidas as classes jeR se $L_{\hat{}} > 2s$

jeE se $L_{\hat{}} < 0$

jeG noutros cas|)s

Segue-se, então, que

$$9opt^{(s)} = e\ p_e(s) + {}^{eS}P_x<^s> + {}^{eS}P_g<^s> + P_c<^s> \qquad <^{2}\text{-}^{25}>$$

onde,

$$p(s) = £\ p\ .\ ,\ p(s) - Z^{p}\ c\ j \qquad (2\text{-}26)$$
$$jeR^{\varrho}\ l \qquad {}^{c} \qquad jeR$$

$$^{P}*^{(S)} \quad {}^{=}jeE \quad \hat{} \quad {}^{+} \quad \hat{} \qquad (2\text{-}27)$$

$$P_g a^{(s)} = {}^{1}\ {}^{2/p}\ e\ i\ {}^{P}{}_{e}\ c\ i_{J} \qquad (2\text{-}28)$$
$$jeG$$

Finalmente, assumindo $\hat{}_{opt}(s) = In\ g_{Qpt}(s)$ e tendo em vista a equação (2-22), tem-se que

$$E(\hat{o}) = \max\ |s\hat{o} - y_{Qpt}(s)| \qquad (2\text{-}29)$$

Agora otimizando pela escolha de s, observa-se que

para o s õtimo verifica-se a relação

$$\text{6 = Y º P TÍS) = ~ \% ^} \qquad \text{(2 " 3 0)}$$

Como $^Qp_t^{s}$^ 6 monotonicamente crescente com s, pois, MÍs) é convexa | 8 |, a equação (2-30) acima admite solução com s > 0 se e somente se 6 > ^C0)« É concluido, portanto, que a menor distância mínima (de Hamming) para a qual a decodificação por distância mínima generalizada pode operar é dada por ^{n}^òpt^^*

2.5.3 - Decodificação por Distância Mínima Generalizada II

Neste seção são feitas algumas considerações adicionais sobre distância generalizada, apresentando um algoritmo de decodificação por distância mínima para códigos de bloco, o qual faz uso da treliça associada ao código. ^

O algoritmo aqui apresentado estabelece um procedimento equivalente ao proposto na seção anterior no sentido de que também' escolhe a palavra código mais próxima da palavra recebida, quando é utilizada como medida de proximidade a distância generalizada. Deve ser mencionado, entretanto, que existem algumas dificuldades práticas na implementação do decodificador descrito pelo teorema 2.2 da seção anterior. As + 1 tentativas de decodificação do corolário[1] deste teorema podem resultar em até *t'*^+ 1 palavras código diferentes[1].

A classe de algoritmos apresentados por Forney sugere o uso de técnicas algébricas para gerar um certo numero de palavras código próximas em algum sentido da palavra recebida. O problema crucial é encontrar uma técnica eficiente de gerar o conjunto de palavras código, o qual tenha alta probabilidade de conter a palavra código ' mais provável, dado a palavra recebida. Será mostrado que o procedimento descrito a seguir se apresenta mais simples em termos de com-

plexidade do decodificador.

2.5.3.1 - Considerações Introdutórias

Em princípio, uma mudança na notação será realizada por conveniência, de modo a tornã-la adequada ao problema. Considere R o espaço de observaçõ̃es e uma partiçã̃o de R denotada por $\{I K^{S}\}$ / $j\ c\ \{0,1\}$ e $s\ c\ S$, onde S ê um conjunto de índices. $H = ITQ\ U\ 11^{\wedge 1}$ denota uma classe de confiabilidade, a qual são associados dois parâmetros B e 3 , $0 < R \quad R < 1$; e a quantidade $a = \$ - 3$ é dita ser o peso da classe n . Também sera considerado um vetor ' $a = (a, a, ..., a)$ definido pelo peso das classes nas quais as componentes da palavra recebida $r = (r-^{\wedge}, r^{\wedge}/\bullet.\bullet, r)$ se encontram; ou seja, se r. e II então a = a Assim, a notação (i) indica que se faz referência a classe de confiabilidade a qual r. pertence.

Uma das perguntas importantes que surgem quando é dada uma partição {IIj} do espaço de observação R, ê como atribuir valores aos parâmetros B_{e} e B para cada classe de confiabilidade' II . E claro que o emprego da teoria da decisão seria oportuno aqui, usando apropriadamente o conceito de função utilidade |l3|. Contudo, será proposta uma maneira de se atribuir valores para os pesos ' de cada classe II da partição, de forma a representarem um grau de confiabilidade associado â classe.

2.5.3.2 - Sobre Atribuição de Pesos a Classes de Confiabi1idade

Dada uma partição qualquer {n }, é proposto associar a cada elemento n da partição os seguintes parâmetros:

classe de confiabilidade P

3_{es} = pídec correto|r.eu $\}e\ 3_{cs}$ =p{dec com erro|r.eII }

Deve ser observado que $0 < 3\hat{}_{cs} < 3_{es}$ 1/ e que esta atribuição implica em $3_{es} + 3_{cs} = 1$.

De acordo com a definição 2.5.1b, a distância generalizada de Forney entre a palavra recebida r e uma palavra código £ será expressa por

$$D_G(\underline{r},\underline{f}) = \sum_{i=1}^{n} d_G(r_i, f_i) \qquad (2\text{-}31)$$

onde,

$$d_G(r_i, f_i) = \begin{cases} 3_{es} = p\{\text{dec correto}|r.eII^s\}\ r.\hat{}f./\ r_i ell^s \\ 3_{cs} = p\{\text{dec com erro}|r.eII^s\}\ r.?f.,\ r_i ell^s \end{cases} \qquad (2\text{-}32)$$

Inicialmente será analisado o caso binário, onde é assumido que a partição é simétrica, ou seja, {II } é escolhida de tal maneira que

$$p\{\text{dec } 0|n^s,0\} = p\{\text{dec } l|n^s,l\} \qquad (2\text{-}33)$$

Para uma classe II^{sO}, o coeficiente de confiança condicional de Kiefer |2l| é dado por $R_{sQ} = p\{\text{dec } j|n^{sO}, j\}$.

Neste caso particular, (2-32) pode ser reescrita na forma

$$d_G(r_i, f.)H \begin{cases} 5_{es} = p\{\text{dec } \cdot I\, n^{sO}\ ;jj\}\ r_{\pm}\hat{}f_{i\#}{}^{r}{}_{\pm}zm''^{sO} \\ \backslash_{cs} = l\text{-}p\{\text{dec } \cdot|n^{sO}, j'')\ ri\text{-}fi,\ r\ e''n^{s0} \end{cases} \qquad (2\text{-}34)$$

Então para a classe II^{s0}, o peso associado a_{gQ} e dado por

$$a_{g0} = 2.p\{\text{dec } j|n^{s\circ},j\} - 1 = 2\ R_{Q0} - 1 \qquad (2\text{-}35)$$

Notando que 0.5 < R _ < 1, tem-se que 0 < a ₙ < 1.
s0 - s0 -

Por outro lado,

$$R_{Q} = p\{dec\ j|n^{s°},j\} = P^{\{dec\ J'^{n}\ l\ j\}}_{s0,\ p\{ji^{s0}|j\}} \qquad (2\text{-}36)$$

ou seja,

$$p\{dec\ j|n^{s0}, j\} = \frac{p\{dec\ j, Il^{s0}|j\}}{p\{dec\ j, n^{s0}|j\} + pídec\ l - j, n^{s0}|j\}} \qquad (2\text{-}37)$$

Desta forma, deve ser assumido que

$$^{6}e_{s} = \frac{p\{dec\ i\text{-}i, n^{B}»h\}}{pídec\ j, n^{s0}|j\}} = \frac{}{1 + \frac{pín\text{T}Jj\}}{pín^{\circ}|j)}}$$

Definindo ^£|j' a razão ^e verossimilhança na classe' sO

$$9B\backslash 4 = \frac{A\ _{p}ín|^{u}|j\}}{Pín^lj)} \qquad (2\text{-}39)$$

e observando que para partição simétrica 11 = ^1 10 = ^' ae< ^ua ~ ção (2-38) pode ser escrita como

$$3_{e_g} = \frac{}{1 - çf}\ a\bullet\ ;\ \text{consequentemente,}\ 3_{c_g} = 1\ i\ \%\text{-}j\ i^{\wedge} = i\text{=-}2\text{-}^{\wedge}y$$

de modo que o peso associado a classe IT será dado por sO

$$a_{sO} = 3^{\wedge}_{e_g} - B_{c_g} = 1\ ]\ ^{\wedge}5 = P \qquad (2\text{-}40)$$

É interessante observar que o algoritmo de Hartmann - Rudolph, descrito no capítulo III, utiliza esta quantidade p no processo de decodificação.

A generalização deste procedimento para o caso muitini

vel e para casos onde a partição do espaço de observações não é simétrica será realizada a seguir. Aqui são supostas conhecidas as probabilidades a priori dos símbolos da fonte, denotadas por $\{P_j\}_0^{q-1}$.

A equação (2-39) resulta em

$$\mathcal{I}_s = \sum_{j=0}^{q-1} p_j \; p\{\text{dec } j \mid n^S, j\} \tag{2-41}$$

$$\mathcal{I}_n = 1 - \sum_{j=0}^{q-1} p_j \; p\{\text{dec } j \mid n^S, j\}$$

e o peso da classe II será agora determinado de acordo com

$$a_{s0} = \sum_{j=0}^{q-1} P\{\text{dec } J \mid n^{s0}_r j\} - 1 \tag{2-42}$$

$$\sum_{j=0}^{q-i} p_j \; 2\, p\{\text{dec } j \mid n^{s0}, j\} - \sum_{j=0}^{q-i} P_j \tag{2-43}$$

$$\sum_{j=0}^{q-1} p_j \,[2\, p\{\text{dec } j \mid n^{s0}, j\} - 1] \tag{2-44}$$

No caso multinível, a equação correspondente a (2-36) é

$$p\{\text{dec } j \mid n^{s0}, j\} \tag{2-45}$$

$$\frac{P_{inf\,j}}{\sum_{i=0} P_{inf\,i}} \tag{2-46}$$

$$p\{n^{s0} \mid j\}$$

Relembrando da definição de $\mathcal{I}_{s|j}$ dada pela equação (2-39), tem-se que

$$p\{\text{dec } j \mid n^{s0}, j\} = \frac{}{\sum_{i=0}^{z} p_i} \tag{C2-47}$$

Desta forma, o termo entre colchetes na equação (2-44) é

$$2'.p\{dec\ j \mid n^{sO},\ j\ ) - l = 2.\frac{1}{q-1} \equiv 1 \qquad (2\text{-}48)$$

$$1=0$$

$$\frac{q-1}{\pounds=0} \qquad (2\text{-}49)$$

$$q-1$$

$$E$$

$$\pounds=0$$

, (2-49) pode

$$2.p\{dec\ 3\ In\ \ r\,j > \text{"}\ 1 - \frac{1 \sim \frac{q-1}{1=0}}{1 + \frac{q-1}{1=0}} f\ Wl \qquad (2\text{-}50)$$

Esta expressão será definida como p_. multinível,

$$p.\ 4\frac{q-i}{3} \frac{-\wedge 0}{q-1} \qquad (2\text{-}51)$$

$$1=0$$

Então, de acordo com (2-44) o peso da classe II será' calculado pela fórmula

$$a_{s\,0} = \frac{q-1}{j=0} P.P. \qquad (4\text{-}52)$$

No caso binário, $P_o = \frac{1}{1 + ET_{1|0}}$ e $p_1 = \frac{1}{1 + p_{0|1}}$,

de modo que quando a partição é não simétrica, tem-se $p_Q \wedge$,

como exemplificado na figura 2.13 a título de ilustração. No caso onde (2-33) se verifica, $\emptyset_{0|1} = \emptyset_{1|0} = \emptyset$ e consequentemente $\rho_0 = \rho_1 = \rho$; portanto, (2-52) se reduz a (2-40).

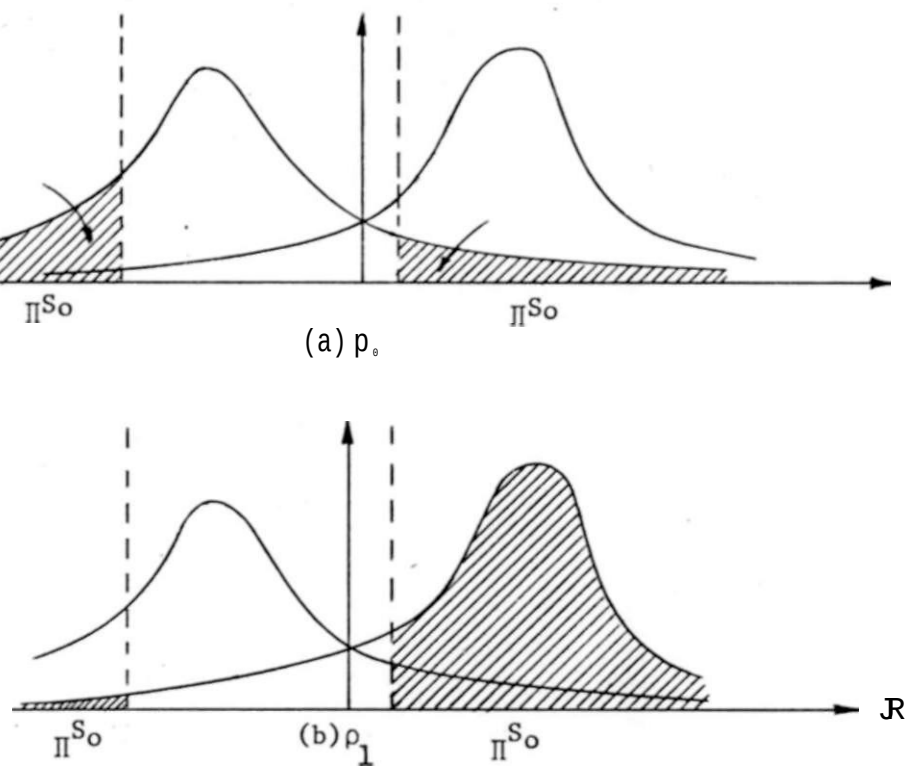


Figura 2.13 - PARTIÇÃO NÃO SIMÉTRICA.

Dada uma partição $\Pi = \{n\}$, o procedimento aqui estabelecido indica como realizar uma atribuição para os pesos a das classes de confiabilidade, sem indicar contudo como proceder para realizar uma "boa" escolha da partição.

2.5.3.3 - O Algoritmo de Decodificação

O decodificador proposto deve operar sobre a palavra recebida r, escolhendo a palavra código f mais próxima segundo o critério da distância generalizada, i.e./ deve escolher uma palavra código f e C tal que $(\forall f' \in C)\ D_6(r,f) < D(r,f')$ (2-53;

Para atingir este fato, é proposta uma regra de decodificação que consiste em estimar a palavra transmitida como sendo a palavra código f e C a qual minimiza a expressão $A(r,f) = \{r(+)f\}\cdot\pounds$ ( )

Isto pode ser facilmente demonstrado através do seguinte lema:

Lema 1: Dado a palavra recebida r, a palavra código f mais próxima segundo o critério de distância generalizada é aquela que minimiza A(r,f).

Prova - Dada a palavra recebida r e uma palavra código f, a distância genralizada entre elas é dada por

$$D(r,f) \underset{o}{\overset{n}{m}} z \; d(r,f) \qquad (2\text{-}54)$$

Se ê observado que $d_{\_}(r.,f.) = d(r.,f)$ a^'+ ,
então, é possível reescrever a equação (2-54) como

$$D_G(r,f) = J_x \; d_H(r.,f_i)_a{}^{(i)} + \qquad (2\text{-}55)$$

Por outro lado, o segundo termo do somatório independe' da escolha da palavra código, de modo que

$$\underset{f}{\text{Min}}\; D_{\text{„}}(r,f) \Rightarrow \underset{f}{\text{Min}} \sum_{i=1}^{n} d_{\text{„}}(.r.,f.)\; a^{u;} \qquad (2\text{-}56)$$

Finalmente, observando-se que

[r(+)f].ct =i£i dHAri'fiAa1 'a Prova ^ completada.

C.Q.D

É interessante notar que se existe somente uma classe ' de confiabilidade, então

$$\underset{f}{\text{Min}}\; D_{\mathcal{C}_j}(\underline{r},\underline{f}) \Rightarrow \underset{f}{\text{Min}}\; D_{\mathbf{ri}}(\underline{r},f).$$

Pode ser observado que o decodificador primeiramente efetua uma decisão abrupta determinando uma palavra recebida r, a qual pode ou não ser uma palavra código. Em seguida, ele determina

um vetor de confiabilidade ^, associado a esta palavra recebida.

É facilmente verificado pelo lema 2 abaixo que se a palavra recebida r resultante do uso de decisão abrupta for uma palavra código, ela própria e a palavra código mais próxima da palavra recebida no sentido de distância generalizada. Segue-se, portanto, que neste caso, o algoritmo resulta em assumir que esta foi a palavra transmitida, o que equivale a desprezar as informações de confiabilidade. No caso contrario, estas informações são realmente utilizadas no processo de decodificação, possibilitando a correção de erros.

Lema 2: Se a palavra recebida r é uma palavra código, então a palavra código mais próxima de r no sentido de distância generalizada é a própria palavra recebida.

Prova - Foi provado anteriormente (Lema 1) que minimizar $D_G(r,f)$ é o mesmo que minimizar A(r,f), através da escolha de uma palavra código, ou seja,

$$\min_{f} D_G(r,f) \iff \min_{f} A(r,f) \qquad (2\text{-}57)$$

É claro que

$$(\forall \text{vec}) \; A(\underline{r},\underline{f}) = \sum_{i=1}^{n} H_{d_u}(r_i, f_i) a_{i\times 1}^{(i)} > 0 \qquad (2\text{-}58)$$

pois, $0 < a^{(i)} < 1 \quad i = 1, 2, 3 \ldots, n$.

A conclusão da demonstração decorre do fato que se r é palavra código, então

$$\min_{f} A(r,f) = A(r,r) = 0 \qquad (2\text{T}59)$$

C.Q.D

O lema 2 permite uma simplificação no procedimento de decodificação. Inicialmente uma decisão abrupta é realizada, resul-

tando em uma palavra recebida r, e a síndrome correspondente a esta palavra recebida ê calculada.

1 - Se esta síndrome e nula, o decodificador libera a palavra, adnú tindo que não ocorreu nenhum erro.

2 - No caso contrario, um vetor de confiabilidade ê calculado e o algoritmo atua da maneira já discutida.

2.5.3.4 - Exemplo:

Será apresentada uma aplicação do algoritmo de decodifi_ cação descrito, utilizando a treliça associada ao código de bloco binário considerado. Admitindo que a escolha dos pesos das classes' de confiabilidade utilizadas esteja de acordo com o que foi proposto (2.5.3.2), e que a partição simétrica foi escolhida de modo a resultar em uma máxima variabilidade dos pesos das classes; a regra de decodificação pode ser anunciada como: "Estime a palavra transmitida como a palavra código para a qual $\sum_{i=1}^{n} d_i (r_i, f_i)_p$ ê mínimo, onde, = 1 - *Çf^^/l* + JJf e 0 ^ *ê* a razão de verossimilhança".

Este procedimento pode ser facilmente implementado com o auxílio de uma treliça. No exemplo, novamente é assumido que o código (5,3,2) é empregado, e que o canal é perturbado por um ruído com distribuição de probabilidade /M0,1). É admitido ainda que a ' fonte bipolar transmite +1V ou -IV correspondendo aos símbolos ' 0 e 1.

Suponha que as amostras recebidas foram ' (-0.7,-0.2,-0.3,+0.7,+1.1),de modo que p ( ) = (0.6,0.2,0.3,0.6,0.8)

A palavra recebida *ê* aquela que seria decodificada por decisão abrupta, ou seja, r = (0,0,0,1,1).

Usando a treliça, são obtidos os seguintes passos no processo de decodificação:

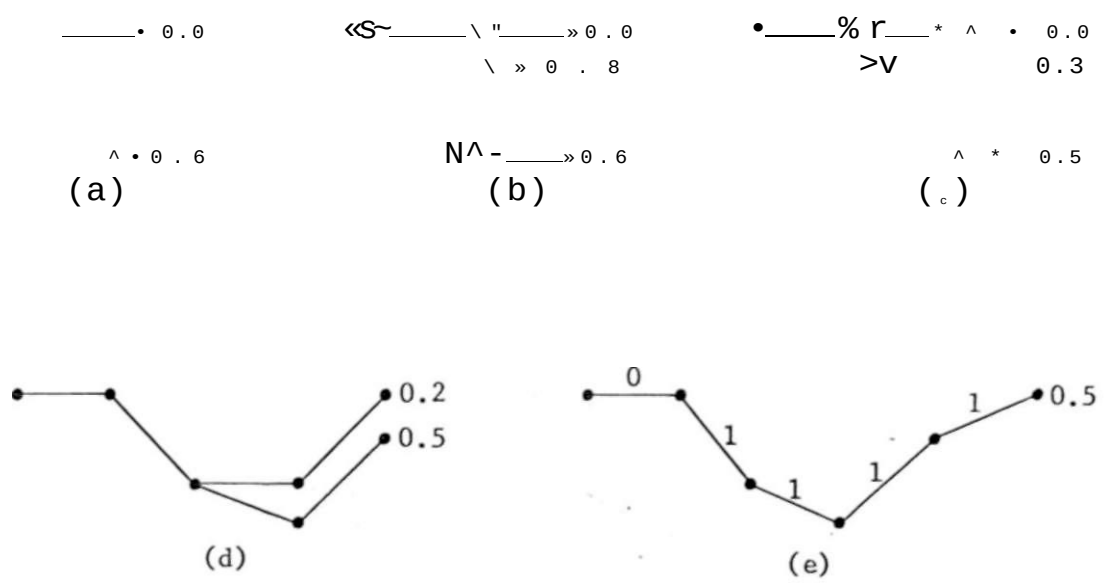


Figura 2.14 - PASSOS NA DECODIFICAÇÃO POR DISTÂNCIA MÍNIMA GENERALIZADA.

A palavra código $(1,0,0,1,1)$ está mais próxima da palavra recebida $(0,0,0,1,1)$ no sentido de distância de Hamming; entretanto, a palavra decodificada foi $(0,1,1,1,1)$, mais próxima no sentido de distância generalizada. Os dígitos corrigidos foram exatamente as posições de mais baixa confiabilidade. Também deve ser observado que a escolha da palavra código $(1,0,0,1,1)$ implicaria em inverter um dígito com confiabilidade relativamente alta. Finalmente, este procedimento apresenta uma vantagem computacional, pois nem sempre é necessário efetuar cálculos, visto que quando $\Lambda JJ\Lambda' j\_j\_)^{-} 0$, nada se adiciona ao somatório.

2.6 - DECODIFICAÇÃO POR DECISÃO SUAVE EMPREGANDO SUBTRELIÇAS

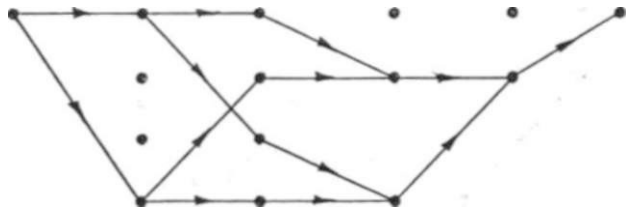
Cl

Matis e Modestino |25| introduziram técnicas que permitem uma redução na complexidade da decodificação de códigos de bloco lineares para algoritmos que empregam a treliça gerada pela matriz ^{1} [H]. Uma das formas proposta para realizar isto é através do uso de

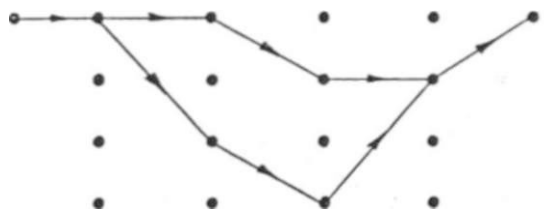
subtreliças bem mais simples que a treliça associada ao código. Se a palavra recebida r tem vários coeficientes de confiabilidade com valores grandes, as saldas de decisão abrupta correspondentes podem ser consideradas corretas. A subtreliça é gerada aceitando como corretas estas saídas de decisão abrupta para os símbolos nas k-p posições mais confiáveis, e permitindo todas as possíveis combinações nas posições restantes. É interessante observar que a escolha p=k corresponde ao emprego total da treliça, enquanto que o extremo oposto p=0 corresponde a realizar uma decisão abrupta sem uso da treliça. Valores intermediários de p fornecem as subtreliças desejadas. A aplicação deste procedimento é melhor ilustrada considerando o exemplo apresentado na sec. 2.5.3.4, onde

$$r = (0,0,0,1,1) \quad \text{e} \quad a^{(\ )} = (,6,.2,.3,.6,.8) .$$

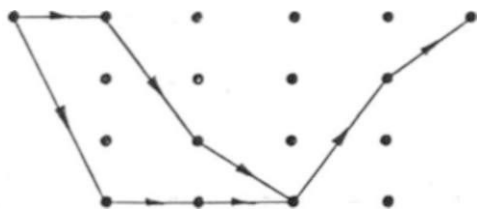
As subtreliças geradas para este caso sao mostradas na figura 2.15 (a) e (c).



(a) P=2



(b) p = l



(c) p = l

Figura 2.15 EXEMPLO DE SUBTRELIÇAS PARA UM CÓDIGO DE BLOCO.

O desempenho da decodificação por máxima verossimilhança (descrito na sec. 2.3) empregando subtreliças é analisado para alguns códigos, na referência |25| / através de simulação em computador.

CAPÍTULO I I I

DECODIFICAÇÃO ÕTIMA DE CÓDIGOS LINEARES

Serão apresentadas duas técnicas ótimas de decodificação de códigos lineares em canais discretos no tempo e sem memória, quando as palavras código são equiprovãveis. Ambas fazem uso de decisão' suave e são ótimas no sentido que minimizam a probabilidade de erro por símbolo e por bloco, respectivamente. A primeira delas, o algoritmo de Hartmann e Rudolph |18], faz uso das propriedades de linearidade do código, sendo portanto, aplicável somente para códigos lineares. Ela é particularmente atrativa para códigos corretores de ' erros com alta eficiência, conforme será visto. A segunda técnica descrita é um algoritmo para maximização da probabilidade a posteriori das palavras código |29I^ aplicável a códigos não lineares tão bem quanto para códigos lineares. Quando estes são considerados, o uso da treliça associada na decodificação torna prático seu emprego por simplificar o processo de decodificação, preservando ao mesmo ' tempo sua otimalidade. O desempenho deste algoritmo é inferior ao desempenho da regra de decodificação de Hartmann-Rudolph com relação a taxa de erros por símbolo, e vice-versa com relação a taxa de erros por palavra. Para ambos os algoritmos será analisado o desempenho assintótico da probabilidade de erro, por bit e por palavra, respectivamente, quando são utilizados códigos de bloco binários e o canal é perturbado por ruído branco gaussiano.

3.1 - O ALGORÍTMO DE HARTMANN E RUDOLPH

3.*1.1 - Introdução

A regra de decodificação apresentada pelo algoritmo de [1] Hartmann-Rudolph é ótima, no sentido de que minimiza a probabilidade de erro por símbolo sobre um canal discreto no tempo e sem memória , para qualquer código linear corretor de erros,quando as palavras código são equiprováveis.

A maioria das técnicas de decodificação de códigos corretores de erros são exaustivas no sentido de que toda palavra código é utilizada no processo de decodificação. Estas técnicas não fazem [1] qualquer uso essencial da linearidade do código. O algoritmo aqui ' apresentado ê também exaustivo, mas no sentido de que toda palavra ' do código dual é utilizada no processo de decodificação; por isto , na prática, esta regra é usada somente para códigos cujo código dual tem um pequeno número de palavras, i£„ ela é particularmente atrativa para códigos de alta eficiência. Por exemplo, para o código de Hamming binário C(15,11,3) existem 2^ palavras código, enquanto

que o código dual C'(15,4,3) possui apenas 2 palavras código. É concluido, portanto, que a complexidade da regra de decodificação de verá variar inversamente com a eficiência do código. Este algoritmo' é de natureza essencialmente estatística e, é aplicável tanto a códigos de bloco quanto a códigos convolucionais. No apêndice A é apresentado um programa em Fortran IV utilizado na investigação do desempenho deste algoritmo para códigos de bloco binários em canais com ruído branco aditivo gaussiano. Os resultados destas simulações serão discutidos no capítulo a seguir.

