

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

Este exemplar corresponde a redação final da tese  
defendida por SAULO ROBERTO SODRÉ  
DOS REIS e aprovada pela Comissão  
Julgada em 03 / 09 / 1999.  
Jaime Portugheis  
Orientador

UM SISTEMA FH-CDMA COM CANCELAMENTO DE  
INTERFERÊNCIA E CÓDIGOS CORRETORES DE ERRO

Por  
SAULO ROBERTO SODRÉ DOS REIS

DISSERTAÇÃO SUBMETIDA À FACULDADE DE ENGENHARIA ELÉTRICA  
DA UNICAMP COMO PARTE DOS REQUISITOS EXIGIDOS PARA A  
OBTENÇÃO DO TÍTULO DE MESTRE EM ENGENHARIA ELÉTRICA.

Banca Examinadora :

Prof. Dr. Jaime Portugheis (Orientador) - FEEC/UNICAMP  
Prof. Dr. Bartolomeu Ferreira Uchôa Filho - EEL/CTC/UFSC  
Prof. Dr. Celso de Almeida - FEEC/UNICAMP

UNICAMP  
BIBLIOTECA CENTRAL  
SEÇÃO CIRCULANTE

Campinas, Julho de 1999.



IDADE BC  
CHAMADA:  
Unicamp  
R277s  
Ex.  
MBO BC/ 42272  
OC. 16-278100  
C  D   
EC R\$11,00  
TA 22109100  
CPD

CM-00144266-B

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

R277s      Reis, Saulo Roberto Sodré dos  
Um sistema FH-CDMA com cancelamento de  
interferência e códigos corretores de erro / Saulo  
Roberto Sodré dos Reis.--Campinas, SP: [s.n.], 1999.

Orientador: Jaime Portugheis  
Dissertação (mestrado) - Universidade Estadual de  
Campinas, Faculdade de Engenharia Elétrica e de  
Computação.

1. Códigos de controle de erros (Teoria da  
informação). 2. Modulação digital. 3. Comunicações  
digitais. I. Portugheis, Jaime. II. Universidade Estadual  
de Campinas. Faculdade de Engenharia Elétrica e de  
Computação. III. Título.

# Resumo

O trabalho apresenta um estudo de cancelamento de interferência para um sistema FH-CDMA. Primeiramente, os desempenhos de dois algoritmos de cancelamento de interferência propostos por Timor e Fiebig são comparados para um canal com ruído e desvanecimento seletivo em frequência. O algoritmo de Timor, que faz uso da estrutura algébrica de endereços dos usuários do sistema, mostrou possuir um compromisso entre desempenho e complexidade de implementação superior ao algoritmo de Fiebig (que é baseado na detecção multiusuário). Posteriormente, um esquema híbrido combinando codificação multinível e decodificação multiestágio com o algoritmo de Timor foi proposto e o seu desempenho foi analisado. E finalmente, esquemas que utilizam decodificação multiestágio iterativa foram também propostos. Todos os esquemas foram analisados através de simulação computacional num canal onde somente a interferência entre os usuários é considerada. O melhor desempenho foi obtido pelo esquema híbrido que combina cancelamento e decodificação multiestágio iterativa.

# Abstract

This work presents a study of interference cancellation for a FH-CDMA system. At first, the performances of two interference cancellation algorithms proposed by Timor and Fiebig are compared for a channel with noise and frequency selective fading. Timor's algorithm, which makes use of the algebraic structure of the users' addresses, showed a better tradeoff between performance and implementation complexity than Fiebig's algorithm (which is based on multiuser detection). Later, an hybrid scheme combining multilevel coding and multistage decoding with Timor's algorithm was proposed and its performance analysed. Finally, schemes that use iterative multistage decoding were also proposed. Computer simulation results for all the schemes were obtained in an only-interference channel. The hybrid scheme which combines cancellation and iterative multistage decoding showed to possess the best performance.

# Agradecimentos

- A Deus, por mais esta caminhada de sucesso em minha vida. Sua presença me confortou e me deu força nas horas difíceis deste trabalho.
- A minha noiva Rubia, pelo seu amor, carinho e compreensão nos momentos de ausência em que tive que me dedicar mais ao trabalho. E também à sua família pelo apoio que me deram.
- Aos meus pais, Paulo e Ana e minhas irmãs Ana Paula e Tatiana pela força e pelo Carinho que me deram para superar todos os obstáculos e poder realizar mais esta etapa em minha vida.
- Ao professor Jaime, pela dedicação e paciência desprendidas ao meu trabalho nestes últimos dois anos.
- Aos Profs. Drs. Bartolomeu F. U. Filho (EEL/CTC/UFSC) e Celso de Almeida(FEEC/UNICAMP), membros da Banca Examinadora.
- Aos amigos de Graduação Zemann, Rivaél, José, Paccola e Jeferson, que contribuíram muito para a realização deste trabalho.
- Aos colegas e funcionários da FEEC, pela amizade e companheirismo, que tornou agradável minha estadia em Campinas.
- Ao CNPQ pelo apoio financeiro através da bolsa de estudo.

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Conceitos e Definições</b>	<b>7</b>
2.1	Codificação de Bloco . . . . .	7
2.1.1	Códigos de Reed-Solomon . . . . .	11
2.2	Modulação on-off e Detecção Não-Coerente . . . . .	13
2.3	Sistemas Com Espalhamento Espectral . . . . .	16
2.3.1	Definições . . . . .	16
2.3.2	Modelo de um Sistema de Comunicação com Espalhamento Espectral . . . . .	16
2.3.3	Espalhamento Espectral por Seqüência Direta (DS-SS) . . . . .	17
2.3.4	Espalhamento Espectral por Saltos em Frequência (FH-SS) . . . . .	18
2.4	Construção Multinível . . . . .	19
2.4.1	Codificação Multinível e Decodificação Multiestágio . . . . .	20
<b>3</b>	<b>Um Sistema FH-CDMA</b>	<b>25</b>
3.1	Sistema FH-CDMA com Modulação on-off . . . . .	26
3.2	Modelos de Endereçamento . . . . .	29
3.3	Algoritmos de Cancelamento de Interferência . . . . .	31
3.3.1	Algoritmo de Cancelamento Proposto por Timor . . . . .	31
3.3.2	Algoritmo de Cancelamento Proposto por Fiebig . . . . .	38
3.4	Comparação dos Algoritmos . . . . .	47
<b>4</b>	<b>Sistema Proposto</b>	<b>49</b>
4.1	Descrição do Esquema de Construção Multinível para um Sistema FH-CDMA . . . . .	50
4.1.1	Desempenho do Sistema . . . . .	54
4.2	Um Esquema Híbrido de Decodificação . . . . .	55

4.2.1	Desempenho do Sistema . . . . .	57
4.3	Aplicação de Iterações no Esquema de Decodificação Multiestágio . . . . .	59
4.3.1	Esquema 1 : Esquema Híbrido e Decodificação no Primeiro Estágio Sem o Algoritmo de Cancelamento de Interferência . . . . .	59
4.3.2	Esquema 2 : Esquema Híbrido e Decodificação no Primeiro Estágio Com o Algoritmo de Cancelamento de Interferência . . . . .	60
4.3.3	Esquema 3 : Esquema Híbrido e uma Iteração Completa Sem o Algoritmo de Cancelamento . . . . .	60
4.3.4	Esquema 4 : Esquema Híbrido com Duas ou mais Iterações Sem o Algoritmo de Cancelamento . . . . .	61
4.3.5	Simulação dos Esquemas 1 a 4 e uma Comparação com o Esquema Híbrido . . . . .	61
4.4	Simulação dos Sistemas Utilizando Dois Códigos de RS (3, 1) . . . . .	61
<b>5</b>	<b>Conclusões</b>	<b>67</b>
<b>A</b>	<b>Esquemas de Endereçamento</b>	<b>71</b>
A.1	Sistema Síncrono . . . . .	71
A.2	Sistema Assíncrono . . . . .	72

# Lista de Figuras

2.1	Diagrama de blocos de um sistema com modulação on-off . . . . .	14
2.2	Modelo de um sistema com espalhamento espectral . . . . .	17
2.3	Ilustração do processo de geração de um sinal DS-SS (a) por multiplicação (b) por soma módulo-2 . . . . .	18
2.4	Ilustração do processo de geração de um sinal FH-SS. . . . .	19
2.5	Representação das partições em $GF(q)$ . . . . .	21
2.6	Diagrama de bloco do codificador para um código multinível . . . . .	22
2.7	Diagrama de bloco do decodificador multiestágio. . . . .	23
3.1	Transmissor de um sistema FH-CDMA. O diagrama de blocos indica as operações de processamento de sinal. As matrizes mostram as sequências de níveis lógicos ou frequências selecionadas para a transmissão. . . . .	27
3.2	Diagrama de blocos do receptor e as matrizes de sinais de um sistema FH-CDMA	28
3.3	Desempenho do algoritmo de Timor para canais com ruído e desvanecimento seletivo em frequência, $q=128$ e $L=7$ . . . . .	39
3.4	Desempenho do algoritmo de Timor para canais com ruído e desvanecimento seletivo em frequência, $q=256$ e $L=10$ . . . . .	39
3.5	Desempenho do algoritmo de Timor para canais com ruído e desvanecimento seletivo em frequência, $q=512$ e $L=11$ . . . . .	40
3.6	Um exemplo do algoritmo de decodificação de Fiebig com duas iterações e 6 usuários simultâneos no canal, . . . . .	42
3.7	Desempenho do algoritmo de Fiebig para canais com ruído e desvanecimento seletivo em frequência, $q=128$ e $L=7$ . . . . .	45
3.8	Desempenho do algoritmo de Fiebig para canais com ruído e desvanecimento seletivo em frequência, $q=256$ e $L=10$ . . . . .	46

3.9	Desempenho do algoritmo de Fiebig para canais com ruído e desvanecimento seletivo em frequência, $q=512$ e $L=11$ . . . . .	46
4.1	Descrição das classes laterais sobre $GF(q)$ . . . . .	54
4.2	Desempenho do esquema proposto para $q=7$ e $L=3,5,6$ . . . . .	55
4.3	Desempenho do esquema híbrido em relação ao esquema convencional para $q=7$ e $L=3,5,6$ . . . . .	58
4.4	Desempenho dos esquemas 1 a 4 para $q=7$ e $L=4$ . . . . .	62
4.5	Desempenho dos esquemas 1 a 4 para $q=7$ e $L=5$ . . . . .	62
4.6	Desempenho dos esquemas 1 a 4 para $q=7$ e $L=6$ . . . . .	63
4.7	Desempenho dos esquemas utilizando o código de repetição (3,1) para $q=7$ e $L=4$ . . . . .	64
4.8	Desempenho dos esquemas utilizando o código de repetição (3,1) para $q=7$ e $L=5$ . . . . .	65
4.9	Desempenho dos esquemas utilizando o código de repetição (3,1) para $q=7$ e $L=6$ . . . . .	65

# Capítulo 1

## Introdução

Nos últimos anos os sistemas de comunicações vêm apresentando uma grande evolução tecnológica e vários são os serviços de telecomunicações oferecidos. Os sistemas de telefonia móvel, em particular, têm se destacado muito neste contexto. Muito tem sido estudado, vários sistemas já foram testados na prática, outros estão em fase experimental de pesquisa. A elevada demanda para os serviços de telecomunicações é a razão para tanto esforço no sentido de prover avanços que os tornem cada dia mais atrativos economicamente e tecnologicamente mais versáteis.

Os sistemas que utilizam a tecnologia de espalhamento espectral através da técnica de acesso múltiplo por divisão de código, CDMA (do inglês, Code Division Multiple Access) têm sido objeto de estudo de vários pesquisadores demonstrando algumas propriedades vantajosas em comunicações móveis e redes locais. A rejeição à interferência multipercurso, a baixa probabilidade de interceptação e a privacidade na comunicação são algumas destas propriedades que fazem do CDMA um forte candidato à tecnologia da terceira geração.

A técnica mais comum para se gerar sinais espalhados no espectro em CDMA é conhecida como seqüência direta, DS (do inglês, Direct Sequence). Nesta técnica, o bit de informação é multiplicado diretamente por uma seqüência pseudo-aleatória de taxa muitas vezes superior a taxa de bit, de tal forma que o sinal transmitido possui uma largura de faixa significamente elevada. Para se recuperar o sinal original, multiplica-se o sinal recebido composto dos sinais de todos os usuários pela seqüência pseudo-aleatória do usuário cujo sinal se deseja recuperar.

Se a correlação entre a seqüência do usuário desejado e as seqüências dos outros usuários for mínima, a multiplicação dos sinais destes usuários pela seqüência original resultará em um sinal aleatório de média aproximadamente nula, que para o sinal desejado será considerado como interferência, assim somente uma pequena parte da potência dos sinais interferentes será introduzida no sinal original. Por outro lado, se uma ou mais seqüências não forem ortogonais com a seqüência do usuário desejado, a interferência dos outros usuários pode tornar-se excessiva, se os níveis de potência dos sinais destes usuários forem suficientemente maiores do que o nível de potência do usuário desejado. Este problema, chamado de próximo versus distante em comunicações multiusuários, geralmente acontece em transmissões móvel-para-base em que a potência recebida dos usuários próximos a estação base se torna maior do que a dos usuários distantes. Isto pode ser resolvido aplicando um algoritmo de controle de potência para que todos usuários sejam recebidos com a mesma potência média na estação base, porém a implementação de um algoritmo de controle de potência é muito complexo. Uma outra técnica de espalhamento espectral em CDMA que utiliza saltos em freqüência, FH (do inglês, Frequency Hopped), tem gerado um grande interesse principalmente pela sua capacidade de combater o desvanecimento seletivo e ser imune ao problema próximo versus distante. Nesta técnica, o bit de informação é multiplicado por uma seqüência pseudo-aleatória e muda periodicamente em freqüência durante o intervalo de transmissão  $T$ . No receptor é realizada a operação inversa, ou seja, multiplica-se o sinal recebido composto dos sinais de todos os usuários pela seqüência pseudo-aleatória.

A diferença fundamental entre o FH e o DS está na ocupação em freqüência. O sistema DS ocupa toda a largura de faixa  $W$ , para transmitir seus sinais durante um intervalo de transmissão. No sistema FH, a largura de faixa é subdividida em subbandas onde durante um intervalo de transmissão o sinal transmitido ocupa uma ou mais subbandas. Para se entender melhor, supõe-se que um sistema FH está transmitindo em uma determinada subbanda  $M_1 \ll W$ , durante o primeiro intervalo de chip  $T_c \ll T$ . Um sistema DS transmitindo no mesmo intervalo de chip, espalha a potência do sinal transmitido em toda a largura de faixa

resultando em uma potência transmitida na subbanda  $M_1$  menor do que no sistema FH. É importante lembrar, que ambos sistemas transmitirão a mesma potência em toda a largura de faixa  $W$ . Esta característica dos sinais FH é que os tornam resistentes ao desvanecimento seletivo em frequência e também imunes ao problema próximo versus distante, ou seja, o sinal recebido de um usuário distante da estação base será recebido, mesmo que um usuário próximo a estação base esteja transmitindo seu sinal, já que estes usuários provavelmente transmitirão em diferentes frequências.

No sistema FH, dois tipos de esquemas de modulação são normalmente utilizados : o OOK(do inglês, on-off keying) e o FSK(do inglês, frequency-shift keying). No OOK, as mensagens são enviadas utilizando tons senoidais que representam o estado “on ”. O acesso múltiplo neste esquema de modulação é obtido dividindo cada mensagem transmitida em intervalos de tempo,  $T_c$ , onde em cada intervalo as mensagens são saltadas para uma das frequências disponíveis utilizando seqüências de saltos aleatórias [Goo80], ou seqüências de saltos estruturadas algebricamente [Ein80]. No esquema FSK, as mensagens são enviadas utilizando  $q$  M-FSK canais. As mensagens moduladas são transmitidas usando sinais FSK ortogonais, e então saltadas para um dos  $q$  M-FSK canais.

Em um sistema CDMA, ao se aumentar o número de usuários no sistema, aumenta-se também a interferência, de forma que o sistema é limitado em número de usuários. No sistema FH-CDMA, este aumento no número de usuários faz com que os sinais transmitidos se choquem em determinadas frequências, havendo assim uma degradação no desempenho do sistema. Desta forma, foram realizados nos últimos anos vários estudos no intuito de se diminuir o grau desta interferência bem como aumentar a eficiência espectral do sistema. Em princípio, foi proposta a utilização de modelos de endereçamento com boas propriedades algébricas em que houvesse o mínimo de correlação entre as seqüências de saltos transmitidas pelos usuários como mostrados em [Ein80]. Algoritmo de cancelamento de interferência baseados nos modelos de endereçamento algébrico [Tim80] [Fie96] também foram propostos e apresentaram uma melhora considerável no desempenho do sistema. Esquemas híbridos

combinando algoritmos de cancelamento de interferência com esquemas de decodificação com códigos corretores de erro mostraram uma melhora no desempenho do sistema ainda mais acentuada [Mab94].

Um esquema que tem sido aplicado com sucesso em diferentes sistemas de comunicação é baseado na codificação multinível juntamente com a decodificação multiestágio. Este esquema permite uma grande flexibilidade na escolha da taxa de informação bem como mantém um excelente compromisso entre desempenho e complexidade de implementação. A aplicação deste esquema de codificação multinível e decodificação multiestágio num sistema que utiliza saltos em frequência (FH) foi proposta em [Mar97], porém o desempenho do esquema não foi obtido.

A proposta deste trabalho é apresentar primeiramente uma comparação entre dois algoritmos de cancelamento de interferência [Tim80][Fie96] em termos de desempenho e complexidade de implementação. A análise é realizada em um canal com ruído e desvanecimento seletivo em frequência. Posteriormente são descritos e apresentados os resultados de simulação do desempenho do esquema de codificação multinível e decodificação multiestágio aplicado em um sistema FH-CDMA, para um canal onde somente a interferência entre os usuários é considerada. Um esquema híbrido combinando o esquema de cancelamento de interferência proposto por Timor [Tim80] com o esquema de codificação multinível e decodificação multiestágio também é apresentado e simulado novamente em canais onde somente a interferência entre os usuários é considerada. Finalmente são propostos esquemas baseados em iterações do decodificador multiestágio buscando melhorar gradativamente o desempenho do esquema original sem obviamente aumentar excessivamente a complexidade de implementação.

A tese é organizada da seguinte maneira : no capítulo 2, como fundamentação para o entendimento do trabalho, são apresentados primeiramente alguns conceitos de codificação de bloco e códigos de Reed-Solomon. Posteriormente é abordado mais detalhadamente o esquema de modulação on-off com detecção não-coerente e os sistemas com espalhamento espectral DS e FH. Ainda no capítulo 2, uma descrição geral do processo de codificação

multinível e decodificação multiestágio é apresentada.

No capítulo 3, é apresentado o sistema FH-CDMA com modulação on-off, bem como modelos de endereçamento para transmissão síncrona e não-síncrona. Em seguida o capítulo descreve e compara dois algoritmos de cancelamento de interferência [Tim80] [Fie96], em canais com ruído aditivo e desvanecimento seletivo em frequência. O desempenho destes dois algoritmos é obtido analiticamente e algumas comparações entre os mesmos são realizadas.

O capítulo 4 apresenta o sistema proposto de codificação multinível e decodificação multiestágio aplicado a um sistema FH-CDMA, bem como um esquema híbrido combinando o esquema de codificação multinível com o algoritmo de cancelamento proposto por Timor [Tim80] analisado no capítulo 3. Também são apresentados esquemas alternativos com iterações buscando melhorar a desempenho do sistema. Os resultados obtidos através de simulações computacionais são apresentados neste capítulo.

O trabalho termina com o capítulo 5, onde é dedicado um espaço para as conclusões que foram obtidas com o estudo e perspectivas de trabalhos futuros.

## Capítulo 2

# Conceitos e Definições

Este capítulo é destinado a apresentar alguns conceitos e definições necessários para o entendimento do trabalho. Em sua primeira parte as definições de codificação em bloco e códigos de Reed-Solomon (*RS*) são descritas, a segunda parte descreve o esquema de modulação on-off utilizado em sistemas com saltos em frequência(Frequency-Hopped). A última parte descreve a construção multinível.

### 2.1 Codificação de Bloco

Um código de bloco consiste de um conjunto de vetores de comprimento fixo, onde cada vetor no código é chamado de palavra-código e o seu comprimento, determinado pelo número de elementos do vetor, é denotado por  $n$ . Os elementos de uma palavra código são selecionados de um alfabeto de  $q$ -elementos. Quando o alfabeto do código consiste de 2 elementos, 0 e 1, o código é binário e os elementos da palavra-código são denominados *bits*. Quando  $q > 2$ , o código é não binário. É interessante esclarecer também que quando  $q$  é um potência de 2, ou seja,  $q = 2^b$  onde  $b$  é um inteiro positivo, cada elemento  $q$ -ário tem um representação equivalente de  $b$  bits, e assim, um código não-binário de comprimento  $N$ , pode ser mapeado num código binário de comprimento de bloco  $n = bN$ .

Considere um código binário possuindo  $M = 2^k$  palavras-código ( $k < n$ ). Pode-se associar um mapeamento um-para-um do conjunto de seqüências binárias de comprimento  $k$  no conjunto de palavras-código. Os  $k$  bits são denominados bits de informação e o código

resultante é referenciado como um código  $(n, k)$ , com taxa de  $k$  bits de informação por  $n$  bits codificados, isto é,  $R_c = \frac{k}{n}$ .

