

Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação

**Receptores Iterativos para  
Canais de Acesso Múltiplo Ruidosos com  
 $N$  Frequências e  $T$  Usuários**

**Autor: Manish Sharma**

**Orientador: Prof. Dr. Jaime Portugheis**

**Tese de Doutorado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Doutor em Engenharia Elétrica. Área de concentração: Telecomunicações e Telemática.**

Banca Examinadora

Prof. Dr. Jaime Portugheis (Orientador)..... DECOM/FEEC/UNICAMP  
Prof. Dr. Paul Jean Etienne Jeszensky..... PTC/EPUSP  
Prof. Dr. Getúlio Antero de Deus Júnior..... EEEC/UFG  
Prof. Dr. Reginaldo Palazzo Júnior..... DT/FEEC/UNICAMP  
Prof. Dr. Max Henrique Machado Costa ..... DECOM/FEEC/UNICAMP

Campinas, SP

23 de Setembro de 2010

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

Sh23r Sharma, Manish  
Receptores iterativos para canais de acesso múltiplo ruidosos com N frequências e T usuários / Manish Sharma. –Campinas, SP: [s.n.], 2010.

Orientador: Jaime Portugheis.

Tese de Doutorado - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Telecomunicações - Modelos matemáticos. 2. Acesso múltiplo por divisão de código. 3. Códigos de controle de erros. 4. Teoria dos grafos - Processamento de dados. 5. Entropia. I. Portugheis, Jaime. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Iterative receivers for an N frequency T users multiple access channel with noise  
Palavras-chave em Inglês: Telecommunications - Mathematical models, Code division multiple access, Error control codes, Graph theory - Data processing, Entropy  
Área de concentração: Telecomunicações e Telemática  
Titulação: Doutor em Engenharia Elétrica  
Banca Examinadora: Paul Jean Etienne Jeszensky, Getúlio Antero de Deus Júnior, Reginaldo Palazzo Júnior, Max Henrique Machado Costa  
Data da defesa: 23/09/2010  
Programa de Pós Graduação: Engenharia Elétrica

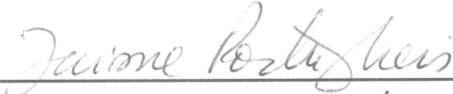
## COMISSÃO JULGADORA - TESE DE DOUTORADO

**Candidato:** Manish Sharma

**Data da Defesa:** 23 de setembro de 2010

**Título da Tese:** "Receptores iterativos para canais de acesso múltiplo ruidosos com N frequências e T usuários"

Prof. Dr. Jaime Portugheis (Presidente):



Prof. Dr. Paul Jean Etienne Jeszensky:



Prof. Dr. Getúlio Antero de Deus Júnior:



Prof. Dr. Max Henrique Machado Costa:



Prof. Dr. Reginaldo Palazzo Júnior:





# Resumo

O objetivo deste trabalho é analisar o desempenho da recepção e detecção conjunta e iterativa para canais de acesso múltiplo. A análise se concentrou em torno de um canal ruidoso com  $N$  frequências compartilhado por  $T$  usuários. Encontramos valores para a capacidade do canal para detecção conjunta e individual. Embora a eficiência espectral do sistema seja relativamente baixa, a combinação deste fator com uma grande faixa de frequências permite altas taxas de transmissão com baixa relação sinal ruído. O receptor foi modelado como um grafo de fatores e foi analisado através de curvas EXIT, que também são utilizadas para otimizar os códigos corretores de erro dos usuários. Propomos alguns sistemas baseados nesta técnica e simulamos a sua probabilidade de erro de bit. Os resultados indicam que é possível transmitir informação com taxas próximas da capacidade do canal. Tanto o grafo do receptor como as análises subsequentes podem ser aplicadas para outros canais de acesso múltiplo, especialmente para sistemas com  $N$  símbolos de transmissão ortogonais.

**Palavras-chave:** Sistemas multiusuário, capacidade de canal, detecção conjunta, processos iterativos, grafos de fatores.

# Abstract

The aim of this work is to analyze the performance of iterative joint reception and detection for multi-user channels. The analysis is centered around an  $N$ -frequency MFSK noisy channel shared by  $T$  users. Channel capacity values are obtained for joint and single user detection. Although the system's spectral efficiency is low, high rates at low signal to noise ratio are achievable by using a wide-bandwidth channel. The receiver is modeled as a factor graph and analyzed by its EXIT charts, which were also used to analyze the users' error correcting codes. Some systems are proposed and simulated to obtain the bit error probability. Results indicate that it is possible to transmit information with rates close to channel capacity. The proposed receiver and the performed analysis can be applied to other types of multiple access channels, in particular for systems with  $N$  orthogonal transmission symbols.

**Keywords:** Multi-user systems, channel capacity, joint detection, iterative processes, factor graphs.



# Agradecimentos

Ao meu orientador pelo direcionamento e motivação.

À minha família pela compreensão e suporte essenciais.

À Unicamp pela experiência.

Ao tempo, por passar.

Este trabalho foi financiado pela Fundação de Amparo à Pesquisa do Estado de São Paulo pelo processo 06/00330-7.

# Sumário

<b>Lista de Figuras</b>	<b>xi</b>
<b>Lista de Tabelas</b>	<b>xiii</b>
<b>Glossário</b>	<b>xv</b>
<b>Lista de Símbolos</b>	<b>xvii</b>
<b>Trabalhos Publicados Pelo Autor</b>	<b>xix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Problema estudado e metodologia . . . . .	2
1.2 Resultados anteriores . . . . .	3
1.2.1 Detecção individual, conjunta e cancelamento de interferência . . . . .	3
1.2.2 Capacidade do sistema MAC . . . . .	5
1.3 Notação utilizada . . . . .	6
1.4 Conceitos básicos . . . . .	7
1.4.1 Comunicação através de canais ruidosos . . . . .	7
1.4.2 Probabilidades . . . . .	8
1.4.3 Entropia, informação mútua e capacidade . . . . .	9
1.4.4 Códigos de bloco . . . . .	10
1.5 Organização da tese . . . . .	11
<b>2 Modelo do Sistema e Capacidade</b>	<b>13</b>
2.1 Modelo de canal . . . . .	14
2.1.1 Cálculo eficiente de $p(\mathbf{R})$ . . . . .	17
2.2 Capacidade soma . . . . .	18
2.2.1 Detecção de um único usuário . . . . .	19
2.2.2 Detecção conjunta . . . . .	21
2.2.3 Caso não ruidoso . . . . .	22
2.3 Resultados de capacidade . . . . .	23
2.3.1 Eficiência espectral . . . . .	25
2.3.2 Limite para taxas individuais para detecção sucessiva . . . . .	28
2.4 Resumo . . . . .	30

<b>3</b>	<b>Modelo e Análise EXIT do Detector Multiusuário</b>	<b>31</b>
3.1	Revisão sobre curvas EXIT . . . . .	32
3.1.1	Modelagem das mensagens recebidas . . . . .	35
3.1.2	Relação entre as mensagens transmitidas e o valor de informação . . . . .	37
3.2	Revisão sobre grafos de fatores . . . . .	38
3.3	Representação do detector como um grafo de fatores . . . . .	39
3.4	Cálculo das mensagens do detector multiusuário . . . . .	41
3.4.1	Mensagens calculadas no nó $P(m^t   b_0^t, b_1^t, \dots, b_{k-1}^t)$ . . . . .	42
3.4.2	Mensagens calculadas no nó $P(c_0^t, c_1^t, \dots, c_{N-1}^t   m^t)$ . . . . .	43
3.4.3	Mensagens calculadas no nó $P(c_n   c_n^1, c_n^2, \dots, c_n^T)$ . . . . .	43
3.4.4	Mensagens calculadas nos nós $p(R_n   c_n)$ . . . . .	45
3.5	Resultados . . . . .	46
3.6	Resumo . . . . .	50
<b>4</b>	<b>Análise de Códigos</b>	<b>53</b>
4.1	Equações fundamentais dos códigos de bloco lineares . . . . .	54
4.1.1	Transferência de informação para um nó-variável . . . . .	54
4.1.2	Transferência de informação para um nó-paridade . . . . .	56
4.1.3	Generalização para códigos irregulares . . . . .	57
4.2	Códigos LDPC . . . . .	58
4.3	Códigos LDGM . . . . .	61
4.4	Códigos RA . . . . .	64
4.5	Obtenção de curva EXIT global dos códigos . . . . .	67
4.6	Comparação entre códigos . . . . .	69
4.7	Resumo . . . . .	72
<b>5</b>	<b>Projeto de Sistemas</b>	<b>73</b>
5.1	Otimização das curvas EXIT dos códigos . . . . .	73
5.2	Construção dos códigos . . . . .	76
5.3	Simulação de probabilidade de erro de bit . . . . .	78
5.4	Resultados . . . . .	80
5.5	Resumo . . . . .	87
<b>6</b>	<b>Considerações Finais</b>	<b>89</b>
6.1	Estudos futuros . . . . .	91
<b>A</b>	<b>Tabelas de Capacidade</b>	<b>95</b>
	<b>Referências bibliográficas</b>	<b>102</b>

# Lista de Figuras

1.1	Exemplo de sistema FFH-CDMA . . . . .	4
2.1	Canal de acesso múltiplo ruidoso . . . . .	14
2.2	Distâncias entre a distribuição que otimiza a informação mútua e a distribuição uniforme. . . . .	23
2.3	Capacidade soma para $N = 4$ , $E_t/N_0 = 5dB$ e $10dB$ . . . . .	24
2.4	Capacidade soma para $N = 8$ , $E_t/N_0 = 5dB$ e $10dB$ . . . . .	24
2.5	Capacidade soma para $N = 16$ , $E_t/N_0 = 5dB$ e $10dB$ . . . . .	25
2.6	Vários valores de capacidade em função da relação sinal-ruído. . . . .	26
2.7	Eficiência espectral para várias combinações de $N$ , $T$ e $E_t/N_0$ . . . . .	27
2.8	Limites individuais para detecção sequencial para $N=8$ , $E_b/N_0 = 25dB$ . . . . .	29
2.9	Limites individuais para detecção sequencial para $N=16$ , $E_b/N_0 = 25dB$ . . . . .	29
2.10	Limites individuais para detecção sequencial para $N=32$ , $E_b/N_0 = 25dB$ . . . . .	30
3.1	Sistema iterativo simplificado. . . . .	33
3.2	Exemplo de análise EXIT . . . . .	34
3.3	Blocos de um decodificador Turbo (concatenação paralela). . . . .	34
3.4	Blocos de um decodificador com concatenação serial. . . . .	35
3.5	Fator local para $P(m^t b_0^t, b_1^t, \dots, b_{k-1}^t)$ . . . . .	41
3.6	Fator local para $P(c_0^t, c_1^t, \dots, c_{N-1}^t m^t)$ . . . . .	41
3.7	Fator local para $P(c_n c_n^1, c_n^2, \dots, c_n^T)$ . . . . .	41
3.8	Fator local para $p(R_n c_n)$ . . . . .	41
3.9	Grafo de fatores de um detector para um sistema com $T$ usuários e $N$ frequências. . .	42
3.10	Curvas EXIT do detector para $N = 8$ , $T = 4$ , $E_t/N_0 = 20dB$ , detecção paralela, parametrizada pelo número de iterações internas . . . . .	47
3.11	Combinação do detector multiusuário com os decodificadores individuais de cada usuário . . . . .	48
3.12	Curvas EXIT para $N = 4$ , detecção paralela, vários $E_t/N_0$ . . . . .	49
3.13	Curvas EXIT para $N = 8$ , detecção paralela, vários $E_t/N_0$ . . . . .	49
3.14	Curvas EXIT para $N = 16$ , detecção paralela, vários $E_t/N_0$ . . . . .	50
4.1	Grafo que representa um código LDPC regular com $d_v = 2$ e $d_c = 3$ . . . . .	60
4.2	Direção de troca de informações para MUD e LDPC. . . . .	60
4.3	Grafo que representa um código LDGM não sistemático regular com $d_v = 3$ e $d_c = 2$ . Para o caso sistemático, os nós que representam os bits também são transmitidos. . .	62

4.4	Grafo que representa um código RA não sistemático regular com $d_v = 3$ e $d_c = 2$ . Para o caso sistemático, os nós que representam os bits também são transmitidos. . . . .	65
4.5	Direção de troca de informações para MUD e RA. . . . .	65
4.6	Representação de uma instância do acumulador. . . . .	65
4.7	Curvas EXIT para alguns códigos LDPC. . . . .	68
4.8	Curvas EXIT para alguns códigos LDGM sistemáticos. . . . .	68
4.9	Curvas EXIT para alguns códigos RA sistemáticos. . . . .	69
4.10	Curvas EXIT para códigos RA sistemáticos obtidas por simulação. . . . .	70
5.1	Capacidade perdida devido a falta de ajuste das curvas. . . . .	74
5.2	Ligação de ramos através da função $\pi(i)$ . . . . .	77
5.3	Sistema multiusuário completo, detalhado para o primeiro usuário. . . . .	78
5.4	Ajuste de curvas EXIT de códigos RA sistemáticos para $N = 4, T = 2, (E_t/N_0)^* = 5dB$ . . . . .	81
5.5	Ajuste de curvas EXIT de códigos RA sistemáticos para $N = 4, T = 3, (E_t/N_0)^* = 5dB$ . . . . .	81
5.6	Ajuste de curvas EXIT de códigos RA sistemáticos para $N = 8, T = 2, (E_t/N_0)^* = 5dB$ . . . . .	82
5.7	Ajuste de curvas EXIT de códigos RA sistemáticos para $N = 16, T = 3, (E_t/N_0)^* = 5dB$ . . . . .	82
5.8	BER simulada do código RA otimizado para $N = 4, T = 2, (E_t/N_0)^* = 5dB$ . . . . .	84
5.9	BER simulada do código RA otimizado para $N = 4, T = 3, (E_t/N_0)^* = 5dB$ . . . . .	85
5.10	BER simulada do código RA otimizado para $N = 8, T = 2, (E_t/N_0)^* = 5dB$ . . . . .	85
5.11	BER simulada do código RA otimizado para $N = 16, T = 3, (E_t/N_0)^* = 5dB$ . . . . .	86
6.1	Subgrafo relacionando as variáveis $c_n$ e $R_n^1, R_n^2, \dots, R_n^L$ para o caso multiantena. . . . .	91

# Lista de Tabelas

2.1	Diferenças no cálculo da capacidade devido ao método de detecção . . . . .	18
3.1	Comparativo entre área sob a curva EXIT para detecção paralela e capacidade soma normalizada do sistema . . . . .	51
4.1	Aproximação linear por partes de $\Delta( x ) \approx \log(1 + \exp - x )$ . . . . .	56
4.2	Comparação entre área e taxas de códigos LDPC . . . . .	71
4.3	Comparação entre área e taxas de códigos LDGM sistemáticos . . . . .	71
4.4	Comparação entre área e taxas de códigos RA sistemáticos . . . . .	71
4.5	Valores que definem a complexidade de decodificação . . . . .	71
5.1	Parâmetros de códigos otimizados . . . . .	83
5.2	Desempenho de códigos encontrados . . . . .	83
A.1	Valores para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD), $N = 4$ . . . . .	96
A.2	Valores para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD), $N = 8$ . . . . .	97
A.3	Valores para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD), $N = 16$ . . . . .	98
A.4	Valores normalizados para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD), $N = 4$ . . . . .	99
A.5	Valores normalizados para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD), $N = 8$ . . . . .	100
A.6	Valores normalizados para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD), $N = 16$ . . . . .	101



# Glossário

ACC Acumulador

AWGN Additive white Gaussian noise

BER Bit Error Rate

bps Bits por segundo

CND Check Node Detector

EXIT Extrinsic Information Transfer

FFH-CDMA Fast Frequency Hopping CDMA

IS-95 Interim Standard 95

LDGM Low Density Generator Matrix

LDPC Low Density Parity Check

LLR Log-Likelihood Ratio

M-FSK Multiple frequency-shift keying

MAC Multiple Access Channel

MUD Multiple User Detection

MV Máxima verossimilhança

OOK On-Off Keying

RA Repeat Accumulate

REC Reduction of the number of candidates

SFH-CDMA Slow Frequency Hopping Code Division Multiple Access

SUD Single User Detection

UWB Ultra-wideband

VND Variable Node Detector



# Lista de Símbolos

- $B$  - Tamanho em bits da palavra código transmitida por usuário
- $\mathbf{c}$  - Vetor  $N$ -ário que indica quantos usuários estão transmitindo em cada uma das frequências do sistema M-FSK.
- $d$  - Relação entre o ruído e a energia por bit transmitido, vale  $d = N_0/E_t \cdot \log_2 N$ .
- $E_t$  - Energia por bit transmitido.
- $\mathbf{G}$  - Matriz geradora de um código de bloco.
- $\mathbf{H}$  - Matriz de verificação de paridade de um código de bloco.
- $\mathcal{H}\{x\}$  - Entropia da variável aleatória  $x$ .
- $\mathcal{H}\{x|y\}$  - Entropia da variável aleatória  $x$ , condicionado ao conhecimento de  $y$ .
- $\mathcal{I}\{x, y\}$  - Informação mútua entre  $x$  e  $y$ .
- $\mathbf{I}_K$  - Matriz identidade com  $K$  linhas.
- $I_{xy}^z$  - Valor da informação transmitida de  $x$  para  $y$ . A variável  $z$  vale  $A$  quando se trata de um valor a priori ou  $E$  quando se trata de um valor extrínseco. Os valores de  $x$ ,  $y$  e  $z$  podem ser omitidos quando definidos pelo contexto.
- $k$  - Número de bits transmitidos por uso do canal  $N$ -ário, vale  $\log_2 N$ .
- $\mathbf{m}$  - Vetor  $T$ -ário com as mensagens transmitidas por todos os usuários.
- $N$  - Número de frequências do sistema M-FSK.
- $N_0$  - Densidade do ruído aditivo Gaussiano branco existente no canal de transmissão.
- $P(x)$  - Função distribuição de probabilidade da variável aleatória  $x$ .
- $P(x|y)$  - Função distribuição de probabilidade da variável aleatória  $x$ , condicionada ao valor de  $y$ .
- $p(x)$  - Função densidade de probabilidade da variável aleatória  $x$ .
- $p(x|y)$  - Função densidade de probabilidade da variável aleatória  $x$ , condicionada ao valor de  $y$ .
- $\mathbf{R}$  - Vetor  $N$ -ário com os sinais recebidos em cada uma das  $N$  frequências do sistema M-FSK.
- $T$  - Número de usuários no sistema.
- $Tx_i$  - Taxa do  $i$ -ésimo usuário.
- $W$  - Banda de transmissão do canal.
- $\Phi(c_n, T, \zeta_n)$  - Distribuição binominal indicando a probabilidade de se haver  $c_n$  usuários ativos entre os  $T$  possíveis, onde  $\zeta_n$  é a probabilidade do usuário estar ativo.



# Trabalhos Publicados Pelo Autor

1. Manish Sharma, Jaime Portugheis, “On the sum capacity of a T-user N-Frequency Multiple Access Channel With Noise”, *European Transactions on Telecommunications*, V. 21, Janeiro de 2010
2. Manish Sharma, Jaime Portugheis, “Limitantes para Decodificação Iterativa de Sistemas MFSK Multi-usuário”, *XVI Simpósio Brasileiro de Telecomunicações (SBrT 2008)*, Rio de Janeiro, RJ, Brasil, Setembro de 2008.
3. Manish Sharma, Jaime Portugheis, “Iterative Decoding Bounds for Multiuser Multilevel FSK”, *IEEE 8th Workshop on Signal Processing Advances in Wireless Communications, 2007 (SPAWC 2008)*, Recife, PE, Brasil, Julho de 2008.
4. Manish Sharma, Jaime Portugheis, “Capacidade Soma para Canais de Acesso Múltiplo com T Usuários e N Frequências”, *XV Simpósio Brasileiro de Telecomunicações (SBrT 2007)*, Recife, PE, Brasil, Setembro de 2007.
5. Manish Sharma, Jaime Portugheis, “Procura de Codificadores BGU para Utilização em Códigos com Concatenação Serial”, *XXII Simpósio Brasileiro de Telecomunicações (SBrT 2005)*, Campinas, SP, Brasil, Setembro de 2005.



# Capítulo 1

## Introdução

A sociedade utiliza alguns sistemas de telecomunicação sem fio. O conjunto de padrões de comunicação sem fio da família IEEE 802 [1], com diversas aplicações, é um exemplo. Há sistemas com alcance menores tais como o Bluetooth [2] e WUSB (do inglês: *Wireless Universal Serial Bus*) [3] tradicionalmente utilizados para conectar dispositivos numa rede local. Estes sistemas tem ficado cada vez maiores, tanto no número de usuários como em taxa de transmissão. Como exemplo podemos comparar o sistema de telefonia celular digital, implantado no Brasil no final da década de 90 com taxas de 14400 bits/segundo/usuário (padrão IS-95), com o sistema de internet sem fio, implantando nos últimos anos com taxas de até 1 Mbps para o padrão HSPA (do inglês: *High Speed Packet Access*).

Os sistemas mencionados tem algumas características em comum. A faixa de frequências utilizadas forma um canal que é compartilhado por vários usuários ao mesmo tempo, denominado MAC (do inglês: *Multiple Access Channel*). Os usuários possuem códigos de acesso que permitem que cada um deles possa transmitir individualmente para um receptor em comum e vice-versa. O canal altera os sinais transmitidos através da adição de ruído e desvanecimento devido a existência de vários percursos entre o transmissor e o receptor.

É de se esperar que sistemas futuros permitam que os usuários transmitam com taxas ainda maiores. A taxa de transmissão de dados por um canal tem um limite que depende basicamente da faixa de transmissão utilizada e da relação sinal-ruído no receptor, como mostrado em [4]. Uma das formas de se transmitir com taxas maiores é utilizar grandes faixas de frequências de transmissão, ou bandas ultra largas (UWB - do inglês: *Ultra-WideBand*).

Entretanto, a utilização de faixas de transmissão muito grandes traz alguns problemas. Um deles é que o desvanecimento causado pela existência de múltiplos percursos pode não ser constante ao longo de toda a faixa, podendo ser mais forte ou mais fraco em algumas partes e ainda variar em função do tempo. Uma forma de solucionar este problema é dividir a faixa maior em  $N$  faixas

menores. Podemos considerar nesta situação que o desvanecimento em cada uma dessas  $N$  subfaixas é constante por um pequeno intervalo de tempo, o que simplifica a análise.

O sistema FFH-CDMA (do inglês: *Fast Frequency Hopping Code Division Multiple Access*) utiliza esta divisão da faixa juntamente com um padrão de salto entre subfaixas para permitir que vários usuários transmitam simultaneamente para o mesmo receptor. Alguns métodos de detecção para este sistema foram elaborados ao longo dos últimos 30 anos. A maioria destes métodos considera que a saída do receptor é discretizada em dois níveis. Os usuários são detectados individualmente ou sequencialmente.

Se os sistemas futuros devem permitir a transmissão de dados com a maior taxa permissível pelo canal, eles devem operar perto da capacidade do canal. A capacidade depende do tipo do sinal recebido (contínuo ou discreto) e do tipo de detecção utilizada (individual ou conjunta). É de se esperar que a capacidade seja maior para o sistema que recebe valores contínuos e realiza a detecção conjunta dos usuários.

## 1.1 Problema estudado e metodologia

Analisaremos um canal com banda  $W$  que sofre desvanecimento seletivo em frequência. Este canal será dividido em  $N$  sub canais com banda  $W/N$  cada. Consideraremos que, em cada um destes sub canais, o desvanecimento é plano, independente dos outros canais e variante no tempo, isto é, a banda de coerência do canal é igual a  $W/N$  [5]. O desvanecimento em cada um destes canais tem uma distribuição de Rayleigh.

Este canal é a base do sistema FFH-CDMA [6], na qual se adiciona um código de endereçamento para que se possa separar os usuários. A tecnologia Bluetooth [2] também é baseada neste canal, mas com um sistema SFH-CDMA (do inglês: *Slow Frequency Hopping Code Division Multiple Access*). A diferença entre o FFH-CDMA e SFH-CDMA está na relação entre a taxa de símbolos transmitidos e a taxa do código de endereçamento: se a taxa de símbolos é menor do que a taxa do código de endereçamento, o sistema é SFH; caso contrário, é FFH.

O código de endereçamento utilizado no sistema FFH-CDMA é uma solução do receptor para permitir o acesso múltiplo, mas ele não é parte integrante do canal. Ele é semelhante a um código de repetição tendo assim baixa eficiência espectral com o aumento do seu comprimento. Como o objetivo é transmitir com taxas altas em relação à capacidade do canal, eliminaremos este código e deixaremos a cargo do decodificador de canal tanto a correção de erros como a separação das palavras transmitidas pelos usuários.

Desta forma, podemos dividir o receptor em duas partes: detector conjunto e decodificadores, um para cada usuário.

Assim, as perguntas que desejamos responder são:

- Qual é a capacidade deste canal?
- Como podemos construir um detector conjunto para este sistema que utilize os valores contínuos recebidos do canal e valores de probabilidade fornecidos pelos decodificadores?
- Quais classes de código podemos utilizar em conjunto com este detector?
- Como podemos projetar sistemas para este canal, e quão próximos da capacidade podemos transmitir?

Estas perguntas podem ser respondidas utilizando conceitos da Teoria da Informação. Tendo em vista o sucesso que algoritmos de propagação de probabilidades (códigos Turbo e LDPC) obtiveram na decodificação de canal, é de se esperar que a aplicação destes conceitos para a detecção de usuários no sistema FFH-CDMA traga bons resultados. O detector será modelado como um grafo de fatores. Através deste modelo podemos obter curvas EXIT [7] (do inglês: *Extrinsic Information Transfer*) que serão úteis na escolha dos códigos para o sistema. Esperamos com isso obter um sistema iterativo para detecção dos usuários que estão transmitindo através deste MAC.