* Uma maneira de utilizar a linearidade é considerar o uso da treliça associada ao código, descrita na sec.2.3.

!

3.1.2 - A Regra de Decodificaçao

A regra de decodificaçao i apresentada para códigos de bloco lineares, e posteriormente uma extensão para códigos convolucionais será feita.

Seja c = Cc-QfC-$_f$•••$_f$c ^) uma palavra de um código de bloco linear C(n,k,d) sobre GF(p), e c! = (c•$_Q$, c•^, . . . , cj$_{n-1}$) ^{a} j-ésima palavra do código dual c' (n,n-k,d'). A palavra c é ^{1} transmitida através de um canal discreto no tempo e sem memória cujos símbolos r^ são números reais, de modo que a palavra recebida é da forma r = (r^,r^,...,^{r}_^)/^{com} r^ e R. A questão que o algoritmo se propõe a resolver ê : dado r, calcular uma estimativa c^^{1} do m-ésimo dígito da palavra transmitida c, de maneira que a probabilidade de se ter c_m igual a seja maximizada. De outra maneira, a estimativa c é tal que
m

$$Pr(c_m = c_m Ir) \geq P_r(c_m = s Ir), \quad Vs \in GF(p). \tag{3-1}$$

A regra de decodificação pode ser assim descrita: Estime o dígito transmitido como sendo o valor de s c GF(p) que maximiza a expressão

$$A_m(s) = \sum_{t=0}^{p-1} \cdots \sum_{j=l}^{p^{n-k}} \prod_{l=Q}^{n-1} P(r^{\wedge} | i) \tag{3-2}$$

ou sena, a estimativa c_m deve ser tal que $A_m(c_m) = \max_{s \in GF(p)} A_m(s)$.

Isto pode ser provado mostrando que $P_r(c_m = s | r) = X\ A^{\wedge}(s)$, onde X é uma constante positiva.

Inicialmente, é possível escrever que

$$pr(c_m = s Ir) = \sum_{c \in G,\ c_m = s} P(c Ir) \tag{3-3}$$

ou

$$Pr(c=s|r) = \sum_{c \in C,\ c_m = s} P(r|c)\,[p(c)/P(r)] \qquad (3\text{-}4)$$

Considerando que os símbolos de C são elementos de GF(p) e que as palavras código são equiprovãveis, tem-se que $P(c) = 1/P^k$, e (3-4) pode ser reescrita como

$$P(c = s\ r) = P^{-k} * P(r\ c) \qquad (3\text{-}5)$$

Onde é o delta de Kronecker,

$$\delta_{ij} = \begin{cases} 1, & \text{se } i=j \\ 0, & \text{em caso contrário} \end{cases} \quad \text{e} \quad e_m = (0, 0, \ldots, 0, 1).$$

Em termos de transformada finita de Fourier, pode ser mostrado que [18]

$$\delta(x) = \frac{1}{p} \sum_{t=0}^{p-1} \omega^{tx} \qquad (3\text{-}6)$$

$$P(r|c) = V_n \sum_{u \in V_n} F(r,u)\, \omega^{u.c} \qquad (3\text{-}7)$$

onde

$$F(r,u) = \sum_{v \in V_n} P(r|v)\, \omega^{-u.y} \qquad (3\text{-}8)$$

e $\omega = \exp(2\pi i/p)$ representa a p-ésima raiz complexa da unidade e V_n o espaço n-dimensional sobre GF(P).

Substituindo (3-6) e (3-7) em (3-5) e levando em conta que devido as propriedades de ortogonalidade de caracteres de grupo

$$\sum_{v \in C^{\perp}} \omega^{u.v} = \begin{cases} p, & \text{se } u \in C^{\perp} \\ 0, & \text{em caso contrário} \end{cases} \qquad (3\text{-}9)$$

É obtida a expressão

$$P(c=s\,r) = \frac{P(r)}{\mathbb{R}} \sum_{t=0}^{n-1} \sum_{j=1}^{n-k} F(\underline{r}, c_l - t_e) \qquad (3\text{-}10)$$

Para canais sem memória, (3-8) pode ser reescrita como

$$F(r,u) = \sum_{\underline{v} \in V_n}^{n-1} P(r_m|v_m) 0^{-iu} = \sum_{i=0}^{n-1} D_i P(r|i) 0^{-iu} \,, \quad \text{de}$$

modo que se obtém a expressão

$$P(c = s|r) = \frac{0^{st}}{P(r)}, \quad \sum_{t=0}^{n-1} \sum_{j=1}^{n-k} \sum_{i=0}^{P-1} a_i \, m_l \, P(r) \qquad (3\text{-}11)$$

Definindo $A = p^n \cdot P(r)$, e observando a definição de $A_m(S)$, segue-se que $P(c_m = s|\underline{r}) = A_m(s)$

C.Q.D.

É interessante fazer uma aplicação deste algoritmo no caso de códigos lineares binários, onde a regra de decodificação se apresenta de uma forma comparativamente mais simples. Neste caso, p = 2, e o procedimento para decodificação consiste em escolher a estimativa

$$m = \begin{cases} 0, & \text{se } A_m(0) > A_m(1) \\ 1, & \text{em caso contrário} \end{cases} \qquad (3\text{-}12)$$

Esta forma comparativa da regra de decisão pode ser apresentada de maneira mais conveniente, quando estabelecida em termos da razão de verossimilhança $\zeta f_m = Pr(r_m|1)/Pr(r_m|0)$.

Para o caso em questão, a desigualdade $A(0) > A(1)$ resulta

$$\frac{1}{2^{n-k}} \sum_{t=0}^{n} \sum_{j=1}^{n} \sum_{i=0}^{1} (-1)^{i} (c_j - t_j)\, p(r_m | i) > \frac{1}{2^{n-k}} \sum_{t=0}^{n} \sum_{j=1}^{n} \sum_{i=0}^{1} (-1)^{i} (c_j - t_j)\, p(r_m | i) \qquad (3\text{-}13)$$

ou seja

$$\sum_{t=0}^{2^{n-k}} \sum_{j=1}^{n} \Pr(r_m | 0) + (-1) \sum_{t=0}^{2^{n-k}} \sum_{j=1}^{n} \Pr(r_m | 1) > 0 \qquad (3\text{-}14)$$

Dividindo ambos os membros de (3-14) pela quantidade positiva $\prod_{i=0}^{n-1} \Pr(r_i | 0)$ e usando a definição de razão de verossimilhança, a desigualdade pode ser escrita como

$$\sum_{t=0}^{2^{n-k}} \sum_{j=1}^{n} \left[1 + (-1)^{c_j} \lambda_j\right] > 0 \qquad (3\text{-}15)$$

Agora dividindo (3-15) pela quantidade positiva $\prod_{i=0}^{n-1} (1 + \lambda_i)$, e utilizando a identidade

$$\frac{1 + \lambda_i - 1}{1 + \lambda_i} = \frac{\lambda_i}{1 + \lambda_i},$$ segue-se que

a regra de decodificação no caso binário pode ser estabelecida conforme enunciada abaixo:

Escolha $c_m = 0$ se $\sum_{t=0}^{2^{n-k}} \sum_{j=1}^{n} \prod_{i=0}^{n-1} (1 - \lambda_i) > 0$, caso contrário faça $c_m = 1$.

Algumas observações devem ser feitas. Em primeiro lugar, deve ser notado que quando esta regra de decodificação símbolo-por-símbolo é usada, a palavra estimada c_m na saída do decodificador

não é" necessariamente uma palavra código. Em segundo lugar, se cód: gos de bloco cíclicos são considerados,a regra de decodificação deve ser determinada para o símbolo recebido $r^{}$, e então, os símbo • los restantes ' i ' " * " ! ' n - l P o c ^ e m s e r decodificados simplesmente permutando ciclicamente a palavra r no registro a deslocamento (acumulador), como ilustrado no exemplo a seguir.

3.1.3 - Exemplos

Será utilizado o código de Hamming (7,4,3) para exemplificar a regra de decodificação binaria aplicada a códigos de bloco que neste caso torna-se:

$$\text{Escolha } CQ = Q \text{ se } \sum_{j=l}^{8} \left(\sum_{l=0}^{6} (1 - \zeta f_{\underline{\ell}}) + \#_{\ell} \right)^{c'_{j\ell} \oplus \delta_{m\ell}} > 0,$$

e CQ = 1 em caso contrário.

A matriz de verificação de paridade deste código é

$$[H] = \begin{matrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{matrix}$$

e consequentemente o espaço linha C será dado por

0 0 0 0 0 0

o o

1 0 0 1 1 o

C¹

0 0 1 1 o

o 0 o

0 Q 1 1

0 0 1

Figura 3.1 - ESPAÇO LINHA DO CÓDIGO DUAL DO CÓDIGO DE HAMMING (7,4,3).

Tomando = (1 - *Çf^/d* + *Çf^)* , a regra de decodifica-ção consiste em escolher 6Q = 0 somente se

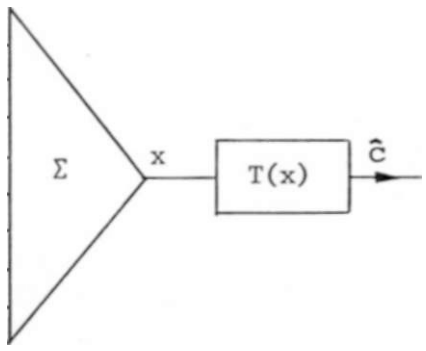
ᵖ0⁺ ᵖ1ᵖ2ᵖ4 ⁺ ᵖ2ᵖ5ᵖ6 ⁺ ᵖ l ᵖ 3 ᵖ 6 ⁺ ᵖ3ᵖ4ᵖ5 ⁺ ᵖ0ᵖ1ᵖ2ᵖ3ᵖ5 ⁺ ᵖ0ᵖ2ᵖ3ᵖ4ᵖ6

⁺ ᵖ0ᵖ1ᵖ4ᵖ5ᵖ6 " °-

O decodificador correspondente está esquematizado na figura 3.2 exibida abaixo

1 - (H
1 + ≺K

Pe			P3	P2	Pi	Po



T(x) = 0 se x > 0
j_ 1 caso contrário

Figura 3.2 - DECODIFICADOR ÓTIMO PARA O CÓDIGO DE BLOCO(7,4,3)

Para ilustrar a aplicação desta regra de decodificação para códigos convolucionais será usado um código convolucional (4,3,3)., A regra de decodificação para o símbolo recebido r^ usando um código convolucional (n^,k.Q,N) será, no caso binário, enunciada como:

Escolha $c_{n_0} = 0$ se e somente se $I_{j=l}^{n} \left(\frac{1 - V_{\ell}}{1 + cf_t}\right)^{c}$ > 0,

de outra forma assuma **CQ = 1.**

Naturalmente, existe somente um número finito de termos não nulos na equação acima, dependendo do comprimento da sequência' transmitida.

Agora, para exemplificar, as porções iniciais da matriz de verificação de paridade descrita no exemplo 1.5 e do espaço linha C' sao mostradas abaixo.

$$
\begin{array}{l}
1\ 1 \\
1\ 0 \qquad 1\ 1\ 1\ 1 \\
1\ 1\ 0\ 0 \qquad 1\ 0 \quad 1\ 0 \quad 1\ 1\ 1\ 1 \\
\qquad\qquad\quad 1\ 1\ 0\ 0 \quad 1\ 0 \quad 1\ 0 \quad 1\ 1\ 1\ 1
\end{array}
$$

4

$$
C^{1} = \begin{array}{l}
0\ 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
1\ 1\ 1\ 1\ 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
1\ 0\ 1\ 0\ 1\ 1\ 1\ 1 \quad 0 \ldots\ldots\ldots\ldots \\
0\ 1\ 0\ 1\ 1\ 1\ 1\ 1 \quad 0 \ldots\ldots\ldots\ldots \\
1\ 1\ 0\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 0
\end{array}
$$

Figura 3.3 - ESPAÇO LINHA DO CÓDIGO DUAL DO CÓDIGO CONVOLUCIONAL (4,3,3).

Como antes, = (1 - &)_/(! + çfA , de modo que a regra de decodificação será expressa por:

$c_0 = 0$ se e somente se $p_Q + P_{jL}P_2P_3 + {}^P2{}^P4{}^P5{}^P6{}^P7{}^+{}^P0{}^P1{}^P3{}^P4{}^P5{}^P6{}^P7{}^+$

+ ... > 0, em caso contrario faça CQ-1.

O diagrama do decodificador é então mostrado na figura' 3.4, tomando forma de diagrama de treliça para o código dual ' c' (4,1,3) com as posições C_!Q nos rótulos das malhas complementadas . (Em geral, para decodificar r^, as posições cj„ devem ser complementadas).

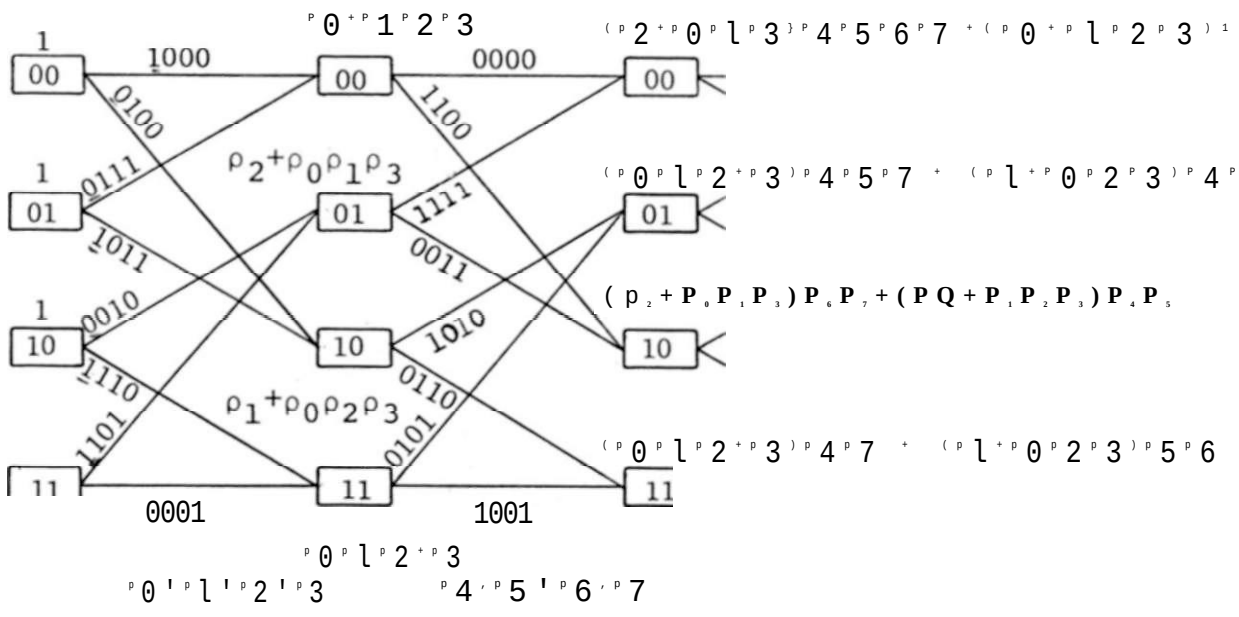


Figura 3.4 - DECODIFICADOR ÕTIMO PARA O CÓDIGO CONVOLUCIONAL (4,3,3).

Desde que diferentes unidades de memória devem ser empregadas para cada símbolo a ser decodificado, a quantidade de memória requerida por este tipo de decodificador cresce linearmente com o comprimento da sequência transmitida. O mesmo também é verdade para o algoritmo de Viterbi aplicado para decodificação de códigos convolucionais.

Veja figura 1.4, diagrama de treliça para o código convolucional (4,1,3).

3.1.4 - Desempenho Assintõtico do Algoritmo

Expressões assintóticas para a probabilidade de erro por bit na decodificação de códigos de bloco binários lineares em canal' com ruído branco aditivo gaussiano foram demonstradas |19 |.

A regra de decodificação õtima bit-a-bit para o m-ésimo[1] dígito de um código C(n,k,d) linear e binário consiste em escolher $c_m = s$, onde s e GF(2) e maximiza $p(c_m = s \mid r)$. Desta forma a probabilidade de erro por bit ê dada por

$$P_{bit} = P\{\hat{c}_m = c_m \mid r\} = p(c_m \neq c_m \mid r) \quad (3\text{-}16)$$

Não há perda de generalidade em assumir que a palavra [1] transmitida foi a ênupla toda nula, desde que o código é linear e o ruído apresenta simetria. Para facilitar a dedução, é considerado [1] que a palavra toda nula foi transmitida. Quando £= 0. é transmitida, a m-ésima componente da palavra recebida r ê $r_m = \sqrt{E} + e_m$, onde E e a energia do sinal por bit e e_m uma amostra de ruído de um processo gaussiano com densidade espectral de potência unilateral DQ watts/hertz. A SNR para este canal ê $\gamma = E/HQ$, OU em termos dos bits de informação transmitidos, $\gamma_b = E_b/N_0 = \gamma n/k = \gamma/R$. A m-ésima componente da palavra recebida r será decodificada incorretamente' se e somente se

$$P(c_m = 0 \mid r) \leq P(c_m = 1 \mid r) \quad (3\text{-}17)$$

onde $r = (\sqrt{E} + e_0, \ldots, \sqrt{E} + e_{n-1})$.

Em outras palavras,

$$P_{bit} = \sum_{c \in S_0} P(c \mid r) \leq \sum_{c \in S_1} P(c \mid r) \quad (3\text{-}18)$$

$$^{AB}\mathbf{Yb\text{-}0}^{(P}\mathbf{bit}^{)}= \begin{cases} (1 - r.c\ 4/RY_b/n_0) < \\ \pounds^{es}0 \\ (1 - r.c\ 4/RY_b/n_0) \\ \pounds^{es}l \end{cases} \qquad (3\text{-}22)$$

Então, como $s_0 = s_{ir}$

$$^{M}Yb\text{-}0^{(P}bit>=^{P} \quad ccs_0 \quad \text{Ç\_}^{es}I \quad r.c\ 4/RY_b/n_Q \qquad 3\text{-}23)$$

Desde que SQ e um código linear binário (n,k-l), se - gue que se os vetores de SQ são arranjados como linhas de uma matriz $M_{'0'}$ então cada coluna será toda nula ou conterá 2^{k-2} zeros e 2^{k-2} l's. Arranjando o conjunto de vetores de s-^ como linhas de uma matriz M^, as colunas de que correspondam a colunas toda nula em M_Q serão toda um, enquanto que as demais colunas terão 2^{k-2} zeros e 2^{k-2} l"s.

Usando este fato, é possível escrever

$$AB_{Yb>0}(P_{bit}) \sim P \quad z\ r\ ,\quad 0' \qquad (3\text{-}24)$$
$$\ll 1$$

onde $3_1, J_2, \cdots, 9, j$ sao colunas nulas de M_U.

Desde que r^ ~ X"(*l2,n /2) para $l = j$ ^ .

$$Z \quad r < 0 \qquad \exp(-x\ /2)dx \qquad 3\text{-}25)$$
$$+00 \qquad /2ÏÏ \qquad /2GRYi$$

De (3-24) e (3-25) segue-se a expressão assintõtica desejada, $AB\ \hat{}\ (P_{fait}) = Q(\ /2TOY\hat{}>$.

Se o código dual de C tem distância mínima maior que 2, então 6 = 1, e o comportamento assintótico pode ser descrito pe

$$P_b \sim P \left(1 - r.c\ 4/RY_b/n\right) < \frac{1}{\left(1 - r.c\ 4/RY_b/n_Q\right)} \qquad (3\text{-}22)$$

Então, como $= s_1$,

$$\left(1 - r.c\ 4/RY_b/n_0\right) \qquad r.c\ 4/RY_b/n_Q \qquad (3\text{-}23)$$

Desde que SQ é um código linear binário (n,k-l), segue que se os vetores de s^ são arranjados como linhas de uma matriz M_{υ}, então cada coluna será toda nula ou conterá 2^{k-2} zeros e 2^{k-2} l's. Arranjando o conjunto de vetores de s-^ como linhas de uma matriz , as colunas de que correspondam a colunas toda nula em MQ serão toda um, enquanto que as demais colunas terão 2^{k-2} zeros e 2^{k-2} l"s.

Usando este fato, ë possível escrever

$$P_b(E) \qquad T_r \qquad (3\text{-}24)$$

onde Ji'32'***'^e colunas nulas de Mg •

Desde que r^ ~ ,x\/2) para = J1#...,j

$$\int_{r<0} \frac{1}{\sqrt{2\pi}} \exp(-x^2/2)dx \qquad (3\text{-}25)$$

$$/20RY_b$$

De (3-24) e (3-25) segue-se a expressão assintótica desejada, $P_b \sim Q(\sqrt{7TOY})$.

Se o código dual de C tem distância mínima maior que 2, então 0 = 1, e o comportamento assintótico pode ser descrito pe

AB Yb-«» (P b i t N(W m) Q (/2R W m YT"). isto, porque para valores suficien-
2

temente grandes de x, Q(x)= *1/x/TH* exp(-x /2) , de maneira que so - mente as'palavras código de peso contribuem de forma signifi - cante no somatório. Deve ser também observado que se o código é cí- clico e tem distância mínima d, então W = d e

$$AB_{yh"*^{\circ\circ}} CP.\cdot b\cdot i\cdot t) \sim N(d)\ Q(/2Rd_{,} y)^{m}_{b}$$

3.2 - ALGORÍTMO PARA MAXIMIZAÇÃO DA PROBABILIDADE A POSTERIORI

3.2.1 - Introdução

Uma outra regra de decodificação será a seguir apresen- tada a qual é também ótima, mas no sentido de que para qualquer có- digo corretor de erros minimiza a probabilidade Ae erro por palavra

sobre qualquer canal discreto no tempo e sem memória, quando as pa- lavras código são equiprovãveis. Esta regra implica em maximizar a probabilidade a posteriori das palavras código. É sabido que exceto para poucos canais clássicos não ê normalmente muito claro como de- ve ser processada a palavra recebida de modo a maximizar a probabi- lidade a posteriori. No que segue é estabelecido um procedimento pa ra realizar tal maximização. Deve ser observado que ao contrário do algoritmo de Hartmann-Rudolph, a decodificação resulta sempre em uma palavra código quando este procedimento é empregado. Será tam - bém mostrado que a regra de decodificação para o caso binário pode ser enunciada como:

Escolher a palavra código c que maximiza a expressão[1]

$$\sum_{i=0}^{n-1} \log \gamma_i$$

Se dado a palavra recebida r = $(r_0, r_1, \ldots, r_{n-1})$ for definido um vetor r^* calculado a partir de r;

$r^* = (\log \gamma_0, \log \gamma_1, \ldots, \log \gamma_{n-1})$ onde $\gamma_i = Pr(r_i|1)/Pr(r_i|0)$, o

decodificador deve escolher a palavra código c que maximiza

$c.r$, ie., $c.r = \text{Max}_{c} c.r$. Isto torna possível uma interpretação in

teressantfe, observando que este procedimento generaliza a decodifi-

cação por correlação para códigos de bloco sobre canais com ruído

aditivo gaussiano, uma das técnicas ótimas conhecidas no sentido

de que minimiza a probabilidade de erro por bloco quando as pala

vras código são equiprováveis. Deste modo, é plausível interpretar*

esta decodificação como sendo uma forma de "correlação generalizada

Quando códigos lineares são usados, o emprego da decodif

ficação usando treliça faz este receptor interessante por simplifi-

car as operações de decodificação, retendo entretanto, a otimalida-

de. É importante observar que este algoritmo de decodificação é

atrativo tanto para códigos com alta eficiência, quanto para os de

baixa eficiência. Se o código utilizado é de baixa eficiência, en -

tão a decodificação pode ser feita exaustivamente, i.e./ utilizando

todas palavras código no processo de decodificação. Ele também é

de particular utilidade na decodificação de códigos com altas taxas

visto que a complexidade da treliça é função do número de dígitos

de paridade.

Se ao invés de empregado como na forma descrita acima

0 algoritmo for utilizado quantizando em 2^ regiões as amostras

recebidas, o resultado é uma pequena degradação no desempenho, po -

rém, facilitando sobremaneira a implementação do decodificador, co-

mo mostrado a seguir. A decodificação agora ê feita através do se -

guinte procedimento:

1 - vetor recebido $r = (r_0, , , . . . , r_{n-1})$ com r^ e R

2 - vetor quantizado r'= (r^, r|, . . . , r^_^) com r£ e

3 - vetor £og razão de verossimilhança

£*=(r0'r]Lrn-l' com ri£Rff

R = {espaço das possíveis saldas do canal ruidoso}

RQ= {espaço das possíveis regiões de quantização}

R^,= {espaço dos valores do *Zog* razão de verossimilhança}

É então utilizada a treliça associada ao código para maximizar a "correlação" $c.r^{*} = \sum_{i=0}^{n-1} c_i r_i^{*}$

O algoritmo será deduzido para códigos de bloco com alfabeto de símbolos p-ãrios,supondo palavras código equiprováveis [1] transmitidas em um canal discreto no tempo e sem memória, e posteriormente uma extensão para códigos convolucionais será feita.

3.2.2 - A Regra de Decodificação

Seja c = CCQrC.*i*•••,c uma palavra código de um código de bloco C(n,k,d) com símbolos em GF(p). Ao receber a palavra r = $(r_0, r^{\wedge}, \ldots, r_{n-1^{\wedge}})$/ ° receptor deve decidir pela palavra código c que maximiza a expressão

$$\sum_{i=0}^{p-1} \sum_{l=0}^{n-1} P(r_l | i)\, \theta^{j(c_l - i)} \qquad \textbf{(3-31)}$$

onde 0 = exp^n/^T/pl representa a p-esima raiz complexa da unidade.

Será provado que a palavra código que maximiza **(3-31)** [1] também maximiza a probabilidade a posteriori P(c|r). A probabilidade de P(çjr) pode ser determinada por

$$P(c|r) = P(c)\ P(.r|c)/P(r) \qquad \textbf{(3-32)}$$

e como por hipótese as palavras são equiprováveis, $P(c) = p^{-k}$, de maneira que

$$P(c|r) = P^{k}\ p(r|cl/P(r) \qquad \textbf{(3-33)}$$

Em termos de transformada finita de Fourier, a probabilidade P(r|c) pode ser escrita como

$$P(r|c) = p^{n} \sum_{u \in V_n} F(r,u) e^{-} \qquad (3\text{-}34)$$

onde

$$F(r,u) = \sum_{v \in V_n} P(r|v) \_ \textcircled{c} \sim - ' - \qquad (3\text{-}35)$$

Como são considerados apenas canais sem memória,

$$F(r,u) - \sum_{v \in V_n} \prod_{l=0}^{n-1} P(r_l | v_j)^{-u_l . v_l}$$

$$= \prod_{l=0}^{n-1} \sum_{i=0}^{p-1} P(r_l | i)^{\sim^{iu}} \qquad (3\text{-}36)$$

Substituindo as expressões (3-36) e (3-34) em (3-33) resulta

$$P(c|r) = [p^{-k-n} / P(r)] \sum_{u \in V_n} \prod_{l=0}^{n-1} \sum_{i=0}^{p-1} P(r_l | i)^{u_l (c_l i)}$$

$$\bullet \sum_{i=0}^{p-1} \left[\prod_{l=0}^{n-1} P(r_l | i) \right] \sum_{j=0}^{p-1} 0 \qquad (3\text{-}37)$$

De modo que maximizar (3-31) pela escolha de uma palavra código é o mesmo que maximizar a probabilidade a posteriori das palavras código.

C.Q.D.