No caso geral, em um código com alfabeto  $q$ -ário, existem  $q^k$  possíveis palavras-código, onde de forma similar ao código binário elas são selecionadas de  $q^n$  seqüências para transmitir blocos de  $\log_2 q^k$  bits de informação. Os elementos do alfabeto  $q$ -ário de um código de bloco podem ser associados a um corpo algébrico. Esta associação permite simplificar as funções de codificação e decodificação, pois as mesmas envolveriam apenas operações de adição e multiplicação realizadas no corpo algébrico [Mac78]. Os códigos são construídos de corpos com um número finito de elementos, onde um corpo de  $q$  elementos é um corpo de Galois e denotado por  $GF(q)$ . Todo corpo possui um elemento nulo e um elemento unitário, e todas as operações envolvendo um corpo sobre  $GF(q)$  com  $q$  primo são realizadas módulo- $q$ .

Um importante parâmetro de um código de bloco  $(n, k)$  é a sua distância de Hamming mínima,  $d_{\min}$ . Sejam duas palavras-código  $C_i$  e  $C_j$ , a distância de Hamming  $d_{ij}$  entre elas é o número de elementos ou posições em que elas diferem. O menor valor de  $d_{ij}, i \neq j$  é a distância mínima do código.

Além de caracterizar os códigos como sendo binários e não binários, pode-se também classificar os códigos como lineares e não-lineares. A maioria dos códigos de blocos pertence à classe dos códigos lineares, ou seja, códigos que são sub-espacos vetoriais do espaço consistindo de todas a  $n$ -uplas  $q$ -árias [Mac78].

No processo de codificação de um código de bloco linear  $(n, k)$  é importante adotar a convenção de que as palavras-código são representadas como vetores linha. Desta forma, considerando  $x_{m1}, x_{m2}, \dots, x_{mk}$  como os  $k$  bits de informação, cada palavra-código na entrada do codificador pode ser representada da seguinte forma

$$X_m = [x_{m1} \quad x_{m2} \quad \dots \quad x_{mk}] \quad (2.1)$$

e a representação desta palavra-código na saída do codificador é expressa como

$$C_m = [c_{m1} \quad c_{m2} \quad \dots \quad c_{mn}] \quad (2.2)$$

Esta operação de codificação realizada, por exemplo, em um codificador de bloco binário pode ser representada pelo conjunto de equações da forma

$$c_{mj} = x_{m1}g_{1j} + x_{m2}g_{2j} + \dots + x_{mk}g_{kj} \quad j = 1, 2, \dots, n. \quad (2.3)$$

onde  $g_{ij} = 0$  ou  $1$ . Estas equações lineares acima podem ser representadas também em uma forma matricial como :

$$C_m = X_m G. \quad (2.4)$$

Onde  $G$  é chamada de matriz geradora do código tendo a seguinte forma

$$G = [I_k \mid P] = \left[ \begin{array}{cccc|cccc} 1 & 0 & 0 & \dots & 0 & p_{11} & p_{12} & \dots & p_{1n-k} \\ 0 & 1 & 0 & \dots & 0 & p_{21} & p_{22} & \dots & p_{2n-k} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & p_{k1} & p_{k2} & \dots & p_{kn-k} \end{array} \right] \quad (2.5)$$

$I_k$  é a matriz identidade de dimensão  $k$  que simplesmente reproduz o vetor mensagem  $X_m$  e  $P$  é uma matriz  $k \times (n - k)$  que gera os  $n - k$  bits de redundância ou bits de paridade do código.

Um dos mais importantes subconjuntos dos códigos lineares são os códigos cíclicos, estes códigos são caracterizados por apresentar a seguinte propriedade cíclica :

✓ Se  $C = [c_{n-1}c_{n-2} \dots c_1c_0]$  é uma palavra-código de um código cíclico, então  $[c_{n-2}c_{n-3} \dots c_0c_{n-1}]$  obtida do deslocamento cíclico dos elementos de  $C$ , é também uma palavra-código. Assim, todos os deslocamentos cíclicos de  $C$  são palavras-código.

Uma das conseqüências desta propriedade cíclica é a possibilidade de implementar códigos de bloco longos com um grande número de palavras-código num sistema de comunicação.

Ao se tratar com códigos cíclicos, é conveniente associar a cada palavra-código  $C = [c_{n-1}c_{n-2} \dots c_1c_0]$  um polinômio  $C(p)$  de grau  $\leq n - 1$ , definido como

$$C(p) = c_{n-1}p^{n-1} + c_{n-2}p^{n-2} + \dots + c_1p + c_0 \quad (2.6)$$

onde para um código binário, os coeficientes do polinômio podem ser 0 ou 1. Então se  $C(p)$  representa uma palavra-código de um código cíclico,  $p^i C(p) \bmod(p^n + 1)$  é também uma palavra-código do código cíclico. Assim podemos escrever

$$p^i C(p) = Q(p)(p^n + 1) + C_i(p) \quad (2.7)$$

onde o polinômio resto  $C_i(p)$  representa uma palavra-código do código cíclico e  $Q(p)$  é o quociente. Pode-se gerar uma palavra-código usando um polinômio gerador denotado por  $g(p)$  de grau  $n - k$  tendo a seguinte forma

$$g(p) = p^{n-k} + g_{n-k-1}p^{n-k-1} + \dots + g_1p + x_0 \quad (2.8)$$

Definindo os  $k$  bits de informação  $[x_{k-1}x_{k-2} \dots x_1x_0]$  como um polinômio mensagem  $X(p) = x_{k-1}p^{k-1} + x_{k-2}p^{k-2} + \dots + x_1p + x_0$ , a multiplicação destes dois polinômios produz um outro polinômio de grau menor ou igual a  $n - 1$ , que representa uma palavra-código.

Uma grande classe dos códigos cíclicos que incluem ambos alfabetos binários e não-binários são os chamados códigos BCH(Bose-Chaudhuri-Hocquenghem). Estes códigos podem ser construídos com parâmetros

$$\begin{aligned}
 n &= 2^m - 1 & (2.9) \\
 n - k &\leq mt \\
 d_{\min} &= 2t + 1
 \end{aligned}$$

onde  $m$  ( $m \geq 3$ ) e  $t$  são inteiros positivos arbitrários. Os polinômios geradores para estes códigos podem ser construídos com fatores de  $p^{2^m-1} + 1$ .

A classe binária destes códigos fornece ao sistema de comunicação uma grande quantidade de comprimentos e taxas de código. No caso não-binário, estão incluídos os poderosos códigos de Reed-Solomon que serão descritos com mais detalhes na próxima seção.

### 2.1.1 Códigos de Reed-Solomon

Os códigos de Reed-Solomon, ou códigos *RS*, são considerados uma importante subclasse dos códigos de bloco não-binários BCH sobre  $GF(q)$ , possuindo a propriedade de que o comprimento do código  $n$  é igual a  $q - 1$ . Estes códigos têm uma importância prática e teórica considerável, por várias razões, dentre elas podemos citar algumas :

✓ São códigos usados quando há a necessidade de se ter um comprimento menor do que o tamanho do corpo.

✓ São códigos convenientes para construção de outros códigos, por exemplo, podem ser mapeados em códigos binários com máxima distância mínima. Também são usados na construção de códigos concatenados e códigos Justesen [capítulo 10, [Mac78]].

✓ São úteis para correção de erros em surto.

Um código *RS* sobre  $GF(q)$  de comprimento  $q - 1$  e distância de projeto dada por  $d$ , tem como polinômio gerador

$$g(x) = (x - \alpha^{m_0})(x - \alpha^{m_0+1}) \dots (x - \alpha^{m_0+d-2}) \quad (2.10)$$

onde  $m_0$  é um inteiro arbitrário escolhido entre 0 e 1 e  $\alpha$  é o elemento primitivo de  $GF(q)$ .

A distância mínima  $d_{\min}$  de um código *RS*, é dada por [Mac78]

$$d_{\min} = n - k + 1 \quad (2.11)$$

onde  $n$  é o comprimento do bloco e  $k$  sua dimensão.

Os códigos em que a distância mínima é igual a  $n - k + 1$  são chamados códigos de máxima distância de separação (MDS), ou simplesmente códigos máximos. Todo código  $RS$  é um código de máxima distância de separação.

Um outro importante parâmetro de um código  $RS$  é a distribuição de peso das palavras-código. Em geral, cada palavra-código tem seu próprio peso e o conjunto de todos os pesos constitui a distribuição de pesos. A distribuição de peso  $\{A_i\}$  das palavras-código de um código  $MDS$  definido sobre  $GF(q)$ , tendo como comprimento de bloco  $n$ , e distância mínima  $d$  é dada por [Mac78] :

$$A_i = \binom{n}{i} (q-1) \sum_{j=0}^{i-d} (-1)^j \binom{i-1}{j} q^{i-d-j}, \quad i = d, d+1, \dots, n. \quad (2.12)$$

Ao se formar um código  $RS$  com  $m_0 = 1$ , pode-se acrescentar a todas as palavras um símbolo de paridade, aumentando assim o comprimento do bloco de um, sem aumentar o número de palavras-códigos. O novo código é chamado de um código de  $RS$  estendido. Sua distância mínima em particular, também é aumentada de um. Um código  $RS$  estendido é um código MDS e portanto, sua distribuição de peso também é dada pela equação (2.12).

Como os códigos  $RS$  são códigos cíclicos, eles podem ser codificados por qualquer um dos métodos de codificação de códigos cíclicos. A seguir será descrito um método simples de codificação (que é o método de Reed e Solomon).

Assim seja  $\bar{C} = (u(1), u(\alpha), \dots, u(\alpha^{n-1})) = (c_0, c_1, \dots, c_{n-1})$ , uma palavra-código de um código  $RS$ , onde  $u(z) = \sum_{i=0}^{K-1} u_i z^i$  e  $\bar{u} = (u_0, u_1, \dots, u_{K-1})$  para  $u_i \in GF(q)$ . É possível mostrar que a palavra-código  $\bar{C}$  pertence a um código BCH, por verificar que  $C(x) = \sum_{i=0}^{n-1} c_i x^i$  possuindo  $\alpha, \alpha^2, \dots, \alpha^{d-1}$  como zeros.

Fazendo  $K = k$ ,  $\alpha = \beta$ ,  $u_i = x_i$  para  $i$  de 0 a  $k - 1$ , tem-se que cada componente de um código  $RS$  é dada pelo seguinte conjunto de equações :

$$\begin{aligned} C_0 &= x_0 + x_1 + x_2 + \dots + x_{k-1} \\ C_1 &= x_0 + x_1\beta + x_2\beta^2 + \dots + x_{k-1}\beta^{k-1} \\ &\vdots = \vdots \quad \vdots \quad \quad \quad \vdots \\ C_{n-1} &= x_0 + x_1\beta^{(n-1)} + x_2\beta^{(n-1)2} + \dots + x_{k-1}\beta^{(n-1)(k-1)} \end{aligned} \quad (2.13)$$

As equações acima podem ser reescritas de uma forma mais geral como abaixo

$$\begin{aligned} \bar{C} &= (c_0, c_1, \dots, c_{n-1}) = x_0 \cdot \bar{1} + x_1 \cdot \bar{\beta}^{(1)} + x_2 \cdot \bar{\beta}^{(2)} + \dots + x_{k-1} \cdot \bar{\beta}^{(k-1)} \\ x_i &\in GF(q), i = 0, 1, \dots, k-1. \end{aligned} \quad (2.14)$$

onde  $\bar{\beta}^{(i)} = (1, \beta^i, \beta^{2i}, \dots, \beta^{(n-1)i})$ . Assim  $\bar{C}$  é uma palavra-código  $RS$  e a equação acima é o método de codificação destes códigos.

## 2.2 Modulação on-off e Detecção Não-Coerente

Num sistema de comunicação onde se utiliza o esquema de modulação do tipo on-off, as mensagens são enviadas utilizando-se um tom senoidal que representa o estado “on”. Este tom é modulado por um pulso em banda base,  $u(t)$ ,  $0 \leq t \leq T$ . Assim a forma de onda no canal é dada da seguinte forma :

$$s(t) = \begin{cases} u(t) \cos 2\pi f_c t, & \text{se o pulso foi enviado} \\ 0 & \text{caso contrário} \end{cases}, \quad (2.15)$$

Na entrada do receptor, o sinal  $u(t)$  é somado com o ruído branco gaussiano de densidade espectral  $2N$ , possuindo componentes em fase,  $x(t)$ , e em quadratura,  $y(t)$ . O ruído pode ser representado por :

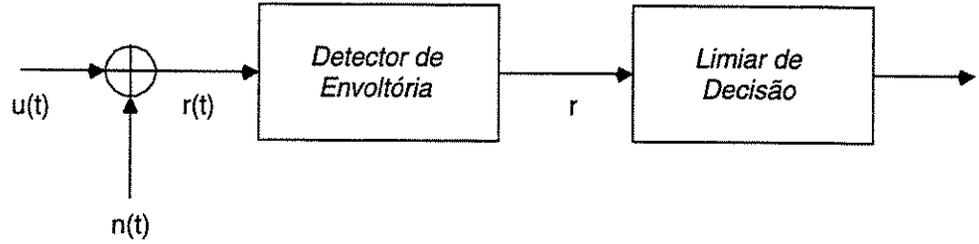


Figura 2.1: Diagrama de blocos de um sistema com modulação on-off

$$n(t) = x(t) \cos 2\pi f_c t + y(t) \text{ sen} 2\pi f_c t \quad (2.16)$$

O método mais comum de detecção destes sinais é a detecção não-coerente, através da detecção da envoltória do sinal,  $r(t) = \sqrt{[u(t) + x(t)]^2 + y^2(t)}$ . A função densidade de probabilidade para a envoltória em  $t = kT$  [Ste66] :

$$p(r) = \frac{r}{N} \exp\left(-\frac{(r^2 + z^2)}{2N}\right) I_0\left(\frac{rz}{N}\right) \quad (2.17)$$

onde  $z^2 = \int_{-\infty}^{+\infty} u^2(t) dt$  é a energia de  $u(t)$  e  $N$  é o valor médio do ruído e  $I_0(x)$  é a função de Bessel de ordem zero definida como :

$$I_0(x) = \sum_{i=0}^{\infty} \frac{\left(\frac{x}{2}\right)^{2i}}{i! \Gamma(i+1)} \quad (2.18)$$

Na entrada do receptor é feita uma comparação de  $r$  com o limiar normalizado  $\lambda$ , como descrito no diagrama de blocos da Figura 2.1.

Assim, se o sinal ultrapassa o limiar de decisão, ou seja,  $r > \lambda$ , a decisão é de que foi transmitido o pulso e caso contrário a decisão é de que não foi transmitido nenhum pulso.

Como há somente estas duas situações na transmissão, se for transmitido o pulso, a probabilidade de erro é dada por :

$$\begin{aligned}
P_{e1} &= \text{prob}(r < \lambda) = \int_0^\lambda p(r) dr = 1 - \int_\lambda^\infty p(r) dr \\
&= 1 - \int_\lambda^\infty \frac{r}{N} \exp\left(-\frac{(r^2 + z^2)}{2N}\right) I_0\left(\frac{rz}{N}\right) dr \\
&= 1 - Q\left(\sqrt{2\gamma}, \lambda_0\right)
\end{aligned} \tag{2.19}$$

onde  $\lambda_0 = \frac{\lambda}{\sqrt{N}}$  é o limiar de decisão normalizado e  $\gamma = \frac{z^2}{2N}$  é a relação sinal-ruído na saída do filtro no instante de amostragem.  $Q(a, b)$  é uma função de Marcum,  $Q(a, b) = \int_b^\infty \exp\left(-\frac{a^2+x^2}{2}\right) I_0(ax) dx$ .

Quando nenhum pulso foi transmitido a probabilidade de erro ocorre quando  $r > \lambda$  e é :

$$\begin{aligned}
P_{e2} &= \text{prob}(r > \lambda) = \int_\lambda^\infty \frac{r}{N} \exp\left(-\frac{r^2}{2N}\right) dr = Q(0, \lambda_0) \\
&= \exp\left(-\frac{\lambda_0^2}{2}\right)
\end{aligned} \tag{2.20}$$

Assumindo que os estados “on” e “off” são equiprováveis, a probabilidade média de erro é :

$$P_e = \frac{1}{2}P_{e1} + \frac{1}{2}P_{e2} = \frac{1}{2} \left[ 1 - Q(\sqrt{2\gamma}, b_0) \right] + \frac{1}{2} \exp\left(-\frac{b_0^2}{2}\right) \tag{2.21}$$

Observa-se que para cada relação sinal-ruído há um limiar de decisão normalizado que produz uma probabilidade de erro mínima. Este limiar foi calculado em [Ste66] e é dado por :

$$\hat{\lambda}_0 = \sqrt{2N \left(1 + \frac{1}{\gamma}\right) \ln(1 + \gamma)} \tag{2.22}$$

## 2.3 Sistemas Com Espalhamento Espectral

### 2.3.1 Definições

Os sinais espalhados espectralmente em comunicações digitais têm por característica uma largura de faixa,  $W$ , muito maior do que a taxa de informação  $R$  (onde  $W$  independe de  $R$ ). Isto significa que o fator de expansão de largura de faixa  $B_e = \frac{W}{R}$  é muito maior que a unidade. Uma outra importante característica no projeto destes sinais é o emprego da pseudo-aleatoriedade, a qual faz com que o sinal se pareça similar a um ruído aleatório, assim dificultando a demodulação por outros receptores. Esta característica foi fundamental para que estes sinais espalhados espectralmente fossem primeiramente utilizados para fins militares. Atualmente esta característica está sendo explorada em sistemas de telefonia celular, garantindo um certo grau de sigilo e segurança na comunicação.

Nos sistemas de telefonia celular, onde vários usuários compartilham a mesma largura de faixa e os mesmos intervalos de tempo, estes sinais são fortemente influenciados pela interferência. Assim, num dado instante, vários usuários podem transmitir informações simultaneamente para seus correspondentes receptores. Assumindo que todos os usuários empregam um código para codificação e decodificação de suas respectivas seqüência de informações, os sinais transmitidos neste espectro comum podem ser distinguidos pelos seus respectivos códigos ou seqüências de endereços. Desta forma, um receptor particular pode recuperar a informação transmitida conhecendo o endereço do transmissor. Este tipo de técnica, que permite múltiplos usuários compartilhando um canal comum para transmissão de informação, é chamado de *Acesso Múltiplo por Divisão de Código (CDMA)*.

### 2.3.2 Modelo de um Sistema de Comunicação com Espalhamento Espectral

A Figura 2.2 mostra o diagrama de bloco dos elementos básicos de um sistema de comunicação com espalhamento espectral. O codificador/decodificador e o modulador/demodulador são elementos básicos para o sistema. Somando-se a estes elementos, o sistema possui dois idênticos geradores de endereços que fazem interface com o modulador na transmissão e com

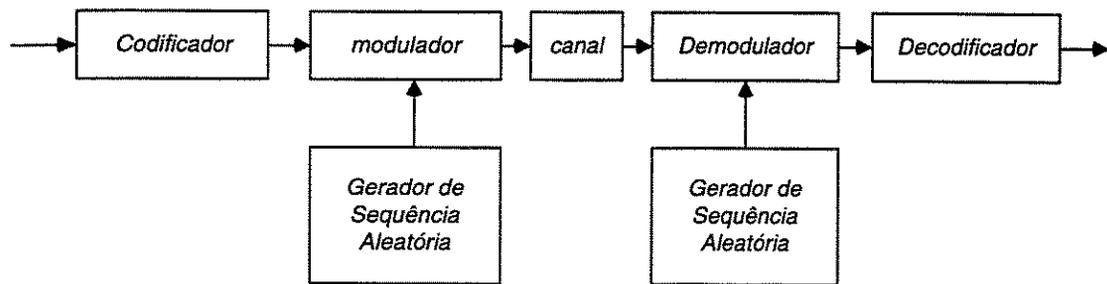


Figura 2.2: Modelo de um sistema com espalhamento espectral

o demodulador na recepção. Estes geram um endereço que é somado com o sinal no modulador e removido do sinal recebido no demodulador. A sincronização do endereço gerado no receptor com o endereço embutido no sinal recebido é necessária para a demodulação do sinal.

Dois tipos de modulação podem ser empregados neste sistema : PSK e FSK. O PSK é apropriado para aplicações onde a coerência de fase entre o sinal transmitido e o sinal recebido pode ser mantida por longos intervalos de tempo. Por outro lado, O FSK é apropriado para os casos onde a coerência de fase não pode ser mantida devido aos efeitos variantes no tempo do canal de comunicação. O endereço gerado no modulador em conjunção com a modulação PSK, trocando as fases dos sinais aleatoriamente é chamada de Sequência Direta (DS). Quando usado em conjunção com a modulação FSK binária ou  $M$ -ária ( $M > 2$ ), o endereço escolhe a frequência a ser transmitida pseudo-aleatoriamente. Este tipo de sinal resultante é chamado Frequency-Hopped (FH).

### 2.3.3 Espalhamento Espectral por Sequência Direta (DS-SS)

É uma técnica de espalhamento espectral no qual os símbolos de informação são multiplicados por um endereço, no caso em que a informação e o endereço são bipolares  $\{\pm 1\}$ . Pode-se também, somar os símbolos de informação unipolar  $\{0, 1\}$  com um endereço também unipolar, esta soma é realizada em módulo 2. O resultado de qualquer uma das operações modula uma portadora senoidal, normalmente em fase (PSK). O vetor endereço possui taxa muitas vezes superior a taxa de bits originais, de tal forma que o espectro resultante possui

uma largura correspondentemente elevada. A Figura 2.3 ilustra uma possível implementação do processo.

