UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE COMUNICAÇÕES

CÓDIGOS DE BLOCO SOBRE ANÉIS DE INTEIROS APLICADOS ÀS MODULAÇÕES QAM

Hélio Pires de Almeida

Orientador: **Prof. Dr. Renato Baldini Filho**

Tese apresentada à Faculdade de Engenharia Elétrica e de Computação - FEEC, da Universidade Estadual de Campinas - UNICAMP, como parte dos requisitos exigidos para obtenção do título de DOUTOR EM ENGENHARIA ELÉTRICA.

Agosto - 1997

Campinas - SP.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE -UNICAMP

Almeida, Helio Pires de

AL64c Códigos de bloco sobre anéis de inteiros aplicados às modulações QAM/ Hélio Pires de Almeida.- -Campinas, SP: [s.n.], 1997.

Orientador: Renato Baldini Filho

Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Códigos de controle de erros (Teoria da informação). 2. Modulação digital. 3. Decodificadores (Eletrônica). 4. Teoria dos grupos. 5. Anéis (Álgebra). I. Baldini Filho, Renato. II Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Para

Meus pais **Zezinho** e **Terezinha**

Minha esposa **Marcia**

Minha filha **Rafaela**

Agradecimentos

- Ao Professor Doutor Renato Baldini Filho, pela sua orientação eficiente e segura, fator fundamental na realização deste trabalho.

- Aos Professores da Banca Examinadora: Prof. Dr. Carlos Eduardo Câmara (CPqD-Telebrás), Prof. Dr. Walter Godoy Júnior (CEFET - PR), Prof. Dr. Dalton Soares Arantes (FEEC-UNICAMP) e Prof. Dr. Lee Luan Ling (FEEC-UNICAMP)

- Ao colega do Departamento de Matemática do CCEN-UFPb, Professor Lenimar Nunes de Andrade, que gentilmente desenvolveu os programas computacionais que usamos neste trabalho.

- Aos colegas do Curso, pelo companheirismo e ajuda mútua. De modo particular, aos amigos José Raimundo, Manoel Bezerra e Martinho Araujo.

- Aos colegas do Departamento de Matemática do CCEN-UFPb, que possibilitaram a realização deste trabalho. De modo particular, aos Professores Antônio de Andrade e Antônio Sales.

- À minha esposa Marcia e à minha filha Rafaela, pelo amor, carinho, paciência, compreensão e estímulo.

- A todos que confiaram em mim, na realização desta tarefa.

Resumo

Neste trabalho, apresentamos uma técnica de modulação $4^m - QAM$ codificada, baseada em códigos de bloco multiníveis, definidos sobre o anel $\mathbb{Z}_4$, onde $m \geq 2$ é um inteiro. Os sinais da modulação são rotulados por m-uplas, cujos símbolos pertencem a $\mathbb{Z}_4$. O processo de codificação usa m códigos em correspondência com os símbolos dos rótulos, e cada código faz a proteção do respectivo símbolo, de forma independente, visando maximizar a distância Euclidiana entre as palavras código. É usado um particionamento de conjunto que divide a constelação $4^m - QAM$ em subconjuntos com distâncias Euclidianas progressivamente crescentes. Foram obtidos ganhos de codificação assintóticos de até $6\,dB$, onde os esquemas codificado e não codificado têm as mesmas taxas de transmissão de informação. O uso de códigos sobre $\mathbb{Z}_4$ permitiu encontrar códigos que são invariantes às ambiguidades de fase da portadora. Também foi apresentado um método para decodificação desses códigos.

Abstract

This thesis presents a coded $4^m - QAM$ modulation technique based on multilevel block codes over the ring of integer $\mathbb{Z}_4$, where $m \geq 2$ is an integer. The modulation signals are labelled by m-tuples, whose symbols are defined over $\mathbb{Z}_4$. The encoding process uses m multilevel block codes over $\mathbb{Z}_4$. Each code protects its corresponding symbol in an independent way in order to maximize the Euclidian distance between codewords. A mapping by set partitioning is used to divide the $4^m - QAM$ constellation in subsets with increasing Euclidian distances. Asymptotic coding gains up to $6\,dB$ were obtained for coded $4^m - QAM$ modulations over equivalent uncoded modulation schemes. The use of $\mathbb{Z}_4$-codes allowed us to find codes which are invariant to phase ambiguities of the carrier for $4^m - QAM$ schemes. A decoding method for these codes are also presented.

Lista de Símbolos

$\mathbf{a}$	sequência de informação
$\mathcal{A}$	anel
$\mathcal{A}^n$	conjunto das n-uplas sobre $\mathcal{A}$
$\mathcal{A}^{\mathbb{Z}}$	conjunto das sequências sobre $\mathcal{A}$
$\mathcal{A}((D))$	conjunto das séries de Laurent sobre $\mathcal{A}$
$\mathcal{A}[D]$	conjunto dos polinômios sobre $\mathcal{A}$
BCM	modulação codificada por código de bloco
$\mathbf{c}$	sequência codificada
$\mathbf{C}$	codificador
$\mathcal{C}$	código linear
$del(\mathbf{a}(D))$	atraso de uma sequência $\mathbf{a}(D)$
$d_H(\mathbf{x}, \mathbf{y})$	distância de Hamming entre $\mathbf{x}$ e $\mathbf{y}$
$dim\mathcal{V}$	dimensão de um espaço vetorial $\mathcal{V}$
D	elemento de atraso
D_E^2	distância Euclidiana quadrada
d_0^2, D_0^2	distância Euclidiana quadrada mínima de um conjunto
$D_{E_{\min}}^2$	distância Euclidiana quadrada mínima
$D_{E_{nc}}^2$	distância Euclidiana quadrada mínima de um esquema não codificado
D_{free}^2	distância Euclidiana quadrada mínima de um código convolucional
DI	distância invariante
e	elemento neutro de um grupo, base do logaritmo natural
E_b	energia média de bit de informação
$\frac{E_b}{N_0}$	relação energia média de bit por ruído
$f(D)$	polinômio de realimentação de um CCM
f_j	coeficientes de um polinômio de realimentação
$\mathbb{F}$	corpo

g	ganho de codificação
$g^{(i)}(D)$	elementos de uma matriz geradora $\mathbf{G}(D)$
$g_j^{(i)}$	coeficientes de um polinômio $g^{(j)}(D)$
g_∞	ganho de codificação assintótico
$\mathbf{G}$	matriz geradora de um código de bloco
$GF(q)$	corpo de Galois de ordem q
$\mathbf{G}(D)$	matriz geradora de um código convolucional
$\mathcal{G}$	grupo
$\mathbf{H}$	matriz verificação de paridade do um código de bloco
$\mathbf{H}(D)$	matriz verificação de paridade do um código convolucional
$\mathcal{H}$	subgrupo
$\mathbf{I}_k$	matriz identidade de ordem k
j	unidade imaginária dos números complexos
k	dimensão de um código
LD	linearmente dependente
$\mathbf{M}^t$	transposta de uma matriz $\mathbf{M}$
M_c	número de sinais de um esquema codificado
M_{nc}	número de sinais de um esquema não codificado
LI	linearmente independente
$mdc\{a,b\}$	máximo divisor comum entre a e b
$\mathcal{M}$	módulo
$\mathbf{n}(t)$	vetor ruído
N	número de estados de um CCM
n	comprimento de um código
N_{free}	número médio de vizinhos com D^2_{free}
$\mathbf{P}$	matriz de paridade de um código de bloco
N_v	número de vizinhos com $D^2_{E_{\min}}$

$p(\mathbf{v})$	soma módulo 4 dos símbolos de uma palavra $\mathbf{v}$
$\mathbf{P}(D)$	matriz de paridade de um código convolucional
$R,\ R_t$	taxa de codificação
$\mathbf{r}$	sequência recebida
rad	radiano
RI	rotacionalmente invariante
$\mathbb{R}$	conjunto dos números reais
$s(t)$	sinal de uma constelação, forma de onda
$s(x)$	sinal de um número real x
$\mathbf{s}(t)$	sequência de sinais
$\mathbf{s}(\mathbf{v})$	síndrome de uma palavra $\mathbf{v}$
T	intervalo de tempo
TCM	modulação codificada por código convolucional
$\mathbf{u}$	palavra código toda 1
$\mathbf{v}$	vetor, palavra recebida
$v^{(\tau)}$	valor armazenado num elemento de memória, num instante τ
$\mathcal{V}$	espaço vetorial
w	peso
w_H	peso de Hamming
w_E^2	peso Euclidiano quadrado
$w_{E_{\min}}^2$	peso Euclidiano quadrado mínimo
$\mathcal{W}$	subespaço vetorial
$\mathbf{x}$	vetor
$\mathbb{Z}$	conjunto dos números inteiros
$\mathbb{Z}_q$	conjunto dos números inteiros módulo q
$\mathbb{Z}_q^n$	conjunto das n-uplas sobre $\mathbb{Z}_q$
$m\mathbb{Z}$	conjunto dos números inteiros múltiplos de m

(n, k)	código de bloco linear de comprimento n e dimensão k
$(\mathcal{S}, \star)$	conjunto $\mathcal{S}$ munido de uma operação binária $\star$
$[\mathcal{G}:\mathcal{H}]$	índice de um subgrupo $\mathcal{H}$ em um grupo $\mathcal{G}$
$[\varphi]_{\beta_1}^{\beta_2}$	matriz de uma transformação linear φ em relação às bases β_1 e β_2
$\|.\|$	cardinalidade de um conjunto, valor absoluto de um número
$\langle S \rangle$	subespaço vetorial ou submódulo gerado por S
$\oplus$	adição módulo q
$\ominus$	diferença módulo q
$\simeq$	relação de isomorfismo
β	base de um espaço vetorial
δ_i, Δ_i	parâmetros de confiabilidade de um símbolo
$\partial(\mathbf{a}(D))$	grau de uma sequência $\mathbf{a}(D)$
$\rho(\mathbf{x})$	rotação de um vetor $\mathbf{x}$ por um ângulo $\frac{2\pi}{q}\, rad$
$\rho_r(\mathbf{x})$	rotação de um vetor $\mathbf{x}$ por um ângulo $\frac{2r\pi}{q}\, rad$
τ	instante de tempo
φ	homomorfismo, transformação linear
$\emptyset$	conjunto vazio

Lista de Figuras

Lista de Tabelas

Conteúdo

Capítulo 1

Introdução

1.1 Introdução

Em 1974 Massey [20] mostrou que ganhos significativos no desempenho de sistemas de comunicações digitais poderiam ser obtidos com a realização simultânea das operações de modulação e codificação mas, até 1982, essas operações continuaram sendo realizadas de forma independente, como mostra a figura 1.1. No sistema de comunicação digital mostrado na figura 1.1, o codificador recebe uma sequência de informação $\mathbf{a}$, que depois de codificada resulta na sequência codificada $\mathbf{c}$. O modulador faz o mapeamento dos símbolos da sequência codificada, nos elementos do espaço de sinais, e associa a cada sinal mapeado uma forma de onda $s(t)$. Assim, a sequência codificada $\mathbf{c}$ é mapeada numa sequência de sinais $\mathbf{s}(t)$, que é enviada através do canal. Neste trabalho assumiremos que o canal é Gaussiano com ruído branco aditivo. Então, na saida do canal temos uma sequência de sinais $\tilde{\mathbf{s}}(t) = \mathbf{s}(t) + \mathbf{n}(t)$, onde $\mathbf{n}(t)$ é um vetor ruído introduzido pelo canal, cujas componentes são variáveis normais de média zero e variância $\sigma^2 = \frac{N_0}{2}$. O demodulador faz a operação inversa do modulador e envia para o decodificador uma sequência $\tilde{\mathbf{c}}$, que depois de decodificada resulta na sequência $\hat{\mathbf{a}}$. O uso de códigos corretores de erros tem como objetivo corrigir os possíveis erros na sequência $\tilde{\mathbf{c}}$, causados pelo ruído, e recuperar a sequência de informação $\mathbf{a}$, ou seja, obter $\hat{\mathbf{a}} = \mathbf{a}$.

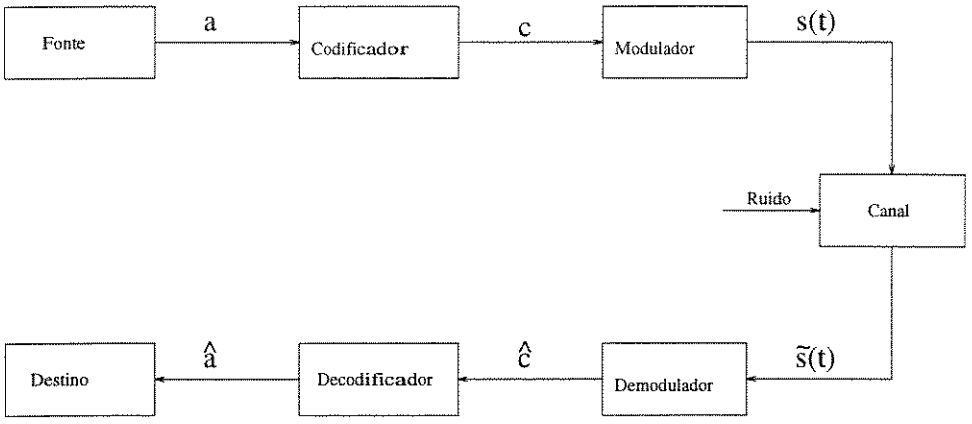

Figura 1.1: Sistema de Comunicação Digital

Podemos aumentar o desempenho de um sistema de comunicação digital, em termos de probabilidade de erro de bit, com a introdução de mais bits de redundância, mas este procedimento requer um aumento na faixa de passagem. Então, se o sistema é limitado em faixa, podemos optar por uma modulação com um número maior de pontos, de modo que os bits extras que forem adicionados possam ser acomodados nos símbolos do novo espaço de sinais. Mas, se o novo sistema tem a mesma distância Euclidiana entre os pontos da modulação, a sua energia média é maior do que a do anterior. Assim, se o sistema é limitado em potência, essa solução não deve ser adotada. Portanto, se o sistema é limitado em faixa e potência, e já opera no limite dessas especificações, nenhuma dessas soluções pode ser adotada. No entanto, o desempenho de um tal sistema ainda pode ser otimizado, tomando uma modulação com um número maior de pontos, mantendo-se a taxa de transmissão de informação e a energia média do sistema. Isso pode ser conseguido combinando-se as operações de modulação e codificação.

Em 1977, no trabalho de Imai e Hirakawa [14], as operações de modulação e codificação foram feitas simultaneamente, num método de codificação multinível que utilizava vários códigos de bloco binários. Essa técnica de combinar as operações de modulação e codificação de um sistema de comunicação digital é conhecida por **modulação codificada**. Quando se

usa código convolucional (treliça) numa modulação codificada, ela é chamada **modulação codificada por treliça** ou simplesmente **TCM** (abreviatura da expressão em Inglês, *Trellis Coded Modulation*). De modo análogo, se a codificação é feita com códigos de bloco, ela é abreviada por **BCM** (*Block Coded Modulation*). A figura 1.2 mostra um sistema de comunicação digital onde é usada a técnica de modulação codificada. Os blocos que representam a modulação e a codificação são tomados como um só, o mesmo acontecendo com os blocos que representam a demodulação e a decodificação.

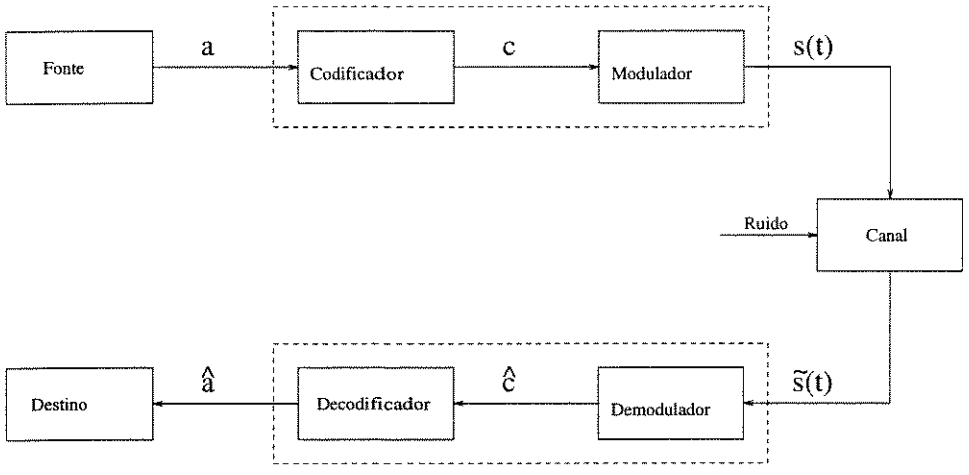

Figura 1.2: Sistema de Modulação Codificada.

Foi em 1982 que a técnica de modulação codificada se consolidou, com o célebre trabalho de Ungerboeck [28], onde ele apresentou uma técnica, conhecida por **mapeamento por particionamento de conjunto**, onde mostrou como implementar sistemas de comunicações digitais onde as operações de modulação e codificação são realizadas simultaneamente, de modo que ganhos significativos de codificação poderiam ser obtidos, sem necessidade de diminuição da taxa de transmissão de informação nem expansão da faixa de passagem. O método de particionamento consiste em dividir sucessivamente o conjunto de pontos do espaço de sinais em subconjuntos com uma distância Euclidiana mínima progressivamente crescente. O trabalho de Ungerboeck despertou o interesse de muitos pesquisadores e contribuiu para o surgimento de vários trabalhos nessa área. Ginzburg [12] utiliza uma

"construção hierárquica", que generaliza o trabalho de Imai e Hirakawa [14]. Em [9] Cusack usou modulações QAM codificadas com um conjunto de códigos de Reed-Muller. Essa idéia de Cusack foi generalizada por Sayegh [27], que utilizou outros códigos de bloco binários já conhecidos e considerou também modulações PSK.

Uma nova técnica de modulação codificada foi introduzida por Baldini *et al* [5], onde são utilizados códigos definidos sobre os anéis de inteiros módulo q, em modulações $q-PSK$. Neste caso não há necessidade de particionamento de conjunto pois os símbolos dos códigos são mapeados diretamente nos símbolos dos sinais da constelação. López [17] usou códigos convolucionais definidos sobre o anel de inteiros módulo 4, em modulações $4^m - QAM$, onde $m \geq 2$ é um inteiro. O processo de codificação é baseado num método de particionamento de conjunto que tem como base o método de particionamento de Ungerboeck. Baldini [2] usou códigos de bloco definidos sobre o anel de inteiros módulo 4, em modulações $16 - QAM$. Cada sinal da modulação é rotulado por um par de símbolos pertencentes ao anel de inteiros módulo 4, e o processo de codificação utiliza dois códigos, cada um atuando em um dos símbolos dos rótulos dos sinais. Aqui também é usado um método de particionamento de conjunto semelhante ao de Ungerboeck [28].

1.2 Organização desta Tese

Neste trabalho usaremos códigos definidos sobre os aneis de inteiros módulo q aplicados à modulações QAM. Esses códigos são chamados de códigos multiníveis sobre $\mathbb{Z}_q$. O conteúdo desta tese está distribuído da seguinte maneira:

No capítulo 2, apresentamos os conceitos e resultados algébricos necessários para o desenvolvimento desta tese. As definições de códigos de bloco e convolucionais sobre anéis, seus parâmetros principais e alguns resultados básicos também são apresentados.

No capítulo 3, apresentamos uma técnica de modulação codificada, introduzida por Baldini *et al* [5], onde esquemas de modulações $q-PSK$ são codificados com códigos multiníveis sobre $\mathbb{Z}_q$. No processo de codificação, em vez de se usar mapeamento por particionamento

de conjuntos, os bits de informação são mapeados diretamente nos símbolos de $\mathbb{Z}_q$ e esses símbolos são então codificados. Isso se deve à semelhança da distribuição espacial dos sinais de uma constelação $q - PSK$ com a representação geométrica e as propriedades do anel $\mathbb{Z}_q$. Essa semelhança também proporciona a esses códigos boas propriedades relativas à linearidade e distância Euclidiana, além de favorecer a busca por códigos rotacionalmente invariantes. O problema de rotação de fase da portadora, introduzida pelo canal, também é analisado. Códigos de bloco e convolucionais são considerados.

No capítulo 4, mostramos um método de rotulamento dos pontos dos espaços de sinais $4^m - QAM$ retangulares, onde cada ponto do espaço é rotulado por uma m-upla ordenada $x_1x_2\ldots x_m$, com $x_i \in \mathbb{Z}_4$, que viabiliza a busca de códigos multiníveis sobre $\mathbb{Z}_4$, rotacionalmente invariantes sob todas as ambiguidades de fase da modulação $4^m - QAM$, e analisamos o problema de rotação de fase. Também apresentamos uma técnica de TCM, introduzida por López [17], onde modulações retangulares $4^m - QAM$, $m \geq 2$ inteiro, são codificadas com códigos convolucionais multiníveis definidos sobre o anel de inteiros módulo 4. O processo de codificação utiliza uma técnica de TCM 4-Dimensional, onde pares de sinais do espaço 2-Dimensional $4^m - QAM$ são associados aos ramos da treliça. Essa técnica é baseada num particionamento de conjunto que tem como base o particionamento de conjunto proposto por Ungerboeck [28]. O uso de códigos definidos sobre $\mathbb{Z}_4$, em modulações retangulares $4^m - QAM$, permite que códigos rotacionalmente invariantes sejam encontrados. Alguns exemplos de TCM são apresentados.

No capítulo 5, apresentamos a nossa principal contribuição para esta tese, que consiste em estender para modulações $4^m - QAM$, $m \geq 2$ inteiro, a técnica de modulação codificada que Baldini usou em modulações $16 - QAM$ [2]. O processo de codificação utiliza m códigos de bloco multiníveis sobre $\mathbb{Z}_4$, $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m$, todos de mesmo comprimento, onde cada código protege o símbolo correspondente dos rótulos dos pontos do espaço $4^m - QAM$, sendo o rotulamento desses pontos feito de acordo com o método descrito no capítulo 4. Para otimizarmos o desempenho desses códigos, fazemos um particionamento do conjunto de pontos do espaço, de modo que, em cada nível da partição, cada subconjunto é dividido

em quatro subconjuntos de mesma cardinalidade e com distâncias Euclidianas quadradas mínimas iguais a quatro vezes às dos subconjuntos do nível anterior. Esse processo de codificação é uma extensão do processo de Sayegh [27] para códigos multiníveis, em modulações $4^m - QAM$. Para a decodificação desses códigos, estendemos o processo de decodificação de Sayegh [27] para códigos multiníveis. Exemplos de BCM onde são obtidos ganhos de codificação assintóticos de até $6\,dB$ também são apresentados.

No capítulo 6 apresentamos as conclusões sobre o nosso trabalho e sugestões para pesquisa nessa área.

Capítulo 2

Introdução à Álgebra e à Teoria da Codificação

2.1 Introdução

Neste capítulo apresentamos os conceitos e resultados básicos necessários para o entendimento e o desenvolvimento deste trabalho. Na seção 2.2 são apresentados os conceitos de grupos, anéis, corpos, espaços vetoriais e módulos. Esses conceitos serão definidos de forma genérica, embora o nosso interesse esteja restrito aos grupos e anéis de inteiros módulo q, uma vez que o objetivo do nosso trabalho está dirigido para códigos que são definidos sobre esses anéis. Para tornarmos o texto mais leve, omitiremos as demonstrações de alguns resultados, que podem ser encontradas na bibliografia citada.

Na seção 2.3 definimos os conceitos de códigos, seus principais parâmetros e alguns resultados básicos que serão usados nos capítulos seguintes.

2.2 Introdução à Álgebra

2.2.1 Grupos

Uma **operação binária** $\star$ num conjunto não vazio $\mathcal{S}$ é uma regra que associa a cada par ordenado de elementos de $\mathcal{S}$ algum elemento de $\mathcal{S}$. Em outras palavras, $\star$ é uma função

$$\begin{array}{rccc} \star & : & \mathcal{S} \times \mathcal{S} & \longrightarrow & \mathcal{S} \\ & & (x, y) & \longmapsto & x \star y \end{array}$$

Vamos denotar um conjunto **S** munido de uma operação binária $\star$ por $(\mathcal{S}, \star)$.

Definição 2.1 *Um conjunto não vazio $\mathcal{G}$ munido de uma operação binária $\star$ é dito um* ***grupo****, se as propriedades seguintes são satisfeitas:*

i) $x \star (y \star z) = (x \star y) \star z$, $\forall x, y, z \in \mathcal{G}$. *(propriedade associativa)*
ii) Existe um (único) elemento e em $\mathcal{G}$ tal que $x \star e = e \star x = x, \forall x \in \mathcal{G}$. Este elemento é chamado de ***elemento neutro*** *ou* ***elemento identidade*** *da operação $\star$ em $\mathcal{G}$.*
iii) Para cada $x \in \mathcal{G}$, existe um (único) $x' \in \mathcal{G}$ tal que $x \star x' = x' \star x = e$. O elemento x' é o ***inverso*** *de x com relação a operação $\star$ em $\mathcal{G}$.*

Se além das propriedades acima, a operação $\star$ é comutativa, ou seja, $x \star y = y \star x, \forall x, y \in \mathcal{G}$, então dizemos que $\mathcal{G}$ é um grupo **comutativo** ou **abeliano**.

Exemplo 2.1 *Um dos exemplos mais naturais de grupo abeliano é o conjunto dos números inteiros munido da operação usual de adição.*

Definição 2.2 *Um grupo $\mathcal{G}$ é dito* ***finito*** *se o número de elementos de $\mathcal{G}$ é finito. Neste caso, se o número de elementos é igual a n dizemos que $\mathcal{G}$ tem* ***ordem*** *n e denotamos isto por $|\mathcal{G}| = n$.*

Definição 2.3 *Sejam $(\mathcal{G}, \star)$ um grupo e $\mathcal{H}$ um subconjunto não vazio de $\mathcal{G}$. Dizemos que $\mathcal{H}$ é um* ***subgrupo*** *de $\mathcal{G}$, e denotamos isto por $\mathcal{H} \leq \mathcal{G}$, se as condições seguintes são satisfeitas:*

i) $x \star y \in \mathcal{H}$, $\forall x,y \in \mathcal{H}$

ii) Se $x \in \mathcal{H}$ *então o inverso de* x *também pertence a* $\mathcal{H}$

Definição 2.4 *Sejam* $(\mathcal{G}, \star)$ *um grupo,* $\mathcal{H} \leq \mathcal{G}$ ***e*** $g \in \mathcal{G}$. *O conjunto*

$$g \star \mathcal{H} = \{x \in \mathcal{G} \mid x = g \star h,\ h \in \mathcal{H}\}$$

é chamado ***classe lateral*** *(à esquerda) de* $\mathcal{H}$ *em* $\mathcal{G}$. *De modo análogo define-se classe lateral à direita.*

Observação 2.1 *Se* $(\mathcal{G}, \star)$ *é um grupo e* $\mathcal{H} \leq \mathcal{G}$, *temos que:*

i) Duas classes laterais ou são iguais ***ou*** *são disjuntas, isto é, se*

$$x \star \mathcal{H} \neq y \star \mathcal{H} \quad \textit{então} \quad (x \star \mathcal{H}) \cap (y \star \mathcal{H}) = \emptyset.$$

ii) Todo elemento de $\mathcal{G}$ *pertence a* ***alguma*** *classe lateral e, portanto, o conjunto de classes laterais forma uma partição do conjunto* $\mathcal{G}$.

iii) Existe uma correspondência ***biunívoca*** *entre quaisquer duas classes laterais de* $\mathcal{H}$. *Como* $e \star \mathcal{H} = \mathcal{H}$, *segue que todas as classes* ***laterais*** *de* $\mathcal{H}$ *têm a mesma cardinalidade de* $\mathcal{H}$. *Em particular, se* $\mathcal{H}$ *é finito, todas as* ***classes*** *laterais de* $\mathcal{H}$ *têm o mesmo número de elementos de* $\mathcal{H}$.

Definição 2.5 *O número de classes* ***laterais*** *de* $\mathcal{H}$ *em* $\mathcal{G}$ *é chamado* ***índice*** *de* $\mathcal{H}$ *em* $\mathcal{G}$ *e é denotado por* $[\mathcal{G} : \mathcal{H}]$.

Teorema 2.1 *[11] Se* $\mathcal{G}$ *é um grupo* ***finito*** *e* $\mathcal{H}$ *é um subgrupo de* $\mathcal{G}$ *então a ordem de* $\mathcal{H}$ *divide a ordem de* $\mathcal{G}$. *Mais precisamente,* $|\mathcal{G}| = |\mathcal{H}|\,[\mathcal{G} : \mathcal{H}]$.

Definição 2.6 *Sejam* $(\mathcal{G}_1, \star_1)$ *e* $(\mathcal{G}_2, \star_2)$ *grupos e* $\varphi : \mathcal{G}_1 \to \mathcal{G}_2$ *uma função. Se*

$$\varphi(x \star_1 y) = \varphi(x) \star_2 \varphi(y), \quad \forall x,y \in \mathcal{G}_1$$

então dizemos que φ *é um* ***homomorfismo***. *Se, além disso,* φ *é uma bijeção então dizemos que* φ *é um* ***isomorfismo***. *Neste caso, dizemos que os grupos* $\mathcal{G}_1$ *e* $\mathcal{G}_2$ *são* ***isomorfos*** *e denotamos isto por* $\mathcal{G}_1 \simeq \mathcal{G}_2$.

Sejam $(\mathcal{G},\star)$ um grupo, a um elemento qualquer de $\mathcal{G}$ e n um inteiro positivo. Se $\star$ é uma operação multiplicativa escrevemos

$$a^n \triangleq \underbrace{a\star\cdots\star a}_{n \text{ vezes}} \quad e \quad a^{-n} \triangleq \underbrace{a^{-1}\star\cdots\star a^{-1}}_{n \text{ vezes}}.$$

Quando a operação em $\mathcal{G}$ é aditiva escrevemos na em vez de a^n e $-(na)$ em vez de a^{-n}. Seja $\mathcal{H} = \{a^n \mid n \in \mathbb{Z}\}$. É fácil ver que $\mathcal{H}$ é um subgrupo de $\mathcal{G}$. Este subgrupo é chamado **subgrupo gerado por** a. $\mathcal{H}$ é o menor subgrupo de $\mathcal{G}$ que contém a. A **ordem** do elemento a é definida como sendo igual a ordem de $\mathcal{H}$.

Definição 2.7 *Seja $\mathcal{G}$ um grupo. Se existir um elemento g em $\mathcal{G}$ tal que o subgrupo gerado por g é igual a $\mathcal{G}$ então dizemos que $\mathcal{G}$ é um* ***grupo cíclico*** *e que g é um* ***gerador de*** *$\mathcal{G}$.*

Exemplo 2.2 *Consideremos um inteiro $q \geq 2$ e seja $\mathbb{Z}_q = \{0, 1, \ldots, q-1\}$. Vamos considerar em $\mathbb{Z}_q$ a operação binária $\oplus$ definida assim: Se x,$y \in \mathbb{Z}_q$, $x \oplus y$ = resto da divisão de $x + y$ por q, onde $+$ denota a operação usual de adição. Esta operação é chamada* ***adição módulo*** *q. O conjunto $\mathbb{Z}_q$ com a operação $\oplus$ é um grupo cíclico. (cf.[11]).*

Exemplo 2.3 *Seja $q \geq 2$ um número inteiro e seja $\mathcal{G}$ o conjunto das raizes q-ésimas da unidade, isto é, $\mathcal{G} = \left\{e^{\frac{2k\pi j}{q}},\ k = 0, 1, \ldots, q-1\right\}$. $\mathcal{G}$ munido com a operação usual de produto é um grupo cíclico.*

Observação 2.2 *No plano complexo as raizes q-ésimas da unidade formam os vértices de um polígono regular de q lados inscrito na circunferência de raio 1 e centro na origem. A figura 2.1 ilustra o caso onde $q = 8$. Se $r' = e^{\frac{2k'\pi j}{q}}$ e $r'' = e^{\frac{2k''\pi j}{q}}$ então $r'r''$ é a raiz representada pelo vértice que se obtém percorrendo a circunferência no sentido anti-horário e deslocando-se k'' vértices a partir de r'. Se considerarmos $r = e^{\frac{2\pi j}{q}}$, temos que todas as outras raizes são potências de r pois $e^{\frac{2k\pi j}{q}} = r^k$. Observemos também que a multiplicação de uma raiz por r significa, no plano complexo, deslocar-se para o vértice seguinte, no sentido anti-horário. Isso justifica o termo "grupo cíclico".*

A representação cíclica acima também é adequada para o grupo $\mathbb{Z}_q$, pois enumerando os

vértices de 0 *a* $q-1$ *no sentido anti-horário, a soma* $k' \oplus k''$ *corresponde a deslocar-se* k'' *vértices a partir do vértice* k'*, no sentido anti-horário. Daí concluimos que as operações binárias desses dois grupos são semelhantes, ou seja, multiplicar duas raizes em* $\mathcal{G}$ *é equivalente a somar dois elementos em* $\mathbb{Z}_q$*. Assim, esperamos que esses dois grupos sejam isomorfos. Isto de fato ocorre pois a função* $\varphi : \mathbb{Z}_q \to \mathcal{G}$*, definida por* $\varphi(k) = e^{\frac{2k\pi i}{q}}$ *é um isomorfismo. Isto também pode ser deduzido a partir do teorema seguinte.*

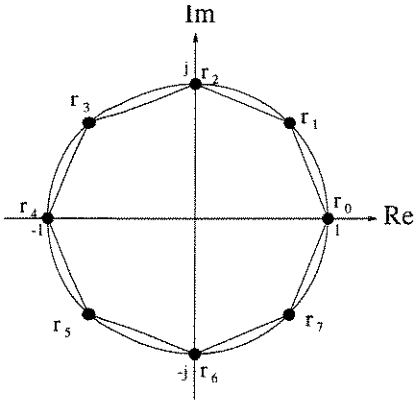

Figura 2.1: Representação Gráfica das raízes oitavas da unidade.