Será feita a seguir uma aplicação deste algoritmo para o caso binário, i.e, quando p=2.

$$(c|r) = [XjJ^{n-1} P(r_l|0) (1+0\ ri + P(r_l|l) (1-0)^{\circ l} \qquad (3\text{-}38)$$

Mais uma vez e conveniente exprimir os resultados em termos da razão verossimilhança $Çf^ = Pr(r^1 1)/Pr(r^|0)$ e de $= (1 -» Çí^)/(1 + \$ç)$ • Isto resulta em

$$P(c|r) = [2^{n}/P(r)l \quad \prod_{1=0}^{n-1} (1 + 0^{l}) + Çf_{t}(l - 0^{l}) \quad P(rJO)$$

Tomando $X = [\_2\sim\ ^{n}/P(£)J \prod_{1=0} P(r.|0)$, tem-se que

$$p(c|r) = x \sum_{£=0}^{n-l} [d + jpr.) + e^{c_{p}} - pr.)!$$

$$x \; i\ h \quad 1 + P_{l}\ 0 \qquad (3\text{-}39)$$

Contudo, $0 = \exp\_\_\_\_\_^{2n} \cdot c_{„} = \begin{cases} 1, & \text{se } c^ = 0 \\ -1, & \text{se } c_{„} = 1 \end{cases}$

ou seja, $0 = (-1)$, de forma que (3-39) pode ser reescrita como

$$P(c|r) = X \sum_{1=0}^{n-l} \mathbf{n}\ 1 + \mathbf{p}.(-l) \qquad (1 + Çf_{z}) \qquad (3\text{-}40)$$

Usando a identidade $1 + P^(-l) = \frac{^{c}\ l\ ^{2} * t}{i+gr}$, obtém-se*

$$P(c|r) = 2À \sum_{1=0}^{n-l} \mathbf{n}\ Çfp^{c^}_{L}, \text{e } À \text{ independe de } c.$$

Como o logaritmo é uma função monotônica crescente, a maximização de $P(c|r)$ no caso binário pode ser realizada escolhendo a palavra código $c = (c^, . . . ,^{c}{}_{n}\_^)$ <3^e maximiza a expressão $\sum_{1=0}^{n-l}$

3.2.3 - Aplicações para Códigos Lineares

É fãcil verificar que no caso da transmissão de sinais' polares em um canal com ruído aditivo gaussiano, a regra de decodificação consiste em maximizar a quantidade $cr = \sum_{1=0}^{n-1} c_p r»$ pela escolha de uma palavra código $c = (c^,C^,...,c_n_^)$• Se os sinais transmitidos são pulsos liga-desliga de amplitude /Ê volts ou 0 volts, então $\sum_{£=0}^{n-1} c»(r» - /Ê/2)$ deve ser maximizada pela escolha de c.

Numa aplicação particular será obtida a regra õtima de decodificação quando são empregados detetores de envoltõria. Considere que o sinal ê transmitido através de pulsos de RF liga-desliga (ASK) em um canal com ruído aditivo branco gaussiano. É assumido ' que a envoltõria da forma de onda recebida é amostrada em interva - los bastante distantes de modo que seja razoável supor que as amostras são estatisticamente independentes. O sinal recebido é

$$Z(t) = A^ \cos Wg_t + x\ Ctl \cos(jOQ_t - x_g(t) \text{ sen } ojg_t \qquad (3\text{-}41)$$

onde AQ = 0 e A^ = constante, e o ruído está escrito na forma de banda estreita 133| de modo que $x_c(t)$ e $x_s(t)$ são variáveis aleatõrias i i d gaussianas.

A saída r (t) do detetor de envoltõria é dada por

$$r(t) = \{J"A. + x_c(tfl^2 + x_s^2(t)\}^{1/2} \quad ie \{0,1\} \qquad (3\text{-}42)$$

A densidade de probabilidade da envoltõria em um instante de tempo t 133 | tem a distribuição de Rayleigh na ausência de sinal, e uma distribuição de Rice quando o sinal está presente, ou seja,

$$P(r_\ell|0) = \begin{cases} I\ \exp(-r\hat{}/2a^2), & > 0 \\ \end{cases} \qquad (3\text{-}43)$$

$$P(r_\ell|l) = \begin{cases} \text{"2}^{ex}P\ \dfrac{r^2\ l + \text{Å}^2\ l}{2o^2}\ \text{/}^{\text{A}}\ 1^{r}\text{£}, & r_\ell > 0 \\ V & \\ r_\ell < 0 \end{cases} \qquad (3\text{-}44)$$

onde $o^2 = E\text{Ín}^2(t)\} = E\{n^2(t)\}$ é a potência do ruído, e $I_{(J}(x)$ e a função de Bessel modificada de primeira espécie e de ordem zero .

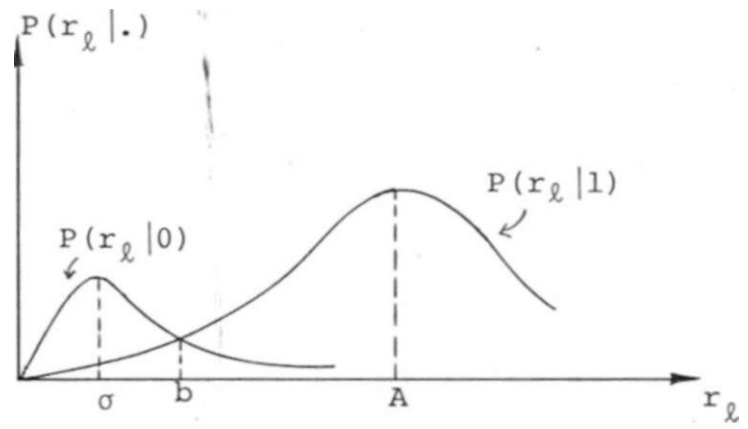


Figura 3.5 - DENSIDADES DE PROBABILIDADE DA ENVOLTÓRIA.

A razão de verossimilhança é dada pela expressão

$$\frac{P\text{ír}\hat{}ll)}{P(r_\ell|0)} = \exp(-A_1^2/2a^2)\ I_Q(r\hat{}A_1/a^2) \qquad 3\text{-}45)$$

Definindo a relação sinal/ruído como $y = A\hat{}/2a$, o valor do limiar b pode ser encontrado resolvendo a equação ^{1} $\exp(-y)IQ(bg/Zy) = 1$, cuja solução pode ser obtida através da excelente aproximação |34|

$$b_{Q} = /2 + y/2 = b/a \qquad (3\text{-}46)$$

Uma analise do comportamento assintõtico da regra de de codificação para baixa e alta relação sinal/ruído será realizada a seguir.

3.2.3.1 - Baixa Relação Sinal/Ruído: $y \ll 1$

Neste caso é quase sempre verdade que | 291

$$I_{Q}(r\hat{}Aj/a^{2}1\ w\ \exp(r\hat{}\ A\text{-}\hat{}a^{2}) \qquad 3\text{-}47)$$

portanto

$$- 1 \qquad (3\text{-}48)$$
$$2a' \qquad 2a'$$

e a regra de decodificação consiste em escolher a palavra código c = (CQ/ . . . ,$^{c}{}_{n}$_2^ ('3^{ue} maximiza^{a} expressão

$$\sum_{1=0}^{n-1} c_{\mu} \left| \frac{2}{\ l} \right. - 1 \qquad (3\text{-}49)$$

Para ilustrar a aplicação desta regra de decodificaçao, será considerado o código de bloco binário (3,2,2) cuja treliça associada está apresentada abaixo

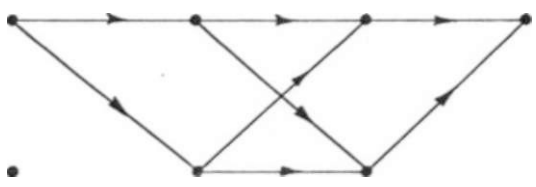


Figura 3.6 - TRELIÇA ASSOCIADA AO CÕDIGO LINEAR (3,2,2)

É assumido que a ênupla recebida é r = (1.7,.1,.5), e que o ruído tem media zero e variância unitária. Os passos no processo de„decodificação estão mostrados na figura 3.7, resultando na palavra código (0,0,0).

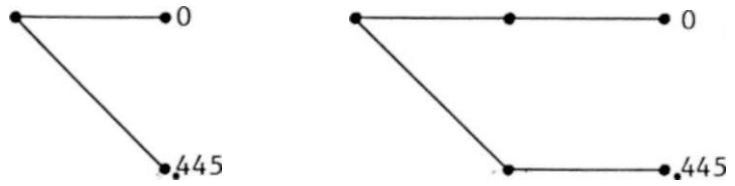


Figura 3.7 - PASSOS NA DECODIFICAÇÃO PARA O CÕDIGO (3,2,2).

Deve ser observado que o emprego de decisão abrupta resultaria na palavra (1,0,0), uma vez que os símbolos r^ são comparados com o limiar b = a/2 + y/2 = a/2". O erro corrigido foi exatamente na amostra mais próxima ao limiar (r = 1.7), ou seja, a amostrademenorconfiabilidade.

3.2.3.2 - Alta Relação Sinal/Ruído

Neste caso a distribuição de Rice pode ser aproximada ' por uma distribuição gaussiana |6

$$P(r\,J\,l) \sim \bullet \qquad \exp\left[-\frac{l^{r}p \sim^{A} - i \sim |^{2}}{\qquad}\right] U. \qquad (3\text{-}50)$$

De modo que o £og da razão de verossimilhança pode ser dado aproximadamente por

•) - Unr_{z}-lno_{e} **(3-51)**

onde = a**//2~n.**

Portanto, a palavra código escolhida deve maximizar a expressão **(3-52)** abaixo

$$\sum_{1=0}^{n-1} c_{p} \; (r \Lambda - A i / 2) - -4^{u}\Lambda i \; - \; {}^{ln}\circ 0) \qquad \textbf{(3-52)}$$

A extensão para aplicação desta regra na decodificaçao[1] de códigos convolucionais (nQ,kg,N) consiste, no caso binário, em escolher o caminho da treliça que maximiza a expressão $\sum_{1=0} c_{p}ZnÇf_{p}$, onde novamente o número de termos não nulos depende do comprimento da sequência transmitida.

O algoritmo de Viterbi para decodificação de códigos [1] convolucionais pode ser facilmente modificado para fazer uso desta regra de decodificação, e a melhoria provida pelo uso desta técnica de decisão suave encontra-se analisada na referência | 1 | .

3.2.4 — Análise do Comportamento Assintótico

Será deduzida uma expressão assintótica para a probabilidade de erro por palavra na decodificação ótima de códigos de bloco lineares binários, usando decisão suave [1] , para canal com ruído branco aditivo gaussiano, quando a relação sinal/ruído é elevada. A expressão obtida ê função da distribuição de pesos das palavras código e da relação sinal/ruído.

o Aqui a atenção será restrita aos códigos de bloco lineares binários com palavras código equiprováveis transmitidas sobre [1] um canal com ruído branco aditivo gaussiano através de pulsos liga-

desliga. É assumido que o código C(n,k,d) considerado tem palavras código denotadas por $c^{(i)}$, $i = 0,1,\ldots,2^k-1$, onde $c^{(i)}=(c^{(i)}_0, c^{(i)}_1, \ldots, c^{(i)}_{n-1})$.

A regra ótima de decodificação é escolher $c = c^{(i)} \in C$ que maximiza $P(c = c^{(i)}|r)$, onde c é a estimativa da palavra código transmitida e r é a palavra recebida. Como visto, isto pode ser realizado maximizando $\sum_{l=0}^{n-1} c_l \ln(\lambda_l)$, onde $\lambda_l = p(r_l|1)/P(r_l|0)$ é a razão de verossimilhança. Novamente a dedução dos resultados é simplificada assumindo que a palavra toda nula foi transmitida.

Conhecida a densidade espectral de potência unilateral do ruído $T_0/2$, as componentes vetor de ruído n são variáveis aleatórias i i d gaussianas, $n_i \sim K(0,T_0/2)$. Novamente será utilizada a relação sinal/ruído do canal $\gamma = E/N_0$ e a relação sinal/ruído por bit de informação transmitido $\gamma_b = E_b/N_0 = \gamma/R$.

Das considerações acima segue-se que a probabilidade de erro por bloco é

$$P_{bloco} = P[e \mid c=0] \tag{3-53}$$

Dado que a ënupla toda zero $c = \underline{0}$ foi transmitida, haverá erro na decodificação se e somente se $\sum_{l=0}^{n-1} c_l \ln(\lambda_l) > 0$ para alguma palavra código $c^{(i)}$, $i = 1,2,\ldots,2^k-1$.

Como os sinais transmitidos são pulsos de amplitude $\sqrt{E}$ volts ou 0 volts, e o canal é gaussiano, segue-se que

$$P(r_l|i) = \frac{1}{\sqrt{\pi N_0}} \exp\left[-\frac{(r_l - i\sqrt{E})^2}{N_0}\right] \qquad i = 0,1 \tag{3-54}$$

e portanto,

$$\sum_{l=0}^{n-1} c_l \ln(\lambda_l) = \frac{\sqrt{E}}{N_0/2} \sum_{l=0}^{n-1} c_l r_l \tag{3-55}$$

Como é suposto que a palavra toda zero foi a palavra transmitida, o vetor recebido $\underline{r} = \underline{c} + \underline{n} = (r_0, r_1, \ldots, r_{n-1})$ tem sua £-ésima componente dada por $r_\ell = n_\ell$.

Se são definidos eventos ; $i = 1,2,\ldots,2^k-1$,

A = { Z c *>-^- 2 c } (3-56)

então ê possível reescrever (3-53) como

$$P_{bloco} = P_f \bigcup_{i=1}^{2^k} V \qquad (3\text{-}57)$$

Utilizando o "union bound", obtêm-se a expressão:

$$P_{bloco} \leq \sum_{i=1}^{2^k-1} P(A_i) \qquad (3\text{-}58)$$

Por outro lado, $\sum_{\ell=0}^{n-1} c_{i\ell} = W(c_i)$, onde $W(c_i)$ representa o peso da palavra c_i. A variável aleatória $X_i = \sum_{\ell=0}^{n-1} c_{i\ell} n_\ell$ é uma combinação linear de variáveis aleatórias gaussianas independentes, o que acarreta em X_i também ser uma variável gaussiana | 9 \, com distribuição de probabilidade $X_i \sim N(0, W(c_i) \frac{N_0}{2})$. Portanto,

2-f-W(c.)) = —-— \ e x / n 0 W (- i } d

4 - w(5i»

ou seja,

$$P(A_i) = Q(\sqrt{E}\, W(c_i)/2\eta_0)$$

$$= Q(\sqrt{RY^\wedge \sim\sim W7CTT72}) \qquad (3\text{-}60)$$

A cota (3-58) é apertada para alta relação sinal/ruído' 19 I, de modo que é possível escrever

$$AB_{\gamma b\text{-}*^{\circ\circ}}(P_{\acute{o}loco}) \mathbf{P}_{\mathbf{1}=\mathbf{1}}^{2^{\wedge}1} l \quad P(A_{\mathbf{1}}.) \qquad (3\text{-}61)$$

* Contudo, somente palavras código com peso d contribuem substancialmente no somatório em (3-61), visto que para valores suficientemente grandes de x, $Q(x)=\frac{1}{x/2n}\exp(-x^{2}/2)$. Logo, o comportamento assintótico para a probabilidade de se decodificar incorretamente uma palavra quando a regra ótima é utilizada em um canal gaussiano com alta relação sinal/ruldo é expresso por

$$AB_{y\acute{b}\text{-}\gg\text{-}^{\circ}}(P_{b^{\prime\prime}loco} Is\ N(d)\ Q(/Rdy,_{b}/2)$$

Considerando pulsos polares de amplitude $+J\hat{E}$ volts e $-/\hat{E}$ volts, o resultado obtido coincide exatamente com o comportamento assintótico determinado para o algoritmo de Hartmann-Rudolph na seção 3.1.4.

CAPÍTULO IV

SIMULAÇÃO EM COMPUTADOR

Neste capítulo são apresentados os resultados de simulações realizadas em computador digital (DEC-10 SYSTEM) para análise do desempenho dos dois algoritmos descritos no capítulo anterior. A necessidade desta simulação advém da complexidade* do procedimento analítico da variação da probabilidade de erro (por símbolo ou bloco) em função da relação sinal/ruído, e da degradação que íesulta quando decisão suave é utilizada quantizando em Q regiões os símbolos recebidos do canal. O desempenho de vários códigos é analisado, e em particular, os códigos de Hamming (7,4,3), (15,11,3) e (31,26,3) são considerados. O intuito i fazer um levantamento das curvas de probabilidade de erro versus relação simal/ruído para cada código de bloco analisado, em um dado canal sem memória. Isto torna possível, por exemplo, que seja feita uma visualização da melhoria provida pelo uso de decisão suave com relação a decisão abrupta e com relação ao número de regiões de quantização utilizado.

Os programas foram escritos na linguagem FORTRAN 10 ' e se encontram listados no apêndice A.

*Veja apêndice B.

4.1 - ANALISE DO DESEMPENHO DO ALGQRÍTMO DE HARTMANN-RUDOLPH

4.1.1 - Considerações Gerais

O desempenho do algoritmo de Hartmann-Rudolph será analisado para vários códigos de bloco lineares, quando os sinais transmitidos são pulsos liga-desliga sujeitos a ação de um ruído aditivo gaussiano. Em particular, resultados para os códigos de Hamming são apresentados.

Na simulação, primeiro é especificado o código linear binário C(n,k,d) fornecendo sua matriz de verificação de paridade [H] , ou o polinómio gerador g(x] no caso de códigos cíclicos. A seguir é fixado o número de regiões de quantização a ser utilizado na simulação, usualmente uma potência de dois. A análise é realizada considerando 2,4,8 ou até 16 regiões de quantização.

A simulação ê feita transmitindo-se a palavra toda zero em um canal com ruído aditivo gaussiano, gerado de acordo com o método polar 122 I• Não há nenhuma perda de generalidade em se assumir que a palavra transmitida foi a palavra toda nula |l9|.

Para os códigos de Hamming analisados, a opção de utilizar a palavra transmitida como sendo a palavra toda um, forneceu praticamente os mesmos resultados obtidos quando a palavra transmitida foi a toda nula, mostrando que o ruído gerado apresenta simetria. Os resultados para o código (7,4,3) também foram verificados considerando os bits do bloco de mensagem gerados de acordo com uma sequência pseudo—aleatória |n |.

As probabilidades de erro por símbolo e por bloco são então estimadas para cada valor de relação sinal/ruído (em decibeis) transmitindo um número suficientemente grande de palavras código através do canal e utilizando a regra ótima de decodificação no receptor. A estimativa é feita utilizando a frequência relativa de

ocorrincia de erros na salda do decodificador.

A utilização da versão fraca da lei dos grandes números de acordo com o teorema de BERNOULLI 9| assegura que os resulta - dos obtidos convergem para os valores reais das probabilidades, pois se ⁿ£/ⁿ é a frequência relativa da ocorrência de erros, então

$$P\left\{\left|\frac{l}{n} - p\right| > \varepsilon\right\} < \frac{1}{4n\varepsilon^2} \quad \text{Para todo } \varepsilon > 0. \qquad (4\text{-}1)$$

Deve ser notado que o número de palavras a serem trans- mitidas pela fonte deve ser aumentado â medida que a relação sinal/ ruído cresce. Tal aumento de relação sinal/ruído acarreta uma dimi- nuição no número de erros provocados pelo canal e consequentemente a interpretação frequentista utilizada para o calculo das probabili dades de erro poderá não mais ser válida. Pode ser facilmente de monstrado 133 | que» para uma variável aleatória com distribuição b i - nominal OJ (n,p)> um estimador não enviezado para p 5a frequên - cia relativa p = k/n, o qual resulta em um espalhamento relativo ¹ em torno do valor esperado da estimativa dado por

$$E\{p\} \quad /£ \qquad (4\text{-}2)$$

Se p representa a probabilidade de erro por b i t na saída do decodificador, e q a probabilidade do evento complemen - t a r, então para p<< 1 segue-se q s l, de modo que

$$\sigma_p = \sqrt{\frac{p}{n}} \qquad (.4\text{-}3).$$

Por exemplo, se a probabilidade de erro ê da ordem de 10 ^ e ê desejado um espalhamento relativo de 10%, então a simula - ção deve ser realizada para n da ordem de 10^ bits transmitidos , Desta forma, para SNR = 4dB ê suficiente que o número de bits

transmitidos seja da ordem de mil bits, enquanto que para SNR = 12 dB é necessário que este número seja cerca de 100 mil bits, para as segurar a validade dos resultados encontrados.

Como a palavra decodificada pelo algoritmo não é necessariamente uma palavra código, surgiu a ideia de escolher como estimativa da palavra transmitida a palavra código mais próxima em termos de distância de Hamming da palavra decodificada. Contudo, os resultados obtidos como o emprego de tal procedimento degradaram sensivelmente a probabilidade de erro por símbolo, enquanto que a melhoria na probabilidade de erro por bloco foi praticamente nula.

O tempo de CPU requerido para realização das simulações é razoavelmente grande, como seria de se esperar, não chegando contudo, a tornar-se proibitivo, exceto para SNR's muito elevadas. Contudo, para altas relações sinal/ruído, o comportamento assintõtico[1] do algoritmo é conhecido.

De posse dos resultados obtidos, foram traçadas as curvas de probabilidade de erro versus relação sinal/ruído mostradas a seguir, nas páginas 108 a 124 .

4.1.2 - Curvas de Desempenho

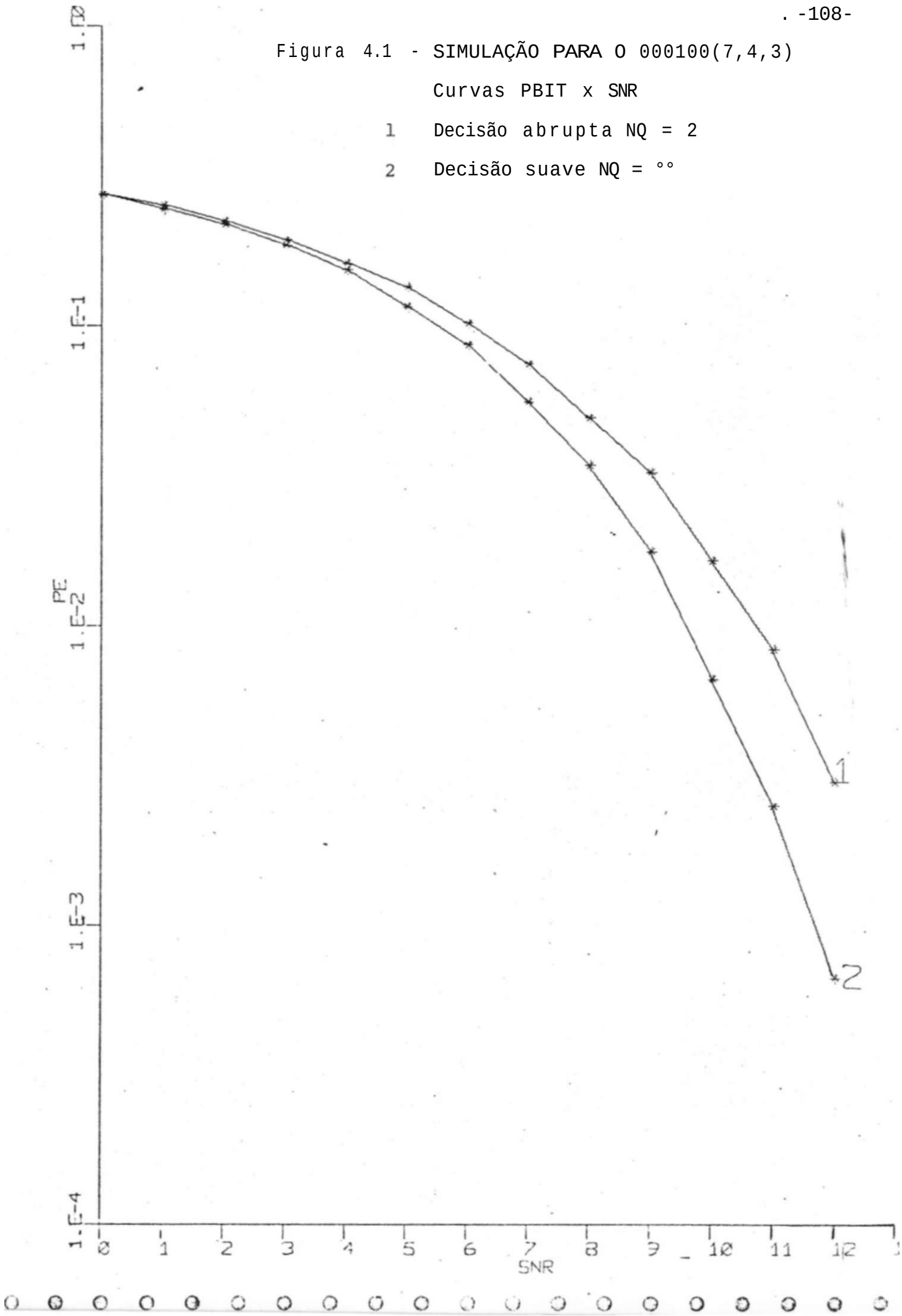
As curvas de desempenho para o algoritmo de Hartmann - Rudolph obtidas através de simulação em computador digital estão[1] apresentadas nesta seção. É admitido um canal perturbado por um ruído branco gaussiano aditivo em todos os casos. O número de regiões' de quantização é abreviado por NQ e PBIT denota a probabilidade de erro por símbolo, enquanto que PBLOCO denota a probabilidade de erro por palavra.