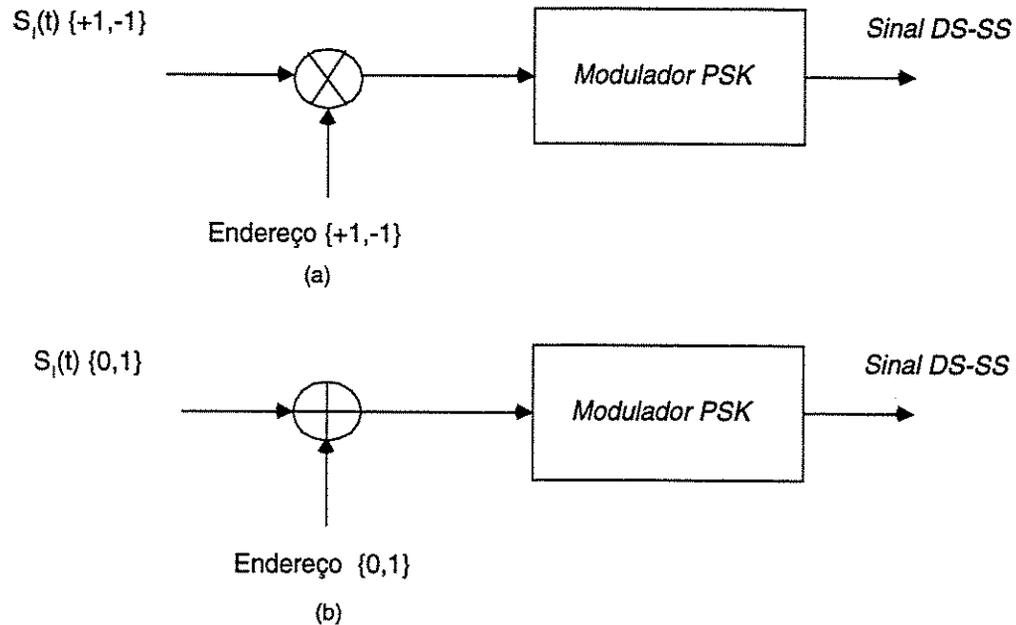


Figura 2.3: Ilustração do processo de geração de um sinal DS-SS (a) por multiplicação (b) por soma módulo-2

#### 2.3.4 Espalhamento Espectral por Saltos em Frequência (FH-SS)

Em um sistema com espalhamento espectral onde se utiliza saltos em frequência (do inglês, Frequency Hopped), a largura de faixa é subdividida em um grande número de slots de frequência. A frequência do sinal de informação modulado muda periodicamente. Durante um intervalo de transmissão, o sinal transmitido ocupa um dos slots de frequência disponíveis, depois deste intervalo de tempo o sinal salta para outro (ou possivelmente o mesmo) slot de frequência. A escolha destes slots a cada intervalo de transmissão é feita de acordo com o endereço gerado no transmissor [Pro95][Pra98].

Dentro do sistema de saltos em frequência é feita uma distinção entre duas técnicas, baseada na taxa do vetor endereço. Se o vetor endereço possui uma taxa muitas vezes superior à taxa de bits de informação, tem-se a implementação de um sistema FH-SS rápido (fast frequency hopping spread spectrum system). Neste caso, cada bit é transmitido em diferentes

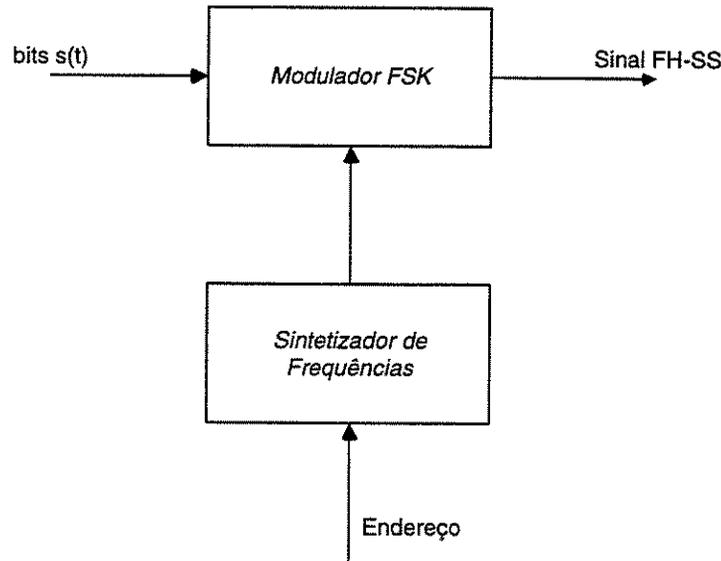


Figura 2.4: Ilustração do processo de geração de um sinal FH-SS.

freqüências durante o intervalo de transmissão  $T$ . Por outro lado, se vetor endereço possui taxa inferior a taxa de bits de informação, tem-se a implementação de um sistema FH-SS lento (slow frequency hopping spread spectrum system). Neste caso, muitos símbolos são transmitidos sem haver saltos de freqüências devidos ao vetor endereço. A Figura 2.4 ilustra um possível implementação do processo.

## 2.4 Construção Multinível

O particionamento de um conjunto de seqüências sobre  $GF(q)$  é uma técnica utilizada para se construir códigos de bloco multiníveis e códigos de treliça multiníveis sobre  $GF(q)$  [Cos92]. Esta técnica se difere dos códigos  $q$ -ários simples devido a utilização de códigos componentes que podem ser códigos de bloco ou convolucionais  $q$ -ários. Esta característica permite aos códigos multiníveis ter um melhor compromisso de distância mínima, taxa de informação e complexidade de decodificação.

A maior vantagem dos códigos multiníveis é a possibilidade de mais opções de decodificação, por exemplo, algoritmos de decodificação multiestágio e algoritmos de decodificação de Viterbi ou até mesmo uma combinação dos dois algoritmos podem ser empregadas, de-

pendendo exclusivamente dos códigos componentes utilizados.

A construção de códigos multiníveis e um algoritmo de decodificação multiestágio são abordadas com mais detalhes na próxima seção.

### 2.4.1 Codificação Multinível e Decodificação Multiestágio

Codificação multinível, que primeiro foi denotada por “Códigos Concatenados Generalizados”, se baseia na utilização de códigos componentes. Através desta construção, podemos controlar as distâncias de Hamming de códigos componentes externos e de códigos internos, garantindo um melhor compromisso entre distância de Hamming, taxa de informação e complexidade na decodificação. Basicamente, uma cadeia de partições é formada de um conjunto  $A_1, A_2, \dots, A_{m+1}$ , onde  $A_i$  é um código linear sobre  $GF(q)$  de comprimento  $n_1$ . Estas partições possuem a seguinte característica :

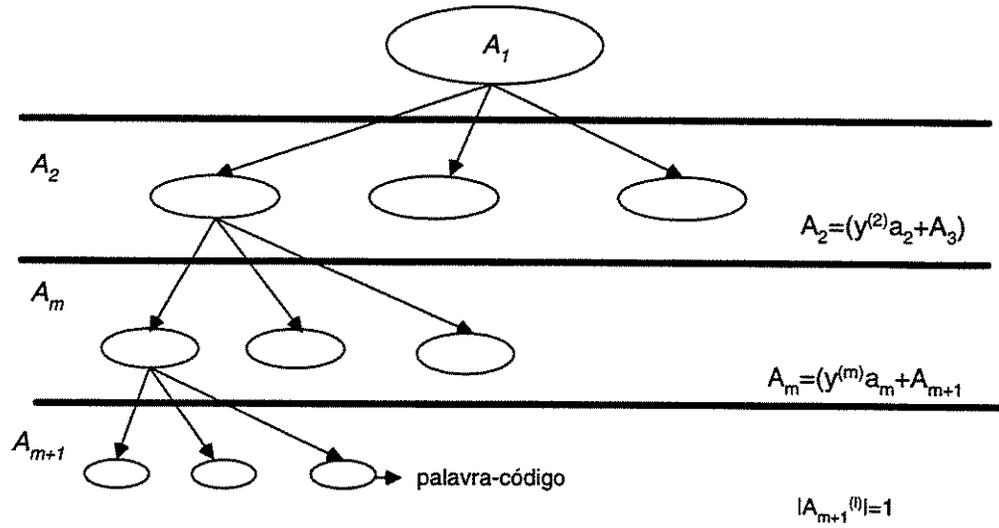
$$\sqrt{A_{i+1} \subset A_i, |A_i/A_{i+1}| = q, \text{ para } i = 1, 2, \dots, m \text{ e } m \leq n_1.}$$

Por simplicidade assume-se que  $A_{m+1} = \{0\}$ , possuindo distância de Hamming mínima igual a infinito. Devido a linearidade dos códigos, é possível afirmar que as partições podem ser expressas da seguinte maneira

$$A_i = \bigcup_{y^{(i)} \in GF(q)} (y^{(i)}\bar{a}_i + A_{i+1}), i = 1, 2, \dots, m \quad (2.23)$$

onde  $\bar{a}_i \in A_i, \bar{a}_i \notin A_{i+1}$ , é um representante “líder” de uma classe lateral de  $A_{i+1}$  em  $A_i$ . Esta descrição de classes laterais será abordada mais detalhadamente no capítulo 4. Utilizando a terminologia dos códigos concatenados generalizados, estes conjuntos  $A_i$  ( $i = 1, 2, \dots, m$ ) são denominados códigos internos e cada palavra-código em  $A_i$  é representada pelo conjunto de números  $y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(m)}$ . As seqüências de partições  $q$ -árias pode ser vista de acordo com a Figura 2.5.

O código multinível é formado utilizando  $m$  códigos  $q$ -ários  $C_i$  ( $i = 1, 2, \dots, m$ ) de comprimento  $n_2$  como os códigos componentes externos, como pode ser visto na Figura 2.6. Agrupa-se então cada palavra-código destes códigos formando-se uma matriz,  $Y$ ,  $m \times n_2$  :

Figura 2.5: Representação das partições em  $GF(q)$ .

$$Y = \begin{pmatrix} y_1^{(1)} & y_2^{(1)} & \cdots & y_{n_2}^{(1)} \\ y_1^{(2)} & y_2^{(2)} & \cdots & y_{n_2}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ y_1^{(m)} & y_2^{(m)} & \cdots & y_{n_2}^{(m)} \end{pmatrix} \quad (2.24)$$

onde cada linha da matriz  $(y_1^{(i)}, y_2^{(i)}, \dots, y_{n_2}^{(i)})$  é uma palavra-código do  $i$ -ésimo código componente  $C_i$  em  $GF(q)$ . O código multinível é obtido mapeando-se cada coluna da matriz  $(y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(m)})^T$  em uma palavra-código de  $A_1$ . O código multinível pode ser expresso por um vetor  $\bar{S} = (\bar{S}_1, \bar{S}_2, \dots, \bar{S}_{n_2})$ , onde

$$\bar{S}_i = [y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(m)}] \cdot \bar{A} \quad (2.25)$$

com  $\bar{A} = [\bar{a}_1, \bar{a}_2, \dots, \bar{a}_m]^T$  e “ $\cdot$ ” representando a multiplicação de vetores.

A distância de Hamming mínima do código multinível é limitada por :

$$d_{\min} \geq d \triangleq \min \{d_a^{(i)}, d_c^{(i)}, 1 \leq i \leq m\} \quad (2.26)$$

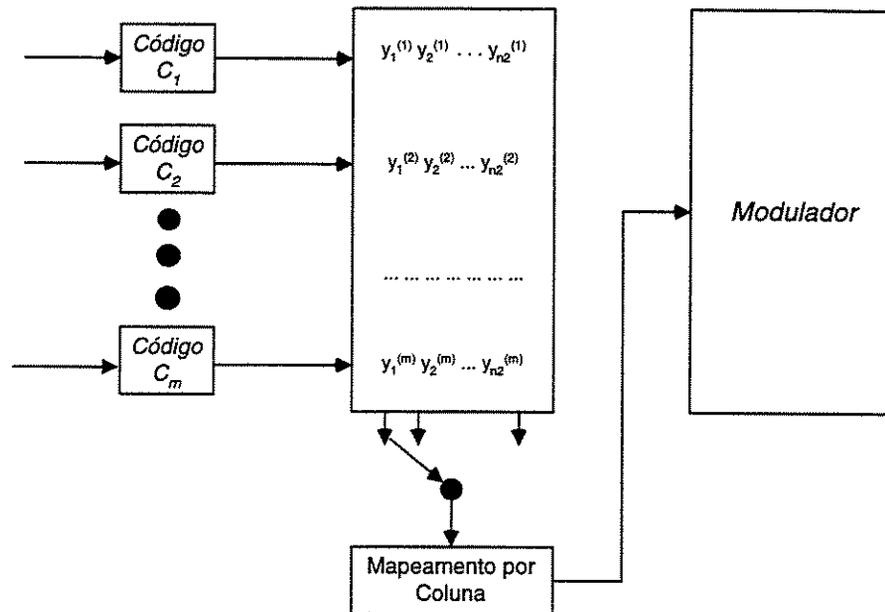


Figura 2.6: Diagrama de bloco do codificador para um código multinível

onde  $d_a^{(i)}$  é a distância mínima de Hamming mínima do  $i$ -ésimo código interno e  $d_c^{(i)}$  é a distância de Hamming mínima do  $i$ -ésimo código componente externo. Um outro parâmetro importante para se analisar o sistema, é sua taxa. Considerando que o número de símbolos de cada código componente  $C_i$  seja dado por  $k_i$ , a taxa do código multinível  $R$  é dada por :

$$R = \frac{\sum_{i=1}^m k_i}{n_2} = \sum_{i=1}^m R_i \quad (2.27)$$

onde  $n_2$  é o comprimento da palavra-código e  $R_i = \frac{k_i}{n_2}$  é a taxa de cada código componente.

Portanto, a construção de um código multinível  $q$ -ário é implementada por  $m$  codificadores componentes seguidos de um mapeador formalizado pela equação (2.25). Os códigos componentes podem ser códigos de bloco, códigos convolucionais, ou uma combinação dos dois, com o esquema de codificação permanecendo essencialmente o mesmo. A Figura 2.6 ilustra o diagrama de blocos do codificador, onde  $C_i$  representa o  $i$ -ésimo código componente.

Para a decodificação de códigos multiníveis pode ser utilizada uma estratégia de decodificação multiestágio. Esta técnica é ilustrada na Figura 2.7, e baseia-se na decodificação

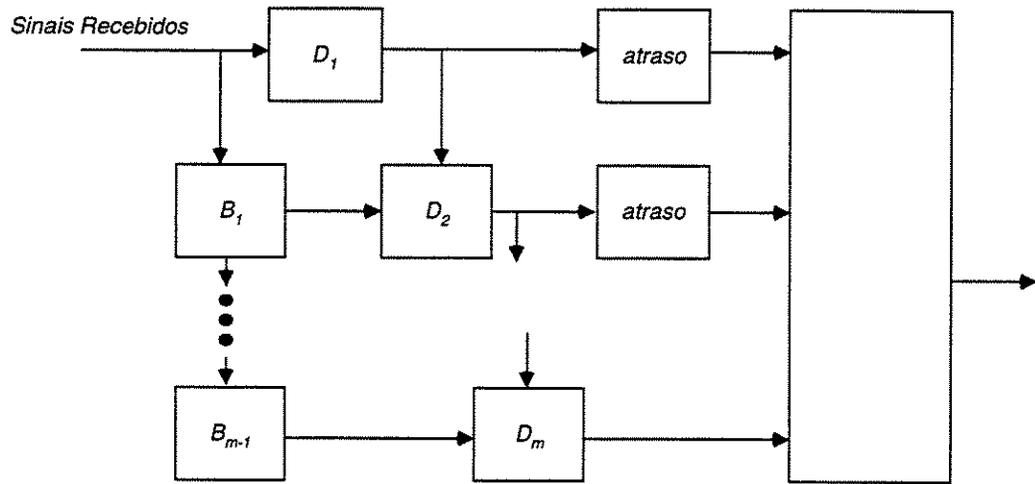


Figura 2.7: Diagrama de bloco do decodificador multiestágio.

dos códigos componentes numa forma seqüencial. De acordo com a figura,  $D_i$  representa o decodificador para o  $i$ -ésimo código componente e  $B_i$  um buffer com atraso igual ao atraso do decodificador  $D_i$ . Assim, o vetor recebido é primeiro decodificado pelo decodificador  $D_1$  produzindo uma estimativa de entrada para o código  $C_1$ . Então o vetor recebido atrasado é decodificado pelo decodificador  $D_2$  auxiliado pela saída de  $D_1$ , produzindo uma estimativa de entrada para o código  $C_2$ . No geral, o codificador  $D_i$  utiliza a informação da saída do decodificador  $D_{i-1}, D_{i-2}, \dots, D_1$  e o vetor recebido para produzir uma estimativa para o código  $C_i$ . O desempenho deste esquema de decodificação é limitado pela distância, uma vez que apenas os padrões de erro com peso de Hamming igual ou menor que a metade da distância mínima podem ser corrigidos.

O procedimento de decodificação é realizado da seguinte forma :

*i)* Considerando  $i = 1$  e uma possível seqüência recebida  $(\bar{S}_1, \bar{S}_2, \dots, \bar{S}_{n_2})$ , procura-se nos conjuntos  $y^{(1)}a_1 + A_2$  a palavra mais próxima em termos de distância de Hamming de cada seqüência recebida  $\bar{S}_j$ . Então guarda-se as distâncias de Hamming mínimas  $(d_H^0, d_H^1, \dots, d_H^{q-1})$  de cada vetor  $\bar{S}_j$  recebido.

*ii)* Para a decodificação do código componente  $C_1$  sobre  $GF(q)$  utiliza-se as métricas obtidas no item anterior. Para isso diversas maneiras de se proceder são válidas. Se o código

$C_1$  for curto, é possível executar a decodificação por tabela. Cada palavra-código do código  $C_1$  possui como métrica a soma das distâncias de Hamming mais próximas aos seus símbolos. Assim obtêm-se a palavra estimada  $(\hat{y}_1^{(1)}, \hat{y}_2^{(1)}, \dots, \hat{y}_{n_2}^{(1)}) \in C_1$ .

iii) Com a decodificação do código  $C_1$  é possível executar a decodificação dos códigos  $C_i, i = 2, 3, \dots, m$ , considerando que a palavra estimada em  $C_1$  está correta, procura-se nos conjuntos  $y^{(i)}a_i + A_{i+1}, i = 2, 3, \dots, n_2$  a palavra mais próxima em termos de distância de Hamming, guardando as distâncias mínimas  $(d_H^0, d_H^1, \dots, d_H^{q-1})$ .

iv) Assim, obtem-se a palavra estimada  $(\hat{y}_1^{(i)}, \hat{y}_2^{(i)}, \dots, \hat{y}_{n_2}^{(i)}) \in C_i, i = 1, 2, \dots, m$ .

Quando os códigos componentes são códigos de bloco ou convolucionais de estrutura simples, a decodificação pode ser realizada por decisão suave “soft decision ” ou decisão brusca “hard decision ”.

Na decodificação multiestágio, o decodificador pode sofrer propagação de erros devido a estimação errada do símbolo nos estágios anteriores. Um dos métodos para amenizar esta propagação seria a de repetir as aplicações da decodificação multiestágio, ou seja, realizar iterações assumindo que as decisões nos primeiros estágios foram somente tentativas. Assim por exemplo, numa segunda iteração, o código  $C_1$  seria decodificado assumindo que as palavras estimadas pelos códigos  $C_i, i = 2, 3, \dots, m$  estivessem corretas.

UNICAMP  
BIBLIOTECA CENTRAL  
SEÇÃO CIRCULANTE

## Capítulo 3

# Um Sistema FH-CDMA

Em sistemas com saltos em frequência (FH), dois tipos de esquemas de modulação são normalmente utilizados : o OOK (do inglês, on-off keying) e o FSK (do inglês, frequency-shift keying). Nestes dois esquemas de modulação é importante se analisar e comparar a capacidade de informação, ou seja, dada uma largura de faixa fixa, qual esquema pode transmitir a informação mais eficientemente num sistema de saltos em frequência (FH). Estudos realizados em [Goh98] mostraram que a capacidade de informação de um sistema utilizando a modulação OOK é maior do que um sistema utilizando a modulação FSK. Em um canal sem ruído, onde somente a interferência provocada pelos usuários causa a degradação do sistema, a capacidade de informação utilizando a modulação on-off é quase o dobro da capacidade de um sistema com modulação FSK.

Quando o sistema de saltos em frequência com modulação on-off opera em um canal de acesso múltiplo, a grande causa de degradação do sistema é a interferência provocada pela transmissão simultânea dos usuários no canal. A interferência é provocada porque, em dado momento, os usuários transmitindo suas mensagens podem se combinar e causar uma ambigüidade da decisão na recepção. Esta interferência pode ser minimizada, como já foi citado no capítulo anterior, pela escolha de modelos de endereçamento que garantem a correlação mínima entre os sinais.

Em princípio, o sistema pode ser melhorado pela decodificação simultânea dos endereços de todos os possíveis usuários. Este tipo de decodificação, chamada de decodificação

completa, se torna muito complexa em sistemas onde o número de frequências disponíveis é grande.

Pensando nisso, Timor [Tim80] propôs um algoritmo de cancelamento de interferência que ao explorar a estrutura algébrica de seqüências de endereçamento proposta em [Ein80], reduz substancialmente a interferência com uma complexidade bem inferior a da decodificação completa. O algoritmo testa possíveis linhas completas na matriz de recepção após a retirada do endereço do usuário. A linha que corresponde à mensagem do usuário na maioria das vezes é identificada após um teste de interferência e a mensagem é decodificada. Para um grande número de frequências esta decodificação possui uma complexidade bem inferior a da decodificação completa. O algoritmo foi analisado para um canal onde somente a interferência provocada pelos usuários causou a degradação do sistema. Porém, não se conhecia até agora o comportamento deste algoritmo quando aplicado em canais com ruído aditivo e desvanecimento seletivo em frequência.

Fiebig [Fie96] também propôs um algoritmo de cancelamento iterativo de interferência para ser usado tanto em canais com desvanecimento seletivo em frequência como canais onde é considerado somente a interferência entre os usuários.