Teorema 2.2 *[11]Quaisquer dois grupos cíclicos de mesma ordem são isomorfos.*

Teorema 2.3 *[11] Se* $\mathcal{G}$ *é um grupo cíclico então:*

i) $\mathcal{G}$ *é abeliano*

ii) Todo subgrupo de $\mathcal{G}$ *é cíclico.*

2.2.2 Anéis

Definição 2.8 *Seja* $\mathcal{A}$ *um conjunto munido de duas operações binárias, soma e produto, que denotaremos por* $+$ *e* $\cdot$*, respectivamente. (As operações* $+$ *e* $\cdot$ *podem ser diferentes das operações usuais de soma e produto). Dizemos que* $\mathcal{A}$ *é um* ***anel*** *se as propriedades seguintes*

são satisfeitas:

i) O conjunto $\mathcal{A}$ munido com a operação $+$ é um grupo abeliano

ii) $x \cdot (y \cdot z) = (x \cdot y) \cdot z$, $\forall x, y, z \in \mathcal{A}$. *(propriedade associativa)*

iii) $x \cdot (y+z) = x \cdot y + x \cdot z$ *e* $(x+y) \cdot z = x \cdot z + y \cdot z$, $\forall x, y, z \in \mathcal{A}$. *(propriedade distributiva)*

Observação 2.3 *Para simplificar a notação, omitiremos o símbolo da operação produto e escreveremos xy em vez de $x \cdot y$, sempre que não haja ambiguidade.*

Observação 2.4 *Seja $\mathcal{A}$ um anel:*

1. *Se a operação produto é comutativa em $\mathcal{A}$, ou seja, $xy = yx$, $\forall x, y \in \mathcal{A}$, então dizemos que $\mathcal{A}$ é um anel* **comutativo**.

2. *Se a operação produto tem um elemento identidade em $\mathcal{A}$, ou seja, se existe um elemento $1 \in \mathcal{A}$ tal que $1x = x1 = x$, $\forall x \in \mathcal{A}$, então dizemos que $\mathcal{A}$ é um anel com* **identidade**.

Definição 2.9 *Seja $\mathcal{A}$ um anel e seja $\mathcal{S}$ um subconjunto não vazio de $\mathcal{A}$. Dizemos que $\mathcal{S}$ é um* ***subanel*** *de $\mathcal{A}$ se:*

i) $\mathcal{S}$ é um subgrupo de $\mathcal{A}$, com a operação soma;

ii) $xy \in \mathcal{S}$, $\quad \forall x, y \in \mathcal{S}$.

Definição 2.10 *Seja $\mathcal{A}$ um anel com identidade 1. Dizemos que um elemento $x \in \mathcal{A}$ é* ***inversível*** *se existir um elemento $y \in \mathcal{A}$ tal que $xy = yx = 1$. Neste caso, y é o* ***inverso*** *(multiplicativo) de x e é denotado por x^{-1}.*

Definição 2.11 *Seja $\mathcal{A}$ um anel e seja $a \in \mathcal{A}$, $a \neq 0$. Se existir $b \in \mathcal{A}$, $b \neq 0$, tal que $ab = 0$ ou $ba = 0$, dizemos que a é um* ***divisor de zero***.

Se $a \neq 0$ não é um divisor de zero então vale a lei do cancelamento, ou seja,

$$ab = ac \Longrightarrow b = c.$$

De fato, $ab = ac \Longrightarrow ab - ac = 0 \Longrightarrow a(b-c) = 0$. Como $a \neq 0$ e não é divisor de zero, devemos ter $b - c = 0$ e, portanto, $b = c$.

Exemplo 2.4 *Consideremos o conjunto $\mathbb{Z}_q$ com as operações de soma e produto módulo q. (O produto módulo q é definido de modo semelhante à soma módulo q.) Temos que (cf.[11]):*

1. *Com essas operações $\mathbb{Z}_q$ é um anel comutativo com identidade, chamado* **anel de inteiros módulo** q.

2. *Se q não é um número primo então $\mathbb{Z}_q$ possui divisor(es) de zero.*

Na tabela 2.1 mostramos as operações de soma e produto módulo 4.

$\oplus$	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

$\cdot$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

Tabela 2.1: Operações de soma e produto módulo 4

Teorema 2.4 *Em um anel com identidade os elementos inversíveis formam um grupo, com a operação produto.*

Demonstração:

i) Se a e b são inversíveis então

$$(ab)(b^{-1}a^{-1}) = a(bb^{-1})a^{-1} = a1a^{-1} = aa^{-1} = 1.$$

Portanto ab é inversível.

ii) Se a é inversível então a^{-1} também é inversível, pois $(a^{-1})^{-1} = a$. ■

Teorema 2.5 *No anel $\mathbb{Z}_q$ os divisores de zero são precisamente aqueles elementos que não são relativamente primos com q.*

Demonstração: Seja $a \in \mathbb{Z}_q$ e seja $d = mdc\{a, q\}$. Se $d > 1$, então $1 \leq \frac{q}{d} \leq q - 1$. Logo, $\frac{q}{d} \neq 0$ em $\mathbb{Z}_q$. Tomando $b = \frac{q}{d}$ temos que $ab = a(\frac{q}{d}) = (\frac{a}{d})q = 0$, em $\mathbb{Z}_q$. Portanto a é um divisor de zero. Por outro lado, se $d = 1$ e $ab = 0$, em $\mathbb{Z}_q$, então $ab = mq$, $m \in \mathbb{Z}$. Daí segue que q divide ab. Como $d = 1$, então q divide b. Logo, $b = 0$ em $\mathbb{Z}_q$. Portanto a não é divisor de zero. ■

Observação 2.5 *De modo análogo mostra-se que em $\mathbb{Z}_q$ os elementos inversíveis são precisamente aqueles que são relativamente primos com q.*

Corolário 2.1 *Se q é primo então $\mathbb{Z}_q$ não possui divisor de zero.*

Observação 2.6 *Um elemento inversível não pode ser divisor de zero.*

Definição 2.12 *Sejam $\mathcal{A}_1$ e $\mathcal{A}_2$ anéis e $\varphi : \mathcal{A}_1 \to \mathcal{A}_2$ uma função. Dizemos que φ é um* ***homomorfismo*** *de anéis se:*

i) $\varphi(x + y) = \varphi(x) + \varphi(y)$, $\forall x, y \in \mathcal{A}_1$
ii) $\varphi(xy) = \varphi(x)\varphi(y)$, $\forall x, y \in \mathcal{A}_1$.

Se, além disso, φ é uma bijeção então dizemos que φ é um ***isomorfismo****. Neste caso, dizemos que os anéis $\mathcal{A}_1$ e $\mathcal{A}_2$ são isomorfos e denotamos isto por $\mathcal{A}_1 \simeq \mathcal{A}_2$.*

Sequências e Espaços de Sequências

Seja $\mathcal{A}$ um anel comutativo com identidade. Uma **sequência a** sobre $\mathcal{A}$ é uma função

$$\begin{array}{rccc} a & : & \mathbb{Z} & \longrightarrow & \mathcal{A} \\ & & i & \longmapsto & a(i) = a_i \end{array}$$

Usamos a notação

$$\mathbf{a} = (\ldots, a_{-i}, a_{-i+1}, \ldots, a_0, a_1, \ldots, a_j, \ldots) \tag{2.1}$$

para representar a sequência $\mathbf{a}$. Outra notação para essa sequência é

$$\mathbf{a} = \mathbf{a}(D) = \sum_{i \in \mathbb{Z}} a_i D^i, \tag{2.2}$$

onde D é uma indeterminada. Se existir $r \in \mathbb{Z}$ tal que $a_i = 0, \quad \forall i < r$, escrevemos

$$\mathbf{a}(D) = \sum_{i=r}^{\infty} a_i D^i. \tag{2.3}$$

O conjunto de todas as sequências sobre $\mathcal{A}$ é denotado por $\mathcal{A}^{\mathbb{Z}}$.

Seja $\mathbf{a}(D) = \sum_{i \in \mathbb{Z}} a_i D^i$ uma sequência não nula sobre $\mathcal{A}$. Então:

1. Se existir um inteiro r tal que $a_r \neq 0$ e $a_i = 0$, para $i < r$, dizemos que r é o **atraso** de $\mathbf{a}(D)$, e denotamos isto por $r = del(\mathbf{a}(D))$. Se não existir um tal inteiro r, dizemos que o atraso de $\mathbf{a}(D)$ é igual a $-\infty$.

2. Se existir um inteiro r tal que $a_r \neq 0$ e $a_i = 0$, para $i > r$, dizemos que r é o **grau** de $\mathbf{a}(D)$, e denotamos isto por $r = \partial(\mathbf{a}(D))$. Se não existir um tal inteiro r, dizemos que o grau de $\mathbf{a}(D)$ é igual a $+\infty$.

Definição 2.13 *Um **polinômio sobre** $\mathcal{A}$ é uma sequência nula ou uma sequência não nula* $\mathbf{a}(D)$ *tal que* $del(\mathbf{a}(D)) \geq 0$ *e* $\partial(\mathbf{a}(D)) < +\infty$. *Vamos denotar o conjunto dos polinômios sobre* $\mathcal{A}$ *por* $\mathcal{A}[D]$.

O conjunto das *séries de Laurent* sobre $\mathcal{A}$, denotado por $\mathcal{A}((D))$, é o conjunto das sequências sobre $\mathcal{A}$ que têm apenas um número finito de termos não nulos com índice negativo, ou seja, sequências que têm atraso finito. Assim, temos que

$$\mathcal{A}((D)) = \left\{ \mathbf{a}(D) = \sum_{i=r}^{\infty} a_i D^i; \quad a_i \in \mathcal{A} \quad e \quad r \in \mathbb{Z} \right\}. \tag{2.4}$$

As operações de soma e produto em $\mathcal{A}$ podem ser estendidas para $\mathcal{A}((D))$. De fato,

$$\left(\sum_{i=r}^{\infty} a_i D^i \right) + \left(\sum_{i=s}^{\infty} b_i D^i \right) \triangleq \sum_{i=t}^{\infty} (a_i + b_i) D^i, \tag{2.5}$$

e

$$\left(\sum_{i=r}^{\infty} a_i D^i \right) \left(\sum_{i=s}^{\infty} b_i D^i \right) \triangleq \sum_{i=t}^{\infty} \left(\sum_{j} a_j b_{i-j} \right) D^i, \tag{2.6}$$

onde $t \leq \min\{r, s\}$ e a soma $\sum_j a_{i-j}b_j$ é realizada em $\mathcal{A}$. Uma vez que todo elemento de $\mathcal{A}((D))$ tem apenas um número finito de coeficientes não nulos com índice $i < 0$, todas as somas $\sum_j a_{i-j}b_j$ têm somente um número finito de parcelas não nulas. Então todos os coeficientes no lado direito da expressão 2.6 estão bem definidos.

Teorema 2.6 *$\mathcal{A}((D))$ com as operações acima é um anel comutativo com identidade.*

Teorema 2.7 *$\mathcal{A}[D]$ é um subanel de $\mathcal{A}((D))$.*

Teorema 2.8 *Se $\mathbf{a}(D) = \sum_{i=r}^{\infty} a_i D^i$, onde $a_r \neq 0$, então $\mathbf{a}(D)$ é inversível se, e somente se, a_r é inversível.*

Demonstração: Suponhamos que $\mathbf{a}(D) = \sum_{i=r}^{\infty} a_i D^i$ é inversível e seja $\mathbf{b}(D) = \sum_{i=s}^{\infty} b_i D^i$ o seu inverso, isto é, $\mathbf{a}(\mathbf{D})\,\mathbf{b}(D) = 1$. Então o termo independente de $\mathbf{a}(\mathbf{D})\,\mathbf{b}(D)$ é igual a 1 e os demais são nulos. De 2.6 segue que $a_r b_{-r} = 1$. Logo a_r é inversível.
Reciprocamente, suponhamos que a_r é inversível. Seja $\mathbf{b}(D) = \sum_{j=-r}^{\infty} b_j D^j$, onde os b_j são obtidos resolvendo-se o sistema infinito de equações lineares

$$\begin{aligned} a_r b_{-r} &= 1 \\ a_{r+1}b_{-r} + a_r b_{-r+1} &= 0 \\ &\vdots \\ \textstyle\sum_{j=0}^{s} a_{r+s-j}b_{-r+j} &= 0 \\ &\vdots \end{aligned} \tag{2.7}$$

É fácil ver que a sequência $\mathbf{b}(D)$ cujos coeficientes são dados por 2.7 é o inverso de $\mathbf{a}(D)$. ■

No que segue, estaremos interessados apenas em anéis comutativos com identidade. Assim, a partir deste momento, a menos que se diga o contrário, o termo anel significa anel comutativo com identidade.

2.2.3 Corpos

Definição 2.14 *Um **corpo** é um anel comutativo com identidade cujos elementos não nulos são todos inversíveis.*

Observação 2.7 *Em um corpo, os elementos não nulos formam um grupo com a operação produto. Se o corpo for finito então esse grupo será necessariamente cíclico.*

Observação 2.8 *Um corpo não possui divisor de zero.*

Observação 2.9 *Como consequência imediata do teorema 2.8 temos que o anel $\mathcal{A}((D))$ é um corpo se, e somente se, $\mathcal{A}$ é um corpo.*

Teorema 2.9 *O anel $\mathbb{Z}_q$ é um corpo se, e somente se, q é um número primo.*

Demonstração: Suponhamos que q é um número primo e seja $a \in \mathbb{Z}_q$, $a \neq 0$. Como q é primo e $1 \leq a \leq q-1$, segue que $mdc\{a, q\} = 1$. Então existem $r, s \in \mathbb{Z}$ tais que $ra + sq = 1$. Como $sq = 0$ em $\mathbb{Z}_q$, segue que $ra = 1$ em $\mathbb{Z}_q$. Então a é inversível. Portanto $\mathbb{Z}_q$ é um corpo. Reciprocamente, suponhamos que $\mathbb{Z}_q$ é um corpo e seja $1 \leq a \leq q-1$. Como $\mathbb{Z}_q$ não possui divisor de zero, segue que $mdc\{a, q\} = 1$. Portanto q é um número primo. ■

Observação 2.10 *É fácil mostrar que quaisquer dois corpos finitos de mesma ordem são isomorfos.*

Para cada número primo p, existe um corpo com p elementos, pois, sendo p primo, então $\mathbb{Z}_p$ é corpo. Mas também existem corpos cujo número de elementos não é um número primo, como afirma o teorema seguinte, cuja demonstração pode ser encontrada em [11].

Teorema 2.10

i) O número de elementos de qualquer corpo finito é potência de algum número primo;
ii) Para cada número primo p e cada número inteiro positivo n, existe um corpo com p^n elementos.

Um corpo com q elementos é chamado **corpo de Galois de ordem** q e é denotado por $GF(q)$. Na tabela 2.2 mostramos as operações de soma e produto de $GF(4)$, onde supomos que $GF(4) = \{0, 1, \alpha, \beta\}$.

$+$	0	1	α	β
0	0	1	α	β
1	1	0	β	α
α	α	β	0	1
β	β	α	1	0

$\cdot$	0	1	α	β
0	0	0	0	0
1	0	1	α	β
α	0	α	β	1
β	0	β	1	α

Tabela 2.2: Operações de soma e produto em GF(4)

2.2.4 Espaços Vetoriais

Seja $(\mathcal{V},+)$ um grupo abeliano e seja $\mathbb{F}$ um corpo. Suponhamos que esteja definida uma operação, que chamaremos de **multiplicação por escalar**, entre os elementos λ de $\mathbb{F}$ e os elementos $\mathbf{v}$ de $\mathcal{V}$ tal que a cada par ordenado $(\lambda, \mathbf{v}) \in \mathbb{F}\times\mathcal{V}$ esteja associado um elemento de $\mathcal{V}$, que denotaremos por $\lambda\mathbf{v}$. Em outras palavras, estamos supondo que esteja definida uma função do tipo

$$\begin{aligned} \mathbb{F} \times \mathcal{V} &\rightarrow \mathcal{V} \\ (\lambda, \mathbf{v}) &\mapsto \lambda\mathbf{v} \end{aligned}$$

Definição 2.15 *Sob as condições acima, dizemos que $\mathcal{V}$ é um **espaço vetorial sobre** $\mathbb{F}$ se as propriedades seguintes são satisfeitas para quaisquer $\lambda, \mu \in \mathbb{F}$ e $\mathbf{u}, \mathbf{v} \in \mathcal{V}$:*

i) $\lambda(\mathbf{u} + \mathbf{v}) = \lambda\mathbf{u} + \lambda\mathbf{v}$
ii) $(\lambda + \mu)\mathbf{v} = \lambda\mathbf{v} + \mu\mathbf{v}$
iii) $(\lambda\mu)\mathbf{v} = \lambda(\mu\mathbf{v})$
iv) $1\mathbf{v} = \mathbf{v}$, *onde* 1 *é o elemento identidade de* $\mathbb{F}$.

Se $\mathcal{V}$ é um espaço vetorial sobre $\mathbb{F}$, os elementos de $\mathcal{V}$ serão chamados **vetores** e os de $\mathbb{F}$ **escalares**.

Definição 2.16 *Sejam $\mathcal{V}$ um espaço vetorial sobre $\mathbb{F}$ e $\mathcal{W}$ um subconjunto não vazio de $\mathcal{V}$. Dizemos que $\mathcal{W}$ é um subespaço vetorial de $\mathcal{V}$ se:*

i) $\mathcal{W}$ *é um subgrupo de* $\mathcal{V}$

ii) $\lambda \mathbf{w} \in \mathcal{W}$, $\forall \lambda \in \mathbb{F}$ *e* $\forall \mathbf{w} \in \mathcal{W}$.

Sejam $\mathcal{V}$ um espaço vetorial sobre $\mathbb{F}$ e $\mathbf{v}_1, \ldots, \mathbf{v}_n$ elementos de $\mathcal{V}$. O vetor

$$\lambda_1 \mathbf{v}_1 + \cdots + \lambda_n \mathbf{v}_n$$

é uma **combinação linear** dos vetores $\mathbf{v}_1, \ldots, \mathbf{v}_n$. Seja S um subconjunto de $\mathcal{V}$ e denotemos por $\langle \mathrm{S} \rangle$ o conjunto de todas as combinações lineares de vetores de S, isto é,

$$\langle \mathrm{S} \rangle = \{ \mathbf{v} \in \mathcal{V} \,|\, \mathbf{v} = \lambda_1 \mathbf{v}_1 + \cdots + \lambda_n \mathbf{v}_n, \quad \lambda_i \in \mathbb{F},\, \mathbf{v}_i \in \mathcal{V} \quad e \quad n \in \mathbb{Z} \}.$$

Então $\langle \mathrm{S} \rangle$ é um subespaço de $\mathcal{V}$, chamado **subespaço gerado por** S. Quando S é um conjunto finito, digamos $\mathrm{S} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, também usamos a notação $\langle \mathbf{v}_1, \ldots, \mathbf{v}_n \rangle$ em vez de $\langle \mathrm{S} \rangle$. Se um conjunto de vetores S gera $\mathcal{V}$, isto é, $\langle \mathrm{S} \rangle = \mathcal{V}$ então todo vetor de $\mathcal{V}$ pode ser escrito como uma combinação linear de vetores de S.

Definição 2.17 *Se* $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathcal{V}$, *dizemos que o subconjunto* $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ *é* ***linearmente independente*** *(LI) se a única combinação linear de* $\mathbf{v}_1, \ldots, \mathbf{v}_n$ *que resulta no vetor nulo é aquela cujos escalares são todos nulos, ou seja,*

$$\lambda_1 \mathbf{v}_1 + \cdots + \lambda_n \mathbf{v}_n = \mathbf{0} \Longrightarrow \lambda_1 = \cdots = \lambda_n = 0.$$

Caso contrário, dizemos que $\mathbf{v}_1, \ldots, \mathbf{v}_n$ *são* ***linearmente dependentes*** *(LD).*
Um subconjunto qualquer não vazio S é dito LI se todo subconjunto finito não vazio de S é LI. Caso contrário, S é dito LD.

Definição 2.18 *Seja* $\mathcal{B}$ *um subconjunto de* $\mathcal{V}$. *Dizemos que* $\mathcal{B}$ *é uma* ***base*** *de* $\mathcal{V}$ *se* $\mathcal{B}$ *é LI e gera* $\mathcal{V}$.

Observação 2.11 *Todo espaço vetorial não nulo possui base. Além disso, quaisquer duas bases de um espaço vetorial têm a mesma cardinalidade. (cf.[26])*

Definição 2.19 *Um espaço vetorial $\mathcal{V}$ sobre um corpo $\mathbb{F}$ é dito de dimensão finita se ele é o espaço nulo ou possui base finita. Se o espaço é nulo dizemos que a sua dimensão é zero. Se ele possui uma base com n vetores, dizemos que a sua dimensão é n e denotamos por $\dim_{\mathbb{F}} \mathcal{V} = n$, ou simplesmente $\dim \mathcal{V} = n$, se não houver necessidade de explicitar o corpo $\mathbb{F}$. Se um espaço tem uma base que não é finita, dizemos que a sua dimensão é infinita.*

Estamos interessados apenas em espaços vetoriais de dimensão finita. Assim, a partir deste momento, quando não houver referência explícita à dimensão, estaremos supondo que ela é finita.

Teorema 2.11 *[26] Sejam $\mathcal{V}$ um espaço vetorial sobre um corpo $\mathbb{F}$ e $\mathcal{B} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ um subconjunto de $\mathcal{V}$. As afirmações seguintes são equivalentes:*

i) $\mathcal{B}$ é uma base de $\mathcal{V}$
ii) Cada $\mathbf{v} \in \mathcal{V}$ pode ser escrito de modo único como combinação linear de $\mathbf{v}_1, \ldots, \mathbf{v}_n$
iii) $\mathcal{B}$ gera $\mathcal{V}$ mas nenhum subconjunto próprio de $\mathcal{B}$ gera $\mathcal{V}$
iv) $\mathcal{B}$ é LI e qualquer conjunto que contém $\mathcal{B}$ propriamente é LD.

Definição 2.20 *Sejam $\mathcal{V}_1$ e $\mathcal{V}_2$ espaços vetoriais sobre um mesmo corpo $\mathbb{F}$. Uma função $\varphi : \mathcal{V}_1 \to \mathcal{V}_2$ tal que*

$$\varphi(\lambda \mathbf{u} + \mathbf{v}) = \lambda \varphi(\mathbf{u}) + \varphi(\mathbf{v}), \quad \forall \mathbf{u}, \mathbf{v} \in \mathcal{V}_1 \quad e \quad \forall \lambda \in \mathbb{F}$$

*é chamada **transformação linear**. Se, além disso, φ for uma bijeção dizemos que φ é um **isomorfismo**. Neste caso os espaços $\mathcal{V}_1$ e $\mathcal{V}_2$ são isomorfos e escrevemos $\mathcal{V}_1 \simeq \mathcal{V}_2$.*

Teorema 2.12 *Sejam $\mathcal{V}_1$ e $\mathcal{V}_2$ espaços vetoriais sobre um mesmo corpo $\mathbb{F}$. Temos que $\dim \mathcal{V}_1 = \dim \mathcal{V}_2$ se, e somente se, $\mathcal{V}_1 \simeq \mathcal{V}_2$.*

Demonstração: Suponhamos que $\dim \mathcal{V}_1 = \dim \mathcal{V}_2$ e seja $n = \dim \mathcal{V}_1 = \dim \mathcal{V}_2$. Sejam $\mathcal{B}_1 = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ e $\mathcal{B}_2 = \{\mathbf{w}_1, \ldots, \mathbf{w}_n\}$ bases de $\mathcal{V}_1$ e $\mathcal{V}_2$, respectivamente, e $\varphi : \mathcal{V}_1 \to \mathcal{V}_2$, definida por

$$\varphi(\lambda_1 \mathbf{v}_1 + \cdots + \lambda_n \mathbf{v}_n) = \lambda_1 \mathbf{w}_1 + \cdots + \lambda_n \mathbf{w}_n.$$

É fácil ver que φ é um isomorfismo.

Reciprocamente, suponhamos que $\mathcal{V}_1 \simeq \mathcal{V}_2$ e seja $\varphi : \mathcal{V}_1 \to \mathcal{V}_2$ um isomorfismo. Se $\mathcal{B}_1 = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ é uma base de $\mathcal{V}_1$, é fácil ver que $\mathcal{B}_2 = \{\varphi(\mathbf{v}_1), \ldots, \varphi(\mathbf{v}_n)\}$ é uma base de $\mathcal{V}_2$. Então $\dim \mathcal{V}_2 = \dim \mathcal{V}_1$. ■

Seja $\mathbf{M}$ uma matriz $m \times n$ sobre um corpo $\mathbb{F}$. As linhas de $\mathbf{M}$ geram um subespaço de $\mathbb{F}^n$, chamado **espaço linha** de $\mathbf{M}$, e as colunas geram um subespaço de $\mathbb{F}^m$, chamado **espaço coluna** de $\mathbf{M}$. Quando $m \neq n$ estes subespaços estão em espaços diferentes mas as dimensões desses subespaços são iguais (cf. [26]). A dimensão desses subespaços é o **posto** de $\mathbf{M}$.

Matriz de uma Transformação Linear

Sejam $\mathcal{V}_1$ e $\mathcal{V}_2$ espaços vetoriais de dimensões m e n, respectivamente, sobre um corpo $\mathbb{F}$ e seja $\varphi : \mathcal{V}_1 \longrightarrow \mathcal{V}_2$ uma transformação linear. Sejam

$$\mathcal{B}_1 = \{\mathbf{u}_1, \ldots, \mathbf{u}_m\} \quad e \quad \mathcal{B}_2 = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$$

bases de $\mathcal{V}_1$ e $\mathcal{V}_2$, respectivamente.

Para cada i, $\quad i = 1, \ldots, m$ existem escalares a_{ij}, $\quad j = 1, \ldots, n$, tais que

$$\varphi(\mathbf{u}_i) = a_{i1}\mathbf{v}_1 + \cdots + a_{in}\mathbf{v}_n.$$

Vamos considerar a matriz $\mathbf{M}$ $\quad m \times n$, cujos elementos são os escalares a_{ij}, isto é,

$$\mathbf{M} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}$$

A matriz $\mathbf{M}$ é chamada **matriz de φ em relação às bases** $\mathcal{B}_1$ e $\mathcal{B}_2$ e é denotada por $[\varphi]_{\mathcal{B}_1}^{\mathcal{B}_2}$.

Se $\mathbf{u}$ é um vetor qualquer de $\mathcal{V}_1$ então $\mathbf{u}$ é escrito de modo único como combinação linear dos vetores de $\mathcal{B}_1$, isto é, existem escalares únicos $\lambda_1, \ldots, \lambda_m$ tais que

$$\mathbf{u} = \lambda_1\mathbf{u}_1 + \cdots + \lambda_m\mathbf{u}_m.$$

Estes escalares são as **coordenadas** de $\mathbf{u}$ em relação à base $\mathcal{B}_1$, e escrevemos $\mathbf{u} = (\lambda_1, \ldots, \lambda_m)$. Como φ é linear, temos que

$$\varphi(\mathbf{u}) = \lambda_1\varphi(\mathbf{u}_1) + \cdots + \lambda_m\varphi(\mathbf{u}_m) = \sum_{j=1}^{n}\sum_{i=1}^{m}\lambda_i a_{ij}\mathbf{v}_j.$$

Então

$$\varphi(\mathbf{u}) = (\sum_{i=1}^{m}\lambda_i a_{i1}, \ldots, \sum_{i=1}^{m}\lambda_i a_{in}).$$

Podemos observar que as coordenadas de $\varphi(\mathbf{u})$ podem ser obtidas pelo produto matricial $\mathbf{uM}$, considerando o vetor $\mathbf{u}$ como a matriz $1 \times m$ $\quad [\lambda_1 \cdots \lambda_m]$. Assim, para determinarmos a imagem de um vetor de $\mathcal{V}_1$, podemos usar φ ou a sua matriz $\mathbf{M}$.

O posto de $\mathbf{M}$ é igual a dimensão da imagem de φ, pois a imagem de φ é o espaço linha de $\mathbf{M}$. Temos também que φ é injetiva se, e somente se, o posto de $\mathbf{M}$ é igual a m.

Mostramos que a cada transformação linear de $\mathcal{V}_1$ em $\mathcal{V}_2$ podemos associar uma matriz $m \times n$. A recíproca também é verdadeira, isto é, a cada matriz $m \times n$ está associada uma transformação linear de $\mathcal{V}_1$ em $\mathcal{V}_2$.

2.2.5 Módulos

O conceito de **módulo** generaliza o de espaço vetorial. Em vez de exigirmos que os escalares pertençam a um corpo, como ocorre com espaço vetorial, consideramos escalares num anel. Algumas propriedades de espaços vetoriais que envolvem multiplicação por escalar não são válidas, de um modo geral, em módulos pois a validade dessas propriedades depende, quase sempre, de divisão por escalar. Isto nem sempre pode ser feito num anel. Por outro lado, sendo o conceito de módulo uma generalização do de espaço vetorial, várias propriedades de espaços vetoriais podem ser aplicadas em módulos. É o que faremos neste trabalho, ao considerarmos **códigos** definidos não apenas sobre corpos, mas também sobre anéis.

Como já mencionado, neste trabalho faremos uso apenas de anéis comutativos com identidade. Assim, na definição de módulo que daremos a seguir, estaremos supondo que o anel é comutativo com identidade.

Seja $(\mathcal{M},+)$ um grupo abeliano e seja $\mathcal{A}$ um anel. Suponhamos que esteja definida uma operação, que chamaremos de **multiplicação por escalar**, entre os elementos de $\mathcal{A}$ e os elementos de $\mathcal{M}$ tal que a cada par ordenado $(\lambda, \mathbf{v}) \in \mathcal{A}\times\mathcal{M}$ esteja associado um elemento de $\mathcal{M}$, que denotaremos por $\lambda\mathbf{v}$. Em outras palavras, estamos supondo que esteja definida uma função do tipo

$$\begin{aligned} \mathcal{A} \times \mathcal{M} &\rightarrow \mathcal{M} \\ (\lambda, \mathbf{v}) &\mapsto \lambda\mathbf{v} \end{aligned}$$

Definição 2.21 *Sob as condições acima, dizemos que $\mathcal{M}$ é um **módulo sobre** $\mathcal{A}$, ou um $\mathcal{A}$-**módulo** se as propriedades seguintes são satisfeitas para quaisquer $\lambda, \mu \in \mathcal{A}$ e $\mathbf{u}, \mathbf{v} \in \mathcal{M}$:*

i) $\lambda(\mathbf{u} + \mathbf{v}) = \lambda\mathbf{u} + \lambda\mathbf{v}$
ii) $(\lambda + \mu)\mathbf{v} = \lambda\mathbf{v} + \mu\mathbf{v}$
iii) $(\lambda\mu)\mathbf{v} = \lambda(\mu\mathbf{v})$
iv) $1\mathbf{v} = \mathbf{v}$, *onde* 1 *é o elemento identidade de* $\mathcal{A}$.

Exemplo 2.5 *Todo anel $\mathcal{A}$ é um $\mathcal{A}$-módulo. Em particular, os anéis $\mathbb{Z}_q$ são $\mathbb{Z}_q$-módulos.*

Exemplo 2.6 *Todo grupo abeliano é um $\mathbb{Z}$-módulo, onde $\mathbb{Z}$ é o anel dos números inteiros.*

Observação 2.12 *As definições de submódulo e submódulo gerado por um conjunto são análogas àquelas dadas em espaços vetoriais, para subespaços. As definições de dependência e independência linear de conjuntos e de base também são idênticas.*

Definição 2.22 *Um módulo é dito **finitamente gerado** se ele pode ser gerado por um conjunto finito.*

Se um espaço vetorial pode ser gerado por um conjunto finito, isto é, se o espaço tem dimensão finita então todo subespaço também tem. Esta propriedade não é válida, em geral para módulos pois um módulo pode ser finitamente gerado e ter um submódulo que não é.

Em um espaço vetorial, qualquer conjunto formado por um único elemento não nulo é *LI*. No caso de módulos isso nem sempre é verdade. Por exemplo, consideremos $\mathbb{Z}_q$ como

um $\mathbb{Z}$-módulo. Neste caso, para qualquer $\mathbf{a} \in \mathbb{Z}_q$ temos que $q\mathbf{a} = 0$. Logo, $\{\mathbf{a}\}$ é LD. Daí concluimos que $\mathbb{Z}_q$, como um $\mathbb{Z}$-módulo, não possui subconjunto LI e, consequentemente, não possui base. Vimos que isto não ocorre em espaços vetoriais.

Em um espaço vetorial, um subconjunto S com mais de um elemento é LD se, e somente se, algum vetor de S é uma combinação linear de outros elementos de S. Isto não é verdade em módulos. De fato, se considerarmos o subconjunto $S = \{(2,0),(3,0)\}$ no $\mathbb{Z}$-módulo $\mathbb{Z}^2$, temos que S é LD e $(2,0)$ não é múltiplo de $(3,0)$. Isto ocorre porque nem sempre podemos efetuar divisão por escalar num anel.

Definição 2.23 *Um módulo é dito* ***livre*** *se ele possui uma base.*

Em um espaço vetorial, quaisquer duas bases possuem o mesmo número de elementos. Em um módulo livre, se o anel não é comutativo, pode haver bases com cardinalidades diferentes. Mas se $\mathcal{A}$ é um anel comutativo com identidade então quaisquer duas bases de um $\mathcal{A}$-módulo livre têm mesma cardinalidade (cf. [26]). Isto nos permite definir um conceito análogo ao de dimensão de um espaço vetorial.

Definição 2.24 *Seja $\mathcal{M}$ um $\mathcal{A}$-módulo livre. Definimos o* ***posto*** *de $\mathcal{M}$ como sendo a cardinalidade de qualquer base de $\mathcal{M}$.*

Exemplo 2.7 *Sejam q e n inteiros positivos e*

$$\mathbb{Z}_q^n = \{(x_1, \ldots, x_n) \,|x_i \in Z_q,\ i = 1, \ldots, n\}.$$

É fácil ver que $\mathbb{Z}_q^n$ é um $\mathbb{Z}_q$-módulo livre de posto n, pois o conjunto de n-uplas

$$\mathcal{B} = \{(1,0,\ldots,0),\ (0,1,0,\ldots,0),\ldots,(0,\ldots,0,1)\}$$

é uma base de $\mathbb{Z}_q^n$. De fato,

i) Se $\mathbf{v} = (x_1, x_2, \ldots, x_n) \in \mathbb{Z}_q^n$ então

$$\mathbf{v} = x_1(1,0,\ldots,0) + x_2(0,1,0,\ldots,0) + \cdots + x_n(0,\ldots,0,1).$$

Logo, $\mathcal{B}$ gera $\mathbb{Z}_q^n$.

ii) Se $x_1(1,0,\ldots,0)+x_2(0,1,0,\ldots,0)+\cdots+x_n(0,\ldots,0,1)=0$ então $(x_1,x_2,\ldots,x_n)=0$, ou seja, $x_1=x_2=\cdots=x_n=0$. Então $\mathcal{B}$ é LI.

