



Nº 16

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DO PARANÁ
CURSO DE PÓS-GRADUAÇÃO EM INFORMÁTICA INDUSTRIAL**

**ESPECIFICAÇÃO DE UM SISTEMA DE
MODULAÇÃO CODIFICADA
COM CÓDIGOS DE BLOCO**

POR

EMÍLIO CARLOS GOMES WILLE

**DISSERTAÇÃO SUBMETIDA COMO REQUISITO PARCIAL À
OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS**

PROF. DR. WALTER GODOY JÚNIOR (Orientador)
PROF. DR. CELSO DE ALMEIDA (Examinador)
PROF. DR. DALTON S. ARANTES (Examinador)

CURITIBA, SETEMBRO, 1991

Wille, Emílio Carlos Gomes; Especificação
de um Sistema de Modulação Codificada com
Códigos de Bloco, Curitiba, CPGII do
CEFET, 1991.

1 v.

Diss. (mestr. inf. ind.) CEFET-CPGII,
PR-BR, 1991

Dissertação : Códigos Controladores de
Erro : Modulação Codificada :
Transmissão de Dados.

À meus pais, Edgard Wille e Ana Balbina,
pelo amor e dedicação;
Ao meu irmão, Sidney Luiz,
pela amizade e companheirismo;
À Roseli Aparecida,
pela paciente espera.

*Amado Senhor, que nos tendes guiado para
a luz de Vossa glória, pela luz da
Natureza, graças a Vós sejam dadas. Vede
que terminei a obra de que me incumbistes
e rejubilo-me em Vossa criação, cujas
maravilhas me permitistes revelar aos
homens. Amém.*

Johannes Kepler

AGRADECIMENTOS

Ao meu orientador, Prof. Dr. Walter Godoy Júnior, que me auxiliou nesta jornada, sempre apontando o caminho correto nos momentos de dúvida.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq, pelo auxílio financeiro através da concessão de bolsa de estudo. À Nutron S.A. - Equipamentos e Sistemas Eletrônicos, pelo incentivo financeiro na etapa inicial deste trabalho.

À Parks Informática S.A., pelo estágio oferecido. Ao Laboratório Central de Eletrotécnica e Eletrônica (LAC) - UFPR/COPEL, pelo desenvolvimento do Circuito Integrado a partir dos algoritmos especificados neste trabalho.

Ao Centro Federal de Educação Tecnológica do Paraná - CEFET/PR. Aos professores, funcionários e amigos do Curso de Pós-Graduação em Informática Industrial - CPGII.

Aos meus familiares, pelo apoio e incentivo.

RESUMO

Neste trabalho é especificado um sistema de Modulação Codificada que utiliza códigos de bloco corretores de erros. O sistema é considerado como um codificador para a formação de códigos de bloco multiníveis, baseado em códigos de bloco binários. O sistema básico proposto é estendido à modulação 16-QAM. São analisados dois processos de decodificação para os códigos multiníveis propostos, baseados na representação destes em códigos binários. Os desempenhos dos algoritmos de decodificação são levantados por simulação em computador. São especificados algoritmos para auxílio à busca de conjuntos de cobertura para o Algoritmo Conjuntos de Informação, e algoritmos para busca de códigos com distâncias mínimas maximizadas. Como resultado, especifica-se um circuito integrado, com finalidade didática, de codificação e decodificação para a modulação 16-QAM.

ABSTRACT

In this work, a Coded Modulation System that utilizes Block Codes for error correction, is proposed. It is regarded as an encoder to generate Multilevel Block Codes, based on binary block codes. The basic system introduced is extended to QAM modulation. Two decoding processes are analysed for the Multilevel Codes introduced here, based on their representation as binary codes. The performance of the decoding algorithm is found by computer simulation. Algorithms are specified for the search of Covering Sets for the Information Set Decoding, and algorithms are given for the search of codes with maximized minimum distances. As a result, an integrated circuit for the encoder and the decoder using 16-QAM modulation is specified here for teaching purpose.

SUMÁRIO

LISTA DE FIGURAS.....	12
LISTA DE TABELAS.....	14
GLOSSÁRIO.....	15

CAPÍTULO 1

INTRODUÇÃO, MOTIVAÇÃO E OBJETIVOS.....	16
---	-----------

CAPÍTULO 2

CONCEITOS SOBRE TRANSMISSÃO DE DADOS.....	20
--	-----------

2.1 Diagrama em blocos do sistema de transmissão de dados.....	20
2.2 Processos de modulação.....	21
2.3 Efeitos indesejados provocados pela rede telefônica..	24
2.4 Capacidade do canal com ruído.....	25
2.5 Modem's analógicos.....	26
2.6 Correção de erros e eficiência na transmissão de dados.....	29
2.7 Conclusões.....	30

CAPÍTULO 3

MODULAÇÃO CODIFICADA.....	31
----------------------------------	-----------

3.1 Modelo discreto de um sistema de comunicação digital.	32
3.2 Códigos corretores de erros - Conceitos básicos.....	33
3.3 Princípio de decodificação.....	40
3.4 Decisão abrupta e suave.....	41
3.5 Parâmetros para avaliação de desempenho.....	42
3.6 Modelos de erros binários	43

3.6.1 Alfabetos redundantes.....	45
3.6.2 Capacidade de canal de sistemas codimodulados..	45
3.6.3 Partição de conjunto.....	47
3.7 Modulação codificada com códigos de treliça.....	47
3.7.1 Códigos convolucionais invariantes quanto a rotação de fase.....	51
3.8 Modulação codificada com códigos de bloco - Histórico e principais sistemas.....	57
3.8.1 Sistema de Imai e Hirakawa.....	59
3.8.1.1 Descrição do sistema.....	59
3.8.1.2 A decodificação.....	60
3.8.2 Códigos de Sayegh.....	63
3.8.2.1 A codificação.....	63
3.8.2.2 A decodificação.....	65
3.8.3 Sistema de decodificação de Williams e Farrel..	67
3.8.3.1 A decodificação.....	67
3.8.4 Sistema de Forney.....	71
3.8.4.1 Particionamento de conjunto.....	71
3.8.4.2 Esquema geral da modulação codificada..	72
3.8.5 Sistema de Tanner.....	73
3.8.5.1 Construção recursiva de códigos.....	74
3.8.6 Sistema de Baldini.....	76
3.8.6.1 A codificação.....	77
3.8.6.2 Distância euclidiana entre sinais qPSK.	78
3.8.6.3 Códigos de bloco multiníveis.....	79
3.8.6.4 Códigos de bloco circulantes sistêmáticos.....	79
3.8.6.5 Códigos pseudo-cíclicos multiníveis....	80
3.8.6.6 Propriedades dos códigos pseudo-cíclicos multiníveis.....	82
3.8.6.7 Códigos pseudo-cíclicos sobre \mathbb{Z}_4 para modulação 16-QAM codificada.....	83
3.8.6.8 Comparação com o sistema de Sayegh....	85
3.8.7 Análise comparativa dos sistemas de modulação codificada.....	86

CAPÍTULO 4
SISTEMA DE MODULAÇÃO CODIFICADA COM CÓDIGOS DE BLOCO
MULTINÍVEIS E DECODIFICAÇÃO POR REPRESENTAÇÃO BINÁRIA.... 88

4.1 Processo de codificação proposto.....	89
4.2 Características dos códigos multiníveis formados.....	91
4.2.1 Exemplo de codificação.....	93
4.3 Processo de decodificação proposto.....	95
4.3.1 Exemplo de decodificação.....	98
4.4 Simulação do sistema de modulação codificada.....	100
4.5 Análise de desempenho.....	102
4.6 A representação do sistema por uma única matriz.....	107
4.7 Algoritmo conjuntos de informação.....	111
4.8 Sistema de decodificação com ACI.....	112
4.9 Comparação com o sistema de Sayegh.....	117
4.10 Comparação com o sistema de Baldini.....	117
4.11 Conclusões.....	118

CAPÍTULO 5
ESPECIFICAÇÃO DE UM SISTEMA DE MODULAÇÃO 16-QAM
CODIFICADO..... 120

5.1 Especificação do sistema 16-QAM.....	121
5.2 Especificação do codificador.....	122
5.3 Especificação do decodificador.....	123
5.3.1 Exemplo de codificação e decodificação.....	125
5.4 Análise dos algoritmos de codificação e	
decodificação.....	128
5.4.1 Complexidade de algoritmos.....	129
5.4.2 Algoritmos do sistema de codificação.....	129
5.4.3 Algoritmos do sistema de decodificação.....	132
5.5 Desempenho do sistema de decodificação.....	135
5.6 Circuito integrado para codificação e decodificação..	137
5.7 Conclusões.....	139

CAPÍTULO 6**BUSCA DE CONJUNTOS DE COBERTURA E DE CÓDIGOS.....141**

6.1 Busca de conjuntos de cobertura.....	142
6.1.1 Algoritmo de verificação de máscara válida.....	142
6.1.2 Algoritmo de determinação de padrões não cobertos.....	144
6.1.3 Algoritmo de pré-seleção de máscaras.....	145
6.1.4 Algoritmo de escolha de máscara de maior cobertura.....	146
6.1.5 Algoritmo de formação de banco de máscaras.....	147
6.2 Os algoritmos em conjunto.....	149
6.2.1 Exemplo de busca de conjuntos de cobertura.....	152
6.3 Automatização de processo de busca.....	156
6.4 Busca de códigos multiníveis melhores.....	157
6.4.1 A estrutura dos códigos buscados.....	158
6.4.2 Algoritmo de busca.....	160
6.4.3 Códigos obtidos com o processo de busca proposto.....	162
6.5 Busca de conjunto de cobertura e desempenho para o código C4.....	165

CAPÍTULO 7**CONSIDERAÇÕES FINAIS.....168****REFERÊNCIAS BIBLIOGRÁFICAS**

LISTA DE FIGURAS

2.1 Diagrama em blocos de um sistema de transmissão de dados.....	21
2.2 Constelações em uma e duas dimensões.....	23
2.3 Diagrama em blocos do transmissor.....	26
2.4 Diagrama em blocos do receptor.....	28
3.1 Modelo de um sistema de comunicação digital.....	32
3.2 Capacidade de canal para canais limitados em faixa para modulação bi-dimensional.....	46
3.3 Particionamento da constelação 8-PSK.....	48
3.4 Estrutura do sistema TCM.....	49
3.5 Diagrama de treliça de quatro estados.....	51
3.6 Constelação 32-CR.....	52
3.7 Codificador do modem V-32 do CCITT.....	55
3.8 Estrutura do codificador.....	57
3.9 Estrutura do sistema.....	60
3.10 Circuito decodificador para $m = 4$	62
3.11 Arranjo-código de 3 linhas e n colunas.....	64
3.12 Ponto recebido na constelação 8-PSK.....	68
3.13 16-QAM construída de duas 4-AM.....	69
3.14 Particionamento da constelação bidimensional.....	72
3.15 Esquema geral de modulação codificada.....	73
3.16 Codificador paralelo para códigos de Reed-Muller de comprimento 16.....	75
3.17 Estrutura do codificador.....	77
3.18 Partição da constelação 16-QAM em quatro conjuntos..	84
4.1 Codificador multinível proposto.....	89
4.2 Exemplo de codificação.....	94
4.3 Fluxograma de decodificação (Chase).....	96
4.4 Símbolo recebido na constelação 4-PSK.....	96
4.5 Sistema de decodificação binário.....	97
4.6 Exemplo de decodificação.....	99
4.7 Diagrama da decodificação MLR.....	104

4.8 Desempenho do algoritmo proposto (Chase).....	105
4.9 Desempenho do código C3.....	106
4.10 Sistema de codificação com matriz T.....	107
4.11 Sistema de decodificação binário com matriz T.....	108
4.12 Desempenho de decodificação com a matriz T.....	109
4.13 Fluxograma de decodificação (ACI).....	113
4.14 Desempenho de decodificação com ACI.....	115
4.15 ACI com apagamento.....	116
5.1 Arranjo-código para a versão 1.....	121
5.2 Arranjo-código para a versão 2.....	122
5.3 Diagrama em blocos do sistema codificador.....	123
5.4 Diagrama em blocos do sistema decodificador.....	124
5.5 Constelação 16-QAM.....	126
5.6 Desempenho do sistema de decodificação 16-QAM.....	136
5.7 Circuito integrado para codificação e decodificação..	138
6.1 Algoritmo de verificação de máscara válida (AI).....	144
6.2 Determinação de padrões não cobertos (AII).....	145
6.3 Escolha de máscaras para cobrir padrões não cobertos (AIII).....	146
6.4 Algoritmo de seleção (AIV).....	147
6.5 Algoritmo de formação de um banco de máscaras válidas (AV).....	148
6.6 Interrelacionamento entre algoritmos.....	150
6.7 Automatização do processo de busca.....	157
6.8 Fluxograma de busca de códigos.....	161
6.9 Desempenho do sistema 4-PSK.....	166
6.10 Desempenho do sistema 16-QAM.....	167

LISTA DE TABELAS

3.I Representação dos elementos de GF(4).....	40
3.II Códigos de Sayegh.....	67
3.III Códigos BLCS definidos sobre Z_4	81
3.IV Códigos BLCS definidos sobre Z_8	81
3.V Códigos pseudo-cíclicos definidos sobre Z_4	82
3.VI Códigos pseudo-cíclicos definidos sobre Z_8	82
3.VII Códigos para 16-QAM.....	85
4.I Códigos multiníveis lineares em A_4 ($R = 1/2$).....	94
6.I Códigos multiníveis para 4-PSK ($R = 1/2$).....	163
6.II Códigos multiníveis para 8-PSK ($R = 1/2$).....	164

GLOSSÁRIO

- ACI - Algoritmo Conjuntos de Informação
ACIM - Algoritmo Conjuntos de Informação Modificado
ARQ - Requisição Automática de Repetição
ASK - Modulação por Chaveamento de Amplitude
AWGN - Canal Gaussiano de Ruído Aditivo Branco
BCM - Modulação Codificada com Códigos de Bloco
BLCS - Códigos de Bloco Lineares Circulantes Sistemáticos
DPSK - Modulação em Fase Diferencial
ECD - Equipamento de Comunicação de Dados
EPROM - Memória de Leitura Programável Apagável
ETC - Equipamento Terminal de Dados
FEC - Correção Automática de Erros
FSK - Modulação por Chaveamento de Freqüência
HDC - Programa de Descrição de Circuito Eletrônico em C
MLD - Decodificador de Máxima Verossimilhança
PSK - Modulação por Chaveamento de Fase
QAM - Modulação em Amplitude e Quadratura
TCM - Modulação Codificada com Códigos de Treliça
TEB - Taxa de Erro de Bit

CAPÍTULO 1

INTRODUÇÃO, MOTIVAÇÃO E OBJETIVOS

O conceito de Modulação Codificada tem sido de grande interesse durante os últimos anos. O uso desta técnica tornou-se um padrão em modem's de alta velocidade ($v \geq 9600$ bit/s) para a transmissão de dados através de canais telefônicos [WEI 84]. No desenvolvimento de sistemas confiáveis de transmissão de dados, a Modulação Codificada é de importância fundamental, pois, além de reduzir a Taxa de Erro de Bit, ela garante a utilização eficiente de recursos, tais como a energia e o espectro de freqüências [UNG 82].

O ruído é a fonte primária de erros na transmissão de dados. A taxa de erros pode ser reduzida pelo aumento da potência (energia) de transmissão, mas ela (potência) está sujeita a limites estipulados por normas. A introdução de redundância à informação, através do uso de Códigos Corretores de Erros, permite a redução da taxa de erro de bit, sem maior necessidade de energia, mas com aumento da faixa de freqüências necessária para a transmissão da mesma quantidade de Bits de Informação por intervalo de tempo. Com a necessidade de taxas de transmissão sempre crescente, a solução seria combinar os sistemas de modulação e codificação, resultando nos sistemas de Modulação Codificada, atualmente pesquisados, cujo desempenho é melhor se comparado a sistemas em que a codificação e modulação são feitas separadamente.

Ungerboeck [UNG 82] introduziu um sistema com esta filosofia que assegura uso eficiente de energia com conservação de faixa de freqüências. Este sistema utiliza um

Código Convolucional que faz o controle da transmissão dos sinais da modulação e, por isso, é chamado de sistema de Modulação Codificada com Códigos de Treliça (TCM). Imai & Hirakawa [I&H 77], Cusack [CUS 84] e Sayegh [SAY 86] estabeleceram sistemas em que se utiliza, no controle de transmissão dos sinais da modulação, Códigos de Bloco Binários como meio de obter sistemas com o mesmo desempenho obtido por Ungerboeck, mas com menor complexidade. Esses sistemas são conhecidos como sistemas de Modulação Codificada com Códigos de Bloco (BCM). Recentemente, Baldini [BAL 89] apresentou um sistema que representa uma completa alteração quanto a filosofia da Modulação Codificada. Esse sistema realiza primeiro um mapeamento da informação binária a ser transmitida em um conjunto expandido de Sinais de Modulação e, em seguida, realiza a codificação com Códigos de Bloco Multiníveis. Esse sistema apresenta desempenhos tão bons quanto os de Ungerboeck e Sayegh.

Embora os sistema TCM tenham-se tornado populares e candidatos a uma grande gama de sistemas práticos, eles ainda não são factíveis economicamente em aplicações de muito alta velocidade, como, por exemplo, em sistemas de rádio microondas digital de alta capacidade ou em comunicações a alta velocidade via satélite. Nessas situações, os sistemas BCM tem sido empregados por apresentarem uma complexidade menor [C&S 90] [KAS 90].

O principal objetivo deste trabalho é desenvolver um sistema de obtenção e decodificação para Códigos de Bloco Multiníveis a partir de códigos de bloco binários para a Modulação Codificada; a partir desse sistema especificar um Circuito Integrado para codificação e decodificação, com finalidade didática, e, através de simulação em computador (padrão IBM-PC), levantar curvas de desempenho de decodificação para os algoritmos propostos. Como objetivos secundários, este trabalho especifica algoritmos para auxílio à busca de dados que são utilizados no Algoritmo Conjuntos de Informação (ACI) usado para decodificação dos códigos

propostos, e, também, especifica algoritmos para busca de códigos melhores, ou seja, com distâncias mínimas maximizadas. Esses códigos são representados e decodificados binariamente pelo algoritmo ACI que apresenta baixa complexidade quando comparado a outros algoritmos existentes, conforme provado recentemente [C&G 90].

Algumas questões estão associadas à representação de um código multinível por um ou mais códigos binários, ou seja :

- A decodificação do código de bloco multinível pode ser realizada pela decodificação do(s) código(s) de bloco binário(s) ?
- Qual é o desempenho em decodificação alcançado por um sistema com esta filosofia ?
- O sistema de decodificação é rápido ?
- Sua implementação é factível ?

Estas questões serão respondidas para os sistemas desenvolvidos neste trabalho.

A abordagem dos assuntos está dividida em sete capítulos :

O capítulo 2 apresenta conceitos básicos sobre transmissão de dados. O capítulo 3 apresenta uma revisão sobre códigos controladores de erros e introduz a Modulação Codificada com Códigos de Treliça (TCM) e com Códigos de Bloco (BCM). Os principais sistemas BCM são apresentados e analisados, já que este é o enfoque principal do trabalho. No capítulo 4, é desenvolvida a proposta para um sistema de obtenção de códigos de bloco multiníveis para a Modulação Codificada. Dois métodos de decodificação são analisados : uma variante do algoritmo de Chase e outra do Algoritmo Conjuntos de Informação. O desempenho dos algoritmos é obtido por simulação computacional. No capítulo 5, são especificados os algoritmos de codificação e decodificação

para um sistema com modulação 16-QAM, bem como especifica-se o circuito integrado didático para realizar a codificação e decodificação e analisa-se a complexidade dos algoritmos envolvidos. No capítulo 6, são especificados algoritmos para auxiliar a busca de Máscaras para a formação de Conjuntos de Cobertura para o Algoritmo Conjuntos de Informação e algoritmos para busca de códigos multiníveis mais poderosos. No capítulo 7, relembram-se os pontos principais do trabalho e apresentam-se conclusões e sugestões.

CAPÍTULO 2

CONCEITOS SOBRE TRANSMISSÃO DE DADOS

A Comunicação de Dados é o campo das telecomunicações de mais rápido crescimento, atualmente. Isto se deve ao uso da ampla rede telefônica instalada e a possibilidade da transmissão analógica de sinais digitais conseguida com o uso do modem.

Este capítulo apresenta conceitos básicos sobre transmissão de dados e situa o conjunto codificador/decodificador para correção de erros (capítulo 3), considerando o diagrama em blocos de um modem analógico.

2.1 - DIAGRAMA EM BLOCOS DO SISTEMA DE TRANSMISSÃO DE DADOS

Geralmente temos uma rede de comunicação de dados sobre a rede telefônica, de modo a permitir que qualquer Equipamento Terminal de Dados (ETD) ligado à rede possa ser conectado a qualquer outro ETD também ligado à rede.

O diagrama simplificado de um sistema de transmissão de dados é mostrado na figura 2.1.

O sistema divide-se em uma parte digital e uma parte analógica. A parte digital é composta de :

- a) Equipamento Terminal de Dados (ETD) : Cumpre as funções básicas de ser fonte e destino de dados e controlar a comunicação.

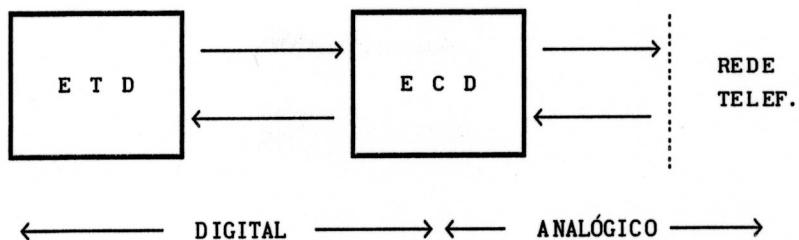


FIGURA 2.1 - DIAGRAMA EM BLOCOS DE UM SISTEMA DE TRANSMISSÃO DE DADOS

b) Equipamento de Comunicação de Dados (ECD) :
 Equipamento que transforma os sinais portadores de informação a transmitir em outros que, contendo a mesma informação mais alguma adicional de uso exclusivo dos ECD's, sejam fáceis de serem transmitidas até o ETD remoto através dos meios de comunicação disponíveis. Na maioria dos sistemas, o ECD utilizado é o modem (modulador/demodulador).

A parte analógica do sistema, ou seja, o canal analógico é obtido da rede telefônica. A qualidade da transmissão depende diretamente de quanto é bom o segmento mais pobre da rede.

2.2 - PROCESSOS DE MODULAÇÃO

Para uma transmissão eficiente, escolhe-se um sinal denominado Portadora, que possa ser facilmente transmitido pelo meio, e altera-se uma das características desta com a informação que se quer transmitir. Este é o processo de modulação. A informação que se deseja transmitir é transportada pela Moduladora.

A escolha da portadora deve-se a razões tecnológicas de facilidade de produção e modulação; a moduladora depende da fonte de origem. Para a transmissão digital de dados pela rede telefônica, utiliza-se a modulação de portadora co-

senoidal com moduladora digital. Neste tipo de modulação, os sinais digitais a serem transmitidos alteram os parâmetros amplitude, fase ou freqüência, que definem uma onda portadora co-senoidal (equação 2.1), formando três sistemas básicos de modulação :

$$s(t) = A \cdot \cos(w_s \cdot t + \phi(t)) \quad (2.1)$$

↑ ↑ ↑
Amplit. Freq. Fase

ASK (Modulação por Chaveamento de Amplitude) - A cada valor de sinal de entrada, faz-se corresponder um nível diferente de amplitude de portadora.

FSK (Modulação por Chaveamento de Freqüência) - Consiste em variar a freqüência da portadora em valores pré-determinados conforme o sinal de entrada.

PSK (Modulação por Chaveamento de Fase) - Desloca a fase da portadora em quantidades discretas em função do sinal de entrada.

Aplicando trigonometria à equação 2.1 chega-se a :

$$s(t) = A \cdot \left\{ \cos(w_s \cdot t) \cdot \cos(\phi(t)) - \sin(w_s \cdot t) \cdot \sin(\phi(t)) \right\}$$

Fazendo : $I(t) = A \cdot \cos(\phi(t))$ e $Q(t) = -A \cdot \sin(\phi(t))$
tem-se :

$$s(t) = I(t) \cdot \cos(w_s \cdot t) + Q(t) \cdot \sin(w_s \cdot t) \quad (2.2)$$

Assim, a portadora modulada da equação 2.2 pode ser vista como um vetor no espaço bidimensional, de componente em fase $I(t)$ e componente em quadratura $Q(t)$. O conjunto de pontos definidos, no espaço, por estas componentes chama-se Constelação de Sinais.

A modulação QAM é uma combinação da modulação em amplitude com modulação em fase. Quando o número de sinais em uma constelação é maior do que dois, os esquemas de modulação são ditos Q-ários. A figura 2.2 mostra constelações em uma e duas dimensões. As vantagens e desvantagens dos diferentes tipos de modulação são, normalmente, relacionadas à largura de faixa necessária, imunidade a ruído e potência média necessária.

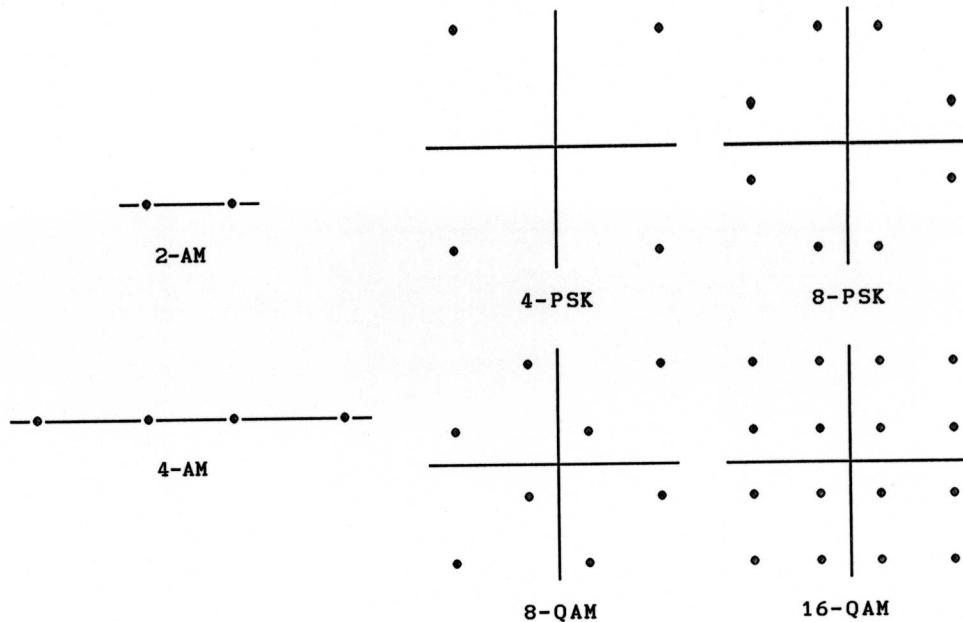


FIGURA 2.2 - CONSTELAÇÕES EM UMA E DUAS DIMENSÕES

Com Q possíveis pontos na constelação pode-se transmitir $\log_2 Q$ mais informação, no mesmo intervalo de tempo, se comparado à condição binária. Deve-se considerar que, quanto mais informação por intervalo de modulação, maior a exigência quanto à relação sinal-ruído do meio para manter a mesma taxa de erro. A disposição dos pontos na constelação de sinais é responsável por maior ou menor imunidade ao ruído, pois esta disposição influencia no processo de demodulação e afeta a Energia Média da constelação, definida por [PRO 83]:

$$E_c = \frac{1}{Q} \sum_{i=1}^Q (x_i^2 + y_i^2) = \frac{1}{Q} \sum_{i=1}^Q dE_i^2 \quad (2.3)$$

para uma constelação de Q pontos, onde x_i e y_i são as coordenadas dos pontos na constelação e dE_i são as distâncias dos pontos em relação à origem. Para duas constelações de mesma potência média e mesmo número de pontos, aquela, cuja distância Euclidiana mínima entre os pontos for maior, apresenta a menor probabilidade de erro de símbolo [THO 74].

O processo inverso à modulação é a demodulação que consiste na reconstrução a partir do sinal recebido dos sinais digitais de dados que o originaram. O problema está no fato de que o sinal recebido não é o mesmo que deixou o modulador de origem, devido a efeitos indesejados da transmissão pelo canal, vistos a seguir.

2.3 - EFEITOS INDESEJADOS PROVOCADOS PELA REDE TELEFÔNICA

Como a rede telefônica é uma sucessão um tanto complexa de meios de transmissão, encadeados de forma permanente ou temporal, suas características são variáveis, o que perturba a transmissão .

Os principais fenômenos que afetam a transmissão de dados são a Distorção em Amplitude, a Distorção por Retardo de Envelope e o Ruído. Os dois primeiros fenômenos podem ter seus efeitos reduzidos a limites toleráveis através do uso de equalização.

O ruído é todo elemento estranho que se apresenta na saída do canal de voz, e não é devido ao sinal de entrada. Em transmissão de dados só são considerados os ruídos Gaussiano (Térmico) e o Impulsivo.

O ruído térmico pode ser de natureza Gaussiana ou puramente aleatório e é causado por movimentos aleatórios de elétrons. Aparentemente seu espectro é relativamente constante, mas seu valor instantâneo varia sobre uma larga faixa de níveis de amplitude. Se a tensão instantânea do ruído é da mesma magnitude do sinal recebido, o equipamento de detecção pode fazer uma interpretação errônea do sinal recebido e um erro pode ocorrer. Para um tratamento analítico do problema de transmissão de dados, é necessário assumir um tipo de ruído possuidor de uma distribuição de amplitude que segue um padrão determinado. Assim, adota-se o ruído Branco com distribuição Gaussiana que é considerado representativo do ruído encontrado no canal analógico telefônico [FRE 81].

O ruído impulsivo é a fonte primária de erros na transmissão de dados sobre a rede telefônica; são esporádicos e podem ocorrer em surtos ou como impulsos discretos e são caracterizados pela sua curta duração e elevada amplitude. Este tipo de ruído pode ter origem natural, como por exemplo o ruído causado pelos relâmpagos, e artificial, como o causado pela ignição de automóveis.

2.4 - CAPACIDADE DO CANAL COM RUÍDO

Shannon [SHA 48,69] provou que a capacidade máxima que um canal de largura de faixa B, sujeito ao ruído Branco Aditivo e Gaussiano de potência N, pode transmitir, é dada por :

$$C = B \cdot \log_2(1 + S/N) \quad [\text{bit/s}] \quad (2.4)$$

onde S é a potência média do sinal.

Por exemplo, para um canal com largura de faixa B = 3100 Hz e com uma relação sinal-ruído de 30 dB (condições normais), tem-se uma capacidade máxima de transmissão sobre o canal de $C = 31000$ bit/s. Na prática, os efeitos indesejados

provocados pela rede telefônica reduzem esta capacidade aos 19200 bit/s dos modem's de alta velocidade.

2.5 - MODEM'S ANALÓGICOS

Para transmissões de dados a longa distância, como por exemplo em ligações interurbanas ou internacionais, sobre um meio limitado em faixa (o canal telefônico), utiliza-se o modem analógico.

Neste item são apresentados os principais blocos que compõem a estrutura interna de um modem analógico, auxiliando a correta localização do sistema codificador e decodificador para correção de erros da modulação codificada.

A figura 2.3 mostra o diagrama em blocos do transmissor para um modem DPSK de 4800 bit/s (recomendação V.27 do CCITT). Trata-se de um exemplo particular [SIL 90], já que várias soluções são plausíveis.

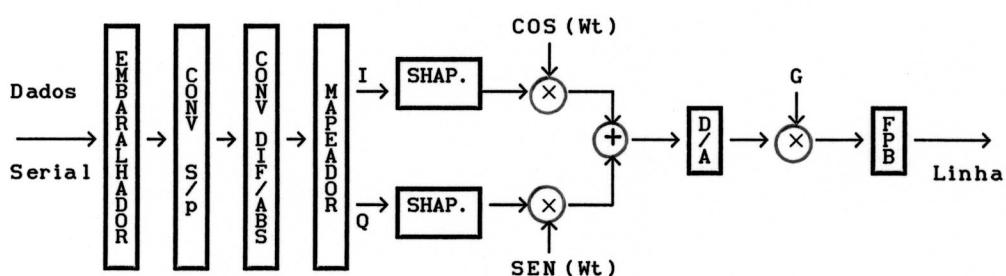


FIGURA 2.3 - DIAGRAMA EM BLOCOS DO TRANSMISSOR

O transmissor é formado por :

- a) Embaralhador - Sua finalidade é evitar a transmissão de uma seqüência de pontos iguais na constelação, facilitar a recuperação de sincronismo e garantir o funcionamento dos algoritmos de equalização do receptor.

b) Conversor Série/Paralelo - Converte a seqüência de bits de informação em tribits, pois na recomendação V.27 a constelação usada possui oito pontos.

c) Conversor Diferencial/Absoluto - A cada tribit corresponde uma troca de fase. O conversor adiciona esta troca de fase à fase absoluta anterior, determinando a fase absoluta a ser transmitida.

d) Mapeador - Determina as componentes em fase (I) e quadratura (Q) correspondente à fase absoluta a ser transmitida.

e) Modulador Digital e Filtros de Amoldamento (Shaping) - A saída do mapeador deve ser modulada para que todo o espectro seja confinado dentro da banda do canal telefônico. A modulação é obtida pela aplicação da equação 2.2. Como as transições instantâneas na saída do mapeador geram harmônicas de alta ordem, faz-se necessário o uso de filtragem.

f) Conversor Digital/Analógico - Representa a última etapa do processamento digital. Observa-se que, embora o modem seja denominado analógico, seu funcionamento é digital com exceção da transmissão analógica pela linha.

g) Filtro Analógico e Estágio de Ganho - A saída analógica do conversor D/A possui formato de escada, ou seja, apresenta componentes de alta freqüência. Assim, deve-se utilizar um filtro passa-baixas que efetua, também, uma pré-equalização em amplitude, para compensar as características da linha. Um estágio de ganho é necessário para entregar à linha a potência estipulada por norma.

A figura 2.4 mostra o diagrama em blocos do receptor do modem [SIL 90].

O receptor é formado por :

a) Filtro Analógico Passa-Faixa - Antes da amostragem, filtra-se o sinal para eliminar o ruído fora da faixa de voz.

b) Controle Automático de Ganho - A amplitude do sinal na linha varia devido à potência transmitida e à atenuação introduzida. Deve-se manter a amplitude constante, pois é fundamental para os algoritmos de recuperação de portadora e de sincronismo.

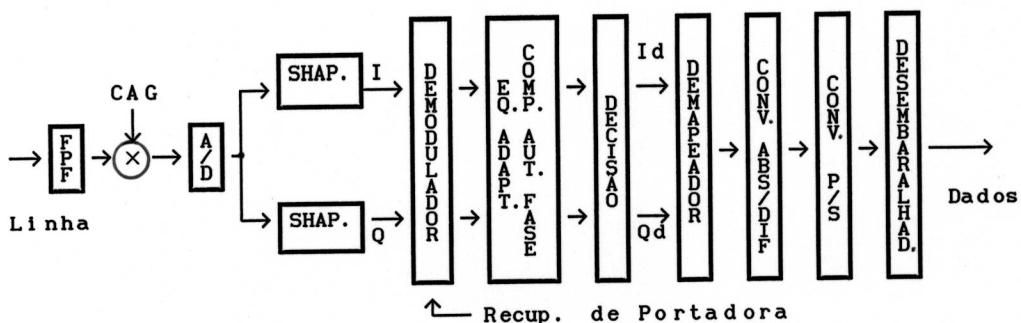


FIGURA 2.4 - DIAGRAMA EM BLOCOS DO RECEPTOR

c) Conversor Analógico/Digital - É a etapa inicial do tratamento digital, realizando operação inversa à ocorrida na transmissão.

d) Filtros de Amoldamento e Demodulador - Os filtros são utilizados para rejeição do ruído fora da faixa de voz, equalização fixa devido às distorções trazidas pela linha e para a introdução de um desvio de fase relativo de 90° , para a separação dos canais I e Q, necessário ao demodulador. O demodulador translada a informação de volta a banda-base.

e) Equalizador Adaptativo e Compensador Automático de Fase - Os sinais após o demodulador não são de todo corretos devido ao retardado de envelope, à tremulação de fase e ao ruído. Os efeitos dos dois primeiros fenômenos são corrigidos pelos algoritmos de equalização e de compensação de fase, respectivamente.

f) Bloco de Decisão, "Demapeador" e Desembaralhador - O bloco de decisão gera os valores decididos Id e Qd. A diferença entre os valores antes e depois do bloco de decisão constituem informação de erro, necessários aos algoritmos de equalização, de compensação de fase e de recuperação de portadora. O "demapeador" determina a fase absoluta atual. O conversor A/D recebe a fase absoluta atual, subtrai da fase absoluta anterior, obtendo a troca de fase transmitida. Cada troca de fase corresponde a um tribit, que é serializado pelo conversor paralelo/serial e desembaralhado.

Este exemplo mostra a estrutura de um modem que não utiliza algoritmos de codificação e decodificação da modulação codificada. Em um modem de alta velocidade, por exemplo o V.32 que utiliza a modulação codificada para combater os erros de bit devido ao ruído, pode-se verificar que o sistema de codificação substitui o bloco Conversor Diferencial/Absoluto da figura 2.3. O sistema de decodificação deve ser colocado logo após o bloco de Decisão da figura 2.4.

Deve-se mencionar que, no caso V.32, outros blocos fazem-se necessários, como por exemplo o bloco de cancelamento de eco [CCITT].

2.6 - CORREÇÃO DE ERROS E EFICIÊNCIA NA TRANSMISSÃO DE DADOS

A comunicação entre os ETD's é realizada através da troca de um grande número de blocos de dados. Quando erros são detectados no bloco recebido no receptor, um reconhecimento negativo é enviado ao transmissor para a repetição do bloco. O processo continua até que o bloco seja recebido corretamente, e, então, é enviado um reconhecimento positivo. Este é um sistema de correção de erros por repetição do tipo ARQ ("Automatic Repeat Request") [L&C 83].

A eficiência da transmissão de dados pode ser dada pela razão entre o tempo gasto para a transmissão de um bloco de dados propriamente dito e o tempo gasto para enviar um bloco e receber um reconhecimento positivo.

Em canais muito ruidosos a eficiência da transmissão diminui devido ao tempo extra, gasto na retransmissão para corrigir blocos errados. Para melhorar a eficiência e confiabilidade, quando se transmite por estes canais, usa-se a modulação codificada que reduz a taxa de erro de bit, empregando a correção automática de erros FEC ("Forward Error Correction"), que será assunto do capítulo 3.

2.7 - CONCLUSÕES

Este capítulo apresentou conceitos básicos sobre transmissão de dados, entre eles : a divisão do sistema de transmissão de dados em uma parte digital e uma analógica, os principais sistemas de modulação, o ruído introduzido pelo canal e o diagrama em blocos de um modem analógico. Concluiu-se que a aplicação da modulação codificada é fundamental em transmissões sobre canais muito ruidosos, com restrições em faixa e potência, por diminuir a taxa de erro de bit do sistema. Sua aplicação em modem's reduz o pedido de retransmissão de blocos de dados, aumentando a eficiência e a confiabilidade na transmissão.

CAPÍTULO 3

MODULAÇÃO CODIFICADA

A Modulação Codificada é uma técnica que torna os sistemas de transmissão de dados mais eficientes e confiáveis, melhor aproveitando a capacidade dos meios de transmissão disponíveis. Os meios de transmissão, geralmente, são limitados em faixa de freqüências, como é o caso das linhas telefônicas, ou limitados em potência, como no caso da comunicação via satélite.

A Modulação Codificada, introduzida por Ungerboeck, permite alcançar ganhos de codificação significativos em relação aos sistemas não codificados convencionais, sem alterar a taxa de transmissão de bits de informação e a largura de faixa [UNG 82]. De um modo simplificado, a não alteração em faixa resulta do fato de que a redundância necessária ao processo de codificação é fornecida por uma expansão no conjunto de sinais da modulação usada. Uma elegante e recente abordagem descreve a Modulação Codificada como um caso especial da Concatenação Generalizada de códigos sobre o espaço Euclidiano [POR 85]. Essa abordagem não será vista neste trabalho.

Este capítulo apresenta conceitos básicos sobre códigos corretores de erro, os princípios da modulação codificada, a aplicação da modulação codificada com códigos de treliça (TCM) a modem's de alta velocidade, e a modulação codificada com códigos de bloco (BCM) que representa uma alternativa aos sistemas TCM.

3.1 - MODELO DISCRETO DE UM SISTEMA DE COMUNICAÇÃO DIGITAL

O modelo discreto de um sistema de comunicação digital é mostrado na figura 3.1.