A primeira parte do capítulo descreve um sistema de saltos em frequência utilizando modulação on-off. Modelos de endereçamento para os casos de transmissão síncrona e assíncrona são também apresentados. Os algoritmos de cancelamento de interferência de Timor e Fiebig são descritos e a análise de seus desempenhos em canais com ruído aditivo e desvanecimento seletivo em frequência é apresentada.

### 3.1 Sistema FH-CDMA com Modulação on-off

O sistema utilizando saltos em frequência e modulação OOK pode ser entendido a partir de uma “matriz A” contendo  $q$  linhas (correspondendo ao número de frequências disponíveis) e  $L$  colunas (correspondendo ao número de intervalos possíveis para transmissão), onde cada elemento da matriz é denominado chip. Para um sistema com uma largura de faixa total  $W$ ,

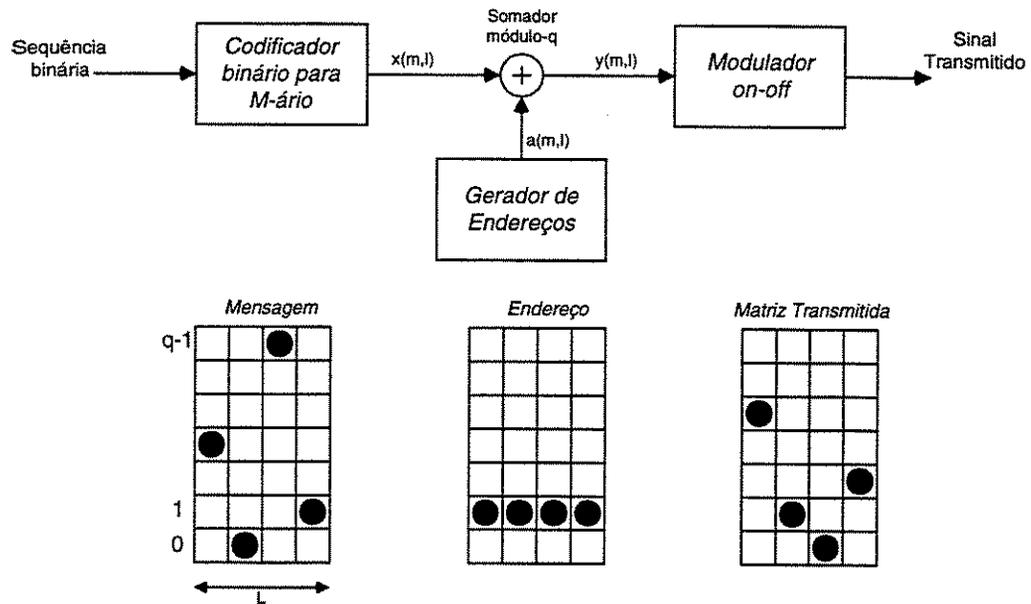


Figura 3.1: Transmissor de um sistema FH-CDMA. O diagrama de blocos indica as operações de processamento de sinal. As matrizes mostram as sequências de níveis lógicos ou frequências selecionadas para a transmissão.

cada linha da matriz ocupa uma largura de faixa de  $\frac{W}{q}$ . A duração de tempo de uma coluna é de  $\frac{T}{L}$ .

Durante um intervalo de mensagem  $T$ , o transmissor envia uma seqüência de  $L$  frequências,  $f_1, f_2, \dots, f_L$ . Estas frequências são determinadas pela soma módulo- $q$  do vetor mensagem do usuário com o seu endereço particular. Transportando o vetor mensagem e o vetor endereço para um corpo algébrico  $GF(q)$ , a operação de soma destes vetores, que será abordada mais detalhadamente na próxima seção é a operação componente a componente no corpo  $GF(q)$ . As frequências resultantes desta soma podem ser representadas como entradas na matriz  $q \times L$ , e a matriz é preenchida se uma dada frequência está presente ou não. A seqüência de operações do transmissor encontra-se ilustrada na Figura 3.1 e as matrizes representam somente um enlace de comunicação num sistema multiusuário.

O receptor, a cada  $\tau = \frac{T}{L}$  segundos, realiza a demodulação do espectro de frequências recebido e decide quais das  $q$  possíveis frequências estão presentes, ou seja, quantos chips estão ativos em cada coluna da matriz recebida. Assim, a cada  $T$  segundos é gerada uma

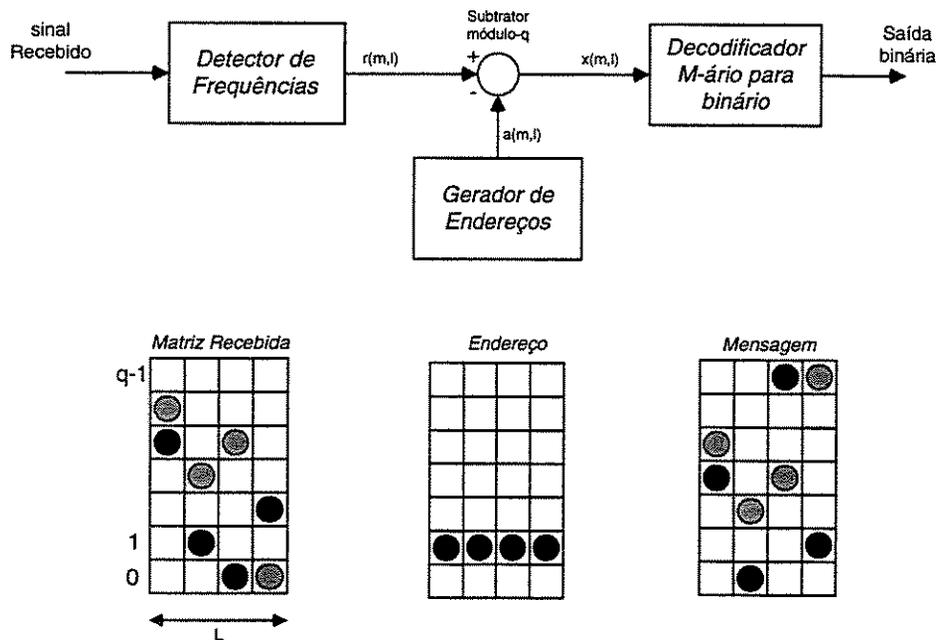


Figura 3.2: Diagrama de blocos do receptor e as matrizes de sinais de um sistema FH-CDMA

matriz com o espectro de frequências transmitido. Para se extrair a informação é realizada a operação de subtração da matriz recebida com o endereço do usuário desejado no corpo  $GF(q)$ , restando na matriz a mensagem do usuário particular e alguns chips provenientes da interferência. A Figura 3.2 descreve a operação de recepção do sistema.

Num canal de múltiplos usuários, onde vários usuários transmitam simultaneamente suas mensagens, podem se encontrar presentes na matriz de recepção de um usuário particular, vários chips que não estavam presentes na matriz que foi originalmente transmitida, como ilustrados na Figura 3.2 os símbolos mais claros. Se os modelos de salto de cada usuário neste sistema forem modelados como seqüências independentes, equiprováveis em relação as  $q$  frequências, assim a probabilidade de um choque,  $p$ , que é a probabilidade de que o pulso transmitido pelos outros usuários seja recebido pelo  $m$ -ésimo transmissor, é dada por :

$$p = 1 - \left(1 - \frac{1}{q}\right)^{K-1} \quad (3.1)$$

onde  $K$  representa o número de usuários ativos. Desta forma, cada receptor está somente

interessado na informação transmitida pelo seu correspondente transmissor e o sistema pode ser entendido como vários canais  $M$ -ários simples, onde cada um está submetido a interferência de acesso múltiplo.

Quando o canal a ser utilizado, está sob o efeito do desvanecimento, caracterizado pela existência de uma componente multiplicativa na amplitude do sinal, com distribuição de probabilidade Rayleigh, o apagamento de chips pode ocorrer porque esta componente pode levar o pulso transmitido a um valor abaixo do limiar de decisão normalizado a cada instante de amostragem. Pode ocorrer também inserções de chips, ou seja, o ruído do canal excede o limiar de decisão e causa detecção de um chip falso. As probabilidades de apagamento  $p_D$  e inserção  $p_F$  são dadas respectivamente de acordo com as expressões abaixo [Ein84] :

$$p_D = 1 - \exp\left(-\frac{\lambda_0^2}{2(1+\gamma)}\right) \quad (3.2)$$

$$p_F = \exp\left(-\frac{\lambda_0^2}{2}\right) \quad (3.3)$$

onde  $\lambda_0$  é o limiar de decisão normalizado e  $\gamma$  é a relação sinal-ruído no canal.

## 3.2 Modelos de Endereçamento

É conveniente descrever os procedimentos de modulação e decodificação em termos algébricos, ou seja, denotando as  $q$  frequências como elementos em  $GF(q)$ . Assim a mensagem  $x_m$  composta de  $\log_2 q$  bits, o endereço  $a_m$  e os números de linha da “matriz  $A$ ” podem ser expressos como elementos em  $GF(q)$  e as operações de adição e multiplicação são realizadas de acordo com as regras de  $GF(q)$ .

Einarsson [Ein80], estudou um modelo de endereços em que cada usuário ativo no sistema é associado a um elemento em  $GF(q)$ . No modelo assume-se que o sistema está em sincronismo,

ou seja, que todos os usuários estão alinhados no tempo. Assim, duas seqüências transmitidas só causarão interferência, se e somente se, elas ocuparem os mesmos chips na matriz de sinal. Um vetor endereço,  $\bar{a}_m$ , consistindo de  $L$  chips é associado a cada usuário  $m$ , com  $\gamma_m$  um elemento de  $GF(q)$  e  $\beta$  o elemento primitivo do corpo, definido da seguinte forma :

$$\bar{a}_m = (\gamma_m, \gamma_m\beta, \gamma_m\beta^2, \dots, \gamma_m\beta^{L-1}) \quad (3.4)$$

Este vetor acima garante  $q$  possíveis endereços. A seqüência de saltos transmitida consiste da soma elemento a elemento no corpo  $GF(q)$  do vetor mensagem do usuário com o seu vetor endereço sendo expressa como :

$$\bar{y}_m = \bar{a}_m + x_m \cdot \bar{1} \quad (3.5)$$

onde  $x_m$  e cada componente de  $\bar{a}_m \in GF(q)$ . A notação  $\bar{1}$  é utilizada para o vetor todo um de comprimento  $L$  que multiplica a mensagem do usuário  $x_m$ . Com a escolha apropriada de  $\beta$  é possível afirmar que quaisquer duas seqüências  $\bar{y}_m$  e  $\bar{y}'_m$  irão coincidir em no máximo um chip, independente da mensagem transmitida pelos usuários(veja o apêndice A).

Na prática, alcançar a sincronização entre os usuários do sistema é muito difícil e por isso a equação acima não é muito adequada para esta situação. A propriedade cíclica mostrada na equação (3.4) pode causar um agravamento na interferência, no caso especial em que  $q = L - 1$ . Desta forma, foi proposto um novo caminho para se obter um sistema resistente a interferência assíncrona. Agora a mensagem do usuário  $x_m$  é multiplicada pelo vetor  $\bar{\beta} = (1, \beta, \beta^2, \dots, \beta^{L-1})$  e somada com o vetor  $\gamma_m \cdot \bar{1}$ , que representa o endereço do usuário  $m$ . O modelo possui a seguinte forma :

$$\bar{y}_m = x_m \cdot \bar{\beta} + \gamma_m \cdot \bar{1} \quad (3.6)$$

onde  $\bar{1}$  é o vetor todo um de dimensão  $L$  e  $x_m, \beta, \gamma_m \in GF(q)$ . Para este caso é possível

mostrar que quaisquer duas seqüências irão coincidir em no máximo um chip, como no caso síncrono(veja o Apêndice A).

A diferença fundamental entre as equações (3.5) e (3.6) é que no primeiro caso, os  $q$  possíveis endereços  $\bar{a}_m$  constituem um código cíclico encurtado e no último caso, as  $q$  possíveis mensagens de um usuário particular (multiplicadas por  $\bar{\beta}$ ) formam o código cíclico.

### 3.3 Algoritmos de Cancelamento de Interferência

#### 3.3.1 Algoritmo de Cancelamento Proposto por Timor

O algoritmo de decodificação de Timor [Tim80] é baseado no teste de verificação de possíveis linhas completas realizada após a retirada do endereço da matriz recebida. O termo “linha completa” se refere a uma linha da forma  $x_m \cdot \bar{1}$  onde todos os elementos da linha estão ativos. Timor considerou que todos os usuários no canal estão em sincronismo, portanto utilizou a equação (3.5) para a transmissão das seqüências. O algoritmo pode também ser aplicado em sistemas onde os usuários não estão em sincronismo, neste caso cada usuário transmite sua seqüência de acordo com a equação (3.6). No caso assíncrono, após a retirada do endereço da matriz recebida, a linha completa não é revelada diretamente. É necessário realizar uma pré-multiplicação componente a componente do vetor resultante  $x_m \cdot \bar{\beta}$  por  $\bar{\beta}^{-1} = (\beta^0, \beta^{-1}, \beta^{-2}, \dots, \beta^{-(L-1)})$  para obter o vetor  $x_m \cdot \bar{1}$  que mostra a linha completa correspondendo a mensagem transmitida pelo usuário. A análise do algoritmo de Timor neste item considera que todos os usuários no canal estão em sincronismo, mas suas seqüências são transmitidas de acordo com a equação (3.6) sendo necessário fazer a pré-multiplicação pelo vetor  $\bar{\beta}^{-1}$  para que o teste de verificação das linhas completas seja realizado.

#### Algoritmo de Decodificação

No esquema de decodificação de Timor, considera-se  $M$  como o número de usuários no sistema, onde cada usuário transmite uma seqüência dada segundo a equação :

$$\bar{y}_m = x_m \cdot \bar{\beta} + \gamma_m \cdot \bar{1} \quad (3.7)$$

Assim, para o usuário  $m$ , após a retirada de seu endereço e a pré-multiplicação, aparece na “matriz  $A_m$ ” deste usuário a mensagem como uma linha completa  $x_m \cdot \bar{1}$ . A combinação dos outros usuários do sistema pode ocasionar a presença de uma outra linha completa  $x' \cdot \bar{1}$  em  $A_m$ . Cada chip desta linha, de acordo com o que foi mostrado na seção anterior (ver também o Apêndice A), é proveniente de no mínimo  $L$  usuários distintos. Assim, o usuário que contribui com a interferência na coluna  $n$  de  $x' \cdot \bar{1}$  é denotado por  $i_n$ . O  $j$ -ésimo ( $j = 1, 2, \dots, L$ ) chip destes usuários na matriz  $A_m$  será a linha  $C_n$  e coluna  $j$  :

$$\begin{aligned} C_n(j) &= (x_{i_n} \cdot \beta^{j-1} + \gamma_{i_n} - \gamma_m) \cdot \beta^{1-j} \\ &= \delta_n \beta^{1-j} + x_{i_n}, \quad j = 1, 2, \dots, L \end{aligned} \quad (3.8)$$

onde  $\gamma_m$  é o endereço do  $m$ -ésimo usuário e :

$$\delta_n \triangleq \gamma_{i_n} - \gamma_m, \quad (3.9)$$

$\delta_n \neq 0$  para todos os interferentes  $i_n$ ,  $n = 1, 2, \dots, L$ .

A linha  $C_n$  é a linha na qual o chip do usuário  $i_n$  aparece na  $n$ -ésima coluna. Por definição esta linha é a própria linha  $x' \cdot \bar{1}$ . Assim, tem-se a seguinte relação :

$$C_n(n) = \delta_n \beta^{1-n} + x_{i_n} = x' \quad (3.10)$$

Considera-se  $D_x^*$  uma matriz obtida da subtração do número correspondente a linha  $x'$  em  $GF(q)$  de todas as linhas da matriz  $A_m$ , isto é equivalente a trocar a linha  $x' \cdot \bar{1}$  pela linha zero. Em  $D_x^*$  o chip do usuário  $i_n$  na coluna  $j$  ( $j = 1, 2, \dots, L$ ), será agora a linha  $d_n(j)$ . Das equações (3.8) e (3.10) tem-se que :

$$\begin{aligned} d_n(j) &= C_n(j) - x' \\ &= \delta_n (\beta^{1-j} - \beta^{1-n}), \quad n, j = 1, \dots, L \end{aligned} \quad (3.11)$$

Trocando a linha  $x' \cdot \bar{1}$  pela linha zero, os chips dos interferentes  $i_1, i_2, \dots, i_L$  se tornam independentes de suas mensagens  $x_{i_1}, x_{i_2}, \dots, x_{i_L}$ . Mas se a linha  $x' \cdot \bar{1}$  em  $A_m$  for formada por  $L$  interferentes com endereços  $\delta_n + \gamma_m$ , seus chips apareceram em  $D_x^*$  como linhas  $d_n$  e coluna  $j$  ( $j, n = 1, 2, \dots, L$ ), portanto tem-se o seguinte teorema :

**Teorema** *A condição necessária para que linha  $x'$  em  $A_m$  seja causada por interferência é a existência de  $L$  números  $\delta_n$  diferentes de zero em  $GF(q)$  tal que todos chips  $(d_n(j), j(n, j = 1, 2, \dots, L))$  apareçam em  $D_x^*$ .*

Da equação (3.11) pode-se escrever para  $n = 1, \dots, L - 1$ .

$$\begin{aligned} d_n(j) &= \delta_n \beta^{1-n} (\beta^{n-j} - 1) \\ d_n(n+1) &= \delta_n \beta^{1-n} (\beta - 1) \end{aligned} \quad (3.12)$$

derivando-se a seguinte relação :

$$d_n(n+1) = f_{n-j} d_n(j) \quad (3.13)$$

onde :

$$f_{n-j} \triangleq \frac{\beta - 1}{\beta^{n-j} - 1}, \quad j = 1, \dots, L, \quad j \neq n \text{ e } n = 1, \dots, L - 1 \quad (3.14)$$

$$f_{L-j}^* = \frac{1}{\beta} f_{L-j}, \quad j = 1, \dots, L \quad \text{e} \quad n = L \quad (3.15)$$

onde  $f_{n-j}$  e  $f_{L-j}^*$  são fixos pelo sistema e independentes das seqüências transmitidas pelos usuários.

A condição estabelecida pela equação (3.11) é verificada multiplicando-se todos os elementos (os números de linha) na coluna  $j$  ( $j = 1, \dots, L, j \neq n, j \neq n + 1$ ) da matriz  $D_x^*$  por  $f_{n-j}$  ( $n = 1, \dots, L - 1$ ) ou por  $f_{L-j}^*$  ( $n = L$ ) e procurando por um linha “completa” (completa exceto a  $n$ -ésima entrada da linha). Se a linha for encontrada, então um possível interferente está contribuindo para a interferência na coluna  $n$  de  $x' \cdot \bar{1}$ . Se esta condição for satisfeita para todos os valores de  $n$  ( $n = 1, \dots, L$ ), assume-se que a linha  $x'$  em  $A_m$  é resultado de interferência e a condição é verificada para as outras linhas completas da matriz. A linha correta  $x_m$  normalmente não atende a condição acima e assim pode ser identificada e decodificada como a mensagem correta.

As multiplicações realizadas na matriz  $D_x^*$  podem ser substituídas por deslocamentos cíclicos de colunas. Estes deslocamentos são implementados expressando cada número de linha da matriz por um expoente do elemento primitivo  $\beta \in GF(q)$ . Assim as operações de multiplicação realizadas na matriz  $D_x^*$  são agora substituídas por permutações (fixas) de linhas, tornando-se mais fácil a implementação. Com este tipo de operação o decodificador requer um aumento de complexidade da ordem de  $L^2$  deslocamentos cíclicos para cada linha completa nas matrizes subtraídas do endereço do usuário.

Para um entendimento deste esquema de decodificação, é mostrado na figura seguinte um diagrama de blocos de uma possível implementação do algoritmo, onde para cada linha completa na matriz o algoritmo procede da seguinte forma :

1. Para cada matriz recebida  $A$ , calcula-se a matriz decodificada  $A_m$ , retirando o endereço do usuário  $a_m$  e realizando a pré-multiplicação pelo vetor  $\beta^{-1}$ . Se em  $A_m$  houver apenas um linha completa, esta linha é decodificada como a linha correta e o procedimento de decodificação é concluído. Se houver  $I$  linhas completas ( $I \geq 2$ ),  $x^{(1)}, \dots, x^{(I)}$ , o algoritmo para cada linha  $x^{(i)}$  se procede da seguinte maneira :
2. Subtrai (o número da linha)  $x^{(i)}$  de todas as linhas de  $A_m$  obtendo-se a matriz  $D_x$ .
3. Para cada  $n$  ( $n = 1, \dots, L$ ) multiplica-se todas as  $j$  ( $j = 1, \dots, L$ ) colunas de  $D_x$  por  $f_{n-j}$  e  $f_{L-j}^*$  e verifica-se um possível linha completa (exceto a  $n$ -ésima coluna). Se para algum

$n$ , não for encontrado tal linha “completa”, decodifica-se  $x_m = x^{(i)}$ , e o procedimento de decodificação é concluído. Por outro lado, se a linha for encontrada, esta linha é assumida ser um resultado de interferência e os passos 2 e 3 são repetidos para  $x^{(i+1)}$ .

4. Se todas as linhas passarem pelo teste de interferência, a escolha é feita aleatoriamente entre as  $x^{(i)}$  ( $i = 1, \dots, I$ ) linhas da matriz  $A_m$ .