Portanto $\mathcal{B}$ é uma base para o $\mathbb{Z}_q$-módulo $\mathbb{Z}_q^n$.

Definição 2.25 *Sejam $\mathcal{M}$ e $\mathcal{N}$ $\mathcal{A}$-módulos e $\varphi:\mathcal{M}\to\mathcal{N}$ uma função. Dizemos que φ é um* ***homomorfismo*** *se $\varphi(\lambda\mathbf{a}+\mu\mathbf{b})=\lambda\varphi(\mathbf{a})+\mu\varphi(\mathbf{b})$, $\forall\lambda,\mu\in\mathcal{A}$, $\forall\mathbf{a},\mathbf{b}\in\mathcal{M}$. Se, além disso, φ é uma bijeção dizemos que φ é um* ***isomorfismo.***

Teorema 2.13 *[26] Dois módulos livres são isomorfos se, e somente se, eles têm postos iguais.*

Observação 2.13 *Sejam $\mathcal{M}$ e $\mathcal{N}$ $\mathcal{A}$-módulos livres de postos m e n, respectivamente, e seja $\varphi:\mathcal{M}\to\mathcal{N}$ um homomorfismo. De modo análogo ao caso de espaços vetoriais, podemos associar a φ uma matriz $m\times n$ e, reciprocamente, a cada matriz $m\times n$ podemos associar um homomorfismo de $\mathcal{M}$ em $\mathcal{N}$.*

2.3 Introdução à Teoria da Codificação

2.3.1 Códigos de Bloco

Vamos considerar um codificador, munido de um código de bloco $\mathcal{C}$, que recebe mensagens cujos símbolos pertencem a um conjunto $\mathcal{A}$. Cada **sequência de informação** $\mathbf{a}=(a_1,\ldots,a_k)\in\mathcal{A}^k$ que entra no codificador, é codificada e transformada em uma n-upla $\mathbf{c}=(c_1,\ldots,c_n)\in\mathcal{A}^n$, $n\geq k$, chamada **palavra código**. Assim, o codificador associa uma palavra código a cada sequência de informação, ou seja, o codificador é uma função $\mathbf{C}:\mathcal{A}^k\longrightarrow\mathcal{A}^n$ e a imagem dessa função, isto é, o conjunto das palavras código, é o código $\mathcal{C}$. Neste caso dizemos que $\mathcal{C}$ é um código sobre $\mathcal{A}$ de comprimento n e **taxa de codificação** $R\triangleq\frac{k}{n}$. Denotando por $|\mathcal{A}|$ o número de elementos de $\mathcal{A}$, segue que o código $\mathcal{C}$ possui $|\mathcal{A}|^k$ palavras código.

Definição 2.26 *A* ***distância de Hamming*** *(d_H) entre dois vetores (palavras) de $\mathcal{A}^n$ é definida como sendo o número de posições em que eles diferem. O* ***peso de Hamming*** *(w_H) de um vetor é o número de coordenadas não nulas desse vetor.*

Teorema 2.14 *A distância de Hamming é uma métrica, isto é, ela satisfaz as propriedades seguintes:*

i) $d_H(\mathbf{x}, \mathbf{y}) \geq 0$

ii) $d_H(\mathbf{x}, \mathbf{y}) = 0 \Longleftrightarrow \mathbf{x} = \mathbf{y}$

iii) $d_H(\mathbf{x}, \mathbf{y}) = d_H(\mathbf{y}, \mathbf{x})$

iv) $d_H(\mathbf{x}, \mathbf{y}) \leq d_H(\mathbf{x}, \mathbf{z}) + d_H(\mathbf{z}, \mathbf{y})$.

Demonstração: As propriedades *(i)* **a** *(iii)* são óbvias. Vamos demonstrar *(iv)*.
Sejam $\mathbf{x} = (x_1, \ldots, x_n)$, $\mathbf{y} = (y_1, \ldots, y_n)$ e $\mathbf{z} = (z_1, \ldots, z_n)$, e definamos

$$\mathbf{I}_{xy} \triangleq \{i \in \{1, \ldots, n\} \, | x_i = y_i\}$$

$$\mathbf{I}_{xz} \triangleq \{i \in \{1, \ldots, n\} \, | x_i = z_i\}$$

$$\mathbf{I}_{zy} \triangleq \{i \in \{1, \ldots, n\} \, | z_i = y_i\}.$$

Temos que

$$d_H(\mathbf{x}, \mathbf{y}) = n - |\mathbf{I}_{xy}|$$

$$d_H(\mathbf{x}, \mathbf{z}) = n - |\mathbf{I}_{xz}|$$

$$d_H(\mathbf{z}, \mathbf{y}) = n - |\mathbf{I}_{zy}|,$$

onde $|\mathbf{X}|$ denota a cardinalidade do conjunto $\mathbf{X}$. Daí segue que

$$d_H(\mathbf{x}, \mathbf{z}) + d_H(\mathbf{z}, \mathbf{y}) = 2n - (|\mathbf{I}_{xz}| + |\mathbf{I}_{zy}|).$$

Mas

$$|\mathbf{I}_{xz}| + |\mathbf{I}_{zy}| = |\mathbf{I}_{xz} \cup \mathbf{I}_{zy}| + |\mathbf{I}_{xz} \cap \mathbf{I}_{zy}|.$$

Então

$$d_H(\mathbf{x}, \mathbf{z}) + d_H(\mathbf{z}, \mathbf{y}) = 2n - |\mathbf{I}_{xz} \cup \mathbf{I}_{zy}| - |\mathbf{I}_{xz} \cap \mathbf{I}_{zy}|.$$

Como $|\mathbf{I}_{xz} \cup \mathbf{I}_{zy}| \leq n$, temos que

$$d_H(\mathbf{x}, \mathbf{z}) + d_H(\mathbf{z}, \mathbf{y}) \geq n - |\mathbf{I}_{xz} \cap \mathbf{I}_{zy}|. \tag{2.8}$$

Se $i \in \mathbf{I}_{xz} \cap \mathbf{I}_{zy}$ então $x_i = z_i$ e $z_i = y_i$. Daí segue que $x_i = y_i$, ou seja, $i \in \mathbf{I}_{xy}$. Então $\mathbf{I}_{xz} \cap \mathbf{I}_{zy} \subseteq \mathbf{I}_{xy}$ e, consequentemente, $|\mathbf{I}_{xz} \cap \mathbf{I}_{zy}| \leq |\mathbf{I}_{xy}|$. Então

$$n - |\mathbf{I}_{xz} \cap \mathbf{I}_{zy}| \geq n - |\mathbf{I}_{xy}| = d_H(\mathbf{x}, \mathbf{y}). \tag{2.9}$$

De 2.8 e 2.9 concluimos que $d_H(\mathbf{x}, \mathbf{y}) \leq d_H(\mathbf{x}, \mathbf{z}) + d_H(\mathbf{z}, \mathbf{y})$. ■

2.3.2 Códigos de Bloco Lineares

Definição 2.27 *Se $\mathcal{A}$ é um anel comutativo com identidade então $\mathcal{A}^k$ e $\mathcal{A}^n$ são $\mathcal{A}$-módulos livres de postos k e n, respectivamente. Se o codificador $\mathbf{C} : \mathcal{A}^k \to \mathcal{A}^n$ é um homomorfismo injetivo, dizemos que $\mathbf{C}$ é um* ***codificador linear*** *sobre $\mathcal{A}$ e que $\mathcal{C}$ é um* ***código de bloco linear*** *sobre $\mathcal{A}$, que denotaremos por (n, k). Os parâmetros n e k são, respectivamente, o* ***comprimento*** *e a* ***dimensão*** *do código $\mathcal{C}$, e $R \triangleq \frac{k}{n}$ é a* ***taxa de codificação*** *de $\mathcal{C}$.*

Observação 2.14 *Se $\mathbf{C} : \mathcal{A}^k \to \mathcal{A}^n$ é um homomorfismo injetivo então a imagem de $\mathbf{C}$ é um submódulo livre de $\mathcal{A}^n$, isomorfo a $\mathcal{A}^k$. Então um código linear sobre um anel $\mathcal{A}$, de comprimento n e dimensão k, é um submódulo livre de $\mathcal{A}^n$ de posto k. Reciprocamente, se $\mathcal{M}$ é um submódulo livre de $\mathcal{A}^n$, de posto k, então, pelo teorema 2.13, $\mathcal{M} \simeq \mathcal{A}^k$. Então existe um homomorfismo injetivo $\mathbf{C} : \mathcal{A}^k \to \mathcal{A}^n$, cuja imagem é $\mathcal{M}$. Portanto todo submódulo livre de $\mathcal{A}^n$, de posto k, é um código linear sobre $\mathcal{A}$, de comprimento n e dimensão k.*
Se $\mathcal{A} = \mathbb{F}$ é um corpo, dizemos que $\mathcal{C}$ é um código sobre o corpo $\mathbb{F}$. Neste caso $\mathcal{C}$ é um subespaço vetorial de $\mathbb{F}^n$, de dimensão k. Reciprocamente, todo subespaço vetorial de $\mathbb{F}^n$, de dimensão k, é um código linear sobre $\mathbb{F}$.

Sendo o conjunto de palavras código de um código de bloco linear um submódulo de $\mathcal{A}^n$, então a sequência toda nula de $\mathcal{A}^n$ é uma palavra código e a soma de palavras códigos

resulta numa palavra código. Um atrativo dos códigos lineares é a estrutura algébrica que eles possuem, dotando-os de propriedades que facilitam a implementação dos mesmos. Estaremos mais interessados em códigos de bloco lineares sobre os anéis $\mathbb{Z}_q$, onde q é uma potência de 2. Estes códigos serão chamados de **códigos de bloco multiníveis** sobre $\mathbb{Z}_q$.

Teorema 2.15 *Em um código linear a distância de Hamming mínima é igual ao peso de Hamming mínimo.*

Demonstração: Seja $\mathcal{C}$ um código linear. Seja $d_{H_{\min}}$ a distância de Hamming mínima e seja $w_{H_{\min}}$ o peso de Hamming mínimo.
Sejam $\mathbf{u}, \mathbf{v} \in \mathcal{C}$. Temos que $\mathrm{d}_H(\mathbf{u}, \mathbf{v}) = w_H(\mathbf{u} - \mathbf{v})$. Como $\mathcal{C}$ é um código linear, temos que $\mathbf{u} - \mathbf{v} \in \mathcal{C}$. Daí segue que a distância de Hamming entre quaisquer duas palavras código é igual ao peso de Hamming de alguma palavra código. Então $d_{H_{\min}} \geq w_{H_{\min}}$. Por outro lado, se $\mathbf{v}$ é uma palavra código de peso mínimo, então

$$w_{H_{\min}} = w_H(\mathbf{v}) = w_H(\mathbf{v} - \mathbf{0}) = d_H(\mathbf{v}, \mathbf{0}) \geq d_{H_{\min}}.$$

Portanto $d_{H_{\min}} = w_{H_{\min}}$. ■

Definição 2.28 *Um **código de distância invariante** (código DI) é um código $\mathcal{C}$ onde o número de palavras código que estão a uma determinada distância Euclidiana quadrada (D_E^2) de uma palavra código $\mathbf{x}$ não depende da escolha de $\mathbf{x}$.*

O **peso Euclidiano quadrado** (w_E^2) de uma palavra $\mathbf{x}$ é definido por

$$w_E^2(\mathbf{x}) \triangleq D_E^2(\mathbf{x}, \mathbf{0}).$$

O teorema 2.15 também é válido para D_E^2. Assim, $D_{E\,\min}^2 = w_{E_{\min}}^2$.

Teorema 2.16 *Os códigos lineares são códigos DI.*

Demonstração: Seja $\mathcal{C}$ um código linear e seja $\mathbf{x}$ uma palavra código de $\mathcal{C}$. Seja d um número real. Podemos supor $d \geq D^2_{E\,\min}(\mathcal{C})$. Sejam

$$H_x = \left\{ \mathbf{v} \in \mathcal{C} \,\middle|\, D_E^2(\mathbf{v}, \mathbf{x}) = d \right\} \quad e \quad H_0 = \left\{ \mathbf{v} \in \mathcal{C} \,\middle|\, D_E^2(\mathbf{v}, \mathbf{0}) = d \right\}.$$

Basta mostrarmos que H_x e H_0 têm mesma cardinalidade.
Como $\mathcal{C}$ é linear, temos que

$$H_0 = \left\{ \mathbf{v} \in \mathcal{C} \,\middle|\, D_E^2(\mathbf{v} + \mathbf{x}, \mathbf{x}) = d \right\}.$$

Daí segue que $H_x = \mathbf{x} + H_0$. Portanto H_x e H_0 têm mesma cardinalidade. ■

Em um código *DI* todas as palavras código têm o mesmo número de vizinhos. Em particular, para códigos lineares, o número de vizinhos mais próximos de cada palavra código é igual ao número de palavras código de peso mínimo.

Matriz Geradora e Matriz de Verificação de Paridade

Seja $\mathbf{C} : \mathcal{A}^k \longrightarrow \mathcal{A}^n$ um codificador linear. Como $\mathbf{C}$ é um homomorfismo injetivo, existe uma matriz $\mathbf{G}$ de ordem $k \times n$, de posto k, cujo espaço linha é a imagem de $\mathbf{C}$, isto é, as linhas de $\mathbf{G}$ geram o código $\mathcal{C}$. Esta matriz é chamada **matriz geradora** do código $\mathcal{C}$. Esta matriz não é única, pois ela depende das bases que escolhemos para $\mathcal{A}^k$ e $\mathcal{A}^n$. Dizemos que $\mathbf{G}$ está na **forma sistemática** se ela está na forma $\mathbf{G} = [\mathbf{I}_k \,|\mathbf{P}]$, onde $\mathbf{I}_k$ é a matriz identidade de ordem k e $\mathbf{P}$ é uma matriz de ordem $k \times (n-k)$.

Se $\mathbf{G} = [\mathbf{I}_k \,|\mathbf{P}]$ então a matriz $\mathbf{H} = [-\mathbf{P}^t \,|\mathbf{I}_{n-k}]$, onde $\mathbf{P}^t$ é a transposta de $\mathbf{P}$ e $\mathbf{I}_{n-k}$ é a matriz identidade de ordem $n-k$, é chamada **matriz de verificação de paridade** de $\mathcal{C}$. Temos que $\mathbf{H}$ é uma matriz de ordem $(n-k) \times n$ e $\mathbf{G}\mathbf{H}^t = \mathbf{0}$. Se $\mathbf{v}$ é um vetor de $\mathcal{A}^n$ então o vetor $\mathbf{s}(\mathbf{v}) \triangleq \mathbf{v}\mathbf{H}^t$ é chamado de **síndrome** de $\mathbf{v}$. Como $\mathbf{G}\mathbf{H}^t = \mathbf{0}$ temos que um vetor $\mathbf{v}$ de $\mathcal{A}^n$ é uma palavra código se, e somente se, a sua síndrome é nula, ou seja, $\mathbf{v}\mathbf{H}^t = \mathbf{0}$.

Observação 2.15 *Seja $\mathcal{C} : (n, k)$ um código multinível sobre $\mathbb{Z}_q$. Os vetores*

$$\mathbf{e}_1 = (1, 0, \ldots, 0), \mathbf{e}_2 = (0, 1, 0, \ldots, 0), \ldots, \mathbf{e}_n = (0, 0, \ldots, 0, 1)$$

são chamados ***padrões de erro*** *do código* $\mathcal{C}$. *Uma palavra* $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{Z}_q^n$ *contém um erro de magnitude* $\lambda \in \mathbb{Z}_q$, *numa coordenada* x_i, *se existe uma palavra código* $\mathbf{v} = (v_1, v_2, \ldots, v_n)$ *de* $\mathcal{C}$ *tal que* $\mathbf{x} = \mathbf{v} \oplus \lambda \mathbf{e}_i$. *Suponhamos que* $\mathcal{C}$ *esteja na forma sistemática e seja* $\mathbf{H}$ *a sua matriz de verificação de paridade. Então* $\mathbf{x}$ *contém um erro de magnitude* λ, *numa coordenada* x_i, *se, e somente se,* $\mathbf{s}(\mathbf{x}) = \lambda \mathbf{h}_i$, *onde* $\mathbf{h}_i$ *denota a* i*-ésima coluna de* $\mathbf{H}$. *De fato, suponhamos que* $\mathbf{x}$ *contém um erro de magnitude* λ, *na coordenada* x_i, *e seja* $\mathbf{v} = (v_1, v_2, \ldots, v_n)$ *uma palavra código de* $\mathcal{C}$ *tal que* $\mathbf{x} = \mathbf{v} \oplus \lambda \mathbf{e}_i$. *Então*

$$\mathbf{s}(\mathbf{x}) = \mathbf{s}(\mathbf{v} \oplus \lambda \mathbf{e}_i) = \mathbf{s}(\mathbf{v}) \oplus \lambda \mathbf{s}(\mathbf{e}_i) = \lambda \mathbf{h}_i$$

Reciprocamente, se $\mathbf{s}(\mathbf{x}) = \lambda \mathbf{h}_i$ *então* $\mathbf{s}(\mathbf{x}) = \lambda \mathbf{s}(\mathbf{e}_i)$. *Daí segue que* $\mathbf{s}(\mathbf{x} \ominus \lambda \mathbf{e}_i) = \mathbf{0}$, *ou seja,* $\mathbf{x} \ominus \lambda \mathbf{e}_i$ *é uma palavra código de* $\mathcal{C}$.

2.3.3 Códigos Convolucionais

Seja $\mathcal{A}$ um anel comutativo com identidade e seja $\mathbf{C} : \mathcal{A}((D))^k \longrightarrow \mathcal{A}((D))^n$, $n \geq k$, um homomorfismo injetivo. Dizemos que $\mathbf{C}$ é um **codificador convolucional** sobre $\mathcal{A}$ se $\mathbf{C}$ pode ser representado por uma matriz $\mathbf{G}(D)$ cujos elementos estão em $\mathcal{A}[D]$. Neste caso, o $\mathcal{A}((D))$-submódulo de $\mathcal{A}((D))^n$, gerado pelas linha de $\mathbf{G}(D)$, denotado por $\mathcal{C}$, é chamado **código convolucional** sobre $\mathcal{A}$. A matriz $\mathbf{G}(D)$ é a matriz geradora de $\mathcal{C}$, n e k são o comprimento e a dimensão, respectivamente, e $R = \frac{k}{n}$ é a taxa de codificação de $\mathcal{C}$. Uma matriz $\mathbf{H}(D)$ tal que $\mathbf{H}(D)\mathbf{G}(D)^t = \mathbf{O}$ é chamada matriz de verificação de paridade de $\mathcal{C}$. Se $\mathbf{G}(D) = [\mathbf{I}_k | \mathbf{P}(D)]$, onde $\mathbf{I}_k$ é a matriz identidade de ordem k, dizemos que $\mathbf{G}(D)$ está na forma sistemática. Se $\mathcal{A} = \mathbb{F}$ é um corpo então um código convolucional sobre $\mathbb{F}$ é um $\mathbb{F}((D))$-subespaço de $\mathbb{F}((D))^n$.

Na seção anterior, vimos que cada sequência de informação $\mathbf{a} = (a_1, \ldots, a_k)$ que entra no codificador de um código de bloco dá origem a uma palavra código que depende apenas dessa sequência $\mathbf{a}$. O mesmo não ocorre em códigos convolucionais. De fato, suponhamos que uma sequência (infinita) de dados de informação é dividida em blocos, cada um com k símbolos e a cada instante de tempo um desses blocos entra no decodificador. Denotemos o bloco que

entra no codificador no tempo discreto τ por $\mathbf{a}^{(\tau)} = \left(a_1^{(\tau)}, \ldots, a_k^{(\tau)}\right)$, e seja $\mathbf{c}^{(\tau)} = \mathbf{a}^{(\tau)}\mathbf{G}(D)$ a sequência codificada resultante. Como D é o elemento de atraso, segue que $\mathbf{c}^{(\tau)}$ não depende apenas de $\mathbf{a}^{(\tau)}$, mas depende também de alguns blocos anteriores a $\mathbf{a}^{(\tau)}$. Isto significa dizer que um codificador convolucional tem **memória**. Os valores armazenados nos elementos de memória determinam os **estados** do codificador. O número de estados de um codificador convolucional sobre um anel $\mathcal{A}$ depende do número de elementos de memória, do número de elementos do anel $\mathcal{A}$ e da matriz geradora $\mathbf{G}(D)$. Um atrativo para se usar códigos convolucionais é a disponibilidade de um algoritmo eficiente para a decodificação, como o algoritmo de Viterbi.

Estamos mais interessados em códigos convolucionais sobre os anéis $\mathbb{Z}_q$, onde q é uma potência de 2. Estes códigos são chamados de **códigos convolucionais multiníveis** sobre $\mathbb{Z}_q$.

2.4 Conclusão

Neste capítulo apresentamos alguns conceitos e resultados básicos que usaremos nos capítulos seguintes. Apresentamos os fundamentos algébricos, as definições de códigos multiníveis e alguns resultados.

Capítulo 3

Modulações PSK Codificadas com Códigos Multiníveis

3.1 Introdução

Um esquema de modulação codificada e um esquema de referência (não codificado) transmitem informação a uma mesma taxa, se o número médio de bits de informação em cada símbolo codificado é igual ao número de bits em cada símbolo do esquema de referência. Em outras palavras, o código usado no esquema codificado deve ser escolhido de acordo com esses dois esquemas de modulação, pois a taxa de codificação do código é quem determina o número médio de bits de informação em cada símbolo codificado.

Numa modulação $2^m - PSK$ cada símbolo carrega m bits. Portanto, se o esquema de referência é um $2^m - PSK$ e o codificado é um $2^r - PSK$, $r > m$, então, para que esses dois esquemas tenham as mesmas taxas de transmissão de informação, cada símbolo codificado deve ter em média $r - m$ bits de redundância. Neste caso, devemos usar um código de taxa $R = \frac{m}{r}$ pois assim, se a fonte binária emite m bits a cada intervalo de tempo T, então cada símbolo codificado carrega em média m bits de informação. Geralmente, em esquema de modulação codificada $q - PSK$, quando o esquema de referência é $2^m - PSK$ e desejamos que os esquemas codificado e de referência tenham as mesmas taxas de transmissão

de informação, consideramos a modulação $2^{m+1} - PSK$ para o esquema codificado e usamos um código de taxa $R = \frac{m}{m+1}$.

Ungerboeck [28] e Sayegh [27] utilizaram códigos binários em seus esquemas de modulação, enquanto que Baldini *et al.* [5] apresentaram esquemas de modulação codificada utilizando códigos multiníveis sobre os anéis $\mathbb{Z}_q$. Este capítulo tem como objetivo descrever os esquemas apresentados em [1, 3, 4]. Vamos considerar modulações $q-PSK$ codificadas com códigos sobre os anéis $\mathbb{Z}_q$. Como veremos a seguir, os códigos mais adequados para serem usados em modulações $q-PSK$ são os códigos multiníveis sobre os anéis $\mathbb{Z}_q$.

Na próxima seção faremos algumas considerações sobre os espaços PSK, onde destacaremos a identificação que pode ser feita entre o conjunto dos pontos de um espaço $q-PSK$ e o anel $\mathbb{Z}_q$. Na seção 3.3 consideraremos esquemas de modulações PSK codificadas, utilizando códigos de bloco multiníveis. Apresentaremos o processo de codificação e discutiremos o problema de rotação de fase. Na seção 3.4 repetiremos o procedimento da seção 3.3, desta vez com códigos convolucionais multiníveis.

3.2 Espaços de Sinais PSK

Um espaço de sinais $q - PSK$ pode ser representado no plano complexo por q pontos, $s_0, s_1, \ldots, s_{q-1}$, dispostos uniformemente sobre uma circunferência de centro na origem, como mostra a figura 3.1. Considerando o sistema com energia normalizada, isto é, a circunferência com raio unitário, esses pontos representam as raizes q-ésimas da unidade, ou seja, os números complexos $z_k = e^{\frac{2k\pi j}{q}}$, $k = 0, 1, \ldots, q-1$. A distância Euclidiana quadrada (D_E^2) entre dois sinais s_l e s_m é dada por

$$\begin{aligned} \mathrm{D}_E^2(s_l, s_m) &= |z_l - z_m|^2 = \left| e^{\frac{2l\pi j}{q}} - e^{\frac{2m\pi j}{q}} \right|^2 = \left| e^{\frac{2m\pi j}{q}} \right|^2 \left| \frac{e^{\frac{2l\pi j}{q}}}{e^{\frac{2m\pi j}{q}}} - 1 \right|^2 \\ &= \left| e^{\frac{2(l-m)\pi j}{q}} - 1 \right|^2 = \left| e^{\frac{2(l\ominus m)\pi j}{q}} - 1 \right|^2, \end{aligned} \tag{3.1}$$

onde $\ominus$ denota a diferença módulo q. O valor da expressão (3.1) depende apenas dos índices l e m. Então a D_E^2 entre dois pontos pode ser definida como uma função dos seus índices,

da seguinte maneira:

$$D_E^2(s_l, s_m) = D_E^2(l, m) \triangleq \left| e^{\frac{2(l \ominus m)\pi j}{q}} - 1 \right|^2. \tag{3.2}$$

Daí segue que o peso Euclidiano quadrado (w_E^2) é dado por

$$w_E^2(l) \triangleq \mathrm{D}_E^2(l, 0) = \left| e^{\frac{2l\pi j}{q}} - 1 \right|^2. \tag{3.3}$$

De um modo geral, temos que

$$D_E^2(l, m) = w_E^2(l \ominus m). \tag{3.4}$$

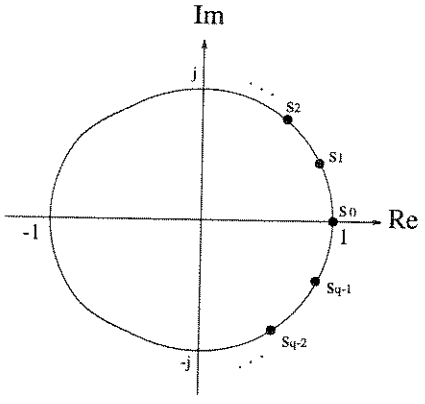

Figura 3.1: Espaço de Sinais q-PSK

Analogamente, a D_E^2 entre duas n-uplas $\mathbf{x} = (x_1, \ldots, x_n)$ e $\mathbf{y} = (y_1, \ldots, y_n)$, onde $x_l, y_l \in \mathbb{Z}_q$, $l = 1, \ldots, n$, é dada por

$$D_E^2(\mathbf{x}, \mathbf{y}) \triangleq \sum_{l=1}^{n} D_E^2(x_l, y_l) = \sum_{l=1}^{n} \left| e^{\frac{2(x_l \ominus y_l)\pi j}{q}} - 1 \right|^2. \tag{3.5}$$

O conjunto de sinais $q-PSK$ pode ser identificado com o anel $\mathbb{Z}_q$, fazendo a correspondência $s_l \longleftrightarrow l$, entre os sinais de $q-PSK$ e os elementos de $\mathbb{Z}_q$. Devido a essa identificação, nos referimos a um símbolo de $\mathbb{Z}_q$ como se ele fosse o sinal correspondente de $q-PSK$. Quando um sinal s_l é rotacionado por um ângulo de $\frac{2r\pi}{q}\, rad$, no sentido anti-horário, o sinal resultante é $s_{l \oplus r}$. Daí vemos que tal rotação resulta no sinal correspondente ao símbolo $l \oplus r$ de $\mathbb{Z}_q$. Denotando por $\rho_r(s_l)$ a rotação do sinal s_l por um ângulo $\frac{2r\pi}{q}\, rad$ temos que $\rho_r(s_l) = s_{l \oplus r}$ ou, de modo equivalente, $\rho_r(l) = l \oplus r$.

Definição 3.1 *Se* $\mathbf{x} = (x_1, \ldots, x_n)$ *é uma n-upla com símbolos em* $\mathbb{Z}_q$, *uma rotação de um ângulo* θ *em* $\mathbf{x}$ *significa uma rotação de* θ *em cada componente de* $\mathbf{x}$.

Observação 3.1 *Se* $\mathbf{x} = (x_1, \ldots, x_n)$ *temos que*

$$\rho_r(\mathbf{x}) = (\rho_r(x_1), \ldots, \rho_r(x_n)) = (x_1 \oplus r, \ldots, x_n \oplus r). \tag{3.6}$$

Denotando por $\rho(\mathbf{x})$ *a rotação de* $\frac{2\pi}{q}$ rad *em* $\mathbf{x} = (x_1, \ldots, x_n)$, *temos que*

$$\rho(\mathbf{x}) = (x_1 \oplus 1, \ldots, x_n \oplus 1). \tag{3.7}$$

De 3.6 e 3.7 segue que

$$\rho_r(\mathbf{x}) = \rho^r(\mathbf{x}), \tag{3.8}$$

onde ρ^r *denota a composição da função* ρ *com ela mesma* r *vezes.*

3.3 Modulações PSK Codificadas com Códigos de Bloco Multiníveis [1, 3]

3.3.1 Introdução

Nesta seção vamos considerar esquemas de modulações PSK codificadas com códigos de bloco sobre os anéis $\mathbb{Z}_q$, onde q é uma potência de 2. Quando o esquema de referência for uma modulação $2^m - PSK$, geralmente consideraremos como conjunto de sinais expandido uma modulação $2^{m+1} - PSK$ e, neste caso, utilizaremos códigos de bloco multiníveis sobre o anel $\mathbb{Z}_q$, de taxa $R = \frac{m}{m+1}$, onde $q = 2^{m+1}$. Dessa maneira, os dois esquemas, codificado e não codificado, terão as mesmas taxas de transmissão de informação. O processo de codificação, que será apresentado na próxima seção, não requer particionamento de conjunto pois os símbolos de informação são mapeados diretamente nos sinais da modulação. O problema de rotação de fase também será analisado.

3.3.2 Processo de Codificação

Cada bloco de $m+1$ bits, saindo de uma fonte binária, é mapeado em um dos símbolos de $\mathbb{Z}_q$, onde $q = 2^{m+1}$. Geralmente é utilizado o mapeamento de Gray pois isto minimiza o número de bits errados quando um símbolo é decodificado erradamente. Cada sequência de k desses blocos, ou seja, cada sequência de $k(m+1)$ bits que sai dessa fonte binária é mapeada numa sequência $\mathbf{a} = (a_1, \ldots, a_k)$ cujos símbolos pertencem ao anel $\mathbb{Z}_q$, chamada **sequência de informação**. Esta sequência de informação é enviada para o **codificador de bloco multinível** (CBM) onde é codificada e transformada numa **sequência codificada** $\mathbf{c} = (c_1, \ldots, c_n)$, cujos símbolos também pertencem a $\mathbb{Z}_q$, que é chamada de **palavra código**. Essa palavra código, depois de modulada é enviada através do canal. Como a taxa do código é $R = \frac{m}{m+1}$ devemos ter $\frac{k}{n} = \frac{m}{m+1}$. A figura 3.2 ilustra esse processo.

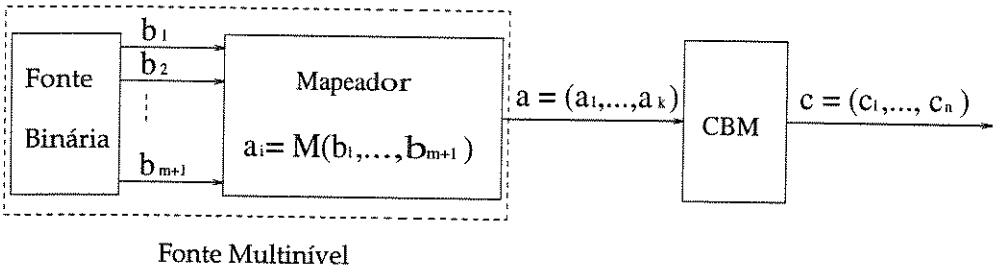

Figura 3.2: Processo de codificação de uma modulação $q - PSK$, utilizando um código de bloco multinível de taxa $R = \frac{k}{n}$.

O conjunto formado pela fonte binária e o mapeador pode ser considerado como um único bloco, chamado **fonte multinível**.

3.3.3 Rotação de Fase

Os espaços de sinais $q-PSK$ têm ambiguidade de fase de múltiplos de $\frac{2\pi}{q}\, rad$. Essas ambiguidades são ocasionadas por rotações de fase da portadora dos sinais transmitidos através do canal. Assim, caso não sejam tomadas providências para anular o efeito dessas

rotações, o desempenho do sistema pode ser seriamente afetado.

Nesta seção veremos que o uso de um codificador/decodificador diferencial juntamente com códigos adequados tornam o sistema imune aos efeitos dessas rotações de fase.

Definição 3.2 *Dizemos que um código $\mathcal{C}$ é* ***rotacionalmente invariante*** *(RI) sob um ângulo θ se $\rho_\theta(\mathbf{x}) \in \mathcal{C}$, $\forall \mathbf{x} \in \mathcal{C}$, onde $\rho_\theta(\mathbf{x})$ significa a rotação do vetor $\mathbf{x}$ por um ângulo θ, no sentido anti-horário.*

Definição 3.3 *Um código é dito* ***transparente*** *se ele é RI sob todas as ambiguidades de fase do esquema de modulação.*

Se $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_q^n$, vimos que (equação 3.8)

$$\rho_r(\mathbf{x}) = (x_1 \oplus r, \ldots, x_n \oplus r) = \rho^r(\mathbf{x}). \tag{3.9}$$

Assim, num esquema de modulação $q - PSK$, para verificarmos se um código multinível sobre $\mathbb{Z}_q$ é transparente basta verificarmos se ele é RI sob o ângulo $\theta = \frac{2\pi}{q}\, rad.$

O teorema seguinte caracteriza os códigos de bloco lineares multiníveis transparentes.