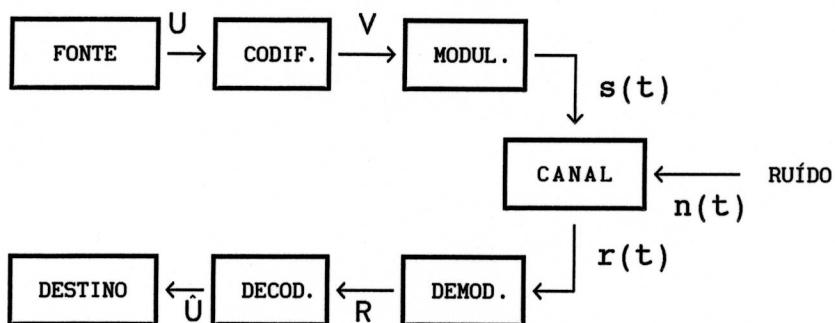


FIGURA 3.1 - MODELO DE UM SISTEMA DE COMUNICAÇÃO DIGITAL

A seguinte notação é adotada :

- U : Seqüência de símbolos de informação;
- V : Seqüência codificada ou palavra-código;
- s(t) : Sinal analógico apropriado para a transmissão de duração T segundos. Cada sinal é representado por um símbolo de canal p na constelação correspondente;
- n(t) : Ruído Gaussiano com densidade espectral unilateral de N_0 Watt/Hertz;
- r(t) : Sinal analógico alterado pelo ruído;
- R : Seqüência recebida que corresponde a v na ausência de erros de transmissão;
- Ú : Seqüência estimada de símbolos de informação.

A informação proveniente da fonte deve ser levada até o destino do modo mais confiável possível. O conjunto Codificador/Decodificador é projetado para combater os erros de transmissão. O conjunto Modulador/Demodulador permite a transmissão pelo meio.

Shannon, em 1948, mostrou teoricamente que, através de codificação apropriada da informação, os erros induzidos pelo ruído apresentado pelo canal podem ser reduzidos a qualquer nível desejado [SHA 48,69]. Na prática, existe um compromisso

entre a taxa de informação que se pode transmitir e a capacidade de correção de erros trazida pelo código. Esta correção é obtida com a adição de redundância à informação, e, quanto mais redundância, menor é a taxa de erro obtida.

3.2 - CÓDIGOS CORRETORES DE ERROS - CONCEITOS BÁSICOS

Neste item são apresentadas definições básicas sobre códigos corretores de erros, o princípio de decodificação e parâmetros para avaliação de desempenho de sistemas codificados.

Dependendo de como a redundância é adicionada aos bits de informação, pode-se ter dois tipos de códigos, a seguir definidos.

Definição 3.1 - Códigos de Bloco

Seja q o número de símbolos distintos empregados em um canal. Um código de bloco $C(q; n, M, d)$ é um conjunto de M seqüências de símbolos de canal de comprimento n . Estas n -uplas q -árias são chamadas palavras-código de cardinalidade $M = q^k$; onde k é o número de símbolos de informação, $(n-k)$ são símbolos de redundância e d é a distância mínima do código.

Para códigos binários q é igual a dois. Para a codificação, a seqüência contínua de informação deve ser dividida em blocos de k bits. A redundância permite verificar a existência ou não de erros num bloco particular de tamanho n e efetuar a sua eliminação.

Definição 3.2 - Código Convolucional

Um código convolucional difere de um código de bloco no sentido de que seu codificador contém memória e as N saídas do codificador, em um dado instante de tempo, não somente dependem das M entradas naquele instante, mas, também, dos \mathcal{L} blocos de entrada anteriores.

Neste tipo de código a redundância permite verificar a existência ou não de erros em mais de um bloco. Quando M é igual a um, a seqüência de informação não é dividida em blocos e pode ser processada continuamente.

Neste trabalho são considerados apenas códigos de bloco. Os códigos de bloco lineares representam parte bem desenvolvida da teoria e são definidos a seguir.

Definição 3.3 - Código Linear [BEN 88]

Um código C é dito ser linear se, e somente se :

- a) Os símbolos de canal formam um Grupo Abeliano [R&W 85] sobre a operação chamada "soma" e denotada por "+".
- b) O código C é um Homomorfismo [R&W 85] dos símbolos de informação nos símbolos de canal, isto é, dado o mapeamento C , para todo $u_1, u_2 \in U$, $C(u_1 + u_2) = C(u_1) + C(u_2)$.

O processo de codificação pode ser representado pelo produto $v = U \cdot G$, onde G é a Matriz Geradora do código, definida a seguir juntamente com a Matriz de Verificação de Paridade H .

Definição 3.4 - Matriz Geradora

A Matriz Geradora G de um código de bloco linear é a matriz de dimensões $k \times n$, cujas k linhas são palavras-código linearmente independentes. Um código linear sistemático binário é especificado pela seguinte matriz :

$$G = \begin{bmatrix} & & & \vdots & j_{1,1} & \dots & j_{1,n-k} \\ & I_k & & \vdots & & & \\ & & & \vdots & j_{k,1} & \dots & j_{k,n-k} \end{bmatrix} \quad (3.1)$$

onde J_{ij} é a sub-matriz de paridade, e I_k é a matriz identidade.

Definição 3.5 - Matriz de Verificação de Paridade

A Matriz de Verificação de Paridade H de um código de bloco é a matriz de dimensões $(n-k) \times n$, com $(n-k)$ linhas linearmente independentes, tal que o espaço de linha de G seja ortogonal a H ; isto é, se v é uma palavra-código em G , então $v \cdot H^T = 0$. Se a Matriz Geradora de um código está na sua forma sistemática, então a matriz H toma a forma :

$$H = \begin{bmatrix} J^T & : & I_{n-k} \end{bmatrix} = \begin{bmatrix} j_{1,1} \dots j_{k,1} & & \vdots & \\ & \ddots & & I_{n-k} \\ j_{1,n-k} \dots j_{k,n-k} & & \vdots & \end{bmatrix} \quad (3.2)$$

As características básicas dos códigos de bloco são definidas a seguir.

Definição 3.6 - Peso de Hamming

O Peso de Hamming de uma palavra-código é o número de posições não nulas da mesma.

Definição 3.7 - Distância de Hamming

A Distância de Hamming d_H , entre duas palavras-código de mesmo comprimento, é o número de posições onde as palavras diferem.

Definição 3.8 - Distância Mínima

A Distância Mínima d de um código é a menor distância encontrada entre suas palavras.

Definição 3.9 - Capacidade de Correção

A Capacidade de Correção t de um código binário é o número de posições que podem ser corrigidas em uma palavra-código, e é dada pela expressão :

$$t = \lfloor (d-1)/2 \rfloor \quad (3.3)$$

onde $\lfloor \cdot \rfloor$ significa "a parte inteira de".

A Síndrome, definida a seguir, fornece informação sobre os símbolos errados, e pode ser usada para correção de erros.

Definição 3.10 - Síndrome

Síndrome é a $(n-k)$ -upla calculada por :

$$\mathcal{S} = R \cdot H^T = (s_1, s_2, \dots, s_{n-k}) \quad (3.4)$$

onde R é uma palavra-código que foi alterada pelo ruído durante a transmissão, ou seja :

$$R = V + E$$

onde V é uma palavra-código e E é um padrão de erro. Se $\mathcal{S} = 0$, supõe-se que não ocorreram erros durante a transmissão.

As definições a seguir são importantes para os códigos definidos no espaço Euclidiano.

Definição 3.11 - Distância Euclidiana

No espaço Euclidiano define-se a Distância Euclidiana d_E entre duas seqüências de símbolos p_i e p_j , pela seguinte equação :

$$d_E^2(p_i, p_j) = \sum_n d_E^2(p_{in}, p_{jn}) \quad (3.5)$$

onde p_{in} e p_{jn} são símbolos de canal pertencentes às seqüências p_i e p_j respectivamente.

Definição 3.12 - Códigos Invariantes em Distância [M&S 77]

Um código C é dito ser Invariante em Distância se o número de palavras-código à distância D de uma palavra-código é independente da escolha desta palavra-código.

TEOREMA 1 - [M&S 77]

Um código linear é, também, um código invariante em distância.

Definição 3.13 - Códigos Invariantes Quanto à Região de Decisão [BAL 87]

Um código Invariante quanto à Região de Decisão é um código onde as regiões de decisão ótimas (máxima verossimilhança) são congruentes.

TEOREMA 2 - [BAL 87]

Um código Invariante quanto à Região de Decisão é, também, um código Invariante em Distância, mas o inverso não é sempre verdadeiro.

Definição 3.14 - Código Superlinear [BEN 88]

Um código C linear é dito superlinear se for possível associar, a cada símbolo de canal p, um número real w(p), chamado Peso do Símbolo, tal que :

$$d^2(p_i, p_j) = w(p_i + p_j)$$

É possível definir, para códigos superlineares, um peso de seqüência :

$$w(P) = \sum_i w(p_i) \quad (3.6)$$

tal que : $d^2(P, P') = w(P + P')$

Os pesos, assim definidos, generalizam os Pesos de Hamming dos códigos binários.

TEOREMA 3 - [BEN 88]

Um código linear é dito ser superlinear, se, e somente se, a seguinte condição é válida :

$$p_i + p_j = p_k + p_l \Rightarrow d^2(p_i, p_j) = d^2(p_k, p_l) \quad (3.7)$$

Apresentam-se, agora, alguns conceitos básicos sobre álgebra finita, necessários ao Capítulo 4.

Definição 3.15 - Grupo

Um Grupo A é uma estrutura algébrica que possui a operação chamada "adição" que satisfaz as seguintes propriedades :

- a) Fechamento : $\forall x_1, x_2 \in A \rightarrow x_1 + x_2 \in A;$
- b) Associatividade :

$$\forall x_1, x_2, x_3 \in A \rightarrow x_1 + (x_2 + x_3) = (x_1 + x_2) + x_3;$$

- c) Identidade : Existe o elemento nulo $0 \in G$, tal que $\forall x \in A \rightarrow x + 0 = 0 + x = x;$
- d) Inverso : Existe o elemento inverso $-x \in G$, tal que $\forall x \in A \rightarrow x + (-x) = (-x) + x = 0.$

Se o grupo satisfaz à propriedade comutativa, é chamado Grupo Abeliano.

Definição 3.16 - Anel

Um Anel Z é uma estrutura algébrica que possui duas operações chamadas "adição" e "multiplicação". O anel é um Grupo Abeliano sobre a adição e satisfaz às propriedades de Fechamento, Associatividade e Distributividade. A adição é sempre comutativa em um anel, mas a multiplicação pode não ser. Existe o inverso aditivo e o inverso multiplicativo pode existir [M&S 77].

Definição 3.17 - Campo

Um Campo F é uma estrutura algébrica que possui duas operações chamadas "adição" e "multiplicação". O campo é um Grupo Abeliano sobre a adição; a multiplicação obedece à propriedade do Fechamento; o conjunto de elementos diferentes de zero forma um Grupo Abeliano sobre a multiplicação. Existem o inverso aditivo e o inverso multiplicativo [M&S 77].

Definição 3.18 - Anel Polinomial

O conjunto de todos os polinômios $f(x)$ sobre o campo F, forma um Anel Polinomial Z. O polinômio $f(x)$ com

coeficientes em Z_n é :

$$f(x) = f_0 + f_1 \cdot x + \dots + f_n \cdot x^n$$

O grau de $f(x)$ é a potência de mais alta ordem. A adição e a multiplicação entre polinômios são fornecidas pelas equações 3.8 e 3.9, respectivamente.

$$f(x) + g(x) = \sum_{i=0}^n (f_i \oplus g_i) \cdot x^i \quad (3.8)$$

$$f(x) \cdot g(x) = \sum_{i=0}^n \sum_{j=0}^n (f_i \otimes g_j) \cdot x^{i+j} \quad (3.9)$$

Um polinômio $f(x)$ de grau m é dito polinômio primitivo, se, e somente se, ele divide $x^n + 1$ para nenhum n menor que $2^m - 1$.

Definição 3.19 - Campo de Galois

Um campo finito F com q elementos, onde q é primo, é chamado Campo de Galois e é denotado $GF(q)$. Para um q primo elevado a uma potência m , $GF(q^m)$ é um campo de extensão de $GF(q)$. Cada $GF(q)$ tem, ao menos, um elemento primitivo α . Cada elemento no campo, exceto zero, pode ser expresso como uma potência de α . O conjunto $GF(2^m) = \{0, 1, \alpha, \alpha^2, \dots, \alpha^{2^m-2}\}$ é um campo de Galois de 2^m elementos.

As adições e multiplicações definidas em $GF(q)$ envolvem a representação de seus elementos na forma polinomial e o uso das equações 3.8 e 3.9, respectivamente, com \oplus e \otimes representando adição e multiplicação módulo 2, neste caso, respectivamente. A construção de um $GF(2^m)$ requer um polinômio primitivo $f(x)$ de grau m (obtido de tabelas) e um elemento primitivo α , que é uma raiz de $f(x)$.

Exemplo : Determine os elementos de GF(4) sobre GF(2) gerados pelo polinômio primitivo $f(x) = 1 + x + x^2$, e as tabelas de adição e multiplicação entre os elementos.

Solução : O elemento primitivo α é raiz do polinômio primitivo; então :

$$f(x) = 1 + x + x^2 \rightarrow 0 = 1 + \alpha + \alpha^2 \rightarrow \alpha^2 = 1 + \alpha$$

Assim, os elementos de GF(4) estão representados na Tabela 3.I.

TABELA 3.I - REPRESENTAÇÃO DOS ELEMENTOS DE GF(4)

Representação em potência	Representação polinomial	Representação em m-uplas
0	0	00
1	1	10
α	α	01
α^2	$1 + \alpha$	11

As tabelas de adição e multiplicação são :

+	0	1	α	α^2		x	0	1	α	α^2
0	0	1	α	α^2		0	0	0	0	0
1	1	0	α^2	α		1	0	1	α	α^2
α	α	α^2	0	1		α	0	α	α^2	1
α^2	α^2	α	1	0		α^2	0	α^2	1	α

3.3 - PRINCÍPIO DE DECODIFICAÇÃO

Supondo um canal sem memória e palavras com a mesma probabilidade de serem transmitidas, seja $v \in C$ uma palavra-código transmitida e R a palavra recebida, a decodificação

por máxima probabilidade "a posteriori" decodifica R na palavra-código \hat{v} , onde :

$$\hat{P}(\hat{v}/R) = \max_{v \in C} \{P(v/R)\}$$

Sabendo que : $P(v/R) = \frac{P(R/v) \cdot P(v)}{P(R)}$

E assumindo $P(v)$ constante, o decodificador de máxima verossimilhança ou MLD ("Maximum Likelihood Decoder") é aquele que possui, como regra de decisão, a seguinte equação:

$$\hat{P}(\hat{v}/R) = \max_{v \in C} \{P(R/v)\} \quad (3.10)$$

Para o canal binário simétrico, o decodificador MLD é idêntico àquele que escolhe \hat{v} como a palavra-código v que minimiza a distância entre R e v [L&C 83].

Esta regra de decisão, a princípio, só pode ser aplicada se todas as palavras-código do código podem ser comparadas com a palavra recebida R . Isto quer dizer que, para códigos de bloco que não possuam um comprimento relativamente pequeno, a quantidade de correlações necessárias torna o processo proibitivo.

3.4 - DECISÃO ABRUPTA E SUAVE

Um sistema de decodificação é dito de decisão abrupta, quando o demodulador passa ao decodificador somente o símbolo recebido. Em um sistema com decisão suave, além do símbolo recebido, uma informação sobre a confiabilidade do símbolo, também, é levada ao decodificador. Esta informação de confiabilidade, passada ao decodificador, melhora o desempenho do sistema em cerca de 2 dB em relação à decisão abrupta [C&C 81]. O decodificador escolhe a palavra-código que minimiza a distância entre as palavras-código e a palavra

recebida. No caso da decisão suave, a distância adequada é a distância Euclidiana.

3.5 - PARÂMETROS PARA AVALIAÇÃO DE DESEMPENHO

O desempenho de um sistema de comunicação digital é, geralmente, avaliado baseando-se na Probabilidade de Erro de Bit, também chamada Taxa de Erro de Bit (TEB). Um sistema é considerado o melhor, entre dois, se este apresentar a menor Taxa de Erro de Bit para a mesma potência transmitida.

Um sistema de Modulação Codificada é, geralmente, comparado a um sistema não-codificado, ambos com mesmas restrições de faixa, mesma Energia Média de Sinal e o mesmo número de bits de informação por símbolo.

O ganho de decodificação e os gráficos TEB \times E_b/N_0 , também, são importantes para avaliação de sistemas combinados de modulação e codificação (sistemas codimodulados).

Definição 3.20 - Ganho de Codificação

O Ganho de Codificação é a diferença em dB entre os valores das relações sinal-ruído E_b/N_0 necessárias para atingir a mesma Taxa de Erro de Bit, para o sistema não-codificado e o codificado respectivamente; isto é :

$$g = \left(\frac{E_b}{N_0} \right)_{n-cod} - \left(\frac{E_b}{N_0} \right)_{cod} \quad [dB] \quad (3.11)$$

O ganho assintótico g_∞ é o ganho de codificação quando a relação E_b/N_0 tende ao infinito :

$$g_\infty = 10 \cdot \log_{10} \left(\left(\frac{dE^2}{E_c} \right)_{cod} \div \left(\frac{dE^2}{E_c} \right)_{n-cod} \right) \quad [dB] \quad (3.12)$$

onde dE é a distância Euclidiana mínima e E_c é a Energia Média do sinal (equação 2.3) para a modulação codificada e não-codificada.

Definição 3.21 - Energia de Bit

Para uma mensagem de comprimento finito de k bits de informação e energia média E_m , a Energia de Bit é definida por :

$$E_b = E_m / k \quad (3.13)$$

Somente os bits de informação são usados para calcular E_b ; os demais símbolos não representam informação transmitida. Para uma mensagem de taxa constante e comprimento infinito, de R' bits de informação por segundo, E_b é dada por :

$$E_b = E_c / R' \quad (3.14)$$

onde E_c é a Energia Média de transmissão (equação 2.3).

A mensagem é afetada pelo ruído Gaussiano de densidade espectral unilateral de N_0 Watt/Hertz. A recepção não pode ser afetada se ambos, sinal e ruído, dobrarem de valor. Somente a razão E_b/N_0 afeta a Taxa de Erro de Bit [BLA 84]. Assim, os esquemas de Modulação Codificada e não-codificada podem ser comparados por seus respectivos gráficos de TEB \times E_b/N_0 .

3.6 - MODULAÇÃO CODIFICADA - HISTÓRICO

Num sistema de comunicação digital não-codificado, baseado numa constelação bidimensional de 2^m pontos, pode-se transmitir m bits/símbolo. Um meio lógico para melhorar o desempenho destes sistemas é o desenvolvimento de constelações ótimas. Campopiano e Glazer [C&G 62] propuseram constelações formadas a partir da seleção de 2^m pontos

próximos pertencentes a um arranjo denso, regular e infinito de pontos. Mais tarde, Forney "et al." [FOR 84] provaram que as melhores distribuições para os pontos de uma constelação se encontram sob a forma circular, já que estas figuras apresentam a menor energia média para uma dada área, sendo o arranjo hexagonal o mais denso conhecido.

Com a introdução da codificação nos sistemas de comunicação digital, seu desempenho foi melhorado em detrimento à taxa de transmissão de informação.

A codificação é obtida pela adição de redundância, de modo a maximizar a distância entre as seqüências de símbolos de canal, diminuindo a taxa de transmissão de informação. Para compensar esta perda pode-se usar dois procedimentos : aumentar a taxa de modulação, se o canal permite uma expansão de faixa, ou aumentar o conjunto de sinais do sistema de modulação, se o canal é limitado em faixa. Esta última opção é mais segura no sentido de trazer mais confiabilidade, manter a taxa de informação e adaptar-se às limitações de faixa e potência normalmente encontradas [BAL 89].

A idéia da Modulação Codificada data da origem da Teoria da Informação, uma vez que Shannon não fez distinção entre a modulação e a codificação em seu trabalho [SHA 48, 69]. Mais tarde, Massey [MAS 74] mostrou que a função correta da modulação é criar um canal discreto, o mais adequado possível, ao processo de codificação, propondo que os processos fossem considerados como sendo um só.

Recentemente, Ungerboeck [UNG 82] sistematizou a Modulação Codificada, descrevendo técnicas de codificação que permitem encontrar ganhos significativos sobre a modulação multinível não-codificada, com a utilização de códigos convolucionais. Os resultados, alcançados por Ungerboeck, incentivaram a pesquisa no sentido de utilizar códigos de bloco associados a esquemas de modulação, área esta sistematizada por Sayegh [SAY 86].

A seguir são apresentadas noções importantes para a compreensão da Modulação Codificada.

3.6.1 - ALFABETOS REDUNDANTES

Nos sistemas de Modulação Codificada, a idéia fundamental, para se transmitir m bits de informação por símbolo de canal, é mapear estes m bits em um alfabeto de $Q > 2^m$ formas de onda, de modo que a redundância necessária ao processo de codificação é fornecida pelo aumento no número de símbolos de canal. Este alfabeto é conhecido por alfabeto redundante. Esta redundância é explorada com a escolha apropriada de 2^m símbolos de canal requeridos, baseando-se nas características do codificador de modo a maximizar a distância Euclidiana entre seqüências codificadas que é a chave para os ganhos de codificação trazidos por este processo. Esses ganhos podem ser alcançados sem energia adicional, mantendo a mesma taxa de transmissão de informação e largura de faixa do sinal.

3.6.2 - CAPACIDADE DE CANAL DE SISTEMAS CODIMODULADOS

Ungerboeck [UNG 82] mostrou como escolher a expansão mais adequada para o conjunto de símbolos de canal, de maneira a obter o desempenho desejado, e os limitantes de ganho para os esquemas de modulação mais comuns.

Supondo um canal livre de interferência intersimbólica, mas limitado em faixa e com ruído Aditivo Gaussiano Branco, a Capacidade de Canal com entrada discreta multinível e saída contínua versus a relação sinal-ruído para este canal é mostrada na figura 3.2, para vários tipos de constelações em duas dimensões.

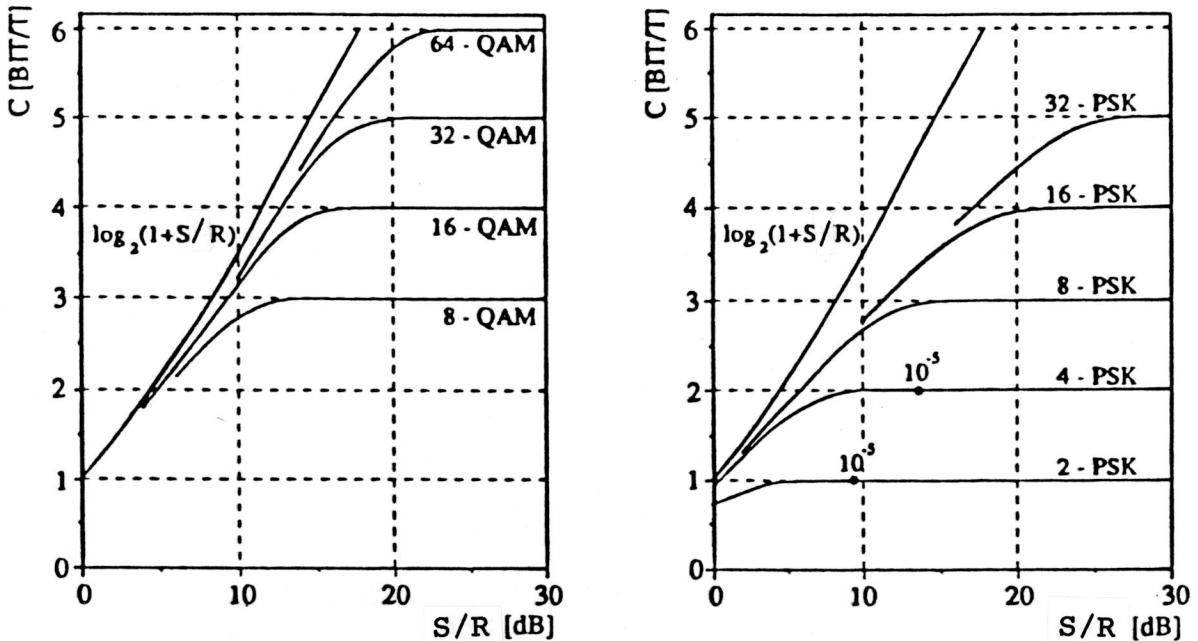


FIGURA 3.2 – CAPACIDADE DE CANAL PARA CANAIS LIMITADOS EM FAIXA PARA MODULAÇÃO BIDIMENSIONAL

Analisando as curvas, observa-se que, enquanto a transmissão de 1 bit/T com probabilidade de erro de símbolo $P_e = 10^{-5}$ pela modulação 2-PSK é alcançada com uma relação sinal-ruído de 9,5 dB, o uso da modulação 4-PSK para transmitir 1 bit/T sem erro é, em teoria, possível com relação sinal-ruído de aproximadamente 0,5 dB com esforço ilimitado de codificação e decodificação. Praticamente obteve-se todo o ganho possível, já que com o aumento indefinido da constelação, apenas 0,5 dB poderia ainda ser obtido. Generalizando, a expansão de uma constelação por um fator de dois permite obter a maior parte do ganho em capacidade de canal.

3.6.3 - PARTIÇÃO DE CONJUNTO

As técnicas de codificação convencionais não podem ser aplicadas em conjunto com técnicas de modulação com limite de faixa, ao menos com ganho significante [FOR 84], ou seja, codificação e modulação operando independentemente são ineficientes. Na codificação independente, interessa ter-se distâncias de Hamming maximizadas entre as palavras-código; mas este processo não necessariamente maximiza as distâncias Euclidianas entre as seqüências de símbolos que são transmitidas pelo canal (apenas em casos triviais). Na Modulação Codificada, a distância Euclidiana deve ser maximizada e, para isto, um processo de mapeamento adequado, da informação codificada aos pontos da constelação expandida, deve ser utilizado.

Ungerboeck [UNG 82] desenvolveu uma regra de partição de conjunto para tornar o mapeamento uma tarefa sistemática. Um conjunto de símbolos de uma constelação é sucessivamente particionado em subconjuntos com distâncias mínimas cada vez maiores $\Delta_0 < \Delta_1 < \Delta_2 \dots$ entre os símbolos destes subconjuntos. A distribuição dos símbolos nos subconjuntos é simétrica e a partição segue um esquema de árvore binária. Como exemplo, o particionamento do conjunto de sinais da modulação 8-PSK é mostrado na figura 3.3.

3.7 - MODULAÇÃO CODIFICADA COM CÓDIGOS DE TRELIÇA

Este item apresenta a modulação codificada com códigos de treliça (TCM), que é aplicada a modem's de alta velocidade, como por exemplo o modem da Recomendação V.32 do CCITT.

Os códigos de treliça podem ser descritos com um diagrama de transição de estados similar aos diagramas de treliça dos códigos convolucionais binários. A diferença reside no fato de que, na modulação codificada com códigos de

treliça, os braços da treliça são rotulados com sinais de modulação redundantes ao invés de símbolos do código binário [BAL 89].

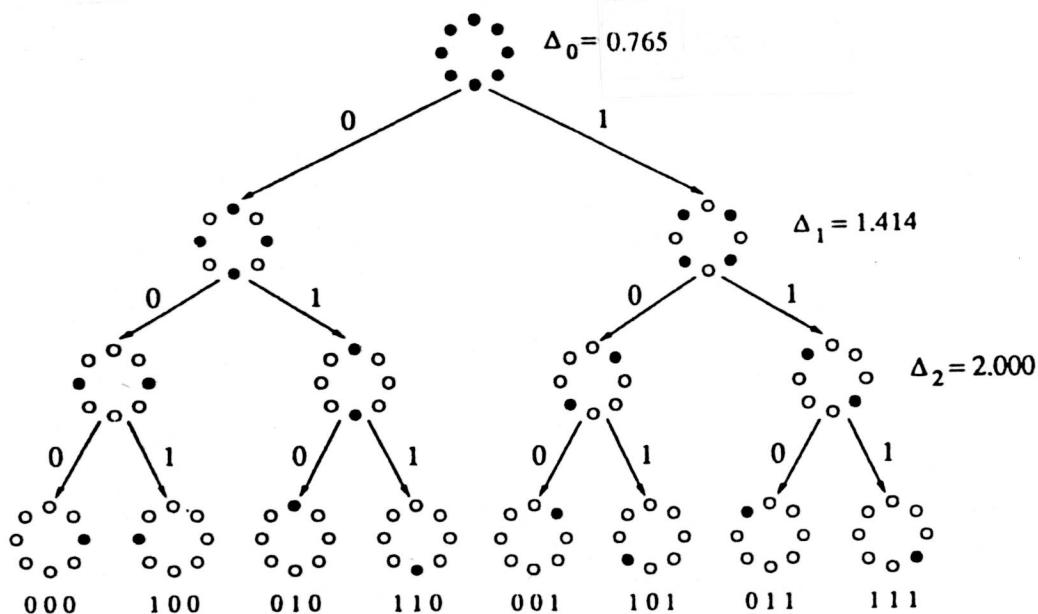


FIGURA 3.3 - PARTICIONAMENTO DA CONSTELAÇÃO 8-PSK

Como mostrado anteriormente, quase todo o ganho em capacidade de canal é obtido dobrando o número de sinais do espaço de sinais usado para a modulação sem codificação. Baseado nisto, necessitando-se transmitir m bit/T de forma redundante, é necessário um conjunto de 2^{m+1} sinais de canal. Então é fácil notar que a seqüência de informação binária de tamanho m pode ser codificada por um codificador convolucional de taxa $R = m/(m+1)$ e, posteriormente, o grupo de $(m+1)$ bits é mapeado para o conjunto expandido de sinais de canal.

A estrutura básica de um sistema TCM é mostrada na figura 3.4 e consiste em um codificador convolucional binário e um sistema que realiza o mapeamento da saída do codificador aos símbolos do espaço de sinais.

No codificador convolucional, um grupo de m bits é aplicado a uma máquina de estados finita, sendo codificados em $(m+1)$ bits, num intervalo de T segundos, de acordo com a lógica combinacional do codificador e em função de ℓ bits armazenados. O codificador pode assumir 2^ℓ estados. O mapeador é uma função biunívoca entre o grupo de $(m+1)$ bits e os símbolos do espaço de sinais.

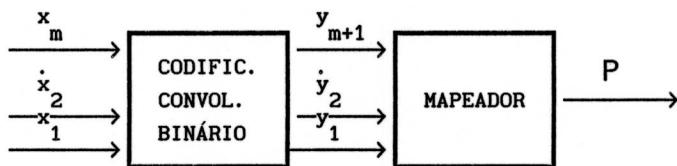


FIGURA 3.4 - ESTRUTURA DO SISTEMA TCM

No projeto do sistema, um diagrama de transição de estados adequado deve ser definido e a atribuição dos 2^{m+1} símbolos de canal da constelação expandida para as transições deve ser realizada de modo a encontrar a máxima distância Euclidiana. Para a realização desta atribuição Ungerboeck [UNG 82] desenvolveu o "Mapeamento por Partição de Conjuntos" em que a associação das transições, entre estados com os subconjuntos terminais obtidos do particionamento, segue as seguintes regras :

- a) Todos os sinais da constelação devem ser usados no diagrama de transição de estados com mesma freqüência e simetria;
- b) Transições que partem de ou chegam a um mesmo estado recebem sinais dos subconjuntos de modo a garantir a maior distância Euclidiana entre todas estas transições;
- c) Transições paralelas (que partem de um estado e chegam a este estado em instante futuro) recebem sinais dos subconjuntos de maior distância Euclidiana.

A regra a) reflete a intuição de que bons códigos devem exibir estruturas regulares, embora Divsalar [DIV 87] tenha mostrado que, com uma disposição não uniforme de símbolos em uma constelação, é possível obter ganhos de codificação em relação a disposições simétricas. As regras b) e c) garantem o ganho sobre a modulação não-codificada de referência devido à maior distância Euclidiana conseguida entre seqüências de símbolos.

As transições paralelas são o limitante para o ganho e, por isto, recebem sinais dos subconjuntos de maior distância Euclidiana. Esta condição implica uma diminuição da probabilidade de ocorrência de eventos de erro unitário na treliça. Este tipo de erro não é desejável, pois com a decodificação da seqüência recebida através do algoritmo de Viterbi [VIT 67], um erro ocorrido durante um quadro de transições decorrente de uma transição paralela não é suficiente para que o algoritmo identifique o sobrevivente correto entre todos os sobreviventes [UNG 82].

Pode ser demonstrado que a distância Euclidiana mínima do código obtido é dada por [BEN 88] :

$$dE_{\min} = \min (dp, dc) \quad (3.15)$$

onde : dp é a distância Euclidiana entre transições paralelas;
dc é a distância mínima entre caminhos não paralelos no diagrama de treliça do código.

A observação destas regras, melhorando a distância Euclidiana entre as seqüências de símbolos de canal, propicia ganho de codificação ao sistema. Deve-se notar que, ao se realizar o mapeamento, especifica-se de maneira única o codificador convolucional.

Como exemplo, a figura 3.5 mostra transições do diagrama de treliça para a modulação 8-PSK codificada com

código de treliça de quatro estados, com taxa $R = 2/3$. O exemplo ilustra a aplicação das regras de Ungerboeck. Assim, associa-se às transições paralelas dos estados 00 e 11 sinais dos subconjuntos de maior distância Euclidiana (Δ_2), satisfazendo a regra c). As transições que partem dos estados, recebem sinais de subconjuntos de distância Euclidiana Δ_1 , satisfazendo a regra b). A regra a) é satisfeita, pois todos os sinais da constelação são usados duas vezes. Há várias alternativas para as atribuições; este exemplo é uma delas.

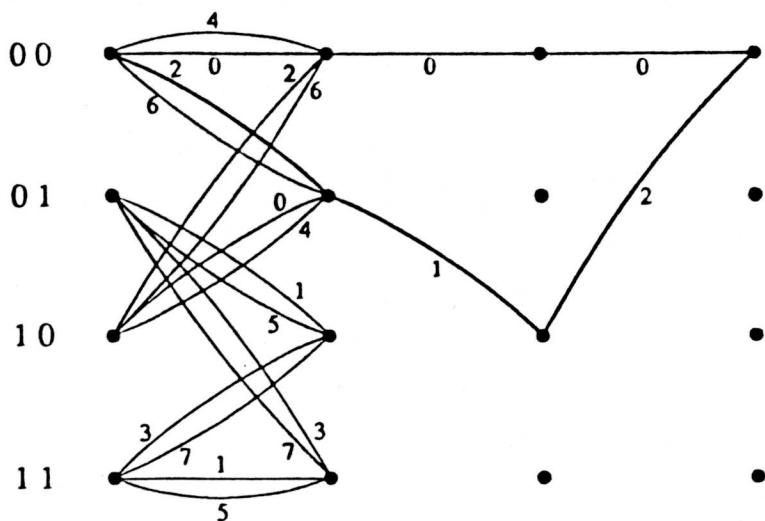


FIGURA 3.5 - DIAGRAMA DE TRELIÇA DE QUATRO ESTADOS

3.7.1 - CÓDIGOS CONVOLUCIONAIS INVARIANTES QUANTO À ROTAÇÃO DE FASE

Apesar dos ganhos obtidos com a codificação, ambigüidades de fase da constelação expandida podem resultar em rotações dos símbolos recebidos, se comparados aos símbolos transmitidos. Isto causa erros em todos os dados recebidos subsequentemente e degrada, seriamente, o desempenho do sistema.

Wei [WEI 84] aplica a técnica de codificação diferencial à codificação de canal para criar códigos transparentes às rotações dos símbolos de canal.

O espaço de sinais com ambigüidades de fase de 90° , 180° e 270° são, geralmente, preferidos em relação ao espaço de sinais com ambigüidades de 180° somente. A constelação 32-CR, mostrada na figura 3.6, apresenta esta característica e as seguintes :

a) Para a mesma potência média, a distância Euclidiana mínima, entre símbolos na constelação 32-CR, é maior que na 32-QAM;

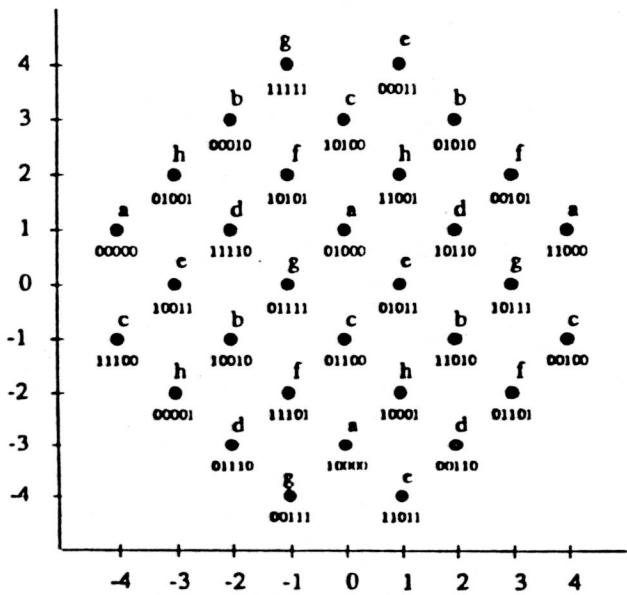


FIGURA 3.6 - CONSTELAÇÃO 32-CR

b) A razão de potência de pico por potência média para a constelação 32-CR é 1,7, enquanto para a 32-QAM é 2,33;

c) A razão de potência média por potência mínima para a constelação 32-CR é 10, enquanto para a 32-QAM é 21.

Para que os códigos convolucionais sejam feitos transparentes à rotação múltipla de 90° de símbolos de canal, algumas regras devem ser seguidas na obtenção do diagrama de

transição de estados e no mapeamento dos bits codificados para os símbolos. O codificador convolucional, obtido com estas regras, não é linear [WEI 84]. São elas :

REGRA 1 - Atribui-se a i , $i = 1, 2, \dots, z$, os estados de um codificador convolucional de z estados. Para cada transição do estado i para o estado j , denota-se F o conjunto de símbolos associados com esta transição e F_{90} , F_{180} e F_{270} os conjuntos de símbolos obtidos quando F é rotacionado 90° , 180° e 270° no sentido horário, respectivamente. O código pode ser feito transparente a rotações de 90° , 180° e 270° se existirem funções $f_l : \{1, 2, \dots, z\} \rightarrow \{1, 2, \dots, z\}$, $l = 1, 2, 3$, tal que : os conjuntos de símbolos associados com transições do estado $f_l(i)$ para o estado $f_l(j)$, $l = 1, 2, 3$, são F_{90} , F_{180} e F_{270} , respectivamente.

Se esta regra é satisfeita, então, para cada seqüência de símbolos válida p , a seqüência de símbolos obtida quando todos os símbolos em p são rotacionados pelo mesmo múltiplo de 90° , também, é válida. Esta regra garante que o código pode ser feito transparente à rotação dos símbolos de 90° , 180° e 270° .

Considerando a classe de codificadores convolucionais de taxa $m/(m+1)$, com realimentação e de 2^k estados, sendo o estado corrente denotado por $W_1 W_2 \dots W_k$, em que os bits de entrada Y_1, Y_2, \dots, Y_m são diretamente levados à saída do codificador, e que a constelação expandida é particionada em 2^{k+1} conjuntos de símbolos (onde k é um número inteiro maior que um), então a atribuição da combinação de bits de saída aos símbolos deve satisfazer as seguintes regras :

REGRA 2 - Os símbolos em cada um dos 2^{k+1} conjuntos são associados com a mesma combinação de valores dos bits Y_I , $I = 0, 1, \dots, k$. Ou seja, Y_I , $I = 0, 1, \dots, k$; definem os conjuntos do particionamento.

REGRA 3 - O conjunto de diferentes combinações dos bits Y_{In} , $I = 1, 2, \dots, k$, atribuídos aos conjuntos de símbolos associados com transições originárias do mesmo estado do codificador, devem assumir todas as possíveis combinações, ou seja, de cada estado partem 2^k transições.

REGRA 4 - Selecionam-se dois bits de saída de Y_{In} , $I = 1, 2, \dots, k$, sem perda de generalidade, diga-se Y_{1n} e Y_{2n} . Então, para cada conjunto de quatro símbolos que tenham o mesmo raio mas afastados de 90° , os quatro valores do par $Y_{2n} Y_{1n}$ associados com estes sinais são diferentes uns dos outros. Em adição, para cada conjunto dos quatro símbolos, disponha os valores associados do par $Y_{2n} Y_{1n}$ em uma seqüência de quatro elementos tal que os símbolos correspondentes ao segundo, terceiro e quarto valores desta seqüência possam ser obtidos rotacionando o símbolo correspondente ao primeiro valor desta seqüência de 90° , 180° e 270° no sentido horário, respectivamente.

Várias maneiras podem ser usadas para atribuir os bits de saída Y_{In} , $I = 3, 4, \dots, m$ aos símbolos. Simplificadamente, o mesmo valor de Y_{In} , $I = 3, 4, \dots, m$, é atribuído a cada conjunto de quatro símbolos definidos acima.

As ambigüidades de fase podem ser removidas codificando diferencialmente o par de bits de entrada, cujo par de saída associado é $Y_{2n} Y_{1n}$, antes que eles sejam aplicados ao codificador convolucional, e realizando operação inversa na saída do decodificador convolucional. A seqüência de quatro elementos do par de bits requerida na definição do codificador diferencial é qualquer uma das seqüências definidas na Regra 4.