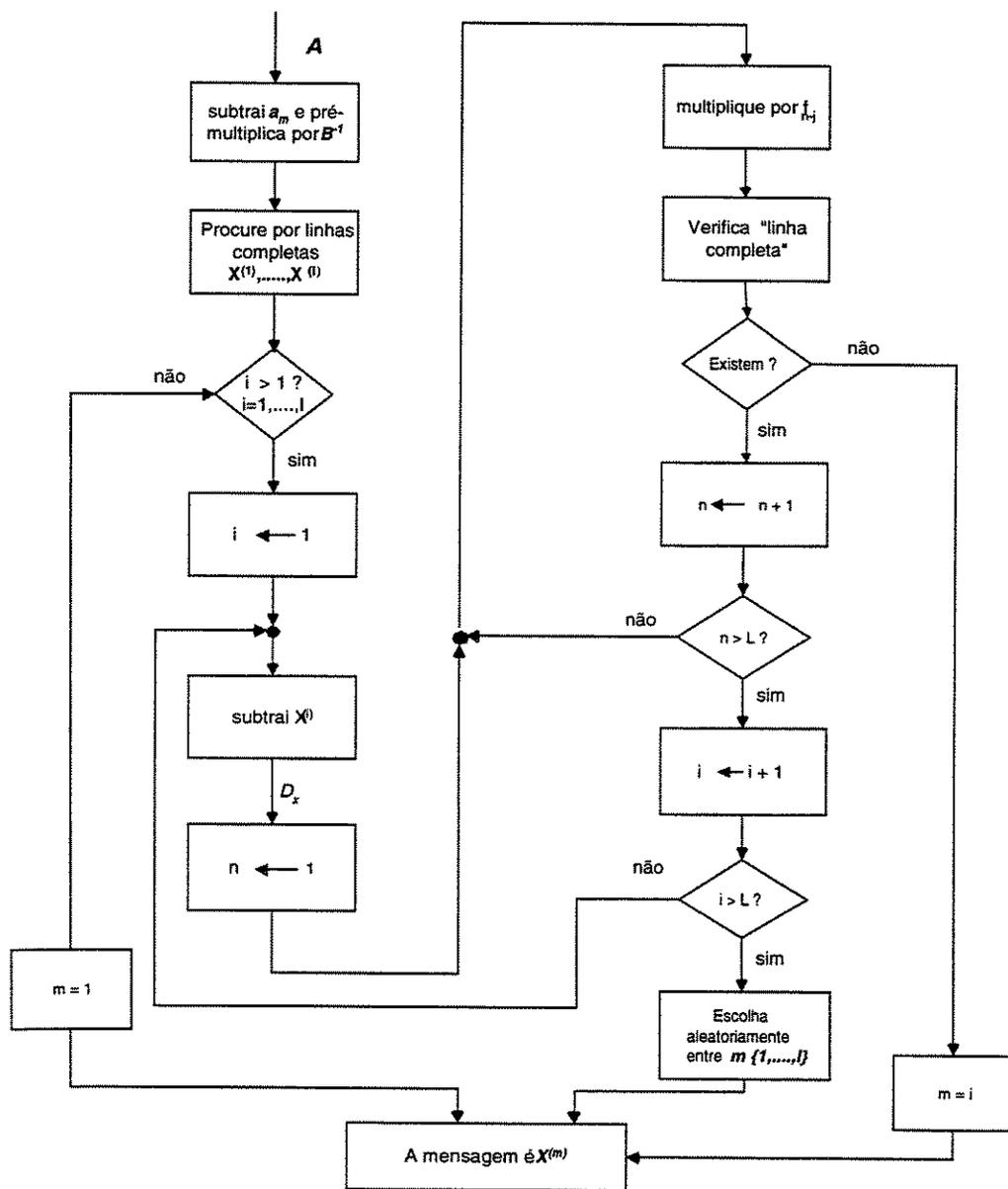


Diagrama de blocos de um possível implementação do decodificador

### Desempenho do Sistema

O esquema de Timor num canal com somente interferência entre os usuários é baseado na verificação de possíveis linhas completas. Assim, se na matriz de um usuário  $m$  houver apenas uma linha completa, o número desta linha que corresponde a um elemento em  $GF(q)$  é decodificada como a mensagem  $x_m$ . Numa situação em que houver duas ou mais linhas completas na matriz é necessário realizar o teste de interferência. Neste caso ocorre uma decodificação incorreta se uma linha que não corresponde a mensagem for incorretamente identificada como a mensagem correta ao passar pelo teste de interferência. A probabilidade do decodificador decidir por uma das  $(q - 1)$  possíveis mensagens é dada por :

$$P_1 < (q - 1) p^L \quad (3.16)$$

onde  $p$ , é a probabilidade de um chip na matriz ser ativado por um ou vários sinais interferentes em um sistema FH-CDMA, dada segundo a equação (3.1).

Uma outra situação em que pode ocorrer uma ambigüidade no sistema é o caso da linha correta  $x_m$  satisfazer a condição estabelecida pela equação (3.11). Nesta situação, o decodificador não identifica a linha correta e a probabilidade de erro pode ser expressa por :

$$P_2 < p^L (1 + S)^L \quad (3.17)$$

onde

$$S = (q - 1) (1 - p) p^{(L-2)} \quad (3.18)$$

Assumindo que os eventos de erro são independentes, a probabilidade de erro por palavra  $P_e$  é dada por :

$$P_e < \frac{1}{2} P_1 \cdot P_2 < \frac{1}{2} (q - 1) (1 + S)^L p^{2L} . \quad (3.19)$$

O fator  $\frac{1}{2}$  vem da escolha aleatória das mensagens quando ocorre uma falha na decodificação por alguma das duas situações. A probabilidade de erro de bit, para o sistema pode ser limitada por :

$$P_b < \frac{q}{4} (1 + S)^L p^{2L} \quad (3.20)$$

O sistema foi analisado em [Tim80] e os resultados mostraram que o algoritmo obteve uma melhora em torno de 60% em relação ao esquema de decodificação convencional. Porém, o sistema não foi analisado para um canal com ruído aditivo e desvanecimento seletivo em frequência. Neste canal o efeito combinado do ruído e interferência entre os usuários pode causar apagamento e inserção de chips nas matrizes. Assim, algumas modificações nas expressões de probabilidade de erro acima foram feitas para analisar o sistema em um canal com desvanecimento seletivo em frequência.

Num canal com desvanecimento seletivo em frequência, a detecção dos chips na matriz recebida é influenciada pelas probabilidades de apagamento e inserção expressas pelas equações (3.2) e (3.3). A consequência disto é a possibilidade de não haver linhas completas nas matrizes após a detecção de todos os chips e neste caso, o teste de interferência de possíveis linhas completas se torna impossível. Para evitar esta possibilidade é empregada a decisão por lógica majoritária, ou seja, escolhe-se a mensagem recebida associada a linha com o maior número de entradas (chips).

A probabilidade total de inserção de um chip na matriz devido ao efeito combinado de interferência e o ruído do canal pode ser expressa como :

$$p_I = p - p_1 \cdot p_D + (1 - p) p_F \quad (3.21)$$

onde o termo  $p$  é dado segundo a equação (3.1) e  $p_1$  é a probabilidade de que exatamente um chip seja ativado pelos sinais interferentes sendo dada por :

$$p_1 = (M - 1) \frac{1}{q} \left(1 - \frac{1}{q}\right)^{M-2} \quad (3.22)$$

Assim as probabilidades  $P_1$  e  $P_2$  são agora limitadas superiormente por :

$$P_1 < (q - 1) p_I^L \quad (3.23)$$

$$P_2 < p_I^L (1 + S)^L \quad (3.24)$$

com

$$S = (q - 1) (1 - p_I) p_I^{L-2} \quad (3.25)$$

A probabilidade de erro de bit neste canal é :

$$P_b < \frac{q}{4} (1 + S)^L p_I^{2L} \quad (3.26)$$

As Figuras 3.3, 3.4 e 3.5 apresentam o desempenho do algoritmo de Timor para o canal com desvanecimento seletivo em frequência. Os resultados analíticos são obtidos para valores de relação sinal-ruído de 16, 25 e 40 dB.

### 3.3.2 Algoritmo de Cancelamento Proposto por Fiebig

O princípio do algoritmo de cancelamento iterativo de Fiebig é partir com a detecção convencional, onde os símbolos decididos em uma primeira etapa são retirados da matriz de recepção e o processo de detecção convencional é re-invocado baseado na matriz de recepção modificada. Desta forma, iterações são realizadas até que todos os símbolos sejam estimados. Da mesma forma que o algoritmo anterior, considera-se que os usuários do sistema estão em sincronismo e suas seqüências de saltos são transmitidas de acordo com a equação (3.6).

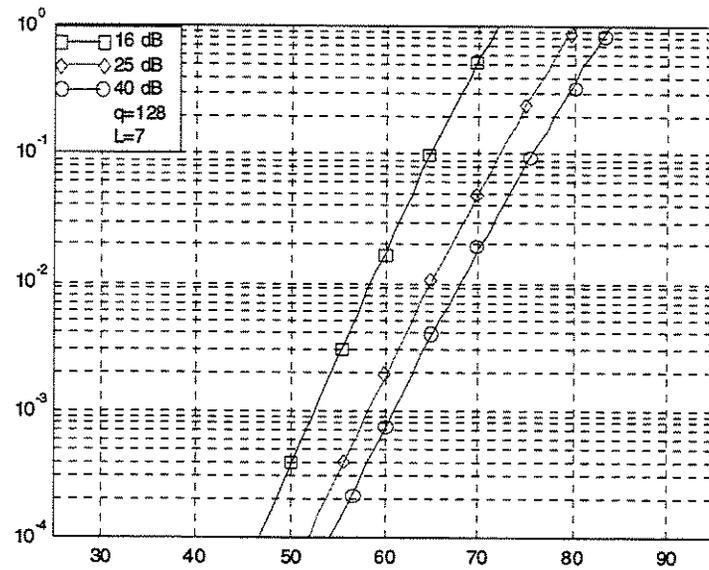


Figura 3.3: Desempenho do algoritmo de Timor para canais com ruído e desvanecimento seletivo em frequência,  $q=128$  e  $L=7$ .

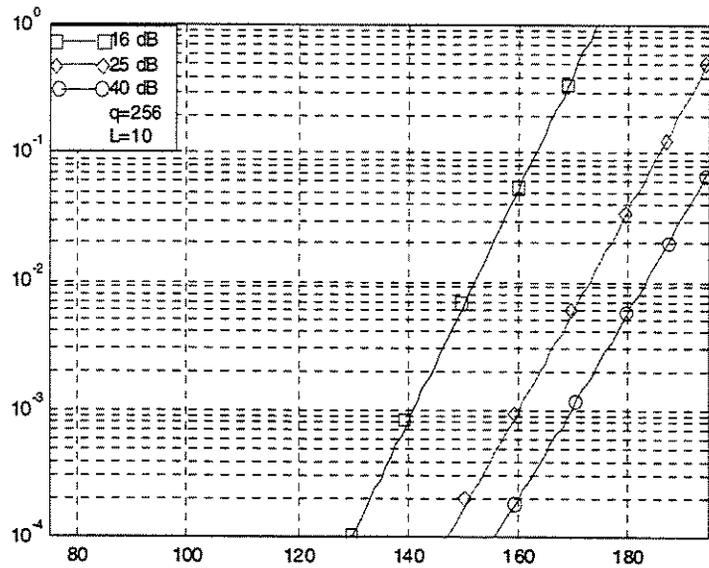


Figura 3.4: Desempenho do algoritmo de Timor para canais com ruído e desvanecimento seletivo em frequência,  $q=256$  e  $L=10$ .

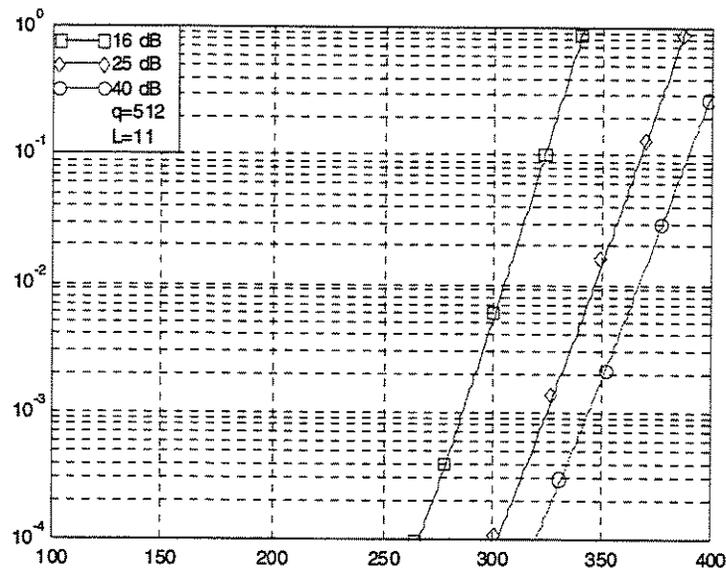


Figura 3.5: Desempenho do algoritmo de Timor para canais com ruído e desvanecimento seletivo em frequência,  $q=512$  e  $L=11$ .

### Algoritmo de Cancelamento

Considerando que  $M$  usuários estão transmitindo suas mensagens simultaneamente no canal, o algoritmo de Fiebig procede como segue :

1. De uma matriz recebida  $R$ , retira-se os endereços dos usuários gerando matrizes  $D_m, m = 1, 2, \dots, M$ . Após a geração destas matrizes realiza-se uma pré-multiplicação pelo vetor  $\bar{\beta}^{-1}$  para a identificação de possíveis linhas completas nas matrizes  $D_m$ .
2. Para cada matriz  $D_m$  que não contenha ambigüidade, ou seja, duas ou mais linhas completas, a decisão é feita pela linha com número máximo de entradas. No caso em que todas as matrizes contêm ambigüidade a decisão é aleatória entre as linhas completas de cada matriz e o processo de detecção é finalizado.
3. Após a decisão pelos símbolos, o processo de iteração é iniciado para  $s = 1, 2, \dots, s_{\max}$  realizando os seguintes passos :
  - (a) As seqüências correspondentes aos símbolos já estimados nas iterações prévias, são retiradas da matriz recebida  $R$  produzindo a matriz  $C^{(s)}$ .

(b) Para os usuários cujos símbolos ainda não tenham sido estimados, baseado na matriz  $C^{(s)}$ , são geradas as matrizes  $D_m^{(s)}$ . Após a geração destas matrizes, checa-se as linhas que foram identificadas como ambigüidade nas matrizes  $D_m$  e entre estas linhas, a que corresponde ao maior número de entradas é considerada como o símbolo correto. No caso em que houver duas ou mais linhas com o mesmo número de entradas, nenhuma decisão é realizada.

4. As iterações são realizadas até que todas as ambigüidades sejam resolvidas e os símbolos estimados. Se na última iteração ainda ocorrer ambigüidade, a decisão é aleatória entre as linhas com o maior número de entradas.

As modificações realizadas na matriz recebida  $R$  são motivadas pelo fato de que os elementos de uma linha incorreta, provenientes de chips interferentes são mais prováveis de serem retirados do que elementos de uma linha correta.

Para ilustrar melhor o procedimento de decodificação do sistema, é mostrado um exemplo de um sistema com  $q = 7$  e  $L = 3$  e 6 usuários simultâneos no canal. A parte superior da figura 3.6 mostra a mensagem dos usuários, o endereço associado a cada um, e as seqüências transmitidas, respectivamente. Em seguida é mostrada a matriz recebida  $R$  e as matrizes  $D_m, m = 1, \dots, 6$ . Pode ser visto que a matriz  $D_2$  possui uma ambigüidade de três linhas e as matrizes  $D_1, D_3, D_5, D_6$  possuem ambigüidades de duas linhas. Somente a matriz  $D_4$  não possui ambigüidade, e assim somente esta matriz, que corresponde ao símbolo do usuário 4 pode ser decodificada corretamente. Com a decodificação do usuário 4, o algoritmo realiza a primeira iteração : de acordo com o passo 1, os elementos 1, 2 e 5 da primeira, segunda e terceira colunas da matriz  $R$  são retirados produzindo a matriz  $C^{(1)}$ . Baseado nesta matriz o processo de detecção convencional é re-invocado para todos os usuários, exceto o usuário 4, cujo símbolo já foi estimado. Obtendo-se as matrizes  $D_m^{(1)}, m = 1, 2, 3, 5$  e 6. Nesta segunda iteração é possível a decodificação das matrizes  $D_1^{(1)}, D_5^{(1)}$  e  $D_6^{(1)}$ , que ainda possuíam ambigüidade de mais de uma linha. Verifica-se então que somente que as matrizes  $D_2^{(1)}$  e  $D_3^{(1)}$  ainda contém ambigüidade sendo necessário realizar mais uma iteração

usuário	→	1	2	3	4	5	6
$x_m$	→	(1,3,2)	(3,2,6)	(2,6,4)	(4,5,1)	(1,3,2)	(5,1,3)
$a_m$	→	(1,1,1)	(2,2,2)	(3,3,3)	(4,4,4)	(5,5,5)	(6,6,6)
$y_m$	→	(2,4,3)	(5,4,1)	(5,2,0)	(1,2,5)	(6,1,0)	(4,0,2)

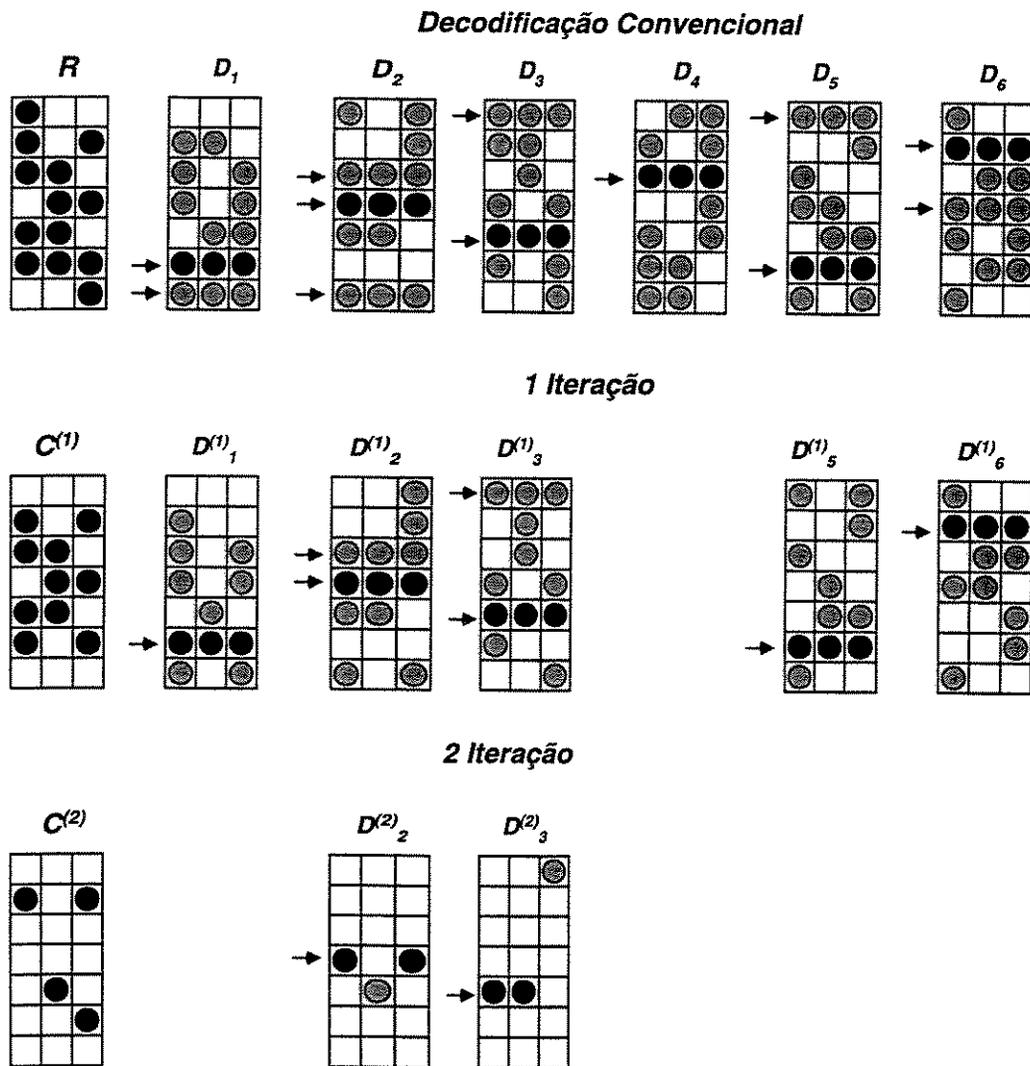


Figura 3.6: Um exemplo do algoritmo de decodificação de Fiebig com duas Iterações e 6 usuários simultâneos no canal,

do algoritmo.