Teorema 3.1 *Um código de bloco linear $\mathcal{C}$ sobre $\mathbb{Z}_q$ é transparente se, e somente se, $\mathbf{u} = (1, \ldots, 1)$ é uma palavra código.*

Demonstração: Sendo $\mathcal{C}$ um código linear, temos que a sequência toda nula $\mathbf{u}_0 = (0, \ldots 0)$ é uma palavra código. Se $\mathcal{C}$ é transparente então $\rho(\mathbf{u}_0)$ também é uma palavra código. Como $\rho(\mathbf{u}_0) = \mathbf{u}$, segue que $\mathbf{u}$ é uma palavra código.
Reciprocamente, suponhamos que $\mathbf{u}$ é uma palavra código e seja $\mathbf{x} = (x_1, \ldots, x_n)$ uma palavra código qualquer de $\mathcal{C}$. Temos que

$$\rho(\mathbf{x}) = (x_1 \oplus 1, \ldots, x_n \oplus 1) = (x_1, \ldots, x_n) \oplus (1, \ldots, 1) = \mathbf{x} \oplus \mathbf{u}. \tag{3.10}$$

Como $\mathbf{x}$ e $\mathbf{u}$ são palavras código e $\mathcal{C}$ é linear, segue que $\rho(\mathbf{x})$ é uma palavra código. Portanto $\mathcal{C}$ é transparente. ■

Do teorema 3.1 concluimos que para um código de bloco multinível ser transparente basta que a sua matriz geradora tenha uma linha com todos os elementos iguais a 1 ou que a soma dos elementos de cada coluna seja 1.

O uso de códigos transparentes juntamente com um codificador/decodificador diferencial tornam o sistema imune aos problemas causados por rotações de fase. Na figura 3.3 o símbolo D dentro dos quadrados significa um atraso de tempo de um símbolo da sequência. Assim, quando a sequência de informação $\mathbf{b}(D) = b_0 + b_1 D + b_2 D^2 + \cdots$ é diferencialmente codificada, obtemos a sequência $\mathbf{a}(D) = a_0 + a_1 D + a_2 D^2 + \cdots$, onde

$$\mathbf{a}(D) = \mathbf{b}(D) \oplus \mathbf{a}(D)D. \tag{3.11}$$

Podemos supor, sem perda de generalidade, que o estado inicial das memórias do codificador e do decodificador diferenciais é zero. De 3.11 temos que

$$a_l = b_l \oplus a_{l-1},\ l = 0, 1, \ldots \tag{3.12}$$

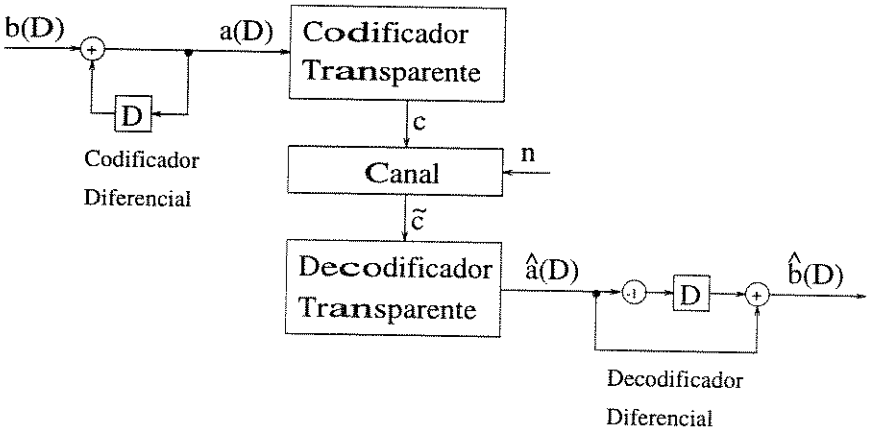

Figura 3.3: Codificação Diferencial

Como o código tem taxa $R = \frac{k}{n}$, a sequência $\mathbf{a}(D)$ é dividida em blocos de k símbolos cada, que entram no CBM e são codificados, originando uma palavra código $\mathbf{c} = (c_1, \ldots, c_n)$, que é enviada através do canal. Se o canal introduz uma rotação de fase de $\frac{2r\pi}{q}\, rad$ na

palavra código $\mathbf{c}$ então a sequência que chega ao decodificador linear é do tipo

$$\bar{\mathbf{c}} = \rho_r(\mathbf{c}) \oplus \mathbf{n},$$

onde $\mathbf{n}$ é uma sequência que representa o ruído do canal. Se o código é transparente então $\rho_r(\mathbf{c})$ é uma palavra código e, neste caso, o decodificador linear realiza o processo de decodificação como se $\rho_r(\mathbf{c})$ fosse a palavra código que saiu do CBM. Então, caso não haja erro de decodificação, a sequência que sai do decodificador linear é

$$\hat{\mathbf{a}} = \rho_r(\mathbf{a}) = \mathbf{a}(D) \oplus r\mathbf{u}(D) \tag{3.13}$$

onde $\mathbf{u}(\mathbf{D})$ é a sequência toda 1. Decodificando diferencialmente a sequência $\hat{\mathbf{a}}$, obtemos

$$\hat{\mathbf{b}}(D) = \hat{\mathbf{a}}(D) \ominus \hat{\mathbf{a}}(D)D. \tag{3.14}$$

De 3.13 segue que

$$\hat{\mathbf{b}}(D) = [\mathbf{a}(D) \oplus r\mathbf{u}(D)] \ominus [\mathbf{a}(D) \oplus r\mathbf{u}(D)]\, D =$$

$$[\mathbf{a}(D) \ominus \mathbf{a}(D)D] \oplus r\,[\mathbf{a}(D) \ominus \mathbf{a}(D)D]\,.$$

Como $\mathbf{u}(D)$ é a sequência toda 1 então $\mathbf{u}(D)D$ também é a sequência toda 1, pois o atraso de 1 símbolo desta sequência não a altera, ou seja, $\mathbf{u}(D) = \mathbf{u}(D)D$. Logo $\mathbf{u}(D) \ominus \mathbf{u}(D)D = \mathbf{0}$. Então

$$\hat{\mathbf{b}}(D) = \mathbf{a}(D) \ominus \mathbf{a}(D)D.$$

Mas de 3.11 temos que $\mathbf{a}(D) \ominus \mathbf{a}(D)D = \mathbf{b}(D)$. Portanto, se não houve erro de decodificação, $\hat{\mathbf{b}}(D) = \mathbf{b}(D)$.

3.3.4 Ganho de Codificação

A taxa de erro de bit de um sistema de comunicação digital é uma função da razão de energia média de bit por ruído $\frac{E_b}{N_0}$. O **ganho de codificação** g de um esquema de modulação codificada, a uma determinada taxa de erro de bit, é dado pela diferença entre os valores de

$\frac{E_b}{N_0}$ $[dB]$ necessários para que os sistemas codificado e não codificado (sistema de referência) atinjam essa taxa de erro de bit, ou seja,

$$g = \left(\frac{E_b}{N_0}\Big|_{nc} - \frac{E_b}{N_0}\Big|_{c} \right) [dB] \tag{3.15}$$

O limite da expressão 3.15 quando $\frac{E_b}{N_0} \longrightarrow \infty$ é chamado de **ganho de codificação assintótico**, é denotado por g_∞, e é dado por

$$g_\infty = 10 \log_{10} \left(\frac{\log M_c}{\log M_{nc}} R \frac{D^2_{E_c}}{D^2_{E_{nc}}} \right) [dB] \tag{3.16}$$

onde M_c e M_{nc} são o número de sinais dos esquemas codificado e não codificado, respectivamente; R é a taxa do código; $D^2_{E_c}$ e $D^2_{E_{nc}}$ são as distâncias Euclidianas quadradas dos esquemas codificado e não codificado, respectivamente. Omitimos as bases dos logaritmos em $\frac{\log M_c}{\log M_{nc}}$ porque o valor desse quociente não depende da base do logaritmo. Entretanto, é conveniente usar a base 2, já que M_c e M_{nc} são potências de 2.

A figura 3.4 mostra o desempenho de um código multinível $(10, 5)$ sobre $\mathbb{Z}_4$ adequado para uma modulação $4 - PSK$, em relação à modulação $2 - PSK$ não codificada. Para uma taxa de erro de bit igual a 10^{-5}, o ganho de codificação é de aproximadamente $3,6\,dB$.

Em [1, 3] são apresentados vários exemplos de códigos de bloco multiníveis sobre $\mathbb{Z}_q$ com os respectivos ganhos de codificação assintóticos, do esquema codificado sobre o esquema de referência. Com um código $(22, 11)$ transparente, sobre $\mathbb{Z}_4$, foi obtido um ganho de $7,0\,dB$ do $4 - PSK$ codificado sobre o $2 - PSK$ não codificado. Com um código $(14, 7)$ sobre $\mathbb{Z}_{16}$, foi obtido um ganho de $7,27\,dB$ do $16 - PSK$ codificado sobre o $8 - PSK$ não codificado. Devemos observar que, neste último caso, os dois esquemas não têm as mesmas taxas de transmissão de informação. Os códigos sobre $\mathbb{Z}_8$, de taxa $\frac{2}{3}$, adequados para $8 - PSK$, não apresentaram desempenhos tão bons quanto estes, uma vez que o maior ganho obtido foi de $3,36\,dB$, do $8 - PSK$ codificado sobre o $4 - PSK$ não codificado, com um código $(12, 8)$.

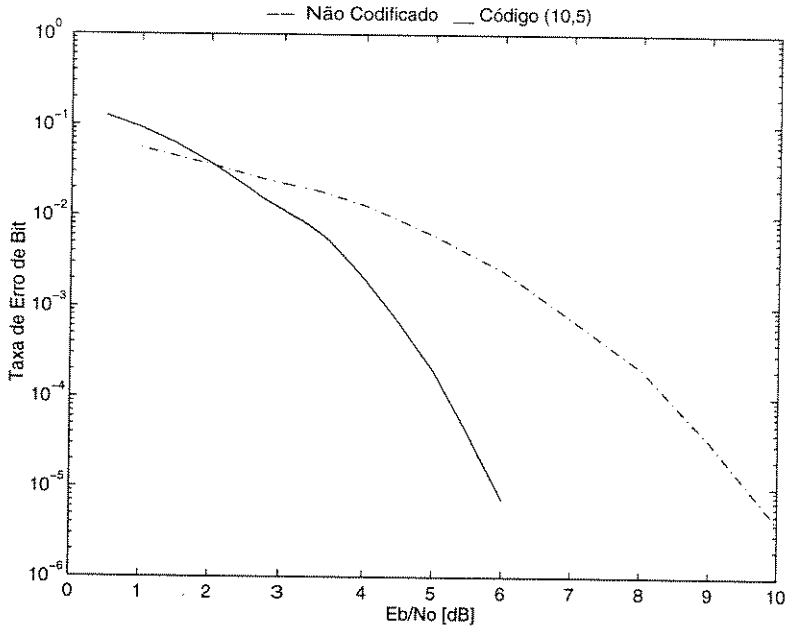

Figura 3.4: Desempenho do $4-PSK$ codificado em relação ao $2-PSK$ não codificado.

3.4 Modulações PSK Codificadas com Códigos Convolucionais Multiníveis [1, 4]

3.4.1 Introdução

Nesta seção vamos considerar esquemas de modulações PSK codificadas com códigos convolucionais sobre os anéis $\mathbb{Z}_q$, onde q é uma potência de 2. Quando o esquema de referência for uma modulação $2^m - PSK$, geralmente consideraremos como conjunto de sinais expandido uma modulação $2^{m+1} - PSK$ e, neste caso, utilizaremos códigos convolucionais multiníveis sobre o anel $\mathbb{Z}_q$, de taxa $R = \frac{m}{m+1}$, onde $q = 2^{m+1}$. Dessa maneira, os dois esquemas, codificado e não codificado, terão as mesmas taxas de transmissão de informação. O processo de codificação, a exemplo do que ocorreu na seção 3.3, não requer particionamento de conjunto pois os símbolos de informação são mapeados diretamente nos sinais da

modulação.

3.4.2 Processo de Codificação

Vamos descrever o processo de codificação, que utiliza um código convolucional multinível sobre $\mathbb{Z}_q$, $q = 2^{m+1}$, de taxa $R = \frac{m}{m+1}$. Primeiramente é feito o mapeamento dos bits, que são emitidos pela fonte binária, nos símbolos de $\mathbb{Z}_q$. Como $q = 2^{m+1}$, cada bloco de $m+1$ bits é mapeado num dos símbolos de $\mathbb{Z}_q$. Geralmente é utilizado o mapeamento de Gray pois isto minimiza o número de bits errados quando um símbolo é decodificado erradamente. Como o código tem taxa $R = \frac{m}{m+1}$, a cada tempo discreto τ, entra no **codificador convolucional multinível** (CCM) uma **sequência de informação** $\mathbf{a} = (a_1, \ldots, a_m)$, cujos símbolos pertencem a $\mathbb{Z}_q$. Esta sequência de informação é codificada e transformada numa **sequência codificada** $\mathbf{c} = (c_1, \ldots, c_{m+1})$, cujos símbolos também pertencem a $\mathbb{Z}_q$, e que, depois de modulada, é enviada através do canal. A figura 3.5 ilustra esse processo. O conjunto formado pela fonte binária e o mapeador pode ser considerado como um único bloco, chamado **fonte multinível**.

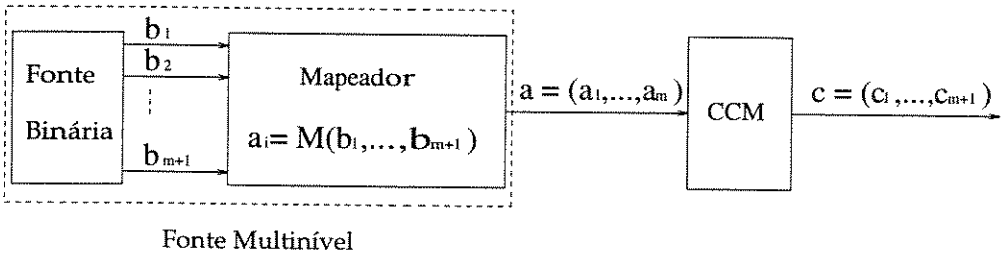

Figura 3.5: Processo de codificação de uma modulação $q - PSK$, utilizando um código convolucional multinível de taxa $R = \frac{m}{m+1}$.

Codificador Convolucional Multinível (CCM)

Vamos representar o código convolucional multinível, usado no processo de codificação descrito acima, pela sua matriz geradora $\mathbf{G}(D)$ e vamos supor que essa matriz esteja na

forma sistemática. Então

$$\mathbf{G}(D) = [\mathbf{I}_m \,|\mathbf{P}(D)] = \left[\begin{array}{c|c} & g^{(1)}(D)/f(D) \\ & g^{(2)}(D)/f(D) \\ \mathbf{I}_m & \vdots \\ & g^{(m)}(D)/f(D) \end{array}\right], \tag{3.17}$$

onde $\mathbf{I}_m$ é a matriz identidade de ordem m, $\mathbf{P}(D)$ é uma matriz de ordem $m \times 1$ e

$$\begin{aligned} g^{(i)}(D) &= g_s^{(i)} D^s + \cdots + g_1^{(i)} D + g_0^{(i)} \in \mathbb{Z}_q[D] \\ f(D) &= f_s D^s + \cdots + f_1 D + 1 \in \mathbb{Z}_q[D] \end{aligned} \tag{3.18}$$

são os polinômios responsáveis pelas conexões de alimentação e realimentação, respectivamente. A figura 3.6 ilustra o CCM. Também usamos a notação

$$\left(g_s^{(m)} \ldots g_s^{(1)}\right) \cdot \cdot \cdot \left(g_1^{(m)} \ldots g_1^{(1)}\right) / \left(f_s \ldots f_1 1\right)$$

para representar o código gerado pela matriz 3.17.

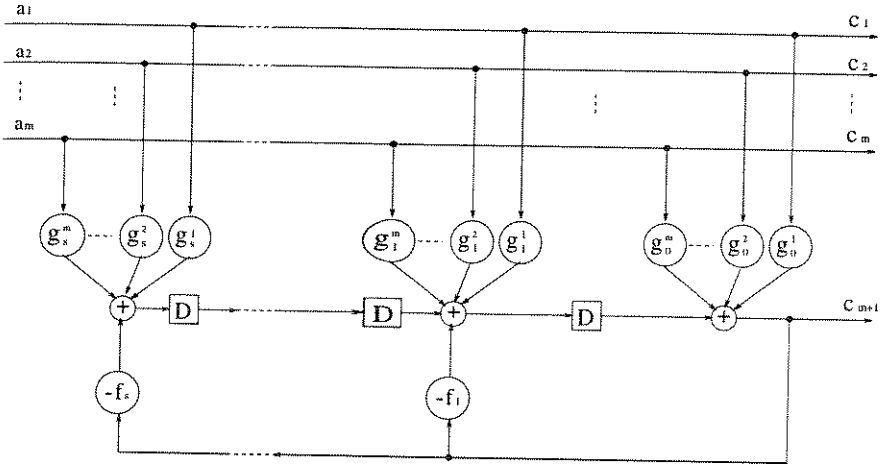

Figura 3.6: Estrutura do Codificador Convolucional Multinível (CCM).

A matriz de verificação de paridade é

$$\mathbf{H}(D) = \left[-\mathbf{P}(D)^t \,|1\right]. \tag{3.19}$$

Em cada tempo discreto τ temos que

$$\mathbf{a}(D)\mathbf{G}(D) = \mathbf{c}(D). \tag{3.20}$$

Observando a figura 3.6 vemos que o valor $v_i^{(\tau)}$ armazenado no i-ésimo elemento de memória, no tempo discreto τ, é

$$v_i^{(\tau)} = \begin{cases} -f_i c_{m+1} \oplus \sum_{j=1}^{m} g_i^{(j)} a_j \oplus v_{i+1}^{(\tau-1)}, & se \quad 1 \leq i < s \\ -f_i c_{m+1} \oplus \sum_{j=1}^{m} g_i^{(j)} a_j, & se \quad i = s \end{cases} \tag{3.21}$$

As operações em 3.21 são realizadas no anel $\mathbb{Z}_q$. Se N é o número de estados do CCM então

$$N \leq q^s. \tag{3.22}$$

A igualdade ocorre, ou seja, $N = q^s$ se, e somente se, todos os valores de $\mathbb{Z}_q$ são asssumidos pelos $v_i^{(\tau)}$ (equações 3.21). Para que tenhamos a desigualdade $N < q^s$, basta que todos os coeficientes $g_i^{(j)}$ e f_i sejam pares, exceto f_0 que é 1. Neste caso todos os $v_i^{(\tau)}$ são pares.

3.4.3 Rotação de Fase

A definição de código transparente dada na subseção 3.3.3 e o teorema 3.1 podem ser facilmente estendidos para códigos convolucionais multiníveis. De modo análogo ao que fizemos para códigos de bloco, podemos mostrar que o uso de um codificador/decodificador diferencial juntamente com um código convolucional multinível transparente tornam o sistema imune aos problemas causados por rotações de fase da portadora

3.4.4 Ganho de Codificação

O ganho de codificação assintótico do esquema codificado em relação ao esquema de referência, neste caso, é dado pela expressão

$$g_\infty = 10 \log_{10} \left(\frac{\log M_c}{\log M_{nc}} R \frac{D_{free}^2}{D_{E_{nc}}^2} \right) [dB], \tag{3.23}$$

onde M_c e M_{nc} são o número de sinais dos esquemas codificado e não codificado, respectivamente; R é a taxa do código; D_{free}^2 e $D_{E_{nc}}^2$ são as distâncias Euclidianas quadradas dos esquemas codificado e não codificado, respectivamente.

Observação 3.2 *Se os esquemas codificado e não codificado são* $2^{m+1}-PSK$ *e* 2^m-PSK, *respectivamente, e o código tem taxa* $R=\frac{m}{m+1}$, *a expressão 3.23 assume a forma*

$$g_\infty = 10\log_{10}\left(\frac{D^2_{free}}{D^2_{E_{nc}}}\right)[dB]\,. \tag{3.24}$$

Exemplo 3.1 *A matriz*

$$\mathbf{G}(D) = [1 \;\; (D+2)/(2D+1)]$$

define um código convolucional linear sistemático com realimentação, de taxa $\frac{1}{2}$, *sobre* $\mathbb{Z}_4$, *adequado para modulações* $4-PSK$. *A figura 3.7 mostra a estrutura do CCM. Na tabela 3.1*

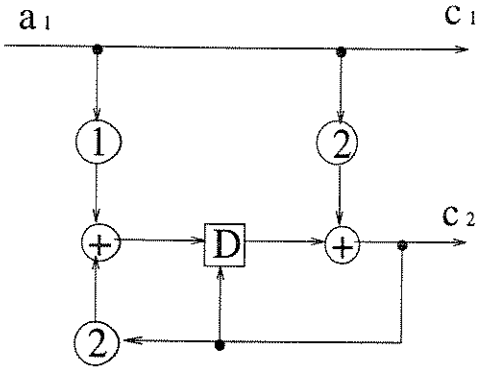

Figura 3.7: Estrutura do CCM do código 12/21.

mostramos todas as possíveis transições entre dois estados no diagrama de treliça e na figura 3.8 mostramos essas transições na treliça. Da figura 3.8 concluimos que a palavra toda 1 *é uma palavra código e que* $D^2_{free}=8$. *Daí segue que* $g_\infty = 3,01\,dB$ *sobre o esquema* $2-PSK$ *não codificado.*

Em [1, 4] são apresentados vários exemplos de códigos convolucionais multiníveis sobre $\mathbb{Z}_q$, com os respectivos ganhos de codificação assintóticos do esquema codificado sobre o esquema de referência. Com códigos sobre $\mathbb{Z}_4$, com 1024 estados, foram obtidos ganhos assintóticos de $8,45\,dB$ com código não transparente, e de $7,0\,dB$ com código transparente, do $4-PSK$ codificado sobre o $2-PSK$ não codificado. Códigos sobre $\mathbb{Z}_8$ também apre-

Entrada (a_1)	*Estado* (τ)	*Estado* ($\tau+1$)	*Saida* (c_1c_2)
0	0	0	00
1	0	1	12
2	0	2	20
3	0	3	30
0	1	2	01
1	1	3	13
2	1	0	21
3	1	1	33
0	2	0	02
1	2	1	10
2	2	2	22
3	2	3	30
0	3	2	03
1	3	3	11
2	3	0	23
3	3	1	31

Tabela 3.1: Transições de estados para o CCM do código 12/21.

sentaram bons desempenhos, com ganhos de até $6,0\,dB$, do $8-PSK$ codificado sobre o $2-PSK$ não codificado, com código transparente de 512 estados.

3.5 Conclusão

Neste capítulo apresentamos técnicas de codificação de modulações $q-PSK$, $q=2^m$, utilizando códigos multiníveis sobre $\mathbb{Z}_q$. Códigos de bloco e convolucionais foram considerados e, em ambos os casos, foram obtidos bons ganhos de codificação assintóticos, considerando-se as modulações $2^{m+1}-PSK$ e 2^m-PSK como esquemas codificado e não

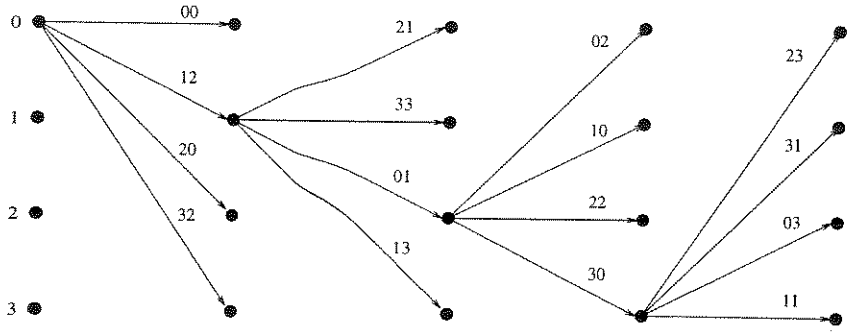

Figura 3.8: Diagrama de Treliça para o Código 12/21.

codificado, respectivamente. Os códigos convolucionais apresentaram, em geral, desempenho melhor que os códigos de bloco, em termos de ganho de codificação assintótico. Os problemas causados por rotações de fase da portadora, introduzidas pelo canal, também foram analisados.

Capítulo 4

Modulações QAM Codificadas com Códigos Convolucionais Multiníveis

4.1 Introdução

As ambiguidades de fase dos espaços de sinais $q-QAM$ retangulares são de múltiplos de $\frac{\pi}{2}\,rad$, enquanto que os códigos transparentes sobre $\mathbb{Z}_q$ são RI sob múltiplos de $\frac{2\pi}{q}\,rad$. Assim, quando $q > 4$, não é possível usar códigos multiníveis sobre $\mathbb{Z}_q$ para eliminar erros causados por rotações de fase da portadora, introduzidas pelo canal. Para contornarmos este problema, vamos considerar apenas os esquemas de modulação $q-QAM$, onde $q = 4^m$, $m \geq 1$, codificados com códigos multiníveis sobre $\mathbb{Z}_4$. Assim podemos encontrar códigos RI sob todas as ambiguidades de fase da modulação.

Como o espaço $4^m - QAM$ tem 4^m pontos podemos rotular os pontos desse espaço com m-uplas $x_1x_2\cdots x_m$, onde $x_i \in \mathbb{Z}_4$, $i = 1,\ldots,m$. No processo de codificação, como usamos códigos multiníveis sobre $\mathbb{Z}_4$, podemos fazer o mapeamento dos símbolos de $\mathbb{Z}_4$ nos símbolos dos rótulos dos pontos do espaço. O processo de codificação também envolve um particionamento de conjuntos, que depende da maneira como os pontos do espaço são rotulados. Como veremos adiante, com rotulamentos adequados podemos otimizar o desempenho dos códigos, maximizando a distância Euclidiana entre sequências codificadas, e também podemos en-

contrar códigos transparentes, que tornam os sistemas imunes aos problemas causados por rotações de fase da portadora. Na seção seguinte apresentamos um método para se fazer tais rotulamentos. A seguir mostramos como utilizar códigos convolucionais multiníveis sobre $\mathbb{Z}_4$ em esquemas de modulações $4^m - QAM$. Apresentamos também o processo de codificação e mostramos como os sinais de $4^m - QAM$ são associados aos ramos do diagrama de treliça [17].

4.2 Rotulamento dos Pontos de um Espaço de Sinais 4^m-QAM

Nesta seção, com o objetivo de simplificarmos a notação, vamos denotar o espaço $4^m - QAM$ por $\mathbf{S}^m$ e o espaço $4 - QAM$ por $\mathbf{S}$. Existe uma correspondência biunívoca entre os pontos (sinais) do espaço $\mathbf{S}^m$ e o conjunto $\mathbb{Z}_4^m \triangleq \{(x_1, \ldots, x_m) \,|x_i \in \mathbb{Z}_4,\, i = 1, \ldots, m\}$. Então podemos rotular cada sinal de $\mathbf{S}^m$ com uma m-upla ordenada $x_1 x_2 \ldots x_m$ cujos símbolos pertencem a $\mathbb{Z}_4$. Além disso, podemos definir operações de soma e produto que tornam $\mathbf{S}^m$ um anel isomorfo a $\mathbb{Z}_4^m$, basta que a soma (produto) de dois sinais s_1 e s_2 seja definida como sendo o sinal s cujo rótulo é a soma (produto) módulo 4 componente a componente dos rótulos de s_1 e s_2. A aplicação

$$\varphi_m : \mathbb{Z}_4^m \rightarrow \mathbf{S}^m \; ; \quad \varphi_m (z_1, \ldots, z_m) = s$$

é um isomorfismo, onde s é o sinal cujo rótulo é $z_1 \cdots z_m$. Assim, o espaço $\mathbf{S}^m$ é, de fato, um produto direto de $\mathbf{S}$, ou seja,

$$\mathbf{S}^m = \underbrace{\mathbf{S} \times \cdots \times \mathbf{S}}_{m \text{ vezes}} \tag{4.1}$$

pois

$$\mathbb{Z}_4 \simeq \mathbf{S} \;\Rightarrow\; \mathbb{Z}_4 \times \cdots \times \mathbb{Z}_4 \simeq \mathbf{S} \times \cdots \times \mathbf{S}. \tag{4.2}$$

Definição 4.1 *Dizemos que um rotulamento de $\mathbf{S}^m$ é **transparente** se a rotação de qualquer sinal de $\mathbf{S}^m$ por um ângulo de $\frac{\pi}{2}$ rad, no sentido anti-horário, corresponde a somar 1, módulo 4, a cada símbolo do seu rótulo.*

Exemplo 4.1 *Na figura 4.1 mostramos os possíveis rotulamentos transparentes de* $\mathbf{S}$.

É claro que um rotulamento de $\mathbf{S}^m$ é transparente se, e somente se, o rotulamento de cada fator do produto direto da equação 4.1 é transparente. Não é necesário que todos os fatores tenham o mesmo rotulamento.

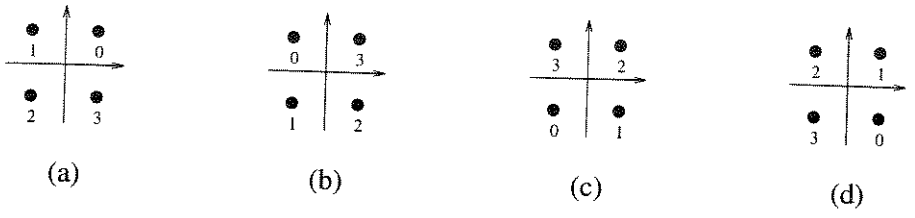

Figura 4.1: Rotulamentos transparentes de 4-QAM.

Vamos descrever um método indutivo de se obter um rotulamento transparente de $\mathbf{S}^m$ a partir de um rotulamento transparente de $\mathbf{S}^{m-1}$, $m \geq 2$. Na figura 4.2 mostramos rotulamentos transparentes de $\mathbf{S}$, $\mathbf{S}^2$ e $\mathbf{S}^3$ que ilustram esse método. No caso do rotulamento de $\mathbf{S}^3$, que foi feito a partir do de $\mathbf{S}^2$, seguimos o seguinte procedimento: dentro de um mesmo quadrante o símbolo da direita é o mesmo para todos os rótulos, e os dois símbolos mais à esquerda de cada rótulo seguem o rotulamento de $\mathbf{S}^2$. Convém observar que o símbolo da direita determina o quadrante em que o ponto se encontra, ou seja, podemos dizer que esse símbolo rotula os quadrantes. Esse rotulamento dos quadrantes deve ser feito de acordo com um dos rotulamentos transparentes de $\mathbf{S}$. Este procedimento pode ser usado para obter o rotulamento de $\mathbf{S}^m$ a partir do de $\mathbf{S}^{m-1}$. Neste caso, colocamos uma "cópia" de $\mathbf{S}^{m-1}$ em cada quadrante, ou seja, dispomos os 4^{m-1} pontos de $\mathbf{S}^{m-1}$, com os seus respectivos rótulos, em cada quadrante e, em seguida, acrescentamos à direita do rótulo de cada ponto o símbolo correspondente ao rótulo do quadrante em que ele se encontra. Dessa maneira, aplicando sucessivamente esse método, a partir de um rotulamento transparente de $\mathbf{S}$, podemos chegar a um rotulamento transparente de $\mathbf{S}^m$. Não é necessário usar o mesmo rotulamento de $\mathbf{S}$ em todas as etapas desse processo. Este método é uma consequência direta da equação 4.1, pois $\mathbf{S}^m = \mathbf{S}^{m-1} \times \mathbf{S}$. Na figura 4.3 mostramos um rotulamento transparente de $64 - QAM$

onde são usados diferentes rotulamentos de S.

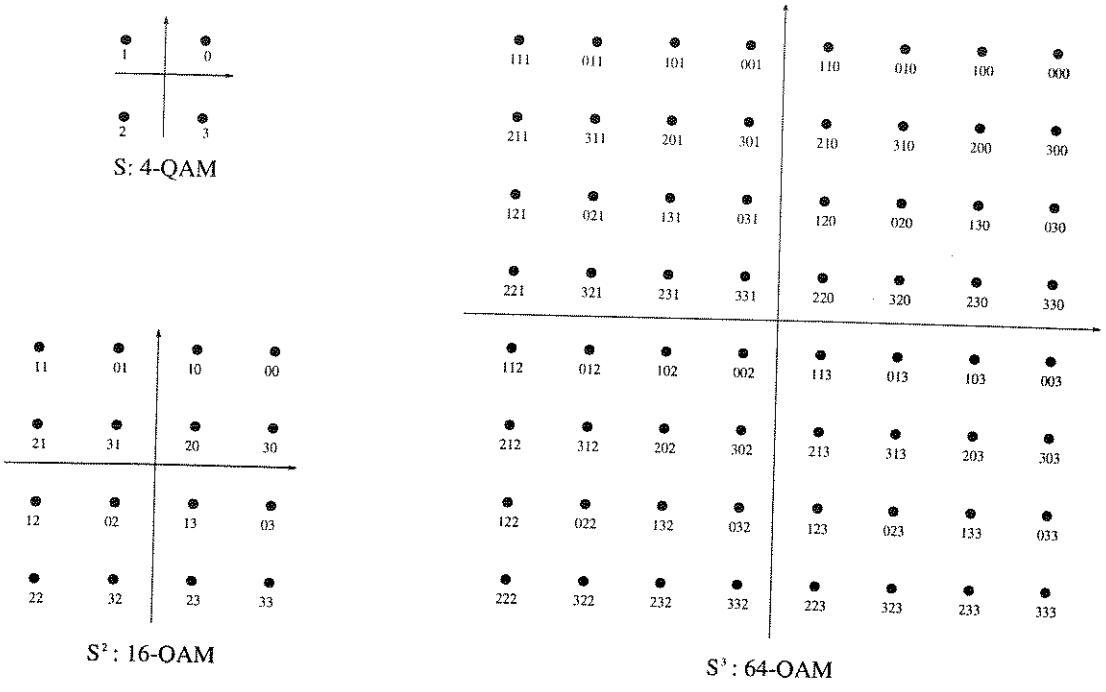

Figura 4.2: Rotulamentos transparentes de 4-QAM, 16-QAM e 64-QAM.