Após a atribuição dos bits de saída do codificador aos símbolos, a tabela verdade do codificador convolucional pode ser obtida desta atribuição e do diagrama de transição de estados. As entradas da tabela verdade são as variáveis de estado correntes $W_{1n}, W_{2n}, \dots, W_{Ln}$ e os bits de entrada

correntes, cujos bits de saída associados são $y_{1_n}, y_{2_n}, \dots, y_{k_n}$. As saídas são as variáveis de próximo estado, $w_{1_{n+1}}, w_{2_{n+1}}, \dots, w_{L_{n+1}}$ e o bit de saída codificado y_{0_n} . A simplificação de equações booleanas leva ao circuito do codificador.

A figura 3.7 apresenta o circuito lógico do codificador convolucional não linear de oito estados, com realimentação, em constelação 32-CR, adotado como padrão internacional (V.32) para modem's de 9600 bit/s [WEI 84].

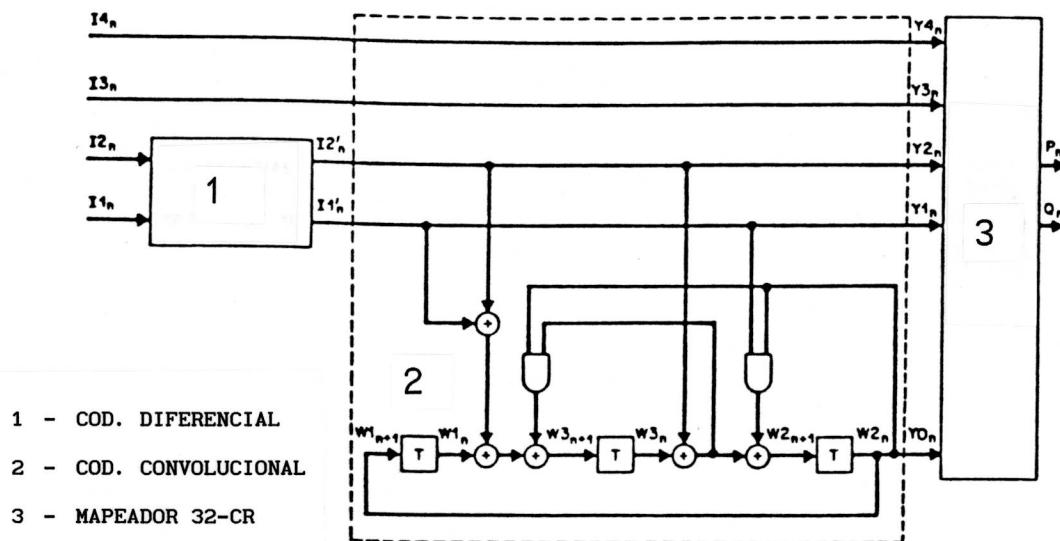


FIGURA 3.7 - CODIFICADOR DO MODEM V-32 DO CCITT

A característica apresentada de que os bits de entrada são diretamente levados à saída, é de interesse prático, pois o receptor tem a opção de não usar o decodificador convolucional se a degradação em desempenho for aceitável.

Uma proposta de implementação deste sistema padrão para modem's V.32 é apresentada por Messer [MES 89]. A codificação e a decodificação pelo Algoritmo de Viterbi [VIT 67] são implementados com o auxílio de um processador digital de sinais de arquitetura apropriada para realizar as funções

de modulador/demodulador e de codificador/decodificador. O "software" empregado para a codificação e decodificação é apresentado e pode ser modificado para trabalhar com qualquer diagrama de transição de estados, e não somente com o caso V.32.

O algoritmo de Viterbi é rápido o bastante para permitir a decodificação em tempo real de códigos com número de memórias (L) menor ou igual a oito [L&C 83], com processadores de alta velocidade; caso contrário o algoritmo não é vantajoso pela grande quantidade de memória para decodificação necessária.

A tendência da teoria em Modulação Codificada com códigos de Treliça está voltada para a utilização de constelações multidimensionais. Tanaka "et al." [TAK 87] desenvolveram um modem para o canal de voz de 19,2 Kbps baseado na codificação octodimensional em treliça.

A codificação multidimensional permite encontrar altos ganhos sem aumentar o número de pontos na constelação de sinais. Nesta técnica, os dados codificados com um bit redundante são mapeados em um sinal multidimensional consistindo de vários sinais bidimensionais consecutivos, reduzindo, assim, o tamanho da constelação de sinais. Isto reduz a degradação causada pelas constelações densas baseadas nos esquemas usuais de codificação bidimensional.

A figura 3.8 mostra a estrutura do codificador empregado neste modem de alta velocidade.

Os bits de informação para quatro intervalos de modulação são tratados como um simples bloco. A taxa de modulação é de 2743 baud e 28 bits são codificados a cada quatro intervalos de modulação. O sinal octodimensional formado é dividido em 16 subconjuntos, mantendo uma distância Euclidiana mínima de 16. Os quatro bits codificados pelo codificador sistemático de 16 estados identificam cada

subconjunto e os bits restantes definem um ponto de sinal no subconjunto.

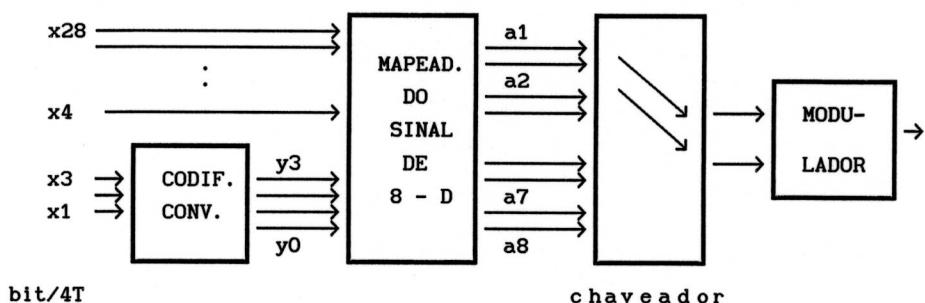


FIGURA 3.8 - ESTRUTURA DO CODIFICADOR

O sinal octodimensional ($a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8$) é transmitido como quatro sinais bidimensionais (a_1, a_2)... (a_7, a_8), cada um selecionado de uma constelação de 160 pontos. O algoritmo de Viterbi no receptor coleta quatro símbolos sucessivos e realiza cálculos de distância usando a distância Euclidiana octodimensional consistindo de quatro distâncias bidimensionais. O ganho de codificação sobre a modulação não codificada 128-QAM é de 5,4 dB. O ganho de uma codificação convencional 256-QAM é de 3,8 dB, ou seja, o uso da codificação octodimensional traz um ganho de 1,6 dB sobre esquemas bidimensionais.

3.8 - MODULAÇÃO CODIFICADA COM CÓDIGOS DE BLOCO - HISTÓRICO E PRINCIPAIS SISTEMAS

Na Modulação Codificada com códigos de Bloco (BCM) são usados códigos de bloco binários, em substituição aos códigos convolucionais, antes do mapeamento. Os sistemas BCM representam uma alternativa aos sistemas TCM.

Os desenvolvimentos nesta área iniciaram com Imai e Hirakawa [I&H 77] que mostraram um método de codificação multinível que usa vários códigos de bloco binários de modo a

melhorar o desempenho da modulação multinível. A decodificação é realizada por etapas, sendo que a seqüência estimada em uma etapa auxilia na decodificação da seqüência seguinte. A determinação das seqüências estimadas é feita através de probabilidades "a posteriori", baseadas na estatística do canal e na estrutura dos códigos.

A generalização deste trabalho foi realizada por Ginzburg [GIN 84] que particionou o conjunto de sinais de modulação em subconjuntos, combinando-os com códigos segundo uma hierarquia para garantir larga separação no espaço de sinais.

Cusack [CUS 84] desenvolveu um método para que os códigos de bloco pudessem ser associados com os sinais da modulação de modo que, na decodificação, demodulador e decodificador se auxiliassem mutuamente, a exemplo do que ocorre com os códigos de treliça. Em seu trabalho, Cusack criou arranjos-código, que são matrizes formadas pelas palavras-código dos códigos de bloco usados, e, usando códigos de Reed-Muller, montou códigos no espaço de sinais para constelações QAM.

Mais tarde, Sayegh [SAY 86] generalizou o trabalho de Cusack, mostrando que a estrutura pode ser construída usando ampla gama de códigos e constelações, tornando a construção mais poderosa e flexível. Sayegh provou matematicamente as propriedades de distância entre os arranjos-código, mostrando que os ganhos significativos (3 a 6 dB), obtidos por Ungerboeck para os códigos de treliça, podem ser obtidos com códigos de bloco e com a vantagem de poderem ser mais facilmente decodificados.

Uma abordagem totalmente nova de se fazer Modulação Codificada é apresentada por Baldini [BAL 89], com o uso de Códigos de Bloco Multiníveis. A seguir será feita uma síntese dos principais sistemas BCM.

3.8.1 - SISTEMA DE IMAI E HIRAKAWA

Imai e Hirakawa [I&H 77] propuseram um esquema de codificação multinível que usa vários códigos corretores de erro. Embora não tenham percebido, o esquema por eles criado mostra os princípios da modulação codificada aplicada a códigos de bloco. Este princípio consta da determinação dos símbolos a serem transmitidos a partir da combinação de bits das palavras-código pertencentes aos códigos usados. O uso da transmissão de símbolos multiníveis permite um aumento na velocidade de transmissão da informação.

O processo de decodificação é o que tornou o esquema bastante conhecido. Esse processo é realizado em etapas, de modo que uma seqüência estimada auxilia na determinação da próxima seqüência, e o processo se repete até que não existam mais seqüências a serem estimadas.

3.8.1.1 - DESCRIÇÃO DO SISTEMA

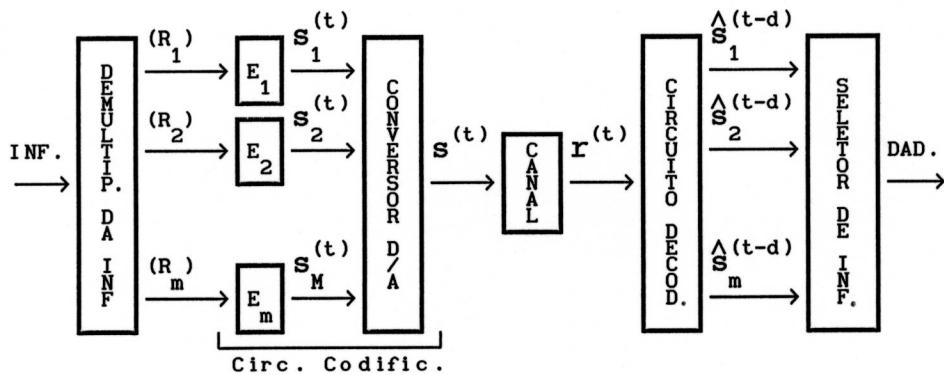
A figura 3.9 mostra o diagrama em blocos do sistema, que é a estrutura básica de todo sistema BCM. A seqüência de informação a ser transmitida é demultiplexada em m componentes de informação com taxas R_1, \dots, R_m . A i -ésima componente entra no codificador E_i , que gera uma palavra-código do código C_i . Um símbolo binário $s_i^{(t)}$, da palavra-código, deixa o codificador no instante de tempo t . O conversor Digital/Analógico DA calcula $s^{(t)}$ (um símbolo do alfabeto 2^m -ário) segundo a equação :

$$s^{(t)} = \sum_{i=1}^m s_i^{(t)} \cdot 2^{i-1} \quad (3.16)$$

Analizando a equação 3.16, observa-se que ela mapeia os bits das palavras-código que deixam os codificadores E_i em símbolos de uma constelação. A escolha apropriada dos códigos

C_i permite adaptar o sistema a canais de diferentes relações sinal-ruído.

O sinal recebido $r^{(t)}$ é tratado como analógico e convertido em digital com apropriado número de bits. O circuito decodificador fornece as estimativas finais $\hat{s}_i^{(t-d)}$, com um retardo de d unidades de tempo. Finalmente, os bits de paridade são retirados para formar os componentes de seqüência de informação e, a partir destes, constrói-se a seqüência de informação recebida final.



E_i : Codificador do Código Binário C_i com taxa R_i

FIGURA 3.9 - ESTRUTURA DO SISTEMA

3.8.1.2 - A DECODIFICAÇÃO

Omite-se, agora, o sobre-escrito t e representa-se a seqüência $\{\dots, x^{(t-1)}, x^{(t)}, x^{(t+1)}, \dots\}$ abreviadamente por $\{x\}$.

Na decodificação tenta-se estimar s_i quando s_1, \dots, s_{i-1} e r são dados, da seguinte forma :

Se $P_i(0/s_1 \dots s_{i-1} r) \geq P_i(1/s_1 \dots s_{i-1} r)$, então $\hat{s}_i = 0$; caso contrário $\hat{s}_i = 1$, onde \hat{s}_i é a estimativa de s_i .

Como no receptor não se conhece s_1, \dots, s_{i-1} , usam-se as estimativas finais $\hat{s}_1, \dots, \hat{s}_{i-1}$, já obtidas, em seu lugar.

Se $\hat{s}_1, \dots, \hat{s}_{i-1}$ são iguais a s_1, \dots, s_{i-1} com alta probabilidade, então a estimação usando $\hat{s}_1, \dots, \hat{s}_{i-1}$ e r é mais segura que usando r somente.

A figura 3.10 mostra o circuito de decodificação para $m = 4$ e a nomenclatura usada.

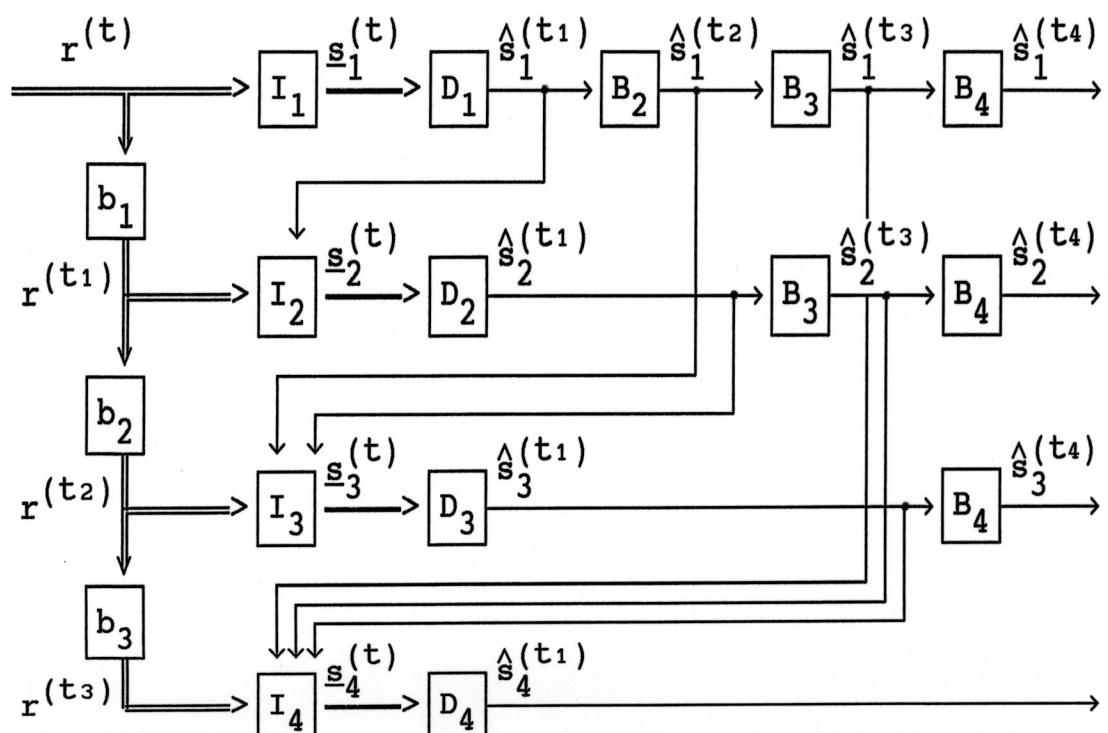
O procedimento de decodificação é descrito a seguir :

1º Passo - A estimativa intermediária \underline{s}_1 é calculada por I_1 usando as probabilidades "a posteriori" $P_1(0/r)$ e $P_1(1/r)$. O decodificador D_1 (para o código C_1) forma a seqüência estimada final $\{\hat{s}_1\}$ corrigindo os erros de $\{\underline{s}_1\}$.

i-ésimo Passo ($i = 2, \dots, m$) - A estimativa intermediária \underline{s}_i é calculada por I_i usando as probabilidades "a posteriori" $P_i(0/\hat{s}_1 \dots \hat{s}_{i-1} r)$ e $P_i(1/\hat{s}_1 \dots \hat{s}_{i-1} r)$. O decodificador D_i (para o código C_i), forma a seqüência estimada final $\{\hat{s}_i\}$ corrigindo os erros de $\{\underline{s}_i\}$.

Se o sistema for construído com m códigos de bloco de mesmo tamanho, com capacidades corretoras t_i , então elas devem obedecer a $t_1 \geq t_2 \geq \dots \geq t_m$, já que as probabilidades de erro das estimativas intermediárias $\underline{s}_1, \underline{s}_2, \dots, \underline{s}_m$ decrescem nesta ordem. A taxa final do sistema é dada por :

$$R = \sum_{i=1}^m R_i \quad (3.17)$$



$t_1 = t - d_1, t_2 = t_1 - d_2, t_3 = t_2 - d_3, t_4 = t_3 - d_4 = t - d;$

d_i : Retardo de tempo do decodificador para C_i ;

I_i : Circuito de estimação intermediária para S_i ;

D_i : Decodificador para C_i ;

B_i, b_i : Buffer de d_i unidades de tempo;

\Rightarrow : Caminho de transmissão (CT) de sinais analógicos (ou apropriado número de caminhos de transmissão de sinais digitais);

\rightarrow : CT para estimativas intermediárias (CT binário digital para decisão abrupta, ou CT não-binário digital para decisão suave);

\longrightarrow : CT binário digital.

FIGURA 3.10 - CIRCUITO DECODIFICADOR PARA $m = 4$

Sistemas de transmissão de dados eficientes podem ser obtidos estabelecendo-se apropriadamente o compromisso entre capacidade de correção e taxa dos códigos usados. A decodificação é considerada sub-ótima devido à sua realização em etapas, ou seja, a decodificação processada por D_1 é a

afetará as demais. A decisão suave pode ser efetivamente implementada, melhorando o desempenho da decodificação.

A seguir são apresentados os códigos para a Modulação Codificada com Códigos de Bloco obtidos por Sayegh.

3.8.2 - CÓDIGOS DE SAYEGH

Sayegh [SAY 86] mostrou de forma clara, em seu trabalho, como determinar uma classe de códigos de bloco binários, de modo que a distância Euclidiana entre os arranjos-código formados fosse maximizada.

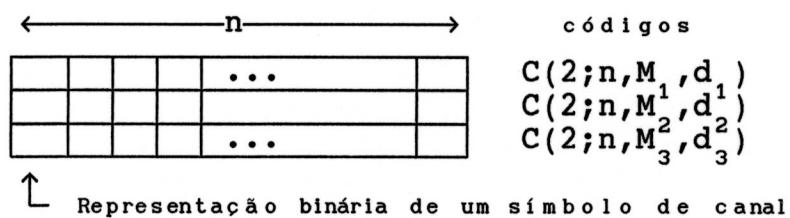
Desta forma ele conseguiu obter ganhos de codificação semelhantes àqueles alcançados por Ungerboeck para códigos convolucionais. Sayegh propôs, também, um procedimento de decodificação para seu sistema, que é assintoticamente (relação sinal-ruído $\rightarrow \infty$) ótimo.

3.8.2.1 - A CODIFICAÇÃO

O assunto será desenvolvido ilustrando o sistema com um exemplo específico. A generalização segue facilmente.

Considerando a modulação 8-PSK, Sayegh desejava encontrar códigos binários necessários à construção de um código ótimo no espaço de sinais.

Esse código ótimo é formado pela disposição de palavras-código dos códigos binários conforme mostra a figura 3.11 (generalização do trabalho de Cusack [CUS 84]) de modo a formar os denominados arranjos-código, que, no caso, possuem três linhas por n colunas (o número de linhas é o logaritmo na base dois do número de pontos da constelação).

FIGURA 3.11 - ARRANJO-CÓDIGO DE 3 LINHAS E n COLUNAS

A cada ponto da constelação são atribuídos dígitos binários de acordo com o particionamento de conjuntos de Ungerboeck (figura 3.3).

O arranjo-código é transmitido coluna a coluna. Cada coluna representa um ponto na constelação, de modo que um bit, na primeira linha, corresponde ao bit mais à direita na constelação.

Sayegh determinou uma equação que relaciona as distâncias de Hamming de cada código usado com a distância Euclidiana entre dois arranjos-código formados.

Considere dois arranjos-código que diferem apenas na primeira palavra-código; observando-se a constelação, ver-se-á que os arranjos-código devem estar distanciados de no mínimo Δ_0 (distância Euclidiana) para cada bit distinto na palavra-código binária. Como a distância de Hamming entre palavras-código é d_1 , ter-se-á uma distância Euclidiana entre arranjos-código dada por :

$$D^2 \geq \Delta_0^2 \cdot d_1$$

Caso se considerar os dois arranjos-códigos diferindo apenas na segunda palavra-código, observa-se que, segundo o particionamento de conjunto (figura 3.3), eles estarão distanciados de no mínimo Δ_1 para cada bit distinto da palavra binária. Assim :

$$D^2 \geq \Delta_1^2 \cdot d_2$$

Para apenas a terceira palavra-código diferente, tem-se :

$$D^2 \geq \Delta_2^2 \cdot d_3$$

Finalmente, a distância Euclidiana mínima entre os arranjos-código, considerando a constelação 8-PSK, deve satisfazer a :

$$dE_{\min}^2 \geq \min (\Delta_0^2 \cdot d_1, \Delta_1^2 \cdot d_2, \Delta_2^2 \cdot d_3) \quad (3.18)$$

Busca-se, então, em tabelas de códigos de bloco [M&S 77], aqueles códigos que tornem máxima a equação 3.18, fazendo com que $d_1 \geq d_2 \geq d_3$. O processo de codificação, proposto por Sayegh, protege os bits "mais fracos" que são transportados por meio de cada símbolo de canal.

3.8.2.2 - A DECODIFICAÇÃO

O processo de decodificação de Sayegh partitiona todos os possíveis arranjos-código em M_1 conjuntos, todos com a mesma palavra-código na primeira linha. Procura, então, selecionar um dos M_1 conjuntos através de comparação com o arranjo-código recebido. O conjunto selecionado é, agora, partitionado em M_2 conjuntos, todos de mesmas palavras-código nas primeira e segunda linhas. Nova comparação com o arranjo-código recebido leva à determinação do arranjo-código decodificado.

A probabilidade de decodificação correta do arranjo-código, neste caso, é dada por :

$$P_a = P(l_1) \cdot P(l_2/l_1) \cdot P(l_3/l_2, l_1)$$

onde $P(l_1)$ é a probabilidade de decodificação correta da primeira linha do arranjo-código; $P(l_2/l_1)$ é a probabilidade

dado que a primeira linha foi corretamente decodificada e $P(l_3/l_2, l_1)$ é a probabilidade de decodificação correta da terceira linha, dado que a primeira e segunda linhas foram decodificadas corretamente. No primeiro passo da decodificação, todos os conjuntos estão distanciados de $dE^2 \geq \Delta_0^2 \cdot d_1 \geq dE_{\min}^2$ uns dos outros. Similarmente, em um dado conjunto, os subconjuntos estão distanciados de $dE^2 \geq \Delta_1^2 \cdot d_2 \geq dE_{\min}^2$ uns dos outros; e, finalmente, para um subconjunto particular, todos os sinais de canal estão distanciados de $dE^2 \geq \Delta_2^2 \cdot d_3 \geq dE_{\min}^2$ uns dos outros.

Então, a alta relação sinal-ruído, em que a distância Euclidiana é o maior fator que afeta a TEB, este processo de decodificação é ligeiramente inferior ao MLD. Entretanto, à relação sinal-ruído moderada, o número de vizinhos próximos (arranjos-código semelhantes) pode ser substancialmente maior que o número no caso de decodificação MLD, e uma alta degradação é esperada; assim o processo de decodificação é assintoticamente ótimo, ou seja, aproxima-se do desempenho do decodificador de máxima verossimilhança (MLD) quando a relação sinal-ruído tende para infinito. A complexidade do algoritmo de decodificação é proporcional a $2^{\sum k_i}$.

Usando-se códigos curtos de repetição e de paridade simples, pode-se construir esquemas com ganhos de até 3 dB sobre esquemas sem codificação, com igual taxa de transmissão. Com códigos BCH (devidos a Bose, Chaudhuri e Hocquenghem) pode-se obter ganhos de 6-7 dB, conforme mostra a tabela 3.II [SAY 86].

Os códigos no espaço de sinais, assim obtidos, apresentam ganhos semelhantes aos obtidos por Ungerboeck com códigos convolucionais. A desvantagem é o uso de vários decodificadores para códigos de bloco diferentes.

TABELA 3.II - CÓDIGOS DE SAYEGH

Constelação	Taxa	n	g_∞ /Ref.	Códigos
4-PSK	1/2	4	3/ 2-PSK	Repetição + Cheque
8-PSK	2/3	7	3/ 4-PSK	
16-PSK	3/4	8	3/ 8-PSK	
4-PSK	1/2	16	6,0/ 2-PSK	BCH
4-PSK	1/2	32	7,8/ 2-PSK	
8-PSK	1/3	32	6,7/ 2-PSK	
8-PSK	2/3	32	6,0/ 4-PSK	

3.8.3 - SISTEMA DE DECODIFICAÇÃO DE WILLIAMS E FARRELL

Williams e Farrell [W&F 88] mostraram um meio de reduzir a complexidade de decodificação aplicando-o aos esquemas concebidos por Cusack e Sayegh, ou seja, esquemas que usam os arranjos-código.

Nestes esquemas a codificação é realizada em etapas, ou seja, a informação a ser transmitida é particionada e, então, cada porção é codificada por um dos códigos binários de modo a formar cada uma das linhas do arranjo-código. Isto acarreta a necessidade de, também, se decodificar em etapas, tornando o processo sub-ótimo.

3.8.3.1 - A DECODIFICAÇÃO

O sistema de decodificação proposto por Williams e Farrell faz uso de uma medida de distância, para a constelação associada, que é fácil de manipular. Assim, uma simples saída do demodulador é capaz de fornecer as informações de distância necessárias ao processo

Numa primeira proposta, envolvendo constelações PSK, a idéia é transformar a representação bidimensional do ponto recebido em uma representação unidimensional, facilitando o cálculo de distâncias entre os pontos e o processo de decodificação.

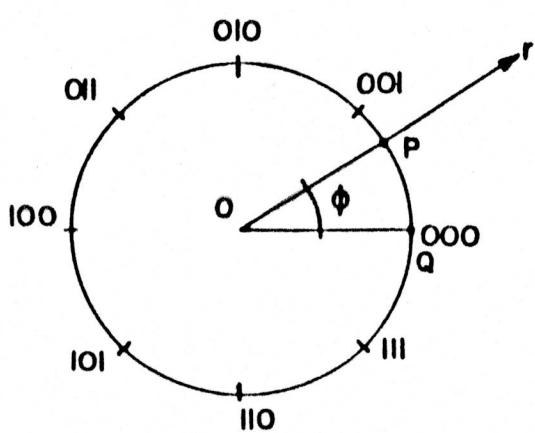


FIGURA 3.12 - PONTO RECEBIDO NA CONSTELAÇÃO 8-PSK

Na figura 3.12, considere-se que o círculo formado pela união dos pontos da constelação 8-PSK tem comprimento igual a oito, e que a cada ponto está associado um número inteiro compreendido no intervalo $[0,8)$.

O ponto recebido r é, então, representado por sua intersecção com o círculo (ponto P) e pelo ângulo ϕ . Como o ponto P sempre estará afastado da origem pelo valor do raio $R = 8/2\pi$, pode-se representar o ponto r apenas pelo ângulo ϕ , transformando a representação bidimensional em unidimensional.

Uma simples regra de três, permite associar o ângulo ϕ com uma medida de distância - de intervalo $[0,8)$ - da seguinte forma :

$$d_r = \frac{8 \cdot \phi_r}{360^\circ} \quad (3.19)$$

onde : d_r = distância na representação unidimensional do ponto recebido r ;
 ϕ_r = ângulo representativo do ponto recebido r .

Esta técnica pode ser aplicada a todas as constelações PSK, embora, na medida em que se dobra o número de pontos da constelação, deve-se aumentar a relação sinal-ruído de 5 a 6 dB para transmitir com o mesmo desempenho [PRO 83]. Por isso, para 16 ou mais pontos, a modulação QAM é preferida; mas, devido à sua geometria, não apresenta as mesmas propriedades da PSK.

Para constelações QAM, como segunda proposta, Farrell e Williams representaram a constelação bidimensional por constelações unidimensionais. Eles usaram o fato de que uma constelação QAM quadrada de 2^{2m} pontos pode ser vista como uma constelação AM de 2^m pontos cruzada com ela mesma.

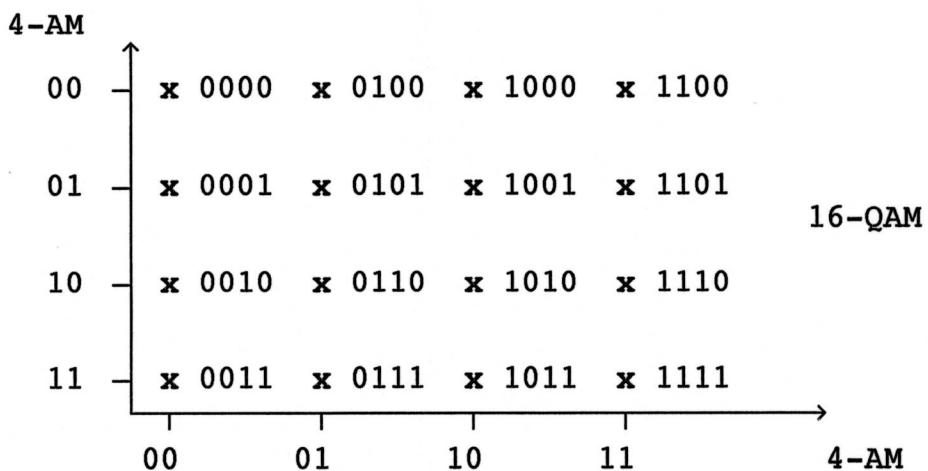


FIGURA 3.13 - 16-QAM COSTRUIÓDA DE DUAS 4-AM

A figura 3.13 mostra uma constelação 4-AM cruzada com ela mesma de modo a formar uma 16-QAM.

O que se faz, então, é codificar (e decodificar na recepção) duas constelações AM separadamente e usar a constelação QAM para a modulação. Assim mantém-se a vantagem que é o uso da modulação QAM e a simplicidade da codificação e decodificação associadas à constelação AM.

Exemplo [BAL 89] : Considere-se um sistema de modulação codificada em 8-QAM (figura 2.2) com $\Delta_0 = \sqrt{2}$ e arranjo-código formado por um código de repetição $C(2;8,2^1,8)$, um código de paridade simples $C(2;8,2^7,2)$ e uma linha não codificada. A taxa total do código formado é $R = \sum k / \sum n = (1+7+8)/(3 \times 8) = 2/3$. O ponto mais à esquerda e abaixo na constelação 8-QAM corresponde às coordenadas $X = 0$ e $Y = 0$. Suponha-se que a seqüência de sinais recebidos seja transformada para :

$$\begin{aligned} X &= (0,4; 0,2; 1,8; 2,3; 0,9; 2,1; 0,3; 2,6) \\ Y &= (2,3; 2,6; 2,1; 0,4; 2,1; 1,6; 1,2; 2,1) \end{aligned}$$

e passadas ao decodificador, que determina as seqüências de inteiros tudo-par e tudo-ímpar mais próximas da seqüência recebida acima, passo esse devido ao código de repetição usado. As seqüências tudo-par são $X = (0,0,2,2,0,2,0,2)$ e $Y = (2,2,2,0,2,2,2,2)$ com distância total igual a 6,3 da seqüência real recebida; e as seqüências tudo-ímpar são $X = (1,1,1,3,1,3,1,3)$ e $Y = (3,3,3,1,3,1,1,3)$ com distância igual a 10,1. Devido ao código de paridade simples, a soma destes números deve ser divisível por quatro. A soma dos pontos pares de X e Y é 22 e a soma dos pontos ímpares é 32. Já que a soma dos pontos pares não é divisível por quatro, deve-se alterar as coordenadas menos confiáveis quanto à distância em relação à seqüência real recebida. O quinto ponto na seqüência tudo-par X é o menos confiável e é alterado para 2, formando $X = (0,0,2,2,2,2,0,2)$ com nova distância igual a 6,5.

Desde que $6,5 < 10,1$ as seqüências finais consideradas são as tudo-par $X = (0,0,2,2,2,2,0,2)$ e $Y = (2,2,2,0,2,2,2,2)$.

Sempre se busca uma maneira de reduzir a complexidade de decodificação para qualquer tipo de código. Para códigos de bloco, Williams e Farrell encontraram esta redução em complexidade com o uso de uma medida de distância que procura conservar a estrutura de distância no espaço bidimensional representando-a em uma dimensão; mas com isto perde-se informação sobre a amplitude do erro tornando a decodificação sub-ótima.

3.8.4 - SISTEMA DE FORNEY

Forney "et al." [FOR 84] mostraram como formar reticulados de N dimensões e como fazer o mapeamento de bits de informação para os pontos destes reticulados. Trata-se de um esquema de modulação codificada, em que é dada ênfase à construção destes reticulados densos usando reticulados bidimensionais, particionamento de conjunto e códigos corretores de erro. Neste trabalho são apresentados apenas aspectos relacionados ao particionamento de conjunto, apresentado por Forney e o uso deste no Esquema Geral de modulação codificada.

3.8.4.1 - PARTICIONAMENTO DE CONJUNTO

Forney "et al." dividem uma dada constelação bidimensional em dois subconjuntos, atribuindo pontos alternados para cada subconjunto, de acordo com a figura 3.14.

Os subconjuntos resultantes (A e B) apresentam as seguintes propriedades :

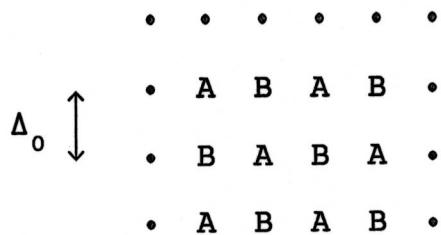


FIGURA 3.14 - PARTICIONAMENTO DA CONSTELAÇÃO BIDIMENSIONAL

a) Os pontos em cada subconjunto recaem em uma grade retangular com rotação de 45° em relação à grade original.

b) A distância mínima ao quadrado entre pontos em um subconjunto é duas vezes a distância mínima ao quadrado (Δ_0^2) entre pontos da constelação original.

Este particionamento pode ser repetido para 4,8,... subconjuntos obtendo distâncias ao quadrado de 4,8,... vezes Δ_0^2 .

3.8.4.2 - ESQUEMA GERAL DE MODULAÇÃO CODIFICADA

Os subconjuntos assim determinados são usados para implementar simples e eficientes esquemas de codificação, segundo o esquema mostrado na figura 3.15 [FOR 84].

Esta figura representa o Esquema Geral de Modulação Codificada, pois o codificador binário pode ser tanto para códigos convolucionais como para códigos de bloco [FOR 84].

Do conjunto de bits de informação a serem transmitidos, selecionam-se alguns que são codificados por um codificador binário. Os bits, assim codificados, são usados como seletores de subconjuntos. Os bits remanescentes não são codificados e selecionam pontos do subconjunto selecionado.

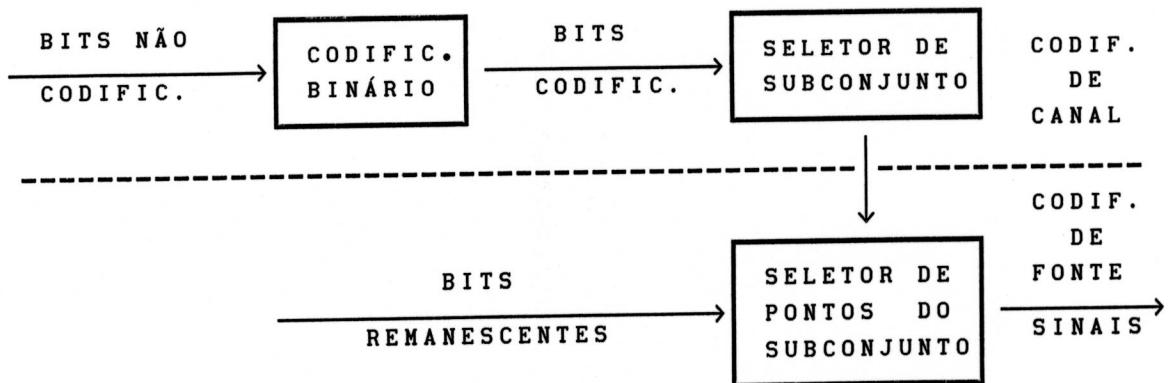


FIGURA 3.15 - ESQUEMA GERAL DE MODULAÇÃO CODIFICADA

A constelação escolhida deve ser grande o bastante para acomodar todos os bits entrantes. Como o esquema de codificação é desacoplado da escolha da constelação, pode-se decompor o Esquema Geral de modulação codificada em Codificação de Fonte (cuja função é representar a saída de uma fonte de dados por uma seqüência de dígitos binários) e Codificação de Canal.

3.8.5 - SISTEMA DE TANNER

Tanner [TAN 87] apresenta um meio para a construção de conjuntos de sinais para sistemas de modulação codificada que nada mais é do que uma extensão da construção de códigos corretores de erro.

Esta analogia é importante, pois os conhecimentos técnicos já estudados no desenvolvimento de códigos corretores de erro são estendidos com facilidade à construção de sistemas de modulação codificada.

3.8.5.1 - CONSTRUÇÃO RECURSIVA DE CÓDIGOS

A base para a construção de conjuntos de sinais para a modulação codificada, segundo Tanner, é a construção recursiva de códigos corretores de erro.

A figura 3.16 mostra o esquema proposto por Tanner para a construção de códigos de Reed-Muller de comprimento dezesseis.

O funcionamento do sistema é mostrado com um exemplo. Para formar o código $C(2;16,2^{11},4)$ de Hamming, o esquema é composto por quatro codificadores E_A , $E_{B,1}$, $E_{B,2}$ e E_C para os códigos $C(2;4,2^4,1)$, $C(2;4,2^3,2)$, $C(2;4,2^3,2)$ e $C(2;4,2,4)$, recebendo quatro, três, três e um bits respectivamente, num total de 11 bits de informação.

Cada codificador fornece uma palavra de quatro bits, sendo que cada bit é levado a um dos quatro circuitos de mapeamento conforme a figura 3.16.

Os quatro bits que chegam ao circuito de mapeamento, formam o vetor de indexação $I = [i_4 \ i_{2,1} \ i_{2,2} \ i_1]$ que multiplica a matriz G pela direita, fornecendo um vetor de canal. A união dos quatro vetores de canal forma uma palavra-código do código $C(2;16,2^{11},4)$ de Hamming. A matriz G escolhida é :

$$G_{4,1} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.20)$$

que é uma matriz geradora para o código $C(2;4,2^4,1)$ de distância mínima de Hamming igual a um.

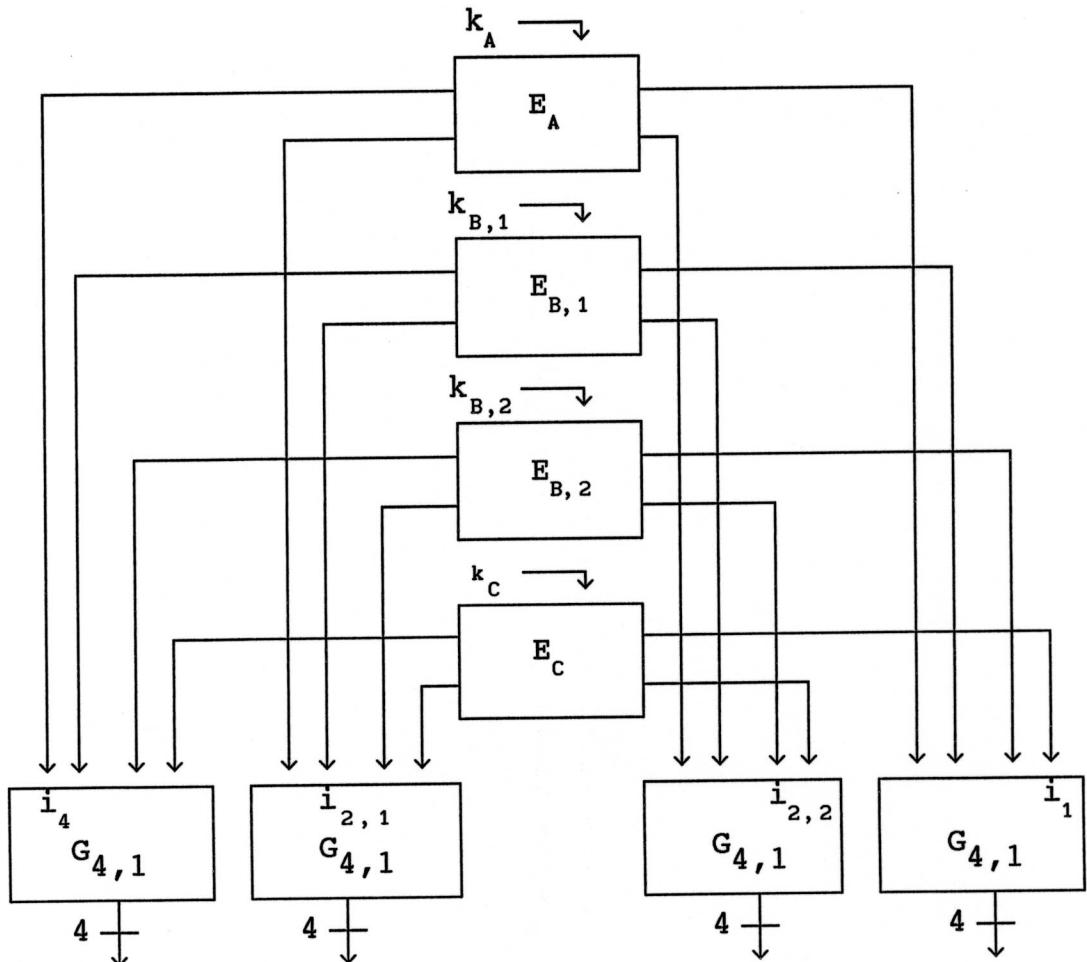


FIGURA 3.16 - CODIFICADOR PARALELO PARA CÓDIGOS DE REED-MULLER DE COMPRIMENTO 16

Ela é escolhida de forma a se obter a seguinte propriedade: considerem-se os vetores de indexação que diferem apenas em i_4 , então os vetores de canal, formados após a multiplicação $I \cdot G_{4,1}$, diferem em quatro bits. Possuem, então, distância de Hamming igual a quatro. Por exemplo, o vetor [1101] é mapeado para (0100) e o vetor [0101] é mapeado para (1011), distanciando quatro de (0100).