Figura 4.1 - SIMULAÇÃO PARA O 000100(7,4,3)

Curvas PBIT x SNR

1 Decisão abrupta NQ = 2

2 Decisão suave NQ = °°



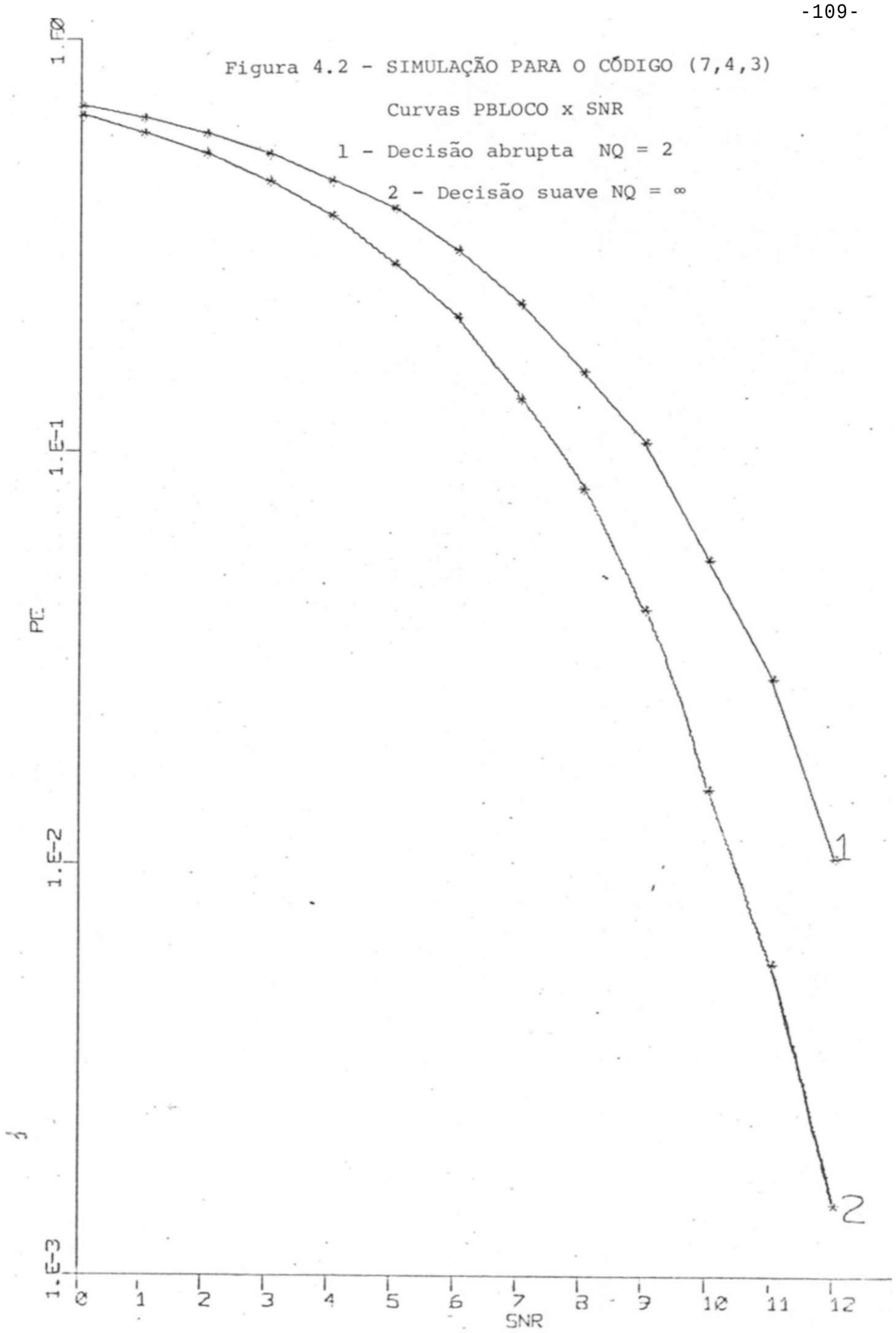


Figura 4.2 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)

Curvas PBLOCO x SNR

1 - Decisão abrupta NQ = 2

2 - Decisão suave NQ = ∞

Figura 4.3 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)

Curvas PBIT x SNR

1 - NQ = 2

2 - NQ = 4

3 - NQ = 8

4 - NQ = 16

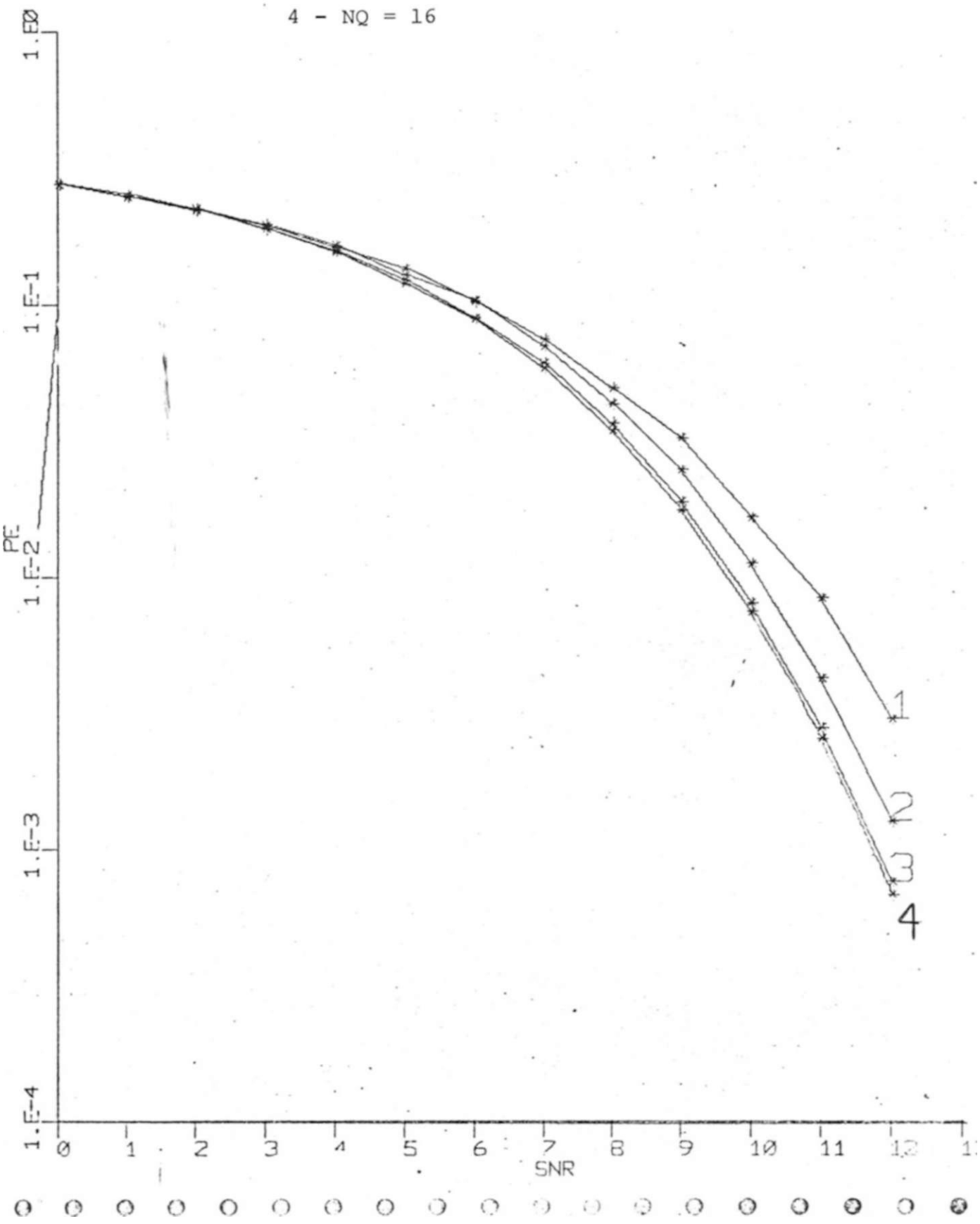


Figura 4.4 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)

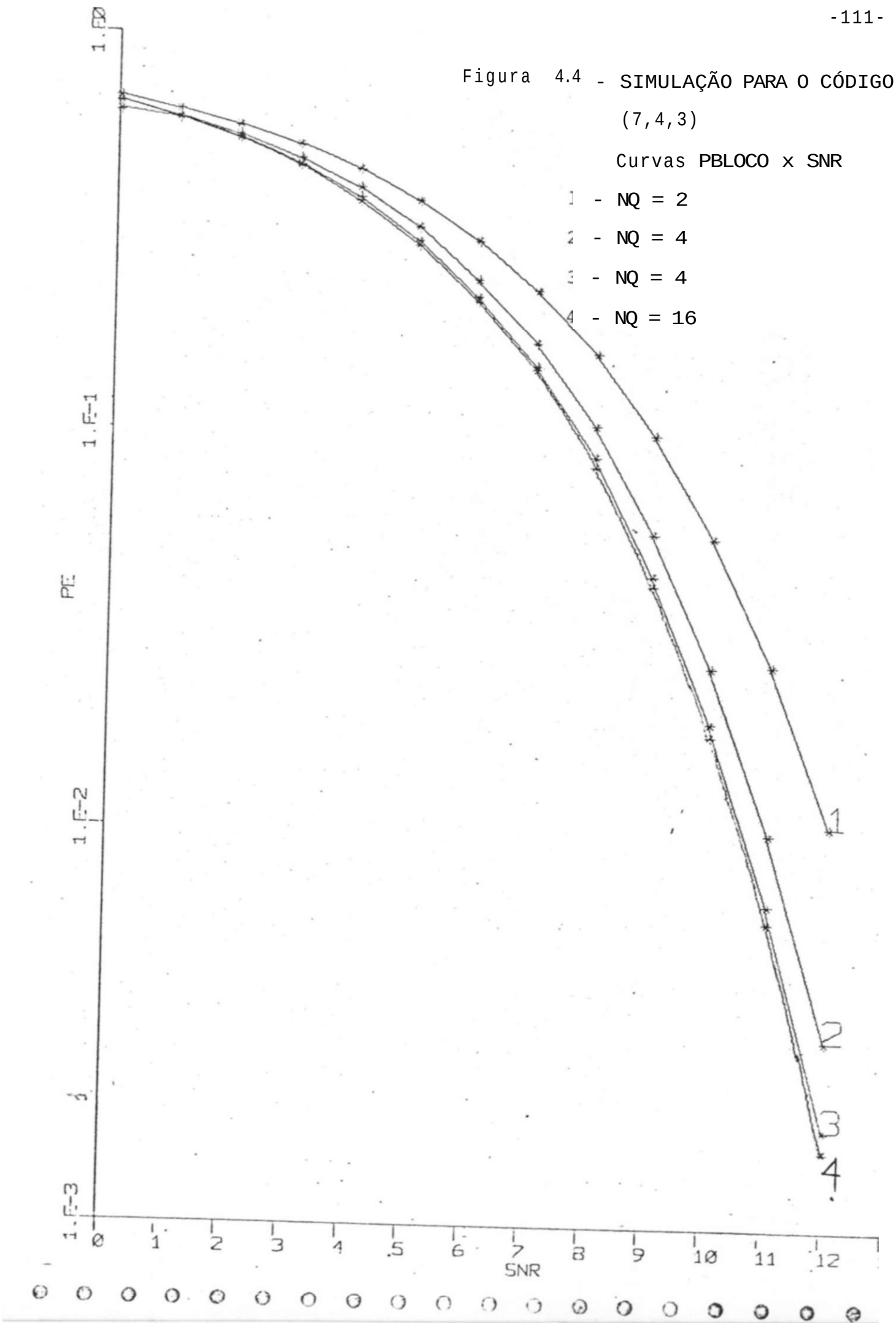
Curvas PBLOCO x SNR

1 - NQ = 2

2 - NQ = 4

3 - NQ = 4

4 - NQ = 16



Fi-gura 4.5 - SIMULAÇÃO PARA O CÓDIGO (15,11,3)

Curvas PBIT x SNR

1 - NQ = 4

2 - NQ = 8

3 - NQ = 16

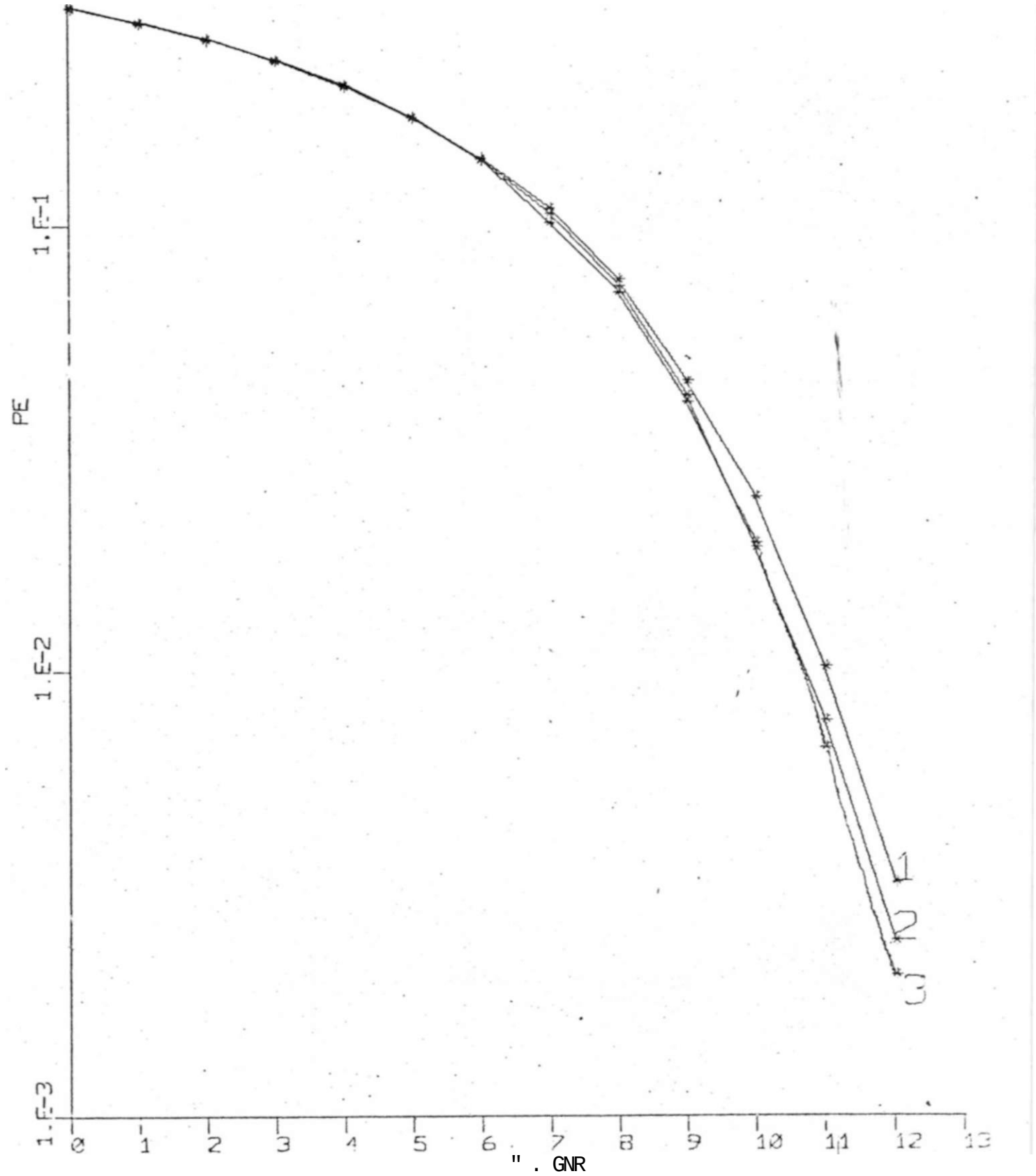


Figura 4.6 - SIMULAÇÃO PARA O CÓDIGO (15,11,3)

Curvas PBLOCO x SNR

1 - NQ = 2

2 - NQ = 4

3 - NQ = 8

4 - NQ = 16

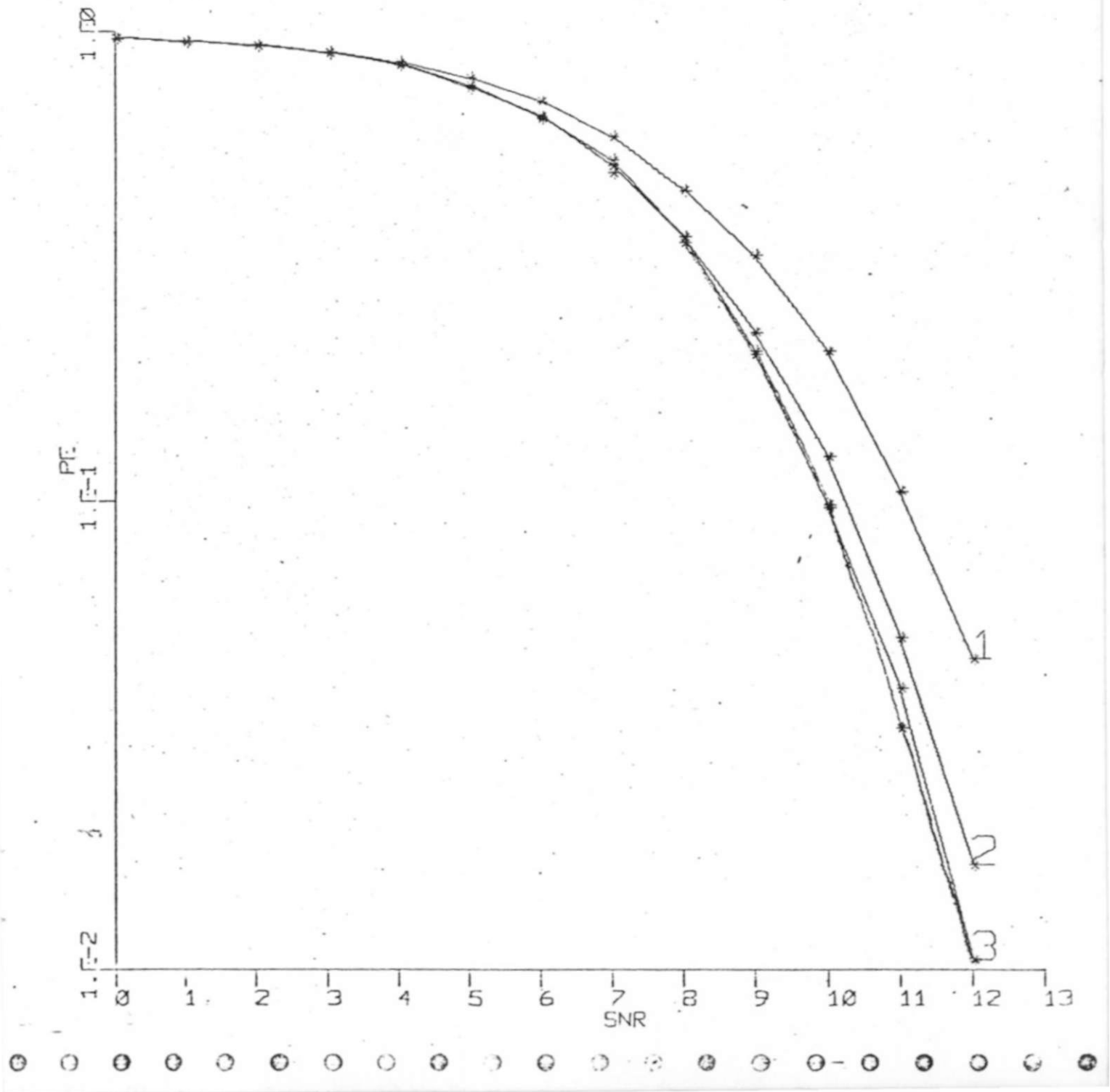


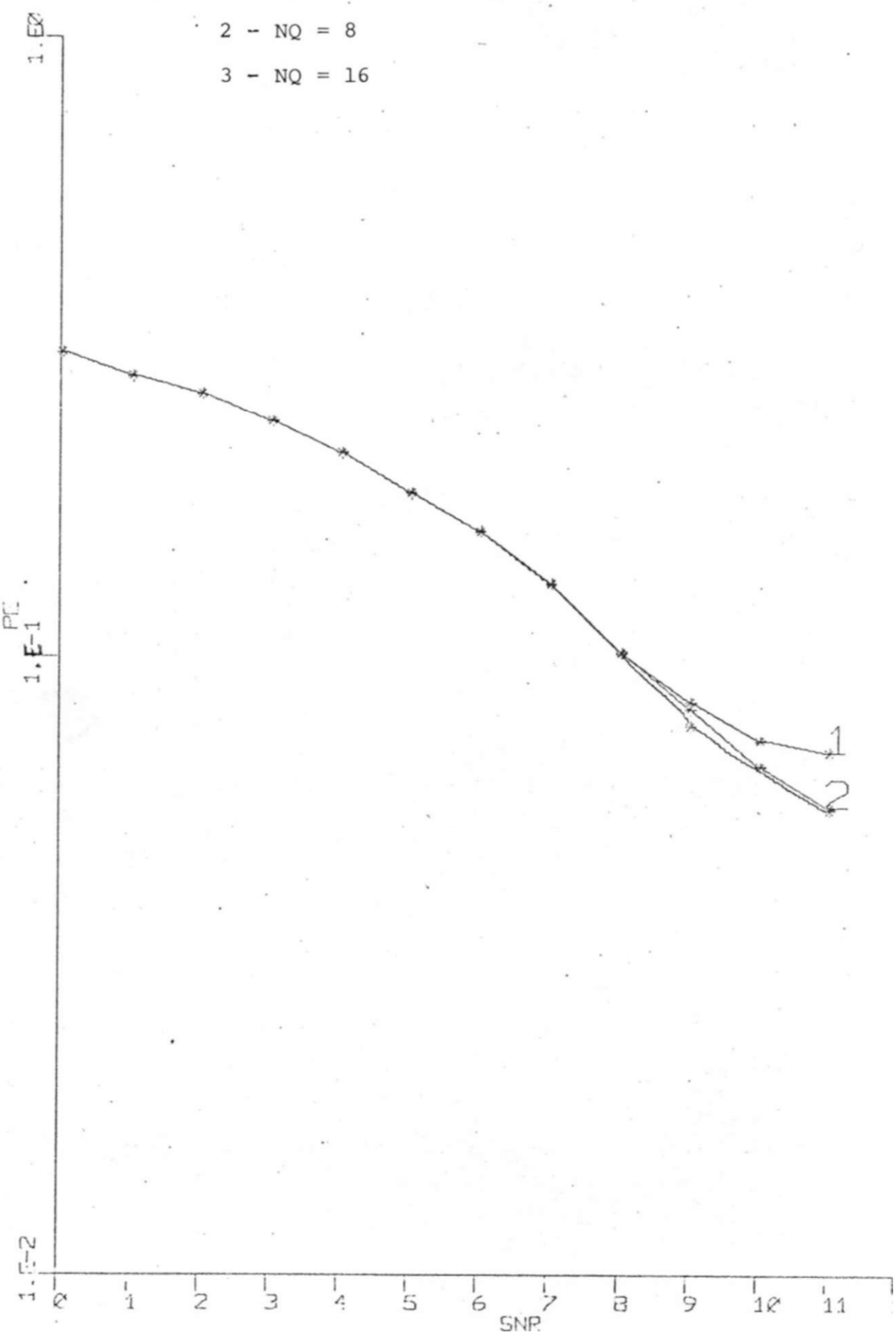
Figura 4.7 - SIMULAÇÃO PARA O CÓDIGO (31,26,3)

Curvas PBIT x SNR

1 - NQ = 4

2 - NQ = 8

3 - NQ = 16



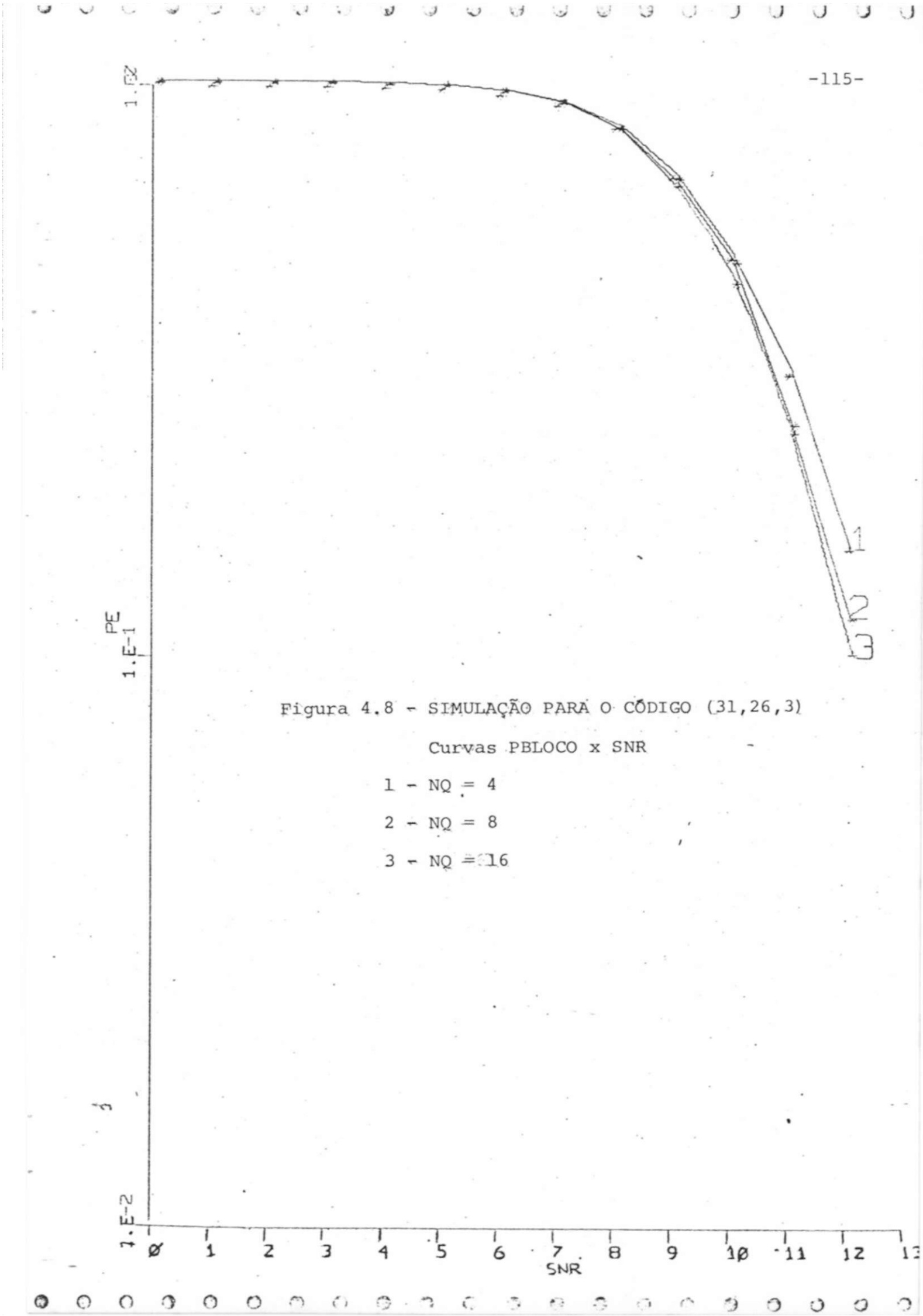


Figura 4.8 - SIMULAÇÃO PARA O CÓDIGO (31,26,3)

Curvas PBLOCO x SNR

1 - NQ = 4

2 - NQ = 8

3 - NQ = 16

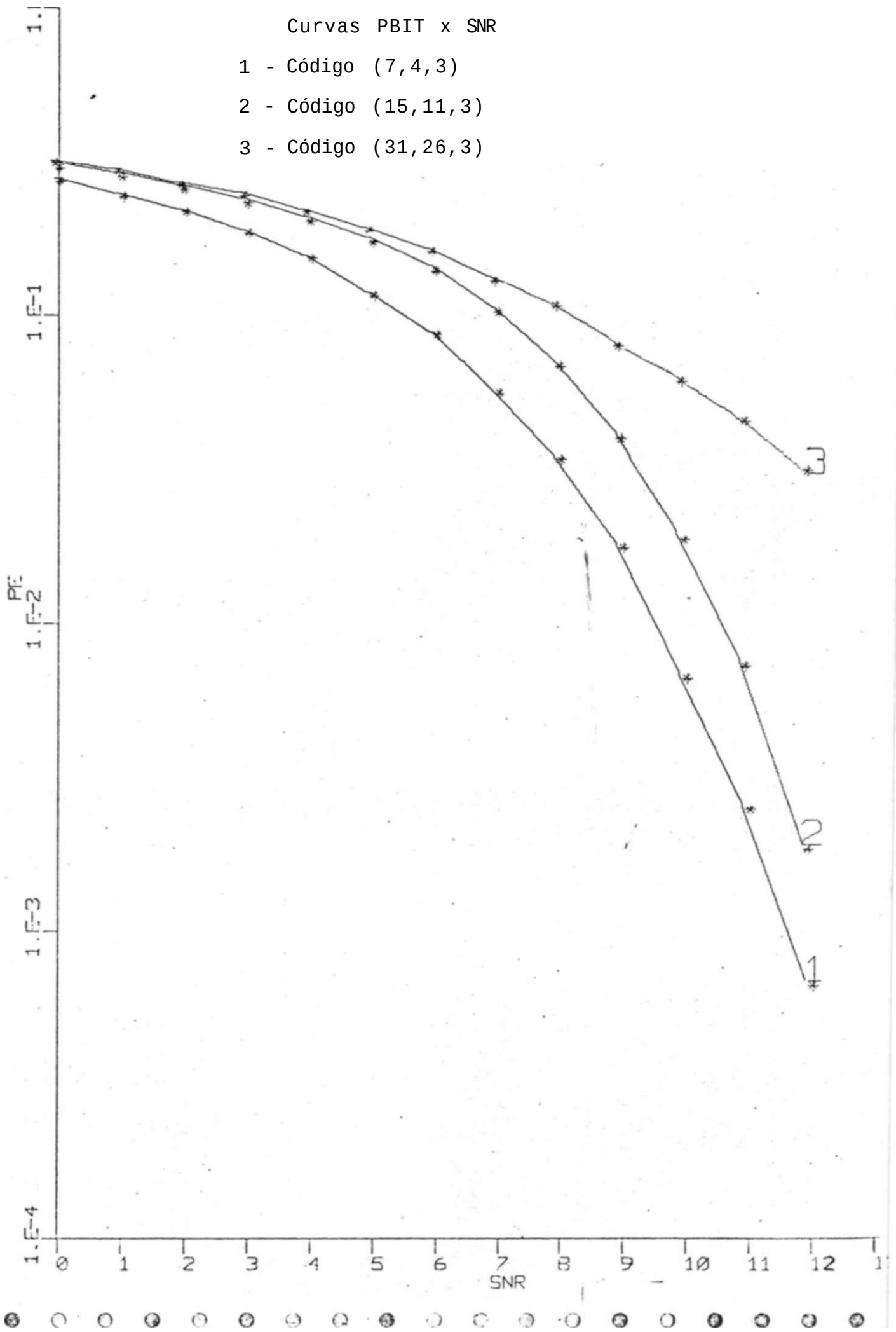
Figura 4.9 - SIMULAÇÃO PARA NQ

Curvas PBIT x SNR

1 - Código (7,4,3)