## 1.2 Resultados anteriores

### 1.2.1 Detecção individual, conjunta e cancelamento de interferência

Em [6] o sistema FFH-CDMA foi apresentado juntamente com um primeiro método de detecção de sinais. O espalhamento espectral é feito utilizando um vetor de endereçamento com dimensão  $L$  (diversidade) e a mensagem a ser transmitida, ambos com elementos do alfabeto  $0, \dots, N - 1$ . O vetor de endereçamento é chamado de padrão de salto. Para gerar o sinal transmitido, soma-se módulo  $N$  a mensagem a cada termo do padrão de salto, gerando assim uma sequência de comprimento  $L$  de elementos do alfabeto. Pode-se associar este vetor à uma matriz com  $N$  linhas e  $L$  colunas. Cada linha representa uma frequência e cada coluna representa um instante de tempo. A matriz recebida é obtida através da combinação das matrizes transmitidas. Os valores recebidos desta matriz podem ser quantizados, representando o uso ou não de uma frequência por um ou mais usuários em um dado instante. Devido ao ruído, pode haver erros na recepção que causem falsos positivos (o receptor decide que há usuários ativos numa célula quando não há) ou falsos negativos (o receptor não detecta a presença de usuários ativos numa célula). A detecção é feita aplicando o inverso do vetor de endereçamento à matriz, obtendo uma matriz desembaralhada. Este processo pode ser visualizado na Fig. 1.1. Decide-se pela linha que contém o maior número de entradas. Como há vários usuários,

é possível que na recepção mais de uma linha esteja completa. Neste caso, escolhe-se aleatoriamente uma das linhas.

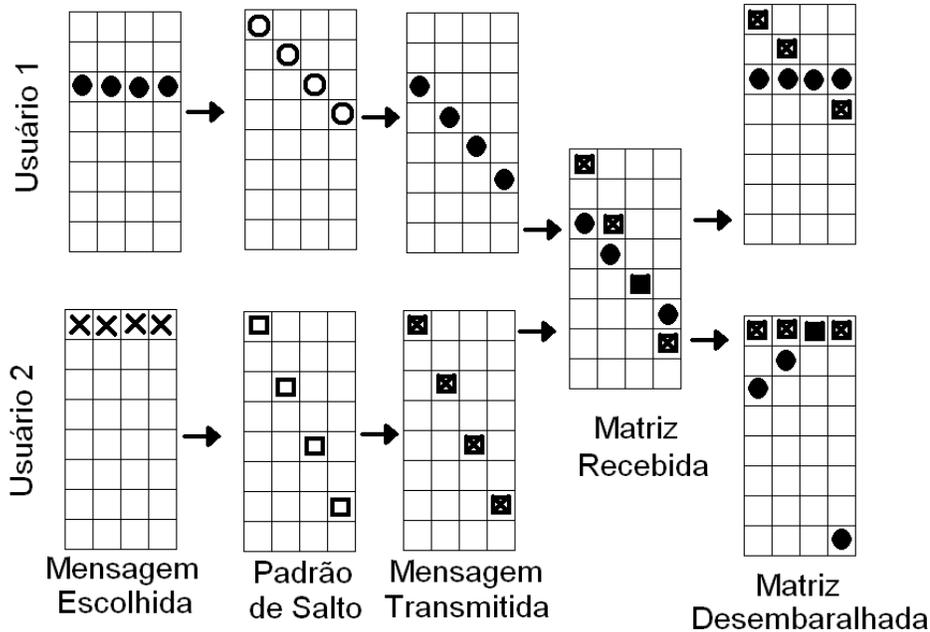


Fig. 1.1: Exemplo de sistema FFH-CDMA

Um método mais elaborado foi apresentado em [8]. Os padrões de salto são obtidos algebricamente, considerando geradores do grupo cíclico  $Z_N$ , com  $N = 2^k$ . Isto garante que, se um usuário sofre a interferência de outro usuário, o interferente o fará no máximo uma vez por linha da matriz. Assim, se uma linha da matriz desembaralhada está cheia, mas o usuário não está transmitindo nesta linha, há no mínimo  $L$  interferentes que preencheram esta linha. Isto permite distinguir quais linhas são causadas por interferência, e escolher apropriadamente.

Uma outra alternativa foi proposta em [9]. Quando há incerteza sobre qual mensagem foi transmitida para um número de usuários, gera-se todas as possíveis combinações de matrizes candidatas (considerando as possibilidades para cada usuário) e escolhe-se aquela mais próxima da matriz transmitida. Este algoritmo realiza decisão por máxima verossimilhança (MV) e realiza detecção conjunta, utilizando matrizes quantizadas. A complexidade deste algoritmo pode ser grande, e por isto uma variante foi proposta em [10]. Neste algoritmo, conhecido com REC (do inglês: *Reduction of the number of candidates*), o número de matrizes candidatas a serem comparadas com a matriz recebida é reduzido.

Métodos de cancelamento de interferência estão presentes em [11]. Quando há ambiguidade na decisão de pelo menos um usuário, as mensagens já decididas são eliminadas da matriz, e escolhe-se

a linha com o maior número de elementos restantes.

Os métodos anteriores utilizam saídas quantizadas em dois níveis: presença ou ausência de sinal. Em [12], uma variante do método anterior considera a saída quantizada em 3 níveis: ausência de sinal, presença de somente um usuário, ou presença de mais de um usuário. Este método tem desempenho semelhante ao algoritmo de [9], mas com complexidade inferior e mais estável (convergente). É apropriado tanto para canais Gaussianos como para canais Rayleigh.

O receptor com saídas contínuas foi estudado em [13], onde um método de detecção de um único usuário por MV foi apresentado. Neste caso, as saídas do receptor devem ser avaliadas por uma função não linear. Esta função foi aproximada utilizando redes neurais em [14]. O sistema FFH-CDMA codificado foi estudado em [15], onde o desempenho do sistema (eficiência espectral) foi comparado com um limitante para a capacidade do mesmo.

Receptores iterativos têm sido estudados para canais de acesso múltiplo mais simples, como o canal Gaussiano com dois ou mais usuários. Para este caso, é sabido que alguns conjuntos de taxas individuais são alcançáveis com detecção sucessiva [16]. Cada usuário tem uma quantidade de interferência que depende da sua ordem na sequência de detecção, pois a contribuição dos usuários já detectados podem ser retirados do sinal recebido, dada a natureza do canal. A interferência limita a taxa permitível para cada usuário, o que faz com que os últimos usuários tenham taxas mais altas do que os primeiros. Para que os usuários tenham taxas iguais, alguns esquemas são possíveis, entre eles o “*Time Sharing*”, onde a ordem de detecção e, conseqüentemente, a taxa dos usuários são trocados periodicamente, e o “*Rate Splitting*” [17], onde taxas iguais são obtidas para todos os usuários através de um arranjo de potências distintas de transmissão e ordem de detecção. Há dificuldades em se obter taxas iguais sem detecção sucessiva ou divisão de potências, como indicado em [18]. Métodos baseados em grafos de fatores tem mostrado resultados promissores [19], obtendo taxas próximas da capacidade para o canal Gaussiano com dois usuários. Resultados semelhantes para canais não coerentes com desvanecimento seletivo em frequência não foram encontrados.

### 1.2.2 Capacidade do sistema MAC

Resultados anteriores para a capacidade de um sistema MAC/FFH-CDMA são na maioria das vezes obtidos considerando simplificações do sistema, tais como detecção de um único usuário ou quantização da saída. Tratando-se de sistemas multiusuários, o conceito de capacidade de canal deve ser interpretado como uma região de capacidade

Um primeiro valor para a capacidade de um MAC com  $T$  usuários e  $N$  frequências foi apresentado em [20], onde foi considerado um sistema sem ruído. Dois canais foram considerados: um sem conhecimento de intensidade no receptor, designado canal A, e outro com conhecimento de intensidade, designado canal B. O principal resultado deste artigo é que, dado um valor de  $N$ , o canal A

apresenta um número de usuários que maximiza a capacidade do sistema, enquanto que a capacidade do canal B é sempre crescente. Não é feita nenhuma consideração sobre a taxa de transmissão individual de cada usuário. A capacidade apresentada refere-se à capacidade soma dos usuários, isto é, a soma das taxas individuais de cada usuário é limitada pela capacidade. Mostra-se também que a capacidade é atingida quando todos usuários utilizam a mesma distribuição uniforme nos símbolos de entrada.

Em [21], a capacidade de um canal FFH-CDMA é vista como a capacidade de um MAC (embora não explicitamente). Considera-se a capacidade de um único usuário, considerando os outros como interferência, e a saída é quantizada em dois valores. A capacidade é atingida quando o usuário utiliza uma distribuição uniforme dos símbolos de entrada. A capacidade soma, para este caso, pode ser definida como  $T$  vezes a capacidade individual.

Em [22], é apresentado um limitante para a capacidade individual de cada usuário. O sistema FFH-CDMA é chamado de OOK FH-CDMA (do inglês: *On-Off Keying FH-CDMA*). A saída é quantizada, e cada uma das  $N$  frequências é considerada como um canal binário cuja capacidade é facilmente calculada. Efeitos de ruído e desvanecimento são incluídos nas probabilidades de inserção/deleção de sinal. Esta capacidade é chamada de capacidade por dimensão. O limitante para a capacidade individual é definida como sendo  $N$  vezes a capacidade por dimensão. Este modelo é um limitante para o sistema quantizado pois não considera a dependência estatística existente entre os sinais presentes em cada uma das frequências.

Em [15], uma abordagem semelhante é feita, considerando que os valores recebidos em cada uma das frequências são estatisticamente independentes. Entretanto, a saída do canal não é quantizada. Utilizando as probabilidades de transição do canal contínuo com desvanecimento Rayleigh e ruído aditivo Gaussiano branco (AWGN - do inglês *Additive white Gaussian noise*) obtém-se uma fórmula que pode ser numericamente calculada. O resultado é um limitante superior para a capacidade individual de um usuário. Este limitante (equação (2.23)) é bem próximo dos valores obtidos para a capacidade real do sistema, como mostramos na Seção 2.3.

### 1.3 Notação utilizada

Para indicar que uma variável é um vetor ou uma matriz, utilizamos a letra em negrito, como por exemplo  $\mathbf{x}$ . Os vetores estão em letra minúscula ( $\mathbf{x}$ ) e as matrizes em letra maiúscula ( $\mathbf{X}$ ). A única exceção é o vetor de sinais recebido  $\mathbf{R}$ , que pode vir a ser uma matriz como comentado no capítulo 6. O  $i$ -ésimo termo do vetor  $\mathbf{x}$  é  $x_i$  ou  $x^i$ . Fazemos esta diferenciação em usar tanto o superscrito como o subscrito para podermos diferenciar por exemplo entre um índice que indica um usuário e o índice que indica uma frequência. O termo que está na linha  $i$  e coluna  $j$  da matriz  $\mathbf{x}$  é referenciado como

$x_{ij}$ . Dada uma variável  $b_i^j$ , um conjunto de variáveis deste tipo é escrito como  $\overline{b_i^j}$ . Este conjunto será para todo  $i$  ou todo  $j$ , sendo que isto será definido no contexto.

Uma função de densidade de probabilidade (*pdf*) é indicada pela letra minúscula  $p(x)$ , onde  $x$  é a variável aleatória. Podemos abusar desta notação para escrever  $p(x = x^*)$ , que indica o valor que a função assume quando  $x$  vale  $x^*$ . Para variáveis discretas, a probabilidade é descrita com a letra maiúscula  $P(x)$ . Em ambos os casos o argumento da função serve para identificar qual é a função. A probabilidade ou *pdf* condicional é indicada pelo símbolo  $|$ . Por exemplo,  $p(x|y)$  é a *pdf* de  $x$  dado o valor de  $y$ .

Os logaritmos são todos na base natural e são representados por  $\log[]$ , a não ser quando indicamos o contrário.

## 1.4 Conceitos básicos

### 1.4.1 Comunicação através de canais ruidosos

Os sistemas de telecomunicação são utilizados para tentar reproduzir num ponto o valor de uma variável que existe em outro ponto. Para que isso seja possível, transmite-se uma mensagem do transmissor para o receptor, através de um canal. A mensagem é transformada para que tenha um formato apropriado de transmissão pelo canal utilizado. Em muitos sistemas a mensagem é representada como uma sequência de bits.

Se o canal fosse perfeito, a mensagem recebida seria exatamente igual à mensagem transmitida, sempre. Entretanto, os canais utilizados na prática distorcem a mensagem transmitida. Esta distorção pode fazer com que, na recepção, o valor da mensagem seja trocado por outro, ou que o valor seja perdido, ou que não se tenha certeza sobre qual mensagem foi transmitida. Nestas situações ocorrem erros de transmissão. Se a distorção fosse exatamente a mesma sempre e para cada mensagem recebida houvesse somente uma mensagem transmitida, o seu efeito poderia ser contornado através de um mapeamento das mensagens recebidas nas mensagens transmitidas. Entretanto, a distorção é normalmente causada aleatoriamente. O canal possui então uma probabilidade de transição que nos diz qual é a probabilidade de recebermos a mensagem  $y$  dado que  $x$  foi transmitido. Quando  $y$  pode assumir valores contínuos, o canal possui uma função densidade de probabilidade de transição.

Para diminuir a chance de que a aleatoriedade resulte em erros de transmissão de mensagens, utiliza-se códigos corretores de erros. Estes códigos adicionam redundância à mensagem transmitida na tentativa de aumentar a confiabilidade da transmissão. Um código simples seria transmitir o mesmo bit três vezes, que é um código de repetição. Neste caso, o erro de transmissão ocorreria se houvesse no mínimo duas trocas de valor do bit. Se a probabilidade de troca de valor for  $p$ , a probabilidade de

ocorrerem dois ou três erros numa sequência de três bits é dada por  $p^3 + 3 \cdot p^2(1 - p)$ , que é menor do que  $p$  se  $p < 1$ .

A adição de redundância tem o seu custo. No exemplo mostrado, foram necessárias três utilizações do canal para se transmitir um único bit. Uma pergunta razoável é saber quanta redundância é necessária e suficiente para se transmitir as mensagens através de um canal. Desta forma, poderemos transmitir com a maior taxa possível.

A maior taxa de transmissão possível para um canal é limitada pela capacidade do canal, como mostrado em [4]. A capacidade é um valor que pode ser calculado através da (densidade de) probabilidade de transição de canal, como será revisto na Seção 1.4.3. O problema de projetar um sistema de comunicação é o problema de tentar transmitir informação de forma confiável e viável com taxas o mais próximas possíveis da capacidade do canal.

## 1.4.2 Probabilidades

Uma variável aleatória  $x$  que assume um conjunto de  $n$  valores discretos  $x_1, x_2, \dots, x_n$  possui uma distribuição de probabilidade  $P(x)$ . O valor de  $P(x = x_i)$  vale o percentual de vezes em que a variável  $x$  assume (ou assumiria) o valor de  $x_i$  dado um número infinito de realizações de  $x$ .

Da mesma forma, podemos obter uma distribuição de probabilidade para um par de variáveis  $x$  e  $y$  como sendo  $P(x, y)$ . A distribuição de  $x$ , condicionado a um valor de  $y$ , é escrita como  $P(x|y)$ . Assim,  $P(x, y) = P(y) \cdot P(x|y) = P(x) \cdot P(y|x)$ . Se  $y$  pode assumir os valores discretos  $y_1, y_2, \dots, y_n$  e sabemos o valor de  $P(y)$ , podemos obter o valor de  $P(x)$  através da seguinte equação:

$$P(x) = \sum_{i=1}^n P(y = y_i) \cdot P(x|y = y_i). \quad (1.1)$$

O Teorema de Bayes se beneficia do fato de que  $P(x, y)$  pode ser fatorado de duas formas para obtermos o valor de  $P(x|y)$  dado os valores de  $P(y|x)$ ,  $P(x)$  e  $P(y)$ :

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}. \quad (1.2)$$

A definição de probabilidade apresentada até o momento não pode ser utilizada se as variáveis assumem valores contínuos. Neste caso, utilizamos a função densidade de probabilidade (*pdf* - do inglês: *probability density function*), indicada pela letra minúscula  $p(x)$ . A função  $p(x)$  pode ser definida como aquela que atende a seguinte equação, para quaisquer valores de  $a$  e  $b$ ,  $a < b$ :

$$z = \int_a^b p(x)dx, \quad (1.3)$$

onde  $z$  é o percentual de vezes em que a variável  $x$  assume um valor entre  $a$  e  $b$  (inclusive), dado um número infinito de realizações de  $x$ . Quando  $a$  e  $b$  tendem respectivamente a  $-\infty$  e  $+\infty$ , o valor de  $z$  deve tender a 1.

Podemos definir as *pdf*'s condicional e conjunta de mesma forma como foi feito para o caso discreto. O Teorema de Bayes, por exemplo, assume o seguinte formato:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}, \quad (1.4)$$

quando tanto  $x$  como  $y$  são variáveis aleatórias contínuas.

Funções determinísticas também podem ser modeladas através de probabilidades utilizando a convenção de Iverson. Ela é definida pela notação  $[\cdot]$ , que vale 1 se o argumento for verdadeiro e 0 caso contrário. A densidade de probabilidade condicional de uma variável  $x$  que vale sempre  $2y$  pode ser escrita então como  $p(x|y) = [x = 2 \cdot y]$ .

### 1.4.3 Entropia, informação mútua e capacidade

Uma variável aleatória  $x$  possui uma entropia que pode ser interpretada como uma medida de incerteza. A entropia de uma variável  $x$  é dada pela função:

$$\mathcal{H}\{x\} = \mathcal{E}\{-\log[P(x)]\} = -\sum_x P(x) \log[P(x)], \quad (1.5)$$

onde  $\mathcal{E}\{\cdot\}$  é a função esperança. Considera-se que  $0 \cdot \log 0 = 0$ . Quando a base do logaritmo for 2, a medida da entropia será em bits. Pode-se interpretar o valor da entropia como o número médio de bits necessários para se representar a variável aleatória.

Da mesma forma como probabilidades e densidade de probabilidades, a entropia de uma variável  $x$  pode estar condicionada ao conhecimento de uma outra variável  $y$ . Se a variável  $y$  pode assumir os valores  $y_1, y_2, \dots, y_n$ , a entropia de  $x$ , dado o conhecimento da variável  $y$ , é dada por:

$$\mathcal{H}\{x|y\} = \sum_{i=1}^n P(y = y_i) \mathcal{H}\{x|y = y_i\}. \quad (1.6)$$

O valor da informação mútua entre duas variáveis nos diz em quanto se reduz a incerteza de uma variável, dado o conhecimento da outra. A informação mútua entre as variáveis  $x$  e  $y$  pode ser escrita como:

$$\mathcal{I}(x, y) = \mathcal{H}\{x\} - \mathcal{H}\{x|y\} = \mathcal{H}\{y\} - \mathcal{H}\{y|x\}. \quad (1.7)$$

Para os casos que estudaremos, podemos manipular a distribuição de uma das variáveis  $P(x)$ ,

que é denominada distribuição de entrada. A relação entre  $x$  e  $y$  não é manipulável e é definida pelo problema. A informação mútua é uma função da distribuição de entrada  $P(x)$  e assume um valor máximo. Este valor máximo é denominado capacidade. A capacidade é uma função côncava da distribuição de entrada [23]

Para o caso onde  $x$  e  $y$  são variáveis contínuas, as probabilidades são substituídas por funções de densidade de probabilidade e as somas por integrais. Neste caso o valor obtido aplicando a definição de entropia pode ser negativo e é denominado entropia diferencial. Entretanto, o valor da informação mútua e capacidade devem ser positivos e mantém o seu significado de transmissão de informação.

Para uma leitura mais aprofundada sobre o tema, vide [23].

#### 1.4.4 Códigos de bloco

Códigos corretores de erros são utilizados para identificar e possivelmente corrigir erros de transmissão introduzidos pelo canal de transmissão. Um dos tipos destes códigos são os códigos de bloco, que transformam um conjunto finito de  $K$  símbolos (mensagem) num conjunto finito de  $B$  símbolos (palavra código). Estudaremos somente os casos onde os símbolos são binários. Para que seja sempre possível determinar qual é a mensagem, dada uma palavra código, é necessário que  $B \geq K$ . Como há  $2^K$  mensagens possíveis, há  $2^K$  palavras-código, dentre as  $2^B$  sequências binárias. A taxa do código é dada por  $K/B$ .

Códigos de bloco lineares podem ser gerados utilizando uma matriz geradora  $\mathbf{G}$  com  $K$  linhas e  $B$  colunas. A mensagem  $\mathbf{u}$  pode ser codificada na palavra código  $\mathbf{v}$  através da seguinte relação:

$$\mathbf{v} = \mathbf{u} \cdot \mathbf{G}, \quad (1.8)$$

onde as operações de soma e multiplicação são respectivamente equivalentes às operações lógicas *XOR* e *AND*<sup>1</sup>, respectivamente. Se a matriz geradora  $\mathbf{G}$  possui o formato  $[\mathbf{I}_K | \mathbf{P}]$ , onde  $\mathbf{I}_K$  é a matriz identidade de posto  $K$ , o código é denominado sistemático, porque os primeiros  $K$  bits da palavra-código são idênticos à mensagem. Caso contrário, o código é denominado não sistemático. A matriz  $\mathbf{P}$  é denominada matriz de paridade e tem  $K$  linhas e  $B - K$  colunas.

Uma forma de identificar se uma palavra de comprimento  $B$  pertence ao código gerado pela matriz  $\mathbf{G}$  é através da matriz de verificação de paridade  $\mathbf{H}$ , com  $B$  linhas e  $B - K$  colunas. A relação entre  $\mathbf{G}$  e  $\mathbf{H}$  é dada por  $\mathbf{G}\mathbf{H}^T = \mathbf{0}$ . Por consequência, toda palavra-código  $\mathbf{v}$  gerada por  $\mathbf{G}$  deve satisfazer  $\mathbf{v} \cdot \mathbf{H}^T = \mathbf{0}$ . Se o código for sistemático, a matriz  $\mathbf{H}$  possui o formato  $[-\mathbf{P}^T; \mathbf{I}_{B-K}]$ .

Para uma leitura mais aprofundada sobre o tema vide [24]

<sup>1</sup>A operação  $XOR(a, b)$  pode ser interpretada como a soma módulo dois de  $a$  com  $b$ . A operação  $AND(a, b)$  vale um se  $a$  e  $b$  valem um e vale zero caso contrário

## 1.5 Organização da tese

Esta tese está dividida em seis capítulos.

No primeiro capítulo expusemos o problema a ser abordado e a motivação para estudá-lo. Apresentamos os resultados obtidos anteriormente, a notação utilizada e conceitos básicos sobre comunicação através de canais ruidosos, probabilidade, informação e capacidade, e códigos de bloco.

No segundo capítulo apresentamos o modelo matemático do canal e a partir deste obtemos valores para a entropia das variáveis. Isto nos permite obter o valor da capacidade do canal para alguns métodos de detecção.

No terceiro capítulo revisamos os conceitos de grafos de fatores e determinamos um grafo de fatores que permite obter os valores das probabilidades marginais dos bits transmitidos por todos os usuários dada uma realização do canal e um conjunto de probabilidades conhecidas de antemão sobre os bits transmitidos por todos os usuários. Revisamos o método de análise através de curvas EXIT e aplicamo-o ao detector.

No quarto capítulo analisamos os códigos dos usuários. Revisamos três tipos de códigos:

- LDPC (do inglês: *Low Density Parity Check*),
- LDGM (do inglês: *Low Density Generator Matrix*),
- RA (do inglês: *Repeat Accumulate*).

Os dois últimos podem ainda ser subdivididos entre sistemáticos e não sistemáticos. Mostramos que a partir das equações analíticas que definem curvas EXIT de parte dos códigos podemos obter curvas EXIT globais. Podemos assim analisar os códigos a partir destas curvas globais ao invés de utilizar as curvas das partes. Isto possibilita a comparação de classes diferentes de códigos, que foi a base da escolha em utilizar códigos RA sistemáticos no sistema projetado.

No quinto capítulo otimizamos os parâmetros dos códigos de forma a maximizar a taxa destes, dada uma curva EXIT do detector. Mostramos que, mesmo sem um código de acesso, é possível transmitir com taxas próximas do valor da capacidade.

No sexto e último capítulo apresentamos as considerações finais e propostas de estudos futuros decorrentes deste trabalho.



## Capítulo 2

# Modelo do Sistema e Capacidade

Neste capítulo desenvolvemos um modelo para o sistema multiusuário estudado, baseado nos trabalhos desenvolvidos em [20] e [13]. Uma classe de MAC com  $T$  usuários e  $N$  frequências foi investigado em [20]. Dois canais específicos foram estudados. Em ambos os modelos, os  $T$  usuários têm um alfabeto de entrada em comum com  $N$  símbolos, o que pode ser interpretado como os sinais de uma constelação M-FSK (do inglês: *Multiple frequency-shift keying*). Entretanto, os alfabetos de saída são diferentes. Para o canal sem conhecimento de intensidade, o símbolo de saída é binário e indica somente se uma frequência de entrada está sendo utilizada por algum usuário ou não. Este canal é referido como canal A. Para o canal com conhecimento de intensidade, o símbolo de saída indica quanto usuários estão utilizando uma certa frequência de entrada. Este canal é referido como canal B. Em ambos os modelos assume-se que não ocorrem erros devido a ruído ou outras fontes externas de alteração do sinal.

Consideramos a presença de erros, modelando o canal MAC ruidoso como a concatenação de um canal B com um canal ruidoso. Para avaliar o desempenho deste MAC ruidoso, consideramos a capacidade soma, um limitante superior para a soma das taxas permissíveis para todos os usuários.

Analisamos a capacidade soma para um MAC ruidoso que inclui desvanecimento Rayleigh e AWGN quando a detecção conjunta é realizada (MUD - do inglês: *Multiple user Detection*). O número de dimensões das integrais envolvidas torna o cálculo numérico destas computacionalmente complexo e inviável atualmente. É necessário utilizar integração por Monte Carlo. Entretanto, os valores de um limitante superior simples estão bem próximos dos valores encontrados através das integrais de Monte Carlo. Mostramos que, para este MAC, um produto de distribuições de entrada uniformes fornece um valor para a informação mútua que é ou a capacidade soma do canal ou um valor muito próximo desta.

Em alguns sistemas de acesso múltiplo, em vez de realizar a detecção conjunta das mensagens transmitidas por todos os usuários, o receptor pode detectar somente a mensagem de um único usuá-

rio. Isto é normalmente denominado detecção de usuário individual SUD (do inglês *Single User Detection*). Para este receptor mais simples, definimos a capacidade soma e a comparamos com os valores obtidos para o MAC ruidoso. Além disso, provamos que a distribuição uniforme dos símbolos de entrada é a distribuição que atinge a capacidade do canal neste caso.