#### Desempenho do Sistema

Para canais onde somente a interferência dos usuários é considerada, a principal causa de degradação do sistema são as ambigüidades de linhas nas matrizes  $D_m, m = 1, \dots, M$ . Suponhamos que uma matriz de um usuário particular denotada por  $D_1$  possui ambigüidade de  $t + 1$  linhas e que exatamente  $M_s$  outras matrizes não possuem ambigüidade. Após a retirada das seqüências da matriz  $R$  dos usuários cujos símbolos já foram estimados na primeira iteração é gerada a matriz  $D_1^{(1)}$  deste usuário particular. Existem situações em  $D_1^{(1)}$  que podem ocasionar erros na decodificação da mensagem. Uma típica situação de erro é quando a linha correta possui menos entradas do que as linhas interferentes, neste caso o erro ocorrerá com probabilidade dada por :

$$P_1 = \left[ \sum_{l=1}^L \binom{L}{l} (P_c)^l (1 - P_c)^{L-l} \right] \cdot \left[ 1 - \left( \sum_{\lambda=1}^L \binom{L}{\lambda} (P_{on})^\lambda (1 - P_{on})^{L-\lambda} \right)^t \right] \quad (3.27)$$

O primeiro termo entre colchetes se refere à linha correta e é a probabilidade de que  $l = 1, \dots, L$  entradas na linha sejam apagadas depois da primeira iteração,  $P_c$  é a probabilidade de que uma entrada da linha correta pertença às outras  $M_s$  matrizes :

$$P_c = 1 - \left( 1 - \frac{1}{q} \right)^{M_s} \quad (3.28)$$

O segundo termo refere-se às  $t$  linhas incorretas da matriz  $D_1^{(1)}$  e é a probabilidade de que uma destas  $t$  linhas possuam um número de entradas maior do que a linha correta.  $P_{on}$  é a probabilidade de que um elemento de uma linha incorreta pertença às  $M_s$  matrizes :

$$P_{on} = \sum_{I=1}^{M-1} \frac{\binom{M-1}{I} \left( \frac{1}{q} \right)^I \left( 1 - \frac{1}{q} \right)^{M-1-I}}{1 - \left( 1 - \frac{1}{q} \right)^{M-1}} \quad (3.29)$$

onde  $I, 1 \leq I \leq M - 1$ , se refere aos interferentes.

Uma outra situação em que ocorre ambigüidade na matriz  $D_1^{(1)}$  é quando a linha correta e  $\mu$  ( $1 \leq \mu \leq t$ ) outras linhas possuem o mesmo numero de entrada. Nesta situação quando o algoritmo é suspenso após a primeira iteração o erro ocorre com probabilidade  $\frac{\mu}{\mu+1}$ . Considerando que o algoritmo é suspenso após resolver todas as ambigüidades, a probabilidade de erro de bit do algoritmo para canais somente de interferência pode ser expressa por :

$$P_b = \sum_{t=1}^{q-1} \binom{q-1}{t} (p) (1-p)^{q-1-t} \cdot \left[ \sum_{M_s=1}^{M-1} P_1 \binom{M-1}{M_s} (P_{cor})^{M_s} (1-P_{cor})^{M-1-M_s} + \frac{t}{t+1} (1-P_{cor})^{M-1} \right] \quad (3.30)$$

onde  $p$  é dado segundo a equação (3.1) e  $P_{cor} = (1-p)^{q-1}$  é a probabilidade de uma das matrizes  $D_m$  não conter ambigüidade.

Nos canais com desvanecimento seletivo em frequência, a probabilidade de detecção de chips na matriz recebida deve considerar os efeitos do ruído do canal expresso pelas probabilidades  $p_D$  e  $p_F$ . Como no caso anterior, considerando uma matriz de um usuário particular  $D_1$ , a probabilidade de que um chip seja detectado pelo receptor é dado por :

$$p_u = 1 - p_d \quad (3.31)$$

As probabilidades de que na matriz  $D_1$  a linha correta e uma linha incorreta possuam  $l$  entradas são dadas respectivamente por :

$$P_{cor}(l) = \binom{L}{l} (p_u)^l (1-p_u)^{L-l} \quad (3.32)$$

$$P_{incor}(l) = \binom{L}{l} (p_I)^l (1-p_I)^{L-l}$$

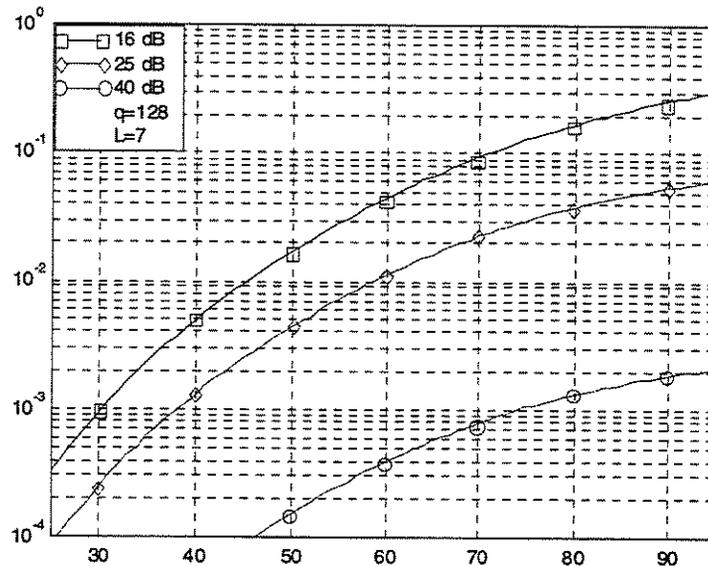


Figura 3.7: Desempenho do algoritmo de Fiebig para canais com ruído e desvanecimento seletivo em frequência,  $q=128$  e  $L=7$ .

onde  $p_I$  é a probabilidade de inserção de um chip na matriz devido ao efeito combinado da interferência e do ruído do canal segundo a equação (3.21).

As mesmas situações de erro citadas anteriormente num canal somente de interferência podem ocorrer no canal com desvanecimento seletivo em frequência. De forma similar pode-se obter um limitante inferior para a probabilidade de erro de bit do algoritmo de Fiebig num canal com desvanecimento seletivo em frequência, :

$$P_b = \sum_{l=0}^{L-1} P_{cor}(l) \left[ 1 - \left( \sum_{v=0}^l P_{inr}(v) \right)^{q-1} \right]. \quad (3.33)$$

É bom lembrar que este algoritmo resolve todas as ambigüidades das matrizes  $D_m$ , porém não corrige decisões de símbolos errados nas iterações.

As Figuras 3.7, 3.8 e 3.9 apresentam o desempenho do algoritmo de Fiebig para o canal com desvanecimento seletivo em frequência. Os resultados analíticos são obtidos para valores de relação sinal-ruído de 16, 25 e 40 dB.

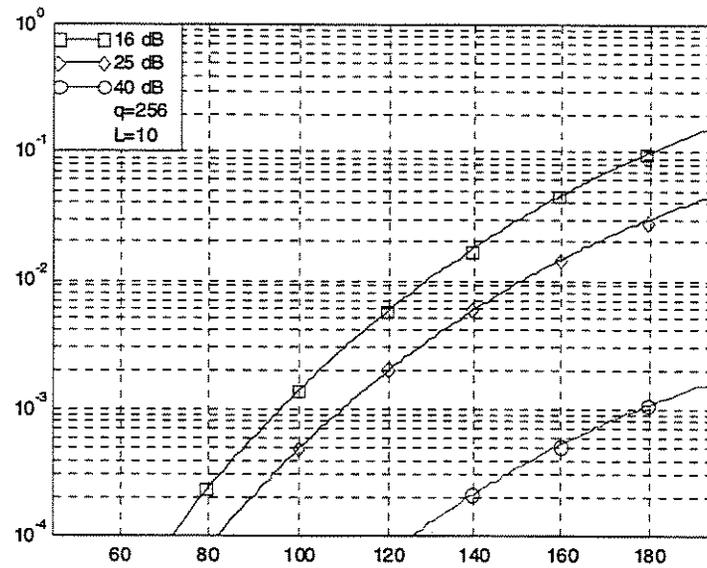


Figura 3.8: Desempenho do algoritmo de Fiebig para canais com ruído e desvanecimento seletivo em frequência,  $q=256$  e  $L=10$ .

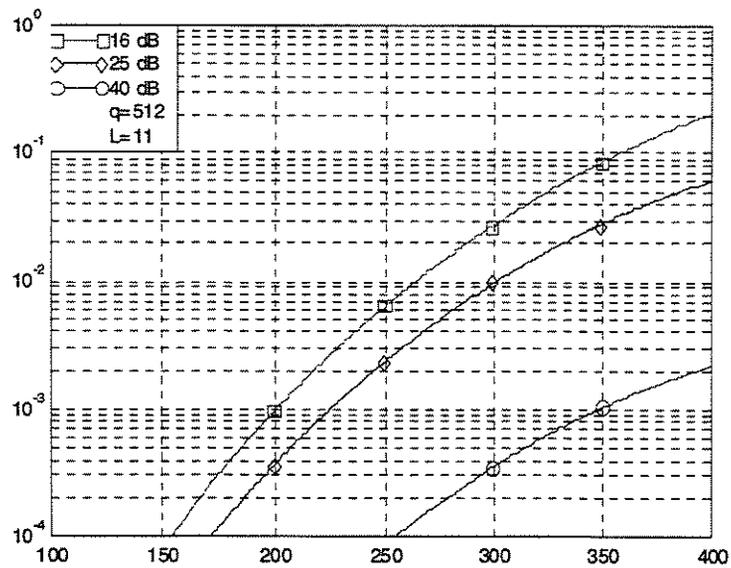


Figura 3.9: Desempenho do algoritmo de Fiebig para canais com ruído e desvanecimento seletivo em frequência,  $q=512$  e  $L=11$ .

### 3.4 Comparação dos Algoritmos

Comparando os dois algoritmos de cancelamento no aspecto de complexidade e desempenho, vemos que o algoritmo de cancelamento proposto por Timor se mostra mais atrativo. Por utilizar a bem definida estrutura algébrica de endereços dos usuários, este algoritmo consegue reduzir significativamente a complexidade de implementação. Nas curvas mostradas nas seções anteriores, observou-se que o algoritmo de Fiebig possui um melhor comportamento do que o algoritmo de Timor. Porém, se fixarmos uma probabilidade de erro de bit em torno de  $P_b \approx 10^{-3}$  (confiável para usuários de voz), vemos que o algoritmo de Timor consegue acomodar mais usuários. Analisando a complexidade de implementação, tem-se que o algoritmo de Timor realiza em torno de  $L(L - 1)$  operações para cada linha completa na matriz decodificada pelo endereço do usuário, enquanto o esquema de Fiebig realiza aproximadamente  $L(2^K - 1)$  operações. Se compararmos a complexidade destes dois algoritmos com parâmetros  $L$  e  $K$  grandes, vemos que algoritmo de Timor possui uma complexidade bem inferior, aumentando significativamente o desempenho.

## Capítulo 4

# Sistema Proposto

O esquema de codificação multinível juntamente com a decodificação multiestágio é um esquema de codificação de canal que tem sido aplicado com sucesso em diferentes sistemas de comunicações. Como já foi citado no capítulo 2, uma das principais características deste esquema é permitir uma grande flexibilidade na escolha das taxas dos códigos interno e externo, possibilitando uma otimização dos parâmetros, distância mínima, taxa de informação, desempenho e complexidade de implementação.

Quando o esquema de codificação multinível é aplicado em um sistema FH-CDMA, o conjunto de seqüências de saltos dos usuários, de acordo com a construção multinível, determinam um código interno. Em [Ein84] foi mostrado que o sistema FH-CDMA possui um melhor desempenho se as seqüências de saltos dos usuários constituírem um código *RS*. Desta forma, o número de geradores do código *RS* determina o número de níveis do esquema da construção multinível. Para se entender melhor o procedimento de codificação, é descrito na primeira parte do capítulo um esquema de codificação multinível para um sistema FH-CDMA com dois níveis de partição. O desempenho do sistema é obtido através de simulações computacionais considerando somente a interferência provocada pelos usuários. É também apresentado um esquema híbrido combinando a codificação multinível com o esquema de cancelamento de interferência de Timor [Tim80] visando melhorar o desempenho do sistema.

No esquema de decodificação multiestágio também é possível melhorar o desempenho

realizando aplicações repetidas (iterações) do decodificador multiestágio. Assim são descritos alguns esquemas com iterações visando melhorar gradativamente o desempenho do sistema. Por fim, simulações de um sistema com uma taxa de transmissão inferior é analisado e conclusões são obtidas.

#### 4.1 Descrição do Esquema de Construção Multinível para um Sistema FH-CDMA

Num sistema FH-CDMA a interferência entre os usuários pode ser minimizada se o conjunto de seqüências de saltos dos usuários (determinadas pela soma do vetor mensagem e endereço) constituírem um código de *RS* em que cada palavra-código de comprimento  $n$  e  $k$  bits de informação é gerada segundo a equação :

$$\bar{x} = x_0 \cdot \bar{1} + x_1 \cdot \bar{\beta}^{(1)} + x_2 \cdot \bar{\beta}^{(2)} + \dots + x_{k-1} \cdot \bar{\beta}^{(k-1)} \quad (4.1)$$

onde :

$$\checkmark x_i \in GF(q), i = 0, 1, \dots, k-1.$$

$$\checkmark \bar{\beta}^{(i)} = (1, \beta^i, \beta^{2i}, \dots, \beta^{(L-1)i}), i = 0, 1, \dots, k-1.$$

$\checkmark \beta$  é o elemento primitivo do corpo e  $\bar{1}$  é um vetor todo um de comprimento  $L$ .

No esquema de codificação multinível, o conjunto de palavras do código de *RS* é considerado como um código interno  $A$ . O código  $A = \{\bar{x}/x_0, x_1, \dots, x_{k-1} \in GF(q)\}$  é então particionado em uma cadeia de partições  $A_1/A_2/\dots/A_{m+1}$  que possui como características :

$$\checkmark A = A_1$$

$\checkmark A_2, \dots, A_{m+1}$  são códigos lineares sobre  $GF(q)$  de comprimento  $n_1$  onde  $A_{i+1} \subset A_i$ .

Estes conjuntos  $A_i$  também são denominados códigos internos.

Considerando que o símbolo  $x_0$  da equação (4.1) é utilizado para atribuir o endereço aos usuários do sistema FH-CDMA, o conjunto  $A_1$  passa a ser expresso como :

$$A_1 = \left\{ x_1 \cdot \bar{\beta}^{(1)} + x_2 \cdot \bar{\beta}^{(2)} + \dots + x_{k-1} \cdot \bar{\beta}^{(k-1)} \right\} \quad (4.2)$$

Desta forma, os números que especificam univocamente o código  $A_1$  são :

$$x_1 = y^{(1)}, x_2 = y^{(2)}, \dots, x_{k-1} = y^{(m)} \quad (4.3)$$

Com a utilização de  $m = k - 1$  códigos  $C_i$  de comprimento  $n_2$ , denominados códigos componentes externos forma-se o código multinível. Agrupa-se então em ordem cada palavra-código de  $C_i$  formando uma matriz  $Y$ ,  $m \times n_2$  :

$$Y = \begin{pmatrix} y_1^{(1)} & y_2^{(1)} & \dots & y_{n_2}^{(1)} \\ y_1^{(2)} & y_2^{(2)} & \dots & y_{n_2}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ y_1^{(m)} & y_2^{(m)} & \dots & y_{n_2}^{(m)} \end{pmatrix} = \begin{pmatrix} x_1^1 & x_1^2 & \dots & x_1^{n_2} \\ x_2^1 & x_2^2 & \dots & x_2^{n_2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{k-1}^1 & x_{k-1}^2 & \dots & x_{k-1}^{n_2} \end{pmatrix} \quad (4.4)$$

onde  $(y_1^{(i)}, y_2^{(i)}, \dots, y_{n_2}^{(i)}) = (x_i^1, x_i^2, \dots, x_i^{n_2})$  é a palavra-código do  $i$ -ésimo código componente externo  $C_i$ .

O código multinível é obtido mapeando-se cada coluna  $(y_j^{(1)}, y_j^{(2)}, \dots, y_j^{(m)})^T$  da matriz  $Y$  numa palavra-código de  $A_1$ , onde este por sua vez também pode ser expresso pelo vetor  $\bar{S} = (\bar{S}_1, \bar{S}_2, \dots, \bar{S}_{n_2})$  :

$$\bar{S}_j = \left( y_j^{(1)}, y_j^{(2)}, \dots, y_j^{(m)} \right)^T \cdot \bar{A} \quad (4.5)$$

com  $\bar{A} = (\bar{\beta}^{(1)}, \bar{\beta}^{(2)}, \dots, \bar{\beta}^{(m)})$ . A distância mínima e a taxa de informação do esquema de codificação multinível são dadas respectivamente segundo as equações (2.26) e (2.27).

Para um melhor entendimento do procedimento de codificação multinível em um sistema FH-CDMA é apresentado a seguir um sistema com três símbolos ( $k = 3$ ). Assim as seqüências de saltos dos usuários são expressas como :

$$\bar{x} = x_0 \cdot \bar{1} + x_1 \bar{\beta}^{(1)} + x_2 \bar{\beta}^{(2)} \quad (4.6)$$

onde  $x_1$  e  $x_2$  são os símbolos de mensagem e  $x_0$  representa o endereço do usuário. Este procedimento permite o máximo de  $q$  usuários com endereços distintos e cada um com  $q^2$  frequências.

Sem perda de generalidade, pode-se considerar o valor de  $x_0 = 0$  e particionar o conjunto  $A_1 = \{ \bar{x} \mid \bar{x} = x_1 \bar{\beta}^{(1)} + x_2 \bar{\beta}^{(2)} \}$ , onde  $x_1 = y^{(1)}, x_2 = y^{(2)}, \bar{\beta}^{(1)} = \bar{a}_1$  e  $\bar{\beta}^{(2)} = \bar{a}_2$ . Estas partições podem ser descritas por :

- ✓  $A_1$
- ✓  $A_2 = \{ x_2 \bar{\beta}^{(2)} \}$ .
- ✓  $A_3 = \{ \bar{0} \}$ , onde  $\bar{0}$  é o vetor todo nulo.

Os códigos componentes externos  $C_i$  são escolhidos de acordo com as necessidades requeridas pelo canal. Para ilustrar o funcionamento do sistema foram inseridos os códigos  $RS$   $C_1(3, 1)$  e  $C_2(3, 2)$  de repetição e paridade respectivamente. Uma observação a ser feita é com relação ao código de paridade  $C_2$ , onde o dígito de paridade é a soma dos dois primeiros dígitos de cada palavra-código.

Agrupa-se então cada palavra destes códigos formando a matriz  $Y$  como segue :

$$Y = \begin{pmatrix} x_1^1 & x_1^1 & x_1^1 \\ x_2^1 & x_2^2 & (x_2^1 + x_2^2) \end{pmatrix} \quad (4.7)$$

multiplicando as colunas temos que as seqüências de saltos enviadas  $\bar{S} = (\bar{S}_1, \bar{S}_2, \bar{S}_3)$  são :

$$\bar{S}_1 = x_1^1 \bar{\beta}^{(1)} + x_2^1 \bar{\beta}^{(2)}$$

$$\bar{S}_2 = x_1^1 \bar{\beta}^{(1)} + x_2^2 \bar{\beta}^{(2)}$$

$$\bar{S}_3 = x_1^1 \bar{\beta}^{(1)} + (x_2^1 + x_2^2) \bar{\beta}^{(2)}$$

Considerando um sistema com  $q = 7$  e  $L = 6$ , tem-se que  $\bar{\beta}^{(1)} = (1, 3, 2, 6, 4, 5)$  e  $\bar{\beta}^{(2)} = (1, 2, 4, 1, 2, 4)$ . Supondo codificar a informação  $(1, 0, 2)$ , inicialmente preenche-se a matriz  $Y$

#### 4.1. DESCRIÇÃO DO ESQUEMA DE CONSTRUÇÃO MULTINÍVEL PARA UM SISTEMA FH-CDMA53

por linhas agrupando cada palavra dos códigos  $C_1$  e  $C_2$  :

$$Y = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 2 & 2 \end{pmatrix} \quad (4.8)$$

e mapeando a matriz por colunas obtém-se as seqüências de saltos dadas por  $\bar{S} = (\bar{S}_1, \bar{S}_2, \bar{S}_3) = ((1, 3, 2, 6, 4, 5), (3, 0, 3, 1, 1, 6), (3, 0, 3, 1, 1, 6))$ .

A decodificação para o esquema de codificação multinível é realizada com múltiplos estágios. Supondo que as seqüências geradas acima fossem recebidas no decodificador, então a decodificação se procederia da seguinte maneira :

1. Sendo o código interno  $A_1$  curto, é possível a construção de uma tabela com todas as possíveis seqüências geradas pela equação (4.1). A tabela foi construída e dividida em classes laterais, onde cada classe lateral representa um símbolo em  $GF(q)$ . Para cada seqüência recebida  $\bar{S} = (\bar{S}_1, \bar{S}_2, \bar{S}_3)$  procura-se em cada uma das classes laterais ilustradas na Figura 4.1 a palavra mais próxima em termos de distância de Hamming. Como um exemplo, para a seqüência  $\bar{S}_1 = (1, 3, 2, 6, 4, 5)$  a palavra mais próxima se encontra na classe lateral  $A_{21}$ . Guarda-se então as distâncias de Hamming mínimas  $\{d_H^1, d_H^2, \dots, d_H^{q-1}\}$  de cada seqüência  $\bar{S}_j$  recebida.
2. Utilizando estas distâncias acima decodifica-se o código de repetição  $C_1$ . Como é um código curto, é possível executar a decodificação de  $C_1$  por tabela. Cada palavra-código possui como métrica a soma das distâncias mais próximas associadas aos seus símbolos. Sendo um código de repetição, todas as seqüências recebidas só possuirão palavras na mesma classe lateral. Assim obtém-se palavra estimada  $(\hat{y}_1^{(1)}, \hat{y}_2^{(1)}, \hat{y}_3^{(1)}) = (1, 1, 1) \in C_1$ .
3. Com a decodificação do código de repetição  $C_1$  é possível executar a decodificação do código  $C_2$  considerando que a palavra  $(1, 1, 1)$  estimada em  $C_1$  está correta. Então procura-se na classe lateral correspondente ao símbolo estimado pelo código  $C_1$  a palavra mais próxima em termos de distância de Hamming de cada seqüência recebida,

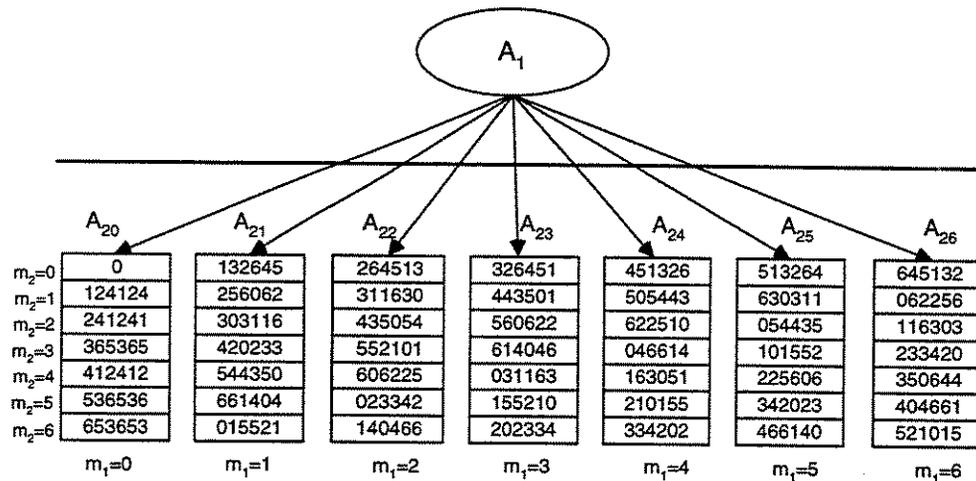


Figura 4.1: Descrição das classes laterais sobre  $GF(q)$ .

guardando as distâncias  $\{d_H^1, d_H^2, \dots, d_H^{q-1}\}$ .