2 - Código (15,11,3)

3 - Código (31,26,3)



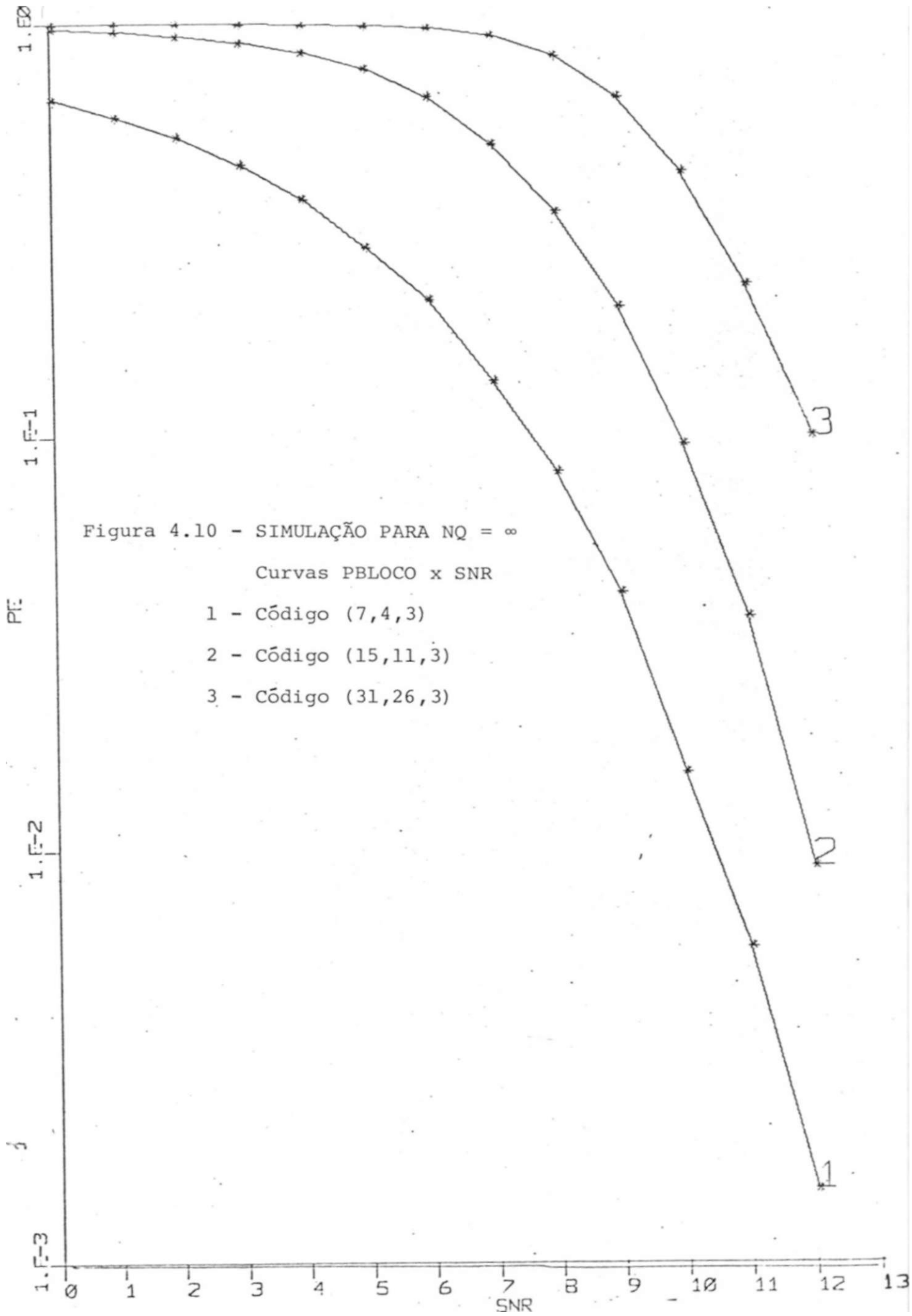


Figura 4.10 - SIMULAÇÃO PARA NQ = ∞

Curvas PBLOCO x SNR

1 - Código (7,4,3)

2 - Código (15,11,3)

3 - Código (31,26,3)

j Q o o o **o** o o- *O o* **O** D o o o **o o o o** <

Figura 4.11 - SIMULAÇÃO PARA NQ = 2 -118-

Curvas PBLOCO x SNR

1 - Código (7,4,3)

2 - Código (15,11,3)

3 - Código (31,26,3)

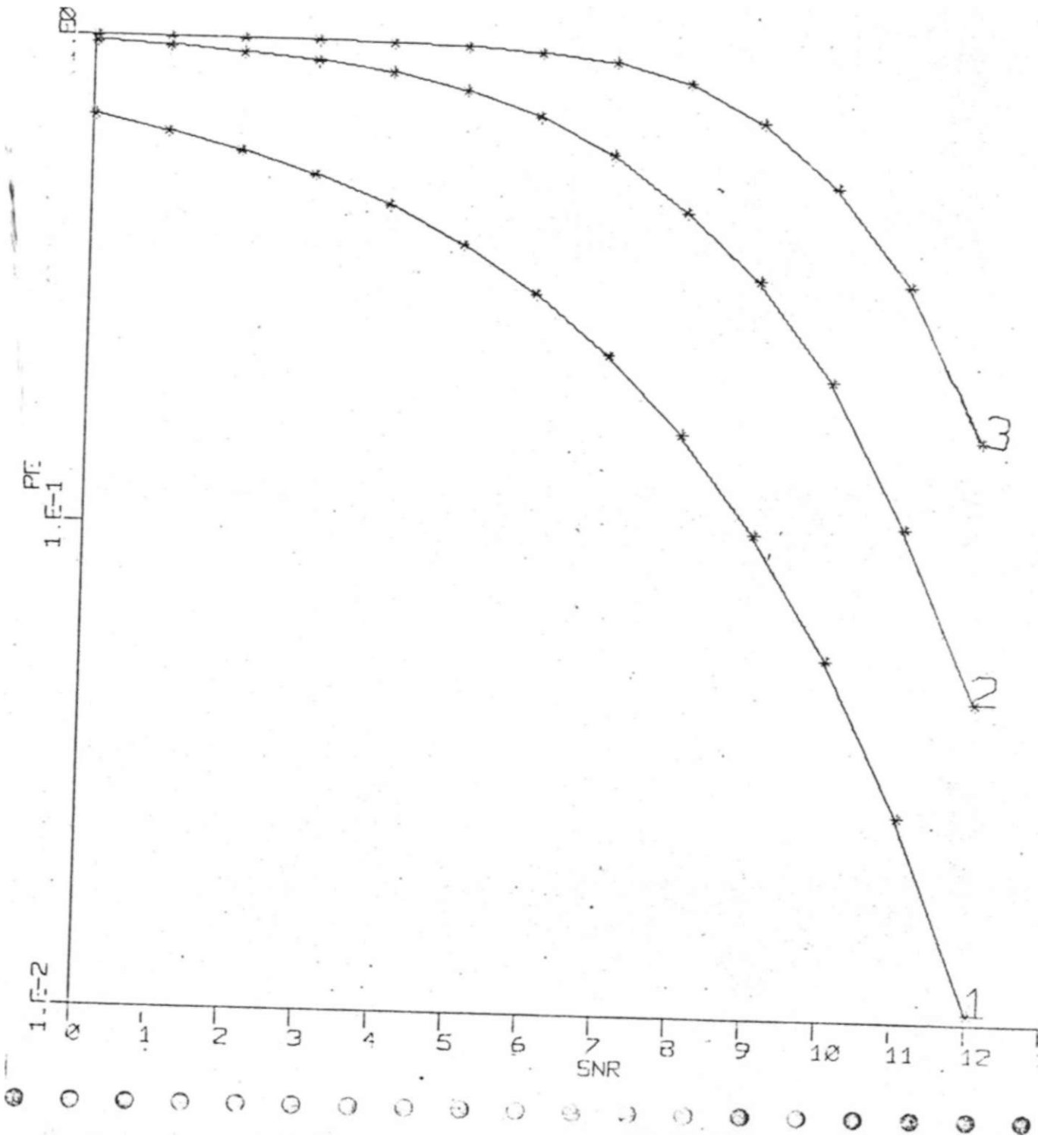


Figura 4.12 - SIMULAÇÃO PARA NQ = 4

Curvas PBIT x SNR

1 - Código (7,4,3)

2 - Código (15,11,3)

3 - Código (31,26,3)

Figura 4.13 - SIMULAÇÃO PARA NQ = 4

Curvas PBLOCO x SNR

1 - Código (7,4,3)
2 - Código (15,11,3)
3 - Código (31,26,3)

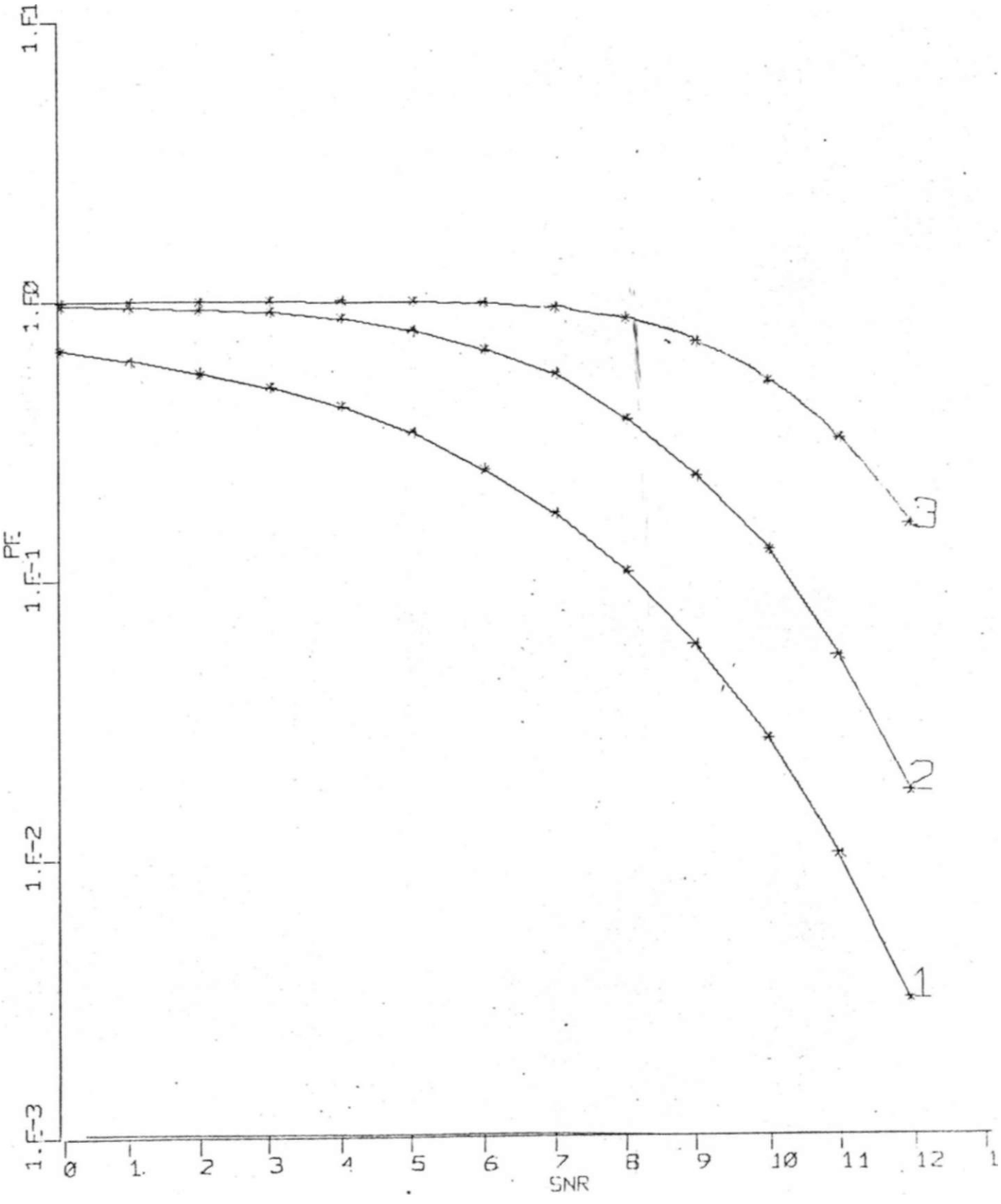


Figura 4.14 - SIMULAÇÃO PARA NQ = 8

Curvas PBIT x SNR

Código (7,4,3)

Código (15,11,3)

Código (31,26,3)

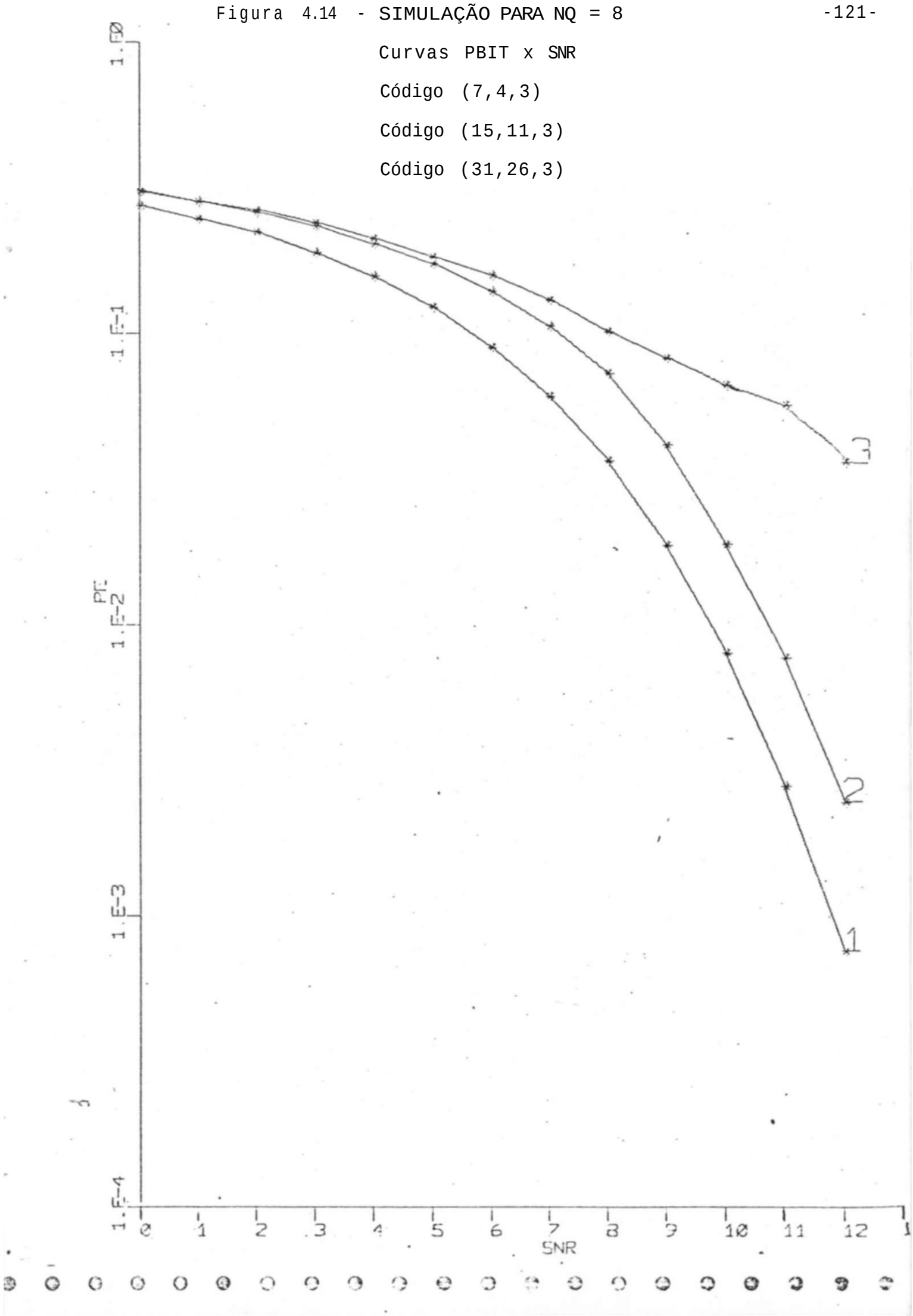


Figura 4.15 - SIMULAÇÃO PARA NQ = 8

Curvas PBLOCO x SNR

1 - Código (7,4,3)

2 - Código (15,11,3)

3 - Código (31,26,3)

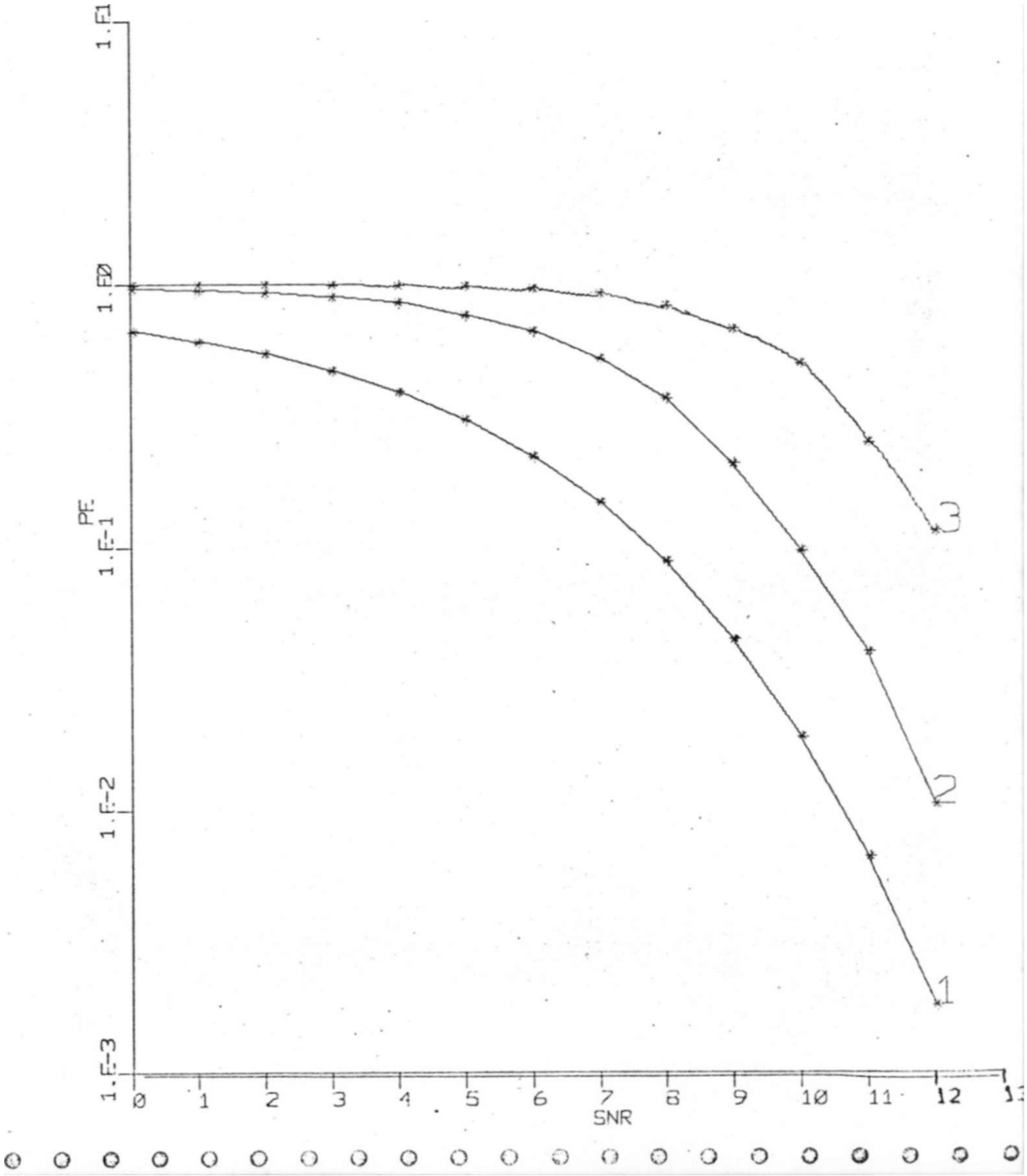


Figura 4.16 - SIMULAÇÃO PARA NQ = 16

Curvas PBIT x SNR

1 - Código (7,4,3)

2 - Código (15,11,3)

3 - Código (31,26,3)

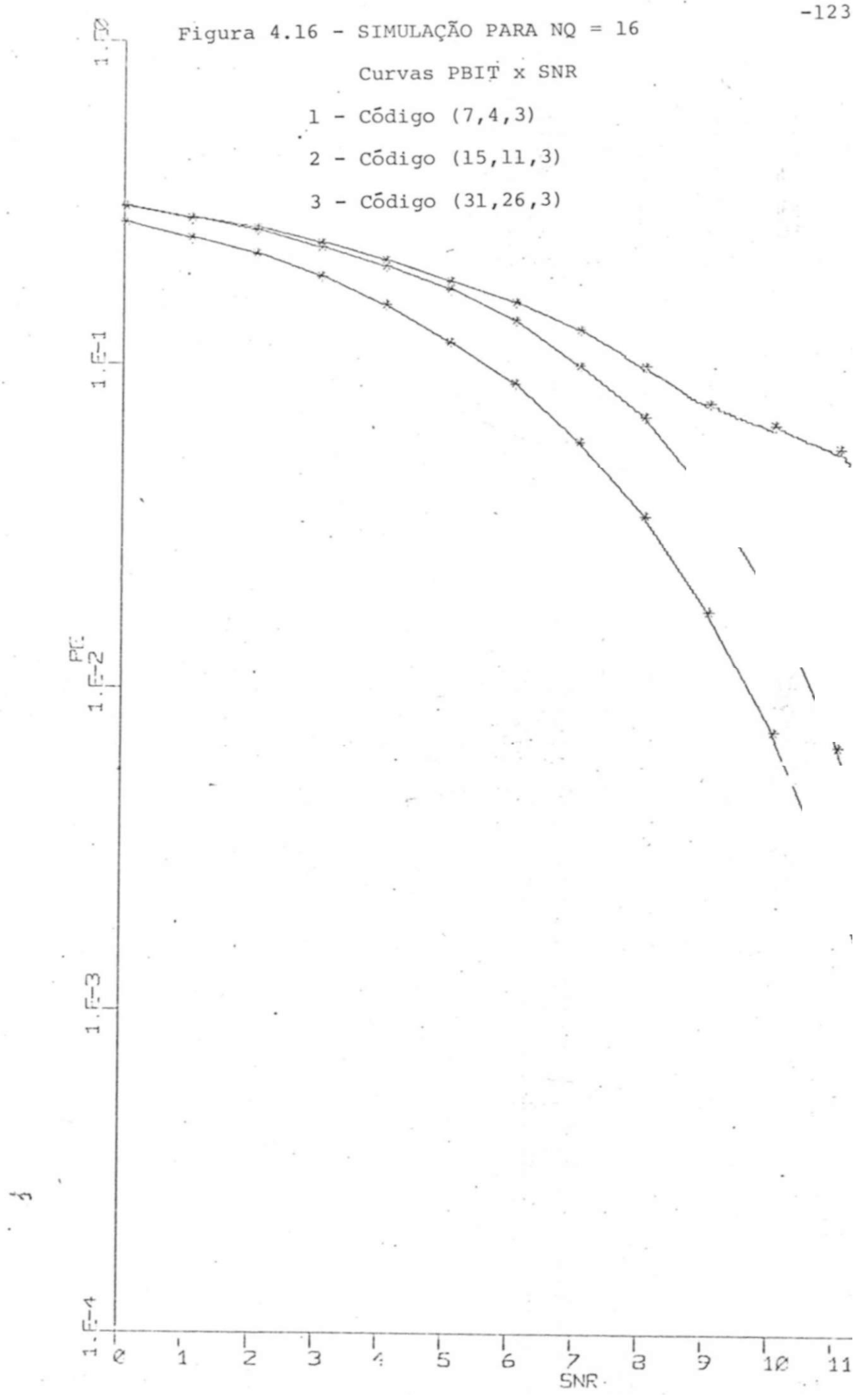


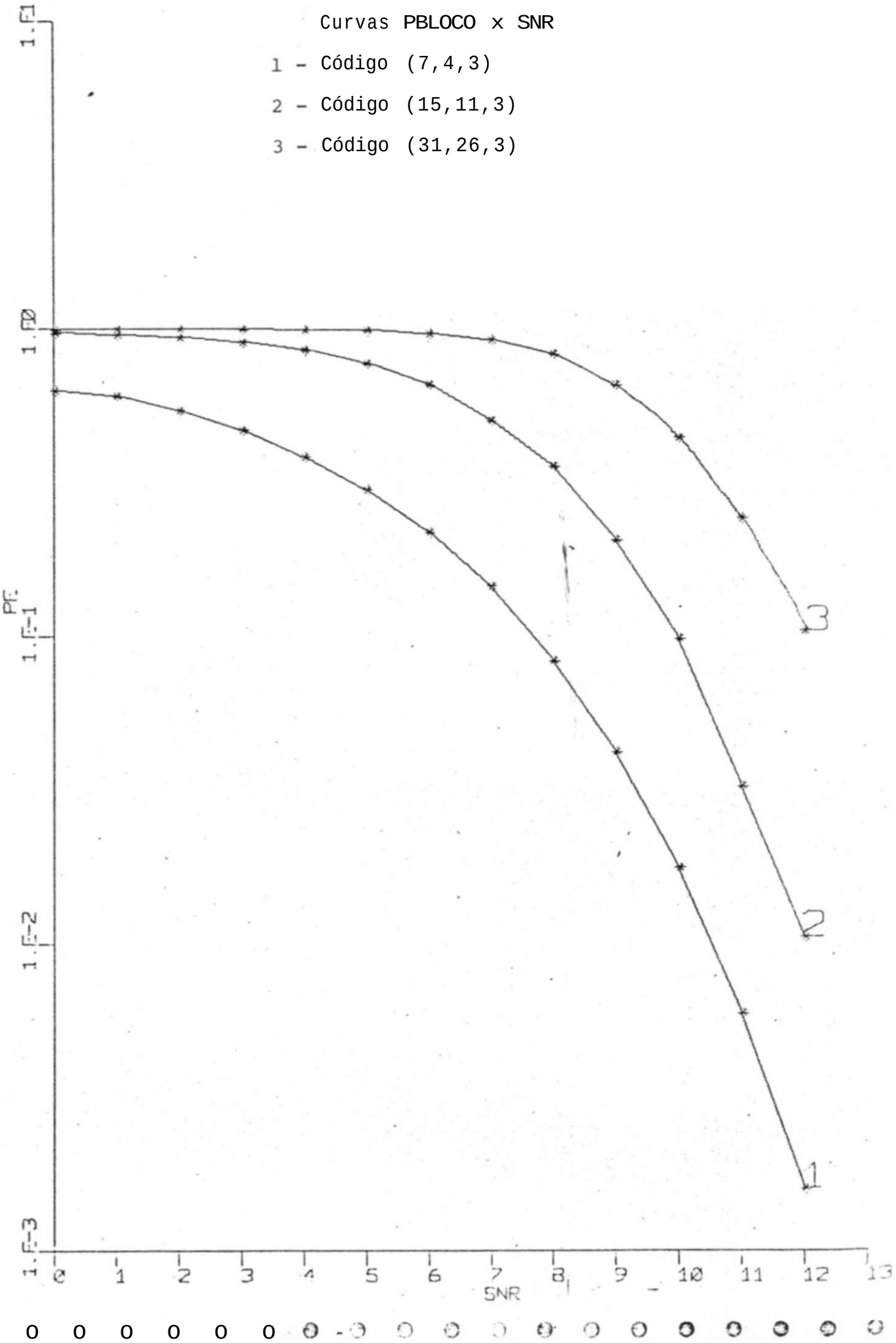
Figura 4.17 - SIMULAÇÃO PARA NQ = 16

Curvas PBLOCO x SNR

1 – Código (7,4,3)

2 – Código (15,11,3)

3 – Código (31,26,3)



& o o o o o o o o - o o o o o o o o o o o o

4.2- DESEMPENHO DA DECQDIFICAÇÃO UTILIZANDO MÁXIMA PROBABILIDADE A POSTERIORI

4.2.1 - Considerações Gerais

Uma avaliação será apresentada do desempenho do segundo algoritmo ótimo descrito no capítulo anterior(sec.3.2)para vários ' códigos de bloco lineares, supondo sinais transmitidos em pulsos de RF liga-desliga através de um canal ruidoso.