## 2.1 Modelo de canal

O modelo para o canal MAC considerado é baseado no diagrama mostrado na Fig. 2.1. O canal ruidoso,  $p(\mathbf{R}|\mathbf{c})$ , é derivado do modelo descrito em [13].

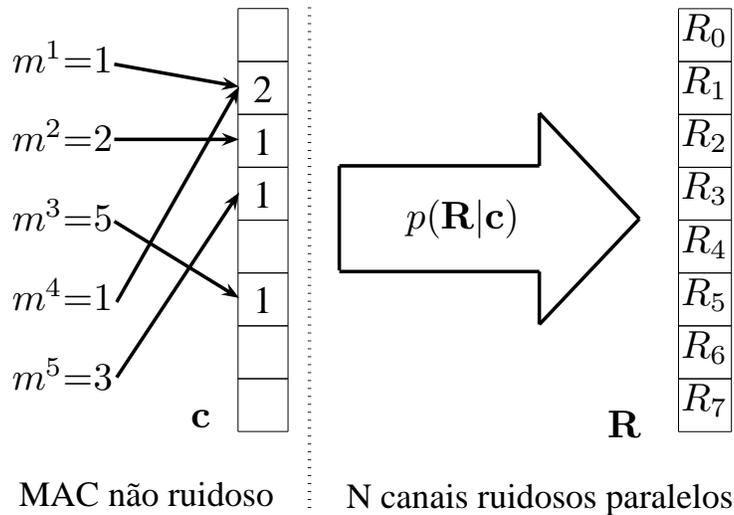


Fig. 2.1: Canal de acesso múltiplo ruidoso

Há  $T$  usuários. Num dado instante, cada um deles independentemente escolhe uma mensagem dentre um alfabeto com  $N$  símbolos. O símbolo escolhido pelo  $j$ -ésimo usuário é  $m^j$ , um inteiro entre 0 e  $N - 1$ . As mensagens escolhidas por todos os usuários podem ser representadas por um vetor  $\mathbf{m} = \{m^1, m^2, \dots, m^T\}$ . Cada símbolo está associado a um chip, uma banda no espectro de frequências. Cada mensagem  $m^j$  é convertida num vetor  $\mathbf{c}^j$  com dimensão  $N$ , com termos  $c_n$  que valem 1 se  $m^j = n$  ou 0 caso contrário.

O elemento  $c_n^j$  (um chip) está associado a sinais com duração  $\tau$ , transmitido com energia  $E_c = S\tau$ . O canal inteiro tem banda  $N/\tau$ , dividido em  $N$  sub-bandas (chips) com largura  $1/\tau$ . A energia por chip está associada com a energia por bit transmitido  $E_t$  por  $E_c = E_t \log_2 N$ .

Definimos as seguintes formas ortogonais básicas, com duração  $\tau$ :

$$\begin{aligned} x_n(t) &= \sqrt{2S} \cos 2\pi[f_0 + n/\tau]t, \\ y_n(t) &= \sqrt{2S} \sin 2\pi[f_0 + n/\tau]t, \end{aligned} \quad (2.1)$$

onde  $f_0$  é uma frequência base e  $n = 0, 1, \dots, N - 1$ .

O sinal que o  $j$ -ésimo usuário transmite no  $n$ -ésimo chip é então dado por:

$$s_n^j(t) = c_n^j x_n(t). \quad (2.2)$$

Inicialmente, podemos considerar que todos os usuários transmitem com a mesma fase. Na recepção, cada usuário sofre um desvanecimento Rayleigh  $\alpha_n^j$  e um atraso de fase aleatório  $\phi_n^j$  (uniformemente distribuído), todos estatisticamente independentes entre si. Há também um único componente de ruído branco Gaussiano aditivo  $n(t)$  com densidade  $N_0$ . O sinal recebido resultante é:

$$r_n(t) = \sum_{j=1}^T \{c_n^j \alpha_n^j [\cos \phi_n^j x_n(t) + \sin \phi_n^j y_n(t)]\} + n(t). \quad (2.3)$$

A energia de cada chip é detectada através de filtros casados:

$$\begin{aligned} X_n &= \frac{1}{E_c} \int_0^\tau r_n(t) x_n(t) dt = \kappa_n + \sum_{j=1}^T c_n^j \alpha_n^j \cos \phi_n^j, \\ Y_n &= \frac{1}{E_c} \int_0^\tau r_n(t) y_n(t) dt = \lambda_n + \sum_{j=1}^T c_n^j \alpha_n^j \sin \phi_n^j, \end{aligned} \quad (2.4)$$

onde  $\kappa_n$  e  $\lambda_n$  são variáveis aleatórias Gaussianas correspondentes ao ruído, com média zero e variância igual a  $d/2$ ,  $d = N_0/E_c$ . Pode ser mostrado que  $X_n$  e  $Y_n$  são descorrelacionadas.

Seja  $c_n = \sum_{j=1}^T c_n^j$  o número de usuários que estão utilizando o  $n$ -ésimo chip. Cada um dos termos  $\alpha_n^j \cos \phi_n^j$  ou  $\alpha_n^j \sin \phi_n^j$  é uma variável Gaussiana com média zero e variância igual a  $1/2$ . Logo,  $X_n$  e  $Y_n$  são variáveis Gaussianas independentes com média zero e variância igual a  $(c_n + d)/2$ .

A saída para o  $n$ -ésimo chip é obtida somando-se o quadrado de  $X_n$  e  $Y_n$ :

$$R_n = X_n^2 + Y_n^2. \quad (2.5)$$

A densidade de probabilidade (*pdf*) de  $R_n$ , condicionada ao número de usuários ativos no  $n$ -ésimo chip, é uma função exponencial em  $R_n$  com parâmetro  $c_n + d$ :

$$p(R_n|c_n) = \frac{1}{c_n + d} \exp \left[ -\frac{R_n}{c_n + d} \right]. \quad (2.6)$$

Assumindo que todos os usuários utilizam a mesma distribuição de entrada, o número de usuários ativos por chip tem uma distribuição binomial, dada por:

$$P(c_n) = \Phi(c_n, T, \zeta_n) = \binom{T}{c_n} \zeta_n^{c_n} (1 - \zeta_n)^{T-c_n}, \quad (2.7)$$

onde  $\zeta_n$  é a probabilidade de um usuário utilizar a  $n$ -ésima frequência.

É possível agora escrever uma equação que descreve a densidade de probabilidade de  $R_n$ , condicionado a  $c_n$ :

$$p(R_n) = \sum_{c_n=0}^T P(c_n) p(R_n|c_n). \quad (2.8)$$

Dada as mensagens de todos os usuários, os valores de  $R_n$  são estatisticamente independentes, e podemos escrever a seguinte equação para a densidade de probabilidade de  $\mathbf{R} = \{R_0, R_1, \dots, R_{N-1}\}$ :

$$p(\mathbf{R}|\mathbf{m}) = p(\mathbf{R}|\mathbf{c}(\mathbf{m})) = \prod_{n=0}^{N-1} p(R_n|c_n), \quad (2.9)$$

onde  $\mathbf{c} = \{c_0, c_1, \dots, c_{N-1}\}$  é uma função determinística de  $\mathbf{m}$ .

Entretanto, descondicionado nos valores das mensagens, os valores de  $R_n$  não são estatisticamente independentes. Podemos escrever a seguinte equação para descrever a densidade de probabilidade do vetor  $\mathbf{R}$ , com dimensão  $N$ :

$$p(\mathbf{R}) = \sum_{\mathbf{c} \in \Gamma(\mathbf{c})} P(\mathbf{c}) \prod_{n=0}^{N-1} p(R_n|c_n), \quad (2.10)$$

onde  $\Gamma(\mathbf{c})$  é o conjunto de todos os possíveis valores para  $\mathbf{c}$  tais que  $\sum_{i=0}^{N-1} c_i = T$ .

Para a detecção de um único usuário, estamos interessados nas formas de  $p(R_n|m^j)$  e  $p(\mathbf{R}|m^j)$ . Para a primeira *pdf*, procedemos de forma semelhante à equação (2.8). O número de usuários ativos varia entre 1 e  $N$  se o usuário está usando a frequência em questão ou entre 0 e  $N - 1$  se não estiver. Assim sendo, podemos escrever:

$$p(R_n|m^j) = \sum_{j=0}^{T-1} \Phi(j, T-1, \zeta_n) \frac{\exp \left( -\frac{R_n}{j+d+\delta_{nmj}} \right)}{j+d+\delta_{nmj}}, \quad (2.11)$$

onde  $\delta_{nm^j}$  é o delta de Kronecker, valendo 1 se  $n = m^j$  ou 0 caso contrário. Esta equação foi obtida em [13] considerando a transformada inversa de Fourier da função característica de  $R_n$ .

Para o cálculo de  $p(\mathbf{R}|m^j)$  procedemos de forma semelhante à equação (2.10), mas considerando um subconjunto de  $\Gamma(\mathbf{c}|m^j = m^{j*})$  que contém todos os  $\mathbf{c}$  que podem ser obtidos dado um valor de  $m^j = m^{j*}$ .

### 2.1.1 Cálculo eficiente de $p(\mathbf{R})$

A forma como  $p(\mathbf{R})$  foi escrita em (2.10) não é apropriada para o cálculo computacional pois o número de elementos em  $\Gamma(\mathbf{c})$  é próximo a  $N^T$ . Reescrevendo (2.10), obtemos:

$$\begin{aligned} p(\mathbf{R}) &= \sum_{\mathbf{c} \in \Gamma(\mathbf{c})} P(\mathbf{c}) \prod_{n=1}^N p(R_n|c_n) \\ &= \sum_{c_1=0}^T \left\{ P(c_1)p(R_1|c_1) \sum_{c_2=0}^T \{ P_2(c_2|c_1)p(R_2|c_2) \right. \\ &\quad \times \left. \left\{ \cdots \sum_{c_N=0}^T \left\{ P_N \left( c_N \mid \sum_{i=1}^{N-1} c_i \right) p(R_N|c_N) \right\} \cdots \right\} \right\}. \end{aligned} \quad (2.12)$$

Esta equação pode ser calculada em  $N$  passos, calculando a somatória mais interna e armazenando os seus  $N + 1$  valores, de acordo com as seguintes equações:

$$\begin{aligned} c'_i &= \sum_{i'=0}^{i-1} c_{i'}, \\ P_i(c_i|c'_i) &= \binom{T-c'_i}{c_i} \frac{1}{N-i} \left( 1 - \frac{1}{N-i} \right)^{T-c'_i-c_i}, \\ f_i(c'_i, \mathbf{R}) &= \sum_{c_i=0}^{T-c'_i} P(c_i|c'_i) p(R_i|c_i) f_{i+1}(c_i + c'_i, \mathbf{R}), \\ f_{N-1}(c'_N, \mathbf{R}) &= \sum_{c_N=0}^{T-c'_N} P(c_N|c'_N) p(R_N|c_N), \\ P(\mathbf{R}) &= f_0(0, \mathbf{R}). \end{aligned} \quad (2.13)$$

A função  $f_i(\cdot)$  é definida para  $i = 0, \dots, N-1$ . O argumento passado são os valores de  $R_i, \dots, R_N$ , e somente  $R_i$  é utilizado diretamente. Os valores de  $P_i$  precisam ser calculados somente uma vez. Os cálculos começam com  $i = N-1$ . Há  $N$  somatórios com no máximo  $T$  termos cada.

De forma semelhante podemos calcular eficientemente  $p(\mathbf{R}|m^j)$ . O cálculo é feito considerando as equações anteriores em conjunto com as seguintes, que substituem algumas das equações acima:

$$\begin{aligned}
P_i(c_i|c'_i) &= \binom{T - c'_i - 1}{c_i} \frac{1}{N - i} \left(1 - \frac{1}{N - i}\right)^{T - c'_i - c_i}, \\
f_{m^j}(c'_i, \mathbf{R}) &= \sum_{c_i=0}^{T - c'_i} P(c_i|c'_i) p(R_i|c_i + 1) f_{i+1}(c_i + c'_i, \mathbf{R}), \\
P(\mathbf{R}|m^j) &= f_0(0, \mathbf{R}),
\end{aligned} \tag{2.14}$$

onde alteramos uma das funções  $f_i(\cdot)$  para incluir a influência do  $j$ -ésimo usuário cuja mensagem é conhecida.

## 2.2 Capacidade soma

A informação mútua entre duas variáveis diz quanto o conhecimento de uma delas nos informa sobre a outra. A capacidade é o maior valor possível de informação mútua entre duas variáveis aleatórias. Ela é um limite superior sobre o quanto o conhecimento de uma variável informa sobre a outra. Para o sistema em questão, as variáveis consideradas para este cálculo dependem do método de detecção, que pode ser feito para cada um dos usuários separadamente ou para todos os usuários conjuntamente. Em ambos os casos, a variável observada é o vetor recebido  $\mathbf{R}$ . No caso de detecção de um único usuário (SUD), deseja-se saber, a partir de  $\mathbf{R}$ , o valor da mensagem  $m^j$  que um usuário transmitiu. No caso da detecção conjunta (MUD), deseja-se saber o valor de  $\mathbf{m}$ , o conjunto das mensagens transmitidos por todos os usuários. O método de detecção altera a informação mútua a ser calculada e conseqüentemente a capacidade soma, o limite sobre a soma das taxas de todos os usuários. As diferenças no cálculo da capacidade devido ao método de detecção estão indicados na Tab. 2.1 e detalhados nas seções seguintes.

Tab. 2.1: Diferenças no cálculo da capacidade devido ao método de detecção

Método de detecção	SUD	MUD
Vetor recebido	$\mathbf{R}$	$\mathbf{R}$
Mensagem desejada	$m^j$	$\mathbf{m}$
Informação mútua calculável	$\mathcal{I}(\mathbf{R}, m^j)$	$\mathcal{I}(\mathbf{R}, \mathbf{m})$
Entropias a serem calculadas	$\mathcal{H}\{\mathbf{R}\}, \mathcal{H}\{\mathbf{R} m^j\}$	$\mathcal{H}\{\mathbf{R}\}, \mathcal{H}\{\mathbf{R} \mathbf{m}\}$
Limite sobre a soma das taxas	$T \cdot \max[\mathcal{I}(\mathbf{R}, m^j)]$	$\max[\mathcal{I}(\mathbf{R}, \mathbf{m})]$
<i>Pdf</i> 's necessárias	$p(\mathbf{R}), p(\mathbf{R} m^j)$	$p(\mathbf{R}), p(\mathbf{R} \mathbf{m})$

### 2.2.1 Detecção de um único usuário

Em [13], detecção por máxima verossimilhança foi feita para cada mensagem  $m^j$ . Isto é, o receptor detecta um único usuário (SUD) em vez de realizar a detecção conjunta de todas as mensagens  $\{m^1, m^2, \dots, m^T\}$ . A taxa máxima na qual qualquer usuário pode transmitir informações corretamente é  $\max \mathcal{I}(m^j, \mathbf{R})$ , que é a mesma para todos os usuários. Esta taxa é dada por:

$$C_{SUD} = \max \mathcal{I}(\mathbf{R}, m^j) = \max [\mathcal{H}\{\mathbf{R}\} - \mathcal{H}\{\mathbf{R}|m^j\}], \quad (2.15)$$

onde a maximização é feita sobre a distribuição de probabilidade de  $m^j$ . A capacidade soma é então dada por  $C_{SUD}^{SOMA} = T \cdot C_{SUD}$ .

Para calcular a informação mútua, precisamos calcular  $\mathcal{H}\{\mathbf{R}\}$  e  $\mathcal{H}\{\mathbf{R}|m^j\}$ . Não encontramos uma forma analítica para estes valores. Utilizamos integrais de Monte Carlo e limitantes para obtê-los.

#### Cálculo por integrais de Monte Carlo

O cálculo de  $\mathcal{H}\{\mathbf{R}\}$  pode ser feito utilizando integração de Monte Carlo onde se obtém numericamente o valor de  $\mathcal{E}\{-\log [p(\mathbf{R})]\}$ . Isto é feito gerando aleatoriamente vários valores de  $\mathbf{R}$  (de acordo com a sua distribuição estatística) e avaliar  $p(\mathbf{R})$  para cada valor gerado. Utilizando as formas apresentadas na Seção 2.1.1, isto pode ser feito tanto para obter os valores de  $\mathcal{H}\{\mathbf{R}\}$  como os valores de  $\mathcal{H}\{\mathbf{R}|m^j\}$ .

#### Limitante inferior

Em [25], os valores de  $R_n$ , dado  $m^j$ , foram considerados estatisticamente independentes. Esta suposição simplificou o cálculo das probabilidades de erro ao mesmo tempo em que forneceu uma boa aproximação para estes valores. Sob esta suposição, podemos obter um limitante superior para  $\mathcal{H}\{\mathbf{R}|m^j\}$  (um limitante inferior para  $C_{SUD}$ ) avaliando numericamente as seguintes integrais:

$$\begin{aligned} \mathcal{H}\{\mathbf{R}|m^j\} \leq & \int_0^\infty p(R_n|m^j = n) \log_2 [p(R_n|m^j = n)] dR_n \\ & + (N-1) \cdot \int_0^\infty p(R_n|m^j \neq n) \log_2 [p(R_n|m^j \neq n)] dR_n. \end{aligned} \quad (2.16)$$

#### Distribuição de entrada que maximiza $C_{SUD}$

A capacidade é o máximo valor que a informação mútua entre  $m^j$  e  $\mathbf{R}$  pode fornecer. A única variável que podemos manipular para tentar maximizar este valor é a distribuição dos símbolos de entrada, já que a distribuição de  $\mathbf{R}$  depende exclusivamente de qual mensagem foi transmitida e das características do canal (que não podemos alterar).

Seja  $I = \mathcal{I}(m^j, \mathbf{R})$  a informação mútua entre os símbolos de entrada  $m^j$  e a saída  $\mathbf{R}$ . Ela pode ser escrita como:

$$I = \sum_{m^j=0}^{N-1} P(m^j) \int_{\mathbf{R}} p(\mathbf{R}|m^j) \log_2 \left[ \frac{p(\mathbf{R}|m^j)}{\sum_{m^{j'}=0}^{N-1} P(m^{j'}) p(\mathbf{R}|m^{j'})} \right] d\mathbf{R}, \quad (2.17)$$

onde  $P(m^j = n) = \zeta_n$ .

É claro que  $I$  é uma função de  $\zeta_0, \zeta_1, \dots, \zeta_{N-1}$ . Sejam  $f_i(\zeta_0, \zeta_1, \dots, \zeta_{N-1})$  as derivadas parciais  $\frac{\partial I}{\partial \zeta_i}$ . Podemos ver de (2.6) que uma permutação nos índices das frequências não altera as densidades de probabilidade das transições do canal,  $p(R_n|c_n)$ . Logo,  $f_i(\zeta_0, \zeta_1, \dots, \zeta_{N-1})$  tem a mesma forma para todo  $i$ , exceto por uma troca de índices.

Sendo a informação mútua uma função côncava em função da distribuição de entrada, a capacidade soma pode ser formulada como o seguinte problema de maximização:

$$L = \max_{\zeta_0, \zeta_1, \dots, \zeta_{N-1}} I(\zeta_0, \zeta_1, \dots, \zeta_{N-1}) - \lambda(\zeta_0 + \zeta_1 + \dots + \zeta_{N-1} - 1). \quad (2.18)$$

Aplicando as condições de Karush-Kuhn-Tucker, sob a hipótese de que a solução não está na fronteira do espaço da distribuição de entrada, obtemos:

$$\begin{aligned} \frac{\partial L}{\partial \zeta_0} &= f_0(\zeta_0, \zeta_1, \dots, \zeta_{N-1}) - \lambda, \\ \frac{\partial L}{\partial \zeta_1} &= f_1(\zeta_0, \zeta_1, \dots, \zeta_{N-1}) - \lambda, \\ &\vdots \\ \frac{\partial L}{\partial \zeta_{N-1}} &= f_{N-1}(\zeta_0, \zeta_1, \dots, \zeta_{N-1}) - \lambda. \end{aligned} \quad (2.19)$$

Para termos uma solução, deve haver um valor de  $\lambda$  que satisfaz simultaneamente todas as equações acima. Assim sendo, a solução deve satisfazer:

$$\lambda = f_0(\dots) = f_1(\dots) = \dots = f_{N-1}(\dots). \quad (2.20)$$

Dado que todos os argumentos são idênticos a um valor, os valores de  $f_0, f_1, \dots, f_{N-1}$  são os mesmos, pois estamos permutando argumentos com valores iguais. Com a restrição de que  $\sum_{i=0}^{N-1} \zeta_i = 1$ , a solução é  $\zeta_0 = \zeta_1 = \dots = \zeta_{N-1} = 1/N$ .

### 2.2.2 Detecção conjunta

Para a detecção conjunta, a capacidade soma pode ser definida como o maior valor da informação mútua entre o vetor recebido  $\mathbf{R}$  e o vetor que contém as mensagens de todos os usuários,  $\mathbf{m}$ :

$$C_{MUD} = \max \mathcal{I}(\mathbf{m}, \mathbf{R}) = \max [\mathcal{H}\{\mathbf{R}\} - \mathcal{H}\{\mathbf{R}|\mathbf{m}\}]. \quad (2.21)$$

Neste caso, não especificamos qual é a taxa máxima que cada usuário pode utilizar. Este valor é um limite para a soma de todas as possíveis combinações de taxas. Assim,  $C_{MUD}^{SOMA} = C_{MUD}$ .

Para manter o sistema simples, podemos considerar que todos os usuários transmitem com a mesma distribuição, que ainda não sabemos qual é. Neste caso, podemos obter uma forma analítica para  $\mathcal{H}\{\mathbf{R}|\mathbf{m}\}$ , em bits:

$$\mathcal{H}\{\mathbf{R}|\mathbf{m}\} = \frac{1}{\log 2} \sum_{n=1}^N \sum_{c_n=0}^T \Phi(c_n, T, \zeta_n) [1 + \log(c_n + d)]. \quad (2.22)$$

Podemos obter o valor de  $\mathcal{H}\{\mathbf{R}\}$  por integrais de Monte Carlo, como indicado na seção anterior, ou por um limitante superior.

#### Limitante superior

Um limitante superior simples para  $\mathcal{H}\{\mathbf{R}\}$  pode ser obtido considerando que os valores de  $R_i$  são estatisticamente independentes. Neste caso:

$$\mathcal{H}\{\mathbf{R}\} \leq -N \int_0^\infty P(R_n) \log_2 P(R_n) dR_n, \quad (2.23)$$

já que todos os  $R_i$  têm a mesma distribuição.

O limitante superior de  $\mathcal{H}\{\mathbf{R}\}$  pode ser combinado com o limitante superior de  $\mathcal{H}\{\mathbf{R}|m^j\}$  para obtermos uma aproximação para  $C_{SUD}$ .

#### Distribuição de entrada que maximiza $C_{MUD}$

Para o caso de detecção de um único usuário, nos baseamos no fato de que a informação mútua é uma função côncava em relação à distribuição de entrada dos símbolos. Considerando que todos os usuários utilizam a mesma distribuição, a função  $\mathcal{I}(\mathbf{m}, \mathbf{R})$  é côncava em relação à distribuição de  $\mathbf{m}$ , mas não necessariamente em relação à distribuição comum a todos os usuários. Para casos simples com  $N = 2$ , por exemplo, a informação mútua entre  $\mathbf{m}$  e  $\mathbf{R}$  não é côncava para  $T > 2$ . A complexidade do cálculo da capacidade (feita através de integração por Monte Carlo) torna proibitiva a aplicação de algoritmos de otimização para  $N > 3$ .

O limitante superior para a informação mútua, por outro lado, tem cálculo bem mais simples e fornece valores próximos aos obtidos pela integração de Monte Carlo. Podemos aplicar algoritmos de otimização para encontrarmos o seu valor máximo em função da distribuição dos símbolos de entrada do canal. Seja  $\mathbf{z}$  a distribuição que maximiza o limitante superior para a informação mútua, com termos  $z_i$  ( $i = 0, 1, \dots, N - 1$ ) que indicam a probabilidade de cada um dos usuários utilizarem a  $i$ -ésima frequência. Se os usuários utilizassem todas as frequências com a mesma probabilidade, todos os valores de  $z_i$  seriam iguais a  $1/N$ , o que caracterizaria uma distribuição uniforme. Definimos a distância quadrática entre  $\mathbf{z}$  e a distribuição uniforme como:

$$dist_{\mathbf{z}} = \sum_{i=0}^{N-1} \left( z_i - \frac{1}{N} \right)^2. \quad (2.24)$$

Esta distância só pode valer zero se a distribuição  $\mathbf{z}$  for igual à distribuição uniforme. Apresentamos na Fig. 2.2 o valor de  $dist_{\mathbf{z}}$  para  $E_b/N_0 = 20dB$  e alguns valores de  $T$  e  $N$ . Podemos ver que, para a maioria dos pontos onde  $T \leq N$ , a distribuição uniforme é a distribuição que otimiza o limitante para a informação mútua. O limitante superior para a informação mútua também possui um máximo em função do número de usuários, que ocorre para uma quantidade de usuários  $T_{MAX} < N$ . Para  $E_b/N_0 = 20dB$  e  $N = 8, 16$  e  $32$ , o valor de  $T_{MAX}$  é respectivamente  $T = 6, 12$  e  $24$ , o que reforça a escolha da distribuição uniforme para os símbolos de entrada do canal. Estas duas características sugerem que este sistema deve operar com  $T < N$  e, conseqüentemente, que cada usuário utilize a distribuição uniforme nos símbolos de entrada para que se atinja a capacidade ou um valor muito próximo, que chamaremos de capacidade efetiva.

Para o caso não ruidoso [20], a distribuição que atinge a capacidade é a distribuição uniforme. Para o caso ruidoso, esta mesma distribuição maximiza o limitante superior para a capacidade quando  $T < N$ . O MUD realiza maximização *a posteriori*, escolhendo  $\mathbf{m}$  que maximize a densidade de probabilidade condicional  $p(\mathbf{R}|\mathbf{m})$ . Ao usarmos a distribuição uniforme, maximizar  $p(\mathbf{R}|\mathbf{m})$  equivale a maximizar  $p(\mathbf{R}, \mathbf{m})$ , a densidade de probabilidade conjunta.