4.3 Rotação de Fase

Os espaços de sinais $4^m - QAM$ têm ambiguidades de fase de múltiplos de $\frac{\pi}{2}\,rad.$ Então, em um esquema de modulação $4^m - QAM$, códigos transparentes devem ser RI sob todos os múltiplos de $\frac{\pi}{2}\,rad.$ (definição 3.3, página 37). Mas, se $\mathbf{v} = (v_1, \ldots, v_n)$, $v_i \in \mathbb{Z}_4$, $i = 1, \ldots, n$ e $\rho_k(\mathbf{v})$ denota a rotação de $\mathbf{v}$ (no sentido anti-horário) por um ângulo $\theta = \frac{k\pi}{2}\,rad$, segue da equação 3.6 que $\rho_k(\mathbf{v}) = (v_1 \oplus k, \ldots, v_n \oplus k)$. Então, em um esquema de modulação $4^m - QAM$, para que um código multinível sobre $\mathbb{Z}_4$ seja transparente basta que ele seja RI sob o ângulo $\theta = \frac{\pi}{2}\,rad.$

No capítulo anterior, quando analisamos o problema de rotação de fase da portadora, não não foi feita alusão explícita ao rotulamento dos pontos do espaço de sinais $q - PSK$. Isto se

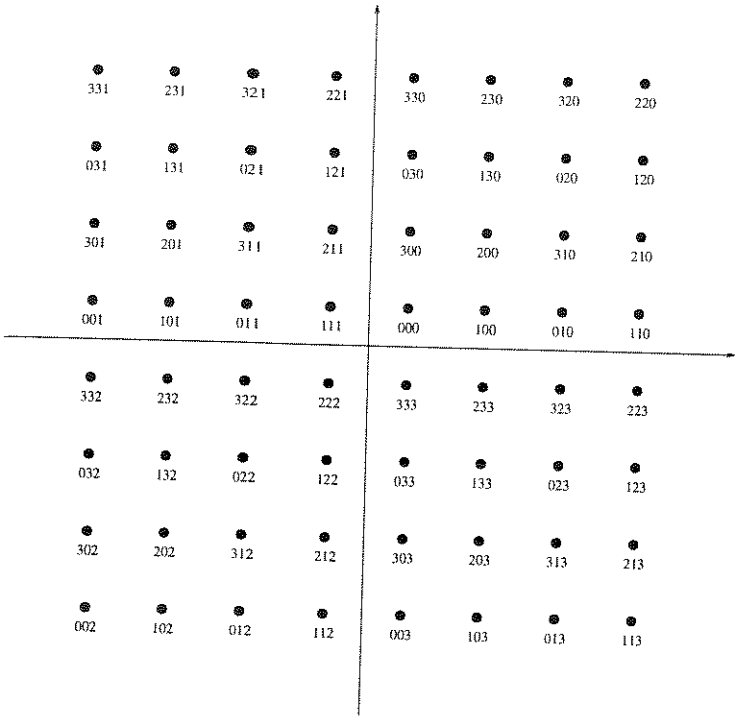

Figura 4.3: Rotulamento transparente de 64-QAM, com diferentes rotulamentos de 4-QAM.

deve ao fato de que para esses espaços existe um rotulamento natural, que casa perfeitamente com os códigos multiníveis sobre $\mathbb{Z}_q$. Como $\frac{2\pi}{q}\,rad$ é a ambiguidade de fase mínima do espaço $q-PSK$, segue da equação 3.7 que esse rotulamento natural é transparente. Este fato foi fortemente usado na demonstração do teorema 3.1, quando usamos a equação 3.10. É fácil ver que o teorema 3.1 também vale para códigos multiníveis sobre $\mathbb{Z}_4$ em modulações 4^m-QAM, desde que o rotulamento seja transparente.

Quando usamos códigos transparentes em esquemas de modulações 4^m-QAM e o rotulamento dos pontos do espaço também é transparente então os problemas causados por rotações de fase da portadora podem ser resolvidos com o uso de um codificador/decodificador diferencial.

4.4 Processo de Codificação [17]

Nesta seção vamos considerar esquemas de modulações $4^m - QAM$, $m \geq 2$, codificadas com códigos convolucionais multiníveis sobre $\mathbb{Z}_4$. Tomaremos como esquema de referência a modulação $\frac{4^m}{2} - QAM$. Em cada esquema utilizaremos um código convolucional multinível sobre $\mathbb{Z}_4$, de taxa $R = \frac{2m-1}{2m}$. Dessa maneira, os dois esquemas terão as mesmas taxas de transmissão de informação. O processo de codificação tem como base um particionamento de conjunto que visa maximizar a distância Euclidiana mínima entre sequências codificadas. O processo de codificação e o método de particionamento de conjunto serão descritos a seguir.

Consideremos um esquema de modulação $4^m - QAM$ codificada, $m \geq 2$, e tomemos como esquema de referência a modulação $\frac{4^m}{2} - QAM$ sem codificação. O processo de codificação utiliza um código convolucional multinível sobre $\mathbb{Z}_4$, de taxa $R = \frac{2m-1}{2m}$. Primeiramente é feito o mapeamento dos bits, que são emitidos pela fonte binária, nos símbolos de $\mathbb{Z}_4$. Cada par de bits é mapeado num símbolo de $\mathbb{Z}_4$ e estes símbolos são enviados em bloco para o codificador convolucional multinível (CCM). Geralmente é utilizado o mapeamento de Gray pois isto minimiza o número de bits errados quando um símbolo é decodificado erradamente. Como o código tem taxa $R = \frac{2m-1}{2m}$, a cada tempo discreto τ, entra no CCM uma sequência de informação $\mathbf{a} = (a_1, \ldots, a_{2m-1})$, cujos símbolos pertencem a $\mathbb{Z}_4$. Esta sequência $\mathbf{a}$ é codificada, dando origem a uma sequência $\mathbf{c} = (c_1, \ldots, c_{2m})$, cujos símbolos também pertencem a $\mathbb{Z}_4$. Como cada sinal de $4^m - QAM$ é rotulado por m símbolos de $\mathbb{Z}_4$, associamos dois sinais de $4^m - QAM$ à sequência $\mathbf{c}$, ou seja, a sequência $\mathbf{c}$ deve ser subdividida em duas subsequências $\mathbf{c}^1$ e $\mathbf{c}^2$, onde $\mathbf{c}^1 = (c_1, \ldots, c_m)$ e $\mathbf{c}^2 = (c_{m+1}, \ldots, c_{2m})$, e associamos um sinal s_i de $4^m - QAM$ à subsequência $\mathbf{c}^i$, $i = 1, 2$. Em outras palavras, à sequência $\mathbf{c}$ associamos um par ordenado $\mathbf{s} = (s_1, s_2)$ de sinais de $4^m - QAM$. Esse par de sinais é modulado e enviado através do canal. A figura 4.4 ilustra esse processo, e a figura 4.5 mostra um CCM sistemático com realimentação. O conjunto formado pela fonte binária e o mapeador pode ser considerado como um único bloco, chamado de fonte multinível. O mapeamento das subsequências $\mathbf{c}^i$, nos sinais s_i, $i = 1, 2$, é feito de acordo com um método de particionamento de conjuntos que garante uma certa distância Euclidiana mínima entre

sequências codificadas.

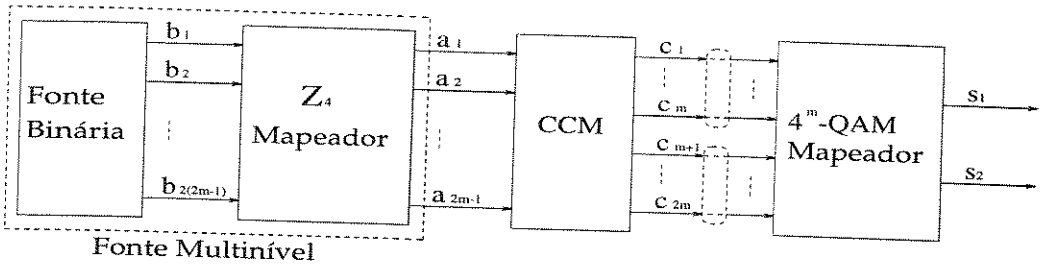

Figura 4.4: Processo de codificação de uma modulação $4^m - QAM$, usando um código convolucional de taxa $R = \frac{2m-1}{2m}$.

4.5 Particionamento [17]

No processo de codificação que acabamos de descrever, cada sequência de informação $\mathbf{a} = (a_1, \ldots, a_{2m-1})$, quando codificada, origina um par ordenado (s_1, s_2) de sinais de $4^m - QAM$. A maneira como esses pares (s_1, s_2) são associados aos ramos do diagrama de treliça é feita por meio de um particionamento do conjunto $(4^m - QAM) \times (4^m - QAM)$. Esse particionamento é feito a partir do particionamento de Ungerboeck [28] do conjunto $4^m - QAM$. Na figura 4.6 mostramos o particionamento de Ungerboeck do conjunto de pontos do espaço $16 - QAM$, que vamos usar para o particionamento de $(16 - QAM) \times (16 - QAM)$. Na figura 4.6 o conjunto $16 - QAM$ está representado por A_0 e a distância Euclidiana quadrada mínima é igual a d_0^2. Esse método de particionamento de conjunto reduz a complexidade de decodificação, quando o método de decodificação de Viterbi é usado em constelações maiores.

No nível 1 da partição dividimos A_0 em dois subconjuntos B_0 e B_1 tais que a $D^2_{E_{\min}}$ em cada um deles é $d_1^2 = 2d_0^2$. No nível 2 cada subconjunto do nível 1 é dividido em dois subconjuntos, resultando num total de quatro subconjuntos C_0, C_1, C_2 e C_3, tais que a $D^2_{E_{\min}}$ em cada um deles é $d_2^2 = 4d_0^2$. No nível 3 repetimos o processo e obtemos os subconjuntos

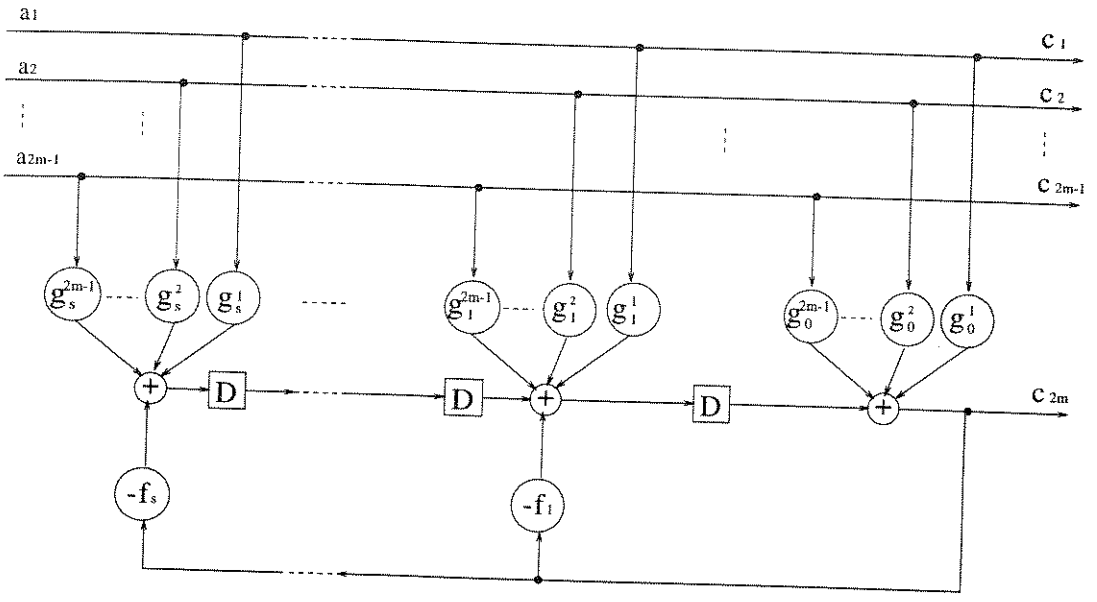

Figura 4.5: CCM de um código sistemático com realimentação, de taxa $R = \frac{2m-1}{2m}$.

D_i, $i = 0, 1, \ldots, 7$, tais que a $D^2_{E_{\min}}$ em cada um deles é $d_3^2 = 8d_0^2$.

Para simplificar a notação, vamos denotar o produto cartesiano de dois conjuntos **A** e **B** por **AB**. Assim, A_0A_0 representa o conjunto de todos os pares (s_i, s_j), onde s_i e s_j são sinais de $16 - QAM$.

Observação 4.1 *Se $S = S_1S_2$ e $T = T_1T_2$, onde S_i e T_i, $i = 1, 2$, podem ser A_0 ou qualquer um dos seus subconjuntos obtidos pelo particonamento de Ungerboeck (figura 4.6), então:*

1. *A $D^2_{E\min}$ entre elementos de um mesmo subconjunto é chamada distância* **intra-conjunto**. *Neste caso, temos que*

$$D^2_{E\min}(S) \triangleq \min\left\{D^2_{E\min}(S_1), D^2_{E\min}(S_2)\right\}. \tag{4.3}$$

2. *A distância* **inter-conjuntos**, *entre dois subconjuntos, é a $D^2_{E\min}$ entre os elementos de um dos subconjuntos e os elementos do outro. Neste caso, temos que*

$$D^2_{E\min}(S,T) \triangleq \min\left\{D^2_E\left((s_1,s_2),(t_1,t_2)\right);\ s_i \in S_i,\ t_i \in T_i,\ i = 1,2\right\} =$$

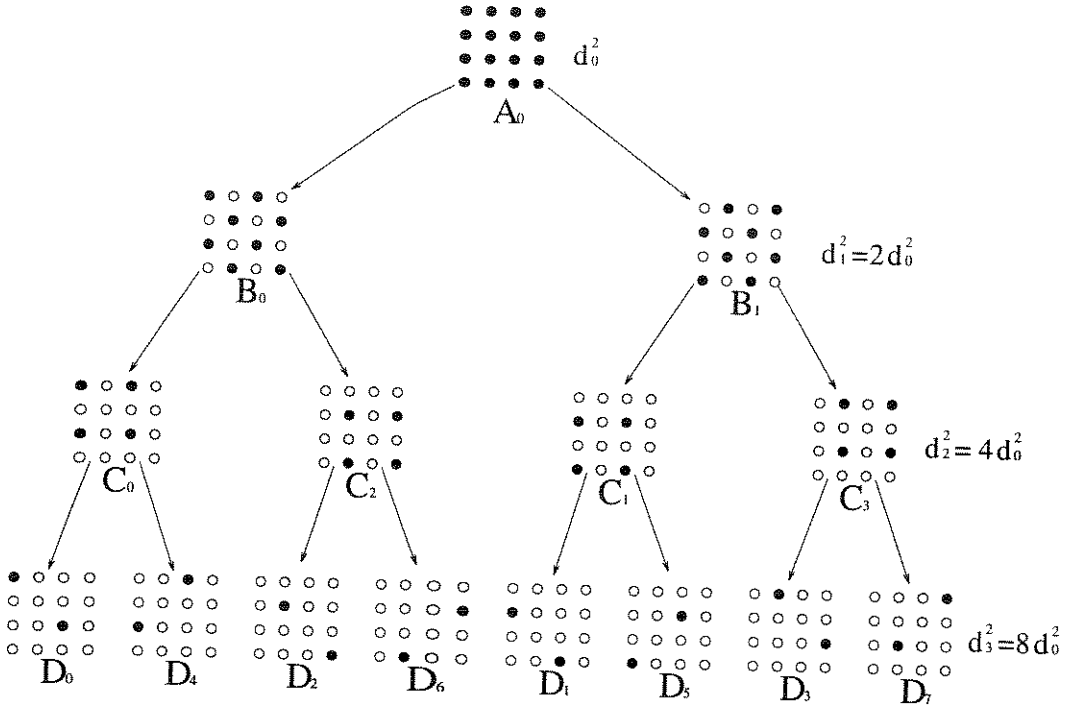

Figura 4.6: Particionamento de Ungerboeck do conjunto de pontos do espaço $16-QAM$.

$$= \min\left\{\sum_{i=1}^{2} D_E^2\left(s_i, t_i\right);\ s_i \in S_i,\ t_i \in T_i\right\}. \tag{4.4}$$

O primeiro nível da partição de A_0A_0 divide o conjunto A_0A_0 em quatro subconjuntos B_0B_0, B_0B_1, B_1B_0 e B_1B_1. Esses subconjuntos serão chamados de B-subconjuntos e este nível da partição será denotado por

$$[A_0A_0] = \begin{bmatrix} B_0B_0 & B_0B_1 \\ \\ B_1B_1 & B_1B_0 \end{bmatrix}. \tag{4.5}$$

Cada B-subconjunto contém 64 elementos, a distância intra-conjunto é $2d_0^2$ e a distância inter-conjuntos é d_0^2. Cada B-subconjunto será associado aos ramos que saem de cada estado do diagrama da treliça.

O segundo nível da partição consiste em dividir cada B-subconjunto em quatro subconjuntos, chamados C-subconjuntos, que são obtidos considerando-se os produtos C_iC_j,

$i, j = 0, 1, 2, 3$, e agrupados conforme a disposição dos B-subconjuntos em 4.5, ou seja,

$$[A_0 A_0] = \begin{bmatrix} C_0C_0 & C_0C_2 & C_0C_1 & C_0C_3 \\ C_2C_2 & C_2C_0 & C_2C_3 & C_2C_1 \\ & & & \\ C_1C_1 & C_1C_3 & C_1C_2 & C_1C_0 \\ C_3C_3 & C_3C_1 & C_3C_0 & C_3C_2 \end{bmatrix} \tag{4.6}$$

Da figura 4.6 vê-se que a distância intra-conjunto de cada C-subconjunto é igual a $4d_0^2$. Temos também que cada C-subconjunto tem, dentro do B-subconjunto ao qual ele pertence, um C-subconjunto a uma distância $4d_0^2$ e dois C-subconjuntos a uma distância $2d_0^2$.

Como estamos considerando um código sistemático (figura 4.7), os símbolos c_1 e c_2 que compõem o rótulo da primeira componente do par de sinais $\mathbf{s} = (s_1, s_2)$, que sai do CCM (figura 4.4), são iguais aos dois primeiros símbolos da sequência de informação, ou seja, $c_i = a_i$, $i = 1, 2$. Logo, a primeira componente do par $\mathbf{s} = (s_1, s_2)$ pode ser qualquer sinal do

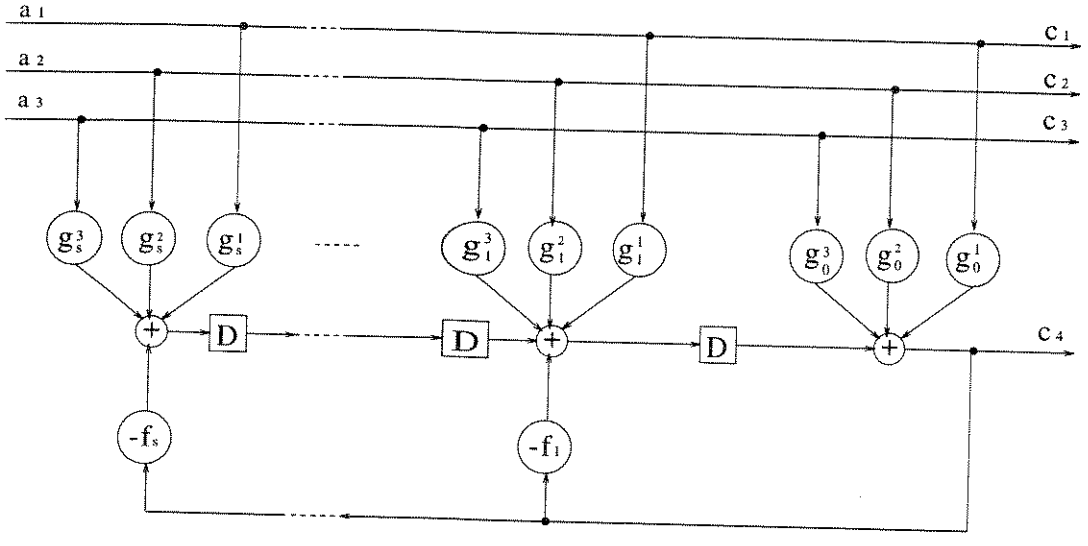

Figura 4.7: CCM de um código sistemático com realimentação, de taxa $R = \frac{3}{4}$.

$16 - QAM$. Então, como cada B-subconjunto será associado aos ramos que saem de cada estado do diagrama da treliça, cada B-subconjunto deve conter todos os pares ordenados

(s_1, s_2) de sinais do $16-QAM$, onde s_1 pode ser qualquer sinal do $16-QAM$. Mas da maneira como o particionamento está sendo feito, cada B-subconjunto contém apenas pares ordenados (s_1, s_2), onde ou $s_1 \in C_0 \cup C_2$ ou $s_1 \in C_1 \cup C_3$. Então é necessário fazer uma redistribuição dos C-subconjuntos nos quatro B-subconjuntos. Essa redistribuição é feita assim:

$$[A_0 A_0] = \begin{bmatrix} C_0C_0 & C_1C_1 & C_0C_1 & C_1C_2 \\ C_2C_2 & C_3C_3 & C_2C_3 & C_3C_0 \\ & & & \\ C_0C_2 & C_1C_3 & C_0C_3 & C_1C_0 \\ C_2C_0 & C_3C_1 & C_2C_1 & C_3C_2 \end{bmatrix} \tag{4.7}$$

Agora, cada novo B-subconjunto contém todos os pares ordenados (s_1, s_2) de sinais do $16-QAM$, onde s_1 pode ser qualquer sinal do $16-QAM$. Além disso, cada C-subconjunto tem, dentro do novo B-subconjunto ao qual ele pertence, um C-subconjunto a uma distância $4d_0^2$ e dois C-subconjuntos a uma distância $2d_0^2$.

No último nível da partição temos

$$[A_0 A_0] = \begin{bmatrix} D_0D_0/D_4D_4 & D_1D_1/D_5D_5 & D_0D_1/D_4D_5 & D_1D_2/D_5D_6 \\ D_0D_4/D_4D_0 & D_1D_5/D_5D_1 & D_0D_5/D_4D_1 & D_1D_6/D_5D_2 \\ & & & \\ D_2D_2/D_6D_6 & D_3D_3/D_7D_7 & D_2D_3/D_6D_7 & D_3D_4/D_7D_0 \\ D_2D_6/D_6D_2 & D_3D_7/D_7D_3 & D_2D_7/D_6D_3 & D_3D_0/D_7D_4 \\ & & & \\ D_0D_2/D_4D_6 & D_1D_3/D_5D_7 & D_0D_3/D_4D_7 & D_1D_4/D_5D_0 \\ D_0D_6/D_4D_2 & D_1D_7/D_5D_3 & D_0D_7/D_4D_3 & D_1D_0/D_5D_4 \\ & & & \\ D_2D_0/D_6D_4 & D_3D_5/D_7D_1 & D_2D_1/D_6D_5 & D_3D_6/D_7D_2 \\ D_2D_4/D_6D_0 & D_3D_1/D_7D_5 & D_2D_5/D_6D_1 & D_3D_2/D_7D_6 \end{bmatrix} \tag{4.8}$$

Em qualquer D-subconjunto a distância intra-conjunto é igual a $8d_0^2$, e cada D-subconjunto

tem, dentro do B-subconjunto ao qual ele pertence, um D-subconjunto a uma distância $8d_0^2$, seis D-subconjuntos a uma distância $4d_0^2$ e oito D-subconjuntos a uma distância $2d_0^2$.

Os subconjuntos obtidos com esse particionamento devem ser associados aos ramos da treliça, com o objetivo de maximizar a $D^2_{E_{\min}}$ entre sequências codificadas. As regras que definem a maneira de associar esses subconjuntos aos ramos da treliça, podem ser resumidas assim:

1. Todos os símbolos (sinais) devem ocorrer com a mesma frequência, dentro de um padrão de regularidade e simetria;

2. Transições paralelas devem ser associadas com pares ordenados de sinais pertencentes ou aos C-subconjuntos ou aos D-subconjuntos, assegurando uma $D^2_{E_{\min}}$ pelo menos igual a $4d_0^2$;

3. Transições que divergem de um mesmo estado devem ser associadas a pares ordenados de sinais de C-subconjuntos que pertencem a um mesmo B-subconjunto (equação 4.7), assegurando uma $D^2_{E_{\min}}$ de $2d_0^2$;

4. transições convergindo para um mesmo estado devem ser associadas com pares ordenados de sinais que pertencem a B-subconjuntos diferentes (equação 4.7), mas que estão a uma $D^2_{E_{\min}}$ de $2d_0^2$.

Este método de particionamento e as regras para associar os subconjuntos aos ramos da treliça podem ser estendidos para códigos usados em modulações $4^m - QAM$, onde $m > 2$. É importante observar que códigos usados em modulações $16-QAM$ podem ser adaptados para modulações $4^m - QAM$, $m > 2$, acrescentando-se transições paralelas. De fato, suponhamos que a matriz geradora de um código $\mathcal{C}$ usado em uma modulação $16 - QAM$ seja

$$\mathbf{G}(D) = \begin{bmatrix} 1 & 0 & 0 & g^{(1)}(D)/f(D) \\ 0 & 1 & 0 & g^{(2)}(D)/f(D) \\ 0 & 0 & 1 & g^{(3)}(D)/f(D) \end{bmatrix} \tag{4.9}$$

onde

$$g^{(i)}(D) = g_s^{(i)} D^s + \cdots + g_1^{(i)} D + g_0^{(i)} \in \mathbb{Z}_4 [D]$$
$$f(D) = f_s D^s + \cdots + f_1 D + 1 \in \mathbb{Z}_4 [D] \tag{4.10}$$

são os polinômios responsáveis pelas conexões de alimentação e realimentação, respectivamente (figura 4.7). A partir do código $\mathcal{C}$ podemos obter outros códigos adequados para modulações $4^m - QAM$, $m > 2$. Por exemplo, se quisermos um código adequado para a modulação $64 - QAM$ adaptando o código $\mathcal{C}$, obtemos o código $\mathcal{C}'$ cuja matriz geradora é

$$\mathbf{G}'(D) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & g^{(1)}(D)/f(D) \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & g^{(2)}(D)/f(D) \\ 0 & 0 & 0 & 1 & 0 & g^{(3)}(D)/f(D) \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.11}$$

Adaptando o código $\mathcal{C}$ para uma modulação $256 - QAM$, obtemos o código $\mathcal{C}''$ cuja matriz geradora é

$$\mathbf{G}''(D) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & g^{(1)}(D)/f(D) \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & g^{(2)}(D)/f(D) \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & g^{(3)}(D)/f(D) \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.12}$$

O código $\mathcal{C}$ fica bem determinado pelos coeficientes dos polinômios $g^{(i)}(D)$ e $f(D)$. Vamos usar a notação

$$(g_s^3 g_s^2 g_s^1) \cdots (g_1^3 g_1^2 g_1^1)\,(g_0^3 g_0^2 g_0^1)/(f_s \ldots f_1 1) \tag{4.13}$$

para representar esses polinômios. Com essa notação, os códigos $\mathcal{C}'$ e $\mathcal{C}''$ são determinados, respectivamente, por

$$(0 g_s^3 g_s^2 0 g_s^1) \cdots (0 g_1^3 g_1^2 0 g_1^1)\,(0 g_0^3 g_0^2 0 g_0^1)/(f_s \ldots f_1 1) \tag{4.14}$$

e

$$(00g_s^3g_s^200g_s^1)\cdots(00g_1^3g_1^200g_1^1)\,(00g_0^3g_0^200g_0^1)/(f_s\ldots f_1 1). \tag{4.15}$$

De modo análogo, podemos adaptar o código $\mathcal{C}$ para modulações $4^m - QAM$, $m > 4$. A figura 4.8 ilustra o CCM do código $\mathcal{C}$ adaptado para uma modulação $64 - QAM$.

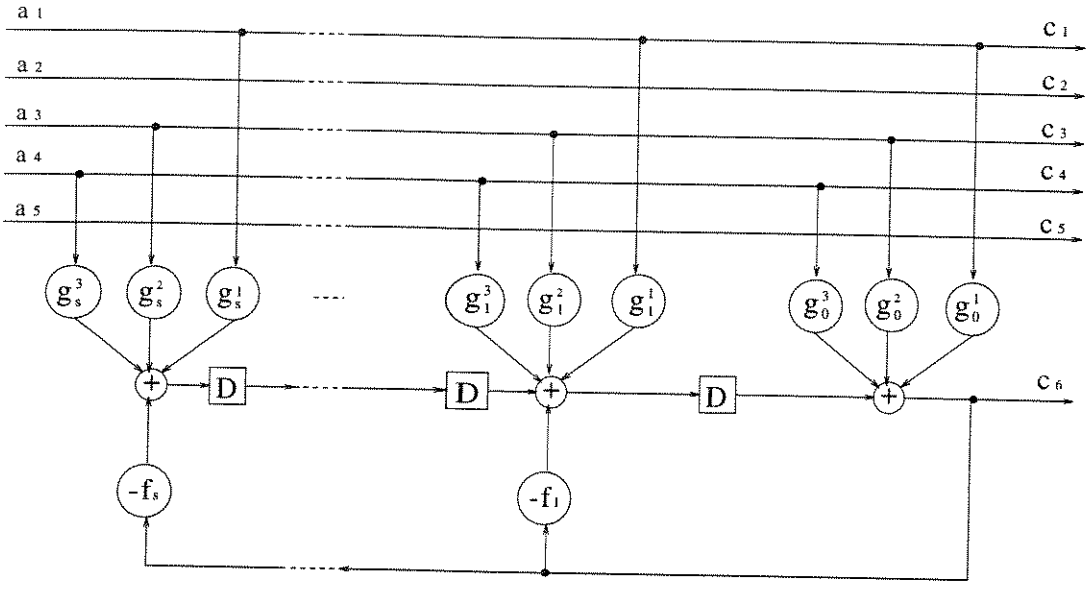

Figura 4.8: CCM de um código sistemático com realimentação, de taxa $R = \frac{5}{6}$.

Na tabela 4.1 listamos alguns códigos convolucionais sobre $\mathbb{Z}_4$, adequados para modulações $16 - QAM$. Os valores de g_∞ foram calculados levando-se em conta que a modulação $8 - QAM$ não codificada é o esquema de referência, com distância Euclidiana quadrada mínima igual a 2. Na primeira coluna, N representa o número de estados.

4.6 Conclusão

Neste capítulo apresentamos um método indutivo de rotulamento dos pontos de um espaço de sinais $4^m - QAM$ retangular, $m \geq 2$, que permite fazer rotulamentos transparentes de um espaço $4^m - QAM$ a partir de rotulamentos de $4 - QAM$, e analisamos o problema

N	Polinômios geradores/realimentador	$RI\ [rad]$	D^2_{free}	$g_\infty\ [dB]$	N_{free}
2	(030) (020) / (0)	$\frac{\pi}{2}$	$4,0$	$3,01$	$19,1875$
4	(220) (030) / (2)	$\frac{\pi}{2}$	$4,0$	$3,01$	$5,937$
8	(022) (231) (013) / (03)	$\frac{\pi}{2}$	$4,0$	$3,01$	$0,75$
8	(010) (232) (020) / (21)	π	$5,0$	$3,98$	$5,234$
16	(010) (232) (010) / (22)	$\frac{\pi}{2}$	$6,0$	$4,77$	$9,796$
32	(020) (212) (010) (020) / (003)	$\frac{\pi}{2}$	$6,0$	$4,77$	$2,469$
32	(020) (222) (010) (010) / (313)	π	$6,0$	$4,77$	$0,801$
64	(012) (200) (232) (030) / (132)	$\frac{\pi}{2}$	$8,0$	$6,02$	$21,743$

Tabela 4.1: Códigos convolucionais sobre $\mathbb{Z}_4$ adequados para $16-QAM$.

de rotação de fase da portadora. Além disso, apresentamos uma técnica de codificação de modulações $4^m - QAM$, $m \geq 2$, que combina um rotulamento transparente do espaço de sinais, com um método de particionamento de conjuntos.

Capítulo 5

Modulações QAM Codificadas com Códigos de Bloco Multiníveis

5.1 Introdução

O uso de códigos de bloco multiníveis em modulações QAM foi introduzido por Baldini em [2]. Ele considerou um espaço de sinais $16-QAM$ com um rotulamento transparente e utilizou dois códigos de bloco multiníveis sobre $\mathbb{Z}_4$, cada um atuando, de modo independente, no símbolo correspondente dos rótulos. Aliado ao rotulamento dos pontos, foi utilizado um método de particionamento de conjunto que proporcionou ganhos de codificação assintóticos de até $6,0\,dB$, sobre o esquema $8-QAM$ não codificado. Atualmente alguns sistemas de comunicações digitais já utilizam esquemas de modulações $128-QAM$ e $256-QAM$, por exemplo, comunicação por rádio digital. Neste capítulo vamos estender para modulações 4^m-QAM, $m \geq 2$, a técnica de codificação proposta por Baldini em [2]. Apresentaremos os processos de codificação e de decodificação.

No processo de codificação usaremos m códigos de bloco lineares $\mathcal{C}_1, \ldots, \mathcal{C}_m$, sobre $\mathbb{Z}_4$, cada código atuando de forma independente no símbolo correspondente dos rótulos. Todos os códigos são da forma $\mathcal{C}_i : (n, k_i)$, ou seja, têm o mesmo comprimento n e o código $\mathcal{C}_i$ tem taxa de codificação $R_i = \frac{k_i}{n}$, $i = 1, \ldots, n$. Nestas condições, a taxa de codificação total

é definida como sendo $R_t \triangleq \frac{k_1+\cdots+k_m}{mn} = \frac{R_1+\cdots+R_m}{m}$. Então, se considerarmos as modulações 4^m-QAM e $\frac{4^m}{2}-QAM$ como esquemas codificado e não codificado (esquema de referência), respectivamente, para que esses esquemas tenham as mesmas taxas de transmissão de informação, sem necessidade de expansão de faixa, os códigos $\mathcal{C}_1, \ldots, \mathcal{C}_m$ devem ser escolhidos de modo que $R_t = \frac{2m-1}{2m}$.