Se os vetores de indexação diferem em $i_{2,1}$ e/ou $i_{2,2}$, então os vetores de canal estão ao menos distanciados de dois e, se diferem em i_1 , estão distanciados de um.

A prova da distância mínima do código $C(2;16,2^{11},4)$ formado apóia-se nessas propriedades e na distância mínima dos códigos escolhidos para os codificadores E_A , $E_{B,1}$, $E_{B,2}$ e E_c [TAN 87].

A construção dos códigos pode ser recursiva, ou seja, a partir dos códigos de comprimento 16 pode-se formar códigos de comprimento 64 e assim por diante.

As propriedades mostradas exibem a idéia central para a construção de conjuntos de sinais com a desejada distância Euclidiana mínima. O objetivo do projeto de conjuntos de sinais é criar sinais que estão largamente separados no espaço Euclidiano para maximizar a probabilidade de que o sinal recebido caia na região associada com o sinal que foi realmente transmitido.

Finalmente, segundo Tanner, para serem construídos sistemas combinados de modulação/codificação, tudo o que se tem a fazer é trocar os circuitos de mapeamento por circuitos moduladores de propriedades análogas para a modulação que está sendo usada. Para a decodificação, Tanner usou um algoritmo similar ao apresentado por Imai e Hirakawa com decodificadores sub-ótimos de múltiplas iterações, que para um exemplo específico obteve ganhos de codificação de aproximadamente 6,5 dB sobre um sistema não-codificado 8-PSK.

3.8.6 - SISTEMA DE BALDINI

O sistema apresentado por Baldini [BAL 89] representa uma completa alteração quanto à filosofia da Modulação Codificada, realizando primeiro o mapeamento da seqüência binária de informação no conjunto expandido de sinais de canal e, em seguida, codificando-os para introduzir redundância aumentando assim a distância Euclidiana mínima. O alfabeto de canal multinível é considerado como um anel de inteiros módulo q , onde q é o número de sinais do alfabeto

expandido. O conjunto de seqüências de informação e o conjunto de seqüências codificadas, também, são considerados como anéis. Baldini apresenta duas classes de códigos de bloco multiníveis obtidos por busca computacional, que permitem ganhos de codificação de 3 a 6 dB.

3.8.6.1 - A CODIFICAÇÃO

A figura 3.17 mostra a estrutura do codificador apresentado :

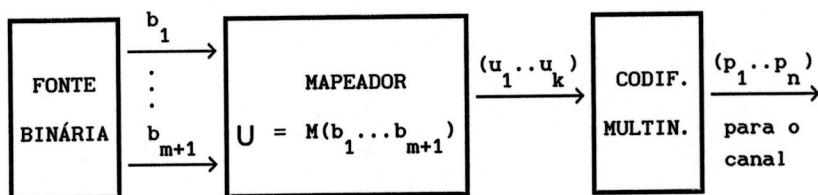


FIGURA 3.17 - ESTRUTURA DO CODIFICADOR

Realiza-se primeiro o mapeamento (que será apresentado em seguida) da seqüência binária de informação no conjunto expandido de sinais de canal e, em seguida, a codificação.

A saída paralela da fonte binária é composta por $m+1$ bits de informação ($b_1 \dots b_{m+1}$) que são mapeados em um dos $q = 2^{m+1}$ símbolos u_i pertencentes ao conjunto Z_q de sinais de modulação. Z_q é um anel de inteiros módulo- q , $Z_q = \{0, 1, 2, \dots, q-1\}$. A entrada do codificador multinível é a seqüência $U = (u_1, \dots, u_k)$ de k elementos de Z_q . A seqüência U é codificada pelo codificador multinível transformando-se na seqüência $P = (p_1, \dots, p_n)$, com $n \geq k$, também de elementos de Z_q , que são levados ao canal.

Segundo Baldini, as constelações mais adequadas à codificação multinível de elementos de um anel de inteiros módulo- q são as q -PSK, pois a distribuição espacial dos símbolos da modulação q -PSK obedece às mesmas regras que definem os anéis de inteiros módulo- q [BAL 89].

Em todo sistema de modulação codificada, o uso do mapeamento por partição de conjunto é imprescindível. Embora Baldini não fale em mapeamento por partição de conjunto em seu método, este está associado à estrutura dos códigos multiníveis.

3.8.6.2 - DISTÂNCIA EUCLIDIANA ENTRE SINAIS q-PSK

A distância Euclidiana entre dois sinais de uma determinada modulação q-PSK é diretamente relacionada aos símbolos correspondentes do anel de inteiros módulo-q atribuídos àqueles sinais de modulação. No esquema geral de modulação q-PSK, os sinais de canal são representados por :

$$p_i = \exp\left(j \frac{2\pi \cdot i}{q}\right), \quad i \in \mathbb{Z}_q \quad (3.21)$$

onde os elementos correspondentes do anel são dados pelos índices i .

A distância Euclidiana ao quadrado entre dois sinais de canal p_i e p_r é dada por :

$$\begin{aligned} dE^2(p_i, p_r) &= |p_i - p_r|^2 = \left| \exp\left(j \frac{2\pi \cdot i}{q}\right) - \exp\left(j \frac{2\pi \cdot r}{q}\right) \right|^2 = \\ &= \left| \exp\left(j \frac{2\pi \cdot (i \ominus r)}{q}\right) - 1 \right|^2 \end{aligned}$$

onde \ominus significa subtração módulo q .

Os índices i e r realizam uma operação definida pelos elementos i e r no anel de inteiros módulo q , assim :

$$dE^2(i, r) = \left| \exp\left(j \frac{2\pi \cdot (i \ominus r)}{q}\right) - 1 \right|^2$$

O Peso Euclidiano ao quadrado é definido como :

$$W_E^2(i) = dE^2(i, 0) = \left| \exp\left(j \frac{2\pi \cdot i}{q}\right) - 1 \right|^2$$

desta forma : $dE^2(i, r) = W_E^2(i \oplus r)$ (3.22)

3.8.6.3 - CÓDIGOS DE BLOCO MULTINÍVEIS

Os códigos de bloco multiníveis $C(q; n, M, dE_{min}^2)$ são definidos pela matriz geradora G de dimensões $k \times n$, de elementos pertencentes ao anel de inteiros módulo q . O codificador multinível da figura 3.17, realiza a operação $p = \cup \cdot G$, onde o vetor \cup é a seqüência de informação e p é a palavra-código multinível.

O problema é determinar a matriz G de modo a maximizar a distância Euclidiana mínima entre todos os pares de palavras-código.

Baldini determinou duas classes de códigos multiníveis ótimos em distância Euclidiana por busca computacional, restringindo a busca para alguns subconjuntos de n -uplas definidas por uma matriz geradora G específica.

3.8.6.4 - CÓDIGOS DE BLOCO LINEARES CIRCULANTES SISTEMÁTICOS

Um código de bloco linear circulante sistemático (BLCS) sobre Z_q é definido pela matriz geradora da equação 3.23 [BAL 89].

É necessário apenas conhecer a primeira linha da matriz, $(g_{11} \ g_{12} \ \dots \ g_{1,n-k})$, para definir completamente o código. As tabelas 3.III e 3.IV mostram alguns códigos BLCS para modulação codificada em 4-PSK e 8-PSK, respectivamente.

$$G = \begin{bmatrix} & & & g_{11} & g_{12} & \cdots & g_{1,n-k} \\ & I & & g_{1,n-k} & g_{11} & \cdots & g_{1,n-k-1} \\ & & \vdots & & & & \vdots \\ k & & & g_{12} & g_{13} & \cdots & g_{11} \end{bmatrix}^n \quad (3.23)$$

O ganho de codificação assintótico é dado por [BAL 89] :

$$g_{\infty} = 10 \cdot \log_{10} \left(\frac{\log q_c}{\log q} R_c \frac{dE_c}{dE} \right) \quad [\text{dB}] \quad (3.24)$$

onde q_c e q são a cardinalidade do alfabeto de modulação codificada e não-codificada, respectivamente; dE_c e dE são as distâncias Euclidianas mínimas para a modulação codificada e não-codificada, respectivamente; e R_c é a taxa de codificação para a modulação codificada.

O sistema de modulação não-codificada de referência escolhido é aquele que transmite o mesmo número de bits de informação por símbolo e que tem a mesma Potência Média de sinal. Assim, na tabela 3.III, o ganho de codificação é calculado em relação à modulação 2-PSK não-codificada com $R_c = 1/2$, e, na tabela 3.IV, em relação à modulação 4-PSK não codificada.

3.8.6.5 - CÓDIGOS PSEUDO-CÍCLICOS MULTINÍVEIS

A segunda classe de códigos, determinada por Baldini, é chamada de códigos pseudo-cíclicos multiníveis devido à sua estrutura matricial e propriedades semelhantes às dos códigos binários cíclicos. Embora haja estas semelhanças, deslocamentos cíclicos de uma palavra-código de um código pseudo-cíclico não resultam, necessariamente, em novas palavras-código [BAL 89].

TABELA 3.III - CÓDIGOS BLCS DEFINIDOS SOBRE Z₄

n	Código	dE _{min} ²	g _∞ [dB]
4	12	8	3,0
10	12233	12	4,7
16	11202303	16	6,0

TABELA 3.IV - CÓDIGOS BLCS DEFINIDOS SOBRE Z₈

n	Código	dE _{min} ²	g _∞ [dB]
6	411	5,76	3,3
10	62221	7,51	4,5
12	773676	7,76	4,6

Esses códigos são códigos de bloco lineares definidos pela matriz geradora G com elementos pertencentes ao anel de inteiros módulo-q.

$$G = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1r} & 0 & 0 & \cdots & 0 \\ 0 & g_{11} & g_{12} & \cdots & g_{1r} & 0 & \cdots & 0 \\ \vdots & & & & & & & \vdots \\ 0 & \cdots & & & g_{11} & g_{12} & \cdots & g_{1r} \end{bmatrix}^n \quad (3.25)$$

onde r depende da taxa de codificação e da modulação q-PSK em uso.

Um código pseudo-cíclico qualquer pode ser representado pela primeira linha de sua matriz geradora G. As tabelas 3.V e 3.VI mostram alguns códigos pseudo-cíclicos para modulação codificada em 4-PSK e 8-PSK, com taxas 1/2 e 2/3, respectivamente.

TABELA 3.V - CÓDIGOS PSEUDO-CÍCLICOS DEFINIDOS SOBRE Z_4

n	código	dE_{\min}^2	g_{∞} [dB]
4	1210	8	3,0
10	3231230000	12	4,7
16	3100112010000000	14	5,4

TABELA 3.VI - CÓDIGOS PSEUDO-CÍCLICOS DEFINIDOS SOBRE Z_8

n	código	dE_{\min}^2	g_{∞} [dB]
6	127000	2,343	0,69
9	564500000	3,172	2,00
12	523670000000	3,515	2,45

3.8.6.6 - PROPRIEDADES DOS CÓDIGOS PSEUDO-CÍCLICOS MULTINÍVEIS

As propriedades mais importantes são :

- a) A matriz de paridade H é obtida a partir da matriz geradora G transformada para a sua forma sistemática G'.

Exemplo : Seja o código pseudo-cíclico em Z_4 , representado por 13023000.

A matriz na forma sistemática G' de G é :

$$G' = \begin{bmatrix} 1 & 0 & 0 & 0 & 2 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 3 \\ 0 & 0 & 1 & 0 & 3 & 2 & 1 & 3 \\ 0 & 0 & 0 & 1 & 3 & 0 & 2 & 3 \end{bmatrix} = [I:P]$$

A matriz de paridade H obtida de G' é :

$$H = \begin{bmatrix} 2 & 3 & 1 & 1 & 1 & 0 & 0 & 0 \\ 3 & 3 & 2 & 0 & 0 & 1 & 0 & 0 \\ 3 & 3 & 3 & 2 & 0 & 0 & 1 & 0 \\ 3 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} = [-P^T:I]$$

- b) Os códigos pseudo-cíclicos são invariantes em distância [M&S 77].
- c) Os códigos pseudo-cíclicos são invariantes quanto à região de decisão [BAL 87].
- d) Os códigos pseudo-cíclicos são superlineares [BEN 88].

3.8.6.7 - CÓDIGOS PSEUDO-CÍCLICOS SOBRE Z_4 PARA MODULAÇÃO 16-QAM CODIFICADA

Usando uma partição do espaço de sinais 16-QAM (figura 3.18) juntamente com códigos pseudo-cíclicos multiníveis definidos sobre Z_4 , pode-se obter ganhos significativos sobre a modulação 8-QAM não-codificada [bal 89].

A cada sinal da modulação 16-QAM é atribuído um par de símbolos pertencentes ao anel de inteiros módulo-4. O símbolo à esquerda seleciona um dos subconjuntos e o símbolo à direita seleciona um dos pontos no subconjunto escolhido. Pode-se, então, usar dois códigos definidos sobre Z_4 com capacidades corretoras diferentes, sendo que o mais potente protege os símbolos à esquerda, que seleciona um dos subconjuntos. Forma-se, então, um arranjo-código multinível.

Os símbolos à esquerda, protegidos pelo código mais potente, possuem probabilidade maior de decodificação correta e selecionam com maior segurança um dos subconjuntos. Como a distância Euclidiana entre os pontos pertencentes a um subconjunto é maior que a distância Euclidiana entre pontos do conjunto inicial, pode-se utilizar um código de menor poder de correção, mas de maior taxa, para proteger os pontos do subconjunto.

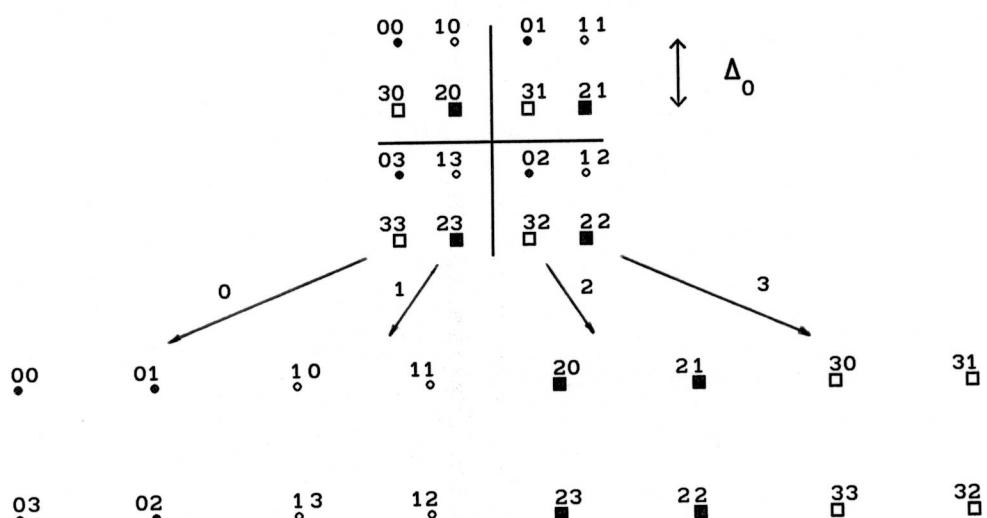


FIGURA 3.18 - PARTIÇÃO DA CONSTELAÇÃO 16-QAM EM 4 CONJUNTOS

A tabela 3.VII mostra alguns conjuntos de códigos adequados à modulação 16-QAM, em que o ganho é calculado considerando a menor distância Euclidiana mínima dos códigos utilizados, em relação à modulação 8-QAM não-codificada. A distância Euclidiana do código da primeira linha do arranjo-código é calculada levando-se em consideração a distância entre pontos no conjunto inicial (16-QAM). Para o código da segunda linha, considera-se a distância entre pontos dos subconjuntos.

TABELA 3.VII - CÓDIGOS PARA 16-QAM

Códigos	Matriz	g_{∞} sobre 8-QAM
$C(4;4,4^2,4)$ $C(4;4,4^4,4)$	1230 não-codificado	3,0
$C(4;14,4^8,6)$ $C(4;14,4^{13},8)$	30120230000000 paridade simples	4,7

3.8.6.8 - COMPARAÇÃO COM O SISTEMA DE SAYEGH

Comparando o sistema de Baldini com o sistema de Sayegh, pode-se levantar os seguintes pontos :

a) Baldini restringe a busca de códigos a uma estrutura definida por uma matriz multinível, encontrando códigos com uma ampla gama de comprimentos e ganhos com taxas fixas de 1/2 e 2/3. Sayegh busca códigos binários curtos de comprimentos fixos de maneira a formar códigos no espaço de sinais de distância Euclidiana maximizada, encontrando códigos com ganhos de 3 a 6 dB e de diversas taxas.

b) Os códigos encontrados por Baldini para a modulação 4-PSK codificada possuem equivalentes encontrados por Sayegh, como por exemplo, o primeiro e o quarto da tabela 3.II com o primeiro e o terceiro da tabela 3.III, respectivamente. Para a modulação 8-PSK codificada, o ganho dos códigos de Baldini é inferior aos equivalentes de Sayegh, como por exemplo, o segundo da tabela 3.II comparado ao primeiro da tabela 3.VI.

c) Os códigos de Baldini apresentam algumas propriedades; mas estas, aparentemente, não são suficientes para a determinação de um processo simples de decodificação para o sistema. Os códigos de Sayegh, por serem formados por códigos binários simples, são decodificados facilmente passo-a-passo, embora de uma forma sub-ótima. Devido à estrutura

imposta por Sayegh, propriedades de reticulados podem ser obtidas com a escolha dos códigos usados, como o realizado inteligentemente por Williams e Farrell [W&F 88].

Os códigos de bloco multiníveis apresentados por Baldini, definidos sobre um anel de inteiros módulo- q , são um sistema alternativo para encontrar ganhos de codificação significativos para a modulação codificada. Baldini não propõe nenhum esquema de decodificação para estes códigos.

3.8.7 - ANÁLISE COMPARATIVA DOS SISTEMAS DE MODULAÇÃO CODIFICADA

Este item apresenta uma pequena análise entre os diversos sistemas de modulação codificada com códigos de bloco apresentados neste capítulo.

Para que um sistema de modulação codificada seja realmente útil, é necessário que ele possua um procedimento de decodificação próximo ao ótimo e um compromisso razoável entre redundância, ganho de codificação e velocidade. A combinação de códigos binários distintos de modo a formar códigos multiníveis no espaço Euclidiano é de interesse prático, pois permite reduzir a complexidade de decodificação. Todos os sistemas apresentados neste capítulo fazem uso desse procedimento, formando arranjos-código a partir de códigos binários curtos, cujos bits são mapeados para os pontos de uma constelação de alfabeto redundante. Sistemas que permitem o uso de palavras sem codificação devidamente protegidas por palavras-código de códigos mais poderosos, são bastante eficientes, como é o caso, principalmente, dos sistemas de Sayegh, Williams e Farrell e Imai e Hirakawa. Quanto ao desempenho de decodificação, todos os sistemas são considerados sub-ótimos, devido à decodificação por etapas e consequente propagação de erros etapa-a-etapa. Embora a decodificação em cada etapa possa ser de máxima verossimilhança, cada decodificador isoladamente

não compartilha de toda a informação a respeito dos símbolos recebidos, assim a decodificação como um todo sofre uma degradação.

Os sistemas mais complexos quanto à implementação são os de Imai e Hirakawa e o de Tanner, pois necessitam de sistemas de retardo para a decodificação das diversas estimativas. O mais simples é o Sayegh. O sistema de Williams e Farrell, apesar da representação das distâncias em uma dimensão que simplifica a utilização de informação de confiabilidade para a decodificação, exige o uso de cálculos algébricos. Essa representação, também, é fator de perda de informação quanto aos sinais recebidos, degradando o desempenho de decodificação.

O sistema de Baldini é a única exceção a essas filosofias, pois baseia-se em códigos multiníveis e, aparentemente, não apresenta facilidade quanto à decodificação.

CAPÍTULO 4

SISTEMA DE MODULAÇÃO CODIFICADA COM CÓDIGOS DE BLOCO MULTINÍVEIS E DECODIFICAÇÃO POR REPRESENTAÇÃO BINÁRIA

Nos sistemas de Modulação Codificada com Códigos de Bloco (BCM) faz-se, primeiramente, a codificação da informação binária com códigos de bloco binários, determinados apropriadamente, e, em seguida, o mapeamento para o espaço de sinais de modo a se ter seqüências de sinais bastante distanciadas. Baldini [BAL 89] apresenta uma proposta completamente diferente, em que primeiro se realiza o mapeamento da informação binária em um conjunto expandido de sinais e, a seguir, codifica-os aumentando, assim, a distância Euclidiana mínima entre as seqüências de sinais. Para a codificação, Baldini usa códigos de bloco multiníveis definidos pela matriz geradora multinível de elementos pertencentes a um anel de inteiros módulo-q. O problema é determinar a matriz geradora de modo a maximizar a distância Euclidiana entre todas as palavras-código.

Godoy e Arantes [G&A 90] apresentam um algoritmo para a decodificação dos códigos propostos por Baldini. Essencialmente este algoritmo é composto de uma variante do Algoritmo de Conjuntos de Informação Modificado (ACIM) conjugada intrinsecamente com apagamentos em símbolos previamente escolhidos do vetor recebido. Os resultados apresentados para o código $C(4;12,4^6,4)$ mostraram que este algoritmo é adaptativo ao canal AWGN; isto é, o número de correlações diminui bastante com o aumento da relação sinal-ruído. Aparentemente este algoritmo é mais apropriado para implementação com circuitos microprocessados.

Seguindo a mesma linha de raciocínio de Baldini, ou seja, buscar códigos multiníveis para a modulação codificada, este capítulo propõe e analisa um novo esquema para a codificação e decodificação de códigos multiníveis baseado na representação binária dos símbolos multiníveis. O estado da arte está em se conseguir uma decodificação mais simples, mesmo que para isto a codificação se torne mais complexa. Assim, esses códigos são obtidos pelo mapeamento de conjuntos de bits, pertencentes a palavras-códigos de códigos binários e de transformações lineares destas, em pontos do conjunto expandido de sinais da modulação. A representação binária dos símbolos multiníveis facilita o processo de decodificação e a implementação física do sistema, consistindo em um compromisso entre a velocidade e o desempenho de decodificação.

4.1 - PROCESSO DE CODIFICAÇÃO PROPOSTO

Uma das dificuldades que ocorrem, quando se trata de codificação e decodificação, é a relativa simplicidade existente na codificação e a complicada tarefa em se realizar a decodificação. A figura 4.1 mostra o codificador multinível proposto que é aparentemente mais complexo, mas tem a finalidade de facilitar a decodificação com a utilização das conhecidas técnicas de decodificação de códigos de bloco binários.

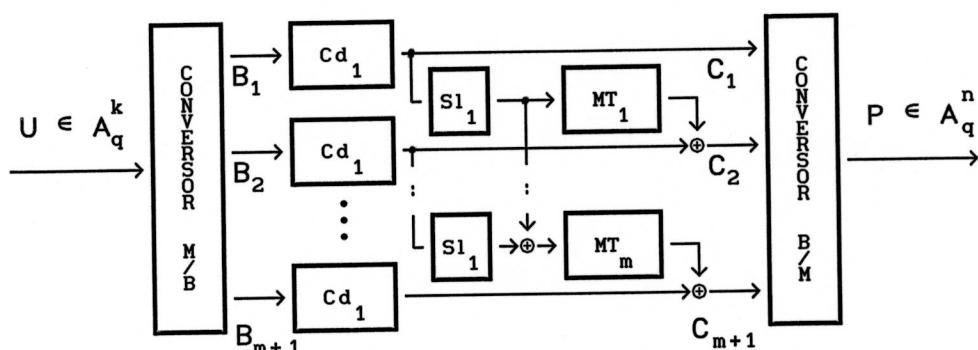


FIGURA 4.1 - CODIFICADOR MULTINÍVEL PROPOSTO

A seqüência de informação $U = (u_1, u_2, \dots, u_k)$ de elementos pertencentes ao Grupo Abeliano de inteiros de q elementos, $A_q = (0, 1, \dots, q-1)$, é representada pelos vetores binários $B_j = (b_{j1}, b_{j2}, \dots, b_{jk})$, $j = 1, 2, \dots, m+1$; pela ação do conversor Multinível/Binário. Os valores de b_{ji} são os que satisfazem o polinômio abaixo, com $q = 2^{m+1}$.

$$u_i = \sum_{j=1}^{m+1} b_{ji} \cdot x^{j-1} \quad (4.1)$$

Os vetores B_j são, então, codificados em palavras-código do código binário $C(2; n, M, d)$, pelo codificador Cd_1 , de matriz binária geradora :

$$G = \begin{bmatrix} I & \vdots & J \\ \vdots & \ddots & \vdots \end{bmatrix}^n \quad (4.2)$$

onde I é a matriz identidade, e $J_{ij} = 0$ ou 1 . Os vetores resultantes da codificação na primeira linha, $C_j = (c_{j1}, c_{j2}, \dots, c_{jn})$, $j = 1, n \geq k$; são levados aos seletores $S1_1$ e ao conversor Binário/Multinível. Esses seletores escolhem k bits que são somados aos demais bits selecionados (a partir da segunda linha) e multiplicam as matrizes de transformação binárias :

$$M_{Tj} = \begin{bmatrix} A_j & \vdots & B_j \\ \vdots & \ddots & \vdots \end{bmatrix}^n \quad (4.3)$$

onde $j = 1, 2, \dots, m$.

Os bits resultantes dessas multiplicações sofrem uma transformação linear, sendo somados (soma módulo dois) às palavras-código provenientes da codificação dos vetores B_j , $j = 2, 3, \dots, m+1$; pelo código binário $C(2; n, M, d)$. Ao final destas operações, os bits resultantes c_{ji} , são convertidos em elementos do grupo de inteiros, pelo conversor

Binário/Multinível, segundo o polinômio abaixo, sendo mapeados numa constelação de q elementos.

$$p_i = \sum_{j=1}^{m+1} c_{ji} \cdot x^{j-1} \quad (4.4)$$

Assim, a seqüência p é uma palavra-código multinível. Os codificadores Cd_1 são todos iguais, visando facilidade de implementação.

4.2 - CARACTERÍSTICAS DOS CÓDIGOS MULTINÍVEIS FORMADOS

Os códigos possuem taxa $R = k/n$, com $n \geq k$. As matrizes M_t são encontradas por busca computacional, para determinada matriz G , de modo que a distância Euclidiana mínima do código multinível formado seja a maior possível. A busca de matrizes binárias pode ser mais fácil que a busca de matrizes multiníveis. Por exemplo, fixadas as submatrizes :

$$A = \emptyset \quad \text{e} \quad J = \begin{bmatrix} 0111 \\ 1111 \\ 1011 \\ 1001 \end{bmatrix}$$

foram encontradas, através de busca computacional exaustiva, 256 submatrizes B que satisfazem a condição $dE_{\min}^2 = 10$, para a modulação 4-PSK.

TEOREMA 4

Os códigos multiníveis propostos, definidos sobre o grupo de inteiros, formados a partir de códigos binários, são Lineares e Invariantes em Distância.

Prova : Os códigos são lineares porque satisfazem à definição 3.3 e são invariantes em distância devido ao Teorema 1.

Em função das características de linearidade e invariância em distância, a busca de códigos é acelerada, já que a distância Euclidiana mínima do código pode ser encontrada comparando todas as palavras do código apenas com a palavra-código multinível tudo-zero.

TEOREMA 5

Os códigos multiníveis propostos possuem distância Euclidiana mínima ao quadrado limitada inferiormente por $\Delta_0^2 \cdot d_1$.

Prova : A distância Euclidiana mínima ao quadrado é calculada por :

$$dE_{\min}^2 \geq \min (\Delta_0^2 \cdot d_1, \Delta_1^2 \cdot d_2, \dots, \Delta_m^2 \cdot d_{m+1})$$

onde : Δ_j , $j = 0, \dots, m$; é a distância Euclidiana entre subconjuntos da i-ésima partição do espaço de sinais usado;

d_j , $j = 1, \dots, m+1$; é a distância de Hamming mínima do código binário formado pelos vetores c_j (figura 4.1).

A operação das matrizes M_T não modifica a distância de Hamming do código $C(2; n, M, d)$ nas linhas $2, \dots, m+1$; assim :

$$d_1 = d_2 = \dots = d_{m+1}$$

E, finalmente, a distância Euclidiana mínima ao quadrado para os códigos formados é dada por :

$$dE_{\min}^2 \geq \Delta_0^2 \cdot d_1 \quad (4.5) \quad \text{C.Q.D.}$$

O grupo de inteiros satisfaz apenas a operação de adição (equação 3.8) através da representação dos inteiros em polinômios (equações 4.1 e 4.4). Estes polinômios são aqueles

formadores de um Campo de Galois $GF(q)$. Considerando cada polinômio como um elemento (α^i) do Campo de Galois, tem-se a seguinte relação entre elementos e inteiros : $\alpha^i \longleftrightarrow i+1$. Por exemplo, para $q = 4$:

$$GF(4) = \{0, \alpha^0, \alpha^1, \alpha^2\} \longleftrightarrow A_4 = \{0, 1, 2, 3\}$$

E, para este caso, a tabela de adição entre os elementos do grupo A_4 é mostrada abaixo.

+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

A tabela 4.I mostra alguns códigos multiníveis lineares, de taxa $R = 1/2$, gerados a partir de códigos de bloco binários, para a modulação 4-PSK codificada e seus ganhos assintóticos sobre a modulação 2-PSK não codificada. A distância mínima Euclidiana ao quadrado é determinada considerando $\Delta_0 = \sqrt{2}$.

4.2.1 - EXEMPLO DE CODIFICAÇÃO

Como exemplo de codificação, considere-se que o código multinível seja definido pelas matrizes da terceira linha da tabela 4.I, que a modulação seja a 4-PSK, que S_{l_1} seleciona os quatro primeiros bits e que a seqüência de informação a ser codificada é $U = (0113)$. A palavra-código multinível formada é $P = (01133321)$. A figura 4.2 mostra os valores binários nos diversos pontos do sistema para este exemplo.

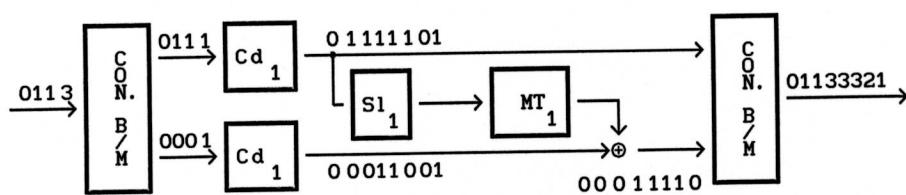


FIGURA 4.2 - EXEMPLO DE CODIFICAÇÃO

TABELA 4.1 - CÓDIGOS MULTINÍVEIS LINEARES EM A₄ (R = 1/2)

n	dE_{\min}^2	g_{∞}	G	M _T
4	8	3,0	$\begin{bmatrix} 1010 \\ 0101 \end{bmatrix}$	$\begin{bmatrix} 0001 \\ 0010 \end{bmatrix}$
6	8	3,0	$\begin{bmatrix} 100100 \\ 010010 \\ 001001 \end{bmatrix}$	$\begin{bmatrix} 000001 \\ 000010 \\ 000100 \end{bmatrix}$
8	10	4,0	$\begin{bmatrix} 10000111 \\ 01001111 \\ 00101011 \\ 00011001 \end{bmatrix}$	$\begin{bmatrix} 00001000 \\ 00000001 \\ 00001101 \\ 00001011 \end{bmatrix}$
10	12	4,7	$\begin{bmatrix} 1000010011 \\ 0100011111 \\ 0010011001 \\ 0001011010 \\ 0000101101 \end{bmatrix}$	$\begin{bmatrix} 0000001001 \\ 0000001010 \\ 0000000101 \\ 0000001101 \\ 00000010100 \end{bmatrix}$
12	12	4,7	$\begin{bmatrix} 100000111110 \\ 010000011111 \\ 001000100011 \\ 000100111101 \\ 000010110010 \\ 000001011001 \end{bmatrix}$	$\begin{bmatrix} 000000000001 \\ 000000000010 \\ 000000000100 \\ 000000001000 \\ 000000010001 \\ 000000100000 \end{bmatrix}$

4.3 - PROCESSO DE DECODIFICAÇÃO PROPOSTO

O primeiro processo de decodificação proposto é uma variante do Algoritmo de Chase para códigos multiníveis, aliado à decodificação binária dos símbolos multiníveis. No algoritmo original de Chase [CHA 72], o procedimento a ser seguido é :

- a) Fazer decisões abruptas em cada símbolo da seqüência recebida R para produzir o vetor P ;
- b) Introduzir vários padrões de erro, E , para gerar a seqüência candidata $\tilde{P} = E + P$, e decodificar cada seqüência criada;
- c) Computar a distância de cada seqüência candidata decodificada em relação à seqüência recebida, e selecionar a seqüência mais próxima.

Chase propôs três variantes, cuja diferença reside apenas no método de formar as seqüências candidatas.

Os padrões de erro produzem apagamentos nas seqüências candidatas, aumentando a probabilidade de decodificação correta, já que um código de bloco de distância mínima $d \geq 2t + s$ é capaz de corrigir qualquer padrão de t erros e s apagamentos [M&L 85].

O primeiro processo de decodificação proposto segue o fluxograma da figura 4.3.

A seqüência recebida $R = (r_1, r_2, \dots, r_n)$, $r_i \in \mathbb{R}$, tem seus símbolos afastados dos pontos da constelação devido ao ruído, caracterizando erros de recepção. Um exemplo de recepção é mostrado na figura 4.4 para a modulação 4-PSK.

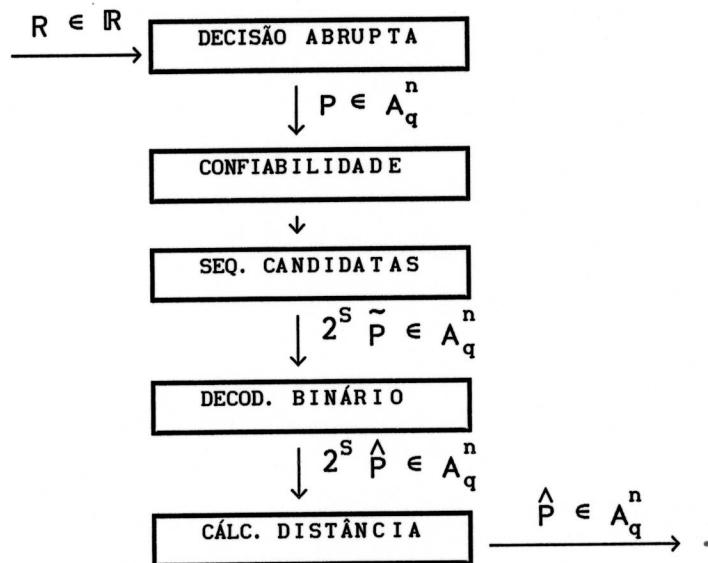


FIGURA 4.3 - FLUXOGRAMA DE DECODIFICAÇÃO (CHASE)

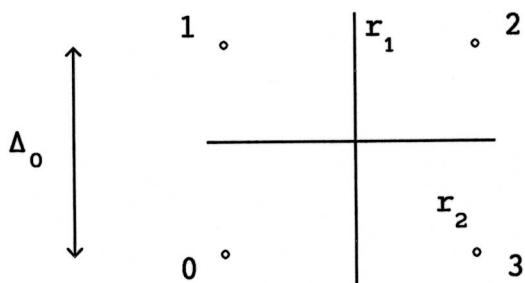


FIGURA 4.4 - SÍMBOLO RECEBIDO NA CONSTELAÇÃO 4-PSK

A seqüência R , inicialmente, sofre decisão abrupta em cada símbolo, obtendo a seqüência $P = (p_1, p_2, \dots, p_n)$, $p_i \in A_q^n$. A confiabilidade $\alpha(r_i)$ de cada símbolo é calculada, sendo que um símbolo é mais confiável quanto mais próximo ele se encontra de um ponto na constelação. Assim, o símbolo r_2 é mais confiável do que r_1 na figura 4.4. Para a constelação apresentada, a confiabilidade absoluta é definida pela equação :

$$\alpha(r_i) = 1 - \frac{2 \cdot dE(r_i, p_i)}{\Delta_0 \cdot \sqrt{2}} \quad (4.6)$$

onde p_i é o ponto pertencente à constelação mais próximo de r_i .

A seguir são formadas 2^S seqüências candidatas à decodificação $\tilde{p} = (\tilde{p}_1, \tilde{p}_2, \dots, \tilde{p}_n)$, $\tilde{p}_i \in A_q^n$, por alterações nos S símbolos menos confiáveis da seqüência p . Essas alterações consistem na aplicação da regra definida pela equação :

$$\tilde{p} = \begin{cases} DA(-x; y) & \text{se } |x| \leq |y| \\ DA(x; -y) & \text{se } |x| > |y| \end{cases} \quad (4.7)$$

onde x e y são as coordenadas do símbolo recebido r , e $DA(\cdot, \cdot)$ é a decisão abrupta de um símbolo.

Assim, por exemplo, se r_1 na figura 4.4 possui coordenadas $(0,1;1,0)$, então $\tilde{p} = DA(-0,1;1,0) = 1$; ou seja, o símbolo 2, na seqüência p , é alterado para 1.

Cada uma das 2^S seqüências candidatas \tilde{p} é decodificada binariamente pelo sistema mostrado na figura 4.5.

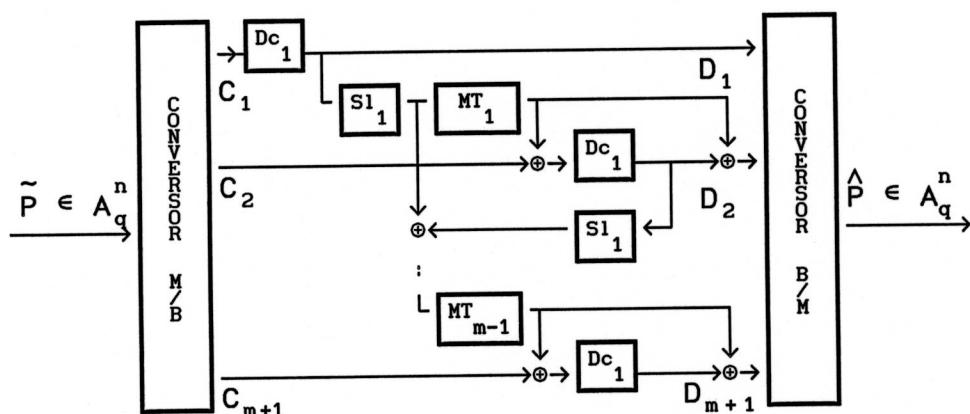


FIGURA 4.5 - SISTEMA DE DECODIFICAÇÃO BINÁRIO

O conversor Multinível/Binário transforma a seqüência p em sua representação binária c_j , $j = 1, 2, \dots, m+1$.

Os vetores c_1 formam uma palavra-código, com erro ou não, que é decodificada pelo decodificador Dc_1 e o resultado dessa operação é levado ao conversor Binário/Multinível e ao seletor $S1_1$. Os seletores selecionam k bits que são enviados às matrizes transformadoras M_t . As seqüências obtidas após as matrizes são adicionadas (soma módulo 2) aos vetores c_j , $j = 2, 3, \dots, m+1$; e o resultado é decodificado pelos decodificadores Dc_1 . Após a decodificação, tem-se palavras-código que devem ser somadas às seqüências provenientes das matrizes M_t , realizando operação inversa à ocorrida na codificação.

O conversor Binário/Multinível determina a seqüência decodificada $\hat{p} = (\hat{p}_1, \hat{p}_2, \dots, \hat{p}_n)$, $\hat{p}_i \in A_q^n$.