O algoritmo fará uso da treliça associada ao código para simplificar as operações de decodificação. Inicialmente um código de bloco linear em GF(q) ê especificado através de (n,k,d) e da sua matriz de verificação de paridade. A treliça associada ao código ê então gerada de acordo com um procedimento melhor esclarecido' na seção 4.2.2.

As palavras código transmitidas pela fonte são geradas aleatoriamente e de tal forma que possuam uma distribuição de probabilidade uniforme. Deste modo, a hipótese requerida para otimalidade do algoritmo é obedecida. 0 ruído na representação de banda estreita é novamente simulado utilizando o método polar 122 | • Serão abordados apenas códigos lineares binários, muito embora a treliça' possa também ser gerada para códigos de bloco lineares multiníveis, As probabilidades de erro por palavra e por bit quando este algoritmo de decodificação é utilizado na recepção são determinadas para [1] cada valor de relação sinal/ruído (em dB), transmitindo um numero ' suficientemente grande de palavras código através do canal ruidoso. A estimativa utilizada para determinação destas probabilidades ê novamente a frequência relativa da ocorrência de erros, de modo que as considerações sobre convergência dos resultados discutidas na seção anterior ainda permanecem válidas. Com relação ao tempo de CPU requerido, considerações similares as enunciadas na seção anterior' também são aplicáveis.

4.2.2 - Implementação Computacional da Treliça

Será agora definida uma matriz associada â treliça gerada por um código de bloco linear em GF(q), com o intuito de permitir uma implementação da treliça sob o ponto de vista computacional [7].

Seja [M] uma matriz com q^{n-k} linhas e n colunas, onde cada elemento desta matriz é um vetor de q posições, i.e.r

$$[M] = \begin{matrix} m_{1,1} & m_{1,2} & \cdots & m_{1,n} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,n} \\ \vdots & \vdots & & \vdots \\ m_{q^{n-k},1} & m_{q^{n-k},2} & \cdots & m_{q^{n-k},n} \end{matrix} \qquad (4\text{-}4)$$

com $m_{i,j} \in \{0,1,\ldots,q-1\}$, $1 \le i \le q^{n-k}$, $1 \le j \le n$.

Portanto, o numero de posições de memoria necessário para definir uma treliça de um código linear (n,k,d) em GF(q) será de

$$n\, q^{n-k}.q = n\, q^{n-k+1} \text{ posições.} \qquad (4\text{-}5)$$

Cada vetor coluna de [M] está associado a uma profundidade $k > 0$ da treliça, e cada elemento $u_{1,k}$ deste vetor coluna está associado a um nó da mesma. Cada elemento $u_{i,k}$ por sua vez possui sua coordenada $u_{i,k}$ igual a numeração correspondente ao nó k da profundidade anterior tal que $s(k) = s_{(k-1)} + h_k$. Deste

modo, cada vetor m^ da profundidade k indica quais os nós e elementos et^ e GF(q) que deram origem ao nó ao qual o vetor nu^. está associado. Fixados um dado nó NO e uma profundidade PROF, MATRIX(NO, PROF, ALFA), ALFA = Cl, 1 , . . , , q - 1 define uma q-upla correspondente a ™NO PROF* A primeira posição da q-upla significa que deve ser considerado que o nó em questão foi atingido através de a = 0, enquanto a segunda posição significa que o ramo considerado se dirige para o nó em questão através de a = 1, e assim por diante. Os valores indicados na q-upla fornecem os nós na profundidade anterior que se ligam com o nó em questão. É assumido que o valor zero indica que nenhuma ligação deve ser considerada. Como exemplo, o valor de MATRIX(.3,2,0) fornece qual o nó na profundidade 1, o qual através de a = 0 atinge o nó 3 na profundidade 2.

O procedimento utilizado para expurgar a treliça associada ao código de bloco considera inicialmente a construção de duas treliças auxiliares. A primeira de-Ias, TRELLI 1, ê gerada a partir da profundidade k = Q até a profundidade k = n; enquanto que a segunda, TRELLI 2, ê gerada iniciando na profundidade k = n até a profundidade k = 0. Os caminhos que coincidirem nas duas treliças fazem parte da treliça expurgada, em caso contrário o caminho ê apagado. Este procedimento ê melhor compreendido quando apresentado nos exemplos mostrados a seguir.

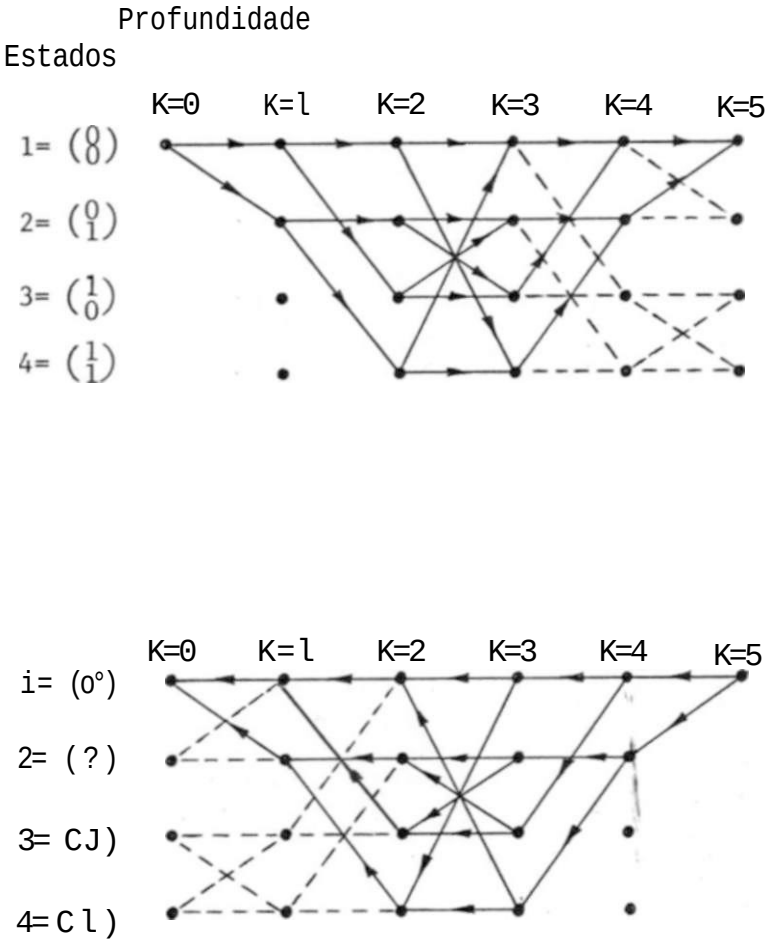
Exemplo 4.1 - Caso binário.

Dado o código de bloco (5,3,2) com matriz de verificação de paridade [H]

$$[H] = \begin{matrix} d & l \\ 1 & Q \end{matrix} \begin{matrix} 1 & 2 & 1 & Q \\ í & ! & Q & í \end{matrix}$$

A matriz [M] que representa a treliça associada ao código é construída de acordo com os passos mostrados abaixo.

Tem-se n=5, k=3, n-k=2, d=2, q=2 e q^"~^ = 4,



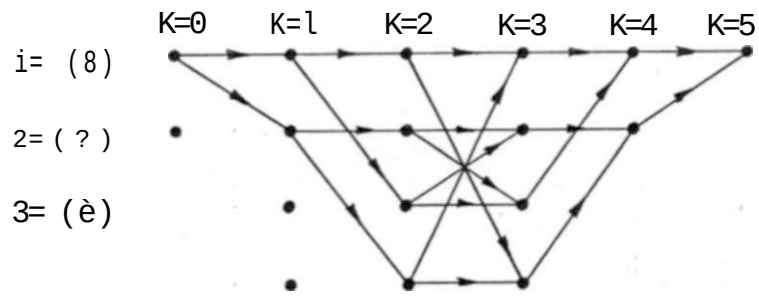
TRELLI 1

[l.Ql [l,Õ) [1,4] [1,3] [1,2)
[0,1] [2,0] [2,3] [2,4] [2,g
[0,0] [0,| [3,2] [3,1] [3,4]
[0,0] [0,| [4,1] [4,2] [4,|

TRELLI 2

[1,2] [1,| [1,4] [1,3] [1,1
[2,j [2,4] [2,| [2,4] [0,3
[3,4] [3,3 [3,2] [0,0] [0,(5
M [4,g [0,0] [0,^

Finalmente a treliça expurgada associada ao código binário (5,3,2) é encontrada usando TRELLI 1 e TRELLI 2,



TRELLI

[1,0] [1,0] [1,4] [1,1 [1,2]
[0,1] [2,0] [2,3] [2,4] [0,0]
[0,0] [0,1] [3,2] [0,0] [0,0]
[0,0] [0,2] [4,5 [0,Õ| [0,0]

Exemplo 4.2 - Caso multinível.

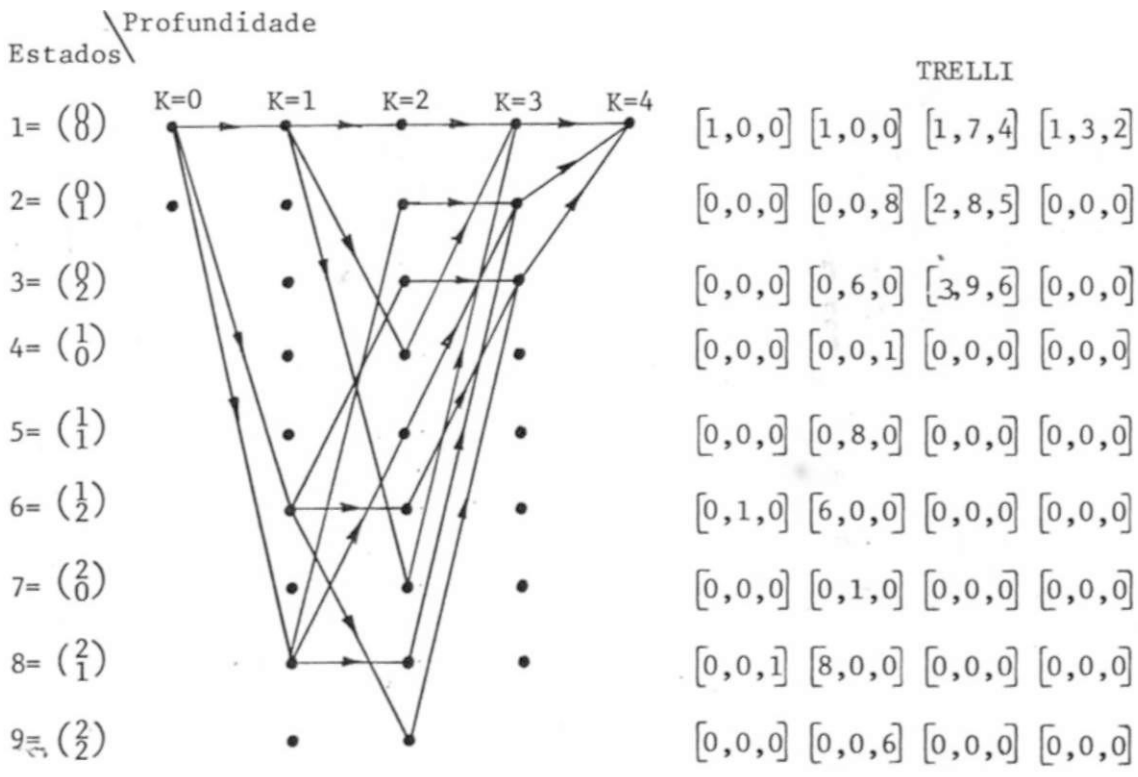
Como um segundo exemplo será considerado um código de bloco (4,2,3) definido em GF(3), com matriz de verificação de paridade dada por

$$[H] = \begin{matrix} 1 & 2 & : & 1 & 0 \\ 2 & 2.*0 & & 1 & \end{matrix}$$

Os parâmetros são:

$$n=4,\ k=2,\ n-k=2,\ d=3,\ q=3 \text{ e } q^{n-k}=9,$$

A treliça associada a este código linear e a matriz [M] representando a treliça são mostradas:



A geração da treliça permite a implementação de diver - sos algoritmos para decodificação de códigos de bloco lineares, al - guns apresentados nesta tese.

4.2.3 - Curvas de Desempenho

As curvas de desempenho para o algoritmo de maximização da probabilidade a posteriori das palavras código obtidas através de simulação em computador digital estão apresentadas nesta seção [1]. As considerações descritas na página 107 sobre a notação utilizada' novamente se aplicam. Estas curvas permitem uma visualização da melhoria obtida com uso de decisão suave. O caso onde decisão suave é utilizada quantizando em Q regiões as amostras recebidas do canal é também considerado.

Figura 4.18 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)

Curvas PBIT x SNR

1 - Decisão abrupta NQ = 2

2 - Decisão suave NQ = ∞

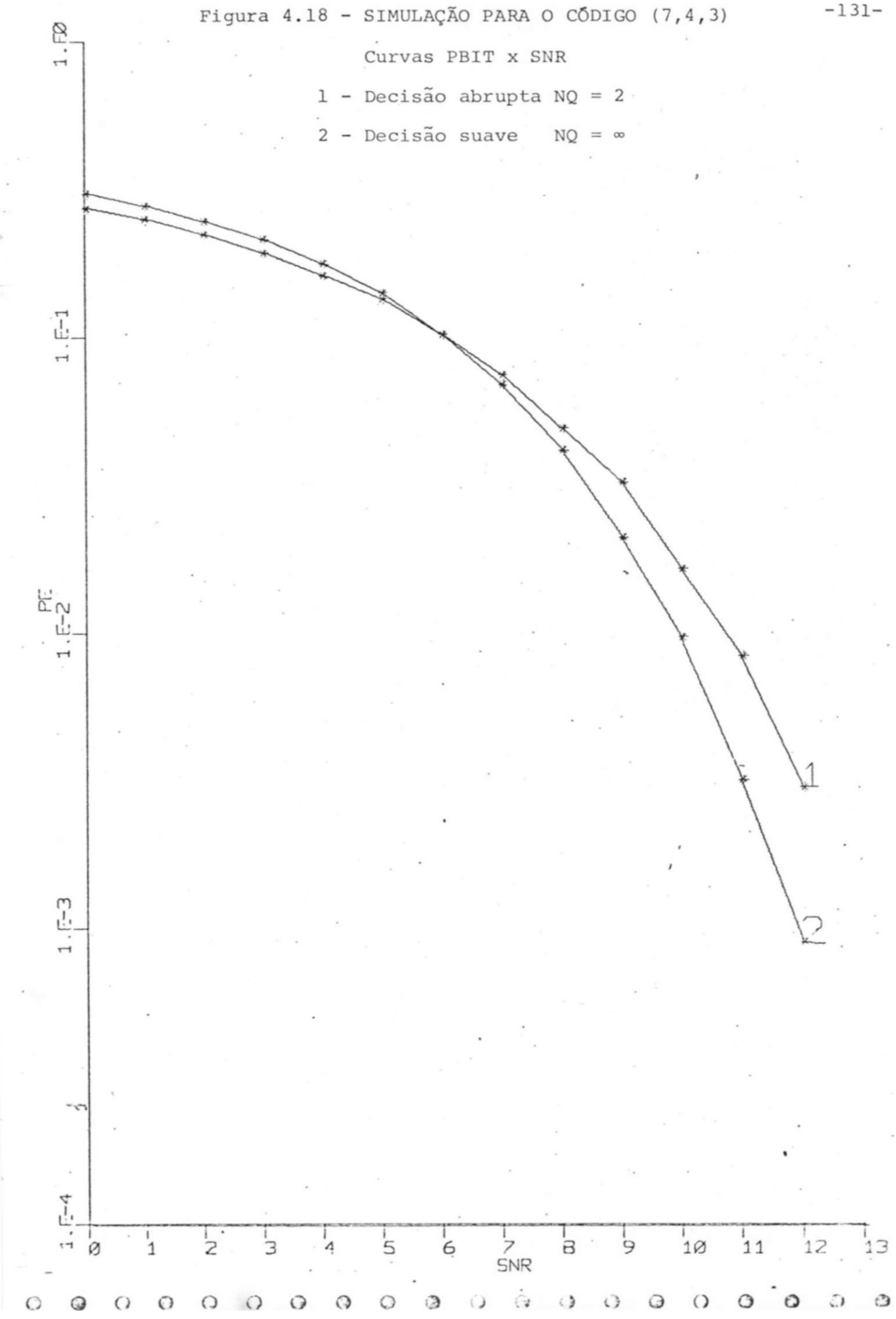


Figura 4.19 - SIMULAÇÃO PARA O CÕDIGO (7,4,3;
Curvas PBLOCO x SNR
1 - Decisão abrupta NQ = 2
2 - Decisão suave NQ = ∞

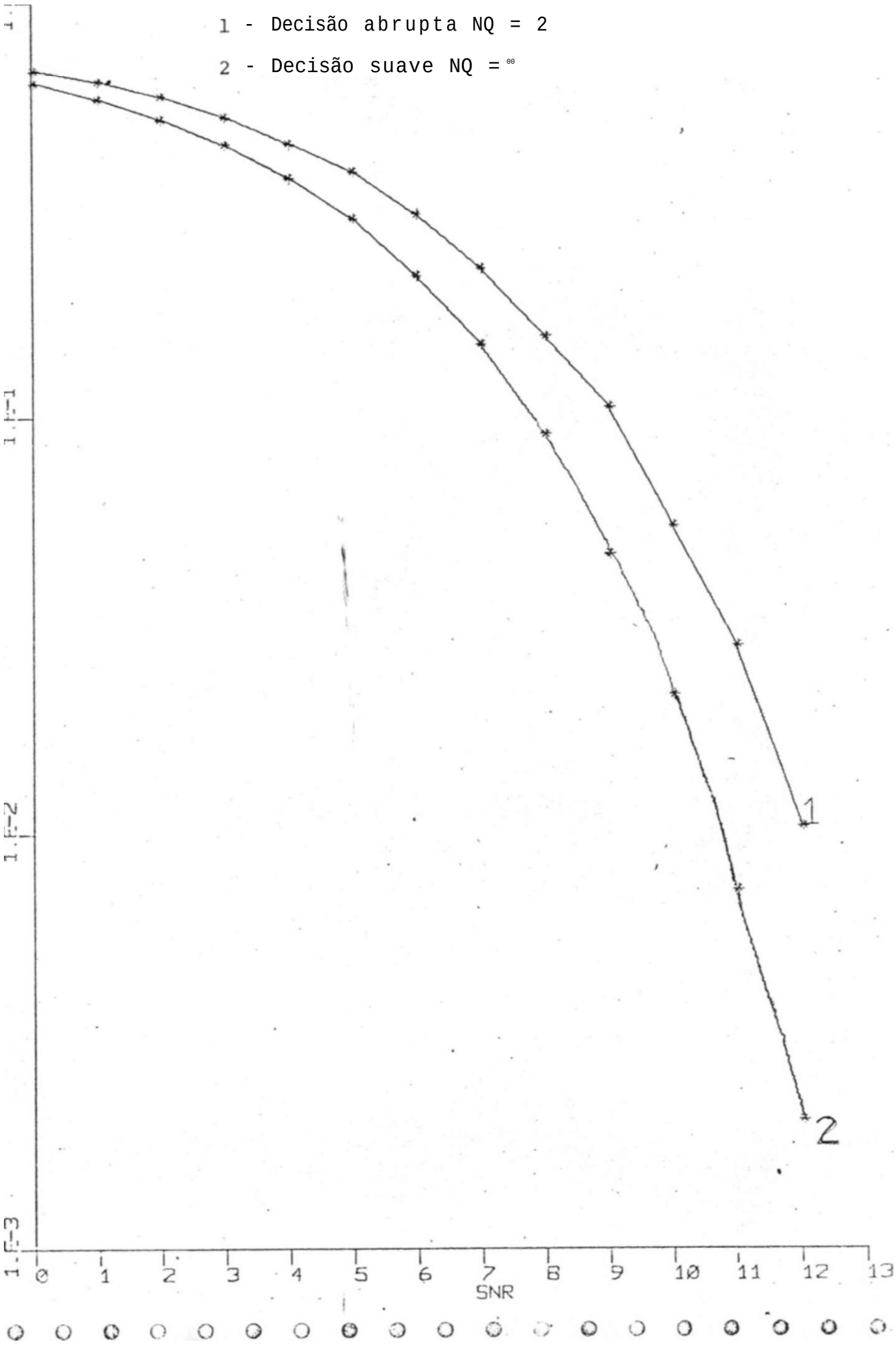
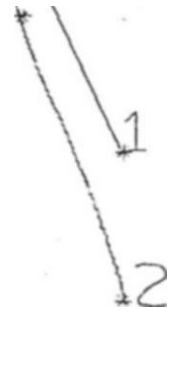


Figura 4.20 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)

Curvas PBIT x SNR (Deteção por envoltória)

1 - Decisão abrupta NQ = 2

2 - Decisão suave NQ = «



SNR

Figura 4.21 - SIMULAÇÃO PARA O CÕDIGO (7,4,3)

Curvas PBLOCO x SNR (Deteção por envoltoria)

. 1 - Decisão abrupta NQ = 2

2 - Decisão suave NQ = °°

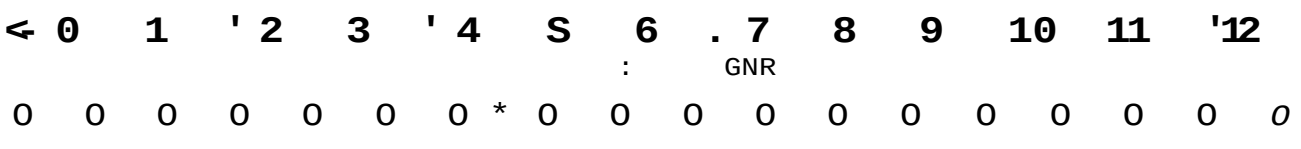


Figura 4.22 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)

Curvas PBIT x SNR

1 - NQ = 2

2 - NQ = 4

3 - NQ = 8

4 - NQ = 16

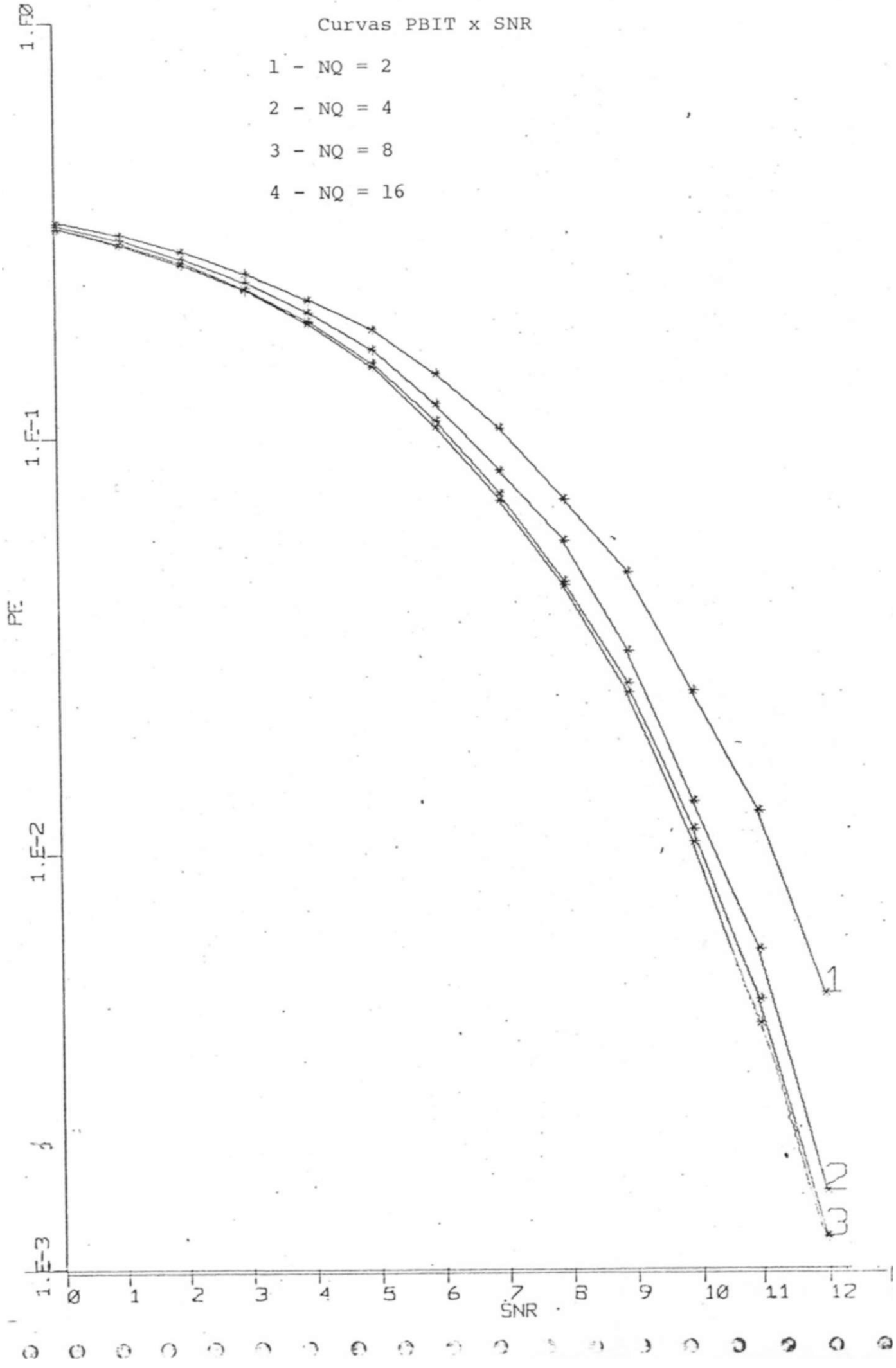
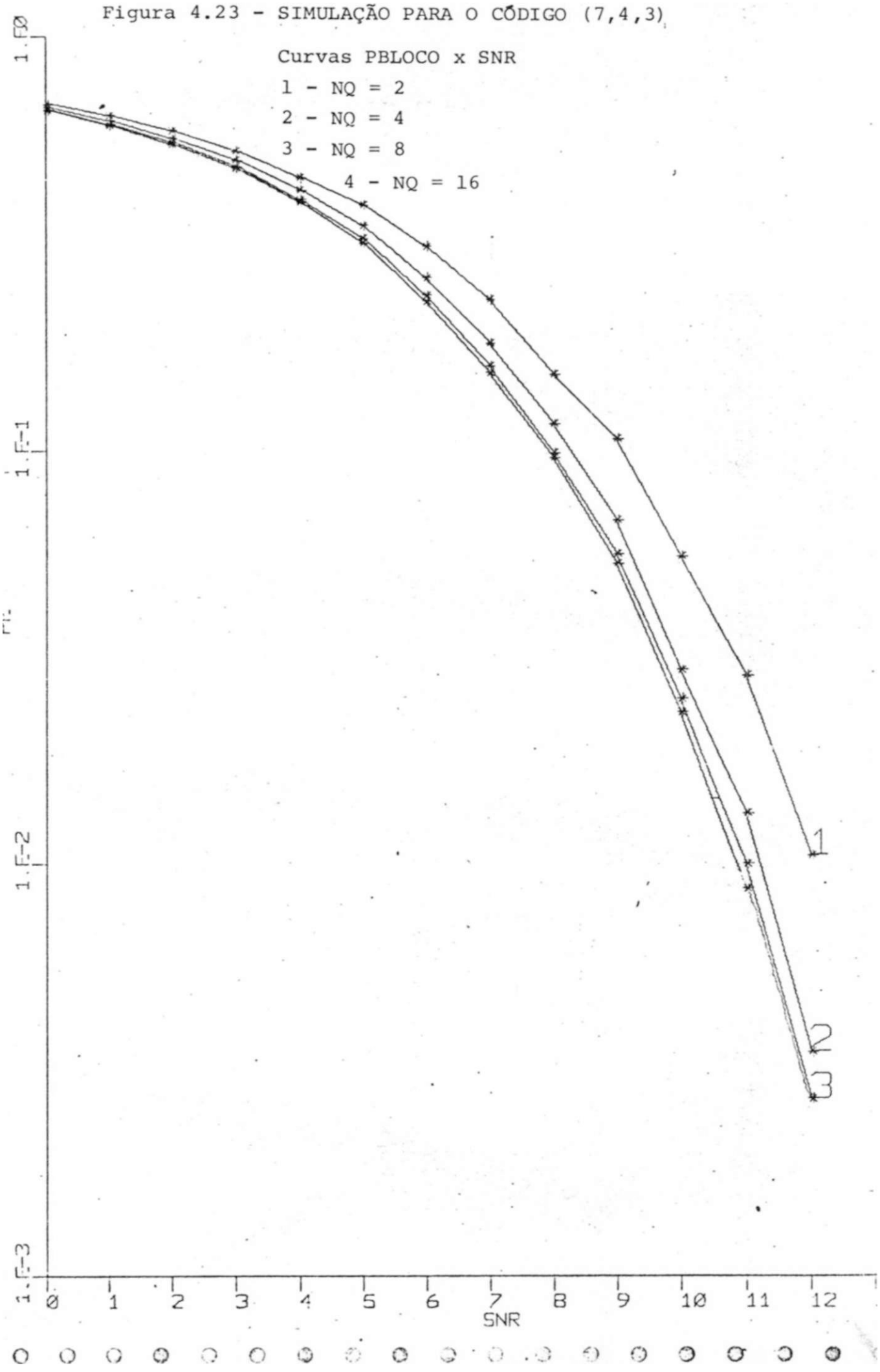


Figura 4.23 - SIMULAÇÃO PARA O CÓDIGO (7,4,3)



CAPITULO V

CONCLUSÕES

5.1 - ANALISE DOS RESULTADOS

O intuito principal deste trabalho é o de apresentar e analisar procedimentos de decodificação de códigos corretores de erros, em função da melhoria proporcionada no desempenho de sistemas codificados. Como visto, tal melhoria é obtida através do uso da informação probabilística associada as amostras recebidas, e em geral é conseguida as custas de um aumento na complexidade do sistema.