### 2.2.3 Caso não ruidoso

Para referência, temos o valor da capacidade para o canal não ruidoso, apresentada em [20]. Neste caso, a informação mútua se reduz a  $\mathcal{H}\{\mathbf{R}\}$ , pois  $\mathcal{H}\{\mathbf{R}|\mathbf{m}\} = 0$ . A expressão de  $\mathcal{H}\{\mathbf{R}\}$  é:

$$C = \sum_{\mathbf{c} \in \Gamma(\mathbf{c})} \frac{T!}{c_0!c_1!\dots c_{N-1}!} \frac{1}{N^T} \log_2 \left[ \frac{N^T}{T!/c_0!c_1!\dots c_{N-1}!} \right]. \quad (2.25)$$

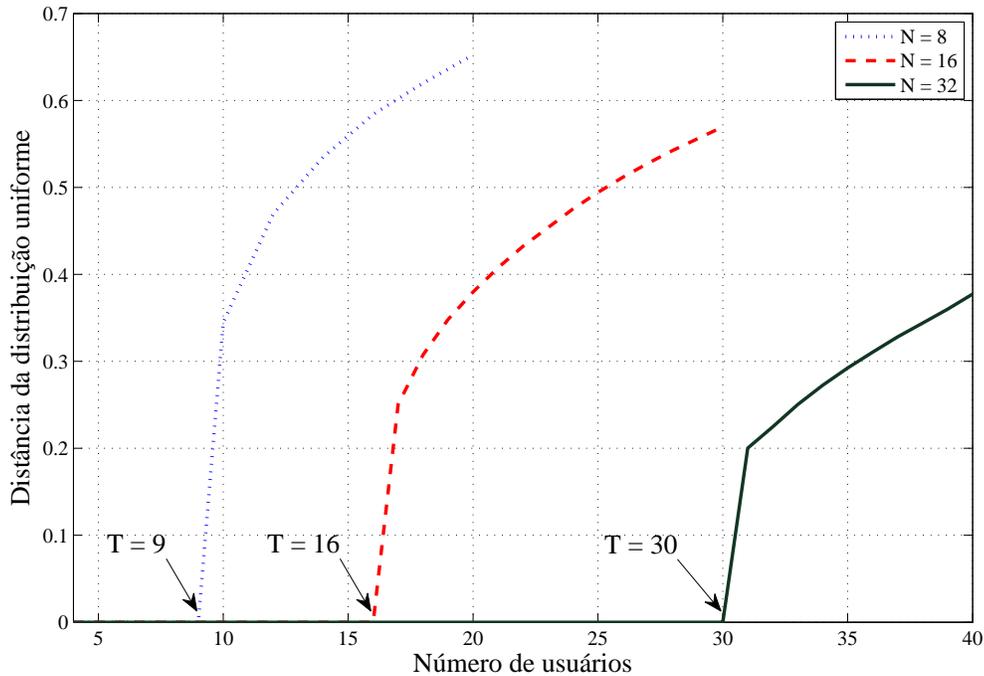


Fig. 2.2: Distâncias entre a distribuição que otimiza a informação mútua e a distribuição uniforme.

## 2.3 Resultados de capacidade

Resultados para a capacidade soma são apresentados nas Figs. 2.3 a 2.5 para  $N = 4, 8$  e  $16$  frequências. Para obter linhas suaves, em torno de  $10^6$  vetores aleatórios foram utilizados por ponto nas integrais por Monte Carlo. As linhas tracejadas indicam o limite superior para  $C_{MUD}^{SOMA}$  e a aproximação para  $C_{SUD}^{SOMA}$  das curvas com o mesmo marcador. Também apresentamos, como parâmetro de comparação, a capacidade soma para o caso não ruidoso.

O limitante superior para a capacidade soma está muito próximo aos resultados encontrados através de integrais de Monte Carlo. Este limitante foi calculado usando a distribuição que o maximiza. Mesmo que alguma outra distribuição resulte num valor maior para a capacidade soma nestas situações, esta ainda seria menor ou igual ao limitante. Sendo estes valores próximos, concluímos que utilizar a distribuição uniforme é uma boa alternativa. Os pequenos ganhos possíveis não justificam a utilização de uma distribuição não uniforme, que adiciona complexidade ao sistema.

O limitante superior para  $C_{MUD}^{SOMA}$  fica mais próximo com um aumento em  $N$ ,  $T$  ou  $d$ . Isto parece natural já que, com um aumento em  $d$  (diminuição em  $E_t/N_0$ ), os valores recebidos estão mais relacionados ao ruído do que aos sinais transmitidos pelos usuários. Para baixos valores de  $T$  e  $d$ , a capacidade efetiva é próxima ao caso não ruidoso. Também podemos observar que há um número ótimo de usuários que maximiza a capacidade efetiva, um resultado semelhante ao canal A em [20].

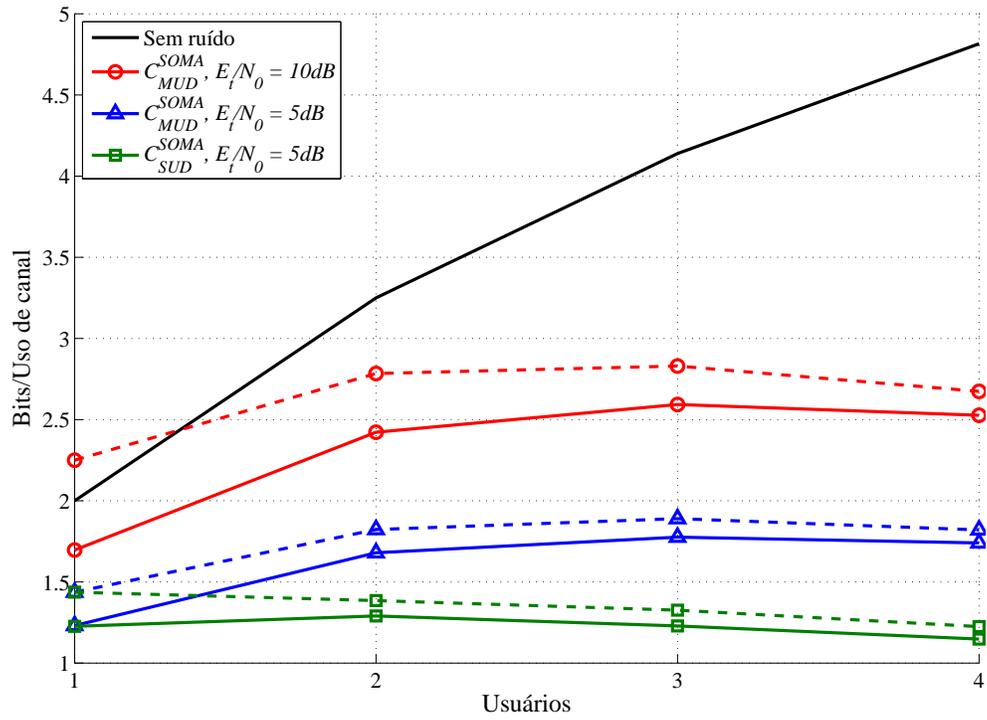


Fig. 2.3: Capacidade soma para  $N = 4$ ,  $E_t/N_0 = 5dB$  e  $10dB$

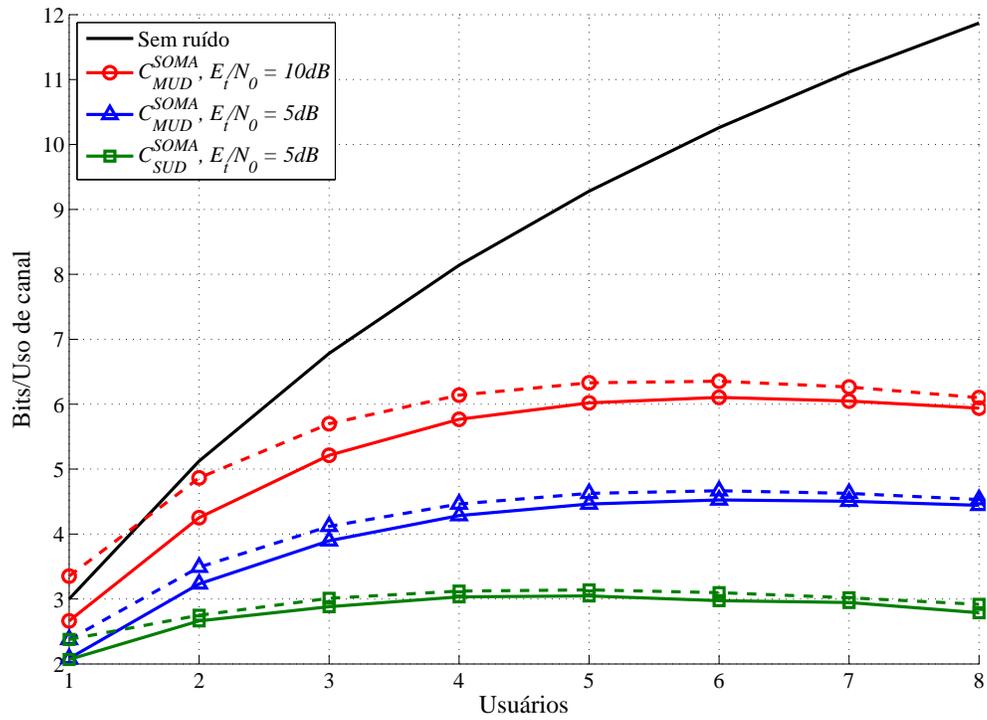


Fig. 2.4: Capacidade soma para  $N = 8$ ,  $E_t/N_0 = 5dB$  e  $10dB$

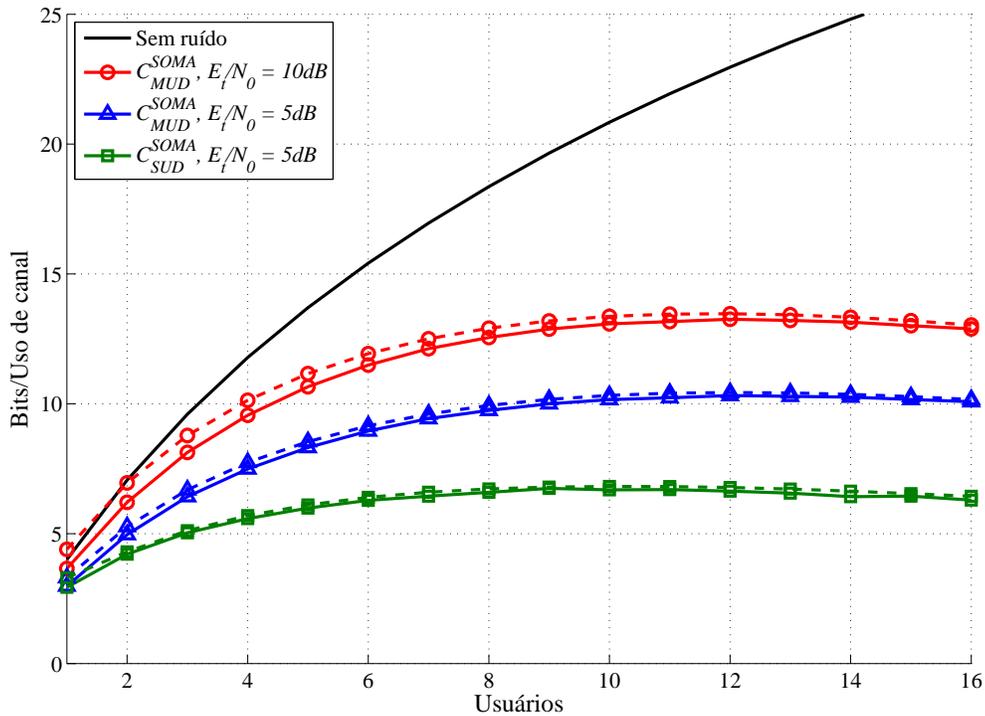


Fig. 2.5: Capacidade soma para  $N = 16$ ,  $E_t/N_0 = 5dB$  e  $10dB$

O valor máximo ocorre próximo ao máximo do limitante superior, servindo este de referência.

A aproximação utilizada para a capacidade soma da detecção individual  $C_{SUD}^{SOMA}$  está próximo aos valores encontrados através das integrais por Monte Carlo. Assim, para o caso MAC, a aproximação fica melhor com um aumento em  $T$ .

Mostramos na Fig. 2.6 o valor da capacidade soma para  $N = 4, 8$  e  $16$  e para  $T = 3, 6$  e  $12$ , respectivamente, para a detecção conjunta (linhas cheias) e individual (linhas tracejadas). Podemos observar que há taxas que são possíveis com detecção conjunta que não são possíveis com detecção individual, independentemente da relação sinal ruído utilizada. Isto fica evidente por exemplo para o caso onde  $N = 16$ : a capacidade soma para detecção conjunta é de  $14$  bits para um valor de  $E_t/N_0$  em torno de  $12$  dB. Esta taxa não seria possível com detecção individual.

### 2.3.1 Eficiência espectral

Tendo em vista a proximidade do limitante superior com o valor calculado pelo método de Monte Carlo, e também o fato de que a capacidade atinge um valor máximo em função do número de usuários, podemos rapidamente calcular através do limitante uma boa estimativa da eficiência espectral máxima para vários valores de  $N$  e  $E_t/N_0$ . Considerando que os sinais transmitidos tem duração  $\tau$

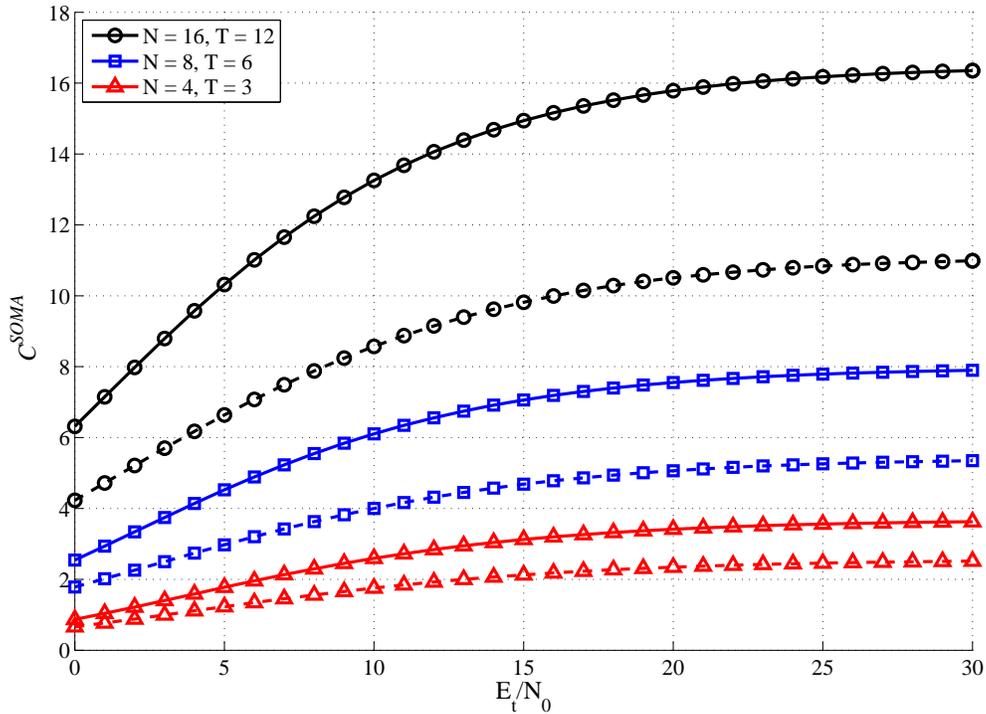


Fig. 2.6: Vários valores de capacidade em função da relação sinal-ruído.

e que cada um dos  $N$  subcanais tem banda  $1/\tau$ , o sistema inteiro utiliza uma banda (em Hertz) de  $W = N/\tau$ . A eficiência espectral é então dada por:

$$\frac{C^{SOMA}}{W} = \frac{C^{SOMA}}{N}. \quad (2.26)$$

Apresentamos estes resultados na Fig. 2.7. A partir de  $E_t/N_0 = 100dB$  as curvas praticamente não variam. Há um limite global para a eficiência espectral em torno de  $1 \text{ bit/uso de canal}/N$ .

O formato destas curvas fornece algumas sugestões para o projeto de um sistema MFSK multi-usuário. Dada uma faixa do espectro com banda  $W$  com desvanecimento seletivo em frequência, é preciso dividi-la em pelo menos  $Q$  subcanais com banda de no máximo  $\delta w$ , onde  $\delta w$  é a banda de coerência do canal. Desta forma, podemos tratar cada um dos  $Q$  canais como tendo desvanecimento plano e independente dos outros canais.

Embora os valores da eficiência espectral sejam baixos, a utilização de altos valores de  $W$  permitiria a transmissão com altas taxas. Como podemos ver da Fig. 2.7, em alguns casos (como para  $E_t/N_0 = 30dB$ ) há pouca variação entre um sistema com  $N = 16$  frequências e um sistema com  $N = 2048$  frequências. Entretanto, o sistema com 2048 frequências com certeza será mais complexo. Se tivermos  $Q = 2048$ , por exemplo, podemos atingir eficiências semelhantes utilizando um único

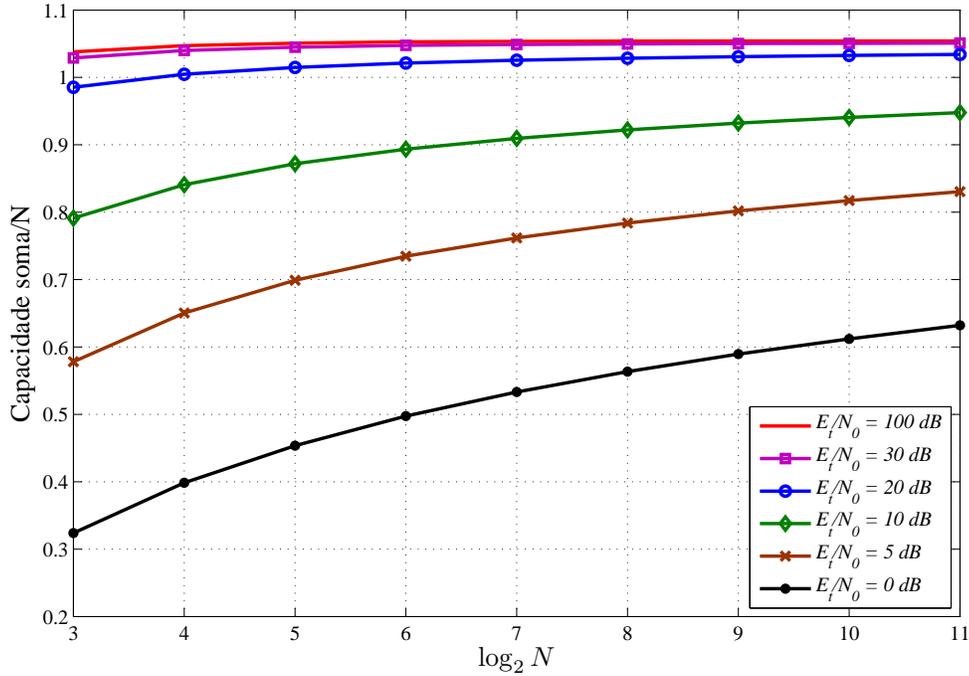


Fig. 2.7: Eficiência espectral para várias combinações de  $N$ ,  $T$  e  $E_t/N_0$

sistema com  $N = 2048$  ou utilizando em paralelo 128 sistemas com  $N = 16$ . A vantagem de usar  $N = 16$  é que a complexidade de cada um dos sistemas é menor e cada um deles pode ser implementado separadamente. Há propostas para sistemas com banda ultra-larga baseados nesta idéia [26]. Estas vantagens práticas estimulam a pesquisa por sistemas com baixos valores de  $N$ .

Um valor que usaremos para comparar taxas entre sistemas com diferentes valores de  $N$  e  $T$  é a capacidade normalizada, indicada pela letra,  $\bar{C}$ . Ela é definida pela seguinte equação:

$$\bar{C} = \frac{C^{SOMA}}{T \cdot \log_2 N}. \quad (2.27)$$

Se somente um usuário estivesse transmitindo sem a presença de ruído, o valor de  $C$  seria igual a 1, já que a capacidade soma seria igual a  $\log_2 N$ . O valor da capacidade normalizada pode ser interpretado como a maior taxa que os códigos dos usuários podem ter.

Para o caso de detecção conjunta, o valor da capacidade normalizada vale:

$$\overline{C_{MUD}} = \frac{C_{MUD}}{T \cdot k}, \quad (2.28)$$

enquanto que para o caso de detecção individual vale:

$$\overline{C_{SUD}} = \frac{C_{SUD}}{k}. \quad (2.29)$$

### 2.3.2 Limite para taxas individuais para detecção sucessiva

A capacidade soma só nos informa um limite superior para a soma das taxas de todos os usuários, isto é,  $C^{SOMA} \geq Tx_1 + Tx_2 + \dots + Tx_T$ , onde  $Tx_i$  é a taxa de transmissão do  $i$ -ésimo usuário em bits/uso de canal. Como estas taxas estão distribuídas entre os usuários é um outro problema. Uma forma de se obter estes valores individuais é através da detecção sucessiva dos usuários. Isto é: tenta-se detectar o primeiro usuário. Como ele está transmitindo com uma taxa menor ou igual a sua capacidade (que neste caso é o valor dado na Seção 2.2.1), conseguimos identificar corretamente a sua mensagem transmitida. Com o conhecimento desta mensagem, detectamos o segundo usuário, que tem restrições semelhantes ao primeiro usuário. Entretanto, a capacidade do segundo usuário é dada pelo máximo (em função da distribuição de entrada) da informação mútua  $\mathcal{I}(\mathbf{R}, m^2|m^1)$ , isto é, a informação mútua entre o vetor recebido  $\mathbf{R}$  e a mensagem do segundo usuário  $m^2$ , dado o conhecimento da mensagem  $m^1$  transmitida pelo primeiro usuário. Este procedimento se repete até o último usuário, cuja taxa deve ser menor do que o valor máximo de  $\mathcal{I}(\mathbf{R}, m^T|m^1, m^2, \dots, m^{T-1})$ .

A soma das taxas é exatamente igual à capacidade soma, evidenciado quando somamos as entropias envolvidas para o cálculo das taxas individuais:

$$\begin{aligned}
 Tx_1 + Tx_2 + \dots + Tx_T &= (\mathcal{I}(\mathbf{R}, m^1) + (\mathcal{I}(\mathbf{R}, m^2|m^1)) + \dots + (\mathcal{I}(\mathbf{R}, m^T|m^1, m^2, \dots, m^{T-1}))) \\
 &= (\mathcal{H}\{\mathbf{R}\} - \mathcal{H}\{\mathbf{R}|m^1\}) + (\mathcal{H}\{\mathbf{R}|m^1\} - \mathcal{H}\{\mathbf{R}|m^1, m^2\}) \\
 &\quad + \dots + (\mathcal{H}\{\mathbf{R}|m^1, m^2, \dots, m^{T-1}\} - \mathcal{H}\{\mathbf{R}|m^1, m^2, \dots, m^T\}) \\
 &= \mathcal{H}\{\mathbf{R}\} - \mathcal{H}\{\mathbf{R}|m^1, m^2, \dots, m^T\} = \mathcal{I}(\mathbf{R}, \mathbf{m})
 \end{aligned} \tag{2.30}$$

Os termos intermediários  $\mathcal{H}\{\mathbf{R}|m^1, m^2, \dots, m^t\}$ , com  $t \leq T$ , podem ser obtidos com o mesmo método utilizado para se obter  $\mathcal{H}\{\mathbf{R}\}$ , modificando-se (2.13) para que se calcule a probabilidade correspondente  $p(R|m^1, m^2, \dots, m^t)$ .

Apresentamos o resultado do cálculo das taxas individuais nas Figs. 2.8 a 2.10. O comportamento destas curvas é esperado, tendo em vista que o primeiro usuário a ser detectado tem  $T - 1$  interferentes desconhecidos e o último tem conhecimento da mensagem dos outros usuários. A capacidade soma para detecção de um único usuário é dada por  $T$  vezes a capacidade do primeiro usuário a ser detectado. A soma das diferenças entre a capacidade do primeiro usuário e as capacidades dos usuários seguintes é a diferença entre as capacidades soma SUD e MUD.

A detecção sucessiva não é o cancelamento de interferência. No cancelamento de interferência, o sinal correspondente à interferência (usuários já detectados) é estimado e subtraído do sinal recebido, limpando-o para o processamento do próximo usuário a ser detectado. Para realizar tal cancelamento seria necessário estimar o valor de  $\alpha_n^j$  para todos os usuários, o que este sistema não faz.

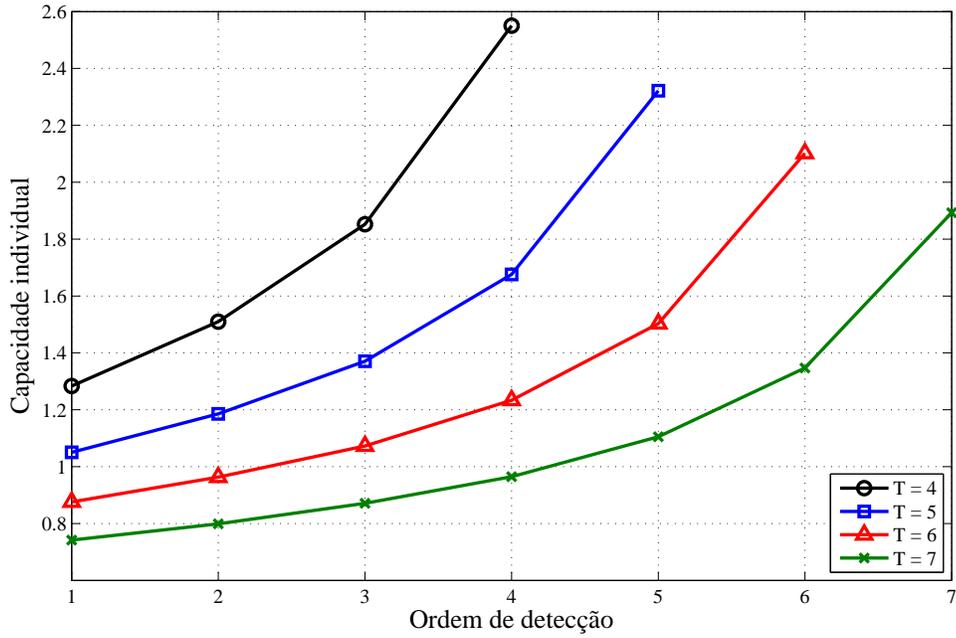


Fig. 2.8: Limites individuais para detecção sequencial para  $N=8$ ,  $E_b/N_0 = 25dB$ .