Finalmente, a seqüência \hat{p} que apresentar a menor distância Euclidiana em relação à seqüência recebida R , é considerada a seqüência decodificada.

Os decodificadores binários Dc_1 , a princípio, são todos iguais visando a facilidade de implementação física. Pode-se implementar um único decodificador e usá-lo em instantes diferentes de tempo. Eles não precisam ser necessariamente do tipo MLD, de forma a manter um compromisso entre complexidade de implementação e desempenho.

4.3.1 - EXEMPLO DE DECODIFICAÇÃO

Como exemplo de decodificação, considere-se que a seqüência codificada, anteriormente, $p = (01133321)$ tenha sido recebida em coordenadas x e y da constelação 4-PSK como:

$$R = (-1; -1)(-1; 1)(-1; 1)(1; 0, 1)(0, 5; -1)(1; -1)(1; 1)(-1; -0, 8),$$

ou seja, $P = (01123320)$ em decisão abrupta com dois erros, e confiabilidade $\alpha(R) = (1;1;1;0,3;0,6;1;1;0,8)$. Supondo $S = 2$, formam-se quatro seqüências candidatas \tilde{P} , alterando as duas posições menos confiáveis de P , assim :

$$\begin{aligned}\tilde{P}_1 &= (01123320) \\ \tilde{P}_2 &= (01133320) \\ \tilde{P}_3 &= (01120320) \\ \tilde{P}_4 &= (01130320)\end{aligned}$$

Essas seqüências são decodificadas binariamente, produzindo as seqüências decodificadas \hat{P}_I , $I = 1,2,3,4$.

$$\begin{aligned}\hat{P}_1 &= (01322320) \\ \hat{P}_2 &= (01133321) \\ \hat{P}_3 &= (01120302) \\ \hat{P}_4 &= (01120302)\end{aligned}$$

A decodificação binária de \tilde{P}_1 é mostrada na figura 4.6.

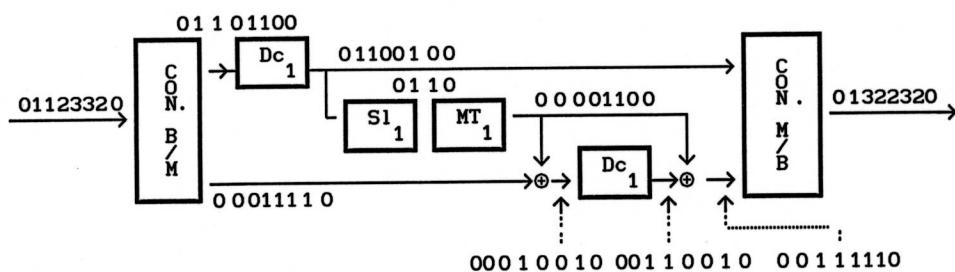


FIGURA 4.6 - EXEMPLO DE DECODIFICAÇÃO

A seqüência \hat{P}_2 é a que apresenta a menor distância Euclidiana em relação a R e, portanto, é a seqüência decodificada.

4.4 - SIMULAÇÃO DO SISTEMA DE MODULAÇÃO CODIFICADA

O objetivo da simulação é avaliar a Probabilidade de Erro de Bit (P_b) nas baixas e médias relações sinal-ruído (~ 1 a 15 dB), já que, nesta faixa, é difícil uma avaliação analítica [BEZ 87]. A P_b pode ser calculada pela razão entre o número de bits errados e o número de bits transmitidos, ambos bits de informação, ou seja, a TEB. Resta estabelecer o número de bits a serem transmitidos na simulação para uma determinada relação sinal-ruído. Para uma boa estimativa, o critério prático, normalmente adotado, é que o número de bits transmitidos deve ser dez vezes o inverso da probabilidade de erro esperada [BEZ 87]. Uma probabilidade de erro de bit de 10^{-4} cobre as baixas e médias relações sinal-ruído; então o número de bits a serem transmitidos é :

$$N_b \geq \frac{10}{P_b} = \frac{10}{10^{-4}} = 10^5 \text{ bits}$$

Para o levantamento do desempenho de decodificação dos algoritmos propostos, desenvolveu-se um programa de computador, em que os símbolos das palavras-códigos multiníveis são mapeados para os pontos da constelação de sinais. Os sinais são transmitidos pelo canal e, posteriormente, são "demapeados" e decodificados. Cada palavra multinível decodificada é comparada com a que foi, originalmente, transmitida, computando-se o número de bits de informação que foram decodificados erroneamente, para o cálculo da Taxa de Erro de Bit (TEB).

Neste trabalho, somente é considerado o efeito do ruído Gaussiano branco e aditivo. Cada ponto da constelação bidimensional é representado por seu componente em fase (x_i) e em quadratura (y_i). Para simular o canal AWGN, duas variáveis aleatórias Gaussianas n_x e n_y de média zero e variância σ^2 são adicionadas às componentes x_i e y_i , respectivamente; então o sinal recebido será $(x_i + n_x, y_i + n_y)$.

A partir de variáveis aleatórias com distribuição uniforme no intervalo [0,1], é possível gerar variáveis aleatórias para determinada distribuição de probabilidade através do método da Transformação Inversa. Variáveis aleatórias Gaussianas, de média zero e variância igual a um, obtidas com este método, são dadas por qualquer uma das fórmulas [FIS 78] :

$$W_1 = \sqrt{-2 \cdot \ln(U_1)} \cdot \sin(2\pi \cdot U_2)$$

$$W_2 = \sqrt{-2 \cdot \ln(U_1)} \cdot \cos(2\pi \cdot U_2)$$

onde : W_1 e W_2 são variáveis aleatórias Gaussianas;
 U_1 e U_2 são variáveis aleatórias com distribuição uniforme.

A relação E_b/N_0 do canal fornece a variância σ^2 para as variáveis aleatórias n_x e n_y . A partir de $E_b = E_c/R'$ (equação 3.14) e sabendo que $\sigma^2 = N_0/2$, para o canal AWGN, obtém-se :

$$\frac{E_b}{N_0} = \frac{E_c}{R' \cdot N_0} = \frac{E_c}{R' \cdot 2\sigma^2} = \frac{E_c}{m \cdot R \cdot 2\sigma^2}$$

dado que o número de bits de informação, transmitido por intervalo de modulação (R'), é igual à taxa de codificação (R) vezes o número total de bits transmitido neste mesmo intervalo (m). Como a relação E_b/N_0 é dada em decibels, obtém-se :

$$\left. \frac{E_b}{N_0} \right|_{dB} = 10 \cdot \log_{10} \left(\frac{E_b}{N_0} \right) = 10 \cdot \log_{10} \left(\frac{E_c}{m \cdot R \cdot 2\sigma^2} \right)$$

Então, a variância para uma relação E_b/N_0 em decibels é dada por :

$$\sigma^2 = \frac{E_c}{2R \cdot m \cdot 10^{0,1} \frac{E_b}{N_0} \mid_{dB}} \quad (4.8)$$

onde : R é a taxa de codificação;

E_c é a energia média de transmissão (equação 2.3).

Finalmente, $n_x = n_y = \sigma \cdot W_1$ (ou $\sigma \cdot W_2$). Desta forma, obtem-se as variáveis aleatórias Gaussianas necessárias à simulação.

4.5 - ANÁLISE DE DESEMPENHO

O processo de decodificação ótimo é o de máxima verossimilhança com decisão suave. Os códigos das terceira (C1) e quarta (C2) linhas da tabela 4.I foram decodificados segundo este procedimento. A figura 4.7 mostra as curvas de desempenho para estes códigos, para a modulação 4-PSK codificada, que apresentam ganhos assintóticos de 4,0 dB e 4,7 dB sobre a modulação 2-PSK não codificada, respectivamente. O eixo vertical do gráfico apresenta a Taxa de Erro de Bit (TEB), e o eixo horizontal mostra a relação Energia de Bit por Densidade Espectral de Ruído (E_b/N_0).

Na decodificação pelo processo proposto os decodificadores binários Dc_1 simulados são do tipo MLD. O número de palavras candidatas à decodificação é variável para que se possa comparar o desempenho do sistema em relação a esta grandeza.

A figura 4.8 apresenta o desempenho do código multinível C1. À medida que aumenta o número de seqüências candidatas, melhora o desempenho do sistema; para 32 palavras candidatas o desempenho é aproximadamente 1,25 dB inferior ao MLD. O aumento do número de palavras candidatas para 64 não traz ganho significativo ao sistema e o torna duas vezes mais lento. Portanto, 32 palavras candidatas representam o ponto

ótimo no compromisso velocidade vezes desempenho de decodificação. Este valor foi determinado experimentalmente.

Para melhorar o desempenho do sistema deve-se buscar códigos multiníveis com distâncias Euclidianas maiores. Um código multinível de comprimento igual a oito e com distância mínima ao quadrado igual a 12 pode ser obtido com a utilização das matrizes binárias :

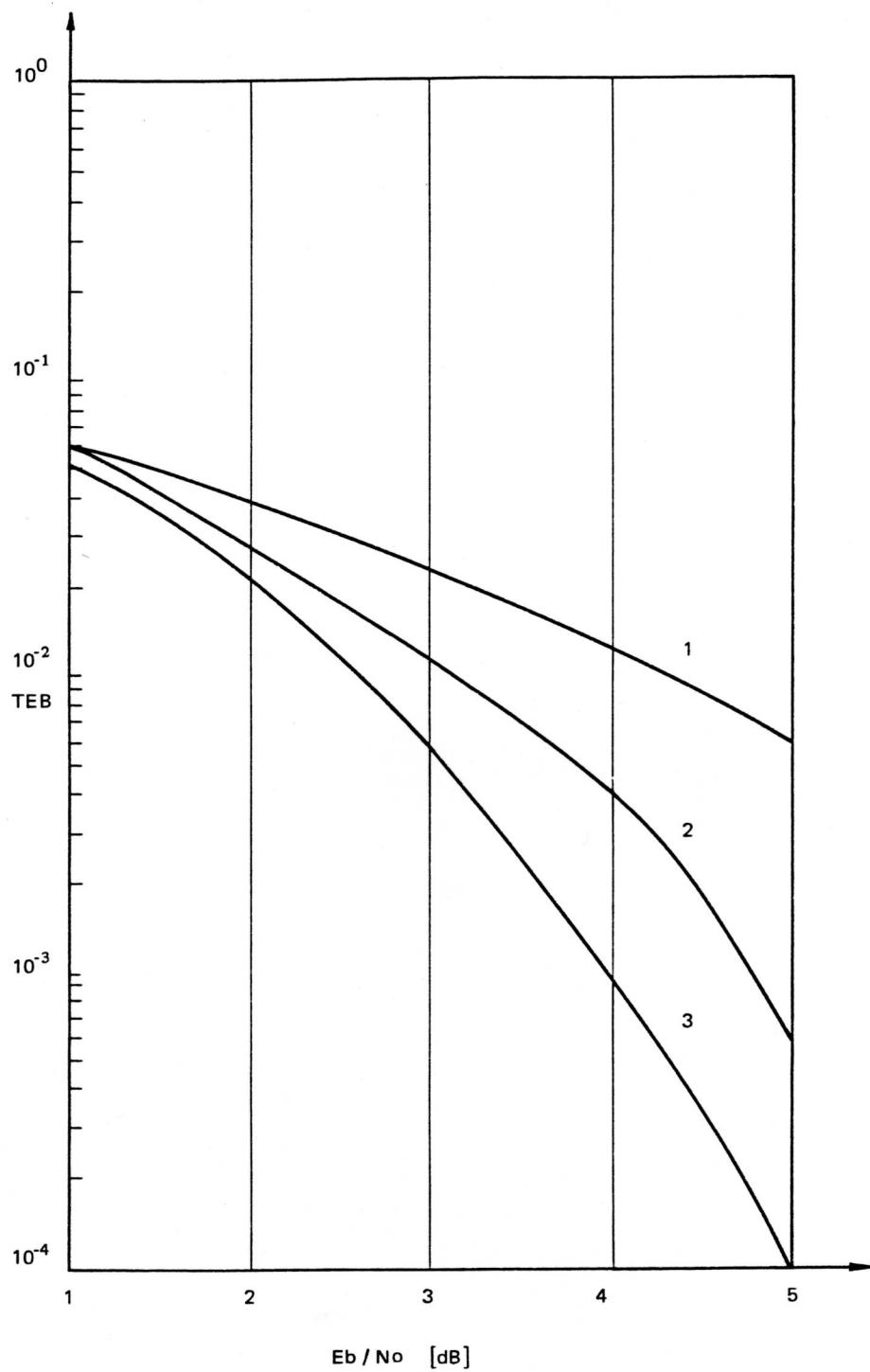
$$G = \begin{bmatrix} 10000111 \\ 01001011 \\ 00101101 \\ 00011110 \end{bmatrix} \quad (4.9) \quad M_T = \begin{bmatrix} 00010100 \\ 00101001 \\ 11010001 \\ 10001100 \end{bmatrix} \quad (4.10)$$

A matriz G assegura um código binário de distância mínima de Hamming igual a quatro. A utilização da matriz M_T permite obter esse código multinível ($dE^2 = 12$) com a retirada (pinçamento) de oito das palavras multiníveis do código que possuem distância Euclidiana em relação à tudo-zero igual a oito. Desta forma, o código formado, C3, terá apenas 248 palavras-código.

O ganho assintótico esperado desse código em relação ao código C1 com dE_{\min}^2 igual a dez é :

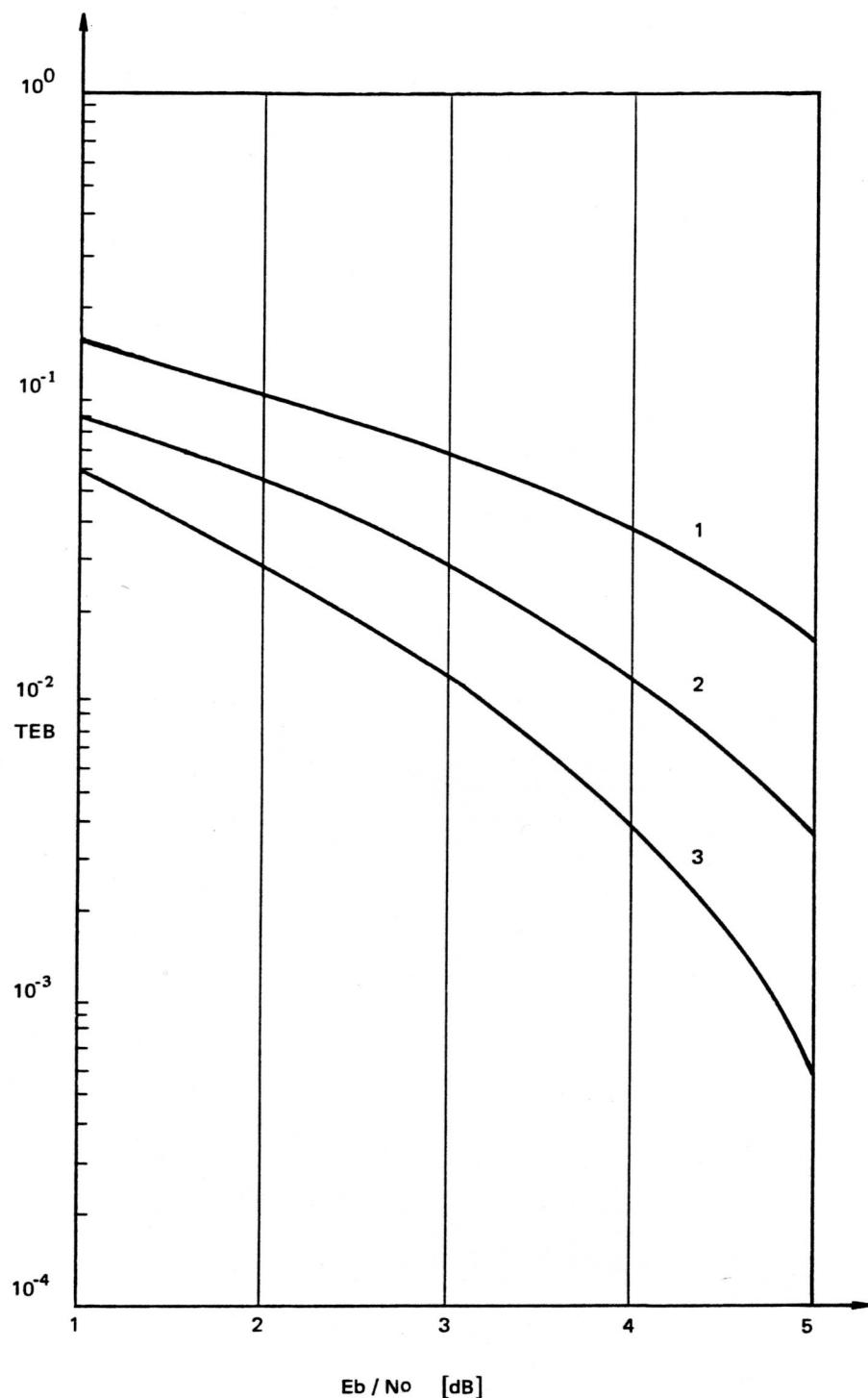
$$g_\infty = 10 \cdot \log_{10} \left(\frac{dE^2}{dE_r^2} \right) = 10 \cdot \log_{10} \left(\frac{12}{10} \right) = 0,79 \text{ dB}$$

As curvas de desempenho de decodificação MLD multinível para este código são mostradas na figura 4.9, comparadas ao desempenho do código C1.



- 1 - MODULAÇÃO 2-QAM NÃO CODIFICADA
- 2 - CÓDIGO C1 - MLD
- 3 - CÓDIGO C2 - MLD

FIGURA 4.7 - DESEMPENHO DE DECODIFICAÇÃO MLD

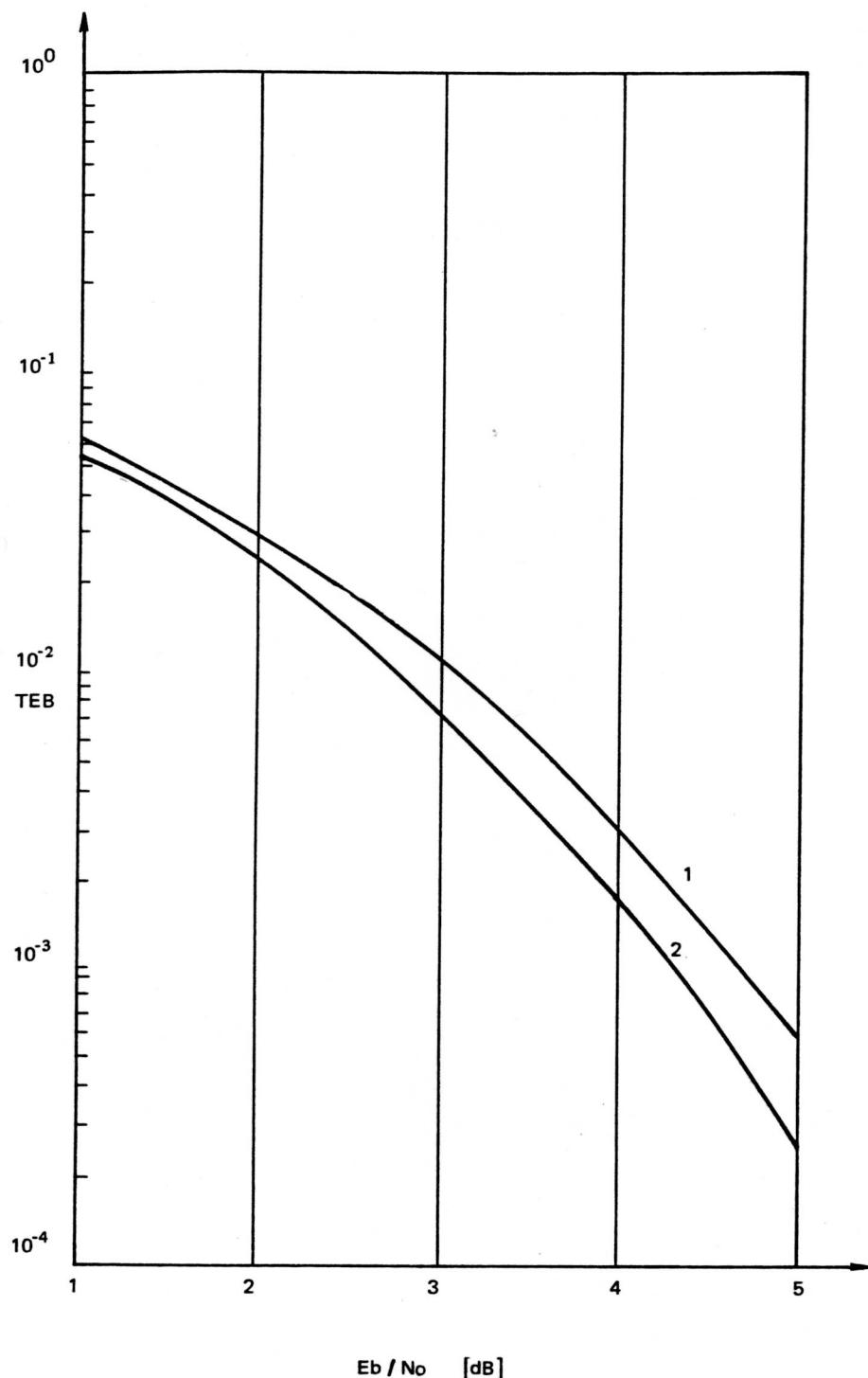


1 - CÓDIGO C1 - CHASE - 4 PALAVRAS CANDIDATAS

2 - CÓDIGO C1 - CHASE - 32 PALAVRAS CANDIDATAS

3 - CÓDIGO C1 - MLD

FIGURA 4.8 - DESEMPENHO DO ALGORITMO PROPOSTO (CHASE)



1 - CÓDIGO C1 - MLD - 256 PALAVRAS-CÓDIGO
2 - CÓDIGO C3 - MLD - 248 PALAVRAS-CÓDIGO

FIGURA 4.9 - DESEMPENHO DO CÓDIGO C3

4.6 - A REPRESENTAÇÃO DO SISTEMA POR UMA ÚNICA MATRIZ

O sistema de codificação proposto (figura 4.1) pode ser visto de maneira diferente. Já que as operações, realizadas por este sistema, são todas lineares, pode-se representar todo o seu funcionamento através de uma única matriz binária. Essa representação só é válida quando os códigos usados possuem mesmo comprimento n e mesmo número de símbolos de informação k . A idéia é desenvolvida para um exemplo específico; a generalização segue facilmente. Considere-se que a modulação é a 4-PSK, que S_1 seleciona os k primeiros bits da palavra binária codificada e considere-se as matrizes G e M_T (equações 4.2 e 4.3), então, a matriz T abaixo representa o funcionamento da parte binária do sistema :

$$T = \begin{bmatrix} I & A & J & B \\ \dots & \dots & \dots & \dots \\ \emptyset & I & \emptyset & J \end{bmatrix}^{2n} \quad (4.11)$$

onde as submatrizes I , J , A e B são provenientes das matrizes G e M_T , e \emptyset é a matriz nula quadrada $k \times k$. As primeiras e as terceiras colunas desta matriz representam o funcionamento da primeira linha do sistema de codificação (figura 4.1), enquanto a segunda e a quarta colunas correspondem ao funcionamento da segunda linha do sistema.

Assim, o sistema proposto de codificação pode ser representado, para 4-PSK, pela figura 4.10.

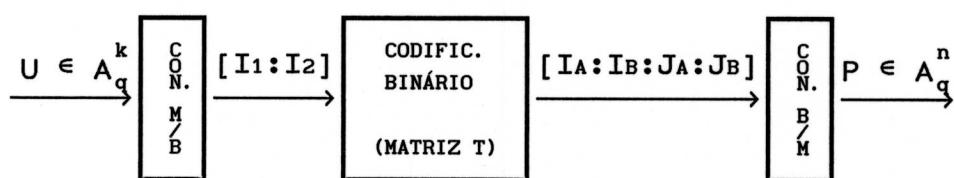


FIGURA 4.10 - SISTEMA DE CODIFICAÇÃO COM MATRIZ T

O vetor $[I_1:I_2]$ corresponde ao vetor B_1 seguido do vetor B_2 . O produto $[I_1:I_2].T$ traz como resultado o vetor binário $[I_A:I_B:J_A:J_B]$, que é uma palavra-código do código de matriz geradora T, sendo que o particionamento desse é $[I_A:J_A] = C_1$ e $[I_B:J_B] = C_2$. Como se pode observar, com a representação do sistema binário através da matriz T, grande simplificação é obtida para o processo de codificação em relação ao sistema da figura 4.1 que, resumidamente, torna-se conversão multinível/binário, multiplicação módulo dois pela matriz T, e conversão binário/multinível.

O processo de decodificação é o mesmo apresentado na figura 4.3, com exceção de que o decodificador binário da figura 4.5 dá lugar ao sistema apresentado na figura 4.11.

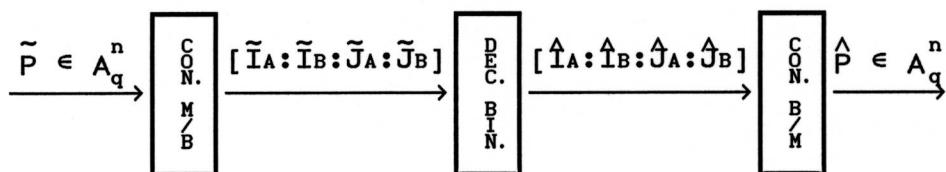
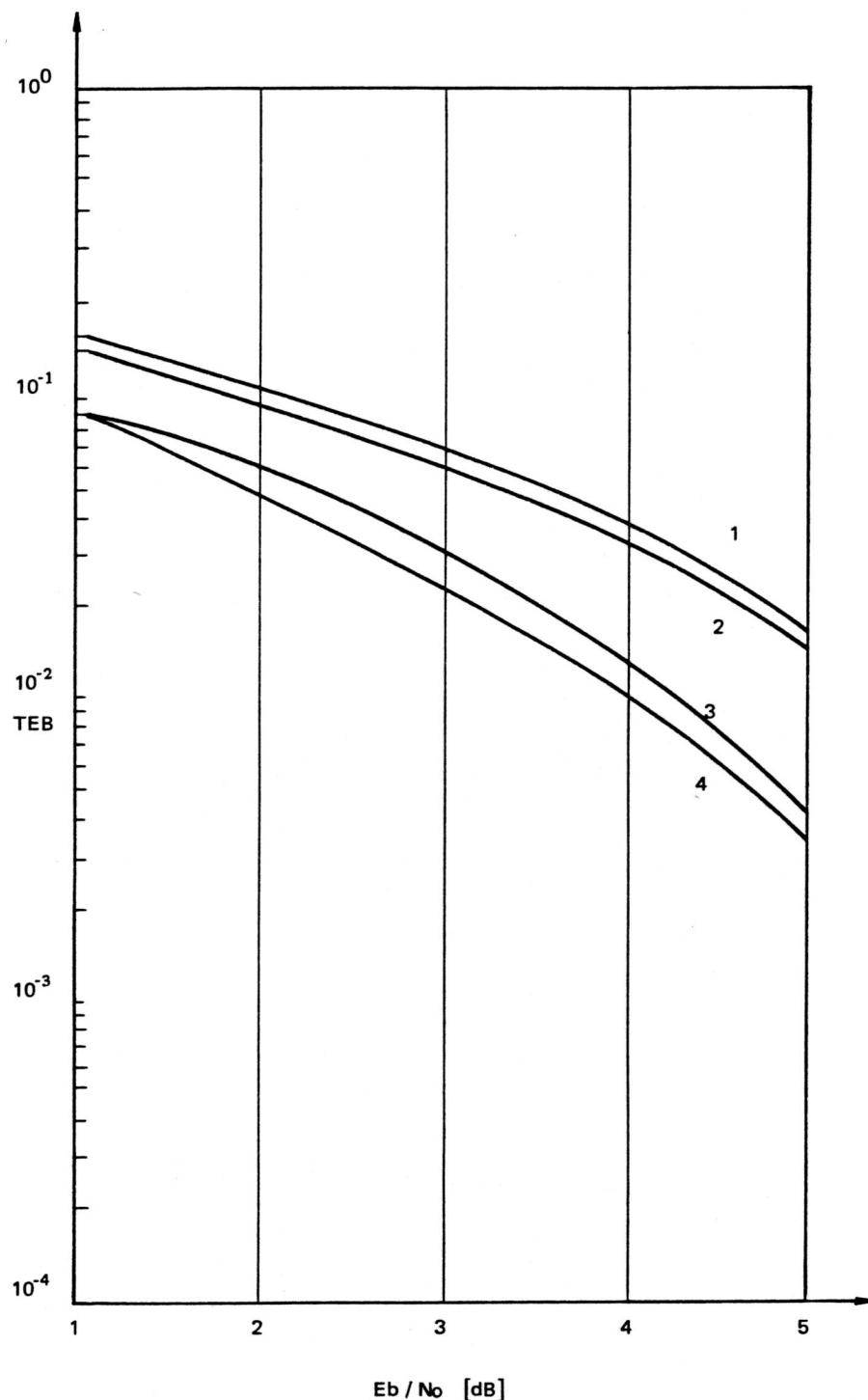


FIGURA 4.11 - SISTEMA DE DECODIFICAÇÃO BINÁRIO COM MATRIZ T

Nesta versão, a seqüência candidata \tilde{P} é convertida para a representação binária $[\tilde{I}_A:\tilde{I}_B:\tilde{J}_A:\tilde{J}_B]$ (uma palavra com erro ou não) que é decodificada pelo decodificador binário na palavra-código $[\hat{I}_A:\hat{I}_B:\hat{J}_A:\hat{J}_B]$, sendo essa convertida na seqüência decodificada \hat{P} .

O desempenho de decodificação para o código C1, representado binariamente pela matriz T e com decodificação binária de máxima verossimilhança, é mostrado na figura 4.12.

A figura mostra o desempenho do processo de decodificação que utiliza a matriz T, comparado ao processo anterior (figura 4.5), para quatro e 32 seqüências candidatas \tilde{P} . Observa-se um ganho de aproximadamente 0,2 dB, no presente caso, em relação ao anterior.



- 1 - CÓDIGO C1 - CHASE - MATRIZ G - 4 PALAVRAS CAND.
- 2 - CÓDIGO C1 - CHASE - MATRIZ T - 4 PALAVRAS CAND.
- 3 - CÓDIGO C1 - CHASE - MATRIZ G - 32 PALAVRAS CAND.
- 4 - CÓDIGO C1 - CHASE - MATRIZ T - 32 PALAVRAS CAND.

FIGURA 4.12 - DESEMPENHO DE DECODIFICAÇÃO COM A MATRIZ T

A representação do código multinível, através da matriz T, permite analisar o desempenho de decodificação do sistema. O código multinível C1 de $dE_{\min}^2 = 10$ corrige todos os padrões de erro que estão a uma distância Euclidiana de no máximo $\lfloor (dE_{\min}^2 - 1)/2 \rfloor = 4$ da palavra-código multinível correta, com decisão abrupta; ou seja, esse código corrige todos os padrões de erro de um símbolo, e alguns dos padrões de erro de dois símbolos, na decodificação por correlação completa.

O código binário de matriz geradora T, que representa o código multinível C1, possui distância mínima de Hamming igual a três, ou seja, corrige todos os padrões de um erro de bit e alguns padrões de dois erros de bit. Isto significa que, no pior caso, não se corrige nenhum símbolo, e o desempenho é inferior, como mostrado na figura 4.8.

Além disso, outro ponto a ser considerado quanto ao desempenho do Algoritmo de Chase, é o número de seqüências candidatas que devem ser formadas para que o desempenho de decodificação se aproxime do melhor desempenho (MLD). O Algoritmo de Chase fundamenta-se em alterações nos símbolos menos confiáveis da seqüência recebida, formando um número de seqüências candidatas que são decodificadas. Espera-se que, entre as seqüências decodificadas, esteja a palavra-código correta, de modo a que não ocorra um erro de decodificação.

Como se pode observar, nesse algoritmo, ocorrem dois tipos de perturbação na seqüência recebida; um na formação das seqüências candidatas e outro na decodificação binária sofrida por estas seqüências, para a formação das seqüências decodificadas. Em função destas perturbações, as seqüências candidatas afastam-se da palavra-código correta. Assim, grande número de seqüências se faz necessário para um desempenho de decodificação próximo ao de máxima verossimilhança.

Para contornar este problema, a utilização de um método de decodificação que perturbe ao mínimo as seqüências candidatas, é necessário. A decodificação através do Algoritmo Conjuntos de Informação (ACI) [C&C 81] satisfaz essa condição. Este é o segundo método de decodificação proposto.

4.7 - ALGORITMO CONJUNTOS DE INFORMAÇÃO

A decodificação ACI baseia-se no fato de que, em um código linear, qualquer conjunto de k posições de uma palavra que possa ser especificado independentemente, constitui um conjunto de informação. Por exemplo, as k primeiras colunas da matriz geradora de um código de bloco na forma sistemática forma uma matriz conjunto de informação (equação 4.12). Qualquer outro conjunto de k colunas da matriz geradora pode formar uma matriz conjunto de informação se for possível, através de operações elementares transformá-las em colunas de peso unitário, como por exemplo as quatro últimas colunas da matriz 4.13. Observe que cada linha da matriz é, obrigatoriamente, uma palavra-código.

$$G_1 = \begin{bmatrix} 10000111 \\ 01001111 \\ 00101011 \\ 00011001 \end{bmatrix} \quad (4.12) \quad G_2 = \begin{bmatrix} 11001000 \\ 01100100 \\ 00110010 \\ 11010001 \end{bmatrix} \quad (4.13)$$

Os símbolos de um conjunto de informação definem unicamente uma palavra-código. Se não houver erros nas posições do conjunto de informação, é possível reconstruir a palavra transmitida. O algoritmo de decodificação segue os seguintes passos :

- a) Construir diversos conjuntos de informação para o código utilizado;

- b) Formar várias estimativas da palavra transmitida, recodificando a palavra recebida utilizando cada uma das matrizes conjuntos de informação;
- c) Escolher a palavra-código decodificada mais próxima da palavra recebida.

Define-se, também, Máscara de uma matriz conjunto de informação o vetor binário que possui k bits unitários nas posições do conjunto de informação e zero nas demais posições. Denomina-se Conjunto de Cobertura a coleção de máscaras de um mesmo código, que diferem no maior número de posições possível e, juntas, cobrem um determinado número de padrões de erro. Por exemplo, a máscara para a matriz 4.12 é 11110000, e para a matriz 4.13 é 00001111. O padrão de dois erros 00110000 não é coberto pela primeira máscara, mas é coberto pela segunda. As duas máscaras juntas formam um conjunto de cobertura capaz de cobrir 12 padrões de dois erros.

Por exemplo, supondo-se o recebimento de vetor 00111011 (com erro no quarto bit) e a decodificação com o uso do conjunto de cobertura formado pelas matrizes 4.12 e 4.13, a recodificação do vetor pela matriz 4.13 forneceria a palavra decodificada correta.

Deve-se determinar o menor conjunto de cobertura para cobrir todos os padrões de t erros que o código em uso pode corrigir. O problema reside no fato de que nem sempre uma certa máscara corresponde a uma matriz conjunto de informação. No capítulo 6 é apresentada uma série de algoritmos para auxiliar a formação de conjuntos de cobertura.

4.8 - SISTEMA DE DECODIFICAÇÃO COM ACI

A figura 4.13 mostra o diagrama em blocos do sistema de decodificação que utiliza o Algoritmo Conjuntos de

Informação.

A seqüência P , proveniente da decisão abrupta da seqüência recebida R , é convertida para o vetor $[\tilde{I}_A:\tilde{I}_B:\tilde{J}_A:\tilde{J}_B]$. Através de N conjuntos de informação, são formados N seqüências decodificadas \hat{P} pela conversão dos N vetores decodificados $[\hat{I}_A:\hat{I}_B:\hat{J}_A:\hat{J}_B]$. A seqüência decodificada \hat{P} mais próxima, quanto à distância Euclidiana, de R , é a seqüência decodificada final.

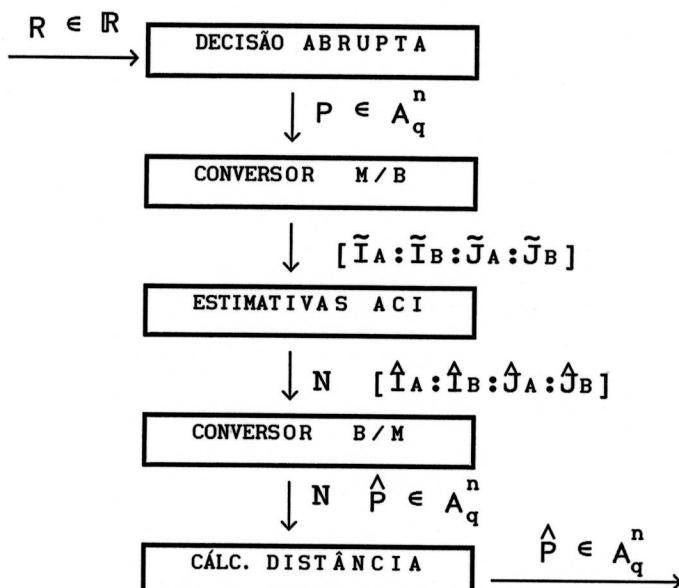


FIGURA 4.13 - FLUXOGRAMA DE DECODIFICAÇÃO (ACI)

A complexidade de decodificação está, diretamente, relacionada com o número de matrizes conjunto de informação usadas no sistema. Este número é calculado pela fórmula de Schönheim [SCH 64] :

$$N_{\text{cob}} \geq \left\lceil \frac{n}{n-k} \left\lceil \frac{n-1}{n-k-1} \cdots \left\lceil \frac{n-t+1}{n-k-t+1} \right\rceil \cdots \right\rceil \right\rceil \quad (4.14)$$

onde $\lceil \cdot \rceil$ significa "o menor inteiro maior que".

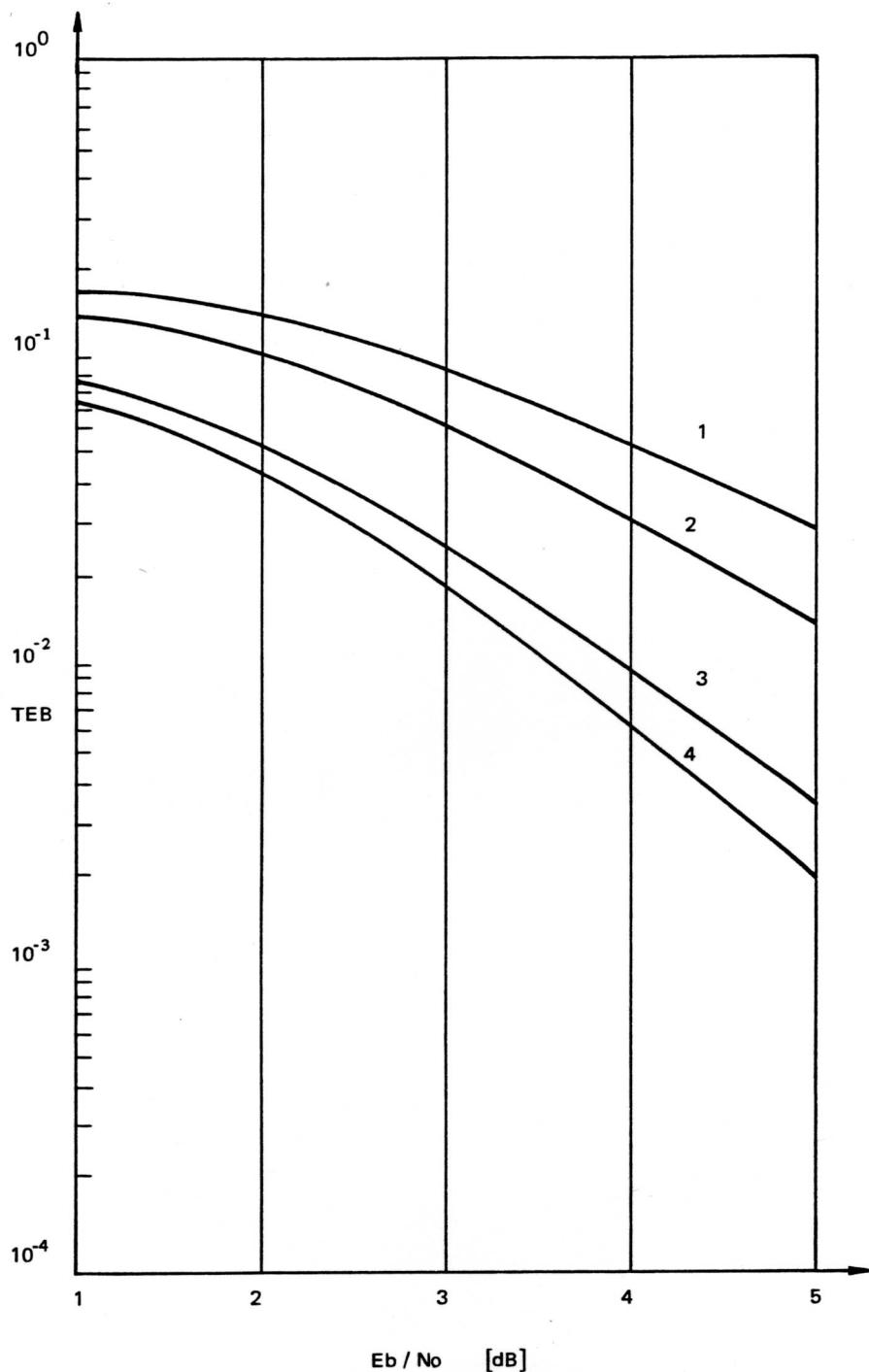
Esta fórmula determina o menor número de matrizes conjunto de informação necessárias para cobrir um determinado número de padrões de t erros para o código binário usado. Mas

trata-se de um limite inferior; na prática mais conjuntos devem ser usados.