No capítulo II, após a revisão sobre códigos lineares, foram introduzidos procedimentos subótimos que empregam técnicas de decisão suave (sec.2.2), os quais proporcionam uma melhoria considerável no desempenho do sistema, como mostrado nas curvas correspondentes as figuras 2.4, 2.6 e 2.8. Eles são muito atrativos levando-se em consideração que o uso de regiões de quantização é adequado por facilitar a implementação utilizando técnicas digitais. O procedimento proposto por Wolfenson-Rocha é de grande importância, visto que é um dos poucos esquemas de decodificação que assume ser desconhecida a distribuição de probabilidade da fonte. A melhoria provida pelo uso desta técnica encontra-se analisada com detalhes na referência |7|. O desempenho do algoritmo de decodificação por dis-

tância generalizada II foi analisado através de simulação em computador, e comparado com o desempenho do sistema empregando decisão abrupta, resultando nas curvas das figuras 5.1 e 5.2 mostradas a seguir.

Para os procedimentos ótimos descritos no capítulo III, o estabelecimento da regra de decodificação permitiu a determinação do comportamento assintótico para alta relação sinal/ruído.As simulações realizadas para estes algoritmos estão apresentadas nas seções 4.1.2 e 4.2.3 e a obtenção destas curvas são de importância' fundamental neste trabalho. A partir de cerca de 12dB, a aproximação utilizando a expressão assintõtica fornece excelentes resultados, e é acima deste valor que a simulação se torna praticamente inviável. Deve ser observado que já existiram problemas na convergência da estimação da probabilidade de erro quando utilizando o código (31,26,3), isto porque a regra de decodificação descrita pela ' equação (3-2) torna-se excessivamente complexa* aumentando de modo sensível o tempo necessário para decodificar um bit. Este fato está refletido em algumas curvas da seção 4.1.2 para os maiores valores' da relação sinal/ruído.

Quando o número de níveis de quantização ê aumentado, obtém-se informação probabilística mais detalhada para ser utilizada pelo decodificador. Entretanto, um aumento no número de níveis de quantização significa um crescimento na complexidade dos circuitos detetores. Em todos os procedimentos descritos, observase que a medida que o valor de NQ aumenta, o ganho incremental na relação sinal/ruído para se obter uma dada probabilidade de erro é cada vez menor, de modo que grande parte da degradação que ocorre quando NQ = 2 pode ser superada sem que a complexidade do sistema se torne proibitiva. A informação adicional obtida por quantizar as amostras recebidas em mais que 16 níveis acrescenta muito pouco ' aquela obtida para 8 níveis.

* Compare este caso com o descrito no exemplo da página 83.

Figura 5.1 - SIMULAÇÃO PARA O CÓDIGO (7,4,3) -139-

Curvas PBIT x SNR

1 - Decisão abrupta NQ = 2

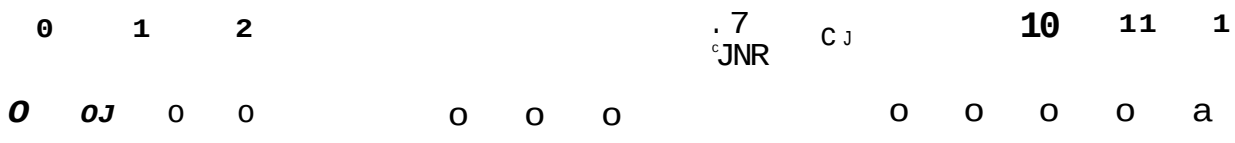
2 - Decodificação por distância generalizada II

Figura 5.2 - SIMULAÇÃO PARA O CÓDIGO (7,4,3) -140

Curvas PBLOCO x SNR

1 - Decisão abrupta NQ = 2

2 - Decodificação por distância generalizada II



As curvas citadas na página 105, usadas para verificar a simetria do ruído Gaussiano gerado são apresentadas a seguir.

5.2 - COMENTÁRIOS

Estas técnicas descritas são de particular interes se para sistemas de comunicação via satélite, em comunicação mili - tar e em comunicação espacial de um modo geral. Uma importante e atrativa aplicação está no emprego da decodificaçao probabilística* em sistemas de comunicação utilizados para transmissão de dados em canais de HF, de modo a assegurar uma boa confiabilidade â* transmis são.

Neste aspecto, ê sugerida uma continuação deste [1] trabalho, propondo-se análise e implementação dos algoritmos estuda dos, aplicados para canais de HF.

Com relação ao procedimento proposto por J.Wolf ' descrito no capítulo II, deve ser mencionado que ê um procedimento' ótimo, equivalente ao algoritmo de maximização da probabilidade a posteriori descrito no capítulo III. Contudo, a formulação matemãtji ca no ultimo caso, além de maior rigor, permite a interpretação de "correlação generalizada" e o estabelecimento de cota para a proba- bilidade de erro por bloco. O mais importante é que permite o uso de decisão suave usando NQ = 2^ regiões de quantização, o que de acordo com a simulação resulta em considerável melhoria com relação a decisão abrupta e uma simplificação enorme no decodificador. Ade- mais, comparando-se as expressões utilizadas na decodificação, vê - se que a segunda apresenta uma vantagem computacional.

No caso de códigos de bloco, fica claro que o algo rítmo de Hartmann-Rudolph pode ser implementado utilizando a treli- ça associada ao código dual. A implementação computacional da treli^

Figura 5.3 - SIMULAÇÃO PARA NQ = 4

Curvas PBIT x SNR

1 - Palavra toda nula

2 - Palavra toda um

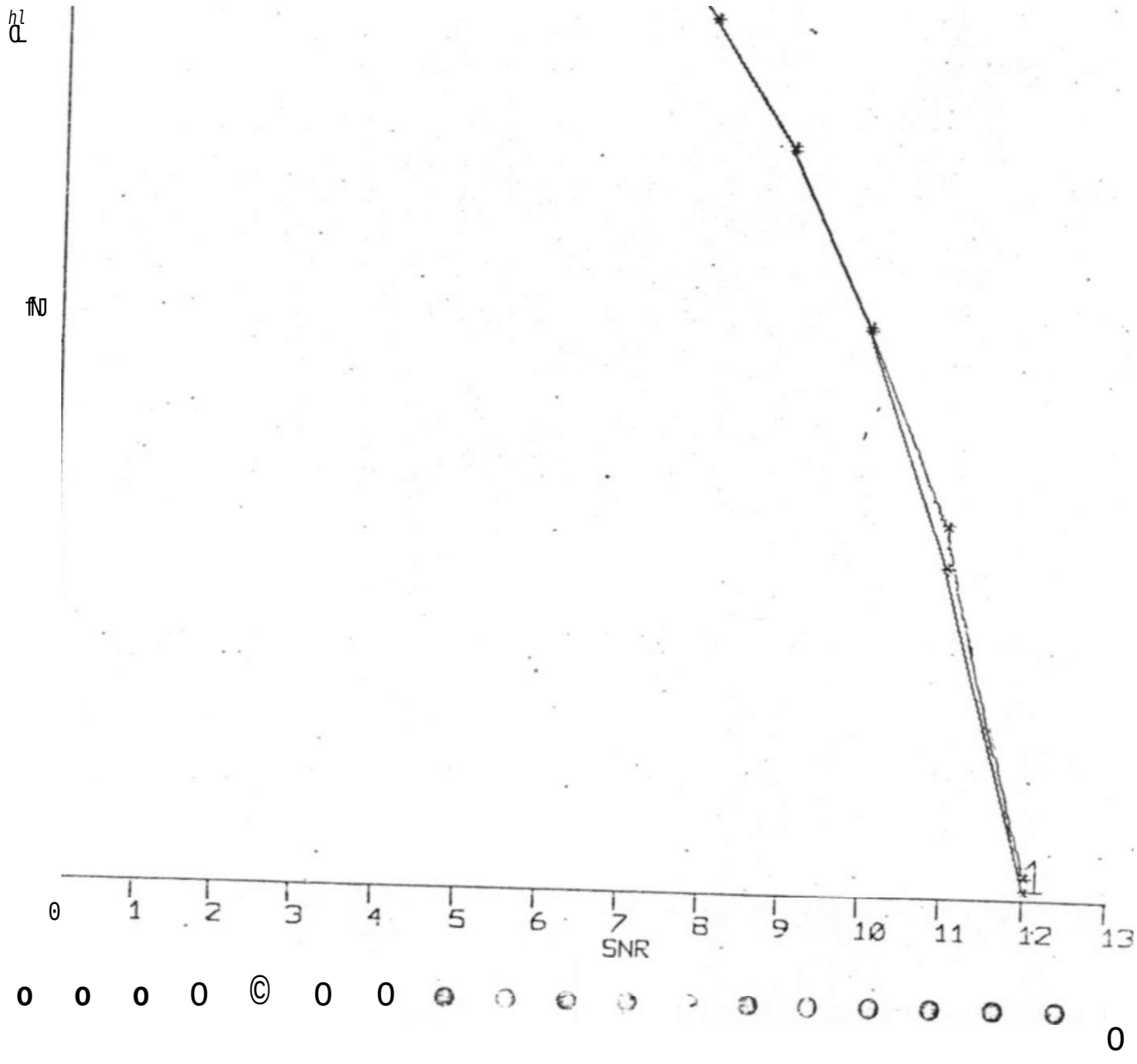
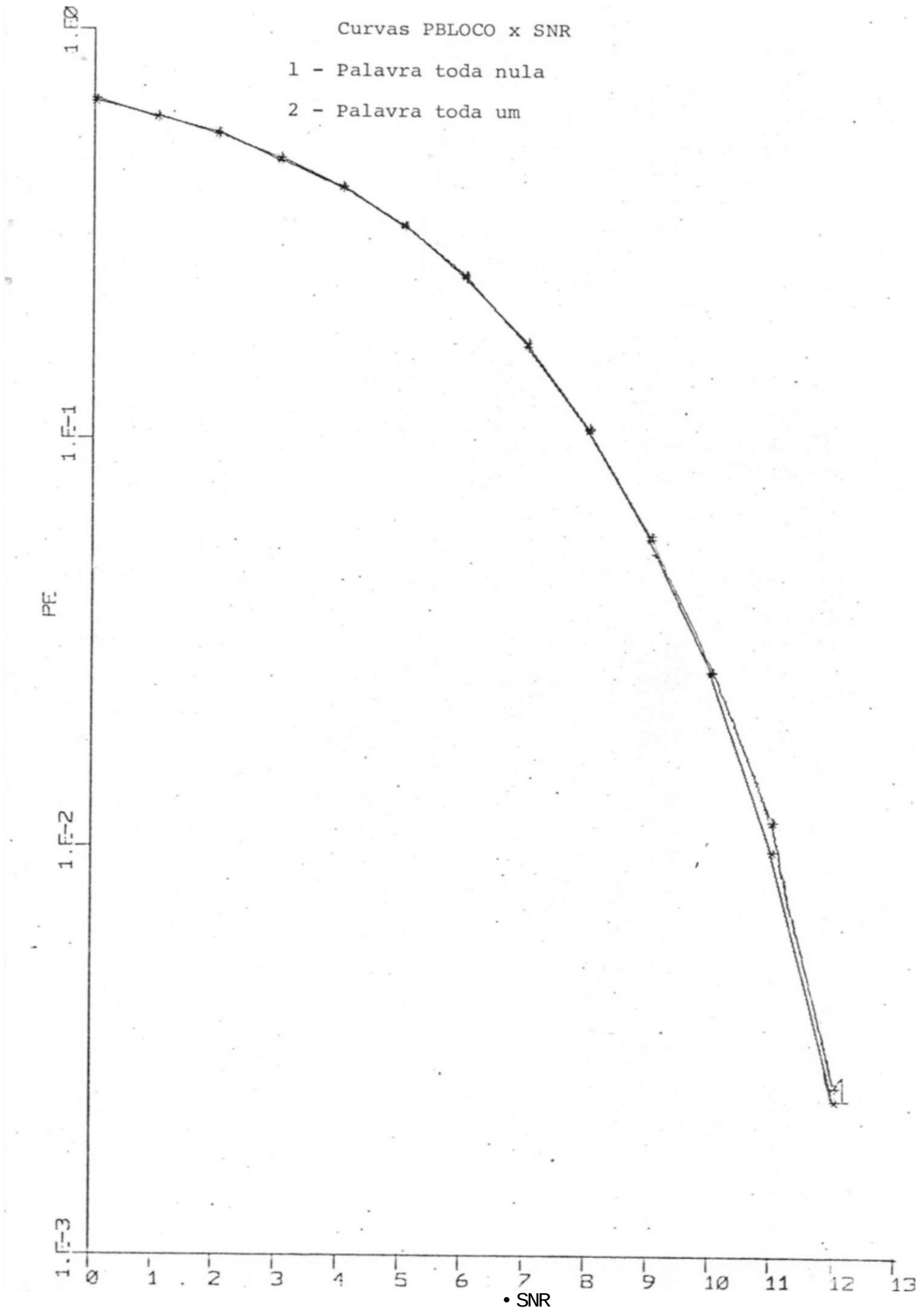


Figura 5.4- SIMULAÇÃO PARA NO = 4



ça na forma descrita na seção 4.2.2 e nas subrotinas das listagens do apêndice A, permite a simulação de diversos tipos de decodificadores de códigos de bloco.

Algumas versões dos programas estão apresentadas nas listagens do apêndice A. Entretanto, devido ao tempo de CPU re - querido, os programas não podem ser codificados para um caso geral , de modo que devem ser processadas algumas pequenas alterações para cada caso específico.

Ainda como sugestões para pesquisas futuras, de- ver-se-ia considerar o uso de microprocessadores para implementação' dos decodificadores ótimos propostos no capitulo III, permitindo in- clusive o emprego de subtreliças. Com relação ao algoritmo de decodi ficação por distância generalizada II, seria interessante tentar de- senvolver uma cota para a probabilidade de erro.

É interessante observar que a decodificação por distância generalizada I I resulta em uma palavra onde a medida dos coeficientes de confiabilidades condicionais dos dígitos é máxima Este fato encontra-se demonstrado no apêndice C.

A medida de confiabilidade associada a uma clas- se pode ser determinada mesmo no caso onde a partição do espaço de observações resulta em classes de confiabilidade não simétricas, e pode ser calculada no caso multinível. É possível conjecturar que exista uma maneira ótima de combinar os coeficientes de confiabilida de p ^ de modo a obter um algoritmo que minimize a probabilidade' de erro por símbolo (similar ao descrito no capítulo III) que possa' ser aplicável a códigos de baixa eficiência e que seja simples no caso multinível. Uma posterior extensão para decodificação de códi- gos convolucionais seria bastante ütil, pois neste caso os códigos ' utilizados geralmente são de taxa l/n.

As técnicas de decodificação apresentadas nesta' tese se mostram muito poderosas e práticas em canais que tenham com-

portamento aproximadamente semelhantes a canais sem memória. Com o custo do hardware digital diminuindo rapidamente, o uso destas técnicas crescerá largamente em aplicações práticas em situações de controle de erros.

```
C       obPh/ubS" t'J"ir'-h
C       UtCOüJFICACAU DL CUUIGUS LJ Nfc.Ahfc.iJ
C       Al»GUnÍ'l*Ú Üfc HAKTHANK h KÜÍiüLHH
        J .iF.Grih ALLi,bJJ,hUUfeU,CHr.C*„Ü,(>ftlN,G,H,*,£>K2,1,A
        ül5»KÍ,S1ÜH HüUnl>(4,4),0(i?,41J,GiJ*J,h(4/J,P1(lo)„FU(16J,K(16),KUld
      -1),XC32J
        CüKftuii U,NC,L,1,J
        DAjA ii/3240/„X/32*0/„yt.S/» ítS¹/                    ^
        t-*r(A)=1.-h-hl-CIXJ
        6U2=^«JK1(/„J
C       LNIKMUA L-L üAüüò i HLUCr CUüt
        nt<ilr.lt), 11 J
11      KUKKA ri 1H0, •t.l«'J fcK N„K„l>='„/J
        ÜFKH CUU1J=^o,í- iLt=*HAND.LM']'j
        KtAl)(M,*)M,r.C,ÜrtlN
        NsNC-fcC
        lh(xL.Gl.32.ÜK.N.G1.b)STUP
        „=2*•v
        NCPlshC+1
        [JCP)s.'«C+l
        l.l'l=l'.41
         Li1---1
        rL1=•—í
        A«1ltiS,izJ
        t ÜKMAJl1nl',¹C1Lbfc. COL-r.? VKS üh flÜ¹1
        Kt.Aü(,J3)rfc-Sir.
4   4     fcUKKAKÃiJ
        11-í1r.Mh.t.c.ir:>)íe«J 1Ú 1000
C       KuiifW p/ CUÜÍGUO CÍCLICOS iPULjLfcOfeiO Gr.HAüUKJ
C       (7,4J => 13,],,Oj = 1,0,1,1
C       llb,llj => 14,1,0J = 1,0,0,1,1
C       131,2bJ => lb,2,0J = 1,0,0,1,0
        «KlTtíb,44j
44      FUKV.ATI 1H0, • fcliJfcK G(A)=',/!
        KbAL»(M,*MGt1J,1=1,!-Fl)
        AU1=1
        ÂINCFI1=1
        üü 200 K=l,i«Ci'l
        IK(Al?\1,LU.O)GO iU 100
        HiK>sI
        CALL SMÜÜ21A,GJ
        CALL SHJfrllGl
        G«l 1U 200
100     C'MLL SHlfcJ(GJ
        CUft 11 ivUb
you     i)U 4uu 1=1,4/
        ChhCK=CMfc.LK•A(11
400     IICCHCCIK•Nfc.O)STÜH
        DU búu 1=1,*b1
        uu 400 J=l,..C
400     Dll,NCPl-Jl=nlJJ
        CALL SHIFTCtiJ
bOO     Cuul1.UK
        Gu 10 2000
C       HUH*A P/ CüUlGUS L)t fi.uCu Aii-LJCLJCüò.
1000    "f\lih Ib,bbJ
b3      Í- Uk fi'it i I 1 MO, ' h »1 C.F
        UU o00 1=1,*
```

```
bo      t UK»*A113*111,1À ) )
bOO     CUftllitUfc,

C       Ufr.KAfcüU  FALAVr<A.S  ÜU  CÜUlGlJ  UüAb.

        ÜÜ 700 1=1,.^bl

        lP!s)+l

        ÜU 700 J=IP1,íV

        CALLJ 5l'hAn
/oo     cumiNUc
        Jt'lfe*bl*3)GU  11»  oOO
        ÜU  eOO  b=/,*bl
        i  lM»L>=lfc.NÜ-  1
        1>Ü  o 00  1 = 1,ÍKNÜ
        bUOf«U(L,J)=r\+1
        JU=bÜüftDlb-1i1*1J
        Jf=bUUNülb"l,NLiJ
        ÜU  bi<y  J=Ju,JI-

        CALI.  SCKA*
tf 00   CUfci liiÜt

        i=(0-  J//!
        A=10,
        ALL=W
O       MJ-.thu  Üt  htGJuKí  üfct uUANTI ZACAÜ  =  Nu
         Uü  bO  J í,N=2,4,l     i

        hhlJKl3,)JUJ
     1  KUHMATllHl*49A,•üftJ VbkülüADL  f-fc.UC.tfAL.  üt.  Pfck»«AMBUCU*,/,t»lX,•MfcSTRAO
       -U fc> tNGfe.KHA.KlA  CL&TK1CA',/„iüA,•**  UtCUiJlKJLCACAU  DK  COUlGUS  LÍNEA
       -KbS  -  ALGÚRUKÜ  UL  HAKIWANN  b  hut-ULPh  **•»/,bX,•KUs•„14)
        MOlsNO-l
Cl      VAKlANUÜ  SIGUMb-lU-NUJSt  KAJ1U.
        ÜU bo SNk=l1.,\i.,1.
        NI1=o

C       K=l  (Lute  NUISL)
        t»=Z*  l i-ul  )/ÍNÜ*SO'd )
        Fl 11 J=0. b»t.Kh C lb )
        b=Z„/  l i^*í>^iJ 1
        Fuilj-ü.bUl.ítKHh))
C       K>    Ni»-1    IriJGH  NUlSfcJ
        PIlhw)=FUl1J
        PUlMJjsPl(1 )
 C       KK<*lr-J
        lPlf«Ü»£U„2JGU  JU  3U
        ÜU  30  K=2„N01
        bls£+lf«0*K+l)/lwU*6U2J
        B2=Z* i s*g*K J / i NQ*&U2 J
        Pl(K)=Ü.5*ltKF(Bl)-t:f<fe (62>)'
        PÜÍNO-IW 1)=P1 IM
     30  CUNUNÜL
C       CALCOLU  ÜÜS  KU'Ò
        ÜU  bO  1=1,NU
        K(l)sci„-Fl(l J/PfHl) J/C1.4P111 J/PUll))
     «O CUMINUL
```

```
      JBh=u.
      íHt=u.
C«
      bll=(10**4)»NC
      ò JIJ-U.
C     51«ÜLACAU  ÜU  ALGÜK1TMU
      Du 4u KD1G=1 , BX1>NC
      M2 = U
      Dü  lu  ]=1,NC
C     GKKAivUü  KUiüÜ  GAUSS1AMJ.
   1b  Vl=/*«AlvlbUJ-l

      S=V 1 »Vl-frV2*
      lt-  (Ò.Gt.l)  bü  1U  lb
      V=vl*Sünl     ALÜGCSI/s;
      K=S1G«A*V
C     FÜSSlblLlDrtutò:
      li-  U.GE.A/2.}ft*2 = Nlí^+l
C     IfrU.LE.-A/2.)NÍ2=N12*1
c     INJEKVALO ONDE CAIU A AMÜSXKA =NK
      NX=(AbL*A-»yj*NÜ/AtI
C     rKOtí  y  DlvlSitíLE  aí  A/NU.  IS  ZEkÜ
      IKUi .LT.l J.\i=l
      lt-(M,b'J,.NUl JNYsftÜ
      KÜll)=KlN)fJ
   lü CÜNJji.ut.
      f**1=M1+ui2
      lb  lNl2.LE.Xj   Gü  TU  40
      DU Jb 1=1,MC
      SP = Ü.
      DO 2b J=l,w
      HK0D=1          .                                    ^
      DO 20 L=1,WC
      DELTIL=1
      IVt J.NE.L)DELI1L=U.
      IFIDIJ,L).NE.DELTIl.)PROD=PkOD*kU(L)
   20 CONTINUE
   2b SP=PkOD*SP
C     DECISÃO : SP>0. ESTIMAR C~=0, SP<0. ES11MAK C~=l
C     POSSIBILIDADES:
      lFlSP.Lfc.O)TBE=TbF*J
C     ÍF(SP.GT.O)fbt=lBE"fl
   3b COI.TINUF
       IF(TbE.GT.STO)TPE=TPfc+J
      STO='JBE
   40 CONTINUE
C     CALCULANDO PKÜHAblí.)DADtS DF EkkO.
      PEB=TòE/BlX
      PEP=JPMrJC/ftJ1
C     SAÍDA DOS DMDOS
      HklTE(3,2)SrJK,TBE,PEB,TPE,kr.P,r.>1
    2 FORMAI(1HO,•SKB=',L14.b,4X,'lbt=",E14.8,4X,'PEB=',El4.tí,4X,•X
     -PE*',£14.fc,4X, "PEPs* ,fcl4.h,/', •0*,37X,'NYl=*,161
   bO CONTIMüfc
      STOP
      END
```

```
SlHRGTlfcA P/ CfclirUlifi l>E Éfcf - AFKfiX. POLINOhTAL
Fi í«ÇAO DISTRIBUICAO KQXV.hh DE PROf AMMDADE
ÜnP.Ü /ECX)/ > 7.5 E-08
i«ê.£ã - DEFMhTAl-t.siíi r-'i- GtwHAklA tLLTKGfc ICA. £ SI STF.;-AS
FÜÍ.CTIOA LHrr(7)
X*sAdS(SôRT(2.)*Z)
;= 5CRT(l/(2*3.141593))
r= ,23Í&419
òl= .3193bl5
b2--.3b65b3B
63=1,78147«
p4=-l.821256
Ébsl,3302740

1=1/(1+P*X)
ERFC= 0*EXP(f-X**2)/2)
£,ãFC = ERPC*T*(RlÍT*(fc24T*(B3+T*(&4+l*65))))
Ei<FC«2.*EftFC
R TLiHi-
r.í.D
SU6R0UTINE SV.0D2 (X.G)
I%TEGER X(32),G(32)
LO 1 1=1,32
1 XCI )=IABS(X(I)-G(I))
P£TU££
Li,ü
SüèROüTIivE SKIFT(G)
11 '7cIGER GC3>"),sTnl,STü2
STClsÇCl)
Lü 1 1=3,31
Siu2=G(I+l)
G(I-H)=STU1 I
£TÜ1=ST02
1 CüãXltfUE
G(1)=STÜ1
RcTUKr.
EjjD
SüBKfjüTIlvE
SüBRDUTINE SCRAH
IUTEGEP. D
DX^EtíSlUM 0(32.31 )
ŒO:'f.Üí- D,NC,K,J»J
DO 1 L=1,;<C
3 D(K,L)=ÀbS(Df1.L)-D(J,L))
RETUPK
```