5.2 Processo de Codificação: Extensão do Processo de Sayegh

Vamos descrever o processo de codificação de uma modulação 4^m-QAM codificada com m códigos de bloco multiníveis $\mathcal{C}_1, \ldots, \mathcal{C}_m$, sobre $\mathbb{Z}_4$, onde $\mathcal{C}_i$ é um código (n, k_i), $i = 1, \ldots, m$. Cada par de bits, saindo de uma fonte binária, é mapeado em um dos símbolos de $\mathbb{Z}_4$. Geralmente é usado o mapeamento de Gray pois isto minimiza o número de bits errados quando um símbolo é decodificado erradamente. Cada sequência formada por $2(k_1 + \cdots + k_m)$ bits que sai dessa fonte binária é mapeada numa sequência de informação $\mathbf{a} = (a_1, \ldots, a_{k_1+\cdots+k_m})$, cujos símbolos pertencem a $\mathbb{Z}_4$. Essa sequência $\mathbf{a}$ é dividida em m subsequências $\mathbf{a}_1, \ldots, \mathbf{a}_m$, onde

$$\mathbf{a}_1 = (a_1, \ldots, a_{k_1}), \ \mathbf{a}_2 = (a_{k_1+1}, \ldots, a_{k_1+k_2}), \ldots, \mathbf{a}_m = \left(a_{k_1+\cdots+k_{m-1}+1}, \ldots, a_{k_1+\cdots+k_m}\right).$$

Para cada $j = 1, \ldots, m$ a subsequência $\mathbf{a}_j$ é a sequência de informação do código $\mathcal{C}_j$. Essas sequências de informação $\mathbf{a}_j$ são enviadas para os seus respectivos codificadores, onde são codificadas.

Denotemos por $\mathbf{C}_j$ o codificador do código $\mathcal{C}_j$ e seja $\mathbf{c}^j \triangleq \mathbf{C}_j(\mathbf{a}_j)$ a palavra código de $\mathcal{C}_j$ resultante da sequência de informação $\mathbf{a}_j$. Temos que $\mathbf{c}^j = \left(c_1^j, \ldots, c_n^j\right)$, onde $c_i^j \in \mathbb{Z}_4$, $j = 1, \ldots, m$ e $i = 1, \ldots, n$. Seja $\mathbf{c}$ a matriz $m \times n$ cujas linhas são as palavras código $\mathbf{c}^j$,

$j = 1, \ldots, m$, isto é,

$$\mathbf{c} = \begin{bmatrix} c_1^1 & c_2^1 & \cdots & c_n^1 \\ \vdots & \vdots & & \vdots \\ c_1^m & c_2^m & \cdots & c_n^m \end{bmatrix}. \tag{5.1}$$

Podemos considerar o conjunto de códigos $\mathcal{C}_1, \ldots, \mathcal{C}_m$ como um único código $\mathcal{C} : (mn, k)$, onde $k = k_1 + \cdots + k_m$. Assim, a matriz $\mathbf{c}$ é a palavra código de $\mathcal{C}$ resultante da sequência de informação $\mathbf{a}$ e $R_t = \frac{k}{mn}$ é a taxa de codificação do código $\mathcal{C}$. Cada coluna da matriz $\mathbf{c}$ pode ser identificada com um sinal de $4^m - QAM$, mais precisamente, a i-ésima coluna de $\mathbf{c}$ pode ser identificada com o sinal de $4^m - QAM$ cujo rótulo é $c_i^1 c_i^2 \cdots c_i^m$. O mapeamento de $\mathbf{c}$, no espaço de sinais, é feito mapeando cada coluna no sinal correspondente. Dessa maneira, o mapeamento de $\mathbf{c}$ resulta numa sequência de sinais $\mathbf{s} = (s_1, \ldots, s_n)$, sendo que cada código $\mathcal{C}_j$ proteje o j-ésimo símbolo, da esquerda para a direita, do rótulo de cada sinal dessa sequência. Essa sequência $\mathbf{s}$ é modulada e enviada através do canal. A figura 5.1 ilustra esse processo. Como o conjunto de códigos $\mathcal{C}_1, \ldots, \mathcal{C}_m$ é considerado como um único código $\mathcal{C}$, o conjunto formado pelos codificadores $\mathbf{C}_1, \ldots, \mathbf{C}_m$ deve ser considerado como um único codificador $\mathbf{C}$, onde todos os símbolos de informação são codificados. Também podemos considerar o conjunto formado pela fonte binária e o $\mathbb{Z}_4-$mapeador como um único bloco, chamado fonte multinível.

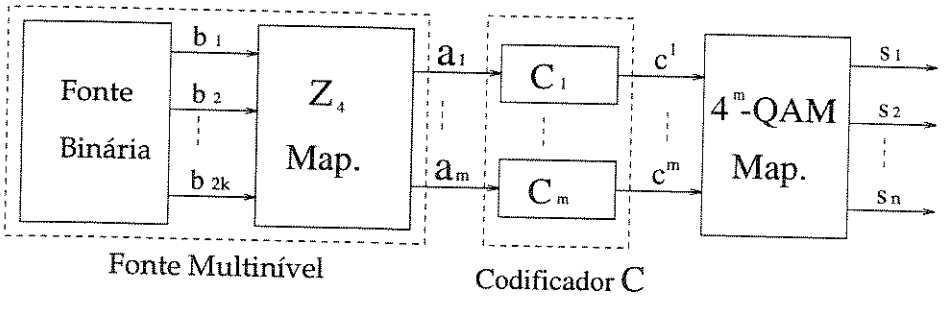

Figura 5.1: Processo de codificação de uma modulação $4^m - QAM$, utilizando m códigos de bloco de comprimento n.

Nosso objetivo agora é maximizar as distâncias Euclidianas quadradas mínimas de cada

código $\mathcal{C}_j$. Com esse objetivo, vamos fazer um particionamento do conjunto de pontos do espaço $4^m - QAM$ que nos permite encontrar códigos com boas distâncias Euclidianas quadradas mínimas. Esse particionamento depende fortemente do rotulamento dos pontos do espaço. Daí se vê a importância de se adotar um rotulamento adequado.

Se todos os códigos $\mathcal{C}_1, \ldots, \mathcal{C}_m$ são transparentes e o rotulamento do espaço também é transparente, os problemas causados por rotações de fase da portadora podem ser resolvidos com um codificador/decodificador diferencial .

5.3 Particionamento do Conjunto

Fazemos um particionamento do conjunto de pontos do espaço de sinais $4^m - QAM$ semelhante ao de Ungerboeck [28]. No nível 1 da partição o conjunto é dividido em quatro subconjuntos, cada um com 4^{m-1} pontos. No nível 2 , cada um desses subconjuntos é dividido em quatro subconjuntos, cada um com 4^{m-2} pontos. Prosseguimos dessa maneira até o nível $m-1$ onde obtemos 4^{m-1} subconjuntos, cada um com 4 pontos.

Supondo que o rotulamento do espaço $4^m - QAM$ é transparente (seção 4.2), esse particionamento é realizado da seguinte maneira: No nível 1 da partição, em cada subconjunto o símbolo mais à esquerda é o mesmo para todos os rótulos. No nível 2, em cada subconjunto os dois símbolos mais à esquerda são os mesmos para todos os rótulos. De um modo geral, no nível j da partição, $j = 1, \ldots, m-1$, em cada subconjunto os j símbolos mais à esquerda são os mesmos para todos os rótulos. Na figura 5.2 mostramos o particionamento de $64 - QAM$ com o rotulamento mostrado na figura 4.3. Os ternos ordenados, abaixo de cada subconjunto do particionamento, indicam como são os rótulos dos pontos do respectivo subconjunto.

Observação 5.1 *Suponhamos que a distância Euclidiana quadrada mínima ($D^2_{E_{\min}}$) do espaço de sinais $4^m - QAM$ seja igual a D_0^2. Então:*

1. Os subconjuntos de um mesmo nível da partição têm a mesma distância Euclidiana quadrada mínima;

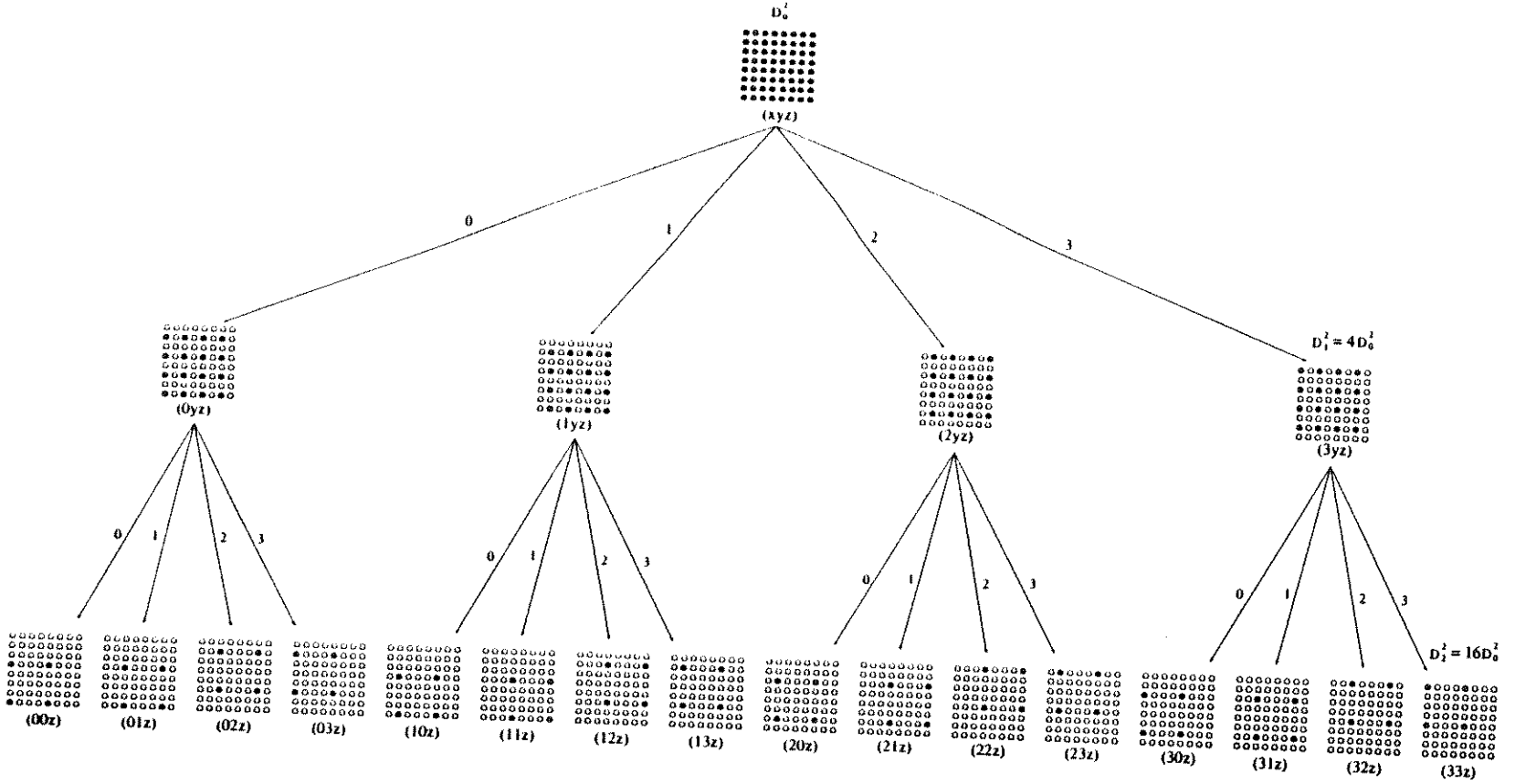

Figura 5.2: Particionamento do conjunto de pontos do espaço 64 – QAM.

2. Se denotarmos por D_j^2 a distância Euclidiana quadrada mínima dos subconjuntos do nível j da partição, $j = 1, \ldots, m-1$, temos que $D_j^2 = 4^j D_0^2$.

5.3.1 Distância Euclidiana Quadrada Mínima do Código $\mathcal{C}$

Sendo cada $\mathcal{C}_j$ um código de bloco linear sobre $\mathbb{Z}_4$ então $\mathcal{C}$ também o é. Assim, a distância Euclidiana quadrada mínima de $\mathcal{C}$ é igual ao peso Euclidiano quadrado mínimo $(w_{E_{\min}}^2)$ das sequências codificadas de sinais. Seja $\mathbf{s}$ uma sequência codificada de sinais não nula, e para cada $j = 1, 2, \ldots, m$, seja d_j a distância de Hamming mínima do código $\mathcal{C}_j$. Se o rótulo de algum sinal de $\mathbf{s}$ tem o primeiro símbolo não nulo então os rótulos de pelo menos d_1 sinais de $\mathbf{s}$ têm o primeiro símbolo não nulo, pois $\mathcal{C}_1$ atua no primeiro símbolo dos rótulos e d_1 é a distância de Hamming mínima do código $\mathcal{C}_1$. Então

$$w_E^2(\mathbf{s}) \geq D_{E_{\min}}^2(\mathcal{C}_1) \geq d_1 D_0^2. \tag{5.2}$$

Se os rótulos de todos os sinais de $\mathbf{s}$ têm o primeiro símbolo nulo e o rótulo de algum sinal de $\mathbf{s}$ tem o segundo símbolo não nulo, então os rótulos de pelo menos d_2 sinais de $\mathbf{s}$ têm o segundo símbolo não nulo. Como todos os rótulos dos sinais de $\mathbf{s}$ têm o primeiro símbolo nulo então todos os sinais de $\mathbf{s}$ estão num mesmo subconjunto do nível 1 da partição. Logo, a distância Euclidiana quadrada entre quaisquer dois deles é pelo menos D_1^2. Então

$$w_E^2(\mathbf{s}) \geq D_{E_{\min}}^2(\mathcal{C}_2) \geq d_2 D_1^2 = 4 d_2 D_0^2. \tag{5.3}$$

Prosseguindo dessa forma, vemos que se os $j-1$ primeiros símbolos dos rótulos de todos os sinais de $\mathbf{s}$ são nulos e algum sinal de $\mathbf{s}$ tem rótulo cujo j-ésimo símbolo é não nulo, então pelo menos d_j sinais de $\mathbf{s}$ têm rótulo cujo j-ésimo símbolo é não nulo e, além disso, todos os sinais de $\mathbf{s}$ estão num mesmo subconjunto do nível $j-1$ da partição e, portanto, a distância Euclidiana quadrada entre quaisquer dois deles é pelo menos D_j^2. Assim,

$$w_E^2(\mathbf{s}) \geq D_{E_{\min}}^2(\mathcal{C}_j) \geq d_j D_j^2 = 4^{j-1} d_j D_0^2, \quad j = 1, 2, \ldots, m. \tag{5.4}$$

Daí segue que

$$w_E^2(\mathbf{s}) \geq \min\left\{ D_{E_{\min}}^2(\mathcal{C}_j), \quad j = 1, 2, \ldots, m \right\}. \tag{5.5}$$

É fácil ver que a igualdade em 5.5 sempre ocorre. Então a distância Euclidiana quadrada mínima do código $\mathcal{C}$ é dada por

$$D^2_{E_{\min}}(\mathcal{C}) = \min\left\{D^2_{E_{\min}}(\mathcal{C}_j),\, j = 1, 2, \ldots, m\right\}. \tag{5.6}$$

5.4 Ganho de Codificação

Um parâmetro para se avaliar a performance do conjunto de códigos $\mathcal{C}_1, \ldots, \mathcal{C}_m$ é o ganho de codificação assintótico g_∞ do esquema de modulação codificada em relação ao esquema de referência (não codificado). Temos que (Seção 3.3.4)

$$g_\infty = 10\log_{10}\left(\frac{\log M_c}{\log M_{nc}} R_t \frac{D^2_{E_{\min}}(\mathcal{C})}{D^2_{Enc}}\right)[dB] \tag{5.7}$$

onde M_c e M_{nc} são o número de sinais dos esquemas codificado e não codificado, respectivamente, e D^2_{Enc} é a distância Euclidiana quadrada mínima do esquema não codificado.

Se $j \geq 3$ segue de (5.4) que $D^2_{E_{\min}}(\mathcal{C}_j) \geq 16D_0^2$ qualquer que seja o código $\mathcal{C}_j$, até mesmo se $\mathcal{C}_j$ é um código sem redundância, isto é, $\mathcal{C}_j : (n, n)$. Se considerarmos $\mathcal{C}_2$ como sendo um código de verificação de paridade simples, ou seja, $\mathcal{C}_2 : (n, n-1)$, temos que $D^2_{E_{\min}}(\mathcal{C}_2) = 8D_0^2$. Então, neste caso, segue de 5.6 que

$$D^2_{E_{\min}}(\mathcal{C}) = \min\left\{D^2_{E_{\min}}(\mathcal{C}_1), 8D_0^2\right\}. \tag{5.8}$$

Então, considerando esses códigos, e tomando um código $\mathcal{C}_1 : \left(n, \frac{n}{2}+1\right)$ (n deve ser par), temos que $R_t = \frac{2m-1}{2m}$. Se além disso, considerarmos $4^m - QAM$ e $\frac{4^m}{2} - QAM$ como sendo os esquemas codificado e não codificado, respectivamente, a equação (5.7) assume a forma

$$g_\infty = 10\log_{10}\left(\frac{\min\left\{D^2_{E_{\min}}(\mathcal{C}_1), 8D_0^2\right\}}{2D_0^2}\right)[dB]. \tag{5.9}$$

Daí vemos que para obtermos g_∞ entre 3 e 6 $[dB]$, onde os esquemas codificado e não codificado têm as mesmas taxas de transmissão de informação, basta que se encontre um código $\mathcal{C}_1 : \left(n, \frac{n}{2}+1\right)$ com $D^2_{E_{\min}}$ entre $4D_0^2$ e $8D_0^2$.

5.4.1 Exemplos de Códigos Multiníveis Transparentes sobre $\mathbb{Z}_4$

Vamos considerar uma classe de códigos de bloco multiníveis (n, k) sobre $\mathbb{Z}_4$, cujas matrizes geradoras são da forma

$$\mathbf{G} = \begin{bmatrix} g_1 & g_2 & \cdots & g_{n-1} & g_n \\ g_n & g_1 & \cdots & g_{n-2} & g_{n-1} \\ \vdots & \vdots & & \vdots & \vdots \\ g_{n-k+3} & g_{n-k+4} & \cdots & g_{n-k+1} & g_{n-k+2} \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix}. \tag{5.10}$$

Cada linha de $\mathbf{G}$, exceto a primeira e a última, é o deslocamento cíclico de um símbolo, para a direita, da linha anterior. Assim, a matriz $\mathbf{G}$ fica completamente determinada pela primeira linha. A importância dessa propriedade se revela quando procuramos novos códigos, via busca computacional, pois a matriz $\mathbf{G}$ se reduz a um vetor de n coordenadas. No processo de busca computacional devemos observar que $\mathbf{G}$ deve ser uma matriz de posto k. A última linha de $\mathbf{G}$, com todas as entradas iguais a 1, garante que esses códigos são transparentes.

Na tabela 5.1 apresentamos alguns códigos dessa classe, adequados para modulações $4^m - QAM$. Nessa tabela temos que:

1. Na terceira coluna, R é a taxa de codificação do código gerado pela matriz $\mathbf{G}$.

2. Os valores de $D^2_{E_{\min}}$ foram calculados considerando que $D^2_0 = 1$. Se este não for o caso, o valor de $D^2_{E_{\min}}$ deve ser multiplicado pelo valor de D^2_0.

3. Na sexta coluna, N_v é o número de vizinhos mais próximos de cada palavra código. Pelo Teorema 2.16, N_v é igual ao número de palavras código de peso Euclidiano quadrado mínimo .

4. g_∞ é o ganho de codificação assintótico obtido com a modulação $4^m - QAM$ codificada com o código $\mathcal{C}$, sobre o esquema de referência $\frac{4^m}{2} - QAM$, sendo $\mathcal{C}$ o código formado pelo conjunto de códigos $\mathcal{C}_1, \ldots, \mathcal{C}_m$, onde:

(a) Para $n = 4$, $\mathcal{C}_1$ é o código $(4, 2)$ gerado pela matriz $\mathbf{G}$ e $\mathcal{C}_j : (4, 4)$ para $2 \leq j \leq m$.

(b) Nos demais casos, $\mathcal{C}_1$ é o código $\left(n, \frac{n}{2} + 1\right)$ gerado pela matriz $\mathbf{G}$, $\mathcal{C}_2$ é o código de verificação de paridade simples $(n, n-1)$ e $\mathcal{C}_j$ é um código sem redundância (n, n), para $3 \leq j \leq m$.

5. Em todos os casos temos que $R_t = \frac{2m-1}{2m}$ e, portanto, os esquemas codificado e não codificado têm as mesmas taxas de transmissão de informação. Além disso, para que o esquema de referência tenha a mesma energia média do codificado, consideramos $D^2_{E_{nc}} = 2$.

(n, k)	Primeira Linha da Matriz Geradora $\mathbf{G}$	R	$D^2_{E_{\min}}$	$g_\infty\ [dB]$	N_v
$(4, 2)$	3210	$\frac{2}{4}$	4	3,01	14
$(12, 7)$	313322131103	$\frac{7}{12}$	5	3,98	48
$(14, 8)$	031010O1220132	$\frac{8}{14}$	6	4,77	196
$(16, 9)$	3112302133333132	$\frac{9}{16}$	6	4,77	120
$(20, 11)$	3312322301OOOOOOOOO	$\frac{11}{20}$	8	6,02	620

Tabela 5.1: Códigos de bloco transparentes sobre $\mathbb{Z}_4$ adequados para $4^m - QAM$.

Um código $\mathcal{C}$ adequado para $4^m - QAM$ é transparente se, e somente se, a palavra cujos símbolos são todos iguais a 1 é uma palavra código de $\mathcal{C}$. Como as palavras código de $\mathcal{C}$ são matrizes cujas linhas são palavras código de $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m$, respectivamente, segue que $\mathcal{C}$ é transparente se, e somente se, cada código $\mathcal{C}_j$ é transparente. Considerando os códigos da tabela 5.1, se $j \neq 2$, os códigos $\mathcal{C}_j$ são todos transparentes pois já sabemos que os códigos $\mathcal{C}_1$ são transparentes e, para $j \geq 3$, $\mathcal{C}_j$ é um código sem redundância, logo transparente. Então, neste caso, $\mathcal{C}$ é transparente se, e somente se, $\mathcal{C}_2$ é transparente. No caso $n = 4$, $\mathcal{C}_2$ é um código $(4, 4)$, logo é transparente. Nos demais casos, como $\mathcal{C}_2$ é um código de verificação de

paridade simples $(n, n-1)$, temos que $\mathcal{C}_2$ é transparente se, e somente se, n é um múltiplo de 4 pois $\mathbf{v} = (1, \ldots, 1)$ é uma palavra código de $\mathcal{C}_2$ se, e somente se,

$$1 \oplus \cdots \oplus 1 = n.1 = n \equiv 0 \,(\mathrm{mod}\, 4)\,.$$

Assim, na tabela 5.1, apenas no caso $n = 14$, $\mathcal{C}$ não é transparente.

Ganhos acima de $6,0\,dB$ podem ser obtidos mas, para isso, é necessário diminuir a taxa de codificação do código $\mathcal{C}_2$ e aumentar a do código $\mathcal{C}_1$, para que os esquemas codificado e não codificado tenham as mesmas taxas de transmissão de informação. Se estivermos considerando as modulações $4^m - QAM$ e $\frac{4^m}{2} - QAM$ como esquemas codificado e não codificado, respectivamente, então devemos tomar os códigos $\mathcal{C}_1$ e $\mathcal{C}_2$ da forma $\mathcal{C}_1 : \left(n, \frac{n}{2} + l\right)$ e $\mathcal{C}_2 : (n, n-l)$. Os demais códigos são códigos sem redundância pois a distância Euclidiana quadrada mínima nos subconjuntos do segundo nível da partição, já é igual a $16\,D_0^2$, onde D_0^2 é a distância Euclidiana quadrada mínima de $4^m - QAM$. Daí nota-se que para se obter ganhos acima de $6,0\,dB$ devemos considerar códigos cujos comprimentos são bem maiores do que aqueles da tabela 5.1. O esforço computacional na busca desses códigos cresce exponencialmente quando aumentamos os seus comprimentos. Por isso, não foram feitas buscas de códigos longos.

Encontramos alguns códigos transparentes adequados para modulações $4^m - QAM$, onde $R_t \neq \frac{2m-1}{2m}$, mas que apresentam ganhos de codificação assintóticos significativos, em relação ao esquema $\frac{4^m}{2} - QAM$ não codificado. Na tabela 5.2 apresentamos alguns desses códigos, adequados para $64 - QAM$. Os parâmetros dos códigos apresentados na primeira coluna da matriz 5.2 são $\left(n, k, D^2_{E_{\min}}\right)$. Os códigos $\mathcal{C}_3$ foram omitidos porque, em todos os casos, eles são códigos sem redundância.

5.5 Decodificação

Vamos apresentar um processo de decodificação para o código $\mathcal{C}$. Antes de descrevermos esse processo, lembramos que:

Códigos	Primeira Linha da Matriz Geradora de $\mathcal{C}_1$	R_t	$D^2_{E_{\min}}(\mathcal{C})$	$g_\infty[dB]$ sobre $32-QAM$	N_v
$\mathcal{C}_1$:(6,2,5) $\mathcal{C}_2$:(6,5,8)	321100	$\frac{13}{18}$	5	3,36	4
$\mathcal{C}_1$:(8,4,6) $\mathcal{C}_2$:(8,7,8)	22111230	$\frac{19}{24}$	6	4,55	112
$\mathcal{C}_1$:(10,5,6) $\mathcal{C}_2$:(10,9,8)	3030310233	$\frac{24}{30}$	6	4,60	90
$\mathcal{C}_1$:(16,8,8) $\mathcal{C}_2$:(16,15,8)	2102221130322102	$\frac{39}{48}$	8	5,91	476
$\mathcal{C}_1$:(8,5,4) $\mathcal{C}_2$:(8,8,4)	12120002	$\frac{21}{24}$	4	3,22	64
$\mathcal{C}_1$:(16,10,6) $\mathcal{C}_2$:(16,15,8)	1302121000000000	$\frac{41}{48}$	6	4,88	496
$\mathcal{C}_1$:(10,7,4) $\mathcal{C}_2$:(10,10,4)	0113231012	$\frac{27}{30}$	4	3,34	173
$\mathcal{C}_1$:(20,13,6) $\mathcal{C}_2$:(20,19,8)	12112010000000000000	$\frac{52}{60}$	6	4,94	1126
$\mathcal{C}_1$:(12,9,4) $\mathcal{C}_2$:(12,12,4)	121233012331	$\frac{33}{36}$	4	3,42	366
$\mathcal{C}_1$:(20,14,5) $\mathcal{C}_2$:(20,19,8)	11301010000000000000	$\frac{53}{60}$	5	4,23	180
$\mathcal{C}_1$:(14,11,4) $\mathcal{C}_2$:(14,14,4)	13131002112302	$\frac{39}{42}$	4	3,48	707

Tabela 5.2: Códigos de bloco sobre $\mathbb{Z}_4$ adequados para $64-QAM$.

1. Estamos considerando m códigos $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m$ sobre $\mathbb{Z}_4$, onde $\mathcal{C}_j : (n, k_j)$. Estamos chamando o conjunto dos códigos $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m$ de código $\mathcal{C}$, e temos que $\mathcal{C} : (mn, k)$, onde $k = k_1 + k_2 + \cdots + k_m$.

2. Cada sequência de informação $\mathbf{a} = (a_1, \ldots, a_{k_1+\cdots+k_m})$ é desmembrada em m subsequências

$$\mathbf{a}_1 = (a_1, \ldots, a_{k_1}),\ \mathbf{a}_2 = (a_{k_1+1}, \ldots, a_{k_1+k_2}), \ldots, \mathbf{a}_m = \left(a_{k_1+\cdots+k_{m-1}+1}, \ldots, a_{k_1+\cdots+k_m}\right),$$

que são as sequências de informação dos códigos $\mathcal{C}_1$, $\mathcal{C}_2, \ldots, \mathcal{C}_m$, respectivamente, que depois de codificadas, resultam nas respectivas palavras código

$$\mathbf{c}^1 = \left(c_1^1, \ldots, c_n^1\right), \mathbf{c}^2 = \left(c_1^2, \ldots, c_n^2\right), \ldots, \mathbf{c}^m = \left(c_1^m, \ldots, c_n^m\right).$$

Essas palavras código são dispostas em uma matriz $m \times n$

$$\mathbf{c} = \begin{bmatrix} c_1^1 & c_2^1 & \cdots & c_n^1 \\ c_1^2 & c_2^2 & & c_n^2 \\ \vdots & \vdots & & \vdots \\ c_1^m & c_2^m & \cdots & c_n^m \end{bmatrix} \tag{5.11}$$

que é a palavra código de $\mathcal{C}$.

5.5.1 Processo de Decodificação: Extensão do Processo de Sayegh

Sayegh[27] propôs um processo de decodificação adequado para um código $\mathcal{C}$ formado por um conjunto de códigos de bloco binários. Esse processo, que se baseia num particionamento do conjunto de palavras código de $\mathcal{C}$, pode ser estendido para códigos de bloco multiníveis. Nesta seção, vamos estendê-lo para um código $\mathcal{C}$ formado por um conjunto de códigos de bloco multiníveis sobre $\mathbb{Z}_4$, como acabamos de descrever. Neste caso o processo também se baseia em um particionamento do conjunto de palavras código de $\mathcal{C}$. Este conjunto é constituido por 4^k palavras e o particionamento é feito em m níveis. No nível 1, dividimos o conjunto em 4^{k_1} subconjuntos e, no nível 2, cada subconjunto do nível 1 é dividido em 4^{k_2} subconjuntos. Prosseguimos dessa maneira até o nível m, onde cada subconjunto do nível $m-1$ é dividido em 4^{k_m} subconjuntos. Esse particionamento é realizado da seguinte maneira: No nível 1, cada subconjunto é constituido pelas palavras código de $\mathcal{C}$ que coincidem na primeira linha da matriz $\mathbf{c}$. No nível 2, cada subconjunto é constituido pelas palavras código de $\mathcal{C}$ que coincidem nas duas primeiras linhas da matriz $\mathbf{c}$. De um modo geral, no nível j, $j = 1, 2, \ldots, m$, cada subconjunto é constituido pelas palavras código de $\mathcal{C}$ que coincidem nas j primeiras linhas da matriz $\mathbf{c}$.

O processo de decodificação é feito em m etapas da seguinte maneira: Na primeira, por meio de um processo de decisão, a palavra recebida é associada a um dos 4^{k_1} subconjuntos do nível 1. Na segunda etapa, supondo que a decisão anterior está correta, a palavra recebida é associada a um dos 4^{k_2} subconjuntos do subconjunto do nível 1 da partição, ao qual a palavra foi associada na etapa anterior. De um modo geral, na j-ésima etapa, $j = 2, \ldots, m$,

supondo que todas as decisões anteriores estão corretas, a palavra recebida é associada a um dos 4^{k_j} subconjuntos do subconjunto do nível $j-1$ da partição, ao qual a palavra foi associada na etapa anterior. Dessa forma, na m-ésima etapa, a palavra recebida é associada a uma palavra código de $\mathcal{C}$. Caso todas as decisões tenham sido corretas, essa é a palavra código que foi enviada. Então essa palavra código é decodificada.

Processo de Decisão

No processo de decodificação que acabamos de descrever, em cada uma das m etapas é necessário fazer uma decisão, que consiste na escolha de um dos subconjuntos. Vamos apresentar uma maneira de fazer essa decisão.

No processo de codificação (seção 5.2) cada sinal s_i pode ser representado por um par ordenado de números reais, que são as coordenadas cartesianas do ponto que representa o sinal s_i. Geralmente as coordenadas desses pontos são consideradas como sendo números inteiros. O modulador associa esse par ordenado a uma forma de onda que é transmitida. Assim, no receptor, a demodulação de um sinal recebido é um par ordenado de números reais, que pode ser diferente das coordenadas do sinal modulado, já que a ação do ruído de canal pode alterar a forma de onda que foi enviada. Portanto, quando uma sequência de sinais $\mathbf{s} = (s_1, \ldots, s_n)$ é enviada através do canal, na saída do demodulador temos uma sequência de pares ordenados reais $\mathbf{r} = ((x_1^1, x_1^2), \ldots, (x_n^1, x_n^2))$, que chamaremos de **sequência recebida**. Cada par ordenado (x_i^1, x_i^2) é mapeado em um sinal de $4^m - QAM$ e o rótulo $v_i^1 v_i^2 \cdots v_i^m$ desse sinal é então associado ao par (x_i^1, x_i^2). Obtemos assim uma matriz

$$\mathbf{v} = \begin{bmatrix} v_1^1 & v_2^1 & \cdots & v_n^1 \\ v_1^2 & v_2^2 & \cdots & v_n^2 \\ \vdots & \vdots & & \vdots \\ v_1^m & v_2^m & \cdots & v_n^m \end{bmatrix} \tag{5.12}$$

onde a i-ésima coluna de $\mathbf{v}$ é formada pelos símbolos do rótulo associado ao par (x_i^1, x_i^2), $i = 1, 2, \ldots, n$. Chamaremos essa matriz $\mathbf{v}$ de **palavra recebida.** Para cada $j = 1, 2, \ldots, m$, a j-ésima linha de $\mathbf{v}$, que denotaremos por $\mathbf{v}^j$, é a palavra recebida correspondente à palavra

código $\mathbf{c}^j = \left(c_1^j, c_2^j, \ldots, c_n^j\right)$, do código $\mathcal{C}_j$, que foi enviada. Assim, a linha $\mathbf{v}^j$ será chamada de **palavra recebida do código** $\mathcal{C}_j$, ou simplesmente palavra recebida, quando não houver dúvida de que ela corresponde ao código $\mathcal{C}_j$. Neste processo vamos supor que os códigos $\mathcal{C}_j$ são todos sistemáticos e denotaremos as matrizes geradora e de verificação de paridade de $\mathcal{C}_j$ por $\mathbf{G}_j$ e $\mathbf{H}_j$, respectivamente.

Primeiro, vamos apresentar, em linhas gerais, o processo de decisão para o caso geral de uma modulação $4^m - QAM$. Depois faremos uma descrição detalhada desse processo para o caso particular de uma modulação $64 - QAM$ e, em seguida, daremos um exemplo ilustrativo.