O desempenho de decodificação para o código multinível C1, utilizando a decodificação binária com Algoritmo Conjunto de Informação, é mostrada na figura 4.14, para quatro e 32 matrizes conjunto de informação. As 32 matrizes cobrem todos os padrões de um e dois erros de bit e 546 padrões de erro de três bits e foram selecionados manualmente. Na figura, também, são mostradas as curvas de desempenho para quatro e 32 seqüências candidatas, com decodificação MLD binária e Algoritmo de Chase. Para quatro matrizes, observa-se uma perda facilmente explicável. Com quatro matrizes conjunto de informação não são cobertos todos os padrões de erro de um e dois bits que o código binário é capaz de corrigir. Para 32 matrizes a decodificação ACI traz um ganho de aproximadamente 0,25 dB em relação ao Algoritmo de Chase.

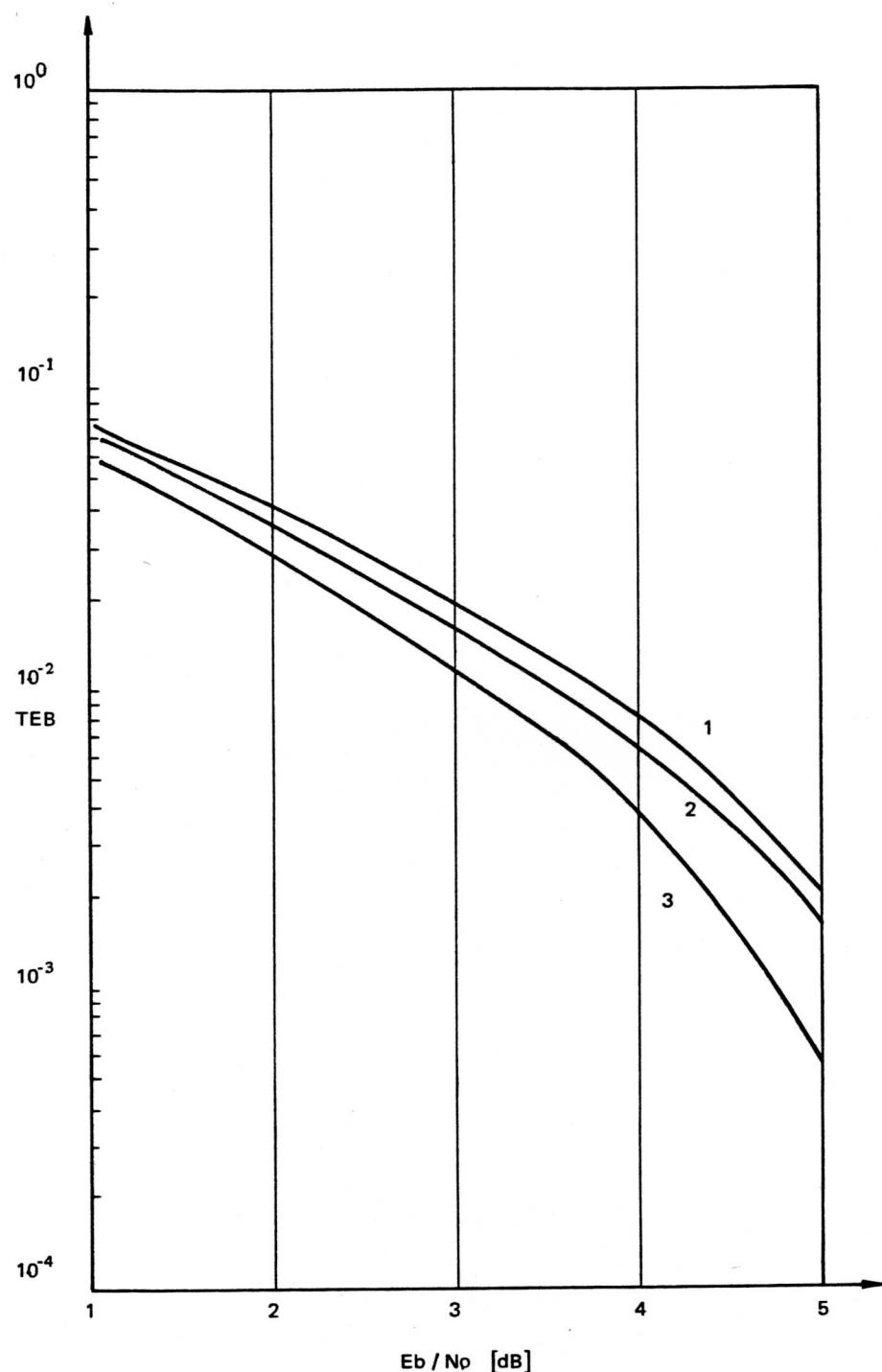
Buscando os limites de desempenho do sistema proposto, foi incorporado ao ACI a troca do símbolo menos confiável da seqüência recebida (apagamento) para a formação das seqüências candidatas. O desempenho dessa versão do sistema é mostrado na figura 4.15 para 32 matrizes conjunto de informação com apagamento no símbolo menos confiável, e comparada com o ACI normal e com a decodificação MLD multinível.

Observa-se um pequeno ganho em relação ao ACI normal e um desempenho em relação ao MLD apenas 0,5 dB inferior, na média. O pequeno ganho em relação ao ACI normal não compensa a complexidade de implementação trazida pela inclusão de apagamentos. Com relação ao Algoritmo de Chase, o ACI mostrou ser muito mais simples de ser implementado e seu desempenho é melhor, ainda que pouco.



- 1 - CÓDIGO C1 - ACI - MATRIZ T - 4 MATR. CONJ. INF.
- 2 - CÓDIGO C1 - CHASE - MATRIZ T - 4 PALAVRAS CAND.
- 3 - CÓDIGO C1 - CHASE - MATRIZ T - 32 PALAVRAS CAND.
- 4 - CÓDIGO C1 - ACI - MATRIZ T - 32 MATR. CONJ. INF.

FIGURA 4.14 - DESEMPENHO DE DECODIFICAÇÃO COM ACI



1 - CÓDIGO C1 - ACI - 32 MATR. CONJ. INF.

2 - CÓDIGO C1 - ACI - 32 MATR. CONJ. INF. + APAGAM.

3 - CÓDIGO C1 - MLD

FIGURA 4.15 - ACI COM APAGAMENTO

4.9 - COMPARAÇÃO COM O SISTEMA DE SAYEGH

Comparando o sistema proposto com o sistema de Sayegh, pode-se levantar os seguintes pontos :

a) Os códigos multiníveis propostos baseiam-se numa estrutura binária bem definida, a matriz T. As submatrizes que compõem a matriz T, são buscadas de modo a satisfazer condições impostas de distância Euclidiana do código multinível desejado e distância de Hamming do código binário que representa o código multinível. Os símbolos transmitidos são formados através de um mapeamento específico dos bits que compõem as palavras-código do código binário. Diferentemente de Sayegh, busca-se apenas um código binário que satisfaz as condições acima. Esta característica é a principal vantagem do sistema proposto em relação ao de Sayegh, pois é necessário somente um decodificador binário para a decodificação.

Sayegh busca códigos binários curtos de comprimentos fixos, de maneira a formar códigos no espaço de sinais de distância Euclidiana a maior possível. Como são usados códigos diferentes, são necessários vários decodificadores binários diferentes.

b) Os códigos multiníveis para a modulação 4-PSK codificada possuem equivalentes encontrados por Sayegh, como por exemplo, o primeiro código da tabela 4.I comparado com o primeiro código da tabela 3.II. Além deste, os códigos da quarta e quinta linhas da tabela 3.II podem ser obtidos para o sistema proposto.

4.10 - COMPARAÇÃO COM O SISTEMA DE BALDINI

Comparando o sistema proposto com o sistema de Baldini, pode-se levantar os seguintes pontos :

a) Baldini restringe a busca de códigos a uma estrutura definida por uma matriz multinível, encontrando códigos com uma ampla gama de comprimentos e ganhos com taxas de 1/2 e 2/3.

A busca dos códigos multiníveis propostos baseia-se na estrutura definida pela matriz T, também fornecendo códigos de ampla faixa de comprimentos e ganhos com taxa de 1/2; porém a busca binária pode ser mais rápida.

b) Os códigos multiníveis propostos possuem equivalentes encontrados por Baldini, como por exemplo, o primeiro e o quarto códigos da tabela 4.I comparados com o primeiro e o segundo códigos da tabela 3.V, respectivamente.

c) Os códigos de Baldini apresentam algumas propriedades, mas estas, aparentemente, não são suficientes para a determinação de um processo simples de decodificação para os mesmos.

Os códigos multiníveis propostos, por serem representados binariamente, são muito mais facilmente decodificados utilizando técnicas conhecidas de decodificação binária; esta é a grande vantagem destes em relação aos códigos de Baldini.

4.11 - CONCLUSÕES

Este capítulo propôs a especificação de um sistema de codificação e decodificação de códigos multiníveis através de sua representação binária. Buscou-se um processo de decodificação que unisse bom desempenho e baixa complexidade. Com o uso do Algoritmo Conjuntos de Informação, este compromisso foi alcançado. Coffey e Goodman [C&G 90] provaram, recentemente, que o ACI apresenta menor complexidade de decodificação quando comparado aos algoritmos de máxima verossimilhança e Algoritmos Vizinhos de Zero [L&H

85], este fato apóia a escolha deste método de decodificação.

O capítulo 5 toma como base os códigos multiníveis propostos para a especificação de um sistema de modulação 16-QAM codificada para implementação em um circuito integrado com finalidade didática. O diagrama em blocos do sistema é apresentado e são especificados e analisados os algoritmos de codificação e decodificação utilizados.

CAPÍTULO 5

ESPECIFICAÇÃO DE UM SISTEMA DE MODULAÇÃO 16-QAM CODIFICADA

Existem, no mercado internacional, circuitos integrados dedicados que realizam a decodificação de códigos convolucionais binários usando o algoritmo de Viterbi. Nos modem's de alta velocidade, que utilizam a modulação codificada, o algoritmo de Viterbi é implementado via "software" em um Processador Digital de Sinais. Um exemplo é a proposta de implementação apresentada por Messer [MES 89]. Além disso, os modem's utilizam um Processador Digital de Sinais exclusivo para a execução deste algoritmo a fim de alcançar alta velocidade [TAK 87].

Este capítulo apresenta uma proposta de implementação de um sistema de codificação e decodificação com códigos de bloco multiníveis para a modulação 16-QAM codificada em um circuito integrado. Os algoritmos de codificação e decodificação são especificados e analisados. Espera-se que o sistema integrado alcance alta velocidade de transmissão de bits por segundo (9600 bit/s). A razão para esta taxa é o uso de um dos algoritmos de decodificação binária para os códigos multiníveis apresentado no capítulo 4. O circuito utiliza o Algoritmo Conjuntos de Informação e opera a taxas de codificação de 3/4 e 1/2 com ganhos assintóticos de codificação de 3 e 0,97 dB sobre as modulações 8-QAM e 4-QAM, respectivamente. Este circuito integrado tem, principalmente, uma finalidade didática, podendo ser usado em laboratório para o desenvolvimento de experiências, para o ensino de modulação codificada, e, é claro, para certificar que os dados levantados por simulação correspondem à realidade.

5.1 - ESPECIFICAÇÃO DO SISTEMA 16-QAM

O sistema baseia-se na partição do espaço de sinais 16-QAM (figura 3.18) proposta por Baldini [BAL 89]; porém utiliza somente os códigos multiníveis propostos no capítulo 4. A cada sinal da modulação 16-QAM é atribuído um par de símbolos pertencente ao grupo de inteiros de 4 elementos. O símbolo à esquerda seleciona um dos subconjuntos e o símbolo à direita seleciona um dos pontos no subconjunto escolhido. Duas versões são propostas e analisadas. A primeira versão (V1) utiliza um código multinível $C(4;8,4^4,5)$ - código C1 representado pelas matrizes binárias da tabela 4.1 -, e um código multinível $C(4;8,4^8,1)$, formando o arranjo-código mostrado na figura 5.1.

$C(4;8,4^4,5)$
$C(4;8,4^8,1)$

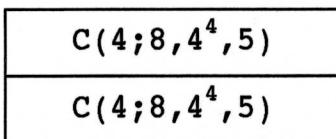
FIGURA 5.1 - ARRANJO-CÓDIGO PARA A VERSÃO 1

Cada coluna deste arranjo-código forma um par de símbolos que representam um ponto na constelação 16-QAM. O código, na primeira linha, protege os símbolos à esquerda na constelação. A taxa de codificação para esta versão é igual a $3/4$.

A segunda versão (V2) utiliza dois códigos multiníveis $C(4;8,4^4,5)$, formando o arranjo-código mostrado na figura 5.2. A taxa de codificação para esta versão é igual a $1/2$.

A distância Euclidiana mínima ao quadrado entre os arranjos-código formados é calculada pela equação :

$$dE_{\min}^2 = \min(dE_1^2, 4 \cdot dE_2^2) \quad (5.1)$$

**FIGURA 5.2 - ARRANJO-CÓDIGO PARA A VERSÃO 2**

Considerando que as distâncias Euclidianas dos códigos multiníveis são calculadas sobre a constelação 16-QAM com $\Delta_0 = 1$, este valor é :

$$\text{para V1 : } dE_{\min}^2 = \min(5,4) = 4$$

$$\text{para V2 : } dE_{\min}^2 = \min(5,20) = 5$$

Para a versão V1 o ganho assintótico sobre a modulação 8-QAM com $\Delta_0 = \sqrt{2}$ é igual a :

$$g_{\infty} = 10 \cdot \log_{10} (4/2) = 3 \text{ dB}$$

Para a versão V2 o ganho assintótico sobre a modulação 4-QAM com $\Delta_0 = 2$ é igual a :

$$g_{\infty} = 10 \cdot \log_{10} (5/4) = 0,97 \text{ dB}$$

5.2 - ESPECIFICAÇÃO DO CODIFICADOR

O diagrama em blocos do codificador do sistema 16-QAM codificado é mostrado na figura 5.3.

Quando o codificador opera à taxa $R = 3/4$, blocos de 24 bits são recebidos e separados em 8 mais 16 bits. Os 8 bits são codificados pelo codificador binário. Esta operação é uma simples multiplicação dos 8 bits pela matriz geradora T armazenada em uma memória. O resultado é uma palavra-código binária que representa a palavra-código multinível da primeira linha do arranjo-código da figura 5.1. Os 16 bits restantes representam uma palavra-código multinível da

segunda linha do arranjo-código. O mapeador transforma estas palavras-código em 8 blocos de 4 bits cada. Cada bloco corresponde a um dos pontos a serem transmitidos da constelação 16-QAM.

Quando o codificador opera à taxa $R = 1/2$, blocos de 16 bits são recebidos e separados em dois blocos de 8 bits cada. Neste caso, dois codificadores iguais realizam a codificação. O restante do processo é idêntico à primeira versão do sistema.

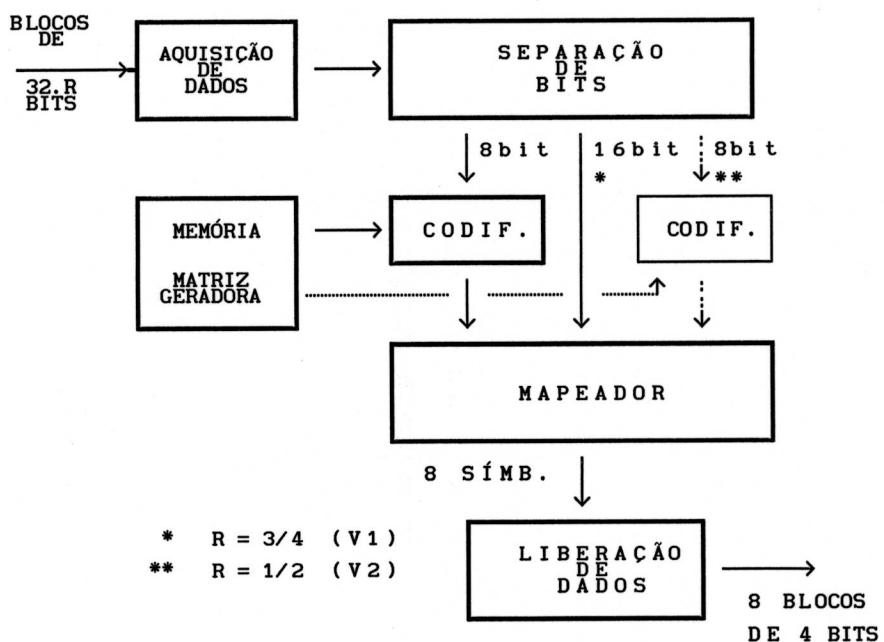


FIGURA 5.3 - DIAGRAMA EM BLOCOS DO SISTEMA CODIFICADOR

5.3 - ESPECIFICAÇÃO DO DECODIFICADOR

O diagrama em blocos do decodificador do sistema 16-QAM codificado é mostrado na figura 5.4.

Na recepção, o demodulador (não especificado neste trabalho) entrega ao decodificador 8 blocos de 4

bits, operando sobre o sinal analógico recebido. O "demapeador" transforma as componentes decididas em duas palavras binárias de 16 bits (PB1 e PB2). Quando o decodificador opera à taxa $R = 3/4$, apenas a palavra binária PB1 é decodificada pelo ACI. À taxa $R = 1/2$, as duas palavras sofrem decodificação.

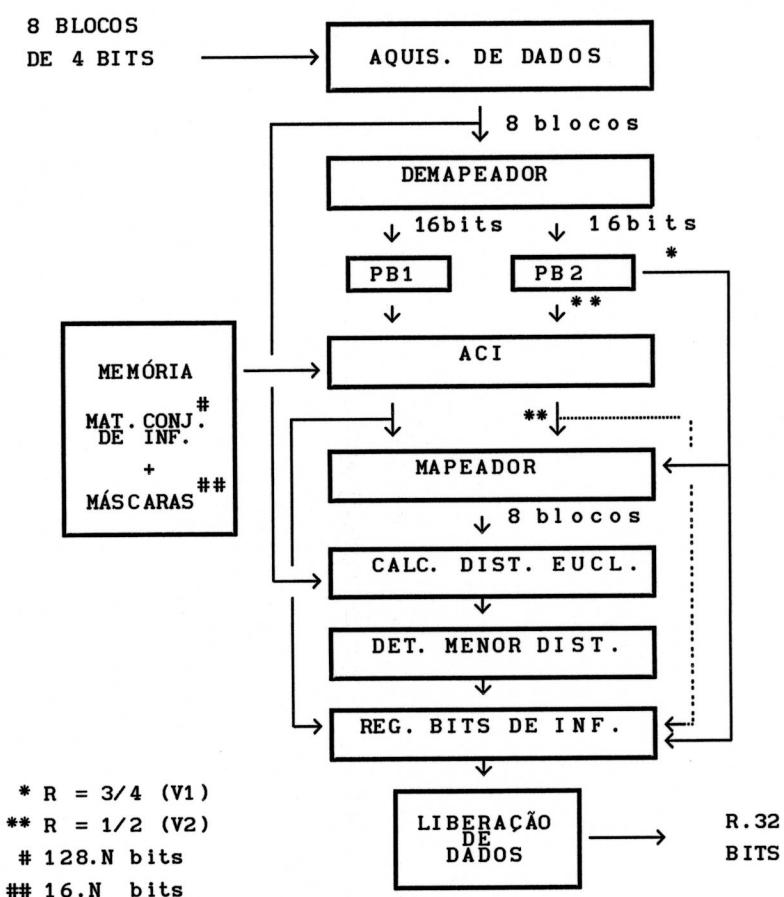


FIGURA 5.4 - DIAGRAMA EM BLOCOS DO DECODIFICADOR DO SISTEMA

O ACI faz uso das máscaras e das matrizes conjunto de informação armazenadas na memória para a decodificação. O algoritmo forma uma (duas à taxa $R = 1/2$) palavra(s) decodificada(s) de 16 bits que são mapeadas para possibilitar o cálculo de distância Euclidiana entre o bloco de bits decodificado e o bloco de bits recebido. Os 32.R bits de informação, provenientes das palavras binárias decodificadas,

são armazenados no Registro de Bits de Informação a comando do Determinador de Menor Distância.

5.3.1 - EXEMPLO DE CODIFICAÇÃO E DECODIFICAÇÃO

Este item apresenta um exemplo de codificação e de decodificação para o sistema operando à taxa $R = 3/4$. O bloco de aquisição de dados (figura 5.3) recebe blocos de $32.R$ bits, ou seja, 24 bits. Suponha-se que um bloco seja o seguinte:

10101010000000000000000000000000 (5.2)

Após o separador de bits, os 8 primeiros bits são codificados utilizando-se a matriz geradora T.

$$\begin{array}{l}
 10101010 \cdot \quad \left[\begin{array}{c} 1000000001111000 \\ 0100000011110001 \\ 0010000010111101 \\ 0001000010011011 \\ 0000100000000111 \\ 0000010000001111 \\ 0000001000001011 \\ 0000000100001001 \end{array} \right] = 1010101011001001
 \end{array}$$

A palavra codificada e os 16 bits remanescentes são dispostos em quatro palavras de 8 bits conforme a transformação (item 4.6) pelo mapeador :

$$[I_A : I_B : J_A : J_B] \longrightarrow \begin{matrix} [I_A : J_A] \\ [I_B : J_B] \end{matrix}$$

Então :

$$\begin{array}{ccc}
 & 10101100 & \\
 1010101011001001 \longrightarrow & 10101001 & (5.3) \\
 0000000000000000 & 00000000 & \\
 & 00000000 &
 \end{array}$$

A primeira e segunda linhas correspondem a uma palavra-código do código multinível. O mesmo vale para a terceira e quarta linhas, formando, então, o arranjo-código multinível. O conjunto de bits de cada coluna deste arranjo, corresponde a um par de símbolos, ou seja, um ponto na constelação 16-QAM (a correspondência entre bits e símbolos foi apresentada no item 4.1 e 4.2). Cada coluna é liberada para o modulador que realiza a transmissão pela linha.

Considerando $\Delta_0 = 2$, na constelação 16-QAM (figura 5.5) os componentes transmitidos são os seguintes :

$$\begin{array}{cccccccccc} x_t & -3 & -3 & -3 & -3 & -3 & -1 & -3 & -1 \\ y_t & 1 & 3 & 1 & 3 & 1 & 3 & 3 & 1 \end{array}$$

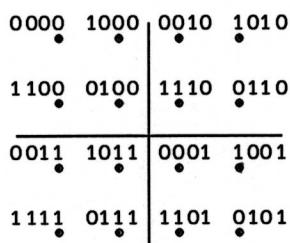


FIGURA 5.5 - CONSTELAÇÃO 16-QAM

Após a transmissão, os componentes recebidos, alterados pelo ruído, podem ser :

$$\begin{array}{cccccccccc} x_r & -3,1 & -2,8 & -2,5 & -1,9 & -3,2 & -0,8 & -3,0 & -2,1 \\ y_r & 1,9 & 3,4 & 0,5 & 2,5 & 0,9 & 2,4 & 3,5 & 1,0 \end{array}$$

O demodulador determina o ponto da constelação mais próximo do ponto recebido. Após este processo, tem-se os seguintes componentes decididos, entregues ao decodificador na forma binária :

$$\begin{array}{ccccccccccccc}
 x_d & -3 & -3 & -3 & -1 & -3 & -1 & -3 & -3 & \longrightarrow & 10111101 \\
 y_d & 1 & 3 & 1 & 3 & 1 & 3 & 3 & 1 & \longrightarrow & 10101001 \\
 & & & & & & & & & & 00000000 \\
 & & & & & & & & & & 00000000
 \end{array}$$

O "demapeador" transforma estes componentes diretamente em duas palavras binárias de 16 bits (PB1 e PB2).

1011101011011001
0000000000000000

O passo seguinte é a decodificação ACI. Apenas a primeira linha sofre decodificação pelo Algoritmo Conjuntos de Informação, nesta versão.

Neste algoritmo os bits, da palavra binária que se encontram nas mesmas posições dos bits unitários da máscara, multiplicam a matriz conjunto de informação associada à esta máscara. Isto corresponde a uma recodificação da palavra binária e é realizada para cada uma das N matrizes conjunto de informação que formam o conjunto de cobertura. Admitindo-se que algumas destas operações já tenham sido realizadas e que, neste passo, a matriz conjunto de informação (associada à máscara 0000111100001111) seja a seguinte, os bits apropriados da primeira linha a multiplicam :

$$10101001 \cdot \begin{bmatrix} 0111100011010000 \\ 1111010010100000 \\ 0001001010010000 \\ 1100000110000000 \\ 1000000001111000 \\ 1110000000110100 \\ 1101000000010010 \\ 0100000011110001 \end{bmatrix} = 1010101011001001$$

Para o cálculo da distância Euclidiana, deve-se mapear a arranjo-código assim formado para o espaço de sinais 16-QAM. Assim, o mapeador representa os 32 bits, em 4 palavras de 8 bits. Cada coluna representa um ponto na constelação

16-QAM.

	10101100
1010101011001001 →	10101001
0000000000000000 →	00000000
	00000000

A distância Euclidiana ao quadrado pode ser calculada pela fórmula :

$$dE^2 = \sum_{i=1}^{8} ((x-x_d)_i^2 + (y-y_d)_i^2) \quad (5.4)$$

ou através de uma tabela de distâncias entre pontos da constelação 16-QAM.

Onde x_d e y_d são as coordenadas em fase e quadratura que correspondem às componentes binárias decididas, e x e y são as coordenadas em fase e quadratura que correspondem às componentes decodificadas. Para este passo $dE^2 = 8$.

O algoritmo varre todas as matrizes conjunto de informação e armazena os bits de informação que correspondem ao arranjo-código decodificado de mínima distância Euclidiana. Após a decodificação os dados são liberados ao usuário.

5.4 - ANÁLISE DOS ALGORITMOS DE CODIFICAÇÃO E DECODIFICAÇÃO

Os algoritmos especificados nesse item foram usados no desenvolvimento de um circuito integrado de codificação e decodificação pelo LAC (Laboratório Central de Eletrotécnica e Eletrônica - COPEL/ UFPR). Para a validação do sistema, o LAC usou o programa HDC (Hardware Description in C - [MAR 90]). No HDC descrevem-se as partes operativa e de controle de um sistema eletrônico em forma de algoritmos na linguagem C, o que possibilita a verificação do funcionamento de um

sistema sem a sua implementação física. As máscaras que definem o CI, são obtidas numa segunda etapa.

Neste item a complexidade dos algoritmos é calculada para possibilitar uma pequena análise.

5.4.1 - COMPLEXIDADE DE ALGORITMOS

A complexidade de um algoritmo $C_{ALG}(h)$ é, normalmente, definida como uma medida do tempo requerido pelo mesmo para solucionar um problema de tamanho h , ou uma medida do espaço de memória necessário para esta execução [KRO 87]. Em geral, o tempo de execução aumenta com o tamanho do problema. Se h é pequeno, algoritmos ineficientes não tomarão muito tempo de execução. Se h aumenta, existirão situações em que o algoritmo levará um período inaceitável de tempo para solucionar o problema. O comportamento assintótico da complexidade determina o tamanho do problema que pode ser resolvido pelo algoritmo, ou seja, qual o maior valor de h .

A complexidade de um algoritmo é dita $O(h^2)$, se, para um problema de tamanho h , o tempo de execução deste é proporcional a h^2 . Em outras palavras, quando se diz que $C_{ALG}(h)$ é $O(h^2)$, significa que existem as constantes positivas c e h_0 tal que, para h igual ou maior que h_0 , tem-se $C_{ALG}(h) \leq c \cdot h^2$. Algoritmos ótimos são aqueles de complexidade $O(\log h)$ [KRO 87].

A seguir será considerado que o tamanho do problema (h) é igual ao comprimento do código multinível (n).

5.4.2 - ALGORITMOS DO SISTEMA DE CODIFICAÇÃO

Com base na figura 5.3, o sistema de codificação é definido pelo algoritmo SIST-COD, para a taxa de codificação $R = 3/4$.

```

SIST-COD()           // sistema codificador //
Entrada : inf[3n]    // informação binária //
Saída :   ponto[n][4] // ponto da constelação //
{
  for (i = 0; i < n; i++) aux[i] = inf[i];
  for (i = 0; i < 2n; i++) pb[i] = inf[i+8];
  COD(aux,pa);
  MAP(pa,pb,ponto);
}

```

Os dois primeiros laços representam a separação dos bits de informação para posterior codificação pelo algoritmo COD, e mapeamento pelo algoritmo MAP.

Para a taxa de codificação $R = 1/2$, o algoritmo SIST-COD é :

```

SIST-COD()           // sistema codificador //
Entrada : inf[2n]    // informação binária //
Saída :   ponto[n][4] // ponto da constelação //
{
  for (i = 0; i < n; i++)
  {
    aux1[i] = inf[i];
    aux2[i] = inf[i+8];
  }
  COD(aux1,pa);
  COD(aux2,pb);
  MAP(pa,pb,ponto);
}

```

O algoritmo de codificação (figura 5.3) pode ser implementado de duas formas. A primeira é a multiplicação clássica de vetor por matriz, dada pela equação :

$$v_i = \sum_{i=1}^{2n} \sum_{j=1}^n u_i \cdot g_{ji} \quad (5.5)$$

onde : v_i = bits da palavra-código;
 u_i = bits do vetor informação;
 g_{ji} = bits da matriz geradora do código.

O tempo de execução deste algoritmo é proporcional a $2n^2$; portanto, a sua complexidade é $O(n^2)$. A segunda forma de implementação, dada pelo algoritmo COD, baseia-se no fato de que basta somar (módulo dois) linhas determinadas da matriz geradora conforme os bits do vetor de informação. A soma módulo dois nada mais é do que a operação ou-exclusivo bit a bit.

```

COD()           // codificador binário //
Entradas : inf[n]      // informação binária //
                  g[n][2n]    // matriz geradora //
Saída :   pc[2n]       // palavra-código //
{
  for (i = 0; i < 2n; i++) pc[i] = 0;
  for (i = 0; i < n; i++)
    if (inf[i] == 1)
      for (j = 0; j < 2n; j++) pc[j] = pc[j] ^ g[i][j];
}

```

Observa-se que a execução do laço mais interno depende do valor do vetor *inf* (informação binária de *n* bits). No melhor caso, esse laço não será executado; no pior caso será executado *n* vezes, e, na média, ele será executado *n/2* vezes, já que o número de bits unitários em *inf* é igual, na média, ao número de bits zeros. Assim, o tempo médio de execução desse algoritmo é proporcional a n^2 . Sua complexidade é $O(n^2)$. Este é o algoritmo a ser implementado, pois é duas

vezes mais rápido que o algoritmo anterior, embora suas complexidades, para ambas, seja $O(n^2)$.

O algoritmo MAP realiza o mapeamento das palavras codificadas para a constelação 16-QAM. Sua complexidade é $O(n)$.

Então, a complexidade final do algoritmo de codificação SIST_COD é a maior complexidade calculada, ou seja, $O(n^2)$, para as taxas $R = 3/4$ e $R = 1/2$.

5.4.3 - ALGORITMOS DO SISTEMA DE DECODIFICAÇÃO

Com base na figura 5.4, o sistema de decodificação é definido pelo algoritmo SIST-DEC para taxa de codificação $R = 3/4$.

```

SIST-DEC()           // sistema decodificador //
Entrada : ponto_d[n][4] // n pontos decididos da //
                           // constelação //
Saídas :   inf[3n]      // inf. binária decodif. //
Inicialização : dmin = 255
{
  DEMAP(ponto_d,pa,pb);
  for (i = 0; i < N1; i++)
  {
    ACI(pa,i,d);
    MAP(d,pb,ponto_c);
    de = DIST(ponto_d,ponto_c);
    if (de < dmin)
    {
      dmin = de;
      for (j = 0; j < n; j++) inf[j] = d[j];
    }
  }
  for (i = 0; i < 2n; i++) inf[i+n] = pb[i];
}

```

Observe-se que, em essência, este é um algoritmo de busca do mínimo. No melhor caso, a condição $de < dmin$ será válida uma vez; no pior caso, N_1 vezes; e na média ela será válida aproximadamente $\ln(N_1)$ vezes [KNU 73]; onde N_1 é o número de matrizes conjunto de informação.

Para a taxa de codificação $R = 1/2$, o algoritmo SIST-DEC é mostrado abaixo, em que N_1 e N_2 é o número de matrizes conjunto de informação usadas na decodificação da primeira e segunda linhas do arranjo-código (figura 5.1 e 5.2), respectivamente.

```

SIST-DEC()           // sistema decodificador //
Entrada : ponto_d[n][4] // n pontos decididos da //
                           // constelação //
Saída :   inf[2n]       // inf. binária decodif. //
Inicialização : dmin = 255
{
  DEMAP(ponto_d,pa,pb);
  for (j = 0; j < N1; j++)
  {
    ACI(pa,j,d1);
    for (i = 0; i < N2; i++)
    {
      ACI(pb,i,d2);
      MAP(d1,d2,ponto_c);
      de = DIST(ponto_d,ponto_c);
      if (de < dmin)
      {
        dmin = de;
        for (k = 0; k < n; k++)
        {
          inf[k] = d1[k]; inf[k+n] = d2[k];
        }
      }
    }
  }
}

```

O algoritmo DEMAP tem complexidade igual a $O(n)$.

O Algoritmo Conjuntos de Informação, mostrado abaixo, também baseia-se na multiplicação de vetores por matrizes; portanto, o mesmo princípio usado para o algoritmo COD é usado.

Observe-se que se deve escolher k dos n bits das palavras recebidas para a multiplicação pelas matrizes conjunto de informação. Essa operação é realizada pelo teste "masc[k] = 0". O pior caso ocorre quando a condição "pr[k] = 1" é sempre verdadeira e o laço "while" é realizado $2n$ vezes. A complexidade média é $O(n^2)$.

```

ACI()           // alg. conj. de informação //
Entradas : pr[2n]    // palavra binária recebida //
                  // matriz conj. de informação //
                  // máscara do conj. de cobertura //
Saída  : pd[2n]    // palavra binária decodificada //
{
  for (i = 0; i < 2n; i++) pd[i] = 0;
  k = 0;
  for (i = 0; i < n; i++)
  {
    while (masc[k] == 0) k++;
    if(pr[k] == 1)
      for (j = 0; j < 2n; j++) pd[j] = pd[j] ^ mat[i][j];
    k++;
  }
}

```

O algoritmo DIST faz o cálculo da distância Euclidiana. Há duas maneiras de implementá-lo. A primeira corresponde à representação de cada ponto decidido (ponto_d) e cada ponto decodificado (ponto_c) em coordenadas da constelação 16-QAM e na aplicação da fórmula de distância

entre pontos, o que envolve a utilização de operações de multiplicação no algoritmo. A segunda implementação faz uso de uma tabela endereçada por `ponto_d` e `ponto_c`. Esta é a opção escolhida. Sua complexidade é $O(n)$.

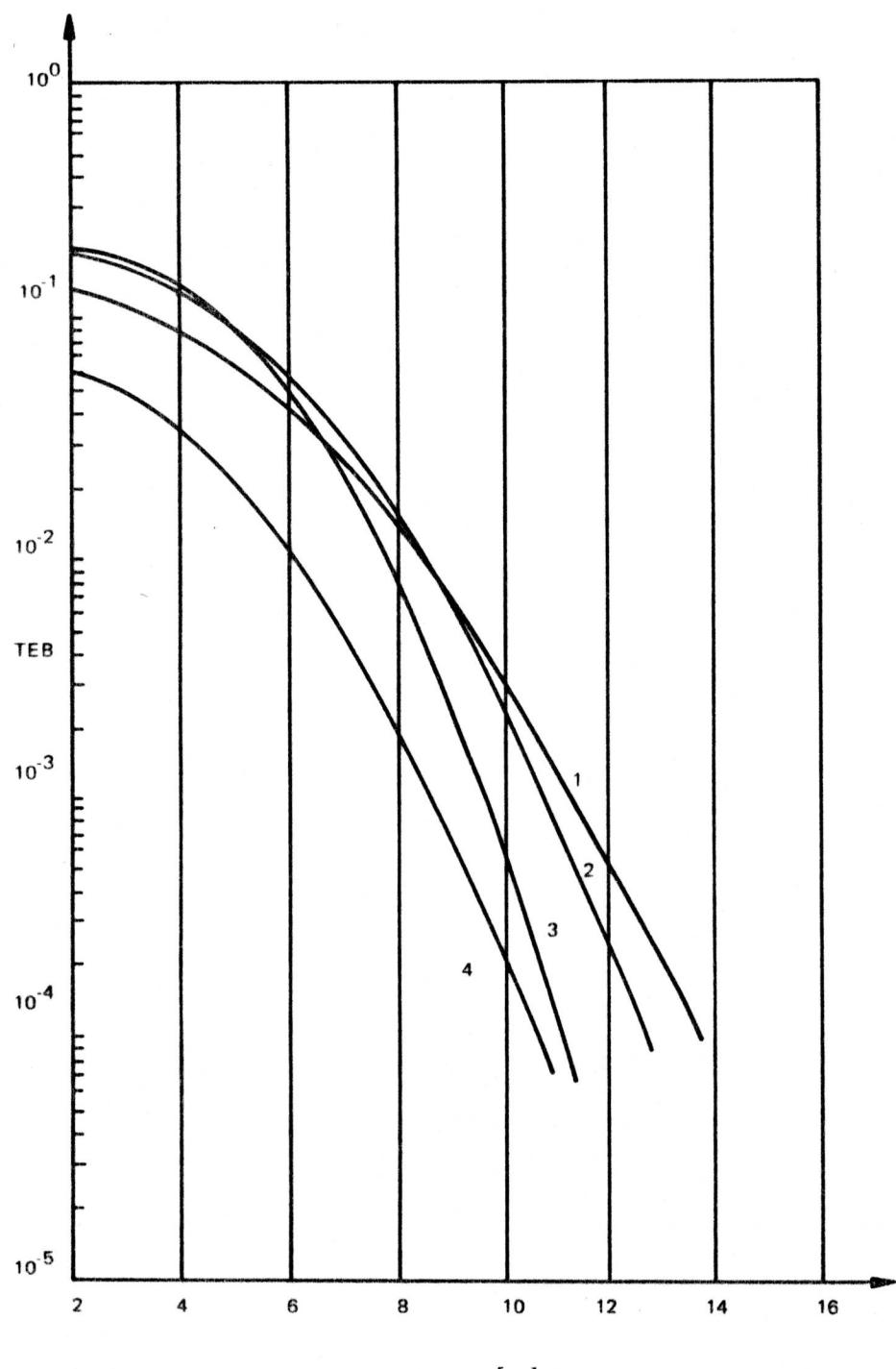
Finalmente, a complexidade para o sistema de decodificação é dada por $O(n^2 \cdot N_1)$ e $O(n^2 \cdot N_1 \cdot N_2)$, para as taxas de codificação $R = 3/4$ e $R = 1/2$, respectivamente, pois o tempo de execução, no pior caso, é proporcional a $n + N_1 \cdot (n^2 + 3n)$, para a taxa $R = 3/4$, e proporcional a $n + n^2 \cdot N_1 + N_1 \cdot N_2 \cdot (n^2 + 3n)$, para a taxa $R = 1/2$. A complexidade em espaço para ambas as versões é $O(N)$, pois a memória necessária é proporcional a N , em que N é o número de matrizes conjunto de informação necessários à decodificação.

Observa-se que, como esperado, o tempo de decodificação, para os dois casos, aumenta com o comprimento da palavra-código multinível e com o número de matrizes conjunto de informação utilizados para a decodificação. A quantidade de memória necessária aumenta linearmente com o número de matrizes conjunto de informação. O algoritmo de decodificação especificado é de complexidade inferior ao algoritmo de correlação completa (MLD), já que este último apresenta complexidade $O(q^k)$ [BER 78], onde q é a cardinalidade do alfabeto de modulação e k é o número de símbolos de informação.

5.5 - DESEMPENHO DO SISTEMA DE DECODIFICAÇÃO

O sistema de decodificação especificado, através dos algoritmos mostrados, foi simulado para o levantamento das curvas de desempenho, segundo o princípio apresentado no item 4.4.

As curvas de desempenho para o sistema com taxas de codificação $R = 3/4$ e $R = 1/2$, são mostradas na figura 5.6.