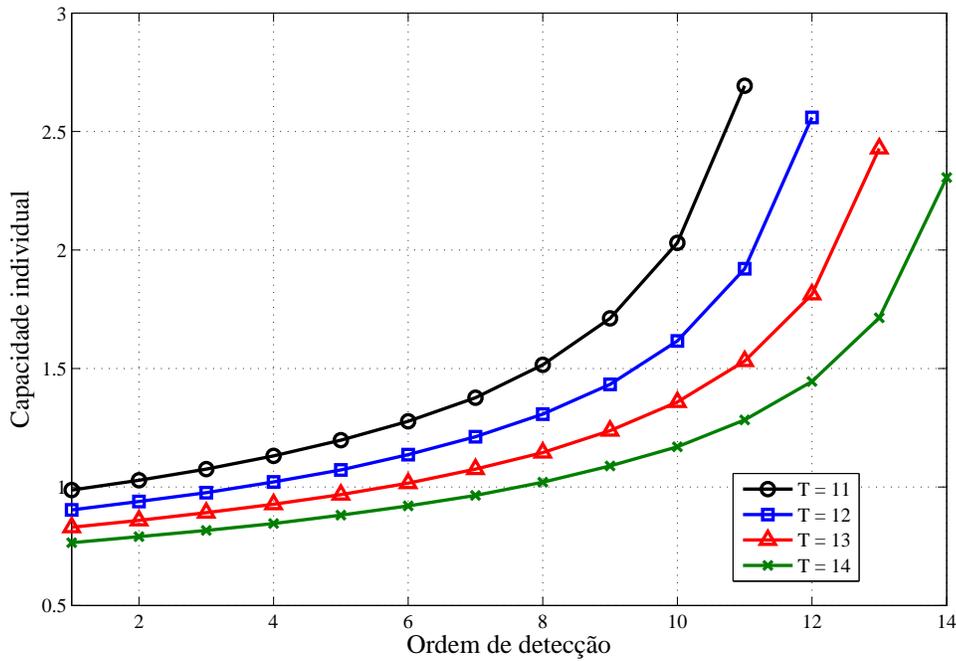


Fig. 2.9: Limites individuais para detecção sequencial para  $N=16$ ,  $E_b/N_0 = 25dB$ .

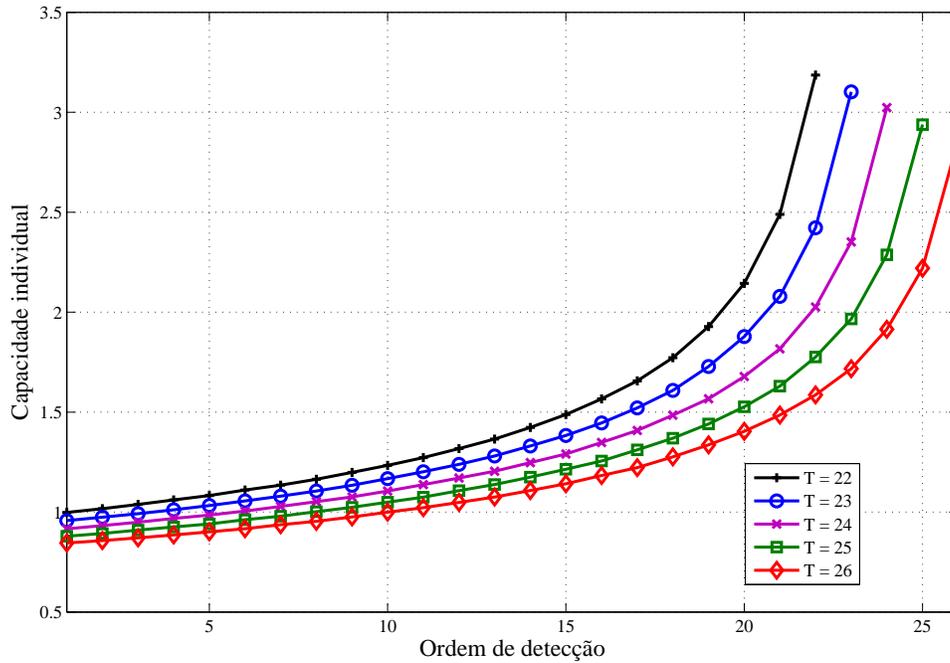


Fig. 2.10: Limites individuais para detecção sequencial para  $N=32$ ,  $E_b/N_0 = 25dB$ .

## 2.4 Resumo

Neste capítulo apresentamos alguns resultados de capacidade para canais de acesso múltiplo com desvanecimento Rayleigh e AWGN. O canal foi modelado como uma concatenação de um MAC não ruidoso com um canal ruidoso. Para obter estes resultados foi necessário derivar as densidades de probabilidade que descrevem o canal ruidoso.

Obtivemos como resultado que a distribuição uniforme de entrada atinge a capacidade para detecção individual. Para detecção conjunta, a distribuição uniforme fornece um valor para a informação mútua que, se não é a capacidade, está muito próxima desta. Mostramos que um limitante superior simples está próximo dos valores obtidos numericamente. Um limitante inferior simples para  $T \cdot C_{SUD}$  também foi obtido. Através do limitante para  $C_{MUD}$  percebemos que, dependendo da relação sinal-ruído, a eficiência espectral não aumenta de forma significativa com o aumento de  $N$ . Desta forma, um único sistema com muitas frequências pode ser substituído por vários sistemas mais simples sem grande perda de eficiência espectral.

Obtivemos, através de uma variação das equações desenvolvidas para o cálculo da entropia do sinal recebido, a capacidade individual de cada usuário para o caso de detecção sucessiva. A ordem de detecção afeta bastante a taxa permissível para cada usuário, podendo o último usuário a ser detectado ter um limite de transmissão muito maior do que o primeiro usuário.

## Capítulo 3

# Modelo e Análise EXIT do Detector Multiusuário

Neste capítulo iniciaremos a análise EXIT de um sistema composto por detector e códigos. Vamos primeiramente estudar o detector. A ferramenta que vamos utilizar será as curvas EXIT (do inglês *Extrinsic Information Transfer*), ou curvas de transferência de informação extrínseca. Esta ferramenta consiste em separar um sistema iterativo em blocos mais simples e analisá-los separadamente. Um dos blocos será o detector. O outro bloco será composto por vários blocos menores que são os decodificadores de cada usuário. Os blocos trocam mensagens entre si que representam a probabilidade dos bits transmitidos serem iguais a zero ou um. Para gerar uma curva EXIT, precisamos determinar o quanto de informação sobre os bits transmitidos o detector pode nos fornecer, dada uma realização do canal e um conjunto de mensagens vindas dos códigos. Estas mensagens fornecem um valor de informação *a priori* sobre os bits transmitidos. Uma hipótese simplificadora permite que estas mensagens sejam geradas aleatoriamente através de uma distribuição Gaussiana com parâmetros definidos exclusivamente pelo valor de informação *a priori* desejado. Isto nos permite gerar as curvas EXIT do detector sem necessidade de considerar os outros blocos.

De posse das mensagens *a priori* e dos valores de saída do canal, o detector deve gerar mensagens em resposta aos códigos. Estas mensagens são probabilidades marginais dos bits transmitidos. O cálculo destas mensagens não é simples dada a equação que define a densidade de probabilidade conjunta dos bits e valores dos canais. Uma forma de calcular estes valores eficientemente é através da aplicação do algoritmo soma-produto sobre o grafo de fatores que representa a densidade de probabilidade.

Para gerar o grafo, a densidade de probabilidade conjunta é quebrada em fatores menores. Cada um destes fatores tem uma representação simples. A combinação destas representações gera o grafo que representa a função do detector. A aplicação do algoritmo soma-produto se dá pela passagem de

mensagens através dos ramos do grafo. Devido ao formato do grafo, o algoritmo não tem terminação natural, mas é em si um processo iterativo. Analisaremos o comportamento do grafo em função do número de iterações. Em seguida, mostramos alguns resultados obtidos.

Para familiarizar o leitor com alguns dos conceitos, apresentamos nas próximas duas seções uma breve revisão sobre curvas EXIT e grafos de fatores.

### 3.1 Revisão sobre curvas EXIT

Curvas EXIT [7; 27] são uma ferramenta para analisar o comportamento iterativo de um sistema. No problema estudado, o objetivo do sistema é determinar quais são os bits de informação de uma mensagem que foi transmitida, dada uma observação do canal. O método consiste em separar um sistema iterativo em poucos blocos (tradicionalmente dois) que trocam mensagens entre si. Obtém-se, para cada bloco, curvas que relacionam o valor da informação mútua entre um conjunto de variáveis e as mensagens recebidas (informação *a priori*) e transmitidas (informação extrínseca). O conjunto de variáveis é usualmente um conjunto de bits, que podem ser bits de informação, bits transmitidos ou bits passados de um bloco para outro durante o processo de geração dos bits transmitidos. As mensagens são usualmente as probabilidades dos bits valerem 1 ou 0. As mensagens transmitidas são denominadas extrínsecas porque a princípio, para um dado bit, a mensagem transmitida não depende da mensagem recebida.

No caso mais genérico e simples temos dois blocos como mostrado na Fig. 3.1, que trocam mensagens entre si. Há um conjunto de variáveis que no caso estudado são bits. Dadas as densidades de probabilidade conjunta das mensagens  $\eta_{AB}$  ou  $\eta_{BA}$  e das variáveis, é possível obter o valor da informação mútua entre as mensagens e as variáveis. Este valor vale  $I_{AB}$  para as mensagens que passam do bloco A para o bloco B e  $I_{BA}$  para as mensagens que passam do bloco B para o bloco A. Pode haver um conjunto de mensagens externas vindas de um canal  $\Theta$ , que fornecem um valor de informação mútua sobre um conjunto de variáveis (não necessariamente o mesmo conjunto de variáveis utilizado para se obter  $I_{AB}$  e  $I_{BA}$ ) igual a  $I_{\Theta}$  que pode estar ligado aos dois blocos ou a somente um.

Cada bloco realiza operações sobre as mensagens recebidas e gera novas mensagens em resposta. A distribuição das mensagens que ele gera depende da distribuição das mensagens fornecidas. Desta forma, temos duas curvas, uma para cada bloco, que relacionam o valor da informação *a priori* com um valor de informação extrínseca. A curva do bloco que recebe as informações do canal pode ser parametrizada pelo parâmetro que define a distribuição das mensagens vindas do canal. A informação extrínseca de um bloco se torna a informação *a priori* do outro bloco, possivelmente resultando num aumento no valor da informação extrínseca deste segundo bloco e assim por diante.

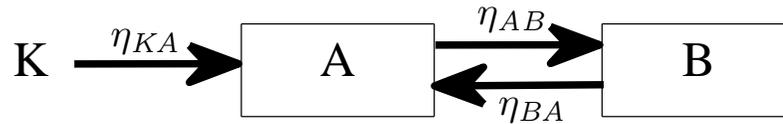


Fig. 3.1: Sistema iterativo simplificado.

Para se prever o comportamento deste sistema iterativo, as curvas de cada bloco podem ser desenhadas num mesmo plano, mas com os eixos apropriadamente relacionados como mostrado na Fig. 3.2. A informação extrínseca de um bloco torna-se informação *a priori* do outro bloco. A curva  $I_{AB}$  depende dos valores de  $I_{BA}$  e de um parâmetro do canal  $\Theta_1$  ou  $\Theta_2$  (por exemplo a relação sinal-ruído). O processo iterativo começa na esquerda, quando a informação *a priori* vale zero. Nesta situação, pelo menos um dos blocos deve ser capaz de fornecer um valor maior do que zero para a informação extrínseca. Este valor se torna a informação *a priori* do outro bloco, que por sua vez deve retornar um valor de informação extrínseca maior do que o utilizado pelo primeiro bloco. Os blocos se alternam no processamento das mensagens com o objetivo de aumentar o valor da informação mútua. Se as curvas se cruzam para algum par de valores de  $I_{AB}$  e  $I_{BA}$  como quando o parâmetro do canal vale  $\Theta_2$ , é de se esperar que o processo não consiga passar deste ponto. Por outro lado, se existe um “túnel” entre as duas curvas como quando o parâmetro do canal vale  $\Theta_1$ , é de se esperar que  $I_{BA}$  possa atingir, com a execução das iterações, o valor 1. Isto equivale a dizer que os bits de informação podem ser determinados corretamente, seja diretamente ou através dos bits aos quais as mensagens trocadas entre os blocos se referem. Como tanto  $I_{AB}$  como  $I_{BA}$  se referem aos mesmos bits, basta que um destes valha 1 para se determinar os bits.

No caso Turbo, os blocos são os códigos convolucionais componente, e o parâmetro fornecido a ambos os blocos são os valores do canal, como mostra a Fig. 3.3. No caso de concatenação serial de códigos, o código externo é o bloco B e o interno é o bloco A, sendo que só este recebe os valores do canal  $\Theta$ , como mostra a Fig. 3.4. As mensagens trocadas entre os blocos são probabilidades dos bits da sequência interna valerem 1 ou 0. O modelo de concatenação serial pode ser utilizado para representar códigos LDPC, LDGM e RA, como será mostrado a seguir.

Curvas EXIT foram inicialmente obtidas por simulação para códigos convolucionais utilizados em códigos Turbo [7] sobre canais Gaussianos. Posteriormente, foi possível obter estas curvas analiticamente para códigos LDPC e RA, dividindo cada um destes códigos em duas partes como detalhado em [28] e [29].

A vantagem de usar este método é que as curvas podem ser obtidas separadamente para cada

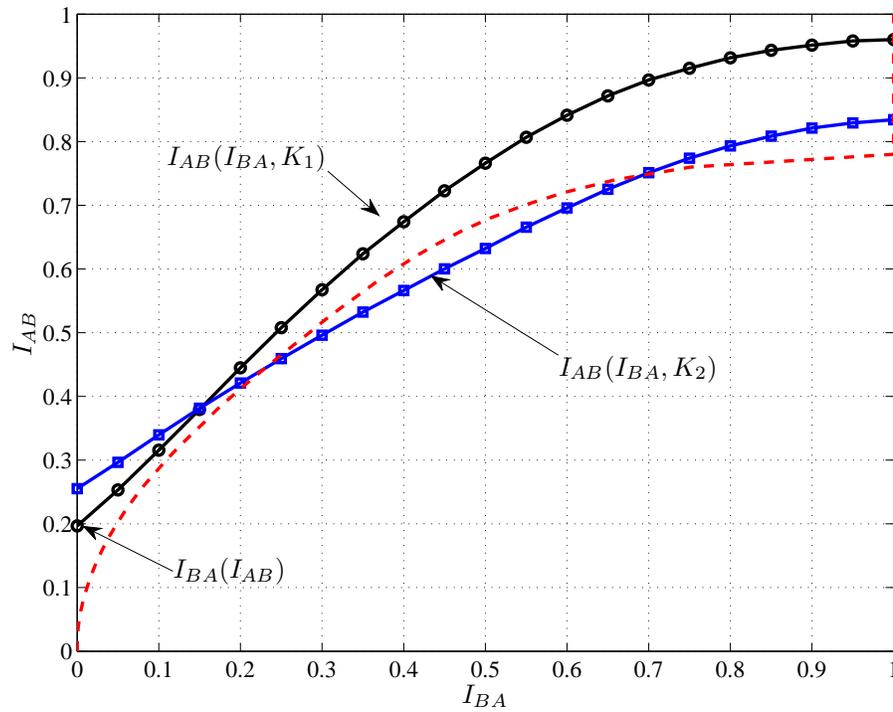


Fig. 3.2: Exemplo de análise EXIT

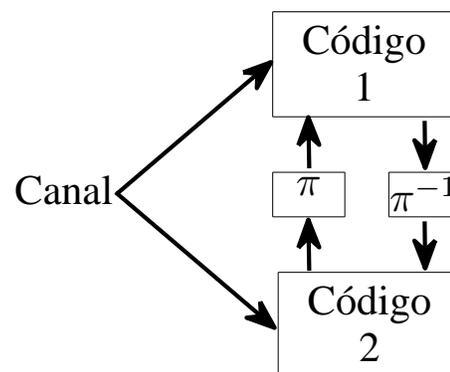


Fig. 3.3: Blocos de um decodificador Turbo (concatenação paralela).

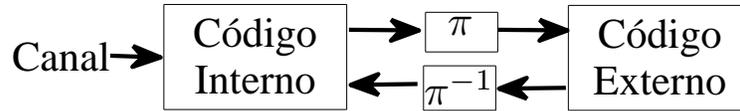


Fig. 3.4: Blocos de um decodificador com concatenação serial.

bloco, sem necessidade de se analisar de uma única vez o sistema inteiro, o que adicionaria complexidade à análise. O sistema estudado aqui pode ser dividido em dois blocos: detector multiusuário e códigos, que ainda serão definidos. As mensagens trocadas entre os blocos serão as probabilidades dos bits transmitidos valerem zero ou um. A curva do detector pode ser obtida através de simulação. A simulação exige realizações do canal e um conjunto de mensagens representando as probabilidades dos bits transmitidos. A realização do canal pode ser gerada utilizando as equações da Seção 2.1. As mensagens provenientes dos códigos devem ser geradas de modo a fornecerem um valor determinado de informação *a priori*, para que a curva EXIT possa ser gerada de forma controlada. A modelagem destas mensagens é tratada na próxima seção

### 3.1.1 Modelagem das mensagens recebidas

A modelagem das mensagens apresentado nesta seção foi descrita em [7] e [27].

No caso de códigos binários, as mensagens transmitidas são valores de Log-verossimilhança (LLR - do inglês: *Log-Likelihood Ratio*) [30]. Seja um bit  $b$ , que nesta seção pode assumir os valores  $-1$  ou  $1$  representando os valores lógicos 0 e 1. Suponha que este bit seja transmitido através de um canal com ruído aditivo Gaussiano branco. O sinal recebido  $z$  é então  $b + n$ , onde  $n$  corresponde ao ruído com distribuição Gaussiana com média zero e variância  $\sigma_n = N_0/2$  (densidade espectral bilateral). A variável  $z$  tem a seguinte *pdf*:

$$p(z|b) = \frac{\exp\left(-\frac{(z-b)^2}{2\sigma_n^2}\right)}{\sqrt{2\pi}\sigma_n}. \quad (3.1)$$

Podemos obter  $P(b|z)$  utilizando o teorema de Bayes:

$$P(b|z) = \frac{p(z|b)P(b)}{p(z)}. \quad (3.2)$$

O valor LLR de  $b$  é o logaritmo da razão entre  $P(b = 1)$  e  $P(b = -1)$ . Quando este valor está condicionado à variável  $z$ , obtemos:

$$\begin{aligned} L(b|z) &= L = \ln \frac{P(b = 1|z)}{P(b = -1|z)} \\ &= \ln \frac{P(b = 1)}{P(b = -1)} + \ln \frac{p(z|b = 1)}{p(z|b = -1)}. \end{aligned} \quad (3.3)$$

Pode-se usar qualquer base para o logaritmo. Por conveniência escolhe-se o logaritmo natural, pois isto torna mais simples as próximas equações. Considerando que os bits são equiprováveis, obtemos a seguinte simplificação ao aplicarmos (3.1) a (3.3):

$$L = \mu_L \cdot b + n_L, \quad (3.4)$$

onde  $\mu_L = 2/\sigma_n^2$  e  $n_L$  é uma variável Gaussiana com média nula e variância  $\sigma_L^2 = 4/\sigma_n^2$ .

Desta forma,  $L$  é uma variável Gaussiana com média e variância relacionados pela seguinte equação:

$$\mu_L = \frac{\sigma_L^2}{2}. \quad (3.5)$$

Esta equação nos permite definir a distribuição dos LLR's através de um único parâmetro,  $\sigma_L$ . Assim, as mensagens  $L$  tem a seguinte distribuição:

$$p(L|b) = \frac{\exp \left[ -\frac{(L-b \cdot (\sigma_L^2/2))^2}{2\sigma_L^2} \right]}{\sqrt{2\pi}\sigma_L}. \quad (3.6)$$

A hipótese assumida para gerar as mensagens *a priori* é que, sendo elas LLR's, elas devem ter uma distribuição Gaussiana que pode ser definida por único parâmetro. Podemos obter o valor da informação mútua entre as mensagens  $L$  e os bits  $b$  através da seguinte equação, que é função exclusiva de  $\sigma_L$ :

$$I(L, b) = \frac{1}{2} \sum_{i=1,-1} \int_{-\infty}^{\infty} p(L|b = i) \cdot \log_2 \frac{2 \cdot p(L|b = i)}{p(L|b = 1) + p(L|b = -1)} dL = J(\sigma_L). \quad (3.7)$$

Para gerar mensagens que resultam num determinado nível de informação *a priori*, devemos primeiro determinar o parâmetro  $\sigma_L$  que gera o valor de informação mútua desejado. Isto pode ser

feito através da função  $J()$ , que não tem forma fechada mas pode ser aproximada através das seguintes equações, fornecidas em [28]:

$$J(\sigma) = \begin{cases} a_{J,1}\sigma^3 + b_{J,1}\sigma^2 + c_{J,1}\sigma & 0 \leq \sigma \leq \sigma^*, \\ 1 - \exp [a_{J,2}\sigma^3 + b_{J,2}\sigma^2 + c_{J,2}\sigma + d_{J,2}] & \sigma^* < \sigma < 10, \\ 1 & \sigma_L \geq 10, \end{cases}$$

onde  $\sigma^* = 1,6363$  e os parâmetros são:

$$\begin{aligned} a_{J,1} &= -0,0421061, & b_{J,1} &= 0,209252, \\ c_{J,1} &= -0,00640081, \\ a_{J,2} &= 0,00181491, & b_{J,2} &= -0,142675, \\ c_{J,2} &= -0,0822054, & d_{J,2} &= 0,0549508. \end{aligned} \tag{3.8}$$

A função  $J^{-1}()$  pode ser aproximada através das seguintes equações:

$$J^{-1}(I) = \begin{cases} a_{\sigma,1}I^2 + b_{\sigma,1}I + c_{\sigma,1}\sqrt{I} & 0 \leq I \leq I^* \\ -a_{\sigma,2} \ln b_{\sigma,2}(1 - I) - c_{\sigma,2}I & I^* < I < 1 \end{cases}$$

onde  $I^* = 0,3646$  e os parâmetros são:

$$\begin{aligned} a_{\sigma,1} &= 1,09542, & b_{\sigma,1} &= 0,214217, & c_{\sigma,1} &= 2,33727, \\ a_{\sigma,2} &= 0,706692, & b_{\sigma,2} &= 0,386013, & c_{\sigma,2} &= -1,75017. \end{aligned} \tag{3.9}$$

Para se gerar valores de  $L$  que forneçam um certo valor de informação a priori sobre um conjunto de variáveis, utilizamos a função  $J^{-1}()$  para determinar qual é o valor de  $\sigma_L$  necessário. Consequentemente sabemos o valor de  $\mu_L$ , o que nos permite gerar aleatoriamente o valor de  $L$  necessário, de acordo com a distribuição dada por (3.6).

### 3.1.2 Relação entre as mensagens transmitidas e o valor de informação

Para obter a curva EXIT de um bloco, precisamos determinar qual é o valor da informação mútua entre as mensagens de saída e o conjunto de variáveis correspondentes. Este valor pode ser calculado se tivermos a distribuição das mensagens em função do valor de informação fornecido *a priori* e do valor que as variáveis assumem.

Como mostraremos nas próximas seções, o detector consegue, a partir das mensagens recebidas e da realização do canal, gerar mensagens  $\eta$  a serem transmitidas para os códigos. A repetição deste processo várias vezes permite a obtenção das densidades de probabilidade das mensagens transmitidas pelo detector quando os bits valem  $-1$  ou  $1$ . Não é assumido que as distribuições das mensagens

transmitidas são Gaussianas. Com as distribuições de  $\eta$  consegue-se obter o valor da informação mútua entre as mensagens transmitidas e os bits de informação através da seguinte integral:

$$I_E(\eta, b) = \frac{1}{2} \sum_{i=1,-1} \int_{-\infty}^{\infty} p(\eta|b=i) \cdot \log \frac{2 \cdot p(\eta|b=i)}{p(\eta|b=1) + p(\eta|b=-1)} d\eta. \quad (3.10)$$

Esta informação mútua é o valor de informação extrínseca gerado. Se as mensagens  $\eta$  forem LLR's de  $b$  e  $N$  valores de  $\eta$  fossem gerados para respectivamente  $N$  bits, o valor de  $I_E$  pode ser aproximado através da seguinte somatória [31]:

$$\begin{aligned} I_E(\eta, b) &= \mathcal{H}\{b\} - \mathcal{H}\{b|\eta\} \\ &\approx 1 - \frac{1}{N} \sum_{i=1}^N \log_2 1 + \exp(-b_i \eta_i), \end{aligned} \quad (3.11)$$

onde assume-se que  $P(b=1) = P(b=-1) = 1/2$ .

Desta forma, podemos obter uma função  $I_{MUD}(I_A, E_T/N_0)$  do detector que associa um valor de informação *a priori*  $I_A$  (que determina a distribuição das mensagens recebidas) com um valor de informação extrínseca  $I_E$ , parametrizado por um valor de  $E_T/N_0$  do canal. O mesmo procedimento pode ser utilizado para obter a curva EXIT de qualquer bloco, desde que saibamos quais são as distribuições das mensagens de saída, dado um valor de  $I_A$ .

## 3.2 Revisão sobre grafos de fatores

Algumas funções são muito complicadas de serem avaliadas. Como veremos a seguir, a função que o detector realiza para determinar o valor dos bits não é fácil de ser calculada. Uma maneira de simplificar este cálculo é representar a função como um grafo de fatores [32]. Esta representação, juntamente com o algoritmo soma-produto, se aproveita do fato que algumas funções globais complicadas podem ser fatoradas em funções locais mais simples. Isto permite que o cálculo da função global seja feito aproximadamente ou exatamente através da computação dos fatores mais simples.