- Utilizando estas distâncias, decodifica-se o código de paridade  $C_2$  por tabela. Novamente, cada palavra-código possui como métrica a soma das distâncias mais próximas associadas ao seus símbolos. Assim, obtêm-se a palavra estimada  $(\hat{y}_1^{(2)}, \hat{y}_2^{(2)}, \hat{y}_3^{(2)}) = (0, 2, 2) \in C_2$ .

#### 4.1.1 Desempenho do Sistema

O desempenho do esquema codificado descrito no exemplo foi obtido através de simulações computacionais. O desempenho foi obtido para um usuário fixo no canal, considerando-se apenas a interferência produzida pelos outros usuários também ativos. Um modelo de interferência aleatória foi adotado, onde a probabilidade de um chip na matriz recebida ser ativado por um ou mais usuários interferentes é dada segundo a equação (3.1).

Assume-se também que todos os usuários do sistema operam em sincronismo com relação ao quadro de  $L$  chips. A Figura ?? mostra os resultados da probabilidade de erro de bit em função do número de usuários no sistema para  $q = 7$  e  $L = 3, 5$  e  $6$ . Cada simulação consistiu de 50000 eventos de transmissão onde a probabilidade de erro de símbolo obtida é convertida para probabilidade de erro de bit aplicando um fator  $\left(\frac{q}{2(q-1)}\right) = \frac{7}{12}$  aos valores

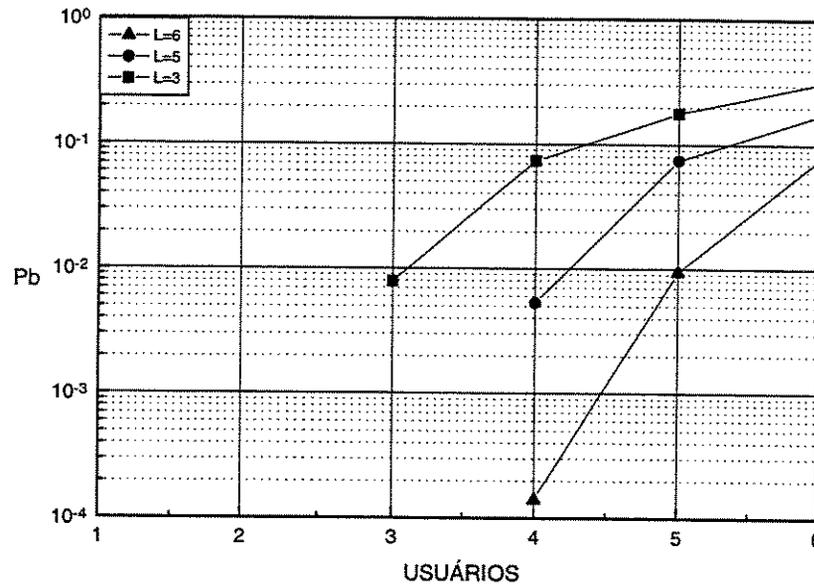


Figura 4.2: Desempenho do esquema proposto para  $q=7$  e  $L=3,5,6$ .

estimados. Foi comparado os resultados desta figura com os resultados mostrados na figura 10 de [Mao95]. Os resultados para  $L = 5$  são similares a curva (C') de [Mao95] com apagamento modificado. Os sistemas, a uma probabilidade  $P_b \leq 10^{-2}$ , comportam quatro usuários ativos.

## 4.2 Um Esquema Híbrido de Decodificação

O esquema de codificação multinível aplicado a um sistema FH-CDMA como visto na seção anterior, mostrou um bom desempenho em canais onde somente a interferência provocada pelos outros usuários causa a degradação do sistema. Um caminho para se melhorar este desempenho é a utilização de um esquema híbrido combinando o esquema de decodificação multiestágio com um esquema de cancelamento de interferência.

No capítulo 2, no estudo de algoritmos de cancelamento de interferência, foi mostrado que o algoritmo de cancelamento de Timor por possuir uma estrutura algébrica bem definida, reduz substancialmente a complexidade de implementação e melhora o desempenho do

sistema. Este sistema então, juntamente com a decodificação multiestágio se torna uma alternativa para se aumentar a eficiência do sistema. Mas, a primeira dúvida levantada é como utilizar o algoritmo de cancelamento de uma maneira eficiente sem que haja um aumento excessivo na complexidade.

É bom lembrar que no algoritmo de Timor, as seqüências de saltos dos usuários são transmitidas de acordo com a equação (3.7). Considerando  $x_0 = 0$ , observa-se que esta equação se torna uma particularização do esquema de codificação multinível, mais especificamente, o último nível da partição. Desta forma, a maneira mais eficiente é utilizar o algoritmo de cancelamento de interferência no último estágio da decodificação multiestágio.

No exemplo da seção anterior, temos que um usuário transmite as seqüências  $(\bar{S}_1, \bar{S}_2, \bar{S}_3)$ . Considerando também que outros  $M - 1$  usuários estão transmitindo suas seqüências de saltos de acordo com a equação (4.6), estas seqüências são então somadas através de uma operação lógica ou sendo recebidas pelo decodificador como matrizes de recepção. Então o esquema híbrido se procede como segue :

1. Para cada matriz recebida  $(R_1, R_2, R_3)$  repete-se os passos 1 e 2 do esquema de decodificação multiestágio descrito na última seção, obtendo-se a palavra estimada  $(\hat{y}_1^{(1)}, \hat{y}_2^{(1)}, \hat{y}_3^{(1)}) \in C_1$ .
2. A palavra estimada  $(\hat{y}_1^{(1)}, \hat{y}_2^{(1)}, \hat{y}_3^{(1)}) \in C_1$  é então multiplicada módulo- $q$  pelo vetor  $\bar{\beta}^{(1)} = (1, 3, 2, 6, 4, 5)$  produzindo os vetores  $(\bar{X}_1, \bar{X}_2, \bar{X}_3)$ . Estes vetores são retirados das matrizes recebidas da seguinte forma :

$$\hat{R}_i = R_i \ominus \bar{X}_i \quad i = 1, 2, 3 \quad (4.9)$$

onde  $\ominus$  denota a operação de subtração módulo- $q$ . Após a subtração, todos os chips permanecem ativos nas matrizes, porém, deslocados para posições diferentes.

3. Para cada matriz resultante  $\widehat{R}_i, i = 1, 2, 3$  aplica-se o algoritmo de cancelamento de Timor, realizando primeiramente a pré-multiplicação pelo vetor  $\overline{\beta}^{(-2)} = (1, 4, 2, 1, 4, 2)$  e conseqüentemente o teste de interferência em possíveis linhas completas. Nesta etapa, é importante destacar que o algoritmo de interferência só será aplicado quando houver duas ou mais linhas completas nas matrizes recebidas, por razões que serão descritas mais adiante.
4. Guarda-se no vetor  $(b_1, b_2, b_3)$ , os números correspondentes às linhas que falharam no teste de interferência. Multiplica-se este vetor módulo- $q$  pelo vetor  $\overline{\beta}^{(2)} = (1, 2, 4, 1, 2, 4)$  produzindo o vetor  $(\overline{Z}_1, \overline{Z}_2, \overline{Z}_3)$ .
5. Com a decodificação de  $C_1$  e a aplicação do algoritmo de interferência no estágio 2 é possível executar a decodificação do código  $C_2$ . Com a retirada da palavra estimada pelo código  $C_1$ , procura-se na classe lateral  $A_{20}$  a palavra mais próxima em termos de distância de Hamming de cada vetor recebido  $(\overline{Z}_1, \overline{Z}_2, \overline{Z}_3)$ , guardando o vetor de distâncias  $\{d_H^1, d_H^2, \dots, d_H^{q-1}\}$ .
6. Utilizando as distâncias, decodifica-se o código de paridade  $C_2$  por tabela. Novamente, cada palavra-código possui como métrica a soma das distâncias mais próximas associadas ao seus símbolos. Assim, obtém-se a palavra estimada  $\{\widehat{y}_1^{(2)}, \widehat{y}_2^{(2)}, \widehat{y}_3^{(2)}\} = (0, 2, 2) \in C_2$ .

#### 4.2.1 Desempenho do Sistema

O desempenho deste esquema híbrido também foi obtido através de simulações computacionais considerando um usuário fixo e somente a interferência provocada pelos outros usuários. O modelo de interferência adotado é baseado no endereçamento descrito na equação (4.6). A utilização deste modelo se deve ao fato do algoritmo de cancelamento de interferência possuir a condição de que uma linha é considerada como interferente se cada um de seus chips forem provenientes de usuários distintos cujas seqüências sejam transmitidas de acordo com a equação (4.6). Assume-se também que todos os usuários estão em sincronismo

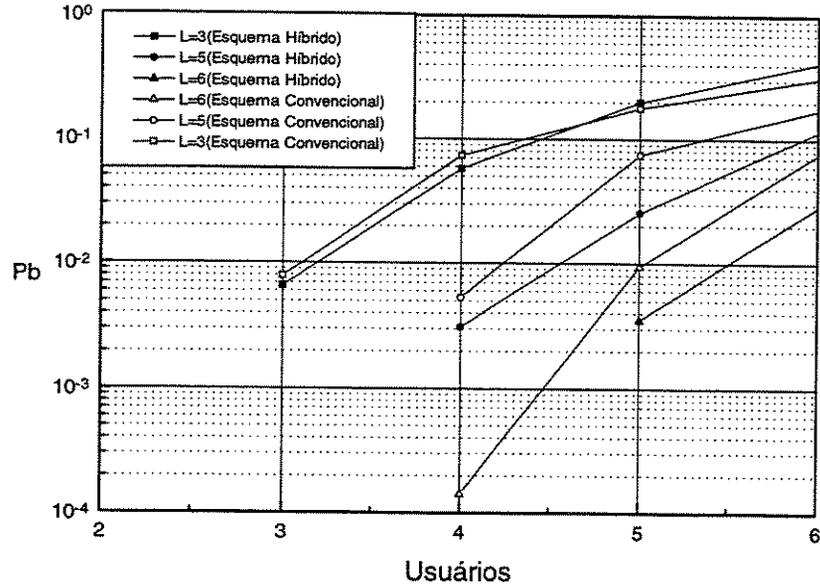


Figura 4.3: Desempenho do esquema híbrido em relação ao esquema convencional para  $q=7$  e  $L=3,5,6$ .

com relação ao quadro de  $L$  chips.

Nas simulações ocorrem situações em que após a retirada da palavra estimada pelo código  $C_1$  e a pré-multiplicação da matriz resultante  $\hat{R}_i$ ,  $i = 1, 2, 3$  pelo vetor  $\bar{\beta}^{(-2)}$ , não aparecem linhas completas nas matrizes. Isto ocorre quando a palavra estimada no primeiro estágio não corresponde a palavra correta. Assim, nas simulações assume-se que o algoritmo de cancelamento de interferência só é aplicado quando houver duas ou mais linhas em ambigüidade nas matrizes.

A Figura 4.3 mostra os resultados do esquema híbrido em relação ao esquema de codificação multinível e decodificação multiestágio descrito na seção anterior (chamado na figura de esquema convencional), para valores de  $L = 3, 5$  e  $6$ . Há uma pequena melhora no desempenho em relação ao esquema de decodificação multiestágio convencional, fato claramente explicado pela propagação de erro causada pela decodificação errada no primeiro estágio ocasionando erros no algoritmo de cancelamento e na decodificação dos estágios subsequentes, limitando o desempenho do esquema.

### 4.3 Aplicação de Iterações no Esquema de Decodificação Multiestágio

Mesmo com a aplicação do esquema híbrido na seção anterior, a propagação de erro causada devido a decodificação errada dos símbolos nos primeiros estágios da decodificação multiestágio é um dos fatores limitantes para o aumento no desempenho do sistema. Este problema pode ser reduzido com um aumento substancial na complexidade de decodificação, ou seja, realizando aplicações repetidas (iterações) do decodificador multiestágio. Estas iterações equivalem a assumir que as decodificações realizadas numa primeira etapa são consideradas apenas tentativas [Eri92].

Assim, buscando a possibilidade de melhora do desempenho do sistema foram propostos alguns esquemas de decodificação iterativos que serão abordados nos itens seguintes. Nos dois primeiros esquemas de decodificação é importante destacar que não foi realizada uma iteração completa após a decodificação multiestágio numa primeira etapa : o algoritmo simplesmente retorna ao primeiro estágio do decodificador sem decodificar o código  $C_2$ . Todos os esquemas foram simulados e comparados com o esquema híbrido e o exemplo das seções anteriores foi utilizado para as simulações.

#### 4.3.1 Esquema 1 : Esquema Híbrido e Decodificação no Primeiro Estágio Sem o Algoritmo de Cancelamento de Interferência

No esquema 1, após a decodificação utilizando o esquema híbrido numa primeira etapa, a palavra estimada pelo código  $C_2$  é considerada como correta e utilizada para a decodificação do código  $C_1$  novamente. Assim, a palavra  $(\hat{y}_1^{(2)}, \hat{y}_2^{(2)}, \hat{y}_3^{(2)}) \in C_2$  é multiplicada pelo vetor  $\bar{\beta}^{(1)}$  produzindo o vetor  $(\bar{L}_1, \bar{L}_2, \bar{L}_3)$  que é subtraído das matrizes recebidas  $(R_1, R_2, R_3)$ , segundo a expressão abaixo :

$$\hat{R}_i^{(1)} = R_i \ominus \bar{L}_i \quad i = 1, 2, 3 \quad (4.10)$$

Com a retirada da palavra estimada pelo código  $C_2$  é possível decodificar novamente o

código  $C_1$ . Procura-se então na primeira linha de cada classe lateral ilustrada na Figura 4.1 a palavra mais próxima em termos de distância de Hamming de cada matriz resultante  $\widehat{R}_i^{(1)}, i = 1, 2, 3$ . Decodifica-se o código utilizando estas novas distâncias encontradas. Os mesmos critérios da primeira decodificação podem ser utilizados novamente. Obtém-se então uma nova palavra estimada do código  $C_1$ .

#### 4.3.2 Esquema 2 : Esquema Híbrido e Decodificação no Primeiro Estágio Com o Algoritmo de Cancelamento de Interferência

A proposta do esquema 2 é utilizar o algoritmo de cancelamento de interferência antes da segunda decodificação do código  $C_1$ . Assim, é aplicado o algoritmo de cancelamento de interferência em cada uma das matrizes resultantes  $\widehat{R}_i^{(1)}, i = 1, 2, 3$  da seção anterior. Os números correspondentes às linhas que falharam no teste de interferência de cada matriz são guardados no vetor  $(b_1, b_2, b_3)$ . Este vetor é multiplicado pelo vetor  $\overline{\beta}^{(1)}$  produzindo o vetor  $(\overline{Z}_1^{(1)}, \overline{Z}_2^{(1)}, \overline{Z}_3^{(1)})$ . Segue a decodificação do código  $C_1$  procurando-se na primeira linha das classes laterais mostradas na Figura 4.1 a palavra mais próxima em termos de distância de Hamming de cada vetor  $\overline{Z}_i^{(1)}, i = 1, 2, 3$ . Obtém-se então uma nova palavra estimada do código  $C_1$ .

#### 4.3.3 Esquema 3 : Esquema Híbrido e uma Iteração Completa Sem o Algoritmo de Cancelamento

No esquema 3, após a decodificação em uma primeira etapa com o esquema híbrido, é realizada uma segunda decodificação dos códigos  $C_1$  e  $C_2$  sem a aplicação do algoritmo de cancelamento de interferência. Voltando ao esquema 1, assumindo que a segunda palavra do código  $C_1$  é considerada correta, é possível realizar a segunda decodificação do código  $C_2$ . Então, procura-se na classe lateral correspondente ao símbolo estimado no primeiro estágio a palavra mais próxima em termos de distância de Hamming das matrizes recebidas  $R_i, i = 1, 2, 3$ , obtendo-se a segunda palavra estimada pelo código  $C_2$ .

#### 4.3.4 Esquema 4 : Esquema Híbrido com Duas ou mais Iterações Sem o Algoritmo de Cancelamento

O objetivo deste esquema foi realizar várias iterações no intuito de verificar um limite no qual o sistema possibilite o máximo de usuários a uma dada probabilidade de erro de bit. Como no caso anterior todas as iterações são realizadas sem a aplicação do algoritmo de cancelamento de interferência que só é utilizado em uma primeira etapa de decodificação.

#### 4.3.5 Simulação dos Esquemas 1 a 4 e uma Comparação com o Esquema Híbrido

As Figuras 4.4, 4.5 e 4.6 mostram os resultados de simulação dos esquemas 1 a 4 em relação ao esquema híbrido para valores de  $L = 4, 5, 6$ . O mesmo exemplo e critérios utilizados nas simulações do esquema híbrido foram utilizados nestes esquemas. Observa-se que a medida que os esquemas vão sendo utilizados o desempenho do sistema em termos de número de usuários a dada probabilidade de erro de bit melhora gradativamente. Este aumento traz como consequência um aumento de complexidade de implementação.

É importante citar também que a variação de alguns parâmetros como taxa dos códigos e distância de Hamming pode levar o sistema a um melhor desempenho sem maiores consequência em termos de complexidade.

Nas simulações realizadas com o esquema 4 observou-se que após duas iterações o sistema não mais obtinha uma melhora no desempenho, assim é possível afirmar que com duas iterações o sistema se aproxima de um valor ótimo.

### 4.4 Simulação dos Sistemas Utilizando Dois Códigos de RS (3, 1)

Todos os esquemas apresentados até aqui foram simulados utilizando como códigos componentes externos  $C_1$  e  $C_2$ , os códigos  $RS(3, 1)$  e  $RS(3, 2)$  de repetição e paridade, respectivamente. De acordo com a equação (2.27) estes códigos fornecem ao sistema uma taxa de transmissão de informação igual a :

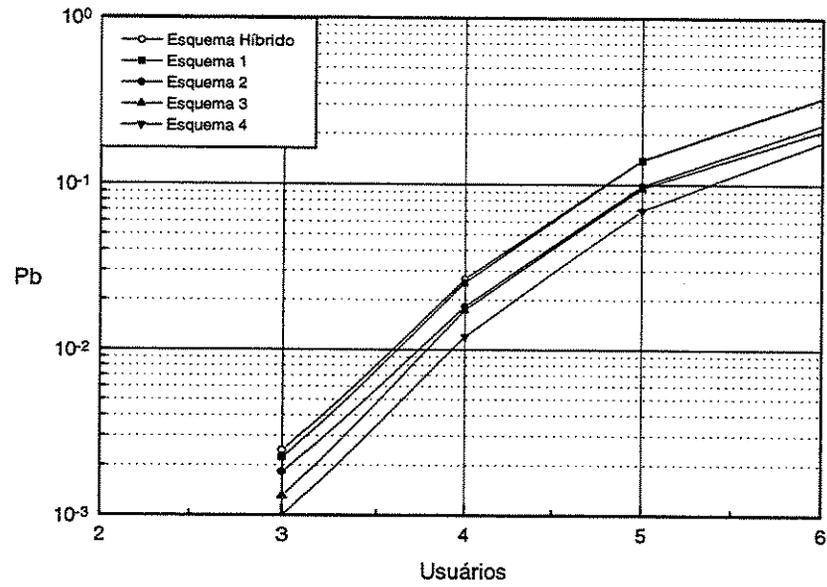


Figura 4.4: Desempenho dos esquemas 1 a 4 para  $q=7$  e  $L=4$ .

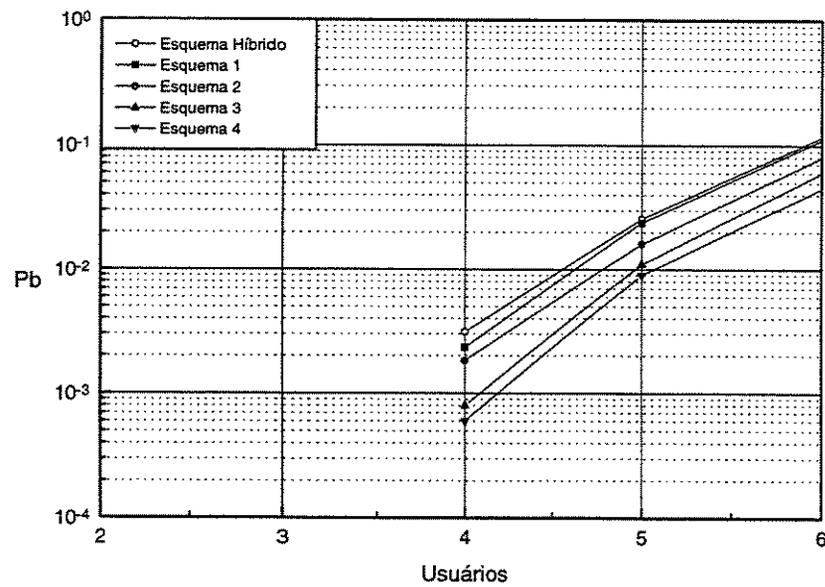


Figura 4.5: Desempenho dos esquemas 1 a 4 para  $q=7$  e  $L=5$ .