Eb / No [dB]

- 1 - MODULAÇÃO 8-QAM NÃO CODIFICADA
- 2 - 16-QAM CODIFICADA ($R = 3/4$) (V1)
- 3 - 16-QAM CODIFICADA ($R = 1/2$) (V2)
- 4 - MODULAÇÃO 4-QAM NÃO CODIFICADA

FIGURA 5.6 - DESEMPENHO DO SISTEMA DE DECODIFICAÇÃO 16-QAM

Estas curvas são comparadas com as curvas de desempenho para as modulações 8-QAM e 4-QAM não codificadas, respectivamente. À Taxa de Erro de Bit igual a 10^{-4} , a primeira versão do sistema ($R = 3/4$) apresenta um ganho em codificação de aproximadamente 1,0 dB em relação ao sistema 8-QAM não codificado. O ganho assintótico teórico calculado é igual a 3 dB. No caso da segunda versão ($R = 1/2$) observa-se que seu desempenho começa a tornar-se melhor que o desempenho da modulação 4-QAM a partir de uma relação sinal-ruído de 12 dB. Este comportamento é explicado em função do baixo ganho assintótico (0,97 dB) previsto para esta versão.

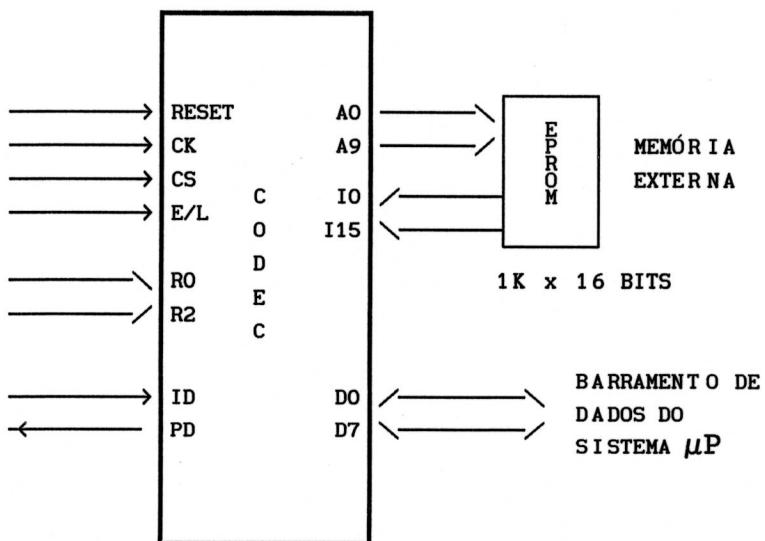
Do exposto torna-se evidente que a versão V1 do sistema apresenta vantagens sobre a versão V2, uma vez que, a primeira, além de permitir um ganho em codificação maior do que a última em relação aos respectivos sistemas não codificados de referência, apresenta uma complexidade em decodificação menor.

5.6 - CIRCUITO INTEGRADO PARA CODIFICAÇÃO E DECODIFICAÇÃO

O circuito integrado para codificação e decodificação realiza as operações necessárias conforme os diagramas das figuras 5.3 e 5.4, trabalhando apenas com as informações binárias. Os processos de modulação e demodulação não são realizados por este circuito. A pinagem do circuito integrado é mostrado na figura 5.7.

Durante a operação do sistema decodificador ACI, este constantemente acessa as informações armazenadas numa memória externa. A memória (EPROM) está dividida em quatro regiões nas quais estão armazenadas a matriz geradora do código em uso, as matrizes conjunto de informação, os conjuntos de máscaras associadas e uma tabela de distâncias entre os pontos da constelação 16-QAM. O uso da EPROM externa deve-se a dois fatores :

- a falta de área de silício para a implementação do circuito propriamente dito mais uma memória,
- a possibilidade de troca do código em uso.



**FIGURA 5.7 - CIRCUITO INTEGRADO PARA CODIFICAÇÃO
E DECODIFICAÇÃO**

Como o ACI é um algoritmo de decodificação completamente independente do código binário utilizado, pode-se alterar o código (alterando os dados da EPROM) sem modificações tanto no circuito codificador quanto no decodificador. Isto representa uma vantagem.

A EPROM é endereçada pelos pinos A_0 a A_9 , sendo que os pinos A_8 e A_9 determinam qual região da memória é acessada. Os dados são transportados ao CI pelos pinos I_0 a I_{15} . Os terminais D_0 a D_7 correspondem ao barramento bidirecional de dados. O circuito possui 6 registradores internos de 8 bits cada. Quatro desses são endereçados (através do barramento R_i) para receber a informação a ser codificada e para fornecer (pelo barramento bidirecional) os dados processados. Os dois registradores restantes recebem os valores do número de conjuntos de informação usados na

decodificação (N_1 e N_2 , do item 5.5.3). A escrita e leitura nos registradores é controlada pelo terminal E/L.

Os dados a serem codificados (por exemplo equação 5.2) são divididos em blocos de 8 bits e carregados em três (dois à taxa $R = 1/2$) dos quatro registradores do sistema. Após a codificação, cada duas colunas da informação codificada (ver equação 5.3) forma uma palavra de 8 bits. Cada palavra é armazenada em um dos quatro registradores, para ser liberada. Para a decodificação, processo inverso é realizado.

O circuito possui dois "flip-flop's" que devem ser ativados pelo sistema externo para estabelecer a taxa de codificação ($R = 3/4$ ou $R = 1/2$) e o modo de operação (Codificador ou Decodificador). A interação com o sistema microprocessado externo se faz através dos pinos ID e PD. O pino ID permite informar que os dados foram processados, e o pino PD recebe informação de que o microprocessador está pronto para receber estes dados.

O pino RESET permite a reinicialização do sistema; o terminal CS, quando ativo, habilita o CI, e quando inativo, o barramento de dados permanece em "tri-state". O terminal CK é a entrada do relógio externo.

Finalmente, ressalta-se que o circuito pode funcionar em várias configurações distintas que são definidas pelos valores do número de matrizes conjunto de informação usadas na decodificação e pela taxa de codificação, permitindo vários níveis de desempenho versus velocidade de decodificação.

5.7 - CONCLUSÕES

Este capítulo apresentou a especificação de um sistema de codificação e decodificação para a modulação 16-QAM codificada. O sistema pode trabalhar com duas taxas de

codificação : $R = 3/4$ e $R = 1/2$. Foram especificados e analisados os algoritmos que definem tanto codificação quanto decodificação e que constituem a informação fundamental para se obter o circuito integrado especificado. Estes algoritmos apresentam complexidade polinomial. A validação dos algoritmos foi realizada através de sua simulação, de onde foram levantadas as curvas de desempenho para as duas versões do sistema. Conclusões preliminares fornecidas pelo LAC, levantadas na fase inicial de desenvolvimento do CI, a respeito de seu desempenho, indicam que o gargalo do sistema encontra-se no acesso às informações contidas na memória externa. Assim se prevê que o sistema não alcançará a velocidade de 9600 bit/s (com clock de 10 MHz) quando operando à taxa $R = 1/2$, mas, à taxa $R = 3/4$, esta velocidade será possível. Observa-se que essas limitações são decorrentes da tecnologia CMOS empregada para a fabricação do CI. O uso de tecnologias mais rápidas permitirá um aumento de velocidade de decodificação.

CAPÍTULO 6

BUSCA DE CONJUNTOS DE COBERTURA E DE CÓDIGOS

Neste capítulo são apresentadas especificações de algoritmos para auxiliar a busca de Conjuntos de Cobertura utilizados no Algoritmo Conjuntos de Informação e para a busca de melhores códigos multiníveis.

Um conjunto de cobertura ótimo é aquele que cobre todos os padrões de um determinado número de erros, utilizando o menor número de máscaras possível. Um processo automatizado que auxilie a busca destes conjuntos, torna-se necessário.

Os códigos multiníveis propostos no capítulo 4 são ótimos em distância Euclidiana, mas o código binário (matriz T) que representa o código multinível, não é ótimo em distância de Hamming. Como os códigos multiníveis propostos são decodificados binariamente é necessário selecionar códigos, que além de possuirem distâncias Euclidianas maximizadas, apresentem as maiores distâncias de Hamming possíveis.

Estas otimizações melhoram o desempenho e a velocidade dos sistemas propostos, pois, com códigos melhores, maior quantidade de erros será corrigida e, com o uso de conjuntos de cobertura com número pequeno de máscaras, maior será a velocidade na execução do algoritmo e consequentemente da decodificação.

6.1 - BUSCA DE CONJUNTOS DE COBERTURA

Quando se utiliza o Algoritmo Conjuntos de Informação para o processo de decodificação, duas questões são de grande importância. A primeira refere-se ao número mínimo de máscaras necessárias para cobrir todos os padrões de um determinado número de erros. A segunda pergunta é como determinar estas máscaras. Infelizmente não há uma solução completamente satisfatória a essas perguntas. A fórmula de Schönheim (equação 4.14) não leva em consideração que cada máscara pertencente ao conjunto de cobertura deve, também, corresponder a uma matriz conjunto de informação. Isto torna o problema mais difícil e ainda em aberto [C&C 81]. Pode-se encontrar, é claro, resposta a estas questões que são aceitáveis do ponto de vista da engenharia. Os algoritmos de busca de conjuntos de cobertura, propostos neste capítulo, propõem-se a responder a segunda pergunta.

O processo mais simples para formar um conjunto de cobertura, para ser utilizado pelo Algoritmo Conjuntos de Informação, consiste em se buscar o maior número de máscaras válidas possível, ou seja, aquelas que correspondem a uma matriz conjunto de informação, e selecionar a menor quantidade destas de modo a cobrir todos os padrões de certo número de erros. Esse processo pode ser realizado manualmente para padrões de dois erros. Acima deste valor, o processo torna-se proibitivo, dada a complexidade em se calcular o número de padrões de erro que determinada máscara cobre, quando adicionada a um conjunto de cobertura em formação.

A seguir são especificados vários algoritmos, apresentados na forma de fluxograma, para auxílio à obtenção de conjuntos de cobertura.

6.1.1 - ALGORITMO DE VERIFICAÇÃO DE MÁSCARA VÁLIDA

Este algoritmo, denominado AI, permite verificar se

uma máscara é válida, determinando a matriz conjunto de informação associada neste caso. Para que uma máscara seja válida, é necessário que, para as k posições de bits unitários daquela, exista um conjunto de informação. Um conjunto de informação é a região de uma palavra-código que contém os bits de informação. Uma matriz conjunto de informação é formada a partir da matriz geradora do código, através de operações elementares e que possue k colunas de peso unitário. Obrigatoriamente, em códigos de bloco lineares, cada linha desta matriz é uma palavra-código do código utilizado.

Esta última afirmação facilita a verificação e formação da matriz conjunto de informação, pois basta verificar, para cada bit unitário da máscara, se existe uma palavra-código que possui um bit unitário nesta posição e zeros nas demais posições onde a máscara possui bits unitários. Essa palavra código será, então, uma das linhas da matriz conjunto de informação.

Por exemplo, considere-se a verificação da máscara 00001111. O quinto bit é um bit unitário. A palavra-código 11001000 possui um bit unitário nesta posição e zeros nas demais posições unitárias da máscara; portanto essa palavra-código forma uma linha da matriz conjunto de informação. O algoritmo é apresentado no fluxograma da figura 6.1, e explicado a seguir.

Uma máscara só será válida se todos os seus bits forem testados. Quando um bit da máscara é igual a "1", inicia-se a busca por uma palavra-código que satisfaça as condições anteriormente mencionadas. Caso todas as palavras-código tenham sido testadas e não se determinou uma linha para a matriz conjunto de informação, a máscara é considerada inválida e o algoritmo termina. Quando as condições são satisfeitas, a palavra-código é armazenada como uma linha da matriz conjunto de informação, e o algoritmo continua.

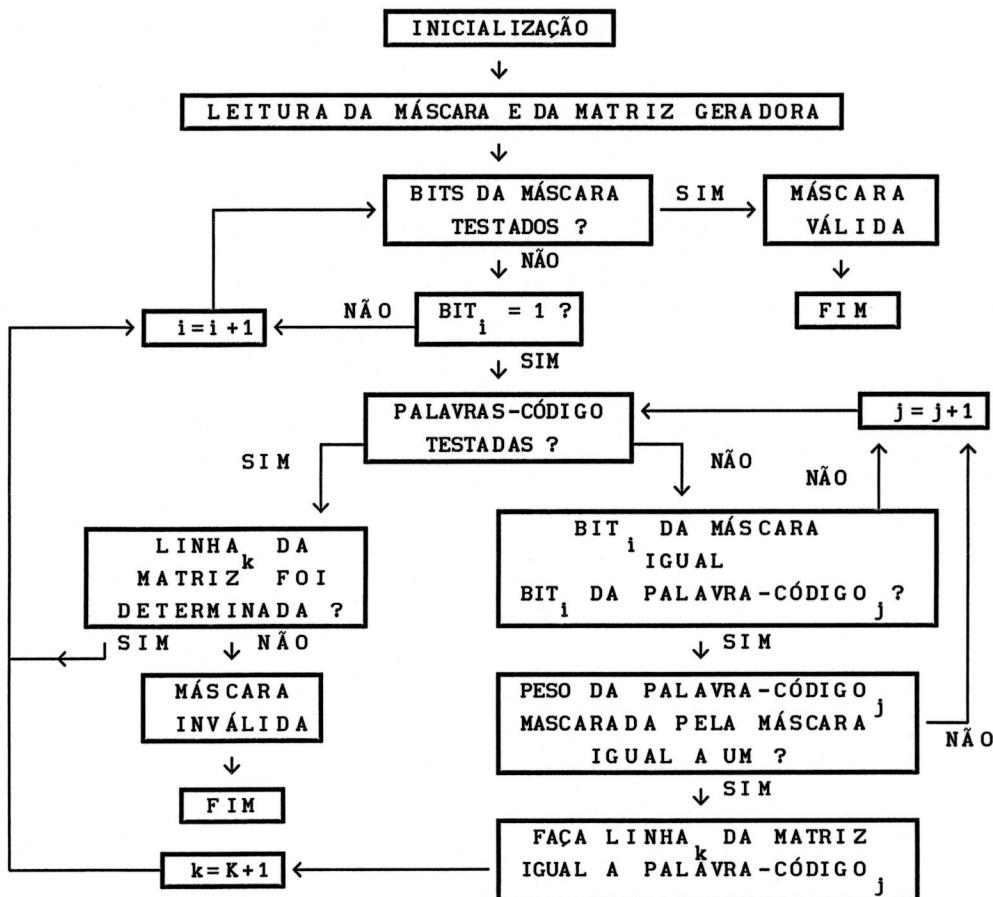


FIGURA 6.1 - ALGORITMO DE VERIFICAÇÃO DE MÁSCARA (AI)

6.1.2 - ALGORITMO DE DETERMINAÇÃO DE PADRÕES NÃO COBERTOS

Um passo para a formação de um conjunto de cobertura é saber quantos padrões de erro esse conjunto cobre, ou quais os padrões não cobertos, como neste algoritmo. Dada uma série de máscaras que compõem o conjunto de cobertura, o Algoritmo denominado AII determina quais os padrões de t erros que não são cobertos por esse conjunto. O fluxograma desse algoritmo é apresentado na figura 6.2.

Seu funcionamento é o seguinte, enquanto todos os padrões de t erros não tiverem sido testados, o algoritmo verifica se determinada máscara do conjunto de cobertura cobre certo padrão de erro. Se essa condição for verdadeira,

um novo padrão de erro é verificado. Caso todas as máscaras do conjunto de cobertura tenham sido testadas e um certo padrão não foi coberto, este é armazenado e corresponde a um dos dados de saída do algoritmo.

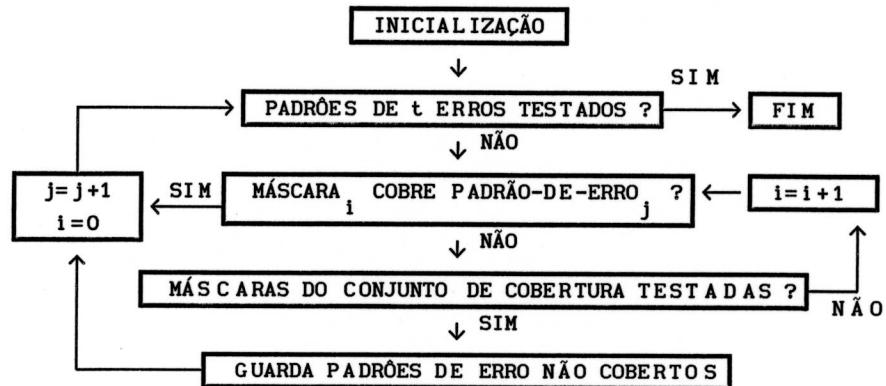


FIGURA 6.2 - DETERMINAÇÃO DE PADRÓES NÃO COBERTOS (AII)

6.1.3 - ALGORITMO DE PRÉ-SELEÇÃO DE MÁSCARAS

Esse algoritmo denominado AIII escolhe máscaras, de um Banco de Máscaras Válidas, que cobrem os padrões de t erros não cobertos, determinados pelo Algoritmo AII. O algoritmo realiza uma primeira seleção de máscaras e é apresentado no fluxograma da figura 6.3.

Para todos os padrões de t erros não cobertos e todas as máscaras de um banco de máscaras válidas, o Algoritmo AIII verifica se certa máscara do banco cobre determinado padrão de erro não coberto. Caso a máscara satisfaça a condição, verifica-se se ela já pertence ao conjunto de máscaras escolhidas. Caso contrário, essa máscara é armazenada. O conjunto formado representa uma primeira seleção de máscaras candidatas ao conjunto de cobertura que está sendo formado.

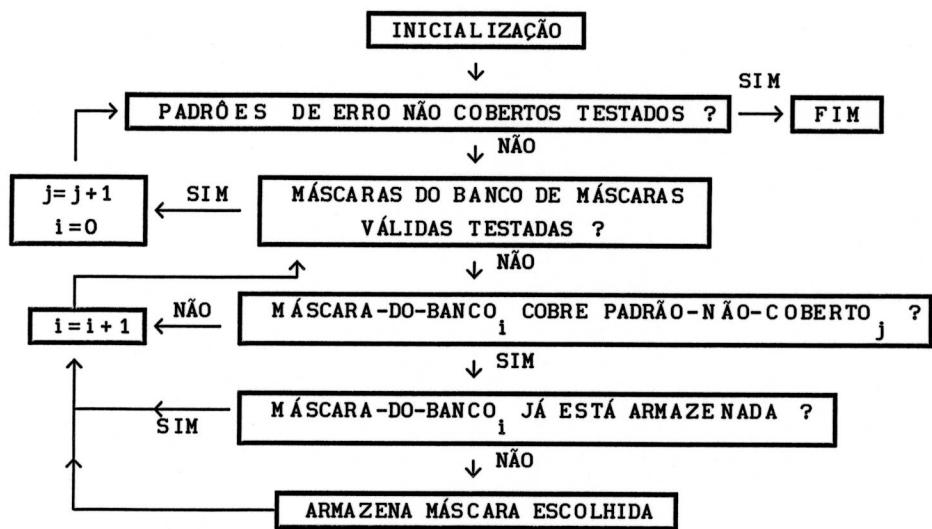


FIGURA 6.3 - ESCOLHA DE MÁSCARAS PARA COBRIR PADRÕES NÃO COBERTOS (AIII)

6.1.4 - ALGORITMO DE ESCOLHA DE MÁSCARAS DE MAIOR COBERTURA

Para a formação do conjunto de cobertura, deve-se selecionar, entre as máscaras escolhidas pelo Algoritmo AIII, aquelas que permitam a maior cobertura de padrões possível. Essas máscaras são, então, consideradas máscaras ótimas.

A este critério de seleção pode-se, também, acrescentar o critério de distância da máscara ótima para as máscaras do conjunto de cobertura em formação [C&C 81], como auxílio à escolha das máscaras que serão acrescentadas ao conjunto.

O algoritmo denominado AIV determina o número de padrões de erro cobertos pela máscara escolhida, escolhe como máscaras ótimas aquelas de maior cobertura e calcula a distância de cada máscara ótima em relação às máscaras do conjunto de cobertura em formação. Esse algoritmo é apresentado no fluxograma da figura 6.4.

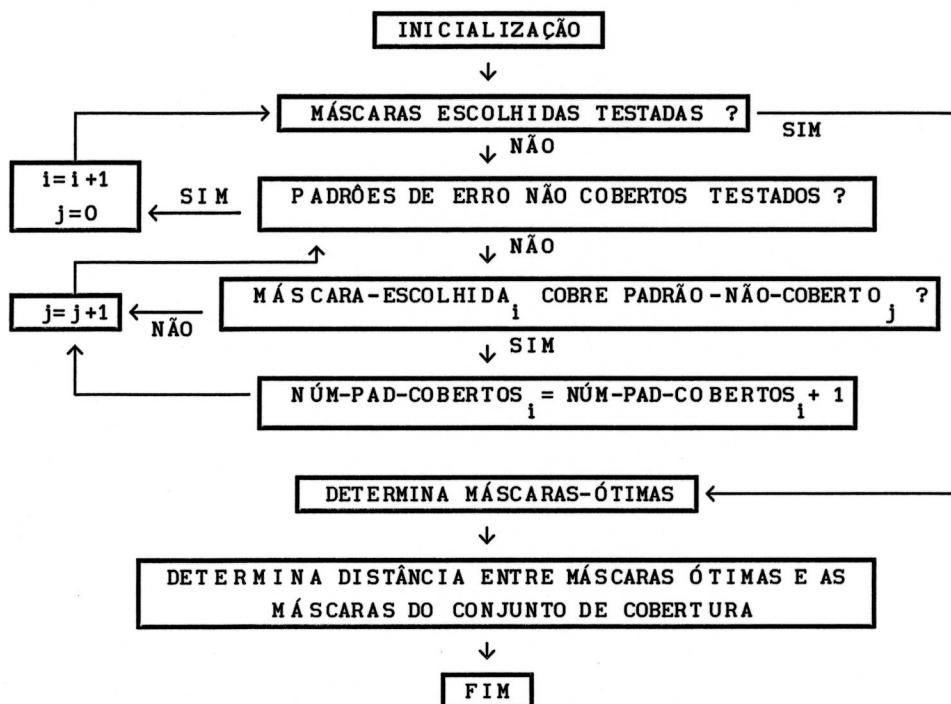


FIGURA 6.4 - ALGORITMO DE SELEÇÃO (AIV)

6.1.5 - ALGORITMO DE FORMAÇÃO DE BANCO DE MÁSCARAS

Esse algoritmo denominado AV é utilizado para determinar um grupo de máscaras válidas que cobrem padrões de t ou mais erros em posições pré-definidas de uma palavra. Por exemplo, admitindo que as posições pré-definidas são fornecidas pelo vetor 00110011 e que se buscam máscaras para cobrir padrões de dois ou mais erros, o Algoritmo AV incluiria, no grupo de máscaras válidas, a máscara 11000101, pois ela cobre um padrão de três erros nas posições unitárias do vetor de posições pré-definidas. A figura 6.5 apresenta o fluxograma desse algoritmo.

Uma Máscara Candidata é qualquer vetor de comprimento n que possua k bits unitários. Para que uma máscara pertença ao conjunto de cobertura em formação, ela deve cobrir t ou mais erros nas posições unitárias do vetor de posições pré-definidas. Este é o próximo teste realizado pelo algoritmo. O Algoritmo AV guarda apenas uma máscara para cada

padrão de erro coberto. Então, é necessário verificar se alguma Máscara Equivalente (aquele que cobre o mesmo padrão de erro que a máscara em teste cobre) já foi armazenada. Assim, não são armazenadas máscaras inúteis. Finalmente, se a máscara candidata for válida, o que é verificado pelo Algoritmo AI, ela é armazenada no banco de máscaras válidas.

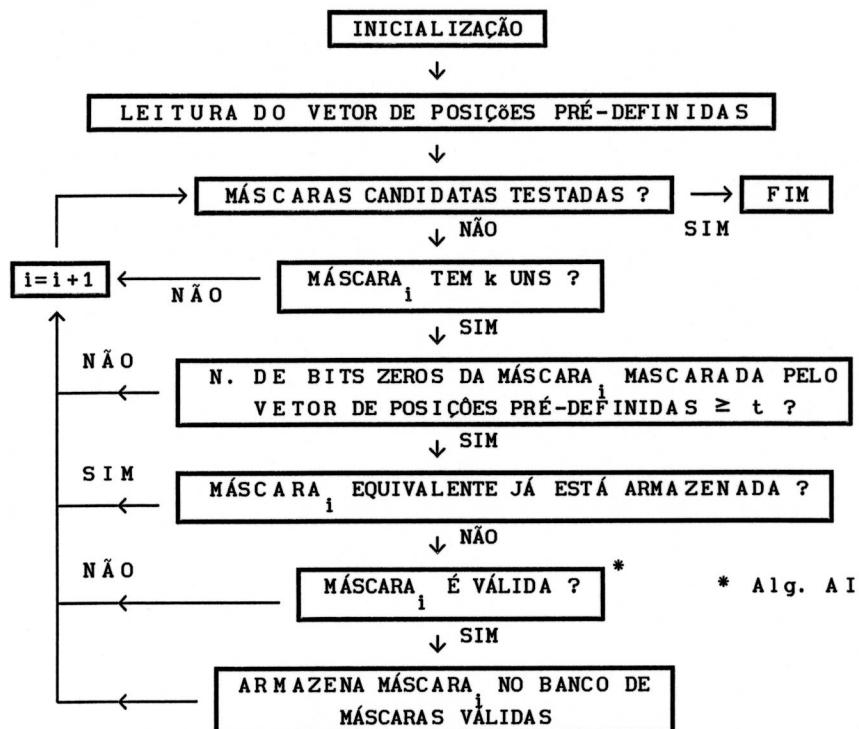


FIGURA 6.5 - ALGORITMO PARA FORMAÇÃO DE UM BANCO DE MÁSCARAS VÁLIDAS (AV)

Observa-se que, com a imposição destas restrições, o número de máscaras armazenadas N_a é dado pela equação :

$$N_a \leq \sum_{i=t}^z C_i^z \quad (6.1)$$

Onde z é o número de bits unitários do vetor de posições pré-definidas, e C_b^a representa o número de combinações de b elementos em um grupo de a elementos. A dedução desta fórmula está baseada em duas restrições

impostas pelo algoritmo :

- apenas uma máscara é armazenada para cada padrão de erro coberto,
- a máscara armazenada deve cobrir t ou mais erros nas posições definidas pelo vetor de posições pré-definidas.

6.2 - OS ALGORITMOS EM CONJUNTO

O uso conjunto de todos esses algoritmos fornece uma ferramenta para auxiliar a formação de conjuntos de cobertura a serem usados com o Algoritmo Conjuntos de Informação. A figura 6.6 mostra o interrelacionamento entre os diversos algoritmos.

O auxílio (computadorizado) à busca de um conjunto de cobertura está dividido em três fases. A primeira fase caracteriza-se pela busca de um pequeno conjunto de máscaras válidas, aproximadamente 2% do total de máscaras candidatas (valor experimental). Assim, forma-se um banco inicial de máscaras válidas que será usado na segunda fase da busca. Um algoritmo forma máscaras candidatas, para o código que se pretende usar, de maneira automática, que são verificadas pelo Algoritmo AI.

A segunda fase da busca consiste em selecionar algumas máscaras válidas e com elas formar um pré-conjunto de cobertura ao qual serão adicionadas máscaras buscando a redução dos padrões de t erros não cobertos. Esse pré-conjunto pode ser formado pela escolha de máscaras que facilitem um cálculo inicial do número de padrões de t erros cobertos (como será visto no exemplo do item 6.3).

Conforme o diagrama da figura 6.6, o usuário entra com o pré-conjunto e o Algoritmo AII determina os padrões de erro não cobertos (se houver). O objetivo é a cobertura completa dos padrões de t erros, mas pode ocorrer que o conjunto de máscaras do banco inicial de máscaras válidas não permita a

cobertura total e sim, o confinamento dos padrões de erro não cobertos a um certo número de posições. Se o confinamento não é claro, utilizam-se os Algoritmos AIII e AIV para selecionar novas máscaras do banco inicial de máscaras. O número de máscaras escolhidas pelos algoritmos pode ser grande e, assim, o usuário deve fazer a escolha final.

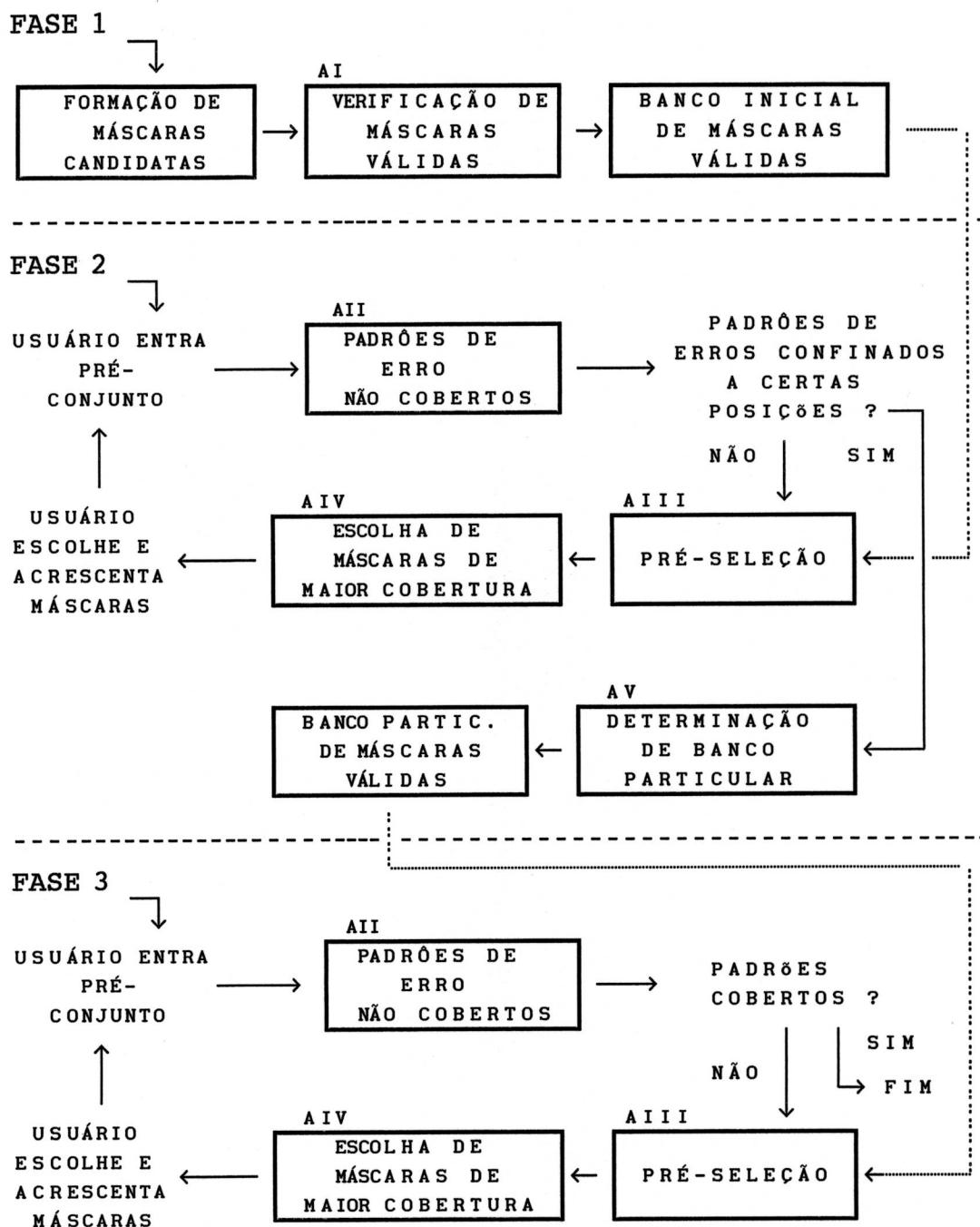


FIGURA 6.6 - INTERRELACIONAMENTO ENTRE ALGORITMOS

Quando os padrões de erro estiverem confinados a apenas certas posições da palavra, utiliza-se o Algoritmo AIV para formar um banco particular de máscaras válidas que cobre os padrões de t ou mais erros em posições pré-determinadas de uma palavra.

Finalmente, na terceira fase, o usuário entra com o pré-conjunto de cobertura formado até o momento e, a partir do banco particular de máscaras, os Algoritmos AIII e AIV indicarão as máscaras de maior cobertura. Desses máscaras, o usuário escolhe aquelas que permitem a completa cobertura dos padrões de t erros.

Neste ponto deve-se mencionar que alguns códigos não permitem a formação de um conjunto de cobertura capaz de cobrir todos os padrões de certo número de erros. Só é possível determinar esta condição, levando o processo de busca de um conjunto de cobertura até o final.

O objetivo principal no desenvolvimento deste método de busca relaciona-se com o tempo gasto para a formação de um conjunto de cobertura para determinado código. Um processo automático de busca, em que se visa à máscara de maior cobertura entre as C_k^n possíveis máscaras, para cada máscara acrescentada ao conjunto, demanda dezenas de horas de processamento. No método proposto, todos os algoritmos são rápidos, com exceção do algoritmo de formação de banco de máscaras válidas AV, que consome dezenas de minutos para a sua execução. Este algoritmo verifica, no pior caso, Nv máscaras para determinar as máscaras válidas. Este valor é dado pela equação :

$$Nv < \sum_{i=t}^z C_i^z \cdot C_{z-i}^{n-z} \quad (6.2)$$

Onde z é o número de bits unitários do vetor de posições pré-definidas, e C_b^a representa o número de combinações de b elementos em um grupo de a elementos. Esta

fórmula pode ser deduzida considerando-se que, no pior caso, a máscara equivalente ainda não foi armazenada e, então, o algoritmo de verificação de máscara válida (AI) é sempre executado (figura 6.5).

6.2.1 - EXEMPLO DE BUSCA DE CONJUNTO DE COBERTURA

Os algoritmos especificados foram implementados, em um caso particular, para a validação de seu funcionamento. A seguir, é apresentado um exemplo de busca de um conjunto de cobertura de padrões de dois erros para o código binário $C(2;16,2^8,3)$ que representa o código multinível C1 da tabela 4.1.

A primeira fase da busca consiste na formação de um Banco Inicial de máscaras válidas. Considerando-se que com aproximadamente 30 máscaras válidas já é possível cobrir todos os padrões de até três erros de bit (para este código), pode-se admitir que, um banco de máscaras (Banco Inicial) dez vezes maior (300 máscaras) pode fornecer a maior parte das máscaras formadoras do conjunto de cobertura. Esta quantia de 300 máscaras corresponde a aproximadamente 2% do número total de máscaras candidatas ($C_8^{16} = 12780$) para este código. Por facilidade formam-se máscaras candidatas combinando padrões simples, tais como 1111, 1100, 1001, etc, evitando o teste de todas as 12780 máscaras possíveis.

Na segunda fase, o usuário forma um pré-conjunto de cobertura com algumas das máscaras válidas do banco inicial, procurando cobrir o maior número possível de padrões de dois erros. Esta escolha pode ser a seguinte :

```
Máscara 1 : 1111111100000000
Máscara 2 : 0000000011111111
Máscara 3 : 000011111110000
Máscara 4 : 1111000000001111
Máscara 5 : 0000111100001111
```

Observa-se que as duas primeiras máscaras cobrem todos os padrões de um erro e que máscaras complementares são escolhidas. O Algoritmo AII fornece o seguinte resultado :

Erros a serem cobertos (t) : 2

Número de padrões de erro totais : 120

Padrões cobertos : 104

Padrões não cobertos :

```

0000100000001000
0000010000001000
0000001000001000
0000000100001000
0000100000000100
0000010000000100
0000001000000100
0000000100000100
0000100000000010
0000010000000010
0000001000000010
00000001000000010
0000100000000001
0000010000000001
0000001000000001
0000000100000001

```

Uma análise dos padrões de erro não cobertos pelo pré-conjunto permite afirmar que, se a máscara 111100001111 0000 fosse válida, com apenas seis máscaras todos os padrões de dois erros seriam cobertos.

Neste ponto, já que os padrões de erro estão confinados a apenas duas regiões (observe o conjunto de padrões não cobertos), pode-se passar à determinação do banco particular de máscaras válidas (Algoritmo AV), para cobrir os erros nestas regiões. O vetor de posições pré-definidas associado a este passo da busca é o seguinte :

0000111100001111.

A terceira fase consiste em determinar as máscaras faltantes para a cobertura de todos os padrões de dois erros. O usuário entra com o pré-conjunto e o Algoritmo AII determina os padrões não cobertos. A partir desses e das informações do banco particular de máscaras válidas, o Algoritmo AIII realiza uma seleção inicial de máscaras. Com base nessas máscaras selecionadas, o Algoritmo AIV escolhe as de maior cobertura e informa a distância de cada máscara em relação às do pré-conjunto de cobertura. As de maior cobertura são :

Máscara/Cobertura/Distância

1111010100000110	4	4
1111011000000101	4	4
1111010100000101	4	4
1111011000000011	4	4
1111100100000110	4	4
1111101000000101	4	4
1111100100000101	4	4
1111101000000011	4	4
1111110000000011	4	4
1111011000001010	4	4
1111010100001010	4	4
1111001100001010	4	4
1111011000001001	4	4
1111010100001001	4	4
1111001100001001	4	4
1111101000001010	4	4
1111011000001100	4	4
1111010100001100	4	4
1111001100001100	4	4
1111101000001100	4	4
1111100100001100	4	4

O usuário escolhe, por exemplo, duas dessas máscaras e as acrescenta ao pré-conjunto, obtendo :

Máscara 1 : 1111111100000000
Máscara 2 : 0000000011111111
Máscara 3 : 000011111110000
Máscara 4 : 1111000000001111
Máscara 5 : 0000111100001111
Máscara 6 : 1111010100000101
Máscara 7 : 1111101000001010

O Algoritmo AII determina os padrões de dois erros não cobertos :

Erros a serem cobertos (t) : 2
Número de padrões de erro totais : 120
Padrões cobertos : 112

Padrões não cobertos :

0000010000001000
0000000100001000
0000100000000100
00000010000000100
00000100000000010
00000001000000010
0000100000000001
00000010000000001

Novas máscaras são escolhidas.

Máscara/Cobertura/Distância :

1111101000000101	4	4
1111010100001010	4	4

E acrescentadas ao pré-conjunto formado.

Máscara 1 : 111111100000000
 Máscara 2 : 000000001111111
 Máscara 3 : 000011111110000
 Máscara 4 : 1111000000001111
 Máscara 5 : 0000111100001111
 Máscara 6 : 1111010100000101
 Máscara 7 : 1111101000001010
 Máscara 8 : 1111101000000101
 Máscara 9 : 1111010100001010

Erros a serem cobertos (t) : 2
 Número de padrões de erro totais : 120
 Padrões cobertos : 120

O processo de busca termina, pois todos os padrões de dois erros foram cobertos.

6.3 - AUTOMATIZAÇÃO DO PROCESSO DE BUSCA

A escolha de máscaras que são acrescentadas ao conjunto de cobertura em formação, conforme a figura 6.6, é de competência do usuário. Para uma automatização completa, basta que o sistema escolha, entre as máscaras de maior cobertura, uma de maior distância em relação às máscaras do conjunto em formação. A figura 6.7 mostra como os algoritmos podem ser agrupados para formar um sistema automático de formação de conjunto de cobertura.

Os dados de entrada são : o pré-conjunto de cobertura fornecido pelo usuário, máscaras de um banco de máscaras válidas e o número de erros a serem cobertos. Como saída, têm-se as máscaras escolhidas e os padrões de t erros não cobertos.

Após a entrada do pré-conjunto, o Algoritmo AII determina os padrões de erros não cobertos. Caso todos os padrões de t erros tenham sido cobertos, o processo termina;

esta é uma das condições de parada. Caso contrário, o Algoritmo AIII realiza uma pré-seleção das máscaras do banco de máscaras e o Algoritmo AIV escolhe uma máscara que é acrescentada ao conjunto de cobertura em formação. Caso nenhuma máscara seja escolhida, por não cobrir qualquer padrão de erros não coberto, o processo termina. Esta é a segunda condição de parada. Um novo banco de máscaras deve ser usado para a determinação das faltantes, neste caso.

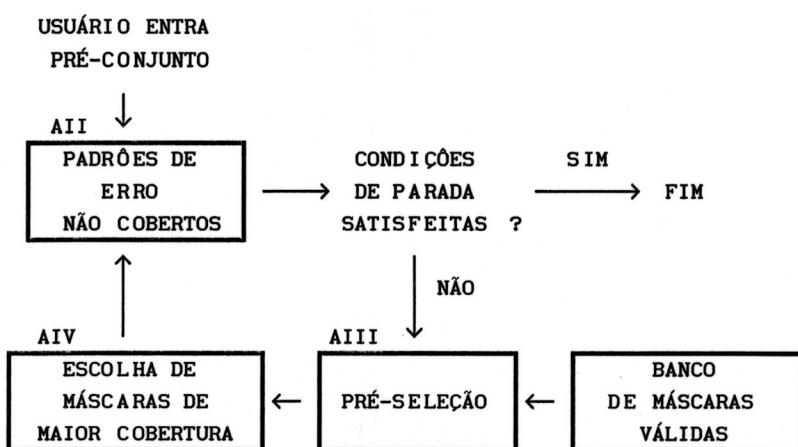


FIGURA 6.7 - AUTOMATIZAÇÃO DO PROCESSO DE BUSCA

Uma característica interessante do processo de busca proposto é o fato de que, a medida que os padrões de erro vêm sendo cobertos, a execução torna-se mais rápida; pois a escolha de máscaras (algoritmo AIII) é fundamentada no número de padrões de erro não cobertos que diminui com o acréscimo de novas máscaras ao conjunto de cobertura em formação. E finalmente, o processo de busca é subótimo, pois a seleção da melhor máscara em um universo reduzido não corresponde a mesma escolha entre todas as possíveis máscaras.