Suponha que haja uma função  $g(\mathbf{x})$  que pode ser escrita como o produto de  $u$  fatores na forma  $g(x_0, x_1, \dots, x_{v-1}) = \prod_{i=0}^{u-1} f_i(\mathbf{x}^i)$ , onde  $\mathbf{x}^i$  é o conjunto de variáveis que são argumento de  $f_i(\cdot)$ . Por conveniência omitiremos os parênteses ao nos referenciarmos às funções. As funções  $f_i$  são chamadas de funções locais ou fatores de  $g(\mathbf{x})$ . Um grafo de fatores é uma forma de representar esta fatoração através de um grafo bipartido que contém nós que representam variáveis e nós que representam funções. A variável  $x_i$  está conectada à função  $f_i$  se esta toma a primeira como argumento. Se calcularmos todas as funções locais, sabemos o valor de  $g(\mathbf{x})$ .

O algoritmo soma-produto é uma forma de se calcular e transmitir as mensagens entre os nós,

de forma a calcular  $g(\mathbf{x})$ . Isto é feito, nos nós-função, através da obtenção dos valores marginais de  $f_i$  em função de cada  $x_i$  (contida em  $\mathbf{x}^i$ ), e através do produto das mensagens de entrada num nó-variável. Dado um conjunto de elementos  $\mathbf{x}$  e  $x_i \in \mathbf{x}$ , a notação  $\mathbf{x} \perp x_i$  indica o mesmo conjunto  $\mathbf{x}$  excetuando-se o elemento  $x_i$ . Formalmente, a mensagem  $\eta_{f_i \rightarrow x_j}$  transmitida de um nó-função  $f_i$  para um nó-variável  $x_j$  é:

$$\eta_{f_i \rightarrow x_j}(x_j = x_j^*) = \sum_{\nu(f_i) \perp x_j} f_i(\mathbf{x}^i | x_j = x_j^*) \prod_{x_k \in \nu(f_i) \perp x_j} \eta_{x_k \rightarrow f_i}, \quad (3.12)$$

onde  $\nu(f_i) \perp x_j$  indica o conjunto de variáveis que são argumento de  $f_i$ , exceto  $x_j$ .

Seja  $\nu(x_i)$  o conjunto de todas as funções que tomam a variável  $x_i$  como argumento. A mensagem  $\eta_{x_i \rightarrow f_j}$  transmitida de um nó-variável  $x_i$  para um nó função  $f_j$  é dada por:

$$\eta_{x_i \rightarrow f_j}(x_i = x_i^*) = \prod_{f_k \in \nu(x_i) \perp f_j} \eta_{f_i \rightarrow x_j}(x_i^*), \quad (3.13)$$

onde  $\nu(x_i) \perp f_j$  indica o conjunto de funções que tomam  $x_i$  como argumento, exceto  $f_j$ .

Um nó possui condições de enviar uma mensagem através de um ramo se há mensagens disponíveis em todos os outros ramos. As mensagens são geradas nas variáveis que estão conectadas a somente uma função e se espalham pelo grafo, até que todos os nós tenham calculado todas as mensagens de saída, encerrando o algoritmo. Quando, partindo de um nó por um ramo e nunca voltando pelo mesmo ramo, retornamos ao nó de origem, o grafo possui ciclos. Neste caso, o algoritmo não tem terminação natural, pois devido a existência do ciclo, uma nova mensagem de saída em um ramo gerará uma nova mensagem de entrada em outro ramo do mesmo nó, requisitando um novo cálculo das mensagens de saída e assim indefinidamente. Na prática, o que se pode fazer é determinar uma ordem de passagem de mensagens e determinar quantas iterações do algoritmo realizaremos, ou parar o cálculo das mensagens de saída quando estas variam menos do que um valor definido. Além do critério de parada, pode ser necessário inicializar algumas variáveis para permitir o início do algoritmo.

A ordem com que as mensagens são geradas e passadas depende do cronograma estabelecido. O mesmo grafo pode ter vários cronogramas que podem influenciar na velocidade com que este calcula a sua função [33].

### 3.3 Representação do detector como um grafo de fatores

O detector vai utilizar os valores da saída do canal e de probabilidades *a priori* sobre os bits transmitidos para gerar probabilidades *a posteriori* sobre estes bits, obtendo assim uma curva EXIT. O cálculo destas probabilidades é muito complicado, o que nos leva a utilização de grafos de fatores para modelar a função que o detector realiza.

Para obter as curvas EXIT do detector, associamos  $k$  bits por usuário às mensagens  $N$ -árias, onde  $k = \log_2 N$ . As medidas de informação *a priori* e extrínseca tem estes bits como referência. O  $i$ -ésimo bit do  $t$ -ésimo usuário é representado por  $b_i^t$ , onde  $i$  varia entre 0 e  $k - 1$ . O conjunto de todos os bits pode ser representado pela matriz  $\mathbf{b}$ , com  $T$  linhas e  $k$  colunas. Como todas as mensagens são ortogonais entre si, a escolha do mapeamento entre bits e mensagens não é crítico em relação à probabilidade de erro de bit. Escolhemos um mapeamento natural. O conjunto de bits de todos os usuários,  $\mathbf{b}$ , pode ser associado ao vetor  $\mathbf{R}$  através das seguintes equações:

$$m^t = \sum_{i=0}^{k-1} \{b_i^t\} \cdot 2^i, \quad (3.14)$$

$$c_n^t = [m^t = n] \quad c_n = \sum_{t=1}^T c_n^t,$$

onde  $[\cdot]$  vale 1 se o seu predicado é verdadeiro ou vale 0 se for falso. Estas equações determinísticas podem gerar funções de probabilidade utilizando a mesma convenção:

$$P(m^t | \bar{b}_i^t) = \left[ m^t = \sum_{i=0}^{k-1} \{b_i^t\} \cdot 2^i \right],$$

$$P(c_0^t, c_1^t, \dots, c_{N-1}^t | m^t) = \prod_{n=0}^{N-1} \{ [c_n^t = 0][m^t \neq n] + [c_n^t = 1][m^t = n] \}, \quad (3.15)$$

$$P(c_n | \bar{c}_n^t) = \left[ c_n = \sum_{t=1}^T c_n^t \right].$$

Juntamente com a *pdf* de  $p(R_n | c_n)$ , podemos escrever  $p(\mathbf{R}, \mathbf{c}, \bar{c}_n^t, \mathbf{m}, \bar{b}_i^t)$  usando a regra da cadeia:

$$p(\mathbf{R}, \mathbf{c}, \bar{c}_n^t, \mathbf{m}, \bar{b}_i^t) = p(\mathbf{R} | \mathbf{c}) P(\mathbf{c} | \bar{c}_n^t) P(\bar{c}_n^t | \mathbf{m}) P(\mathbf{m} | \bar{b}_i^t). \quad (3.16)$$

Os fatores (funções locais) desta equação são:

$$p(\mathbf{R} | \mathbf{c}) = \prod_{n=0}^{N-1} p(R_n | c_n),$$

$$P(\mathbf{c} | \bar{c}_n^t) = \prod_{n=0}^{N-1} P(c_n | \bar{c}_n^t),$$

$$P(\bar{c}_n^t | \mathbf{m}) = \prod_{t=1}^T P(c_0^t, c_1^t, \dots, c_{N-1}^t | m^t),$$

$$P(\mathbf{m} | \bar{b}_i^t) = \prod_{t=1}^T P(m^t | \bar{b}_i^t). \quad (3.17)$$

Os fatores acima podem ser quebrados em fatores menores, um para cada termo do produto. Estes fatores estão representados nas Figs. 3.5 a 3.8.

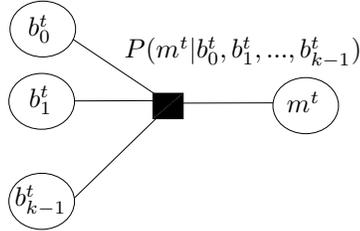


Fig. 3.5: Fator local para  $P(m^t | b_0^t, b_1^t, \dots, b_{k-1}^t)$ .

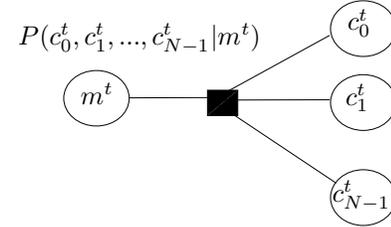


Fig. 3.6: Fator local para  $P(c_0^t, c_1^t, \dots, c_{N-1}^t | m^t)$ .

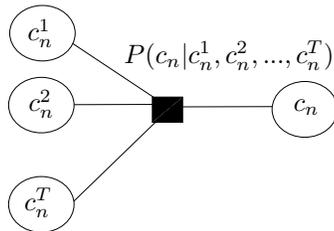


Fig. 3.7: Fator local para  $P(c_n | c_n^1, c_n^2, \dots, c_n^T)$ .

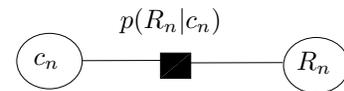


Fig. 3.8: Fator local para  $p(R_n | c_n)$ .

A combinação destes fatores através das variáveis em comum nos permite obter um grafo que representa a função global, apresentado na Fig. 3.9. Este grafo nos permite, através do algoritmo soma-produto, computar todos os fatores locais através da troca de mensagens entre os seus nós como descrito em [32].

### 3.4 Cálculo das mensagens do detector multiusuário

Nesta seção apresentaremos as funções que descrevem as mensagens que passam pelo grafo. Por se tratar de um grafo que representa uma densidade de probabilidade, as mensagens passadas pelos ramos são as probabilidades marginais das variáveis assumirem os valores. De posse da definição de cada uma das funções locais, as mensagens são diretamente obtidas a partir das equações (3.12) e (3.13) do algoritmo soma produto. Os nós que representam variáveis neste grafo simplesmente repassam as mensagens de entrada para os outros ramos de saída, pois nenhuma variável está conectada a mais de duas funções locais.

Uma propriedade do algoritmo soma-produto é que as mensagens emitidas por um nó através de um ramo dependem exclusivamente das mensagens que estão entrando nos outros ramos do nó. Por

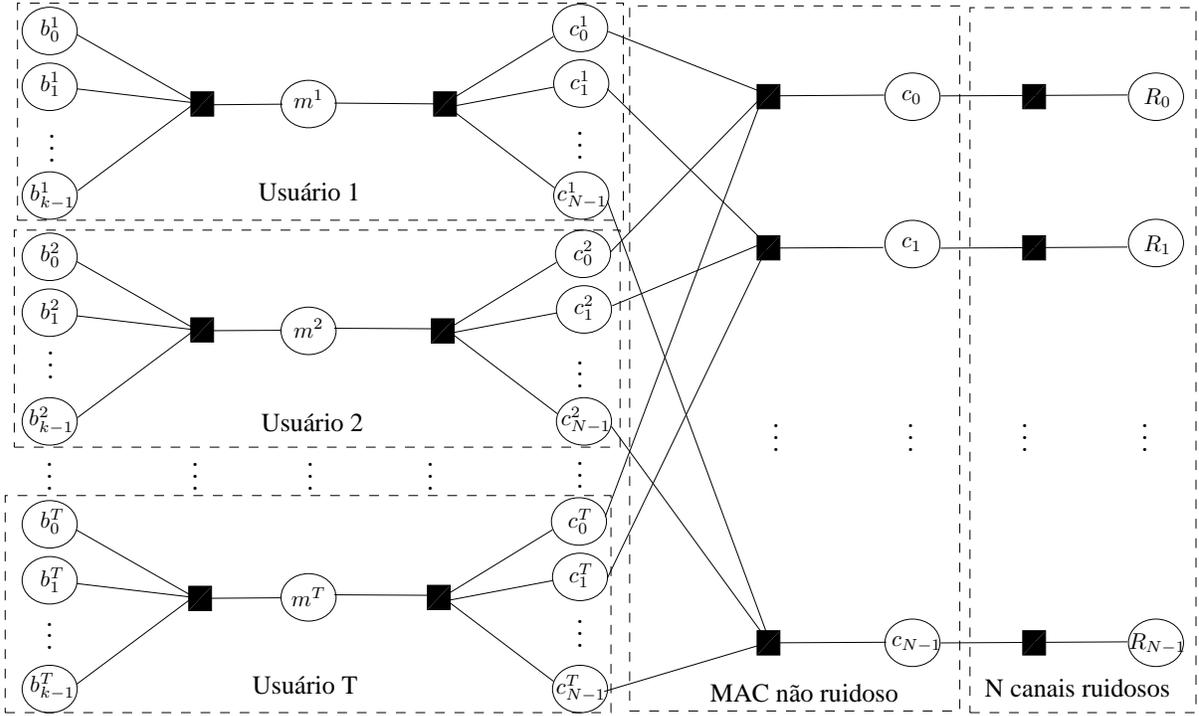


Fig. 3.9: Grafo de fatores de um detector para um sistema com  $T$  usuários e  $N$  frequências.

causa desta propriedade, as probabilidades marginais obtidas geram informação extrínseca, por não considerar as probabilidades *a priori* conhecidas sobre a variável.

### 3.4.1 Mensagens calculadas no nó $P(m^t | b_0^t, b_1^t, \dots, b_{k-1}^t)$

Este nó representa o mapeamento dos  $k$  bits do  $t$ -ésimo usuário na sua mensagem  $N$ -ária  $m^t$ . Como os sinais são ortogonais par a par, a escolha do mapeamento não é crítica para a probabilidade de erro de bit. Desta forma, escolhemos o mapeamento natural, obtendo assim:

$$P(m^t | b_0^t, b_1^t, \dots, b_{k-1}^t) = \left[ m^t = \sum_{i=0}^{k-1} \{b_i^t\} \cdot 2^i \right]. \quad (3.18)$$

Vamos representar este mapeamento pela função  $\varepsilon(\cdot)$ , que toma como argumento os  $k$  bits e retorna a mensagem, sendo  $\varepsilon_i^{-1}(\cdot)$  a função inversa que faz o mapeamento da mensagem  $N$ -ária para o  $i$ -ésimo bit,  $i = 0, 1, \dots, k - 1$ .

Como há somente um conjunto de bits que gera uma mensagem  $m^t$ , o cálculo de  $P(m^t = m^*)$  se resume em multiplicar as probabilidades dos bits assumirem os valores que geram  $m^*$ :

$$P(m^t = m^*) = \prod_{i=0}^{k-1} P(b_i = \varepsilon_i^{-1}(m^*)). \quad (3.19)$$

Para calcularmos os valores de  $P(b_i^t)$  precisamos realizar a seguinte operação:

$$P(b_i^t) = \sum_{m^t=0}^{N-1} P(m^t) \prod_{j=0, j \neq i}^{k-1} P(b_j^t = \varepsilon_j^{-1}(m^t)). \quad (3.20)$$

### 3.4.2 Mensagens calculadas no nó $P(c_0^t, c_1^t, \dots, c_{N-1}^t | m^t)$

A variável  $c_n^t$  pode assumir somente os valores 0 ou 1. A probabilidade desta variável assumir estes valores, dadas as probabilidades das outras variáveis deste nó, é:

$$P(c_n^t = 1) = P(m^t = n) \prod_{i=0, i \neq n}^{N-1} P(c_i^t = 0), \quad (3.21)$$

$$P(c_n^t = 0) = \sum_{m^*=0, m^* \neq n}^{N-1} P(m^t = m^*) P(c_m^t = 1) \prod_{i=0, i \neq n, m}^{N-1} P(c_i^t = 0).$$

A probabilidade da variável  $m^t$  assumir cada um dos  $N$  possíveis valores pode ser calculada através da seguinte equação.

$$P(m^t = n) = P(c_n^t = 1) \prod_{i=0, i \neq n}^{N-1} P(c_i^t = 0), \quad (3.22)$$

$$P(m^t \neq n) = \sum_{i=0, i \neq n}^{N-1} P(c_n^t = 0) P(c_i^t = 1) \prod_{j=0, j \neq i, n}^{N-1} P(c_j^t = 0).$$

### 3.4.3 Mensagens calculadas no nó $P(c_n | c_n^1, c_n^2, \dots, c_n^T)$

A variável  $c_n$  nos indica quantos usuários estão utilizando a  $n$ -ésima frequência. As variáveis  $c_n^t$  nos indicam se o  $t$ -ésimo usuário está utilizando a  $n$ -ésima frequência. Desta forma temos que  $c_n = \sum_{t=1}^T c_n^t$ , e a probabilidade  $P(c_n | c_n^1, c_n^2, \dots, c_n^T)$  é dada por:

$$P(c_n | c_n^1, c_n^2, \dots, c_n^T) = \left[ c_n = \sum_{t=1}^T c_n^t \right]. \quad (3.23)$$

A partir desta equação obtemos as seguintes:

$$\begin{aligned}
P(c_n = t^*) &= \sum_{\bar{c}_n^t \in H_T} \left[ t^* = \sum_{t=1}^T c_n^t \right] \prod_{t=1}^T P(c_n^t) \\
&= P\left( \sum_{t=1}^T c_n^t = t^* \right), \\
P(c_n^i = 1) &= \sum_{c_n=0}^T \left\{ \sum_{\bar{c}_n^t \perp c_n^i \in H_{T-1}} \left[ c_n = 1 + \sum_{t=1, t \neq i}^T c_n^t \right] \prod_{t=1, t \neq i}^T P(c_n^t) \right\} \\
&= \sum_{c_n=1}^T P(c_n) \cdot P\left( \sum_{t=1, t \neq i}^T c_n^t = T - c_n - 1 \right), \\
P(c_n^i = 0) &= \sum_{c_n=0}^T \left\{ \sum_{\bar{c}_n^t \perp c_n^i \in H_{T-1}} \left[ c_n = \sum_{t=1, t \neq i}^T c_n^t \right] \prod_{t=1, t \neq i}^T P(c_n^t) \right\} \\
&= \sum_{c_n=0}^{T-1} P(c_n) \cdot P\left( \sum_{t=1, t \neq i}^T c_n^t = T - c_n \right),
\end{aligned} \tag{3.24}$$

onde  $H_T$  é o espaço de Hamming  $T$ -ário<sup>1</sup>.

Os valores *a priori* de  $P(c_n = t^*)$  já estão disponíveis na entrada correspondente. Precisamos calcular os valores de  $P(\sum_{t=1}^T c_n^t = t^*)$  e  $P(\sum_{t=1, t \neq i}^T c_n^t = t^*)$  (o último somatório possui um termo a menos que o primeiro). Representando as mensagens de entrada referentes aos nós  $c_n^i$  como  $P(c_n^i) = \delta(0)P(c_n^i = 0) + \delta(1)P(c_n^i = 1)$ <sup>2</sup>, podemos obter estes valores através da convolução (representada por  $*$ ) das mensagens de entrada. Isto pode ser visualizado através do seguinte exemplo com três entradas:

<sup>1</sup>Um espaço de Hamming  $T$ -ário contém todos as  $2^T$  seqüências de  $T$  símbolos binários possíveis

<sup>2</sup>A função  $\delta(x)$  é um impulso em  $x$

$$\begin{aligned}
P(c_n^1) * P(c_n^2) &= \delta(0)P(c_n^1 = 0)P(c_n^2 = 0) + \delta(1)(P(c_n^1 = 0)P(c_n^2 = 1) \\
&\quad + P(c_n^1 = 1)P(c_n^2 = 0)) + \delta(2)P(c_n^1 = 1)P(c_n^2 = 1), \\
(P(c_n^1) * P(c_n^2)) * P(c_n^3) &= \delta(0)P(c_n^1 = 0)P(c_n^2 = 0)P(c_n^3 = 0) \\
&\quad + \delta(1)(P(c_n^1 = 0)P(c_n^2 = 0)P(c_n^3 = 1) \\
&\quad + P(c_n^1 = 0)P(c_n^2 = 1)P(c_n^3 = 0) + P(c_n^1 = 1)P(c_n^2 = 0)P(c_n^3 = 0)) \\
&\quad + \delta(2)(P(c_n^1 = 1)P(c_n^2 = 1)P(c_n^3 = 0) \\
&\quad + P(c_n^1 = 1)P(c_n^2 = 0)P(c_n^3 = 1) + P(c_n^1 = 0)P(c_n^2 = 1)P(c_n^3 = 1)) \\
&\quad + \delta(3)P(c_n^1 = 1)P(c_n^2 = 1)P(c_n^3 = 1).
\end{aligned} \tag{3.25}$$

A convolução pode ser implementada sequencialmente através do seguinte algoritmo:

$$\begin{aligned}
\mathbf{z}_0 &= [1 \ 0 \ 0 \ \dots], \\
\mathbf{z}_t &= \mathbf{z}_{t-1} \cdot P(c_n^t = 0) + [0 \ \mathbf{z}_{t-1}] \cdot (P(c_n^t = 1)).
\end{aligned} \tag{3.26}$$

Podemos descartar todos os valores de  $\mathbf{z}_t$  com índice maior do que  $T + 1$ , pois eles serão sempre iguais a zero. O vetor  $\mathbf{z}_T$  possui os valores de  $P(\sum_{t=1}^T c_n^t = t^*)$ . Para obtermos  $P(\sum_{t=1, t \neq i}^T c_n^t = t^*)$ , basta repetir o procedimento sem incluir o valor de  $P(c_n^i)$ .

Aparentemente este cálculo não pode ser simplificado. As mensagens que saem em direção ao nó  $c_n$  não precisam ser normalizadas porque, como todas as combinações de  $\overline{c_n^t}$  resultam num valor válido para  $c_n$ , o valor resultante de  $P(c_n)$  satisfaz  $\sum_{t=0}^T P(c_n = t) = 1$ . O mesmo não pode ser garantido sobre a distribuição de  $P(c_n^t)$ , cujo valor precisa ser normalizado.

### 3.4.4 Mensagens calculadas nos nós $p(R_n|c_n)$

Estes nós são responsáveis por calcular a probabilidade de  $c_n$  assumir um valor, dado o conhecimento de  $R_n$ , ou seja,  $P(c_n|R_n)$ . Estes valores podem ser diretamente calculados através da seguinte equação:

$$P(c_n|R_n) = \frac{p(R_n|c_n)P(c_n)}{p(R_n)}. \tag{3.27}$$

Os termos da equação acima estão dados na Seção 2.1. Dado que este valor depende exclusivamente dos valores do canal, ele precisa ser calculado somente uma vez por realização do canal. O valor  $p(R_n)$  normaliza os valores transmitidos para o nó que representa a variável  $c_n$  e não precisa ser diretamente calculado. Para o problema, não é necessário calcular as mensagens em direção ao nó  $R_n$ , já que o seu valor é conhecido na recepção.

### 3.5 Resultados

As curvas EXIT do detector foram obtidas simulando realizações do canal e processando o algoritmo soma-produto no grafo que corresponde ao detector. As mensagens recebidas foram modeladas como LLR's com distribuição Gaussiana com parâmetros dados pelo valor de informação *a priori* desejado. Uma vez geradas, o detector as converteu em valores de probabilidade utilizando a transformação:

$$P(b = 1) = \frac{\exp L}{1 + \exp L}. \quad (3.28)$$

Para cada curva foram gerados 21 pontos. O valor da informação *a priori* foi variado de 0 a 1, com graduação de 0,05 bits. Os passos para gerar a curva em cada ponto foram:

1. definir o valor de  $\sigma_L$  para obtermos o valor de  $I_A$  desejado, utilizando a função  $J^{-1}()$ ;
2. gerar aleatoriamente e armazenar 1 Mbits com valores iguais a -1 ou 1, com mesma probabilidade;
3. Gerar aleatoriamente e armazenar 1 milhão de variáveis Gaussianas com média nula e variância  $\sigma_L$
4. obter as mensagens de entrada do receptor ao somar os valores do passo anterior com o os bits multiplicados por  $\mu_L$ ;
5. agrupar os bits e mensagens gerados em grupos de  $T$  usuários e  $k = \log_2 N$  bits por usuário;
6. para cada grupo de  $T \cdot k$  bits, converter os bits em mensagens  $N$ -árias e posteriormente no vetor  $\mathbf{c}$  correspondente;
7. para cada  $\mathbf{c}$  obtido no passo anterior, gerar um vetor  $\mathbf{R}$  de acordo com as estatísticas do canal e a relação sinal ruído desejada;
8. para cada grupo de  $T \cdot k$  bits e o seu  $\mathbf{R}$ , processar o algoritmo soma-produto no grafo, obtendo assim 1 milhão de mensagens de saída  $\eta$  em formato LLR;
9. com o conhecimento dos bits gerados armazenados e com as mensagens de saída, obter o valor  $I_E$  utilizando a equação (3.11).

Um nó pode atualizar as suas mensagens de saída assim que novas mensagens de entrada estão disponíveis. Já que o grafo possui ciclos, acontece de mensagens serem passadas por um dado ramo

várias vezes, e o algoritmo de passagem de mensagens não tem uma terminação natural. Considerando isto, estabelecemos um cronograma “ping-pong” de passagem de mensagens: uma vez que as mensagens são geradas nos nós-bits e nós- canal ( $R_n$ ), as mensagens são atualizadas da direita para esquerda e de volta, de acordo com a Fig. 3.9. Fazer isto uma vez corresponde a uma iteração interna do detector multiusuário. Pode-se observar que cinco iterações internas são suficientes para estabilizar a distribuição de saída das mensagens que chegam nos nós-bit. Também observamos que há uma diferença significativa entre realizar uma ou cinco iterações, como mostra a Fig. 3.10. Para os resultados apresentados, utilizamos dez iterações internas.