Caso Geral: $4^m - QAM$

Nosso objetivo é encontrar uma matriz $\tilde{\mathbf{c}}$ que coincida com a matriz $\mathbf{c}$ (equação 5.11), caso todas as decisões sejam corretas. Essa matriz $\tilde{\mathbf{c}}$ será construida linha por linha, a partir da primeira, da seguinte maneira: Calculamos $\mathbf{s}(\mathbf{v}^1)$, a síndrome da palavra recebida $\mathbf{v}^1$. Se $\mathbf{s}(\mathbf{v}^1)$ é nula então $\mathbf{v}^1$ é uma palavra código de $\mathcal{C}_1$. Neste caso, fazemos a primeira linha da matriz $\tilde{\mathbf{c}}$ igual à primeira linha de $\mathbf{v}$, ou seja, $\mathbf{v}^1$ é a primeira linha de $\tilde{\mathbf{c}}$. Se $\mathbf{s}(\mathbf{v}^1)$ não é nula então $\mathbf{v}^1$ não é uma palavra código de $\mathcal{C}_1$. Há dois casos possíveis:

1. A síndrome de $\mathbf{v}^1$ é um múltiplo de alguma coluna de $\mathbf{H}_1$, digamos $\mathbf{s}(\mathbf{v}^1) = \lambda \mathbf{h}_1^i$, $\lambda \in \mathbb{Z}_4$, onde $\mathbf{h}_1^i$ denota a i-ésima coluna de $\mathbf{H}_1$. Neste caso, $\mathbf{v}^1$ contém erro apenas na i-ésima coordenada e esse erro tem magnitude λ (Observação 2.15, página 29). Então corrigimos esse erro de $\mathbf{v}^1$ e obtemos uma palavra código $\tilde{\mathbf{v}}^1$, do código $\mathcal{C}_1$. Essa palavra código será a primeira linha da matriz $\tilde{\mathbf{c}}$.

2. A síndrome de $\mathbf{v}^1$ não é múltiplo de nenhuma coluna de $\mathbf{H}_1$. Neste caso $\mathbf{v}^1$ contém erro em mais de uma coordenada. Se apenas duas coordenadas de $\mathbf{v}^1$ contêm erro então, corrigindo um desses erros, obtemos uma palavra $\hat{\mathbf{v}}^1$ cuja síndrome é um múltiplo de alguma coluna de $\mathbf{H}_1$ e aplicamos à palavra $\hat{\mathbf{v}}^1$ o procedimento do caso anterior. Se mais de duas coordenadas de $\mathbf{v}^1$ contêm erro, devemos corrigir erro em mais de

uma coordenada de $\mathbf{v}^1$ para obtermos uma palavra $\hat{\mathbf{v}}^1$ tal que $\mathbf{s}(\hat{\mathbf{v}}^1)$ seja múltiplo de alguma coluna de $\mathbf{H}_1$. Mais precisamente, se t coordenadas de $\mathbf{v}^1$ contêm erro, então, se corrigirmos os erros de $t-1$ dessas coordenadas, obtemos uma palavra $\hat{\mathbf{v}}^1$ cuja síndrome é um múltiplo de alguma coluna de $\mathbf{H}_1$. Esse é o procedimento que adotamos no caso em que nenhuma coluna de $\mathbf{H}_1$ é múltiplo de $\mathbf{s}(\mathbf{v}^1)$, e é realizado assim:

(a) Estabelecemos um critério de confiabilidade dos símbolos de $\mathbf{v}^1$;

(b) Se v_i^1 é o símbolo menos confiável de $\mathbf{v}^1$, fazemos uma mudança em v_i^1 e obtemos uma palavra $\hat{\mathbf{v}}^1$;

(c) Se $\mathbf{s}(\hat{\mathbf{v}}^1)$ é múltiplo de alguma coluna de $\mathbf{H}_1$ então corrigimos o erro de $\hat{\mathbf{v}}^1$ e obtemos uma palavra código $\tilde{\mathbf{c}}^1$, do código $\mathcal{C}_1$, cujas coordenadas formam a primeira linha de $\tilde{\mathbf{c}}$;

(d) Se $\mathbf{s}(\hat{\mathbf{v}}^1)$ não é múltiplo de nenhuma coluna de $\mathbf{H}_1$, desfazemos a mudança feita no símbolo menos confiável de $\mathbf{v}^1$ e repetimos o procedimento com o segundo símbolo de $\mathbf{v}^1$ menos confiável. Se necessário, aplicamos esse procedimento aos demais símbolos de $\mathbf{v}^1$, seguindo a ordem crescente de confiabilidade dos mesmos, e sempre com o objetivo de encontrarmos uma palavra $\hat{\mathbf{v}}^1$ tal que $\mathbf{s}(\hat{\mathbf{v}}^1)$ seja um múltiplo de alguma coluna de $\mathbf{H}_1$. Se nenhuma dessas mudanças resultar numa tal palavra $\hat{\mathbf{v}}^1$, o procedimento é aplicado mudando-se dois ou mais símbolos de $\mathbf{v}^1$ simultâneamente.

3. Uma vez encontrada a primeira linha de $\tilde{\mathbf{c}}$, partimos para encontrar a segunda linha, aplicando à palavra $\mathbf{v}^2$ o procedimento anterior. Procedimento análogo é adotado para encontrar as demais linhas, até que a matriz $\tilde{\mathbf{c}}$ seja completada. Dessa forma, se todas as decisões foram corretas, devemos ter $\tilde{\mathbf{c}} = \mathbf{c}$, onde $\mathbf{c}$ é a palavra código que foi enviada.

Caso Particular: $64-QAM$ **codificado**

Vamos descrever o processo de decodificação de um código $\mathcal{C}$, formado por um conjunto composto por três códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, usados na codificação de uma modulação $64-QAM$, onde $\mathcal{C}_1$ é um código (n, k_1), $\mathcal{C}_2$ é um código de verificação de paridade simples $(n, n-1)$ e $\mathcal{C}_3$ é um código sem redundância (n, n). Vamos denotar as matrizes geradora e de verificação de paridade de $\mathcal{C}_1$ por $\mathbf{G}$ e $\mathbf{H}$, respectivamente, e vamos supor que essas matrizes estão na forma sistemática. O rotulamento do espaço 4^m-QAM é o mostrado na figura 5.3. Então, se um par ordenado de números reais (x, y) é mapeado em um ponto do espaço $64-QAM$, rotulado por abc, devemos ter:

$$a = \begin{cases} 0, \textit{ se} \begin{cases} x < -6 \textit{ ou } -4 \leq x < -2 \textit{ ou } 0 \leq x < 2 \textit{ ou } 4 \leq x < 6 \\ e \\ y < -6 \textit{ ou } -4 \leq y < -2 \textit{ ou } 0 \leq y < 2 \textit{ ou } 4 \leq y < 6 \end{cases} \\ 1, \textit{ se} \begin{cases} -6 \leq x < -4 \textit{ ou } -2 \leq x < 0 \textit{ ou } 2 \leq x < 4 \textit{ ou } x \geq 6 \\ e \\ y < -6 \textit{ ou } -4 \leq y < -2 \textit{ ou } 0 \leq y < 2 \textit{ ou } 4 \leq y < 6 \end{cases} \\ 2, \textit{ se} \begin{cases} -6 \leq x < -4 \textit{ ou } -2 \leq x < 0 \textit{ ou } 2 \leq x < 4 \textit{ ou } x \geq 6 \\ e \\ -6 \leq y < -4 \textit{ ou } -2 \leq y < 0 \textit{ ou } 2 \leq y < 4 \textit{ ou } y \geq 6 \end{cases} \\ 3, \textit{ se} \begin{cases} x < -6 \textit{ ou } -4 \leq x < -2 \textit{ ou } 0 \leq x < 2 \textit{ ou } 4 \leq x < 6 \\ e \\ -6 \leq y < -4 \textit{ ou } -2 \leq y < 0 \textit{ ou } 2 \leq y < 4 \textit{ ou } y \geq 6 \end{cases} \end{cases} \tag{5.13}$$

$$b = \begin{cases} 0, \, se \begin{cases} 0 \leq x < 4 \; ou \; x < -4 \\ e \\ 0 \leq y < 4 \; ou \; y < -4 \end{cases} \\ 1, \, se \begin{cases} -4 \leq x < 0 \; ou \; x \geq 4 \\ e \\ 0 \leq y < 4 \; ou \; y < -4 \end{cases} \\ 2, \, se \begin{cases} -4 \leq x < 0 \; ou \; x \geq 4 \\ e \\ y \geq 4 \; ou \; -4 \leq y < 0 \end{cases} \\ 3, \, se \begin{cases} 0 \leq x < 4 \; ou \; x < -4 \\ e \\ y \geq 4 \; ou \; -4 \leq y < 0 \end{cases} \end{cases} \tag{5.14}$$

$$c = \begin{cases} 0, \, se \; x \geq 0 \; e \; y \geq 0 \\ 1, \, se \; x < 0 \; e \; y \geq 0 \\ 2, \, se \; x < 0 \; e \; y < 0 \\ 3, \, se \; x \geq 0 \; e \; y < 0 \end{cases} \tag{5.15}$$

Seja $\mathbf{a} = (a_1, a_2, \ldots, a_k)$, uma sequência de informação cujos símbolos pertencem a $\mathbb{Z}_4$, onde $k = k_1 + 2n - 1$, e sejam

$$\mathbf{a}_1 = (a_1, \ldots, a_{k_1}), \; \mathbf{a}_2 = (a_{k_1+1,\ldots,a_{k_1+n-1}}) \; e \; \mathbf{a}_3 = (a_{k_1+n}, \ldots, a_{k_1+2n-1})$$

as sequências de informação dos códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, respectivamente. A codificação dessas sequências resulta nas palavras código

$$\mathbf{c}^1 = \left(c_1^1, c_2^1, \ldots, c_n^1\right), \; \mathbf{c}^2 = \left(c_1^2, c_2^2, \ldots, c_n^2\right) \text{ e } \mathbf{c}^3 = \left(c_1^3, c_2^3, \ldots, c_n^3\right)$$

dos códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, respectivamente, que formam a palavra código (equação 5.11)

$$\mathbf{c} = \begin{bmatrix} c_1^1 & c_2^1 & \cdots & c_n^1 \\ c_1^2 & c_2^2 & \cdots & c_n^2 \\ c_1^3 & c_2^3 & \cdots & c_n^3 \end{bmatrix} \tag{5.16}$$

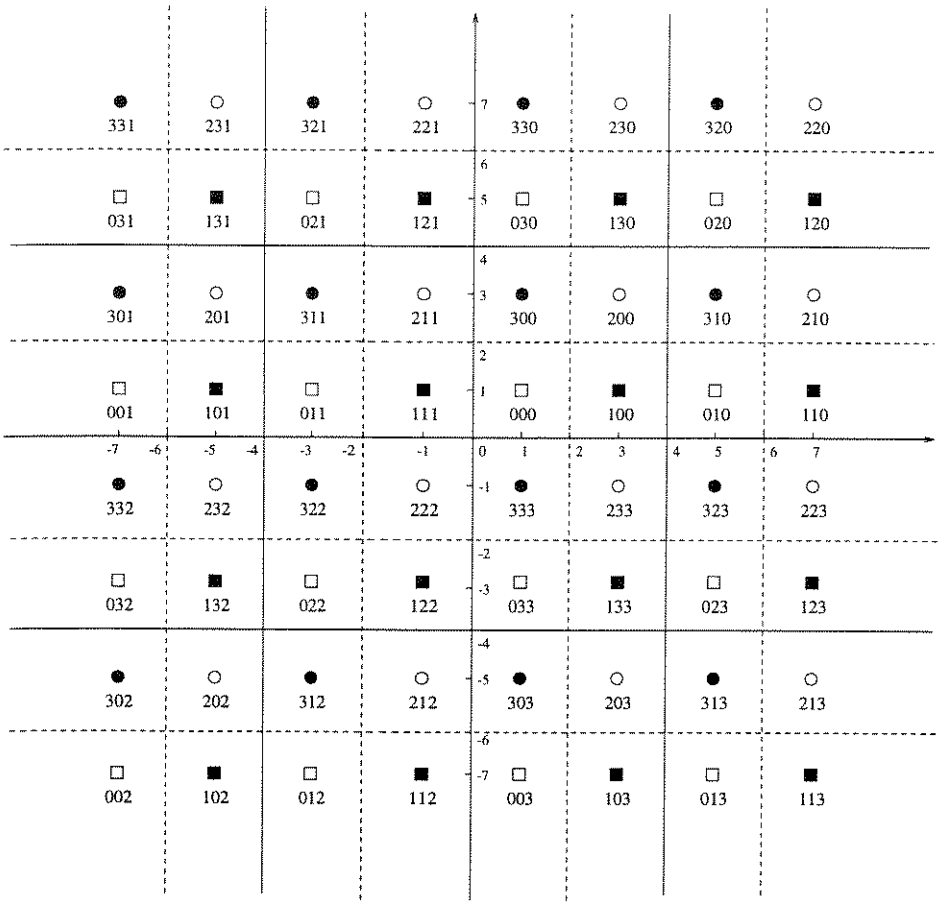

Figura 5.3: Regiões de decisão dos códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$.

do código $\mathcal{C}$. Essa palavra código é modulada e enviada através do canal. Seja

$$\mathbf{r} = \left(\left(x_1^1, x_1^2\right), \left(x_2^1, x_2^2\right), \ldots, \left(x_n^1, x_n^2\right)\right) \tag{5.17}$$

a sequência recebida. As aquações 5.13, 5.14 e 5.15 determinam as regiões de decisão dos códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, no espaço de sinais $64 - QAM$. Fazendo o mapeamento da sequência $\mathbf{r}$, de acordo com essas equações, obtemos a palavra recebida do código $\mathcal{C}$ (equação 5.12)

$$\mathbf{v} = \begin{bmatrix} v_1^1 & v_2^1 & \cdots & v_n^1 \\ v_1^2 & v_2^2 & \cdots & v_n^2 \\ v_1^3 & v_2^3 & \cdots & v_n^3 \end{bmatrix}. \tag{5.18}$$

Seja $\mathbf{v}^j = \left(v_1^j, v_2^j, \ldots, v_n^j\right)$, $j = 1, 2, 3$, a palavra recebida correspondente ao código $\mathcal{C}_j$. Vamos construir uma matriz $\tilde{\mathbf{c}}$ cuja j-ésima linha é uma palavra código de $\mathcal{C}_j$, $j = 1, 2, 3$. Caso não haja erro nas decisões, essa matriz $\tilde{\mathbf{c}}$ deve ser igual à matriz $\mathbf{c}$ (equação 5.16).

Inicialmente, vamos mostrar o procedimento que adotaremos na mudança de um símbolo v_i^1 da palavra $\mathbf{v}^1$, na tentativa de corrigirmos um possível erro nesse símbolo. Em vez de mudarmos a coordenada v_i^1 diretamente na palavra $\mathbf{v}^1$, fazemos uma mudança adequada no par ordenado (x_i^1, x_i^2) da sequência recebida $\mathbf{r}$ (equação 5.17), e depois fazemos o mapeamento da sequência obtida com essa mudança no par ordenado (x_i^1, x_i^2). Esse mapeamento resulta numa matriz que difere da matriz $\mathbf{v}$ apenas na i-ésima coluna. Essa operação altera o valor de v_i^1 e pode alterar ou não os valores de v_i^2 ou v_i^3. Essa mudança no par (x_i^1, x_i^2) consiste em substituir esse par por um dos pares $(\hat{x}_i^1, x_i^2)$, $(x_i^1, \hat{x}_i^2)$ ou $(\hat{x}_i^1, \hat{x}_i^2)$, onde $\hat{x}_i^j$ é escolhido de modo que o mapeamento de um desses dois primeiros pares resulte num sinal de $64 - QAM$ que esteja mais próximo do ponto $P\left(x_i^1, x_i^2\right)$, depois do sinal que resultou do mapeamento do par (x_i^1, x_i^2).

A escolha de um símbolo v_i^1 que deve ser mudado, é feita de acordo com um critério de confiabilidade dos símbolos de $\mathbf{v}^1$. A confiabilidade de um símbolo v_i^1 é determinada pela distância entre o ponto $P\left(x_i^1, x_i^2\right)$ e o sinal que resultou do mapeamento de (x_i^1, x_i^2), ou seja, a distância entre o ponto $P\left(x_i^1, x_i^2\right)$ e o ponto do espaço $64 - QAM$ cujo rótulo é $v_i^1 v_i^2 v_i^3$. Por outro lado, como v_i^1 faz parte de um terno ordenado $v_i^1 v_i^2 v_i^3$, que foi obtido por meio do mapeamento do par ordenado (x_i^1, x_i^2), há casos em que é razoável alterar o valor de v_i^1 e manter os de v_i^2 e v_i^3, mas em alguns casos uma alteração no valor de v_i^1 deve ser seguida por alteração também nos valores de v_i^2 ou v_i^3. Isso depende da posição do ponto $P\left(x_i^1, x_i^2\right)$ em relação ao ponto do espaço $64 - QAM$,.

Vamos estabelecer um parâmetro para compararmos as confiabilidades dos símbolos de $\mathbf{v}^1$. Seja $\mathbf{r} = \left(\left(x_1^1, x_1^2\right), \ldots, \left(x_n^1, x_n^2\right)\right)$ uma sequência recebida . Se $x_i^j \geq 7$ $(x_i^j \leq -7)$ então consideramos $x_i^j = 7$ $(x_i^j = -7)$ (figura 5.3). Para cada $i = 1, 2, \ldots, n$ e $j = 1, 2$; seja $\alpha_i^j \in 2\mathbb{Z}$ tal que $\alpha_i^j \leq x_i^j < \alpha_i^j + 2$, onde $2\mathbb{Z}$ denota o conjunto dos números inteiros pares, e seja $\delta_i^j \triangleq x_i^j - \alpha_i^j - 1$. Se P_i é o ponto do plano que representa o sinal de $64 - QAM$

cujo rótulo é $v_i^1 v_i^2 v_i^3$, então os dois valores de δ_i^j, $j = 1, 2$; determinam a distância entre os pontos $P(x_i^1, x_i^2)$ e P_i, e também a posição do ponto $P(x_i^1, x_i^2)$ na região de decisão à qual P_i pertence. De fato, se um dos valores de $\left|\delta_i^j\right|$ está próximo de 1 então o ponto $P(x_i^1, x_i^2)$ está próximo da fronteira dessa região de decisão, enquanto que se os valores de δ_i^j são próximos de zero então o ponto $P(x_i^1, x_i^2)$ está próximo do ponto P_i. Se ambos os valores são nulos então os pontos $P(x_i^1, x_i^2)$ e P_i coincidem. Se $\delta_i^1 > 0$ ($\delta_i^1 < 0$) então o ponto $P(x_i^1, x_i^2)$ está à direita (à esquerda) de P_i; Se $\delta_i^2 > 0$ ($\delta_i^2 < 0$) então o ponto $P(x_i^1, x_i^2)$ está acima (abaixo) de P_i. Convém observarmos que $-1 \leq \delta_i^j \leq 1$. A confiabilidade do símbolo v_i^1 da palavra recebida $\mathbf{v}^1$ é determinada pelos valores de $\left|\delta_i^j\right|$, $j = 1, 2$. Se $|\delta_i^1| > |\delta_i^2|$ ($|\delta_i^1| < |\delta_i^2|$) então, se houver necessidade de mudar o valor de v_i^1, a primeira mudança deve ser feita na direção do eixo X (eixo Y). Assim, a direção na qual a mudança deve ser feita, também é determinada pelos valores de $\left|\delta_i^j\right|$, enquanto que o sentido (direita, esquerda, para cima ou para baixo) é determinado pelos sinais dos números δ_i^j.

Vamos ordenar o conjunto $\left\{\left|\delta_i^j\right|,\ i = 1, 2, \ldots, n;\ j = 1, 2\right\}$ escrevendo

$$\left\{\left|\delta_i^j\right|, i = 1, 2, \ldots, n;\ j = 1, 2\right\} \triangleq \{\delta_1, \delta_2, \ldots \delta_{2n}\}$$

onde $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_{2n}$. Como o sinal do número δ_i^j também deve ser levado em conta quando mudamos um símbolo, vamos definir a função

$$sinal : \mathbb{R} \longrightarrow \{-1, 1\};\ sinal(x) = \begin{cases} 1, \ se\ x \geq 0 \\ -1, \ se\ x < 0 \end{cases} \tag{5.19}$$

onde $\mathbb{R}$ denota o conjunto dos números reais.

Quando alteramos um ou mais símbolos de uma palavra $\mathbf{v} = (v_1, v_2, \ldots, v_n)$ dizemos que foi feita uma **mudança** em $\mathbf{v}$. Se essa mudança resulta numa palavra $\tilde{\mathbf{v}}$ tal que $D_E^2(\tilde{\mathbf{v}}, \mathbf{v}) = w$, dizemos que a mudança tem peso w. Existem duas possíveis maneiras de se fazer uma mudança de peso 1 na palavra $\mathbf{v}^1$, mudando-se apenas o símbolo v_i^1. Isto é feito mudando o par (x_i^1, x_i^2) para $(\hat{x}_i^1, x_i^2)$ ou $(x_i^1, \hat{x}_i^2)$, onde

$$\hat{x}_i^j = \begin{cases} 5, & se \quad x_i^j \geq 6 \\ x_i^j + sinal(\delta_i^j), & se \quad x_i^j < 6 \end{cases}, \quad j = 1, 2. \tag{5.20}$$

Se, por exemplo, $|\delta_i^1| > |\delta_i^2|$ então, se houver necessidade de se fazer uma mudança de peso 1 em v_i^1, a primeira que deve ser feita é aquela em que mudamos o par (x_i^1, x_i^2) para $(\hat{x}_i^1, x_i^2)$, onde $\hat{x}_i^1$ é dado pela equação 5.20, com $j = 1$. O mapeamento do par ordenado $(\hat{x}_i^1, x_i^2)$ resulta no sinal $\hat{s}_i$ mais próximo do ponto $P(x_i^1, x_i^2)$, depois do sinal s_i. Assim, se s_i não é o sinal que foi enviado então o sinal mais provável de ter sido enviado é $\hat{s}_i$. Para fazermos uma mudança de peso 2 em um símbolo v_i^1, basta fazermos essas duas mudanças de peso 1 simultaneamente, o que equivale a mudar o par (x_i^1, x_i^2) para $(\hat{x}_i^1, \hat{x}_i^2)$ e fazer o mapeamento desse último.

Já vimos que os valores de $\left|\delta_i^j\right|$ determinam a ordem em que os símbolos de $\mathbf{v}^1$ devem sofrer uma eventual mudança e determinam também a direção em que essa mudança deve ser feita (eixo X ou eixo Y). Em outras palavras, o valor de $\left|\delta_i^j\right|$ indica a posição e a magnitude de um suposto erro num símbolo de $\mathbf{v}^1$. Por outro lado, o sinal de δ_i^j determina como esse suposto erro deve ser corrigido, ou seja, se essa correção deve afetar ou não os valores de v_i^2 ou v_i^3. Assim, para corrigirmos um erro do qual já conhecemos a posição e a magnitude, consideramos apenas o sinal de δ_i^j, e essa correção é feita assim: Se o erro tem magnitude λ e ocorre na coordenada v_i^1, substituimos o par ordenado (x_i^1, x_i^2) por $(\tilde{x}_i^1, \tilde{x}_i^2)$, onde

$$\text{Se } \lambda = 1 \text{ e } v_i^1 \text{ é par, então } (\tilde{x}_i^1, \tilde{x}_i^2) = \begin{cases} (x_i^1, x_i^2 + sinal(\delta_i^2)), \text{ se } x_i^2 < 6 \\ (x_i^1, 5), \text{ se } x_i^2 \geq 6 \end{cases} \tag{5.21}$$

$$\text{Se } \lambda = 1 \text{ e } v_i^1 \text{ é ímpar, então } (\tilde{x}_i^1, \tilde{x}_i^2) = \begin{cases} (x_i^1 + sinal(\delta_i^1), x_i^2), \text{ se } x_i^1 < 6 \\ (5, x_i^2), \text{ se } x_i^1 \geq 6 \end{cases} \tag{5.22}$$

$$\text{Se } \lambda = 2, \text{ então } (\tilde{x}_i^1, \tilde{x}_i^2) = \begin{cases} (x_i^1 + sinal(\delta_i^1), x_i^2 + sinal(\delta_i^2)), \; se \; x_i^j < 6 \\ (5, x_i^2 + sinal(\delta_i^2)), \; se \; x_i^1 \geq 6 \; e \; x_i^2 < 6 \\ (x_i^1 + sinal(\delta_i^1), 5), \; se \; x_i^1 < 6 \; e \; x_i^2 \geq 6 \\ (5, 5), \; se \; x_i^1 \geq 6 \; e \; x_i^2 \geq 6 \end{cases} \tag{5.23}$$

$$\text{Se } \lambda = 3 \text{ e } v_i^1 \text{ é par, então } (\tilde{x}_i^1, \tilde{x}_i^2) = \begin{cases} (x_i^1 + sinal(\delta_i^1), x_i^2), \; se \; x_i^1 < 6 \\ (5, x_i^2), \; se \; x_i^1 \geq 6 \end{cases} \tag{5.24}$$

$$\text{Se } \lambda = 3 \text{ e } v_i^1 \text{ é ímpar, então } (\tilde{x}_i^1, \tilde{x}_i^2) = \begin{cases} (x_i^1, x_i^2 + sinal(\delta_i^2)), \text{ se } x_i^2 < 6 \\ (x_i^1, 5), \text{ se } x_i^2 \geq 6 \end{cases} \tag{5.25}$$

Agora vamos retomar o processo de decisão. Se $\mathbf{s}(\mathbf{v}^1)$ é nula então $\mathbf{v}^1$ é uma palavra código de $\mathcal{C}_1$, e a primeira linha de $\tilde{\mathbf{c}}$ será $\mathbf{v}^1$. Se $\mathbf{s}(\mathbf{v}^1)$ não é nula então $\mathbf{v}^1$ não é uma palavra código de $\mathcal{C}_1$, ou seja, $\mathbf{v}^1$ contém erro em uma ou mais coordenadas. Há dois casos possíveis:

1. A síndrome de $\mathbf{v}^1$ é um múltiplo de alguma coluna de $\mathbf{H}$. Se $\mathbf{s}(\mathbf{v}^1) = \lambda \mathbf{h}^i, \lambda \in \mathbb{Z}_4$, onde $\mathbf{h}^i$ denota a i-ésima coluna de $\mathbf{H}$, então $\mathbf{v}^1$ contém erro apenas na i-ésima coordenada e esse erro tem magnitude λ (Observação 2.15, página 29). Neste caso, corrigimos esse erro de $\mathbf{v}^1$, usando as equações 5.21 a 5.25, e obtemos uma palavra código $\tilde{\mathbf{v}}^1$, do código $\mathcal{C}_1$. Essa palavra código será a primeira linha da matriz $\tilde{\mathbf{c}}$.

2. A síndrome de $\mathbf{v}^1$ não é múltiplo de nenhuma coluna de $\mathbf{H}$. Neste caso $\mathbf{v}^1$ contém erro em mais de uma coordenada. Se apenas duas coordenadas de $\mathbf{v}^1$ contêm erro, e um desses erros é de peso 1, então, podemos eliminá-los fazendo mudanças de peso 1, pois numa dessas mudanças eliminamos o erro de peso 1 e obtemos uma palavra que contém erro em apenas uma das coordenadas, ou seja, sua síndrome é um múltiplo de alguma coluna de $\mathbf{H}$. Então aplicamos a essa palavra o procedimento do caso anterior. Se os dois erros são de peso 2 então necessitamos de mudanças de peso 2. Lembramos que, quando uma mudança não elimina o erro de uma das coordenadas, essa mudança é desfeita antes de fazermos a seguinte.

   É importante observar que se $\mathbf{s}(\mathbf{v}^1)$ não é múltiplo de nenhuma coluna de $\mathbf{H}$, não sabemos, *a priori*, quantas coordenadas de $\mathbf{v}^1$ contêm erro. Por isso, adotamos o seguinte procedimento: Fazemos mudanças de peso 1 em $\mathbf{v}^1$, obedecendo o critério de confiabilidade dos seus símbolos, ou seja, obedecendo a ordem decrescente dos δ_i, com o objetivo de encontrarmos uma palavra cuja síndrome seja múltiplo de alguma coluna de $\mathbf{H}$. Se isso ocorrer, aplicamos a essa palavra o procedimento do ítem 1. Se nenhuma das mudanças de peso 1 resultar numa tal palavra, fazemos mudanças de peso 2, 3,... até que uma palavra cuja síndrome seja múltiplo de alguma coluna

de $\mathbf{H}$ seja encontrada. Todas essas mudanças são feitas de acordo com o critério de confiabilidade dos símbolos de $\mathbf{v}^1$.

Vamos agora encontrar a segunda linha de $\tilde{\mathbf{c}}$. Seja $\tilde{\mathbf{r}} = ((\tilde{x}_1^1, \tilde{x}_1^2), \ldots, (\tilde{x}_n^1, \tilde{x}_n^2))$ a última sequência usada para encontrarmos a primeira linha de $\tilde{\mathbf{c}}$. ($\tilde{\mathbf{r}}$ deve ser a sequência recebida $\mathbf{r}$, no caso da palavra recebida $\mathbf{v}^1$ ser uma palavra código de $\mathcal{C}_1$, ou a última sequência usada no ítem 1.) Seja $\tilde{\mathbf{v}}$ a palavra obtida com o mapeamento de $\tilde{\mathbf{r}}$, conforme as equações 5.13, 5.14 e 5.15, e seja $\tilde{\mathbf{v}}^j = \left(\tilde{v}_1^j, \tilde{v}_2^j, \ldots, \tilde{v}_n^j\right)$ a j-ésima linha de $\tilde{\mathbf{v}}$, $j = 1, 2, 3$. É claro que $\tilde{\mathbf{v}}^1$ é igual à primeira linha de $\tilde{\mathbf{c}}$, que já foi encontrada. Se $\tilde{\mathbf{v}}^2$ é uma palavra código de $\mathcal{C}_2$, então tomamos a matriz $\tilde{\mathbf{c}}$ igual à matriz $\tilde{\mathbf{v}}$, já que $\mathcal{C}_3$ é um código sem redundância. Se isso não ocorre, devemos corrigir o(s) erro(s) de $\tilde{\mathbf{v}}^2$, e isto deve ser feito de modo que a palavra código $\tilde{\mathbf{v}}^1$ não seja alterada. Como os símbolos de $\tilde{\mathbf{v}}^2$ são os símbolos do meio, nos rótulos dos sinais do espaço $64 - QAM$, e a variação desses símbolos é determinada pela equação 5.14 (cf. figura 5.4), vamos definir os parâmetros de confiabilidade dos símbolos de $\tilde{\mathbf{v}}^2$ da seguinte maneira: Se $\tilde{x}_i^j \geq 6$ consideramos $\tilde{x}_i^j = 6$ e se $\tilde{x}_i^j \leq -7$ consideramos $\tilde{x}_i^j = -7$. Para cada $i = 1, 2, \ldots, n$ e $j = 1, 2$; seja $\beta_i^j \in 4\mathbb{Z}$, tal que $\beta_i^j \leq \tilde{x}_i^j < \beta_i^j + 4$, onde $4\mathbb{Z}$ denota o conjunto dos números inteiros múltiplos de 4. Sejam $\Delta_i^j \triangleq \tilde{x}_i^j - \beta_i^j - 2$ e

$$\left\{\left|\Delta_i^j\right|, \quad i = 1, 2, \ldots, n \quad e \quad j = 1, 2\right\} \triangleq \{\Delta_1, \Delta_2, \ldots, \Delta_{2n}\}$$

onde $\Delta_1 \geq \Delta_2 \geq \cdots \geq \Delta_{2n}$.

Existem duas possíveis maneiras de se fazer uma mudança de peso 1 na palavra $\tilde{\mathbf{v}}^2$, mudando-se apenas o símbolo $\tilde{v}_i^2$. Isto é feito mudando o par $(\tilde{x}_i^1, \tilde{x}_i^2)$ para $(X_i^1, \tilde{x}_i^2)$ ou $(\tilde{x}_i^1, X_i^2)$, onde

$$X_i^j = \begin{cases} 2, & se \quad \tilde{x}_i^j \geq 6 \\ \tilde{x}_i^j + 4sinal(\Delta_i^j), & se \quad \tilde{x}_i^j < 6 \end{cases}, \quad j = 1, 2. \tag{5.26}$$

O fator 4 multiplicando $sinal(\Delta_i^j)$ garante que o símbolo v_i^1 não será alterado no mapeamento dos pares $(X_i^1, \tilde{x}_i^2)$ ou $(\tilde{x}_i^1, X_i^2)$, segundo as equações 5.13, 5.14 e 5.15.

Agora, seja $p(\tilde{\mathbf{v}}^2) \triangleq \tilde{v}_1^2 \oplus \tilde{v}_2^2 \oplus \cdots \oplus \tilde{v}_n^2$, onde $\oplus$ denota a adição módulo 4. Como $\mathcal{C}_2$ é um código de verificação de paridade simples, se $p(\tilde{\mathbf{v}}^2) = 0$ então $\tilde{\mathbf{v}}^2$ é uma palavra código de $\mathcal{C}_2$.

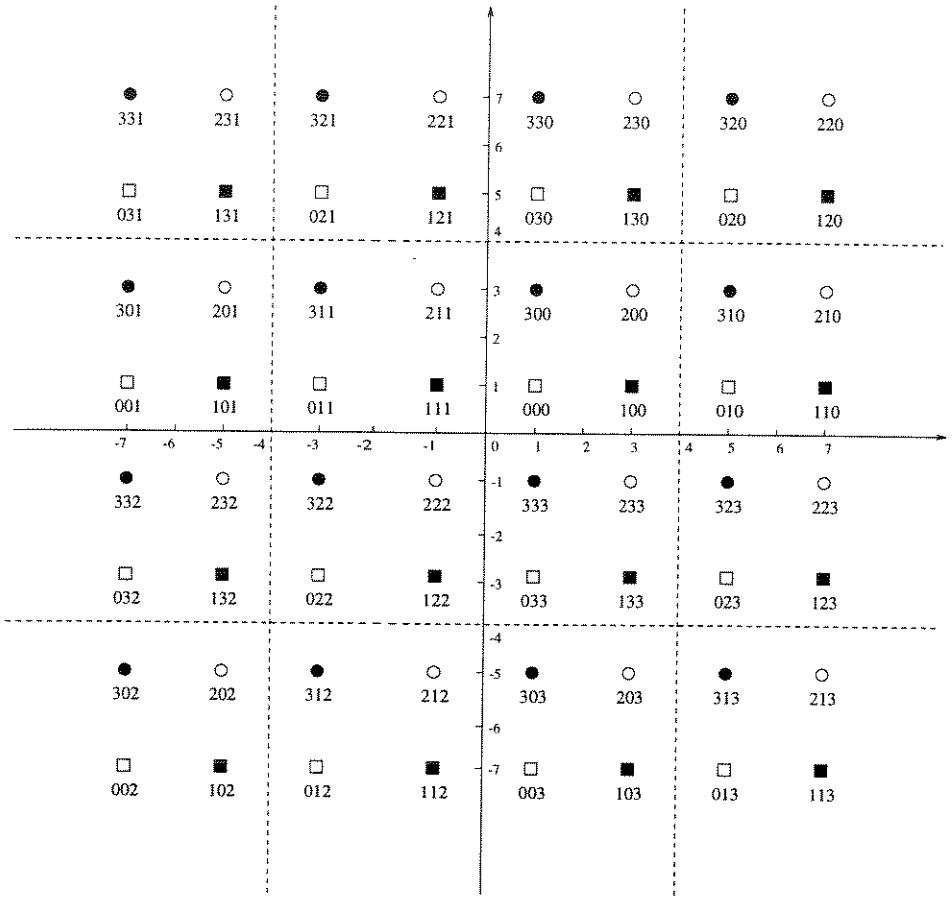

Figura 5.4: Regiões de decisão dos códigos $\mathcal{C}_2$ e $\mathcal{C}_3$.