NÚCLEO DE PROCESSAMENTO DE DADOS DA UNIVERSIDADE FEDERAL DE PERNAMBUCO-

```
C       üFPE/DES  -   Cnry  :F
C       OcCüDlFlCACAO  DE  CÓDIGOS  LIKEARES
C       A..Gú?lT.-iO  DL  DFCODIr 1 CACAü  ROR  HAXIHA  PROb ABlLI DADL  A  POSTERIORI
         CO ri- OU  GF .ri.tfPS.J
         1M11óV.H  C(7l,r.f .GFQ,hAT(7) ,H(3,7)
         ül.-»SIÜ.J  I-ÍAJ RI x(3,7,0:J),MATRIZ(8,7,0:1),pETRIX(R,7,0:1)
         DIriEf.SIÜt, Kf7),STAR(7),P1(1r>),PO(lt>),ALOGFj(16)
         cRr  U) = l.-LRFC(X)
         ÓQ2=Ò0KT(2.1
         NQsSo;NQlsftO~1                                                *^
         GFf 1
         GFO=GF+1
         p,« 7; ri=4
         J=:.-!N
         í«P5=GF0**J
         *RIT£(5,33)N.K ,GFQ
         vRIT£(b,11)
         ü?cf'(UM7=20,FlT.E.='H.CAT')
         DO 1 JJ=i.J
         K.EADC20,*) CH(Jü.iVW),fvNrl , N)
1        AR1.T£(5«10)fHfüJ,  ),NN=1|N)
         CALL THFI.Ll fh.í-ATRlX, VATKIZ)
         CALL EXPUR(KATRTX,MATRIZ)
         i.h1Ts.Í5,2?)
         DO 2 K1=J.UpS
2        .vRÍT£(S,10üH f::aTRJX(Kl,K2,KJ),K3=0,GF),K2=1,N)
         DC b  S'ÍK=0.,l2.,lj

         jT£ = 0.
C
          Z=10.**CSfoR/?ú.)
          Ps7*?í01/(NO*S02)
          PI(i)=0.5*RPrC(B)
          BsZ/(NO*SÜ?)
          PO(l)s0.5*fl.+FRF(e))
          Pl(NO)spo(1)
          Pü(ivO)=Pl(1)
          lf(i\O.L£.2')STüP
          DD 17 IREG=2.NQl
          61=Z*(NQ-IRÊG*1 )/(NQ*S02)
          6z=Z*(NO-I$£G)/(HÔ*S02)
          PI fIREG)sO»5*(FRF(&1)-ERF(B2) )
          Pu(.\Q-IKEG+l )spl MRc.G)
17        CUI-'TInU£
          Dü 231 IPtG=1,N.0
231       ALPGFI (IREG)sAl.OG(Pl (IREG)/Pü(IREG))
C
          UO 4 Kü-'•=1.N'*ORDS
          CALL >;0Rr;5(MATRTX,C)  •
          XS:vR=S..R
          CALL  NUl£>E(XS»3R,J> ,K,C,STAP)
          L>0 24 *X=!,M
24        STMPÍKA)=ALnGFl( lFJX(S'iAR(KX)))
          CALI  CüRRELf•ATR1X,PcTRIX,STAR)
          CALL SURVlVfraXRIX,P^ThlA,hAT)
          IERRüRsO
          ÚC  3  KX=1.X
3         IPr.H}R=lLRrrWlArS(C(KX)-HAT(KX))
          T&£s1nt+IERPÜP
```

```
      IF(TLRKOK.NF.O)TP^STPE-* 1
      CO  .Tl?üE
      PEô=lBt/( M* rjv.rm pg)
      pEr=TPt/üVORDS
      . F.lTt.(5,<iOS1.W,Pr.fc ,Pr P
5     Ci-.TinuE
C
10          AT(IH   RX  10(11#1X))
ÍCH   t Gr : ÁT(1 HO        • . ' ,2X,1ü( ' . [  ' ,11 . ' , l , I 1 , ' J ' ,2X))
11    F ORí ATdriO  BX.'MATKIZ DE VtRlFlCACAO DL FARlüADE H'• )
22    r On.v AT(1n0 //. bX , ' t> A'í RJZ AíjSUCIADA A 1 HEL1 CA ****• )
33    rfin: AT(lttl 44X; • UFIVbRSlDADF FEDERAL üt PERNAMBUCO' #/» 47X #' MESTRA
                    ELÉTRICA».//,30X,•** DECCDIFICACAü DE CÓDIGOS LINE
      AtiES - ÀLGOKITMO uF. ROCHA JH **'.///,9X#'CÓDIGO D£ DLÜCO C',I2,','
      .12, l )',3x.»80rRF  GFC',li, ' )' ,///)
44    r l"'h' ATCJH0.5X. 'S.vH=' ,F4.1 , 5X, ' PF.b=' .LH.fc.bX, ' PhlP-• , Eí4.8)
      STÜP
```

```
C       SUfc»küTlf»A  H/  CALCULO [lfc. tHI  -  AHkfiX.  PULJMMJAL                  "
C       KU;«CAU  DJMKjnJlCAO  NüktiAL  L'h  PKUbABlLJ D.AlJh
C       frKl  /r(Aj/  >  7.b  K-UÓ
C       üttà  -  OfcPAklAwfchliJ  ruGbMiAklA  KLtTküí»ICA  b  MSlL-.Ao
        HUNCTJUÍÍ  EftFC(Z)
        A=AbS(SQK'J  (2*  )*Z)
        y=  S^kll 1/(2*3*141593 )J

        Bl=  »3193«15
        B2=-.35bbb3ò
        03=).70147«
        B4=-l.821256
        ob=l.Í30274U
C
        í=1/(1+P*X)
        EKFCs  0*tXPl(-A**2)/2)
        r.hr C=  tkFC*T»(BI41*(H2*T*(B3*!•(b4«r3•«b)>1)
        thFC=2.*bhfr C
        Kb1ÜK
        bNO
CtllKEbLli2NOCALiiTkAfts»ü>41kAkOB#bkXPUki6«kUPüSf7Ai012>E*dCOKitfc,bft9&Uf(Vl V
C
CHI     GERANDO  /*  IhtLiCA  MAÜ-EXPUHGADA
C
        SübHüUTlhib  íkbLbJ lH,*AlklX,*SA1k1Z)
        CU*- r [i,N  Gè,'*,NPÔ,J
        lwibGfcK  ML.*A,Gr,ri(J,A),KATfCJA(fcPS,  ,Ü:Gr),MATfcl>.(NPS,hj,0:GFJ
C       GbkAftDü  A  IKfc.LblS  1  IKfr  NATklXJ
        CALL MICAL(1»J,1*-'ATk1A,MJ


        DO  1  KX=2,•,l                                                    '


        bi)  1  NAT=],Í;FS                                                 >
        "'NÍ l-ɨ AI
        1SU'^=0
        DU  13  ALr;v=ü,Gr
13      15UyslSUH4SA.T»*]X{* \A'J,(•)',.°.Lr AJ
        lbdSüy.KO.ÜlGU  Tu  )
        C*LL  ISfÜCAL(1#  fckX,  AT,«AlkJX,H)
1       CLMJINUb
C       GLfcAfcfrO  A  IKLLL16  2  (fc*  rvATKlZJ
        CALL  NÜCAL12,N,1,MATRIZ,H)
        L»U  i  KX~N-1,1,-1
        KKA2KA
        kY=KX41
        DO  2  r.Ml=l,r»PS
        KXAT-NA1
        DU  ALFA=Ü,Gr
        1FlHA1hlZ(«ÜAT,ki,ALI-A).r.u•U1(*Ü lu 2
        CALL >«üCAL(2*r:kA,»ATtf1Zl*í*AT#ki,ALrA),HA1klZ,h)
2       CüKJlnÜb
        ht. IL.hr.
C
C<2     CALCULAkDü  US  .OS  -  PktèüCHlKhitlU  UÁ-  "ATrilZ  DA  TKÊLJCA
C
        2>UB£OÜ11 r. NüCÂl(il,M,Aj,^AJk,h)
        Cü'-."ü,\ Gr,vi,l>Pò,J
        1u11Gbh M1,ALFA,r.I(b),Cll5),Gr,GFO,r.ATKÍWPS,N,Ü:GF),H(J,u)
        151G=1-1J(  T-J)
```

```
      &FU=GF-»J
      ÇALL rkAï.ObtGFW,J,AI ,hI )
      DU i Mbt* M=U„G*
      í»Ú 1 J l=1,J
      LL=hï(Jl)•15JG*ALFA*HIJJ,KA)
      CI (Jl )=rUD( LL,GF U)
      lF/CTCJJ ).L1 .OJCKJJ )=CT(J1 MGift
1     CONTINUE
      CALb TftANBDlGFO*J,NO,CI )
      JF (&T.tu.1)MATH(NU,KA,Al.f-A)=A'J                    ^
      IF (NT˳EO«2 )MATK( Al˳fcX, ALP A )=i»ü
2     CÜMlNUfc
      frei URN
      F.* D
C
C t J 1nANSFUh ACA»» Hft
C
      SUBROUTlNE lh Awou(Grft,J,OU,oft)
      li.TcGFF DÜ,bft(b),GFft
      OU-0
      DO I KSl,J
      i)D=l)D4bß( J<t 1 »N J *GF'j* * I M-i )
l     CONTINUE
      UU=UD<1
      f.r.IUH
      E.ND
C
C*1   TRANSFORMACAU DP

   1  SubROUlJNL i H AH í/H ( Gfft , J / i'U , no )
      1 Mt.Gfc.rt DD,ofe(bJ ,GFU,ft
      0 = l>ü-l
      ÜÚ 1 *1=1,J                                           ^
      Bb( J+l -«' } = ~Ul>(u,GF 0)
      U=Q/GFQ
1     CONTINUE
      RETURN
      c U0
C
C«S   GEMANDO A IRfcLJLCA EXPURGADA
C
      SUBROUTINE bAPUk(VA1hJX,KA1H1ZJ
      Cü.Mvu.\ GF ,'.,'«^,J
      li<lEGEN ALFA,Gr , •-A1RJA(KPS,hi,O:Gr),*iAIR1Z(NPS»N*0:GfJ
      DU 1 NA'JSJ ,NP2>
      Í)U 1 KX=1,H
      UÙ 1 ALf* = 0,GF
      1FC *ATKJAtKA1,rA,ALFA) .i.t .NAÏKJ2,(NAT,KA,ALKA))
   :  f Ain)A(aAT,KA,ALF *)=0
      «AÏRiZOAX,* A,ALFA)=0
1     CONflNUE
      KFJUrN
      RNU
C
C»t   GERANDO PALAVKA'S-CUDIGÜ ALEAIUPJAGENTE
C
      SUhROUl1-.c. AUHI-5( -ATHJX,C)
      Cü*Ü.. GF ,;> ,r»PS,J
      1u1EGEN AurA˛C(w),GF ,KAINIAluPö, f*, 0 :Gr )
      Gn,=FLÜA1CGF*1)
```

```
          M>=1
          UJ  1  KX=N,J
10        ALrA=f-Anlbl'bHJ
          IFlALtA.Ku.Gf,j JòluP
          1* l*AlhlAí NU,KX,ALFA).KO.OJtílí  J ü  10
          ClK x)=ALFA
          rç,Ü=rtA1hlAlMJ,KA,ALI¹ 1
1         CÜ«)1NUK


C
C»7       SI-MILMMJU CAIAL- H.IDUSO - Fi*CüMhAf.uü VtTUH KKCtHJLDU
C
          SMHKUUTJrife ftüiSK lSwR,-v,»',C,SIÀ>HJ
          luJtGrK Clf«J
          ¡»y=lo;.\ul=Ny-i
          -=io.
          Z=lO.•*(SvH/z-O.J
          SI GrA=A/Z
          v AHsSJ6KA**2«
          üíi  2  1=1,^
1         X=2.*KAN(bOl-l.
          *=?.*RAN(34
          S=A* A-i** »
          1FíS.GK.1.IGU  IÜ  1
          v=b1G*A*6OK1(-Z.*ri,OG(S)/S1
          À = » V
          ¡=í*v                                                    k
          h(I)=C(1)•A4A                                            ,
          Nl=h(ll*.v.J/Af1
          lf (.'.*. LI .11,1 = 1
          1i UJf.Gl . 101 l-i)=vü
          STAk(l} = Ni
2         CONTINUE
          KETUKN
          tNÜ
C
CtU       OKIKP.M1HA>üü A CÜKKKLACAU G£-\KhA L JZAüA
C
          SUüROUTJME   CUKKfcL(*ATRIX,PtTkIX,Vfc.CluKl
          COP.aOft'   Gt,N,upS,J
          1N1K G Kh   ALFA,Gf
          Dl«£NSlÜh  "Al kl A(r. PS,'<, O:Gh 1,KKlKl A( *PS, ',o:GFl, V£C'lUk(ft)
          KPSL0N=1,fc>3b
          JJ = 0
          üü b J2=l,l.
          óü  b  Jl=l,JPS
          00  2  03=0,Gi-
          ll-1*A1R1X(J1,J2,J3).KO.01GÜ Ju 2
          lt lJ2«»f .1 )tiü  i Ü  1
          PtjhlA(Jl,Ji,03)=0 i*tfFCTOR(02}+ fe.P&LUN
          Gü  l ü  2
1         L»u  10  ALFA=0,Gt
          1F (PE1 Kl At kA i » 1 X (Jl, JZ, J3), J/-1, <*Lr A 1 .  1 . 0.0 ) 0O=ALF A
10        COf.ljNÜt
          PKTKlAÍ Jl,Jz,J3l=Pfcl«lX(*AiRlA(Jl,J2,J3J,J2-1,J J l
      :     4 J 3» vr.Cl UK(J2 )+r PSL0-\
^         CONTINUE
          P*AA=PfcTh1  A  tJl,J2,01
```

APÊNDICE B

Seja um código de bloco linear binário $C(n,k,d)$ com $m = 2^k$ palavras códigos, denotadas $c^{\wedge}\ i = 0,1,2,\ldots,m-1$. As palavras são transmitidas pela fonte equiprovavelmente através de um canal perturbado com ruído aditivo branco gaussiano com densidade espectral de potência unilateral $N_0/2$. Então

$$P_{Bloco} = P \qquad \text{(B-1)}$$

Como é considerado que a énupla transmitida foi $*$ C , a palavra toda zero, a decodificação por máximo de verossimilhança será correta se e somente se

$$\sum_{l=0}^{n-1} c_l p \; \pounds n \; (J)_{m} \leq 0 \text{ para cada } i = 1, 2, 3, \ldots, m-1 \quad \text{(B-2)}$$

Se os sinais são transmitidos através de pulsos liga-desliga com amplitude $+/\ddot{E}$ Volts ou 0 Volts, então

$$\sum_{l=0}^{n-1} c_l p \; (r_l - /EV2) < 0 \quad V\ c.\ e\ C,\ c.\ ^{\wedge}\ c\ . \qquad \text{(B-3)}$$

Definindo eventos $A^{\wedge}$, para $i = 1,2,\ldots,m-1$,

$$\text{(B-4)}$$

tem-se que (B-1) pode ser reescrita como (B-5) usando (B-2),

$$P_{Bloco} = 1 - P^{m-1} \qquad \text{(B-5)}$$

Por outro lado, $W(c_i) = \sum_{l=0}^{n-1} l\, c_l$ é o peso da palavra c_i e $\gamma = R\gamma_b = E/N_o$ é a relação sinal/ruído do canal.

Sejam variáveis aleatórias $\{N_i\}_1^{n-1}$, com distribuição normal

$$N_i = \frac{}{\sqrt{E\, c_i, N_o/2}} \Rightarrow N_i \sim K(Q,D \qquad \text{(B-6)}$$

os eventos A_i podem ser agora expressos sob a forma

$$A_i = \{N_i < \sqrt{R\gamma_b}\, W(c_i)/2\} \quad i = 1,2,\ldots,m-1 \qquad \text{(B-7)}$$

As variáveis aleatórias $\{N_i\}_1^{m-1}$ são conjuntamente gaussianas, com matriz de correlação (χ_{ij}), onde $\chi_{ij} = E\{N_i N_j\}$, portanto a correlação χ_{ij} será

$$\chi_{ij} = \frac{\sum_{l=0}^{n-1} c_{il}\, c_{jl}}{\sqrt{W(c_i)\, W(c_j)}} \qquad \text{(B-8)}$$

Usando o fato que $\{n_l\}_0^{n-1}$ são variáveis aleatórias com variância $n_0 = N_0/2$, tem-se que

$$\chi_{ij} = \frac{W(c_i) - 2W(c_i, c_j)}{\sqrt{W(c_i)\, W(c_j)}} \qquad \text{(B-9)}$$

Notando que $\sqrt{E_{c_i}} = \sqrt{W(c_i)}$, segue-se que

$$X_{13} = \frac{c_1 \cdot c_3}{\|c_1\| \cdot \|c_3\|} = \cos c_1, c_3 \qquad \text{(B-LQ)}$$

Deve ser observado que $Q < \quad < 1$, e $\wedge = 0$ $\Rightarrow c_i \perp c_n$, ademais $X_{..} = 1$.

Desta maneira (B-5) pode ser reescrita como sendo

$$\text{Bloco} \quad N_1 N_2 \ldots N_{m-1}, \quad b \quad -1 \qquad b \quad -m-1$$

é a função distribuição correspondente a $m-1$ variáveis aleatórias conjuntamente Gaussianas com matriz de covariância $X = (\wedge - jj)^*$

No cálculo da probabilidade de erro por bloco para o código (15,11,3), por exemplo, é necessário a avaliação da função distribuição conjunta de 2047 variáveis aleatórias dependentes.

APÊNDICE C

LEMA: O algoritmo de decodificação por distância mínima generalizada escolhe a palavra código que maximiza a confiabilidade media dos sim bolos.

PROVA: A partir das amostras recebidas do canal, dois vetores são de terminados,

$r = (r_1, \ldots, r_n)$ palavra recebida

$R^{(-)} = (R^{(1)}, \ldots, P^{(n)})$ vetor de confiabilidades

Aqui § o coeficiente de confiança condicional[1] de Kiefer associado à decisão r_i.

Se uma palavra código f é assumida com tendo sido ' transmitida, a confiabilidade associada aos seus dígitos é expressa' por

$$1\text{-}R'^{1}\text{»} \quad \text{se} \quad d_H(r_i, f_i) = 1$$

Deste modo, a confiabilidade média dos símbolos da pa- lavra, admitindo que a palavra transmitida foi f, é

$$R_S^{(i)} = \sum_{i=1}^{n} d_H(r_i, f_i)\{1 - R^{(i)}\} + \{1 - d_H(r_i, f_i)\}R^{(i)}$$

Um algoritmo de decodificação que maximize a confiabi- lidade média das decisões deve escolher uma palavra f de acordo com

$$\text{fcC } i = 1^{S} \quad i = 1^{H} \quad R_c^{(U)} - \sum_{i=1}^{n} d_u(r_i, f_i)\{2R_c^{(i)} - 1\}$$

Lembrando que $p^{"^} = 2R~ - l_f$ obtém-se

$$\text{Max} - \sum_{i=1}^{n} d(r.,f.)p^{(i)}$$
$$f\,e\,c$$

o que equivale a $\text{Min} \sum_{i=1}^{n} d_u(r.,f.)p^{(i)}$
$f\,e\,c$

A seguir são apresentados dois exemplos ilustrativos.

A fonte, o canal e código são admitidos os mesmos do exemplo da pãgi^ na 73.

Exemplo 1 - Admitindo que as amostras foram(-0 . 7,-0.2,-0.3,+0.7,+1.1) tem-se que

r= (0,0,0,1,1) e Rg^{1} =(.80,.60,.65,.80,.90).

Daí pode ser encontrado p/ = (.6,.2,.3,.6,.8).

A palavra recebida r não é uma palavra código. Com relação a duas palavras código $f^{=}$ (1,0,0,1,1) e *fj* ~ (0/1/1/1,1), como exemplo, qual delas seria preferível? Aplicando-se o algoritmo, a palavra f^ seria escolhida:

$$A(r,^) = p^{(1)} = .6$$

$$A(r,f_2) = p^{(2)} + p^{(3)} = .2+.3 = .5$$

Com relação aos coeficientes de confiança condicio - nal

$-1^{=>R}l^{}$ = C-^{20}/-^{60}"^{65}'-^{80}'-^{90}) Média R }^{1} ^ 3.15/5 = .63

f_2 => R^ $^{)}$ = (.80,.40,. 35,.80,.90) Media R£$^{f)}$= 3.25/5 = .65

Exemplo 2 - Considerando um segundo caso onde as amostras recebidas' são (-0.7,-0.3,-0.3,+0.7,+1.1}, tem-se

r = (0,0,0,1,1) e $Rg^{}$ = C.80,.65,.65,.80,.90) ,

portanto p^ ^ = (.6,.3,.3,.6,.8)

$A(r, f) = p^{(1)} = .6$ não há preferência

$A(r, f_2) = p^{(2)} + p^{(3)} = .3 + .3 = .6$

Com relação aos coeficientes de Kiefer,

-1 Média $R|^{i)} = 3.2/5 = .64$

-2 Média $R^{\wedge} = 3.2/5 = .64$

É realmente indiferente para o valor da confiabilida de media dos símbolos se o algoritmo escolhe f- ou f^ como sendo a palavra decodificada.

BIBLIOGRAFIA E REFERÊNCIAS

l| - ALMEIDA, M.A.M., "Decodificação de Códigos Convolucionais", , Tese de Mestrado, Universidade Federal de Pernambuco. (Em preparação).

2| - BARTLE, R.G., "The Elements of Real Analysis", John Wiley & Sons, Inc., 1964.

3| - BERLEKAMP, E.R., "Algebraic Coding Theory", McGraw-Hill Book Company, New York, 1968.

4| - BLOOM, F.J., CHANG, S.S.L., HARRIS, B., HAMPTSCHEIN, A. and MORGAM, K.C., "Improvement of Binary Transmission by Null Zone Reception", IRE procs., vol.45, 1957.

5| -^CAMPELLO DE SOUZA, R.M., "Decodificação Probabilística de CÓdigos Lineares", Tese de Mestrado, Universidade Federal de Pernambuco, 1979.

6| - CARLSON, A.B., "An Introduction to Signal and Noise in Electrical Communication", McGraw-Hill Kogakusha, ltda., 2nd edition , 1975.

7| - CASANOVA, F.A., "Utilização de uma Medida de Conclusividade no Processo de Decodificação: Coeficientes de Confiança. Estudo ' Comparativo", Trabalho de Graduação, Departamento de Eletrônica e Sistemas, Universidade Federal de Pernambuco, 1982.

8| - CHERNOFF, H., "A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on a Sum of Observations", Ann.Math".Stat. , Vol.23, pp.493-507, dec.1952.

9-| - DAVENPORT JR., W.B., "Random Processes - An Introduction for Applied Scientists and Engineers", McGraw-Hill Book Company , 197Q.

|10| - DAVENPORT JR.,W.B., ROOT,W.L., "An Introduction to the Theory of Random Singnals and Noise", McGraw-Hill Book Company,1958.

111 I - FÁRRELL,P.G., "Lectures Notes on Communication Theory", COME NE, Universidade Federal de Pernambuco, junho 1977.

112 I - FARRELL,P.G., MUNDAY,E., and KALLIGEROS,K., "Digital Communication Using Soft Decision Detection Techniques", AGARD Symp. on Dig. Comms in Avionics, Munich, pp.141/9 , June 1978.

113 I - FERGUSON,T.S., "Mathematical Statistics", Academic Press, , 1967.

114 I - FORNEY JR., G.D., "Generalized Minimum Distance Decoding" , IEEE Trans. Inform. Theory, vol. IT-12, pp. 125-131, April 1966.

115 I - GALLAGER,R.G., "Information Theory and Reliable Communication", John Wiley & Sons., Inc., 1968.

116 I - HAMMING,R.W., "Error Detecting and Error Correcting Codes" , Bell Systemes Tech.J., 29,. pp. 147-160, April 1950.

11V I - HARRISON,C.N., "Application of Soft Decision Techniques to Block Codes", proc.IERE Conference on Digital Processing of Signals in Communication, Loughborough, England, n? 37, 1977.

)18 i - HARTMANN,C.R.P., RUDOLPH,L.D.," "An Optimum Symbol-by-Symbol Decoding Rule for Linear Codes", IEEE Trans.Inform.Theory , Vol.IT-22, n9 5, pp. 514-517, sep. 1976.

119 I - HARTMANN,C.R.P., RUDOLPH,L.D., and MEHROTRA,K.G.,"Asymptotic Performance of Optimum Bit-by-Bit Decoding for the White Gaussian Channel", IEEE Trans, Inform. Theory, vol. IT-22 , n9 5, pp. 520-522, July 1977,

120 I - HERSTEIN,I.N., "Topics in Algebra", Blaisdell Publishing Company, Massachussets, 1964.

121 I - KÍEFER,J., "Conditional Confidence Statements and Confidence Estimators", J.ASA, 72, pp. 789-827, 1977.

122 I - KNUTH,D.E., "The Art of Computer Programming Seminumerical Algorithms - Vol.11", Addison-Wesley series in computer science and information processing, mass. 1973.

123 I - LATHI,B.P., "Random Signals and Communication Theory", International text book company, Pennsylvania, 1968.

124 J - LUCKY,R.W., SALZ,J., WELDON JR., E.J., "Principles of Data Communication", Bell telephone laboratories, Inc., McGraw Hill Book Company, 1968.

125 I - MATIS,K.R., MODESTINO.J.W., "Reduced-Search Soft-Decision ^ Trellis Decoding of Linear Block Codes", IEEE Trans. Inform', Theory, vol.IT-28, n? 2, March 1982.

126 J - PETERSON,W.W., WELDON JR.,E.J., "Error Correcting Codes", MIT Press., 2nd Edition, Massachussets, 1972.

127] - ROCHA JR.V.C, "Decoding Using Soft-Decision", DES publica - ção n9 4, Universidade Federal de Pernambuco, dez.1979.

128I - ROCHA JR.V.C., "Versatile Error Control Coding Systems", Ph.D.Thesis, University of Kent at Canterbury, England, 1976.

129 J - ROCHA JR.,V.C, "An Optimum Soft-Decision Receiver for Block Codes", Comunicação privada, 1982.

13QI - ROCHA JR.V.C, "Códigos Corretores de Erros, Minicurso apresentado no V Congresso nacional da SBMAC, João Pessoa, agosto 1982.

131 I - ROCHA JR.V.C, OLIVEIRA,H.M. , Receptor ótimo para Códigos de Bloco usando Decisão Suave", Comunicação apresentada no V Congresso da SBMAC, João Pessoa, agosto 1982.

132I - RUDIN,W., "Principles, óf Mathematical Analysis", McGraw-Hill book company, 1964.

l33 l - SCHwARTZ,M., "Information Transmission, Modulation and Noise", McGraw-Hill kogakusha, l t d . , 2nd e d i t i o n , 1970.

134 l - SCHWARTZ,M., BENNETT.W.R., STEIN,S., "Communications Systems and Techniques", McGraw-Hill book company, 1966.

135 I - SHANNON,C, "A Mathematical Theory of Communication", Bell systems tech.j., vol.27, (pt.Il, pp.374-427, (pt.II), pp.623-656, 1948.

136 l - SHU LIN, "An Introduction to Error-Correcting Codes", Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1970.

I37 l - VITERBI,A.J., OMURA,J.V., "Principles of Digital Communications and Coding", McGraw-Hill book company, 1979.

]38 J - WOLF,J.K., "Efficient Maximum Likelihood Decoding of Linear Block Codes Using a Trellis", IEEE Trans,Inform.Theory, vol. IT-24, n9 1, pp. 76-80, jan. 1978.

J39 I - WOLFENSON,M. , ROCHA JR.V.C, "Soft Decision Decoding with Unknow Source Distribution", Electronics Letters, vol.16, n9 25/26, dec. 1980.

140 l - WOZENCRAFT,J.M., JACOBS,I.M., Principles of Communication Engineering", John Wiley & Sons, Inc. New York, 1967.