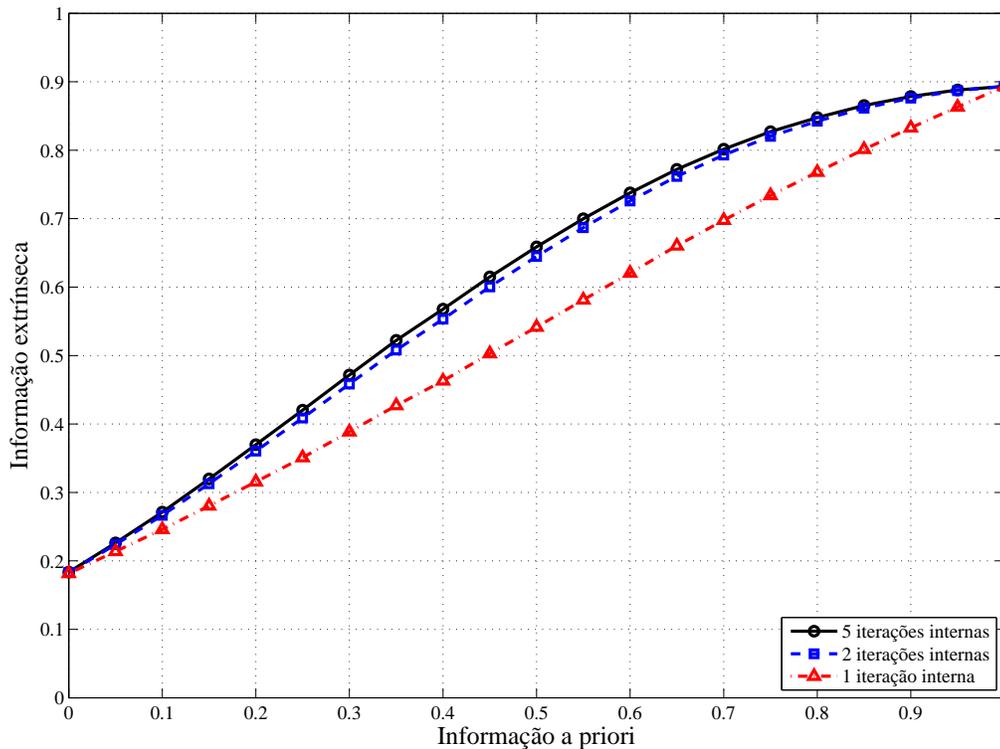


Fig. 3.10: Curvas EXIT do detector para  $N = 8$ ,  $T = 4$ ,  $E_t/N_0 = 20dB$ , detecção paralela, parametrizada pelo número de iterações internas

O sistema detector/decodificadores pode ser visto como a concatenação serial dos códigos dos usuários com um “código” de taxa 1 (o mapeamento dos bits nas mensagens  $N$ -árias). Isto nos permite comparar a área sob a curva EXIT do detector com a capacidade do canal, como mostrado em [34]. Para o caso do canal binário de apagamento, a área é exatamente igual à capacidade. Para o canal utilizado, a área pode ser considerada uma aproximação para a capacidade normalizada  $\bar{C}$  (vide equação (2.27)).

De acordo com a ordem como as mensagens são passadas pelo sistema detector/códigos, dois

cenários de detecção das mensagens são possíveis, como pode ser visualizado na Fig. 3.11: detecção paralela ou serial. Na detecção paralela, todos os usuários são detectados ao mesmo tempo. As mensagens que o detector gera alimentam os códigos de todos os decodificadores, que por sua vez alimentam o detector com novas mensagens. Neste caso, a curva EXIT do detector é a mesma para todos os usuários, que podem transmitir com a mesma taxa. Na detecção serial, somente um decodificador processa as mensagens vindas do detector por vez. Os outros decodificadores ou já terminaram o processo de decodificação ou ainda não iniciaram. Os usuários são detectados um por vez: os decodificadores dos usuários já detectados informam ao detector da sua decisão sobre os bits transmitidos, enquanto que os não detectados não informam nada ao detector. Para permitir taxas iguais a todos os usuários neste esquema, seria necessário uma troca constante de ordem de detecção dos usuários, o que acarretaria numa complexidade adicional. Para manter o esquema de detecção simples, escolhemos a detecção paralela.

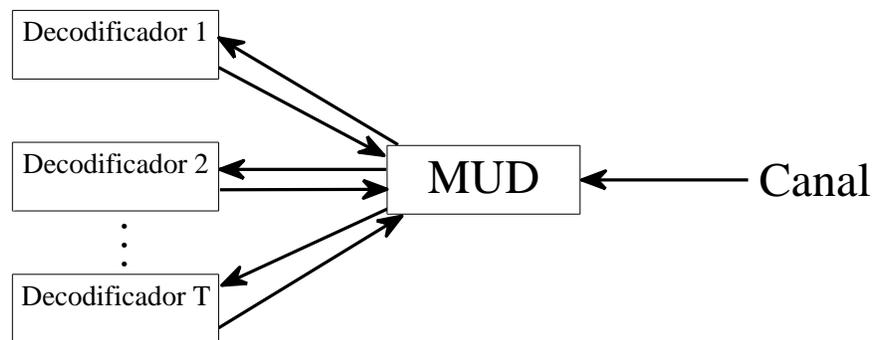


Fig. 3.11: Combinação do detector multiusuário com os decodificadores individuais de cada usuário

Resultados para  $N = 4, 8$  e  $16$  estão apresentados nas Figs. 3.12 a 3.14, onde  $I_D$  é a informação extrínseca fornecida pelo MUD. Para cada  $N$ , há um valor  $T_{MAX}$  que maximiza a capacidade soma efetiva. Escolhemos  $T = T_{MAX}, 2T_{MAX}/3$  e  $T_{MAX}/3$ , onde  $T_{MAX}$  foi obtido para  $E_t/N_0 = 20dB$ . Para  $N = 4, 8$  e  $16$ , o valor de  $T_{MAX}$  é respectivamente  $T = 3, 6$  e  $12$ , o que reforça a escolha da distribuição uniforme para os símbolos de entrada do canal. Omitimos a situação  $N = 4, T = 1$  por haver somente um usuário, e os casos onde  $T = 2$ , pois nesta situação o cálculo sofre problemas de precisão numérica. Os valores de  $E_t/N_0$  correspondem a energia do bit transmitido, e não à energia por bit de informação.

A Tab. 3.1 compara a área entre as curvas EXIT obtidas com 1 ou 10 iterações e a capacidade normalizada para a situação correspondente. Conclui-se que o aumento no número de iterações internas do detector contribui para uma melhora no desempenho esperado do sistema.

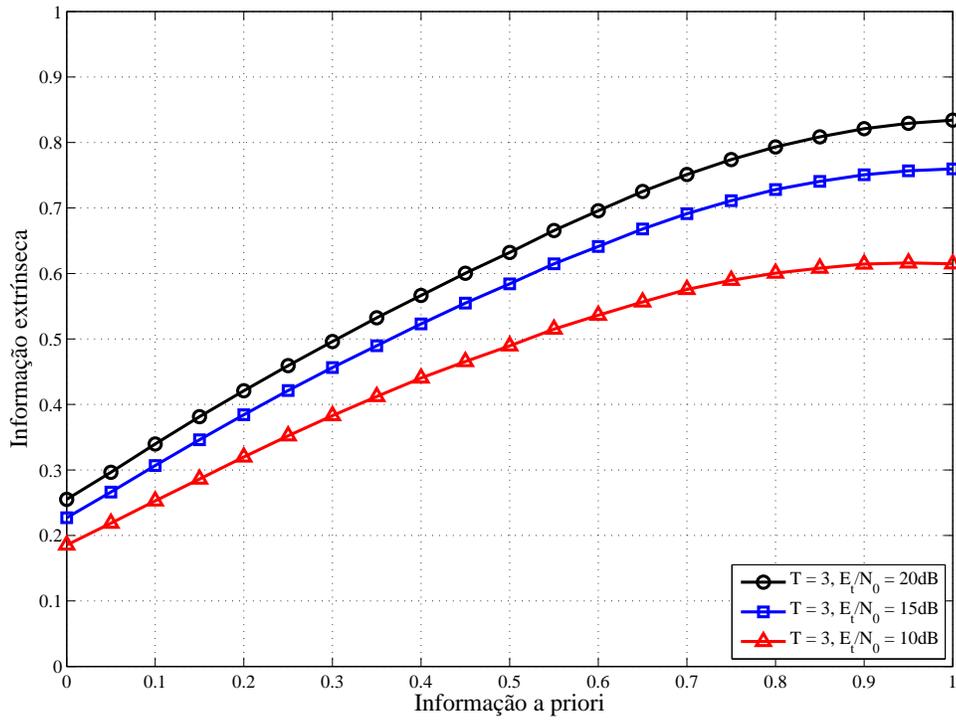


Fig. 3.12: Curvas EXIT para  $N = 4$ , detecção paralela, vários  $E_t/N_0$

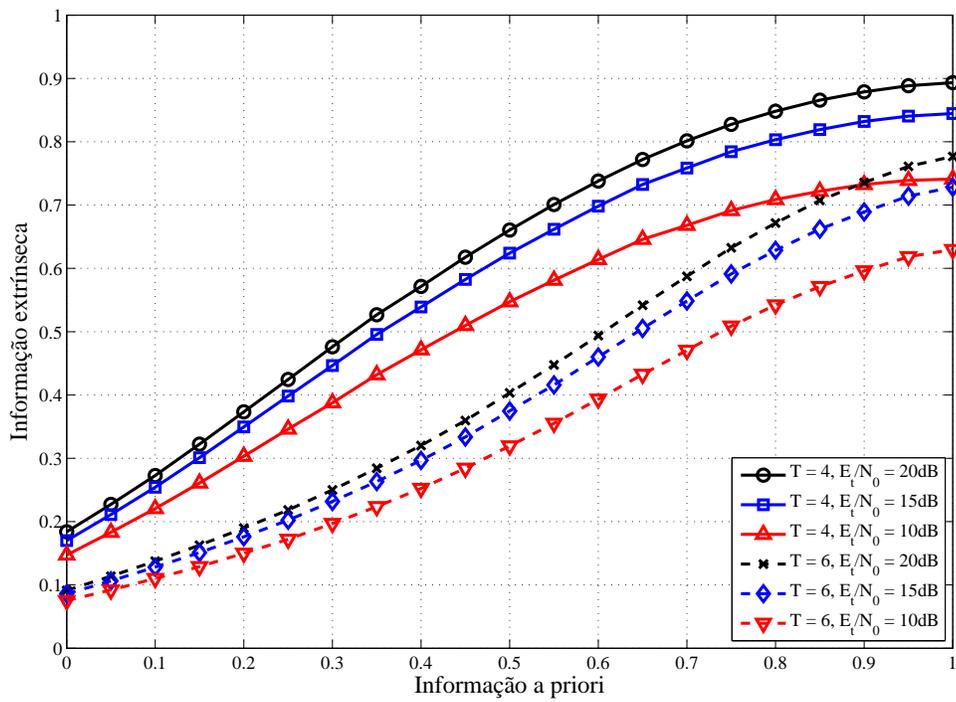


Fig. 3.13: Curvas EXIT para  $N = 8$ , detecção paralela, vários  $E_t/N_0$

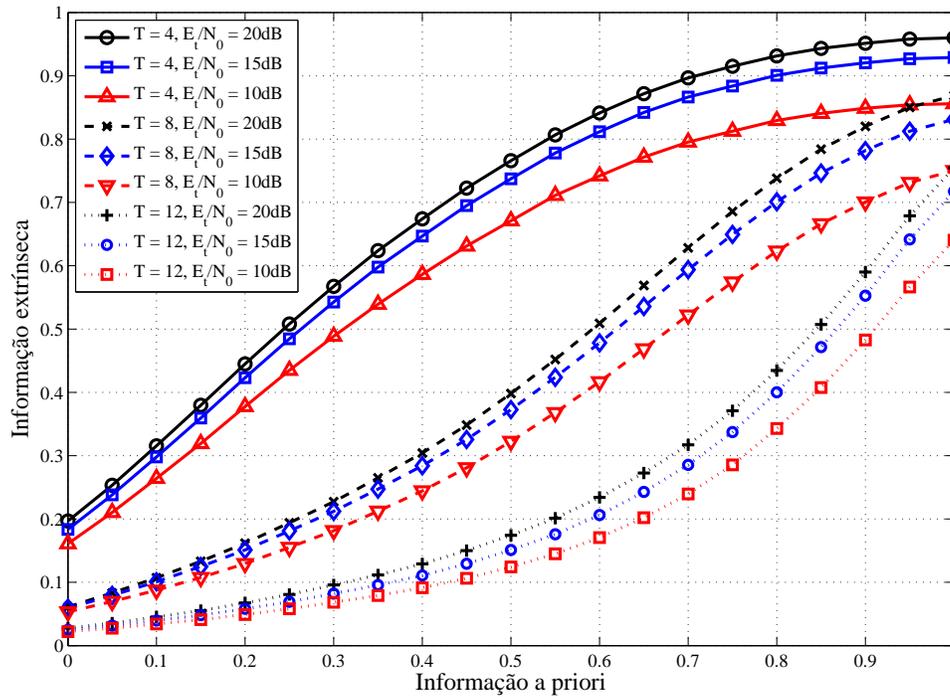


Fig. 3.14: Curvas EXIT para  $N = 16$ , detecção paralela, vários  $E_t/N_0$

### 3.6 Resumo

Neste capítulo desenvolvemos um grafo de fatores que representa a função matemática executada pelo detector multiusuário. Obtivemos a partir do algoritmo soma produto as mensagens que são passadas pelos ramos deste grafo.

A partir do grafo, obtivemos as curvas EXIT do detector para diversos casos. A partir destas curvas podemos perceber que a interferência entre os usuários é um fator mais forte do que a relação sinal-ruído para determiná-las. Estas curvas serão utilizadas posteriormente para obtermos sistemas multiusuário.

Tab. 3.1: Comparativo entre área sob a curva EXIT para detecção paralela e capacidade soma normalizada do sistema

N	T	$E_t/N_0$	Capacidade normalizada	Área	
				1 iteração	10 iterações
4	3	10	0,432	0,395	0,462
		15	0,520	0,485	0,556
		20	0,569	0,535	0,607
8	4	10	0,480	0,442	0,510
		15	0,550	0,508	0,582
		20	0,585	0,541	0,617
	6	10	0,339	0,300	0,338
		15	0,392	0,351	0,394
		20	0,420	0,377	0,423
16	4	10	0,597	0,543	0,612
		15	0,662	0,605	0,671
		20	0,690	0,635	0,697
	8	10	0,392	0,332	0,363
		15	0,441	0,377	0,412
		20	0,464	0,400	0,4362
	12	10	0,276	0,172	0,193
		15	0,311	0,208	0,225
		20	0,329	0,229	0,247



# Capítulo 4

## Análise de Códigos

Neste capítulo vamos analisar algumas classes de códigos que podem ser utilizadas em conjunto com o detector multiusuário apresentado no capítulo anterior. O objetivo é escolher uma classe para o projeto do sistema, detalhado no próximo capítulo.

As classes analisadas são:

- LDPC (do inglês: *Low Density Parity Check*), são códigos de bloco definidos através de uma matriz  $\mathbf{H}$  esparsa, ou seja, que possui principalmente zeros e, neste caso, relativamente poucos 1's [35];
- LDGM (do inglês: *Low Density Generator Matrix*), são códigos de bloco com matriz geradora  $\mathbf{G}$  esparsa [36];
- RA (do inglês: *Repeat-Accumulate*), pode ser interpretado como a concatenação serial de um código LDGM com um acumulador, um código convolucional com taxa um e uma memória cuja saída e o próximo estado da memória valem a soma módulo-2 do bit de entrada com o estado atual da memória [37].

Omitiremos a palavra “classe” neste capítulo de forma que o termo “código LDPC” se refira à classe dos códigos LDPC.

Os três códigos acima podem ser sistemáticos se os bits de informação fizerem parte da palavra código, ou não sistemáticos caso contrário. Os três códigos também admitem construção regular ou irregular. A construção é regular se  $\mathbf{H}$  (para LDPC) ou  $\mathbf{G}$  (para LDGM ou RA) possui o mesmo número de 1's em cada linha da matriz e ao mesmo tempo o mesmo número de 1's em cada coluna. O número de 1's em cada linha pode ser diferente do número de 1's em cada coluna. Se o número de 1's por linha ou por coluna variar, a construção é irregular.

Estes códigos podem ser representados através de grafos que possuem nós de variável e nós de paridade, que são nós de função que representam uma equação de paridade. A análise usual destes códigos [27; 28; 29] consiste em separar o grafo correspondente ao código em duas partes e gerar curvas EXIT para cada uma destas partes. Estas curvas dependem das condições do canal e dos parâmetros dos códigos. Para os códigos LDPC e LDGM, as curvas EXIT são facilmente obtidas através de equações analíticas que descrevem o comportamento de cada um dos módulos [28]. Para os códigos RA, utilizava-se uma combinação de equações analíticas e de simulações (para o acumulador) [29].

Para podermos comparar os diversos candidatos entre si, alteramos um pouco a análise usual de modo a obter, para cada código, uma única curva EXIT. Esta curva combinada pode ser analisada em conjunto com a curva EXIT do detector. Além disso, removemos a necessidade de simulação do acumulador.

## 4.1 Equações fundamentais dos códigos de bloco lineares

Todos os códigos considerados podem ser modelados como grafos de fatores, com nós que representam variáveis binárias e nós que representam as equações de paridade (ou soma módulo 2, indicado pelo símbolo  $\oplus$ )<sup>1</sup>. A obtenção das curvas EXIT dos códigos pode ser simplificada considerando duas equações fundamentais que relacionam o valor médio da informação *a priori* com o valor médio da informação extrínseca para cada um dos tipos de nós. Estas equações podem ser generalizadas para os casos irregulares agrupando os nós em camadas.

### 4.1.1 Transferência de informação para um nó-variável

Seja um nó-variável conectado a  $d_v$  ramos indexados. O valor  $d_v$  é chamado de grau do nó. Este nó representa um bit  $b$  num grafo. Denominaremos as mensagens que entram neste nó como  $\eta_A^i, i = 1, 2, \dots, d_v$ , onde  $i$  indica índice do ramo e o subscrito  $A$  indica que é uma mensagem de entrada (*a priori*). As mensagens de saída serão denominadas  $\eta_E^i, i = 1, 2, \dots, d_v$ , onde o subscrito  $E$  indica que é uma mensagem de saída (extrínseca).

Se as mensagens (de entrada e saída) representam a probabilidade do bit ser igual a um, as mensagens de saída podem ser calculadas a partir da seguinte equação, obtida através da aplicação de (3.13):

<sup>1</sup>Se  $a \oplus b \oplus c = d$ , a equação de paridade correspondente é  $a \oplus b \oplus c \oplus d = 0$

$$\eta_E^i = \frac{\prod_{j=1, j \neq i}^{d_v} \eta_A^j}{\prod_{j=1, j \neq i}^{d_v} \eta_A^j + \prod_{j=1, j \neq i}^{d_v} (1 - \eta_A^j)}, \quad (4.1)$$

onde o denominador serve para normalizar os valores das mensagens.

Para simplificar os cálculos, pode-se utilizar os valores LLR [30] (vide Seção 3.1.1) no lugar das probabilidades. Dado um valor  $p(b = 1)$  que representa a probabilidade do bit  $b$  ser igual a 1, o seu valor LLR é dado por  $L(b)$ , definido como:

$$L(b) = \log \frac{p(b = 1)}{p(b = 0)} = \log \frac{p(b = 1)}{1 - p(b = 1)}. \quad (4.2)$$

Se as mensagens (de entrada e saída) representam o valor  $LLR$  das probabilidades, as mensagens de saída podem ser facilmente calculadas através da seguinte equação:

$$\eta_E^i = \sum_{j=1, j \neq i}^{d_v} \eta_A^j. \quad (4.3)$$

As mensagens LLR de entrada (informação *a priori* sobre o bit representado por este nó) podem ser modeladas por uma distribuição Gaussiana com média e variância relacionados através da função  $J(\cdot)$ . Se a variável LLR aleatória  $\eta_A^i$  fornece um valor  $I_A^j$  sobre o bit  $b$ , a sua variância  $\sigma_A^j$  pode ser obtida através da função inversa  $J^{-1}(\cdot)$  (vide Seção 3.1.1). A mensagem de saída  $\eta_E^i$ , sendo a soma de  $d_v - 1$  variáveis Gaussianas aleatórias, é também uma variável Gaussiana variância dada por:

$$\sigma_i^{E^2} = \sum_{j=1, j \neq i}^{d_v} \sigma_A^j{}^2. \quad (4.4)$$

Assim, a mensagem de saída fornece um valor de informação extrínseca sobre o bit que pode ser calculada através da seguinte equação:

$$I_i^E(I_A, d_v) = J \left( \sqrt{\sum_{j=1, j \neq i}^{d_v} J^{-1}(I_A^j)^2} \right). \quad (4.5)$$

Esta equação informa quanto *em média* as mensagens de saída de um nó-variável binário informam sobre o bit representado, dado o valor médio da informação *a priori* fornecido pelas mensagens de entrada.

Tab. 4.1: Aproximação linear por partes de  $\Delta(|x|) \approx \log(1 + \exp -|x|)$ 

$ x $	$[0 \rightarrow 0, 5)$	$[0, 5 \rightarrow 1, 6)$	$[1, 6 \rightarrow 2, 2)$	$[2, 2 \rightarrow 3, 2)$	$[3, 2 \rightarrow 4, 4)$	$[4, 4 \rightarrow \infty)$
$\Delta( x )$	$\frac{- x }{2} + \frac{7}{10}$	$\frac{- x }{4} + \frac{23}{40}$	$\frac{- x }{8} + \frac{3}{8}$	$\frac{- x }{16} + \frac{19}{80}$	$\frac{- x }{32} + \frac{11}{80}$	0

### 4.1.2 Transferência de informação para um nó-paridade

Um nó de paridade representa uma equação de soma módulo 2. Ele pode representar tanto uma operação de soma realizada para a obtenção de um bit (o resultado da soma) como para representar uma equação de paridade (quando o resultado da soma deve valer zero).

A probabilidade de um bit ser igual a um é igual a probabilidade da soma de todos os outros bits serem iguais a um. Sendo  $p_1$  e  $p_2$  as probabilidades dos bits  $b_1$  e  $b_2$  serem iguais a um, a probabilidade da soma módulo 2 ser igual a um vale:

$$p(b_1 \oplus b_2 = 1) = p_1 \cdot (1 - p_2) + p_2 \cdot (1 - p_1). \quad (4.6)$$

Utilizando os valores LLR correspondentes  $L_1 = LLR(b_1)$  e  $L_2 = LLR(b_2)$ , foi mostrado em [30] que:

$$L(b_1 \oplus b_2) = \log \frac{e^{L_1} + e^{L_2}}{1 + e^{L_1 + L_2}}. \quad (4.7)$$

Indicaremos esta operação através do operador binário  $\boxplus$ . Também foi mostrado que  $L(b_1 \oplus b_2 \oplus \dots \oplus b_n) = L_1 \boxplus L_2 \boxplus \dots \boxplus L_n$ . A operação  $\boxplus$  pode ser eficientemente calculada utilizando a seguinte equação:

$$\begin{aligned} L_1 \boxplus L_2 &= \log [\exp(L_1) + \exp(L_2)] - \log [1 + \exp(L_1 + L_2)] \\ &= LSE(L(b_1), L(b_2)) - LSE(0, L(b_1) + L(b_2)), \end{aligned} \quad (4.8)$$

onde a função  $LSE(a, b)$  é o logaritmo da soma dos exponenciais. A função  $LSE(a, b)$  pode ser aproximada de algumas formas como mostrado em [38]. O método escolhido foi a de linearização por partes, resultando em:

$$\begin{aligned} LSE(a, b) &= \max a, b + \log 1 + \exp(-|a - b|) \\ &\approx \max a, b + \Delta(|a - b|) \end{aligned} \quad (4.9)$$

onde  $\Delta(|x|)$  é uma aproximação de  $\log(1 + \exp -|x|)$  como mostrado na Tab. 4.1. Desta forma substitui-se o cálculo de exponenciais e logaritmos por comparações e equações lineares, que são bem menos custosas computacionalmente.

A função que determina o valor da informação extrínseca em um ramo de um nó de verificação de paridade a partir dos valores da informação a priori existente considera uma propriedade das curvas EXIT mostrada em [34], que relaciona a curva EXIT de um código linear com o seu dual através da seguinte equação:

$$I_{\perp}^E(I^A) = 1 - I^E(1 - I^A), \quad (4.10)$$

onde  $I^E$  é a curva EXIT de um código e  $I_{\perp}^E$  é a curva EXIT do seu dual. Esta propriedade é exata para o canal binário de apagamento, sendo uma aproximação para o canal Gaussiano [39].

O nó de paridade com  $d_c$  ramos pode ser utilizado para representar um código que possui somente um bit de paridade (SPC - do inglês *Single Parity Check*) e comprimento  $d_c$ . O código dual de um código SPC é o código de repetição com comprimento  $d_c$ , que pode ser representado como um nó-variável com  $d_c$  ramos. Desta forma, aplicando (4.5) a (4.10), obtemos a equação que relaciona o valor médio da informação extrínseca das mensagens de saída com o valor médio da informação *a priori* das mensagens de entrada:

$$I_i^E(I^A, d_c) = 1 - J \left( \sqrt{\sum_{j=1, j \neq i}^{d_c} J^{-1}(1 - I_A^j)^2} \right). \quad (4.11)$$

### 4.1.3 Generalização para códigos irregulares

Em códigos irregulares, os nós de um tipo não possuem a mesma quantidade de ramos. As equações (4.5) e (4.11) podem ser generalizadas para este caso agrupando os nós em camadas e considerando uma função de distribuição de graus.

Os nós podem ser agrupados em três tipos de camadas:

- VND (do inglês: *Variable Node Detector*), é a camada que agrupa os nós-variável que representam um conjunto de bits, sejam transmitidos ou de informação;
- CND (do inglês: *Check Node Detector*), é a camada que agrupa os nós-função que representam as equações de paridade (ou soma módulo-2);
- ACC, é a camada que agrupa os nós que representam o acumulador.

A única variação possível na estrutura da camada ACC é o seu comprimento. As camadas VND e CND possuem, além do comprimento, uma distribuição de graus que informa quanto ramos estão conectados aos nós das camadas. A distribuição pode ser representada através das funções  $d_v(i)$  ( $d_c(i)$ ) que retornam, para qualquer valor inteiro de  $i > 0$ , a fração dos nós de VND(CND) que

possuem  $i$  ramos conectados a ele. Para facilitar a comparação entre os códigos, os ramos que estão conectados ao canal ou ao acumulador (no máximo um ramo por nó) não serão contabilizados por estas funções. Os valores médios  $\tilde{d}_v$  e  $\tilde{d}_c$  podem ser obtidos através das seguintes equações:

$$\begin{aligned}\tilde{d}_v &= \sum_{i=1}^{d_v^{MAX}} d_v(i) \cdot i, \\ \tilde{d}_c &= \sum_{i=1}^{d_c^{MAX}} d_c(i) \cdot i,\end{aligned}\tag{4.12}$$

onde  $d_v^{MAX}$  e  $d_c^{MAX}$  são respectivamente os maiores valores de  $i$  tais que  $d_v(i) > 0$  e  $d_c(i) > 0$ .