6.4 - BUSCA DE CÓDIGOS MULTINÍVEIS MELHORES

Os códigos multiníveis propostos, até este momento, foram selecionados, levando-se em consideração apenas a sua distância Euclidiana, já que esta é o parâmetro responsável

pelo ganho de codificação no espaço de sinais, considerando um esforço ilimitado em decodificação. Como os códigos propostos são decodificados binariamente, é necessário que selecione-se o código binário, que representa o código multinível, que tenha a maior distância de Hamming possível, para um desempenho máximo em decodificação.

A seguir será desenvolvido um método de busca de códigos, em que se deseja maximizar a distância Euclidiana selecionando, também, os códigos binários de maior distância de Hamming. O método desenvolvido, também, preocupa-se com o tempo de busca envolvido no processo.

6.4.1 - A ESTRUTURA DOS CÓDIGOS BUSCADOS

Como mostrado no item 4.6, pode-se representar o código multinível por apenas uma matriz geradora binária denotada matriz T (equação 4.11). Como visto, os códigos multiníveis encontrados, embora com distância Euclidiana maximizada, apresentam distância de Hamming (na representação binária) reduzida devido a uma região de zeros no conjunto de paridade da matriz geradora T . Como o processo de decodificação dos códigos multiníveis propostos se baseia na sua representação binária, esta reduzida distância representa uma deficiência.

Para reduzir essa limitação, define-se uma nova matriz geradora binária T_4 , composta de Submatrizes Cíclicas J , B_1 e E_0 . Para a modulação 4-PSK, por exemplo, a matriz geradora T_4 é a seguinte :

$$T_4 = \left[\begin{array}{c|c|c} I & \begin{matrix} J & B_0 \\ \hline \dots & \dots \end{matrix} \\ \hline E_0 & J \end{array} \right] \quad (6.3)$$

Todo o raciocínio será desenvolvido para a modulação PSK, mas é igualmente válido para a modulação QAM.

O objetivo, agora, é encontrar as submatrizes J , B_0 e E_0 , de modo a satisfazer as necessidades em distâncias de Hamming e Euclidiana do código. Observa-se que, utilizando essa estrutura, a codificação segundo a figura 4.1, e a decodificação, segundo o sistema apresentado na figura 4.5, deixam de ser válidas, já que T_4 não mais representa o sistema da figura 4.1, pois a submatriz E_0 é diferente de zero.

Para a extensão à uma modulação $2q$ -PSK, propõe-se a utilização de códigos já determinados para a modulação q -PSK acrescidos de submatrizes cíclicas B_1 à direita de B_0 , E_1 abaixo de E_0 , e J na diagonal principal da matriz geradora T_{2q} formada.

Como exemplo, para a modulação 8-PSK, forma-se a matriz T_8 a partir da matriz T_4 .

$$T_8 = \begin{bmatrix} I & \left[\begin{array}{|c|c|c|} \hline J & B_0 & B_1 \\ \hline E_0 & J & B_1 \\ \hline E_1 & E_1 & J \\ \hline \end{array} \right] \end{bmatrix} \quad (6.4)$$

Neste caso, já conhecidas J , B_0 e E_0 , buscam-se apenas as submatrizes B_1 e E_1 . Assim, o tempo tomado pelas variações de B_i e E_i depende das dimensões da matriz T_q e independe do alfabeto de modulação usado.

Todas as submatrizes são cíclicas, para facilidade de busca, e apresentam a estrutura, por exemplo, de B_0 :

$$B_0 = \begin{bmatrix} b_{01} & b_{02} & \dots & b_{0i} \\ b_{0i} & b_{01} & \dots & b_{0,i-1} \\ \vdots & & & \vdots \\ b_{02} & b_{03} & \dots & b_{01} \end{bmatrix}$$

Nota-se que somente é necessário conhecer a primeira linha da submatriz para sua definição completa.

6.4.2 - ALGORITMO DE BUSCA

A busca de códigos consiste em testar todas as possíveis combinações de submatrizes B_i e E_i para a modulação $2^{(i+2)}\text{-PSK}$, $i \geq 0$, dada a submatriz J (e as submatrizes B_j e E_j , $0 \leq j < i$, se for o caso). Para cada combinação determinam-se as palavras-código e calculam-se as distâncias de Hamming e Euclidiana, armazenando as submatrizes que formam um código que satisfaz o critério de distâncias maximizadas. O fluxograma de busca é apresentado na figura 6.8.

Inicialmente, faz-se a leitura da matriz inicial J ou de J , B_i e E_i , quando $i > 0$. Deve-se ler também os valores iniciais de B_i e E_i , os valores de distância que servirão como valores de comparação com os valores calculados para o código em teste, e o número de palavras-código geradas (N_{pg}), esclarecido adiante. O algoritmo varre todas as possíveis combinações de B_i com E_i atualizando a matriz geradora T_q . A seguir, geram-se N_{pg} palavras-código, calculam-se as distâncias de Hamming e Euclidiana do código em teste, armazenando os valores obtidos quando as condições forem satisfeitas.

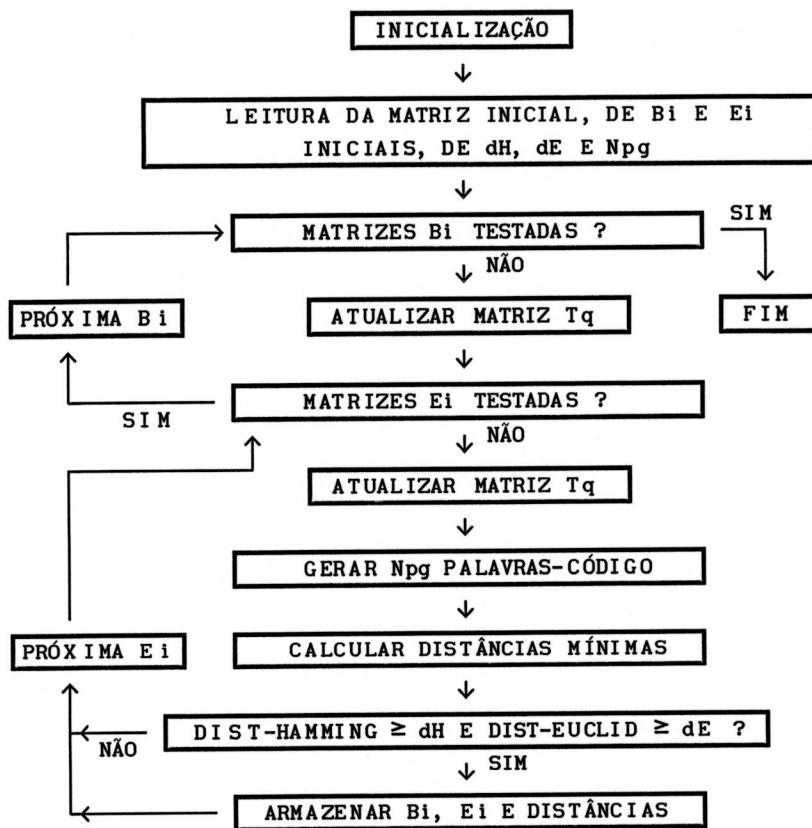


FIGURA 6.8 - FLUXOGRAMA DE BUSCA DE CÓDIGOS

As rotinas que mais consomem tempo de execução são : a rotina de geração de palavras-código e a de cálculo de distâncias. Devido à linearidade dos códigos, calcular as distâncias mínimas consiste em determinar o peso da palavra-código de menor peso. Mesmo assim, para códigos com grande número de palavras-código, o processo de busca demanda horas de execução.

Para um código multinível $C(q; n, q^k, dE)$, o tempo de busca t_b é proporcional a $2^{2(n-k)} \cdot q^k$. O termo em 2 é devido às combinações de B_i com E_i , e o termo q^k é o número de palavras-código que é dependente da modulação usada. Para códigos com grande número de palavras-código pode-se utilizar o seguinte método de busca, de modo a diminuir o tempo gasto nesta tarefa. Estipula-se o número de palavras-código geradas (N_{pg}) com valor cinco vezes menor que o número de palavras-código (q^k), e realiza-se a busca. Este valor de N_{pg}

é experimental. O resultado desta busca fornece uma pré-seleção de valores de B_i , E_i e de distâncias (não corretas, pois não foram testadas todas as palavras-código). A partir desses dados pode-se eliminar os valores de B_i e E_i que fornecem distâncias baixas e utilizar os demais, como parâmetros de entrada, em uma nova busca com $N_{pg} = q^k$. Dessa forma pode-se reduzir o tempo de busca de novos códigos.

6.4.3 - CÓDIGOS OBTIDOS COM O PROCESSO DE BUSCA PROPOSTO

Utilizando-se o método proposto, foram encontrados alguns códigos multiníveis para a modulação codificada. Esses códigos estão representados pelas submatrizes cíclicas binárias, formadoras de T_q , nas Tabelas 6.I e 6.II, onde as primeiras linhas de J , B_i e E_i são dados por sua representação decimal. A tabela 6.I apresenta códigos multiníveis, de taxa $R = 1/2$, para a modulação 4-PSK, e a Tabela 6.II para a modulação 8-PSK.

Por exemplo, na Tabela 6.I, o código $C(4; 8, 4^4, 10)$ com $J = 3$, $B_0 = 4$ e $E_0 = 13$, denotado C4, fornece a seguinte matriz geradora T_4 :

$$T_4 = \begin{bmatrix} 10000000011000010 \\ 0100000001100001 \\ 0010000000111000 \\ 0001000010010100 \\ 0000100010111100 \\ 0000010011010110 \\ 0000001011100011 \\ 0000000101111001 \end{bmatrix} \quad (6.3)$$

TABELA 6.I - CÓDIGOS MULTINÍVEIS PARA 4-PSK ($R = 1/2$)

n	J	B_0	E_0	d	dE_{\min}^2
4	1	2	0	2	8
		2	3	3	
		3	0	2	
4	2	1	0	2	8
		1	3	3	
		3	0	2	
6	3	4	1		8
		4	2		
		4	7	4	
		7	1		
		7	2		
6	5	7	4		8
		2	1		
		2	4		
		2	7	4	
		7	1		
8	3	7	2		10
		7	4		
		8	9		
		8	11		
		8	14		
		11	12		
		13	1		
		13	4		
		14	2		
		14	8		
10	3	9	6		12
		9	15		
		9	17		
		10	6		
		12	5		
		13	10		
		17	5		
		18	3		
		20	3		
		20	12		
		20	23		
		21	18		

TABELA 6.II - CÓDIGOS MULTINÍVEIS PARA 8-PSK ($R = 1/2$)

n	J	B_0	E_0	B_1	E_1	d	dE_{\min}^2
4	1	2	3	0 3	1	3	3,17
	2	1	3	0 3	2	3	3,17
	2	1	0	1	3	3	4,58
6	3	4	1	7	3	3	5,17
	3	4	2	7	3	3	5,17
	3	7	4	5 6	6 5	3	5,17
8	3	4	13	8	0 1	3 5	5,17 4,68
	3	8	14	4	0 2	3 5	5,17 4,68

Observa-se que os códigos da Tabela 6.I, para 4-PSK, apresentam distâncias Euclidianas iguais às dos códigos encontrados por Baldini. Já os códigos da Tabela 6.II, para 8-PSK, apresentam distâncias Euclidianas maiores que as dos códigos pseudo-cíclicos, mas menores que as dos códigos BLCS de Baldini. Este fato se deve à estrutura imposta aos códigos propostos que facilita o processo de busca, mas, por outro lado, limita o valor das distâncias encontradas. Além disso, há uma relação inversa entre as distâncias de Hamming e Euclidianas, como se observa, por exemplo, para os códigos com $n = 8$ da Tabela 6.II, ou seja, o aumento do valor de uma implica a redução do valor da outra.

6.5 - BUSCA DE CONJUNTO DE COBERTURA E DESEMPENHO PARA O CÓDIGO C4

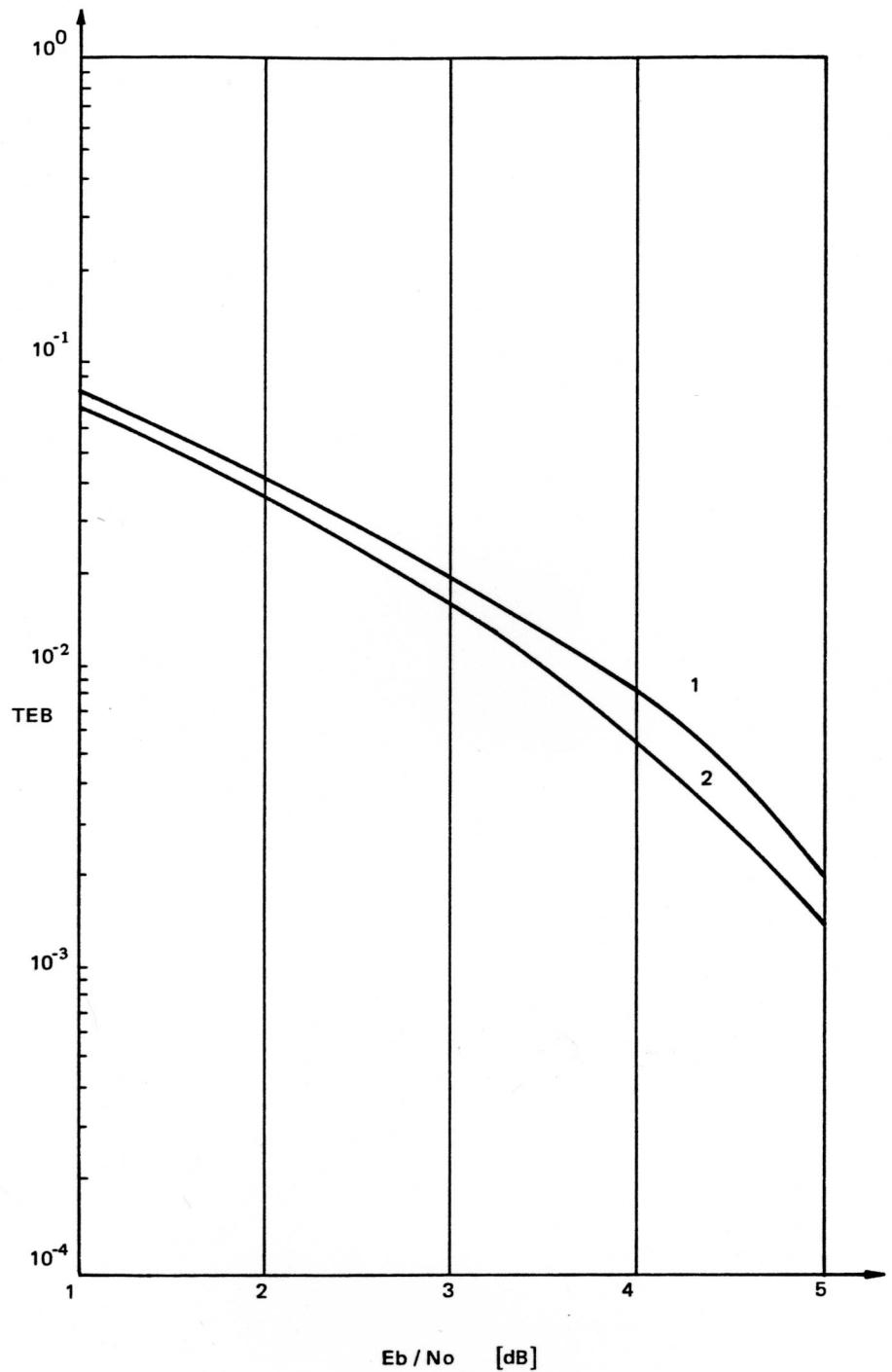
Para finalizar este capítulo, apresentam-se os desempenhos do sistema 4-PSK (capítulo 4) e do sistema 16-QAM (capítulo 5) que utilizam o código ótimo C4.

O conjunto de cobertura para esse código foi determinado com o auxílio do processo mostrado na figura 6.7. O número de máscaras para cobrir todos os padrões de 3 erros foi reduzido a 25, o que representa um ganho em relação ao conjunto de cobertura, de 32 máscaras, obtido manualmente no capítulo 4.

A figura 6.9 mostra o desempenho do código C1 (capítulo 4), decodificado com 32 matrizes conjunto de informação, comparado ao desempenho do código C4 decodificado com 25 matrizes conjunto de informação. O código C4 permite um ganho de aproximadamente 0,25 dB sobre o código C1, devido à maior distância de Hamming e à cobertura de todos os padrões de três erros de bit do código binário que o representa.

A figura 6.10 mostra o desempenho do sistema 16-QAM com o uso do código C4 à taxa 1/2 (curva 3) comparado ao desempenho do sistema com o código C1 (curva 2). No caso, há um ganho de aproximadamente 0,75 dB a taxa de erro de bit de 10^{-4} . O desempenho do sistema à taxa 3/4 com o código C4 não mostrou melhora em relação ao desempenho com uso do código C1.

Os objetivos, com o desenvolvimento dos métodos deste capítulo, foram alcançados, ou seja, melhorar o desempenho e a velocidade em decodificação dos sistemas propostos, mesmo que pouco.



1 - CÓDIGO C1 - ACI - 32 CONJ. DE INFORM.

2 - CÓDIGO C4 - ACI - 25 CONJ. DE INFORM.

FIGURA 6.9 - DESEMPENHO DO SISTEMA 4-PSK

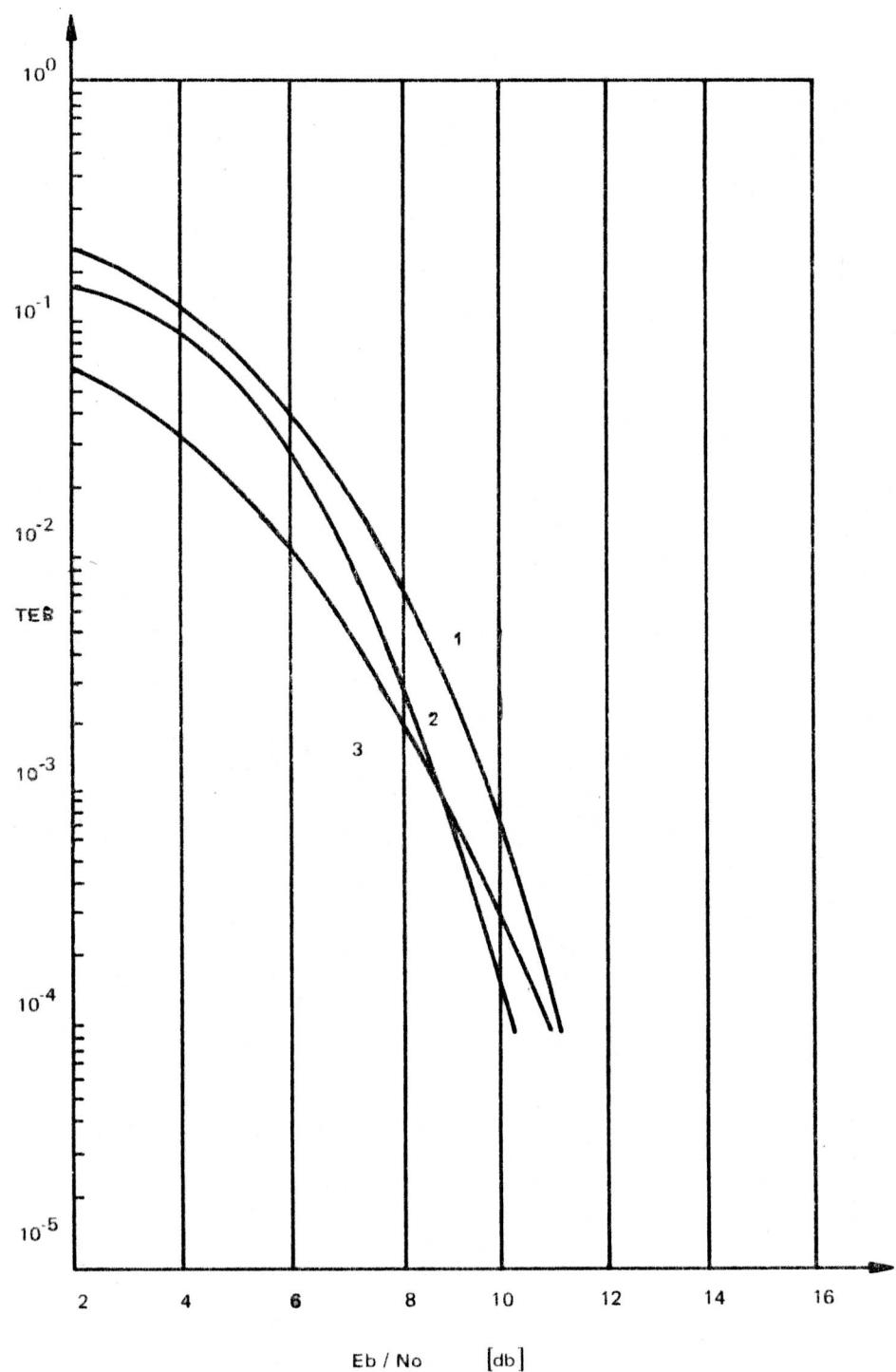


FIGURA 6.10 - DESEMPENHO DO SISTEMA 16-QAM

CAPÍTULO 7

CONSIDERAÇÕES FINAIS

O problema na comunicação confiável de dados é o uso eficiente tanto de faixa de freqüências como de energia a altas taxas de transmissão. A solução para este problema, atualmente, é o emprego de sistemas combinados de modulação e codificação. Esses sistemas asseguram uso eficiente de energia com conservação de faixa e apresentam desempenho superior quando comparados ao desempenho de sistemas em que a codificação e a modulação são feitas separadamente [UNG 82].

Nos sistemas de Modulação Codificada, sob o ponto de vista de Ungerboeck [UNG 82] e Sayegh [SAY 86], os bits de informação a serem transmitidos são codificados binariamente e grupos desses determinam quais sinais do alfabeto de modulação devem ser transmitidos. Sob o ponto de vista de Baldini [BAL 89], sabe-se, inicialmente, quais sinais de modulação devem ser transmitidos e a esses acrescenta-se a redundância, que são novos sinais, sempre levando-se em consideração que as seqüências de sinais transmitidos devem estar afastadas o máximo possível umas das outras, para garantir o ganho em decodificação.

Uma contribuição ao estudo da Modulação Codificada com Códigos de Bloco foi apresentada neste trabalho. As questões colocadas no capítulo 1, serão respondidas descrevendo-se o conteúdo de cada capítulo. O capítulo 2 apresentou conceitos básicos sobre transmissão de dados, introduzindo os processos de modulação, o conceito de contelação de sinais, o problema do ruído e o uso de sistemas de codificação e decodificação em modem's de alta velocidade.

O capítulo 3 apresentou conceitos básicos sobre códigos controladores de erros, introduziu o importante conceito de Modulação Codificada, e comparou os principais sistemas BCM. Dessa comparação, levantou-se que o sistema de mais simples implementação e decodificação é o de Sayegh.

No capítulo 4, desenvolveu-se um sistema para obtenção e decodificação de Códigos de Bloco Multiníveis para a Modulação Codificada. A primeira forma de obtenção é baseada no uso de duas matrizes : a matriz geradora G e a matriz de transformação M_T . A relação entre bits e símbolos é fornecida a partir de polinômios sobre o campo de Galois. Os códigos multiníveis, assim formados, são lineares e invariantes em distância. O primeiro processo de decodificação proposto, que utiliza o algoritmo de Chase, consiste no cálculo de confiabilidade dos símbolos recebidos, na formação de palavras candidatas à decodificação e na decodificação binária. Esse processo não se mostrou satisfatório, pois, além de ser lento devido ao cálculo de confiabilidade, seu desempenho em relação ao melhor desempenho (MLD) é baixo (figura 4.8). A segunda forma de obtenção de códigos é baseada na representação do código multinível por um código binário. De uma forma simples, um código de bloco multinível $C(q;n,q^k,d_E)$ pode ser representado por um único código de bloco binário (matriz T) $C(2;N,2^k,d)$, em que $N = q \cdot n$ e $K = q \cdot k$, através do mapeamento adequado de bits em símbolos. Dessa forma, os processos de codificação e decodificação tornam-se mais simples em relação ao sistema básico. O desempenho de decodificação do código multinível com algoritmo de Chase, quando realizado utilizando a matriz T , é melhor que quando se usam as matrizes G e M_T (figura 4.12). O segundo processo de decodificação, proposto com o objetivo de reduzir o número de perturbações na seqüência recebida, utiliza o Algoritmo Conjuntos de Informação. Nesse caso não mais se calcula a confiabilidade dos símbolos recebidos, diminuindo, assim, a complexidade de decodificação. O desempenho de decodificação com este processo mostrou ser melhor e mais rápido que o sistema com algoritmo de Chase

(figura 4.14). Uma comparação com os códigos de Sayegh e Baldini revela que os códigos multiníveis propostos podem obter os mesmos ganhos e são mais facilmente decodificados se comparados aos de Baldini.

No capítulo 5, foi proposto um sistema para a modulação 16-QAM, que se baseia na partição do espaço de sinais 16-QAM proposta por Baldini, mas que utiliza os códigos multiníveis propostos no capítulo 4. Duas versões para o sistema foram especificadas, ambas utilizando o Algoritmo Conjuntos de Informação como algoritmo de decodificação. Foram especificados os algoritmos de codificação e decodificação com os quais o sistema foi validado através de simulação em computador. O sistema de decodificação proposto é de complexidade inferior ao algoritmo de correlação completa (MLD). A versão V1 do sistema é preferível em relação a versão V2, pois aquela possibilita um ganho em codificação maior e uma menor complexidade de decodificação. Como resultado, especificou-se um circuito integrado, com finalidade didática, para a codificação e decodificação com o qual se espera obter uma velocidade de transmissão de 9600 bit/s à taxa $R = 3/4$ com a tecnologia disponível. Os resultados dos testes a serem realizados com o CI, permitirão verificar a sua velocidade de decodificação e se seu desempenho real corresponde ao encontrado via simulação.

No capítulo 6, desenvolveu-se algoritmos para auxílio à formação de conjuntos de cobertura utilizados no Algoritmo Conjuntos de Informação, e um processo de busca de melhores códigos multiníveis. Os primeiros são algoritmos importantes do ponto de vista da velocidade de decodificação, pois com o uso de conjuntos de cobertura com um número mínimo de máscaras, mais rápida será a decodificação. Códigos melhores são importantes do ponto de vista do desempenho do sistema. Um processo de busca de conjuntos de cobertura, em que o usuário interage com grande freqüência durante a execução, foi apresentado. Esta característica torna o processo lento.

Uma versão automática foi proposta (figura 6.7) com base na versão anterior (figura 6.6), em que tudo o que é necessário, são bancos de máscaras válidas para o código em uso. A validação desses algoritmos foi realizada através de sua implementação para um caso particular, ou seja, para busca de máscaras para códigos binários de comprimento igual a oito e taxa de codificação $R = 1/2$. Este processo de busca é subótimo pois a seleção da melhor máscara em um universo reduzido não corresponde a mesma escolha entre todas as máscaras possíveis. O processo de busca de códigos preocupa-se tanto com a distância Euclidiana do código multinível quanto com a distância de Hamming em sua representação binária, já que os códigos multiníveis são decodificados binariamente. Uma nova matriz geradora T_q foi proposta eliminando a deficiência (região de zeros no conjunto de paridade) da matriz T do capítulo 4. A busca de códigos, para qualquer modulação, consiste na alteração de apenas duas submatrizes cíclicas e no armazenamento dos parâmetros que definem o código quando as distâncias calculadas ultrapassam valores pré-estabelecidos. Para códigos com grande número de palavras-código, de modo a diminuir o tempo de execução, foi proposta a busca com um número de palavras inferior ao número de palavras-código do código. Após, nova busca seria realizada num universo já limitado de possibilidades. A simulação de um dos códigos encontrados permitiu verificar que, como esperado, seu desempenho é melhor se comparado ao desempenho dos códigos do capítulo 4.

Como se pode observar, dado o problema que era encontrar Códigos Multiníveis para a Modulação Codificada, procuraram-se soluções para todos os outros problemas daí decorrentes.

Resumindo, as contribuições deste trabalho, são :

- a) Análise dos principais sistemas de Modulação Codificada com Códigos de Bloco (BCM);

- b) Proposta de um sistema de obtenção de códigos de bloco multiníveis;
- c) Proposta de algoritmos de decodificação para os códigos propostos;
- d) Levantamento do desempenho em decodificação dos algoritmos propostos por simulação em computador (IBM-PC);
- e) Desenvolvimento de um método para busca de códigos ótimos;
- f) Desenvolvimento de algoritmos para auxílio à busca de conjuntos de cobertura para o Algoritmo Conjuntos de Informação; e
- g) Especificação dos algoritmos de codificação e decodificação para o desenvolvimento de um circuito integrado com finalidade didática em modulação 16-QAM.

Para finalizar, apresentam-se alguns pontos que podem ser desenvolvidos futuramente :

- a) A determinação de um limitante superior em distância para os códigos multiníveis propostos;
- b) A exploração da linearidade dos códigos na determinação de um novo algoritmo de decodificação que procure evitar o constante acesso ao banco de dados (esse acesso representa o "gargalo" do sistema);
- c) A utilização de Regras de Parada nos algoritmos propostos, como por exemplo, o limiar do GMD [BLA 83].
- d) A implementação de um sistema completamente genérico para a busca de conjuntos de cobertura (neste trabalho o sistema foi implementado para um caso particular);
- e) A realização dos testes no CI para a verificação de sua velocidade e desempenho em decodificação.

É desejo do autor que este trabalho venha a incentivar e auxiliar novas pesquisas na área de comunicação de dados.

REFERÉNCIAS BIBLIOGRÁFICAS

- [BAL 87] Baldini Filho, Renato; Pessoa, A. C. F. & Arantes, D. S.; "Systematic Linear Codes over a Ring for Encoded Phase Modulation", ISICT'87, Campinas-SP, 1987.
- [BAL 89] Baldini Filho, Renato. "Códigos Pseudo-Cíclicos Multiníveis", Anais do 7º Simpósio Brasileiro de Telecomunicações, Florianópolis - SC, p. 402-405, setembro, 1989.
- [BEN 88] Benedetto, B., Marsan, M. A., Albertengo, G. & Giachin, E., "Combined Coding and Modulation : Theory and Applications", IEEE Transactions on Information Theory, p. 290-293, março, 1987.
- [BER 78] Berlekamp, E.R., McEliece, R.J., Tilborg, H.C.A. van, "On the Inherent Intractability of Certain Coding Problems", IEEE Trans. on Inf. Theory, maio, 1978.
- [BEZ 87] Bezerra, João Batista. "Códigos Convolucionais para Modulação Codificada", Dissertação de Mestrado, Campinas, 1987.
- [BLA 83] Blahut, R. E., "Theory and Practice of Error Control Codes", Addison-Wesley, 1983.
- [CCITT] CCITT, Red Book Vol. VIII.1, "Data Communication over the Telephone Network - Recommendations of the V Series".

- [CHA 72] Chase, D., "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information", IEEE Trans. on Inf. Theory, janeiro, 1972.
- [C&C 81] Clark, G. C. & Cain, J. B., "Error-Correction Coding for Digital Communications", New York, Plenum Press, 1981.
- [C&G 62] Campopiano, C. M. & Glazer, B. G. "A Coherent Digital Amplitude and Phase Modulation Scheme", IRE Trans. on Comm. Syst., p. 90-95, março 1962.
- [C&G 90] Coffey, John T. & Goodman, R.M. "The Complexity of Information Set Decoding", IEEE Trans. on Inf. Theory, p. 1031-1037, setembro, 1990.
- [C&S 90] Chouly, Antoine & Sari, Hikmet. "Design and Performance of Block-Coded Modulation for Digital Microwave Radio Systems", IEEE Trans. on Commun., p. 727-732, maio, 1990.
- [CUS 84] Cusack, B. L., "Error Control for QAM Signalling", Electronics Letters, p. 62-63, janeiro, 1984.
- [DIV 87] Divsalar, D.; Simon, M.K.; Yuen, J.H. "Trellis Coding with Asymmetric Modulations", IEEE Trans. on Commun., p. 130-141, fevereiro, 1987.
- [FIS 78] Fishman, George S. "Principles of Discrete Event Simulation", John Wiley & Sons, 1978.
- [FOR 84] Forney, G. D. Jr, Gallager, R. G., Long G. R., Longstaff F. M. e Qureshi S. V., "Efficient Modulation for Band-Limited Channels", IEEE Journal on Selected Areas in Communications, p. 632-647, setembro, 1985.

- [FRE 81] Freeman, L. Roger. "Telecommunication Transmission Handbook", 2nd Ed. John Wiley & Sons, 1981.
- [GIN 84] Ginzburg, V. V., "Multidimensional Signals for a Continuos Channel", Problemy Peradachi Informatsii, p. 28-46, Vol. 20, 1984.
- [G&A 88] Godoy Júnior, Walter & Arantes, Dalton Soares. "Suboptimum Soft-Decision Decoding of Block Codes Using the Zero-Neighbors Algorithm", IEEE Int. Symposium of Inf. Theory, Kobe, Japão, 1988.
- [G&A 89] Godoy Júnior, Walter & Arantes, Dalton Soares. "Decodificação de Códigos de Bloco com Decisão Suave e Correlação Incompleta", Anais do 7º Simpósio Brasileiro de Telecomunicações, Florianópolis - SC, p. 193-197, setembro, 1989.
- [G&A 90] Godoy Júnior, Walter & Arantes, Dalton Soares. "Um Algoritmo para Decodificação de uma Classe de Códigos Multiníveis com Modulação M-PSK", International Conference on Communications Systems - Singapore ICCS'90, Singapura, 1990.
- [G&P 89] Godoy Júnior, Walter & Palazzo Jr., Reginaldo. "Construções de Códigos Concatenados Generalizados via Reticulados e Decodificação Suave Parcial", Anais do 7º Simpósio Brasileiro de Telecomunicações, Florianópolis - SC, p. 180-186, setembro, 1989.
- [I&H 77] Imai, Hideki & Hirakawa, Shuji, "A New Multilevel Coding Method Using Error-Correcting Codes", IEEE Transactions on Information Theory, p. 371-377, maio, 1977.

- [KAS 90] Kasami, T.; Takata, T.; Fujiwara, T.; Lin, Shu. "A Concatenated Coded Modulation Scheme for Error Control", IEEE Trans. on Commun., p. 752-762, junho, 1990.
- [KNU 73] Knuth, D. "The Art of Computer Programming", Adison Wesley Reading, Vol.1, 1973.
- [KRO 87] Kronsjö, Lydia. "Algorithms : Their Complexity and Efficiency", John Wiley & Sons, 2º Ed., 1987.
- [L&C 83] Lin, Shu & Costello Jr., Daniel J. "Error Control Coding : Fundamentals and Applications", Prentice-Hall, 1983.
- [L&H 85] Levitin, Lev B. & Hartmann, C.R.P. "A New Approach to the General Minimmun Distance Decoding Problem : The Zero-Neighbors Algorithm", IEEE Trans. on Inf. Theory, p. 378-384, maio, 1985.
- [MAR 90] Marcon, César A. M., Dissertação de Mestrado (em preparação), UFRS, Porto Alegre.
- [MAS 74] Massey, James L.; "Coding and Modulation in Digital Communications", Proc. 1974 Int. Zurich Seminar on Digital Communications, Zurich, Switzerland, março 1974.
- [MES 89] Messer, Dion D.; "Convolutional Encoding and Viterbi Decoding using the DSP56001 with a V.32 Modem Trellis Example", Motorola INC. 1989.
- [M&L 85] Michelson, A. M. & Levesque, A. H.; "Error-Control Tecnicas for Digital Communication", John Wiley & Sons, Canadá, 1985.

- [M&S 77] MacWilliams, F. J. & Sloane, N. A.; "The Theory of Error-Correcting Codes", North-Holland Mathematical Library, 1977.
- [POR 85] Portnoi, S. L., "Characteristics of Coding and Modulation Systems from the Standpoint of Concatenated Codes", Problemy Peradachi Informatsii, Vol. 21, p. 14-27, julho-setembro, 1985.
- [PRO 83] Proakis, J. G., "Digital Communications", McGraw-Hill, 1983.
- [R&W 85] Ross, Kenneth A. & Wright, Charles R. B., "Discrete Mathematics", Prentice-Hall Inc., 1985.
- [SAY 86] Sayegh, Soheil L., "A Class of Optimum Block Codes in Signal Space", IEEE Trans. on Communications, p. 1043-1045, outubro, 1986.
- [SCH 64] Schönheim, J., "On Coverings", Pacific Journal of Math., p. 1405-1411, 1964.
- [SHA 48] Shannon, C.E.; "A Mathematical Theory of Communication", Bell Syst. Tech. J., julho, 1948.
- [SHA 69] Shannon, C. E. & Weaver, W.; "The Mathematical Theory of Communication", The University of Illinois Press, 1969.
- [SIL 90] Silva Filho, Jorge F. da, "Processamento Digital de Sinais em Modem's de Alta Velocidade", Dissertação de Mestrado, UFRS, Porto Alegre, maio, 1990.
- [TAK 87] Tanaka, Y.; Unagami, S.; Kinoshida, T.; Kaku, T. e Arai, Y.; "A 19,2 Kb/s High-Speed Voiceband Data Modem based on Eight-Dimensional Trellis Coding", Global Telecommunications Conference - 2º Vol.,

novembro, Tokio, Japão, 1987.

- [TAN 87] Tanner, R. Michael, "Algebraic Construction of Large Euclidian Distance Combined Coding/Modulation Systems", Univ. da California, Santa Cruz, Computer Research Laboratory - Technical Report, junho, 1987.
- [THO 74] Thomas, C. M.; Weidner, M. Y. e Durrani, S. H.; "Digital Amplitude-Phase Keying with M-ary Alphabets", IEEE Trans. on Comm., p. 168-179, fevereiro, 1974.
- [UNG 82] Ungerboeck, Gottfried; "Channel Coding with Multilevel/ Phase Signals", IEEE Transactions on Information Theory, p. 55-67, janeiro, 1982.
- [VIT 67] Viterbi, A. J.; "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm", IEEE Trans. on Inf. Theory, p. 260-269, abril, 1967.
- [WEI 84a] Wei, L. F.; "Rotationally Invariant Convolutional Channel Coding with Expanded Signal Space - Part I: 180 °", IEEE J. on Sel. Areas in Comm., p. 659-671, setembro, 1984.
- [WEI 84a] Wei, L. F.; "Rotationally Invariant Convolutional Channel Coding with Expanded Signal Space - Part II : Nonlinear Codes", IEEE J. on Sel. Areas in Comm., p. 672-686, setembro, 1984.
- [WIL 91] Wille, E.C. Gomes & Godoy Jr., Walter; "Um Esquema não Algébrico para Modulação Codificada". Aceito no IX Congresso Chileno de Engenharia Elétrica, Arica - Chile.

- [WIL 91] Wille, E.C. Gomes & Godoy Jr., Walter; "Códigos Multiníveis para a Modulação Codificada". Anais do 9º Simpósio Brasileiro de Telecomunicações - São Paulo - SP, p. 3.4.1-3.4.5, setembro, 1991.
- [W&F 88] Williams, R. G. C. & Farrell P. G., "Reducing the Decoding Complexity of Block Coded Modulation Schemes", MIEEE, Communications Research Group, Department of Electrical Engineering, University of Manchester, 1988.

Estas curvas são comparadas com as curvas de desempenho para as modulações 8-QAM e 4-QAM não codificadas, respectivamente. À Taxa de Erro de Bit igual a 10^{-4} , a primeira versão do sistema ($R = 3/4$) apresenta um ganho em codificação de aproximadamente 1,0 dB em relação ao sistema 8-QAM não codificado. O ganho assintótico teórico calculado é igual a 3 dB. No caso da segunda versão ($R = 1/2$) observa-se que seu desempenho começa a tornar-se melhor que o desempenho da modulação 4-QAM a partir de uma relação sinal-ruído de 12 dB. Este comportamento é explicado em função do baixo ganho assintótico (0,97 dB) previsto para esta versão.

Do exposto torna-se evidente que a versão V1 do sistema apresenta vantagens sobre a versão V2, uma vez que, a primeira, além de permitir um ganho em codificação maior do que a última em relação aos respectivos sistemas não codificados de referência, apresenta uma complexidade em decodificação menor.

5.6 - CIRCUITO INTEGRADO PARA CODIFICAÇÃO E DECODIFICAÇÃO

O circuito integrado para codificação e decodificação realiza as operações necessárias conforme os diagramas das figuras 5.3 e 5.4, trabalhando apenas com as informações binárias. Os processos de modulação e demodulação não são realizados por este circuito. A pinagem do circuito integrado é mostrado na figura 5.7.

Durante a operação do sistema decodificador ACI, este constantemente acessa as informações armazenadas numa memória externa. A memória (EPROM) está dividida em quatro regiões nas quais estão armazenadas a matriz geradora do código em uso, as matrizes conjunto de informação, os conjuntos de máscaras associadas e uma tabela de distâncias entre os pontos da constelação 16-QAM. O uso da EPROM externa deve-se a dois fatores :