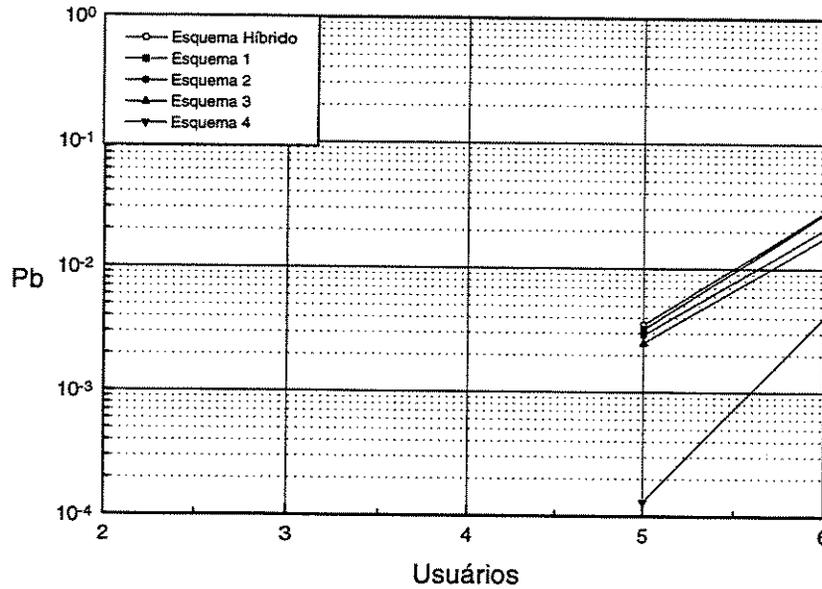


Figura 4.6: Desempenho dos esquemas 1 a 4 para  $q=7$  e  $L=6$

$$R = \sum_{i=1}^2 R_i \cdot \log_2 q = \frac{3}{3} \log_2 7 \text{ bits / uso do canal} \quad (4.11)$$

onde  $R_i = \frac{\log_2 q_i}{n}$  é a taxa do  $i$ -ésimo código componente externo  $C_i$  e  $r = \frac{R}{\log_2 q}$  é a taxa de transmissão de informação normalizada.

Acredita-se que o número de usuários no sistema pode ser aumentado pelo uso de códigos com menores taxas e com maior capacidade de correção, entretanto, o uso destes códigos pode levar o sistema a um aumento na largura de faixa de transmissão da ordem de  $\frac{1}{r}$ . Nesta seção está sendo proposta uma substituição do código de paridade  $C_2(3,2)$  por um código de repetição idêntico a  $C_1(3,1)$ , levando o sistema a uma taxa de transmissão de informação igual a :

$$R = \sum_{i=1}^2 R_i \log_2 q = \frac{2}{3} \log_2 7 \text{ bits / uso do canal} \quad (4.12)$$

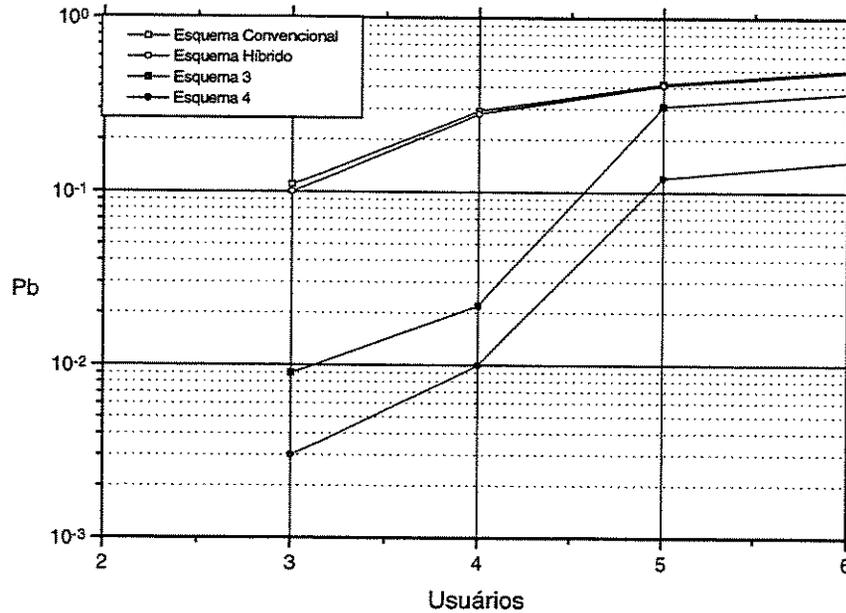


Figura 4.7: Desempenho dos esquemas utilizando o código de repetição (3,1) para  $q=7$  e  $L=4$

Este sistema foi simulado considerando os mesmos critérios e parâmetros utilizados nas seções anteriores. As simulações foram realizadas com todos os esquemas descritos exceto os esquemas 1 e 2 e os resultados obtidos estão ilustrados nas Figuras 4.7, 4.8 e 4.9.

Comparando estes resultados obtidos com os ilustrados nas Figuras 4.4, 4.5 e 4.6 observa-se que ocorre uma degradação no desempenho do sistema em termos de probabilidade de erro de bit por número de usuários.

Como já foi citado, o uso de códigos com pequenas taxas ocasiona uma expansão de largura de faixa da ordem de  $\frac{1}{r}$ . Por causa deste compromisso, existe uma taxa de código ótima que maximiza capacidade de usuários normalizada pela taxa do código. Em [Kaw91] foi derivada uma expressão em que a taxa de corte do canal, que representa um dos limitantes superiores da taxa de informação, é expressa em função do número de usuários  $M$ . Assim, é possível obter uma comunicação confiável num sistema codificado cuja taxa é  $r = \frac{R}{\log_2 q}$  se  $M$  satisfaz :

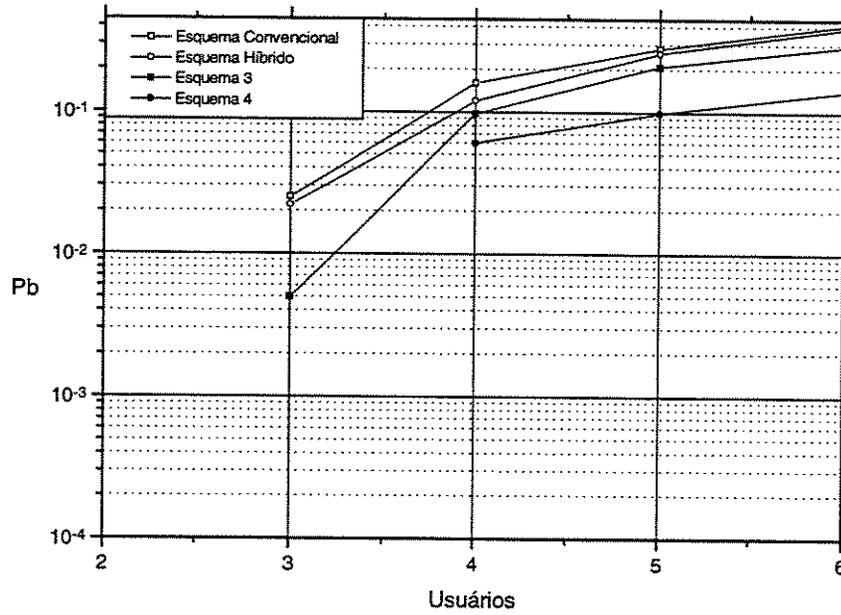


Figura 4.8: Desempenho dos esquemas utilizando o código de repetição (3,1) para  $q=7$  e  $L=5$ .

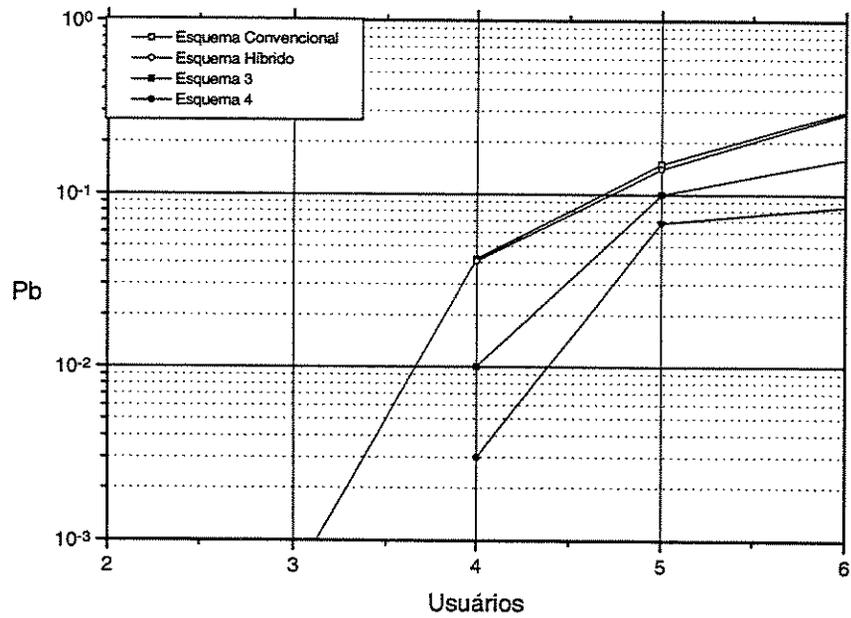


Figura 4.9: Desempenho dos esquemas utilizando o código de repetição (3,1) para  $q=7$  e  $L=6$ .

$$M \leq R_0^{-1} (r \log_2 q) \quad (4.13)$$

. Onde  $R_0$  é a taxa de corte do canal. Ainda em [Kaw91] foi calculada uma taxa ótima de aproximadamente 0.85 para um sistema com  $L = 19$  e  $2^k = 256$ .

Isto talvez seja a justificativa para a queda no desempenho mesmo ao se utilizar códigos com taxas menores e com maior capacidade de correção.

## Capítulo 5

# Conclusões

Esta tese apresentou um estudo de cancelamento de interferência para um sistema FH-CDMA. O objetivo foi propor algoritmos de cancelamento que melhorassem o desempenho do sistema original, sem um aumento excessivo em termos de complexidade. Os resultados obtidos através de simulações computacionais mostraram que os algoritmos mantiveram um bom compromisso entre desempenho e complexidade.

No capítulo 3 foi realizada uma comparação entre dois algoritmos de cancelamento de interferência com filosofias distintas : um primeiro baseado na estrutura algébrica dos endereços (proposto por Timor) e um segundo baseado na detecção multiusuário. ( proposto por Fiebig). Os resultados obtidos analiticamente mostraram que o algoritmo de cancelamento proposto por Timor se mostrou mais atrativo por possuir uma estrutura algébrica bem definida que reduz significamente a complexidade. As curvas mostram que mesmo o algoritmo de Fiebig mostrando um comportamento melhor, se fixarmos uma probabilidade de erro de aproximadamente  $P_b \approx 10^{-3}$  (satisfatória para sinais de voz), vemos que o algoritmo de Timor tem um melhor desempenho tanto para canais somente de interferência como canais com ruído Gaussiano Branco e Desvanecimento Seletivo em Frequência.

No capítulo 4, foi analisado um esquema de codificação multinível, juntamente com a decodificação em múltiplos estágios aplicado a um sistema FH-CDMA, proposto recentemente em [Mar97]. Foi mostrado através de simulações, que este sistema possui um bom desempenho para canais onde somente a interferência entre os usuários é considerada. O sistema

possibilita a transmissão de várias mensagens num intervalo de tempo, ou seja, um sistema bem mais geral do que esquemas recentemente propostos em que o número de mensagens transmitidas num intervalo de tempo está limitado a somente duas [Mao95]. Há também a possibilidade de se controlar a taxa dos códigos externos, garantindo um melhor compromisso entre desempenho do sistema e complexidade.

Ainda no capítulo 4, foi sugerido um esquema híbrido combinando o algoritmo de cancelamento proposto por Timor com o esquema de codificação multinível e decodificação multiestágio. Os resultados mostraram uma pequena melhora no desempenho quando comparado ao esquema de codificação multinível “convencional”. A justificativa encontrada foi que a ocorrência de erros nos primeiros estágios do decodificador, provoca uma propagação de erros que atinge os estágios subseqüentes e por sua vez também o algoritmo de cancelamento introduzido no último estágio.

Com a possibilidade de se realizar iterações no esquema de codificação multinível, foram propostos esquemas alternativos buscando amenizar a ocorrência destes erros de propagação. Os resultados obtidos nas simulações mostraram um aumento gradativo do desempenho, porém este desempenho vem acompanhado com um aumento de complexidade. Este fato nos levou a não implementar o algoritmo de cancelamento de Timor nas iterações realizadas com os esquemas 1 a 4, com exceção do esquema 2 onde é utilizado o algoritmo de cancelamento na segunda decodificação do código  $C_1$ .

Por fim, foi realizada uma análise dos esquemas utilizando como códigos componentes externos  $C_1$  e  $C_2$ , dois códigos de repetição (3, 1). Estes códigos fornecem ao sistema uma taxa de transmissão inferior ao exemplo mostrado no início do capítulo 4. Esperava-se um aumento no desempenho do sistema, porém, ocorreu uma degradação. A justificativa para esse fato pode ser baseada intuitivamente na literatura [Kaw91], onde é colocada que a utilização de códigos com pequenas taxas ocasiona uma expansão na largura de faixa da ordem de  $\frac{1}{R}$ . Por causa deste compromisso entre largura de faixa e taxa dos códigos existe uma taxa ótima que maximiza a capacidade de usuários normalizada pela taxa do código.

Para estudos futuros pretende-se explorar mais profundamente o esquema de codificação multinível com a implementação deste esquema em canais com desvanecimento seletivo em frequência e com outros esquemas alternativos, visando melhorar a capacidade do sistema.

## Apêndice A

# Esquemas de Endereçamento

### A.1 Sistema Síncrono

Considere  $y_1$  e  $y_2$  denotando duas seqüências geradas por dois diferentes endereços  $a_1$  e  $a_2$ , respectivamente.

$$\begin{aligned}\bar{y}_1 &= \bar{a}_1 + x_1 \cdot \bar{1} \\ \bar{y}_2 &= \bar{a}_2 + x_2 \cdot \bar{1}\end{aligned}\tag{A.1}$$

Em uma posição arbitrária  $i$  os símbolos são :

$$\begin{aligned}y_{1i} &= a_{1i} + x_1 = \gamma_1 \beta^{i-1} + x_1 \\ y_{2i} &= a_{2i} + x_2 = \gamma_2 \beta^{i-1} + x_2\end{aligned}\tag{A.2}$$

Assumindo  $y_{1i} = y_{2i}$  tem-se que :

$$(\gamma_1 - \gamma_2) \beta^{i-1} + x_1 - x_2 = 0\tag{A.3}$$

Em uma outra posição arbitrária  $j \neq i$

$$y_{1j} - y_{2j} = (\gamma_1 - \gamma_2) \beta^{j-1} + x_1 - x_2\tag{A.4}$$

substituindo em (A.2)

$$y_{1j} - y_{2j} = (\gamma_1 - \gamma_2) (\beta^{j-1} - \beta^{i-1}) \quad (\text{A.5})$$

De acordo com as suposições,  $a_1 \neq a_2$  implica que  $(\gamma_1 - \gamma_2) \neq 0$ . A equação (A.5) se aplica a qualquer posição arbitrária  $j \neq i$  e mostra que, com a escolha apropriada de  $\beta$ , dois vetores com endereços distintos coincidirão em no máximo um chip, independente das mensagens transmitidas.

## A.2 Sistema Assíncrono

Considerando novamente duas seqüências transmitidas  $y_1$  e  $y_2$  de usuários distintos

$$\bar{y}_1 = x_1 \cdot \bar{\beta} + \gamma_1 \cdot \bar{1} \quad (\text{A.6})$$

$$\bar{y}_2 = x_2 \cdot \bar{\beta} + \gamma_2 \cdot \bar{1} \quad (\text{A.7})$$

Considere a interferência entre  $y_1$  e  $y_2$  deslocada  $k$  posições a esquerda. O símbolo na posição  $i$  de  $y_1$  é :

$$y_{1i} = x_1 \beta^{i-1} + \gamma_1 \cdot 1 \quad (\text{A.8})$$

O símbolo da seqüência  $y_2$  que pode interferir com  $y_{1i}$  é :

$$y_{2i+k} = x_2 \beta^{i+k-1} + \gamma_2 \cdot 1 \quad (\text{A.9})$$

A condição de interferência é dada por :

$$y_{1i} - y_{2i+k} = x_1 \beta^{i-i} - x_2 \beta^{i+k-1} + \gamma_1 - \gamma_2 = 0 \quad (\text{A.10})$$

Dois símbolos  $y_{1j}$  e  $y_{2j+k}$  numa posição arbitrária  $j \neq i$  possui a diferença

$$y_{1j} - y_{2j+k} = x_1\beta^{j-i} - x_2\beta^{j+k-1} + \gamma_1 - \gamma_2 \quad (\text{A.11})$$

combinando as equações (A.10) e (A.11) tem-se :

$$\begin{aligned} y_{1j} - y_{2j+k} &= -\beta^{j-i}(\gamma_1 - \gamma_2) + \gamma_1 + \gamma_2 \\ &= (1 - \beta^{j-i})(\gamma_1 - \gamma_2) \neq 0 \end{aligned} \quad (\text{A.12})$$

Esta equação acima mostra que duas seqüências de usuários distintos irão se coincidir em no máximo um chip para qualquer deslocamento cíclico entre as seqüências.

# Bibliografia

- [Cos92] J. Wu, Costello D. J., "New Multilevel Codes Over  $GF(q)$ ," *IEEE Transactions on Information Theory*, vol. 38, no 3 , p. 933-939, maio 1992.
- [Ein80] G. Einarsson, "Address Assignment for a Time-Frequency Coded Spread Spectrum System," *Bell System Technical Journal*, vol. 59, n<sup>o</sup>7, p. 1241-1255, setembro 1980.
- [Ein84] G. Einarsson , " Coding for a Multiple-Access Frequency-Hopping System," *IEEE Transactions on Communications*, vol. 32, no 5, p. 589-596, maio 1984.
- [Eri92] C. Erik , W. Sundberg , N. Seshadri , "Coded Modulations for Fading Channels : An Overview ," *ETT*, vol. 4, no 3, p. 309-324, maio-junho 1993.
- [Fie96] U. C. G. Fiebig, " Iterative Interference Cancellation for FFH/MFSK MA systems ," *IEE Proceedings Communications*, vol. 143, n<sup>o</sup>6, p. 380-388, dezembro 1996.
- [Goh98] J. G. Goh and S. V. Maric, " The Capacities of Frequency-Hopped Code-Division Multiple-Access Channels ," *IEEE Transactions on Information Theory*, vol. 44, no 3, p.1204-1211, maio 1998.
- [Goo80] D. J. Goodman, P. S. Henry and V. K. Prabhu, "Frequency-Hopped Multilevel FSK for Mobile Radio ", *The Bell System Technical Journal* ,vol. 59, n<sup>o</sup>7, p. 1257-1275, setembro 1980.
- [Kaw91] T. Kawahara and T. Matsumoto, " Forward Link Capacity of Coded FFH/CDMA Mobile Radios ," *The 14th Symposium on Information Theory and its applications (SITA '91)*, p. 357-360, Ibusuki, Japão, 11 a 14 de dezembro,1991.

- [Mab94] T. Mabuchi, R. Kohno and H. Imai, "Multiuser Detection Scheme Based on Canceling Cochannel for MFSK/FH-SSMA System," *IEEE Journal on Selected Areas in communications*, vol 12, n°4, p.593-604, maio 1994.
- [Mac78] F. J. Macwillinas and N.J.A. Sloane, "The Theory of Error-Correcting Codes," *Amsterdam : North-Holland*, capítulo 10, p. 294-310, 1978.
- [Mao95] W. Mao, R. Kohno and H. Imai, "MFSK/FH-CDMA System with Two-Stage Address Coding and Error Correcting Coding and Decoding," *IEICE Transactions Fundamentals*, vol. E78-A, n°9, p. 1117-1125, setembro 1995.
- [Mar97] M. L. Mello, J. Portugheis, "Análise de um Sistema FH-CDMA," *LP-04-Relatório 1- Convênio CPqD 387/90, FEEC, UNICAMP*, março 1997.
- [Pra98] R. Prasad, "An Overview of CDMA Evolution Toward Wideband CDMA," *IEEE Communications Surveys - www.comsoc.org/pubs/surveys*, vol. 1, n°1, 1998.
- [Pro95] J.G. Proakis, "*Digital Communications*," McGraw-Hill, Third Edition, New York, 1995.
- [Ste66] M. Schwartz, W. R. Bennett and S. Stein, "*Communications Systems and Techniques*," McGraw-Hill, New York, p. 395-403, 1966.
- [Tim80] U. Timor, "Improved Decoding Scheme for Frequency-Hopped Multilevel System," *The Bell System Technical Journal*, vol. 59, no 10, p. 1839-1855, dezembro 1980.
- [Tim81] U. Timor, "Multistage Decoding of Frequency-Hopped FSK System," *The Bell System Technical Journal*, , vol. 60, n°4, p. 471-483, abril, 1981.