Neste caso, tomamos a matriz $\tilde{\mathbf{c}}$ igual à matriz $\tilde{\mathbf{v}}$, já que $\mathcal{C}_3$ é um código sem redundância. Se $p(\tilde{\mathbf{v}}^2) \neq 0$ então:

1. Se $p(\tilde{\mathbf{v}}^2) = 1$ ou $p(\tilde{\mathbf{v}}^2) = 3$, então alguma mudança de peso 1 corrige o erro de $\tilde{\mathbf{v}}^2$;

2. Se $p(\tilde{\mathbf{v}}^2) = 2$ então alguma mudança de peso 2 corrige o(s) erro(s) de $\tilde{\mathbf{v}}^2$.

Na figura 5.5 mostramos curvas de taxa de erro de bit versus $\frac{E_b}{N_0}$ $[dB]$ de uma modulação $64-QAM$ codificada e do esquema de referência $32-QAM$ não codificado. No esquema codificado foi utilizado um código $\mathcal{C}$ formado por três códigos $\mathcal{C}_1 : (10,3)$, $\mathcal{C}_2 : (10,9)$ e $\mathcal{C}_3 : (10,10)$, onde a $D^2_{E_{\min}}(\mathcal{C}) = 7$ e $g_\infty = 4,89\,dB$. Observamos que, a uma taxa de erro

de bit igual a 10^{-5}, o esquema codificado apresentou um ganho de aproximadamente $1,0\,dB$ em relação ao esquema não codificado.

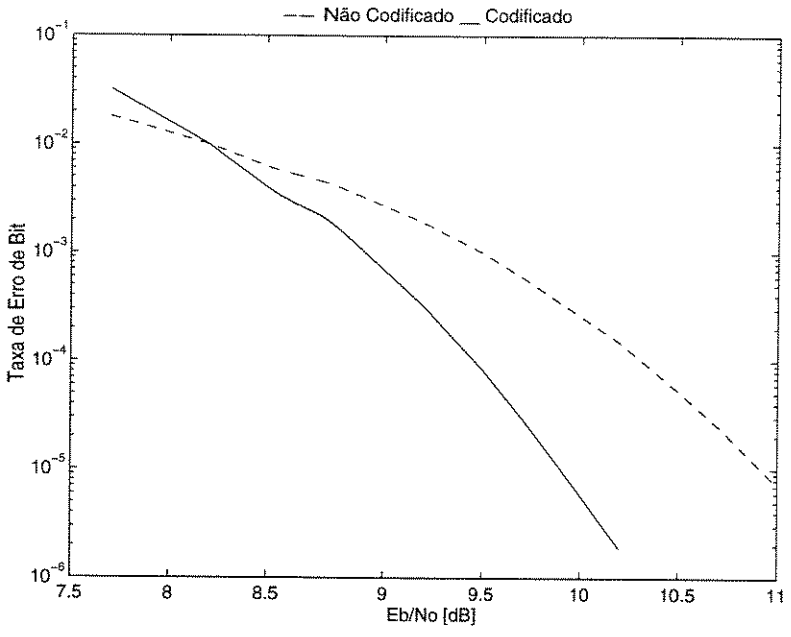

Figura 5.5: Desempenho do $64-QAM$ codificado em relação ao $32-QAM$ não codificado.

Exemplo 5.1 *Seja $\mathcal{C}$ um código formado por um conjunto de três códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, sendo $\mathcal{C}_1 : (10,3)$, $\mathcal{C}_2 : (10,9)$ e $\mathcal{C}_3 : (10,10)$, onde $\mathcal{C}_1$ é o código cuja matriz geradora é*

$$G = \begin{bmatrix} 1 & 0 & 0 & 3 & 2 & 0 & 1 & 0 & 3 & 3 \\ 0 & 1 & 0 & 0 & 3 & 2 & 2 & 2 & 1 & 1 \\ 0 & 0 & 1 & 2 & 0 & 3 & 2 & 3 & 1 & 1 \end{bmatrix}. \tag{5.27}$$

Temos que a matriz de verificação de paridade de $\mathcal{C}_1$ é

$$H = \begin{bmatrix} 1 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 3 & 2 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.28}$$

Suponhamos que uma fonte binária emite uma sequência de bits $\mathbf{b} = (b_1, b_2, \ldots, b_{44})$. Então essa sequência $\mathbf{b}$ é dividida em 22 pares de bits $b_1b_2, b_3b_4, \ldots, b_{43}b_{44}$ e cada um desses pares é mapeado num símbolo de $\mathbb{Z}_4$, por meio de um mapeamento de Gray:

$$0 \leftrightarrow 00,\ 1 \leftrightarrow 01,\ 2 \leftrightarrow 11,\ 3 \leftrightarrow 10. \tag{5.29}$$

Então o mapeamento da sequência $\mathbf{b}$ resulta numa sequência de informação do código $\mathcal{C}$, $\mathbf{a} = (a_1, a_2, \ldots, a_{22})$ cujos símbolos pertencem a $\mathbb{Z}_4$. Essa sequência $\mathbf{a}$ é dividida em três subsequências $\mathbf{a}_1 = (a_1, a_2, a_3)$, $\mathbf{a}_2 = (a_4, a_5, \ldots, a_{12})$ e $\mathbf{a}_3 = (a_{13}, a_{14}, \ldots, a_{22})$, que são as sequências de informação dos códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, respectivamente. A codificação dessas sequências resulta numa palavra código de $\mathcal{C}$

$$\mathbf{c} = \begin{bmatrix} c_1^1 & c_2^1 & \cdots & c_{10}^1 \\ c_1^2 & c_2^2 & \cdots & c_{10}^2 \\ c_1^3 & c_2^3 & \cdots & c_{10}^3 \end{bmatrix} \tag{5.30}$$

Cada coluna dessa palavra código representa um sinal de $64-QAM$. Portanto, essa palavra código representa uma sequência codificada $\mathbf{s} = (s_1, s_2, \ldots, s_{10})$ de sinais de $64-QAM$. Cada sinal de $\mathbf{s}$ é modulado e enviado através do canal. Então na saída do demodulador temos uma sequência recebida $\mathbf{r} = ((x_1^1, x_1^2), (x_2^1, x_2^2), \ldots, (x_{10}^1, x_{10}^2))$.

Vamos adotar a seguinte notação para a sequência $\mathbf{r}$:

$$\mathbf{r} = \begin{array}{|c|c|c|c|} \hline x_1^1 & x_2^1 & \cdots & x_{10}^1 \\ \hline x_1^2 & x_2^2 & \cdots & x_{10}^2 \\ \hline \end{array}$$

Suponhamos que a sequência recebida seja

$$\mathbf{r} = \begin{array}{|c|c|c|c|c|c|c|c|c|c|} \hline -0,7 & -4,2 & -3,2 & -1,4 & 3,5 & -5,2 & 1,3 & 0,3 & -2,8 & 5,4 \\ \hline -6,8 & 1,7 & 5,4 & 3,3 & 5,2 & 2,7 & 3,4 & 0,2 & -7,5 & -3,3 \\ \hline \end{array} . \tag{5.31}$$

Fazendo o mapeamento de $\mathbf{r}$, de acordo com as equações 5.13, 5.14 e 5.15, obtemos a palavra recebida do código $\mathcal{C}$

$$\mathbf{v} = \begin{bmatrix} 1 & 1 & 0 & 2 & 1 & 2 & 3 & 0 & 0 & 0 \\ 1 & 0 & 2 & 1 & 3 & 0 & 0 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 2 & 3 \end{bmatrix} . \tag{5.32}$$

Temos que

$$\mathbf{v}^1 = (1,1,0,2,1,2,3,0,0,0)\,,\ \mathbf{v}^2 = (1,0,2,1,3,0,0,0,1,2) \text{ e } \mathbf{v}^3 = (2,1,1,1,0,1,0,0,2,3)$$

são as palavras recebidas dos códigos $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$, respectivamente. Calculando a síndrome de $\mathbf{v}^1$ obtemos $\mathbf{s}(\mathbf{v}^1) = (3,0,0,0,2,0,0)$, que nem é nula nem múltiplo de alguma coluna de $\mathbf{H}$. Então devemos fazer mudanças de peso 1 em $\mathbf{v}^1$. Temos

$$\begin{array}{lllll} \delta_1^1 = 0,3 & \delta_2^1 = 0,8 & \delta_3^1 = -0,2 & \delta_4^1 = -0,4 & \delta_5^1 = 0,5 \\ \delta_6^1 = -0,2 & \delta_7^1 = 0,3 & \delta_8^1 = -0,7 & \delta_9^1 = 0,2 & \delta_{10}^1 = 0,4 \\ \delta_1^2 = 0,2 & \delta_2^2 = 0,7 & \delta_3^2 = 0,4 & \delta_4^2 = 0,3 & \delta_5^2 = 0,2 \\ \delta_6^2 = -0,3 & \delta_7^2 = 0,4 & \delta_8^2 = -0,8 & \delta_9^2 = 0 & \delta_{10}^2 = -0,3 \end{array} \tag{5.33}$$

Então

$$\begin{array}{lllll} \delta_1 = |\delta_2^1| & \delta_2 = |\delta_8^2| & \delta_3 = |\delta_2^2| & \delta_4 = |\delta_8^1| & \delta_5 = |\delta_5^1| \\ \delta_6 = |\delta_3^2| & \delta_7 = |\delta_4^1| & \delta_8 = |\delta_7^2| & \delta_9 = |\delta_{10}^1| & \delta_{10} = |\delta_1^1| \\ \delta_{11} = |\delta_4^2| & \delta_{12} = |\delta_6^2| & \delta_{13} = |\delta_7^1| & \delta_{14} = |\delta_{10}^2| & \delta_{15} = |\delta_1^2| \\ \delta_{16} = |\delta_3^1| & \delta_{17} = |\delta_5^2| & \delta_{18} = |\delta_6^1| & \delta_{19} = |\delta_9^1| & \delta_{20} = |\delta_9^2| \end{array} \tag{5.34}$$

Na tabela 5.3 mostramos as sete primeiras mudanças de peso 1 em $\mathbf{v}^1$. Na sétima delas obtivemos uma palavra $\hat{\mathbf{v}}^1$ tal que $\mathbf{s}(\hat{\mathbf{v}}^1) = 2\mathbf{h}^8$, onde $\mathbf{h}^8$ é a oitava coluna de $\mathbf{H}$.

Então a palavra $\hat{\mathbf{v}}^1 = (1,1,0,3,1,2,3,0,0,0)$ contém erro apenas na oitava coordenada, e esse erro é de magnitude 2. A sequência cujo mapeamento no espaço $64 - QAM$ resultou

δ_i	Símbolo de $\mathbf{r}$ mudado (mudança)	Símbolo de $\mathbf{v}^1$ mudado (mudança)	Palavra obtida: $\hat{\mathbf{v}}^1$	$\mathbf{s}(\hat{\mathbf{v}}^1)$
δ_1	x_2^1 $(-4,2 \longrightarrow -3,2)$	v_2^1 $(1 \longrightarrow 0)$	$(1,0,0,2,1,2,3,0,0,0)$	$(3,3,2,2,0,1,1)$
δ_2	x_8^2 $(0,2 \longrightarrow -0,8)$	v_8^1 $(0 \longrightarrow 3)$	$(1,1,0,2,1,2,3,3,0,0)$	$(3,0,0,0,1,0,0)$
δ_3	x_2^2 $(1,7 \longrightarrow 2,7)$	v_2^1 $(1 \longrightarrow 2)$	$(1,2,0,2,1,2,3,0,0,0)$	$(3,1,2,2,0,3,3)$
δ_4	x_8^1 $(0,3 \longrightarrow -0,7)$	v_8^1 $(0 \longrightarrow 1)$	$(1,1,0,2,1,2,3,1,0,0)$	$(3,0,0,0,3,0,0)$
δ_5	x_5^1 $(3,5 \longrightarrow 4,5)$	v_5^1 $(1 \longrightarrow 0)$	$(1,1,0,2,0,2,3,0,0,0)$	$(3,3,0,0,2,0,0)$
δ_6	x_3^2 $(5,4 \longrightarrow 6,4)$	v_3^1 $(0 \longrightarrow 3)$	$(1,1,3,2,1,2,3,0,0,0)$	$(1,0,3,2,1,1,1)$
δ_7	x_4^1 $(-1,4 \longrightarrow -2,4)$	v_4^1 $(2 \longrightarrow 3)$	$(1,1,0,3,1,2,3,0,0,0)$	$(0,0,0,0,2,0,0)$

Tabela 5.3: Mudanças de peso 1 feitas em $\mathbf{v}^1$.

na palavra $\hat{\mathbf{v}}^1$ foi

$$\hat{\mathbf{r}} = \begin{array}{|r|r|r|r|r|r|r|r|r|r|} \hline -0,7 & -4,2 & -3,2 & -2,4 & 3,5 & -5,2 & 1,3 & 0,3 & -2,8 & 5,4 \\ \hline -6,8 & 1,7 & 5,4 & 3,3 & 5,2 & 2,7 & 3,4 & 0,2 & -7,5 & -3,3 \\ \hline \end{array} \tag{5.35}$$

que é a sequência recebida $\mathbf{r}$ (equação 5.31) com o símbolo x_4^1, do par ordenado (x_4^1, x_4^2), alterado de $-1,4$ para $-2,4$. Corrigirmos o erro de $\hat{\mathbf{v}}^1$, mudando o par ordenado (x_8^1, x_8^2) de $\hat{\mathbf{r}}$, de acordo com a equação 5.23, e obtemos a sequência

$$\tilde{\mathbf{r}} = \begin{array}{|r|r|r|r|r|r|r|r|r|r|} \hline -0,7 & -4,2 & -3,2 & -2,4 & 3,5 & -5,2 & 1,3 & -0,7 & -2,8 & 5,4 \\ \hline -6,8 & 1,7 & 5,4 & 3,3 & 5,2 & 2,7 & 3,4 & -0,8 & -7,0 & -3,3 \\ \hline \end{array}$$

Fazendo o mapeamento de $\tilde{\mathbf{r}}$, de acordo com as equações 5.13, 5.14 e 5.15, obtemos a palavra

$$\tilde{\mathbf{v}} = \begin{bmatrix} 1 & 1 & 0 & 3 & 1 & 2 & 3 & 2 & 0 & 0 \\ 1 & 0 & 2 & 1 & 3 & 0 & 0 & 2 & 1 & 2 \\ 2 & 1 & 1 & 1 & 0 & 1 & 0 & 2 & 2 & 3 \end{bmatrix}. \tag{5.36}$$

Denotemos por $\tilde{\mathbf{v}}^j$ a j-ésima linha de $\tilde{\mathbf{v}}$, $j = 1,2,3$. Sabemos que $\tilde{\mathbf{v}}^1 = (1,1,0,3,1,2,3,2,0,0)$ é uma palavra código de $\mathcal{C}_1$. Portanto $\tilde{\mathbf{v}}^1$ será também a primeira linha da matriz $\tilde{\mathbf{c}}$.

Vamos verificar se a segunda linha de $\tilde{\mathbf{v}}$ é uma palavra código de $\mathcal{C}_2$. Temos que a soma, módulo 4, dos símbolos de $\tilde{\mathbf{v}}^2$ é $p(\tilde{\mathbf{v}}^2) = 1 \oplus 2 \oplus 1 \oplus 3 \oplus 2 \oplus 1 \oplus 2 = 0$. Como $\mathcal{C}_2$ é um código de verificação de paridade simples, então $\tilde{\mathbf{v}}^2$ é uma palavra código de $\mathcal{C}_2$. Assim, como $\mathcal{C}_3$ é um código sem redundância, a matriz $\tilde{\mathbf{c}}$ é igual a $\tilde{\mathbf{v}}$ (equação 5.36), ou seja,

$$\tilde{\mathbf{c}} = \begin{bmatrix} 1 & 1 & 0 & 3 & 1 & 2 & 3 & 2 & 0 & 0 \\ 1 & 0 & 2 & 1 & 3 & 0 & 0 & 2 & 1 & 2 \\ 2 & 1 & 1 & 1 & 0 & 1 & 0 & 2 & 2 & 3 \end{bmatrix}. \tag{5.37}$$

Então as palavras códigos de $\mathcal{C}_1$, $\mathcal{C}_2$ e $\mathcal{C}_3$ são

$$\tilde{\mathbf{c}}^1 = (1,1,0,3,1,2,3,2,0,0),\ \tilde{\mathbf{c}}^2 = (1,0,2,1,3,0,0,2,1,2) \text{ e } \tilde{\mathbf{c}}^3 = (2,1,1,1,0,1,0,2,2,3),$$

respectivamente. Como a matriz geradora do código $\mathcal{C}_1$ está na forma sistemática, $\mathcal{C}_2$ é um código de verificação de paridade simples e $\mathcal{C}_3$ é um código sem redundância, a decodificação dessas palavras código resulta nas sequências

$$\tilde{\mathbf{a}}_1 = (1,1,0),\ \tilde{\mathbf{a}}_2 = (1,0,2,1,3,0,0,2,1) \text{ e } \tilde{\mathbf{a}}_3 = (2,1,1,1,0,1,0,2,2,3)$$

cujos símbolos pertencem a $\mathbb{Z}_4$. Convertendo os símbolos dessas sequências para bits (5.29), temos as sequências binárias

$$(010100),\ (010011011000001101) \text{ e } (11010101000100111110).$$

Portanto, caso todos os erros tenham sido corrigidos, a sequência de bits emitida pela fonte binária foi

$$01010001001101100000110111010101000100111110.$$

5.6 Conclusão

Uma nova técnica de codificação para as modulações $4^m - QAM$ retangulares, $m \geq 2$, foi apresentada neste capítulo. Essa técnica utiliza um conjunto de códigos formado por m códigos de bloco multiníveis sobre $\mathbb{Z}_4$. Um método de decodificação também foi apresentado.

Com códigos transparentes de comprimento $n \leq 20$, foram obtidos ganhos de codificação assintóticos de até $6,0\,dB$, onde os esquemas codificado e não codificado têm as mesmas taxas de transmissão de informação. Também foram considerados casos em que esses esquemas não têm essas mesmas taxas, e ganhos de quase $6\,dB$ foram obtidos.

Capítulo 6

Comentários, Sugestões e Conclusão

6.1 Comentários

Baldini [2] propôs uma técnica de modulação codificada que utiliza códigos de bloco multiníveis, definidos sobre o anel $\mathbb{Z}_4$, em modulações $16-QAM$. Neste trabalho, nós estendemos essa técnica para modulações 4^m-QAM, onde $m \geq 2$ é um inteiro qualquer. Neste caso também utilizamos códigos de bloco multiníveis sobre $\mathbb{Z}_4$. As razões que justificam o uso desses códigos são:

- Um espaço de sinais 4^m-QAM pode ser visto como um produto

$$4^m-QAM = (4-QAM) \times \cdots \times (4-QAM) \tag{6.1}$$

onde o lado direito possui m fatores. Como o espaço $4-QAM$ possui as mesmas características do espaço $4-PSK$ então, conforme mostrado no capítulo 3, os códigos sobre $\mathbb{Z}_4$ são os mais naturais para serem usados em modulações $4-QAM$. Assim, o produto 6.1 sugere que a codificação de uma modulação 4^m-QAM seja feita por meio de m códigos definidos sobre $\mathbb{Z}_4$. Esse produto também sugere que os pontos de um espaço 4^m-QAM sejam rotulados por m-uplas, com símbolos pertencentes a $\mathbb{Z}_4$.

- As ambiguidades de fase de uma modulação 4^m-QAM são de múltiplos de $\frac{\pi}{2}\,rad$, e códigos definidos sobre $\mathbb{Z}_4$ podem ser rotacionalmente invariantes sob todos os múltiplos

de $\frac{\pi}{2}\,rad.$

Assim, adotamos um processo de codificação que envolve m códigos de bloco multiníveis $\mathcal{C}_1, \ldots, \mathcal{C}_m$, sobre $\mathbb{Z}_4$, em correspondência com os símbolos dos rótulos dos sinais, e cada código protege o respectivo símbolo, de forma independente. Os códigos $\mathcal{C}_i$ têm o mesmo comprimento e taxa de codificação R_i. Esse conjunto de códigos pode ser visto como um único código $\mathcal{C} : (mn, k)$, onde $k = k_1 + \cdots + k_m$ e a taxa de codificação é $R_t = \frac{k_1+\cdots+k_m}{mn} = \frac{R_1+\cdots+R_m}{m}$. Esse processo de codificação é uma extensão, para códigos multiníveis, do processo proposto por Sayegh [27]. O principal parâmetro que determina o desempenho do código $\mathcal{C}$ é a sua distância Euclidiana quadrada mínima. Assim, o processo de busca desses códigos foi direcionado para encontrar códigos cujas distâncias Euclidianas sejam as maiores possíveis.

A maneira como os pontos do espaço $4^m - QAM$ são rotulados tem influência direta na distância Euclidiana quadrada mínima de $\mathcal{C}$. Então, um rotulamento adequado pode otimizar o desempenho desses códigos, e isso é conseguido mediante um particionamento do conjunto de pontos do espaço $4^m - QAM$, que divide esse conjunto em subconjuntos com distâncias Euclidianas mínimas progressivamente crescentes.

Considere uma modulação $16 - QAM$ codificada por dois códigos de bloco multiníveis $\mathcal{C}_1$ e $\mathcal{C}_2$, onde $\mathcal{C}_1$ é um código $\left(n, \frac{n}{2} + 1\right)$ (n deve ser par) e $\mathcal{C}_2$ é um código de verificação de paridade simples $(n, n-1)$. Se a modulação $8 - QAM$ não codificada é tomada como esquema de referência, então esses dois esquemas têm as mesmas taxas de transmissão de informação. Suponhamos que o ganho de codificação assintótico do esquema codificado sobre o de referência é igual a $g\,dB$. Esses mesmos códigos $\mathcal{C}_1$ e $\mathcal{C}_2$ podem ser usados na codificação de qualquer esquema de modulação $4^m - QAM, m > 2$, bastando acrescentar $m-2$ códigos sem redundância $\mathcal{C}_3, \ldots, \mathcal{C}_m$. O esquema $4^m - QAM$ codificado dessa maneira tem a mesma taxa de transmissão de informação do esquema $\frac{4^m}{2} - QAM$ não codificado e o ganho de codificação assintótico entre esses dois esquemas também é igual a $g\,dB$. Assim, para se obter ganhos assintóticos entre $3,0$ e $6,0$ $[dB]$, basta que se encontre um código $\left(n, \frac{n}{2} + 1\right)$ com distância Euclidiana quadrada mínima entre 4 e 8 (tabela 5.1). Ganhos acima de $6,0\,dB$ podem ser obtidos mas, para isso, é necessário diminuir a taxa de codificação do código $\mathcal{C}_2$

e aumentar a do código $\mathcal{C}_1$, para que os esquemas codificado e não codificado tenham as mesmas taxas de transmissão de informação. Se estivermos considerando as modulações $4^m - QAM$ e $\frac{4^m}{2} - QAM$ como esquemas codificado e não codificado, respectivamente, então devemos tomar os códigos $\mathcal{C}_1$ e $\mathcal{C}_2$ da forma $\mathcal{C}_1 : \left(n, \frac{n}{2} + l\right)$ e $\mathcal{C}_2 : (n, n - l)$. Os demais códigos são códigos sem redundância pois a distância Euclidiana quadrada mínima nos subconjuntos do segundo nível da partição, já é igual a $16\,D_0^2$, onde D_0^2 é a distância Euclidiana quadrada mínima de $4^m - QAM$. Daí nota-se que para se obter ganhos acima de $6,0\ dB$ devemos considerar códigos cujos comprimentos são bem maiores do que aqueles da tabela 5.1. O esforço computacional na busca desses códigos cresce exponencialmente quando aumentamos os seus comprimentos. Por isso, não foram feitas buscas de códigos longos. Devemos salientar que em nossa busca computacional consideramos apenas uma classe de códigos (Seção 5.4.1). Como os códigos dessa classe são transparentes, com o uso de codificação diferencial, sempre podemos tornar os sistemas transparentes às rotações de fase da portadora, ao contrário do que ocorre com os esquemas de Ungerboeck que, em geral, não são invariantes a todas as rotações de fase. Esquemas de modulações codificadas com taxas de transmissão de informação diferentes das taxas dos esquemas de referência, também apresentaram bons desempenhos (tabela 5.2).

Para a decodificação desses códigos foi proposto um método que é uma extensão, para códigos de bloco multiníveis, daquele proposto por Sayegh [27]. Esse método, em seu processo de decisão, leva em conta a confiabilidade dos símbolos das palavras recebidas, e isto tem como base a distância Euclidiana. Esse processo se revelou bastante complexo para fins práticos, uma vez que o número de operações pode se tornar muito elevado. Para códigos de comprimento $n \geq 10$ ele se torna impraticável.

6.2 Sugestões para Trabalhos Futuros

Como todo trabalho de pesquisa, os resultados obtidos não completam e não finalizam as investigações. Existe sempre a possibilidade de aprimoramento do trabalho apresentado

e de sua extensão, analisando pontos não abordados. Dentre estes pontos destacamos:

- Busca de outras classes de códigos que ofereçam desempenhos melhores do que os apresentados.
- No nosso trabalho consideramos apenas modulações QAM cujo número de sinais é uma potência de 4, isto é, modulações com constelações quadradas. Seria interessante estender essa técnica aos demais casos, ou seja, quando o número de sinais é uma potência de 2, com expoente ímpar (modulações não quadradas).
- Métodos eficientes de decodificação com baixa complexidade, tais como Divisão de Síndrome [10], Vizinhos de Zero [15], etc. devem ser investigados.
- Métodos sub-ótimos de decodificação suave também devem ser investigados, para aplicação nos esquemas BCM apresentados.
- Generalização do método de codificação apresentado para constelações multidimensionais.

6.3 Conclusão

Neste trabalho apresentamos uma nova técnica de modulação codificada, que utiliza códigos de bloco multiníveis sobre $\mathbb{Z}_4$ em modulações $4^m - QAM$, $m \geq 2$. Os códigos definidos sobre $\mathbb{Z}_4$ podem ser rotacionalmente invariantes sob todas as ambiguidades de fase das modulações $4^m - QAM$. Então podemos encontrar códigos transparentes às rotações de fase da portadora.

O processo de codificação de uma modulação $4^m - QAM$ é feito por meio de m códigos de bloco sobre $\mathbb{Z}_4$. Esses mesmos códigos podem ser usados na codificação de qualquer modulação $4^{m'} - QAM$, $m' > m$, bastando acrescentar $m' - m$ códigos sem redundância. Esses dois esquemas codificados dessa maneira apresentam os mesmos ganhos de codificação assintóticos, sobre os respectivos esquemas de referência.

Com códigos transparentes de comprimento $n \leq 20$, foram obtidos ganhos de codificação assintóticos de até $6,0\,dB$, onde os esquemas codificado e não codificado têm as mesmas taxas de transmissão de informação. Também foram considerados casos em que esses esquemas não têm essas mesmas taxas, e ganhos de quase $6\,dB$ foram obtidos. Um método de decodificação para os códigos utilizados também foi proposto.

Bibliografia

[1] **Baldini Filho**, R., "*Coded Modulation Based on Rings of Integers*", PhD Thesis, University of Manchester, 1992.

[2] **Baldini Filho**, R., "*Phase Rotation Invariant Block Codes for Coded 16-QAM* ", 2nd. International Symposium on Communication Theory and Applications, Ambleside, Lake District, U. K., July 11-16, 1993.

[3] **Baldini Filho**, R., **Farrell**, P. G., "*Coded Modulation Based on Rings of Integers modulo-q. Part 1: Block Codes*", **IEE** Proc.-Commun., Vol. 141, No. 3, June 1994.

[4] **Baldini Filho**, R., **Farrell**, P. G., *"Coded Modulation Based on Rings of Integers modulo-q. Part 2: Convolutional Codes*", IEE Proc.-Commun., Vol. 141, No. 3, June 1994.

[5] **Baldini Filho**, R., **Pessoa**, A. C. F., **Arantes**, D. S., "*Systematic Linear Codes over a Ring for Encoded Phase Modulation* ", International Symposium on Information and Coding Theory (ISICT'87), 27 Jul.- 1 Aug. 1987, Campinas-SP-Brazil

[6] **Biglieri**, E., **Divsalar**, D., **McLane**, P. J., **Simon**, M. K., "*Introduction to Trellis-Coded Modulation with Applications*", Macmillan Publishing Company, New York, 1991.

[7] **Blahut**, R. E., "*Theory and Practice of Error Control Codes*", Addison-Wesley, 1984.

[8] **Clark**, Jr. G. C., **Cain**, J. B., "*Error-Correction Coding for Digital Communications*", Plenum-Press, 1988.

[9] **Cusack**, E. L., "*Error Control Codes for QAM Signalling*", Electronics Letters, Vol. 23, Jan. 1984.

[10] **Eiguren**, J., **Dumer**, I. I., **Farrell**, P. G., "*Split Syndrome Soft-Decision Decoding for Block Codes*", 4^{th} IMA Conf. on Cryptography and Coding, Cirencester, U.K., Dec. 13-15, 1993.

[11] **Fraleigh**, J. B., "*A First Course in Abstract Algebra*", Adison-Wesley Publishing Company, 1973.

[12] **Ginzburg**, V., "*Multidimensional Signals for a Continuous Channel*", Problemi Paredachi Informatsii 20(1), 1984.

[13] **Gonçalves**, A., "*Introdução à Álgebra*", Instituto de Matemática Pura e Aplicada, Projeto Euclides, 1979.

[14] **Imai**, H., **Hirakawa**, S.,"*A New Coding Method Using Error-Correcting Codes*", IEEE Trans. on Infomation Theory, Vol. IT-23, No. 3, 1977.

[15] **Levitin**, L. B., **Hartmann**, C. R. P., "*A New Approach to the General Minimum Distance Decoding Problem: The Zero-Neighbors Algorithm*", IEEE Trans. on Infomation Theory, Vol. IT-31, No. 3, May 1985.

[16] **Lin**, S., **Costello**, Jr. D. J. "*Error Control Coding: Fundamentals and Applications*", Prentice-Hall, 1983.

[17] **López**, F. J.,"*Optimal Design and Application of Trellis Coded Modulation Techniques defined over The Ring of Integers*" , PhD Thesis, Staffordshire University, May 1994.

[18] **López**, F. J., **Carrasco**, R. A., **Farrell**, P. G., "*Ring-TCM Codes for QAM*", Third IEE European Conference on Satellite Communications, Manchester, U. K., Nov. 1993.

[19] **MacWilliams**, F. J., **Sloane**, N. J. A. ,"*The Theory of Error-Correcting Codes*", North-Holland, 1988.

[20] **Massey**, J. L., "*Coding and Modulation in Digital Communications*", Proc. 1974 Int. Zurich Seminar on Digital Communications, Zurich, Switzerland, March 1974.

[21] **Massey**, J. L., **Mittelholzer**, T., "*Codes over Rings - A Pratical Necessity*", AAECC7 International Conference, Université P. Sabatier, Toulouse, France, June 1989.

[22] **Massey**, J. L., **Mittelholzer**, T., "*Convolutional Codes over Rings*", Proceedings of Fourth Joint Swedish-Soviet International Workshop on information Theory, Gotland, Sweden, August 1989.

[23] **Michelson**, A. M., **Levesque**, A. H., "*Error-Control Techniques For Digital Communication*", John Willey & Son, 1985.

[24] **Peterson**, W. W., **Weldon**, Jr. E. J., "*Error Correcting Codes*", The MIT press, 1984.

[25] **Piret**, P.,"*Convolutional Codes - An Algebraic Approach*", The MIT press, 1988.

[26] **Roman**, S., "*Advanced Linear Algebra*", Springer-Verlag, 1992.

[27] **Sayegh**, S., "*A Class of Optimum Block Codes in Signal Space*", IEEE Trans. on Communications, Vol. COM-34, No. 10, Oct. 1986.

[28] **Ungerboeck**, G.,"*Channel Coding with Multilevel/Phase Signals*", IEEE Trans. on Information Theory, Vol. IT-28, No. 2, Jan. 1982.

[29] **Ungerboeck**, G.,"*Trellis-Coded Modulation with Redundant Signal Sets - Part I: Introduction, and Part II: State of the Art*", IEEE Communications Magazine, Vol. 25, No. 2, Feb. 1987.

[30] **Voloch**, J. F., "*Códigos Corretores de Erros*", Instituto de Matemática Pura e Aplicada, 16º Colóquio Brasileiro de Matemática, 1987.

[31] **Wei**, L. F., "*Rotationally Invariant Convolutional Channel Coding with Expanded Signal Space. Part I: 180°*", IEEE Journal on Selected Areas in Comm., Vol. SAC-2, Sept. 1984.

[32] **Wei**, L. F., "*Rotationally Invariant Convolutional Channel Coding with Expanded Signal Space. Part II: Nonlinear Codes*", IEEE Journal on Selected Areas in Comm., Vol. SAC-2, Sept. 1984.

[33] **Wei**, L., F., "*Trellis-Coded Modulation with Multi-dimensional Constelations*", IEEE Trans. on Information Theory, Vol. IT-33, No. 4, July 1987.