É possível obter o valor médio da informação extrínseca  $I^E$  das mensagens que saem de uma camada em função do valor médio da informação *a priori*  $I^A$  das mensagens que entram na camada. Para isto, considera-se que todos as mensagens de entrada fornecem o mesmo valor de  $I^A$ . Isto permite calcular os valores de  $I^E(I^A, i)$  em função do número de ramos dos nós, que é o mesmo valor para nós com o mesmo número de ramos e diferente entre nós com quantidades de ramos desiguais. Utilizando as funções  $d_v(i)$  e  $d_c(i)$ , pode-se obter o valor  $I^E$  da camada irregular através de uma soma ponderada dos valores de  $I^E(I^A, i)$ , onde a ponderação é feita em função do percentual de ramos que retornam um valor  $I^E(I^A, i)$ .

Data uma distribuição de graus genérica  $d_x(i)$  (que pode ser  $d_v(i)$  ou  $d_c(i)$ ) com valor médio  $\tilde{d}_x$ , a fração de ramos que saem de um nó com grau  $i$  em uma camada com  $B$  nós é:

$$\frac{B \cdot i \cdot d_x(i)}{\sum_{j=0}^{d_x^{MAX}} b \cdot j \cdot d_x(j)} = \frac{i \cdot d_x(i)}{\tilde{d}_x}.\tag{4.13}$$

Como consequência, o valor de  $I^E$  em função de  $I^A$  e de uma distribuição de graus  $d_x()$  pode ser definido através da seguinte equação:

$$I^E(I^A, d_x(i)) = \sum_{i=2}^{d_x^{MAX}} I^E(I^A, i) \cdot \frac{i \cdot d_x(i)}{\tilde{d}_x}.\tag{4.14}$$

## 4.2 Códigos LDPC

Códigos LDPC foram propostos em [35] em 1962, mas foram esquecidos por aproximadamente 30 anos. Com o advento dos códigos Turbo, estes códigos voltaram a chamar a atenção devido a sua natureza iterativa de decodificação. A complexidade do decodificador é baixa. O desempenho, para comprimentos longos, se aproxima à capacidade do canal AWGN. Vamos considerar somente

códigos LDPC binários.

Códigos LDPC são definidos através de uma matriz de verificação de paridade  $\mathbf{H}$  (com  $B - K$  linhas e  $B$  colunas) esparsa, isto é, ela possui poucos diferentes de zero e muitos elementos iguais a zero. As palavras código  $\mathbf{v}$  (um vetor com  $B$  elementos) de um código LDPC são todas aquelas que satisfazem  $\mathbf{H} \cdot \mathbf{v}^T = \mathbf{0}$ . As  $B$  colunas da matriz  $\mathbf{H}$  pode ser associadas aos  $B$  bits da palavra código. As  $B - K$  linhas da matriz representam  $B - K$  equações de paridade que somam bits de acordo com a entrada da matriz: o bit  $i$  faz parte da equação  $j$  se a entrada da matriz na coluna  $i$  e linha  $j$  vale um.

Um código LDPC pode ser representado através de um grafo que mostra a conexão entre os  $B$  bits da palavra código e as  $B - K$  equações de paridade, como mostra a Fig. 4.1. Isto pode ser feito através da concatenação de uma camada VND e de uma camada CND. Os nós das camadas são conectados de acordo com a matriz  $\mathbf{H}$ : o nó  $i$  da camada VND está conectado ao nó  $j$  da camada CND se a entrada da matriz na coluna  $i$  e linha  $j$  vale um. O número de 1's por linha é o valor de  $d_v$ . O número de 1's por coluna é o valor de  $d_c$ . Se todos os nós-variável possuírem o mesmo grau  $d_v$  e todos os nós-paridade possuírem o mesmo grau  $d_c$ , o código é regular. Caso contrário, o código é irregular e as funções  $d_v(i)$  e/ou  $d_c(i)$  são necessárias.

Além das conexões entre as camadas VND e CND, há também uma conexão da camada VND com o “exterior” do código, que fornece alguma informação sobre a palavra código. A parte externa pode ser um canal, um detector ou um outro código, por exemplo. Para facilitar a notação e explicitar a diferença dos valores de informação provenientes da camada CND e do exterior, esta conexão não é contabilizada em  $d_v$  ou  $d_v(i)$ , isto é, um nó da camada VND de um código LDPC com grau  $i$  possui  $i + 1$  ramos conectados a ele.

O número de ramos que saem do VND em direção ao CND é necessariamente igual ao número de ramos que saem do CND em direção ao VND. Desta forma podemos obter a taxa por projeto do código:

$$\begin{aligned} B \cdot \tilde{d}_v &= (B - K) \cdot \tilde{d}_c, \\ Tx_{LDPC} &= \frac{K}{B} = 1 - \frac{\tilde{d}_v}{\tilde{d}_c}. \end{aligned} \quad (4.15)$$

Como a taxa do código deve ser positiva, conclui-se que o valor de  $\tilde{d}_v$  deve ser menor do que  $\tilde{d}_c$ .

Para que o código LDPC seja sistemático, a matriz de paridade deve ter o formato  $\mathbf{G} = [\mathbf{I}_K | \mathbf{P}]$ . Isto permite que a matriz geradora seja escrita como  $[-\mathbf{P}^T | \mathbf{I}_{B-K}]$ , resultando em  $\mathbf{G} \cdot \mathbf{H}^T = P - P = 0$ .

A Fig. 4.2 mostra a direção das trocas de informação entre as camadas. As letras do subscrito V, C e D representam respectivamente o VND, CND e detector. Quando há duas letras, a primeira letra indica a origem e a segunda o destino. Por exemplo,  $I_{VC}$  esta associado às mensagens que partem do VND em direção ao CND. Os blocos denominados por  $\Pi$  e  $\Pi^{-1}$  implementam as conexões entre

as camadas VND e CND de forma aleatória: as mensagens de uma camada entram sequencialmente no bloco; a ordem de saída é definida pela permutação. Desta forma as conexões não fazem parte das camadas dos códigos. Mais detalhes sobre a implementação das ligações entre as camadas serão vistos na Seção 5.2.

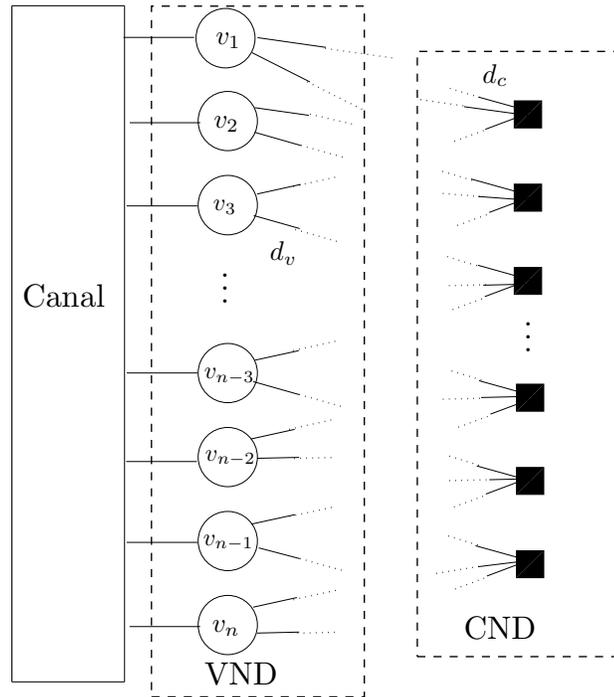


Fig. 4.1: Grafo que representa um código LDPC regular com  $d_v = 2$  e  $d_c = 3$ .

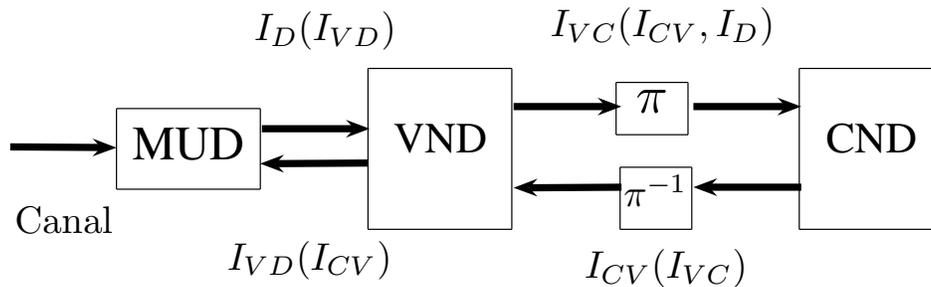


Fig. 4.2: Direção de troca de informações para MUD e LDPC.

Pode-se obter curvas EXIT para as camadas VND e CND. O CND envia mensagens para o VND. O VND esta conectado ao canal, que nos fornece um valor  $I_D$  de informação a priori sobre o bit transmitido. O VND possui duas interfaces e consequentemente duas curvas EXIT. Considerando que na entrada de cada camada, o valor de informação a priori proveniente da outra camada é o mesmo em todos os ramos, as curvas EXIT  $I_{VC}(\cdot)$ ,  $I_{VD}(I_{CV}, d_v)$  e  $I_{CV}(\cdot)$  de um código LDPC regular podem então ser escritas como:

$$\begin{aligned} I_{VC}(I_{CV}, I_D, d_v) &= J \left( \sqrt{J^{-1}(I_D)^2 + (d_v - 1)J^{-1}(I_{CV})^2} \right), \\ I_{CV}(I_{VC}, d_c) &= 1 - J \left( J^{-1}(1 - I_{VC})\sqrt{d_c - 1} \right), \\ I_{VD}(I_{CV}, d_v) &= J(J^{-1}(I_{CV})\sqrt{d_v}). \end{aligned} \quad (4.16)$$

Para códigos LDPC irregulares, as equações se tornam:

$$\begin{aligned} I_{VC}(I_{CV}, I_D, d_v(i)) &= \sum_{i=1}^{d_v^{MAX}} \frac{a_i \cdot d_v(i)}{\tilde{d}_v} \cdot I_{VC}(I_{CV}, i), \\ I_{CV}(I_{VC}, d_c(i)) &= \sum_{i=1}^{d_c^{MAX}} \frac{b_i \cdot d_c(i)}{\tilde{d}_c} \cdot I_{CV}(I_{VC}, i), \\ I_{VD}(I_{CV}, d_v(i)) &= \sum_{i=1}^{d_v^{MAX}} a_i \cdot J(J^{-1}(I_{CV})\sqrt{i}). \end{aligned} \quad (4.17)$$

### 4.3 Códigos LDGM

Códigos LDGM [40] (do inglês - *Low Density Generator Matrix*) são definidos, assim como códigos LDPC, através de uma matriz esparsa. Entretanto, enquanto a matriz dos códigos LDPC é a matriz de verificação de paridade  $\mathbf{H}$ , a matriz do código LDGM é uma matriz de geração da palavra código  $\mathbf{G}$ , de forma que a palavra código  $\mathbf{v}$  é gerada através de uma sequência de informação  $\mathbf{b}$  é através de  $\mathbf{v} = \mathbf{b} \cdot \mathbf{G}$ . Desta forma, a matriz  $\mathbf{G}$  possui  $K$  linhas e  $B$  colunas. Da mesma forma como códigos LDPC, estes códigos podem ser analisados através dos parâmetros  $d_v$  e  $d_c$  para códigos regulares ou pelas funções  $d_v(i)$  e  $d_c(i)$  para códigos irregulares. O parâmetro  $d_v$  indica quantos 1's estão presentes por linha de  $\mathbf{G}$ , menos um se o código for sistemático. O parâmetro  $d_c$  indica quantos 1's estão presentes em cada coluna de  $\mathbf{G}$ . No caso de códigos LDGM, o valor de  $d_c$  representa quantos ramos estão conectados a um nó de paridade menos um, que é o ramo que transmite o bit através do canal. Para códigos LDGM não sistemáticos, o valor de  $d_v$  é igual ao número de ramos que estão conectados aos nós-variável.

A taxa deste código é  $K/B$ , e pode ser dada em função dos parâmetros igualando-se o número de ramos que saem das camadas VND e CND em direção à outra camada. Quando o código é sistemático temos:

$$\begin{aligned} K \cdot \tilde{d}_v &= (B - K) \cdot \tilde{d}_c, \\ Tx_{LDGM}^S &= \frac{K}{B} = \frac{\tilde{d}_c}{\tilde{d}_v + \tilde{d}_c}. \end{aligned} \quad (4.18)$$

Quando o código é não sistemático temos:

$$\begin{aligned} K \cdot \tilde{d}_v &= (B) \cdot \tilde{d}_c, \\ Tx_{LDGM}^{NS} &= \frac{K}{B} = \frac{\tilde{d}_c}{\tilde{d}_v}. \end{aligned} \quad (4.19)$$

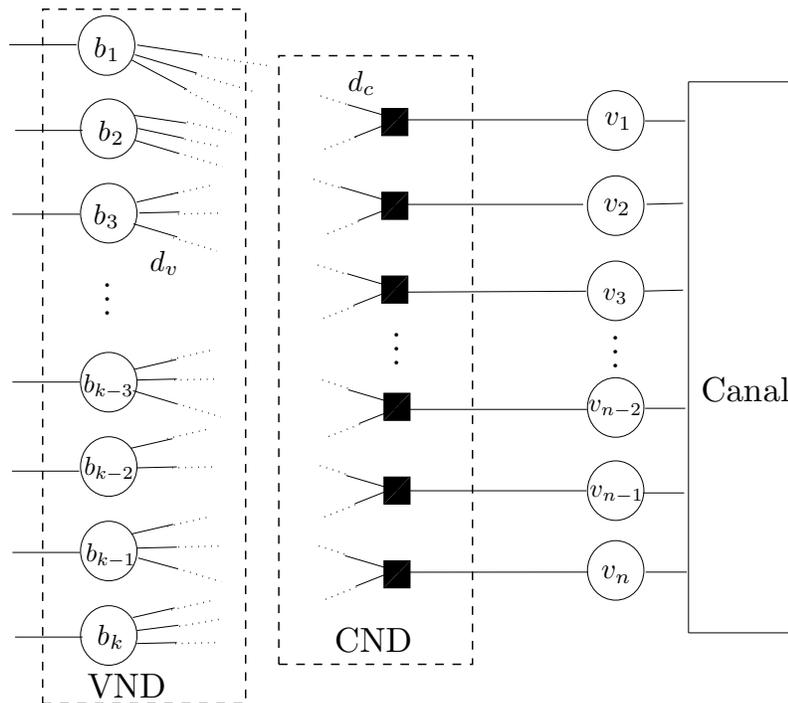


Fig. 4.3: Grafo que representa um código LDGM não sistemático regular com  $d_v = 3$  e  $d_c = 2$ . Para o caso sistemático, os nós que representam os bits também são transmitidos.

A codificação para estes códigos é direta, feita através da matriz  $\mathbf{G}$ . Como ela é esparsa, pode-se armazenar, no lugar da própria matriz, a relação dos bits que geram cada bit de saída. A decodificação pode ser feita através da passagem de mensagens pelo grafo equivalente, apresentado na Fig. 4.3.

As curvas EXIT de um código LDGM regular são dadas por:

$$\begin{aligned}
 I_{VC}(I_{CV}, I_D, d_v) &= J \left( \sqrt{J^{-1}(s \cdot I_D)^2 + (d_v - 1)J^{-1}(I_{CV})^2} \right), \\
 I_{CV}(I_{VC}I_{AC}, d_c) &= 1 - J \left( \sqrt{(d_c - 1)J^{-1}(1 - I_{VC})^2 + J^{-1}(1 - I_{AC})^2} \right), \\
 I_{CD}(I_{VC}, d_c) &= 1 - J \left( J^{-1}(1 - I_{VC})\sqrt{d_c} \right), \\
 I_{VD}(I_{CV}, d_v) &= s \cdot J \left( \sqrt{d_v}I_{CV} \right),
 \end{aligned} \tag{4.20}$$

onde  $s$  vale um se o código for sistemático e zero caso contrário.

As curvas EXIT para um código LDGM irregular são dadas por:

$$\begin{aligned}
 I_{VC}(I_{CV}, I_D, d_v(i)) &= \sum_{i=1}^{d_v^{MAX}} d_v(i) \cdot I_{VC}(I_{CV}, I_D, i), \\
 I_{CV}(I_{VC}, I_{AC}, d_c(i)) &= \sum_{i=1}^{d_c^{MAX}} d_c(i) \cdot I_{CV}(I_{VC}, I_{AC}, i), \\
 I_{CD}(I_{VC}, d_c(i)) &= \sum_{i=1}^{d_c^{MAX}} d_c(i) I_{CA}(I_{VC}, i), \\
 I_{VD}(I_{CV}, d_v(i)) &= \sum_{i=1}^{d_v^{MAX}} d_v(i) I_{VD}(I_{CV}, i).
 \end{aligned} \tag{4.21}$$

Códigos LDGM não sistemáticos precisam ter alguma fração de nós VND com grau 1. Isto pode ser visto da Fig. 4.3 e pela equação (4.11). Se o valor da informação *a priori* que entra em um nó-paridade for igual a zero, o valor da informação extrínseca que sai em todos os outros ramos será igual a zero, como mostra a equação (4.11). No início do processo de decodificação, o valor de  $I_{VC}$  é zero. Assim, se todos os nós de VND tiverem grau maior do que 1, sempre haverá pelo menos um ramo de entrada (vindo de VND) com um valor de informação *a priori* igual a zero. Desta forma, o valor de  $I_{CV}$  é sempre igual a zero e consequentemente  $I_{VC}$  também. Por outro lado, se parte dos nós de VND tiverem grau 1, o valor de  $I_D$  passa através da camada VND e atinge VND, que pode responder com um valor de  $I_{VC} > 0$ .

Este problema não ocorre em códigos LDGM sistemáticos pois a camada VND utiliza o valor de  $I_D$  diretamente, o que permite que a camada VND gere valores de  $I_{CV}$  maior do que zero. Para códigos sistemáticos, o valor da informação mútua passada ao detector é a soma ponderada de  $I_{CD}$  e  $I_{VD}$ , ponderados por  $(B - K)/B$  e  $K/B$  respectivamente.

## 4.4 Códigos RA

Códigos RA (do inglês: *Repeat-Accumulate*) [37] são uma alternativa aos códigos LDPC e turbo, tendo como vantagem a simplicidade da obtenção de palavras-código. Como o próprio nome diz, a codificação é feita repetindo-se os bits de informação e acumulando o resultado da soma modulo-2 de uma permutação destes bits. Assim como os códigos LDPC e LDGM, estes códigos também podem ser irregulares [41], sistemáticos ou não. Eles podem ser vistos como a concatenação de um código LDGM com o acumulador. O acumulador é um código convolucional definido pelas seguintes equações:

$$\begin{aligned} t_n &= i_n \oplus s_{n-1}, \\ o_n &= t_n, \\ s_n &= t_n, \end{aligned} \quad (4.22)$$

onde  $s_n$  é o  $n$ -ésimo estado da memória do acumulador,  $i_n$  e  $o_n$  são os  $n$ -ésimos bits de entrada e saída. A forma como estas equações foram escritas facilitará posteriormente a obtenção da curva EXIT do acumulador.

Como o acumulador é sempre definido pelas mesmas equações (exceto por uma variação de tamanho), a matriz geradora  $\mathbf{G}$  é suficiente para definir um código RA. Os valores de  $d_v$  e  $d_c$  ou  $d_v()$  e  $d_c()$  seguem o mesmo padrão definido para códigos LDGM. Um código RA pode ser genericamente representado pela Fig. 4.4, onde podemos dividir o grafo em três camadas: VND, CND e acumulador. As direções de troca de informação estão mostradas na Fig. 4.5, onde o subscrito  $A$  se refere ao acumulador.

A taxa de um código RA é obtida igualando-se o número de ramos das camadas VND e CND que saem em direção a outra camada. Como o acumulador tem taxa 1, a taxa do código RA é igual a taxa de um código LDGM com os mesmos parâmetros. Ela vale:

$$Tx_{RA}^S = \frac{K}{B} = \frac{\tilde{d}_c}{\tilde{d}_v + \tilde{d}_c} \quad (4.23)$$

para códigos sistemáticos e :

$$Tx_{RA}^{NS} = \frac{K}{B} = \frac{\tilde{d}_c}{\tilde{d}_v} \quad (4.24)$$

para códigos não sistemáticos.

A geração de códigos é simples, bastando gerar as conexões entre o VND e o CND, de acordo com a matriz  $\mathbf{G}$ . A codificação consiste então em passar os bits pelas camadas, do VND para o CND e depois para o acumulador.

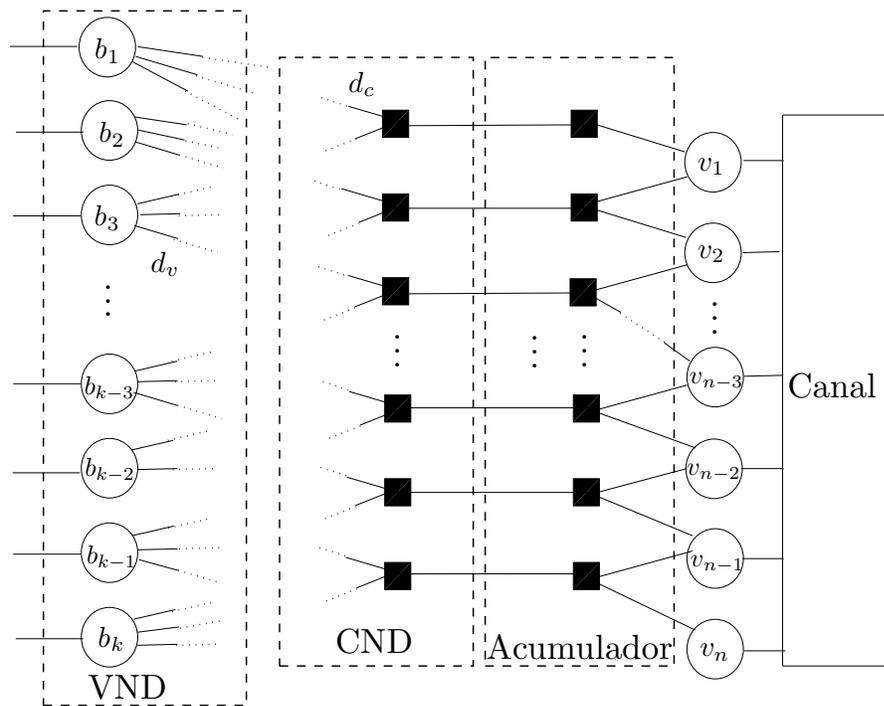


Fig. 4.4: Grafo que representa um código RA não sistemático regular com  $d_v = 3$  e  $d_c = 2$ . Para o caso sistemático, os nós que representam os bits também são transmitidos.

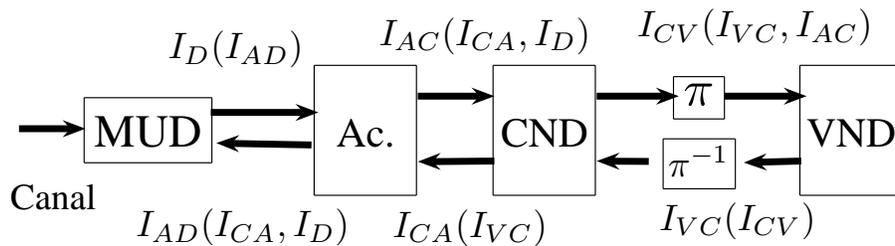


Fig. 4.5: Direção de troca de informações para MUD e RA.

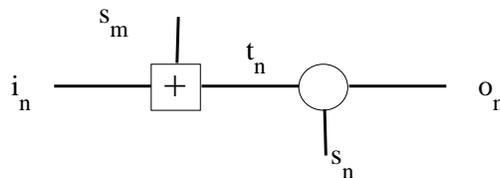


Fig. 4.6: Representação de uma instância do acumulador.

A decodificação inicia-se pelo acumulador. Como ele é um código convolucional de taxa 1, podemos utilizar o algoritmo *Forward-Backward* ou BCJR [42]. O algoritmo soma-produto continua pelo resto do grafo. Pelos mesmos motivos de códigos LDGM, códigos RA não sistemáticos precisam ter uma fração dos nós que compõem o CND com grau 1, ou o processo iterativo não se inicia.

Um código RA sistemático pode ser visto como um código LDPC, com uma fração  $(B - K)/(B)$  de nós-paridade com grau 2 representando o acumulador. Com essa restrição podemos obter códigos LDPC com rápida codificação [43].

As equações EXIT das camadas VND e CND de um código RA regular são iguais às curvas para as camadas correspondentes de um código LDGM, descritas na seção anterior, com pequenas variações devido ao fato da camada CND estar conectada ao acumulador e não ao canal.

As curvas do acumulador são geralmente parametrizadas pela relação sinal-ruído e obtidas por simulação [29]. Um método alternativo que propomos é obter as curvas numericamente utilizando um sistema de equações. Em vez de usar a relação sinal-ruído como parâmetro para a curva do acumulador, utilizamos o valor de informação *a priori* correspondente. Considere a Fig. 4.6, um grafo que representa a  $n$ -ésima entrada do acumulador. Seja  $I_X^A$  e  $I_X^E$  os valores da informação *a priori* e da informação extrínseca da variável  $X$ , respectivamente. Usando as equações (4.5) e (4.11) obtemos o seguinte sistema:

$$\begin{aligned}
 I_{t_n \rightarrow} &= 1 - J(\sqrt{J^{-1}(1 - I_{s_m}^A)^2 + J^{-1}(1 - I_{i_n}^A)^2}), \\
 I_{t_n \leftarrow} &= J(\sqrt{J^{-1}(I_{o_n}^A)^2 + J^{-1}(I_{s_n}^A)^2}), \\
 I_{i_n}^E &= 1 - J(\sqrt{J^{-1}(1 - I_{s_m}^A)^2 + J^{-1}(1 - I_{t_n \leftarrow})^2}), \\
 I_{s_m}^E &= 1 - J(\sqrt{J^{-1}(1 - I_{i_n}^A)^2 + J^{-1}(1 - I_{t_n \leftarrow})^2}), \\
 I_{o_n}^E &= J(\sqrt{J^{-1}(I_{t_n \rightarrow})^2 + J^{-1}(I_{s_n}^A)^2}), \\
 I_{s_n}^E &= J(\sqrt{J^{-1}(I_{o_n}^A)^2 + J^{-1}(I_{t_n \rightarrow})^2}),
 \end{aligned} \tag{4.25}$$

onde a seta indica a direção das mensagens transmitidas de acordo com a Fig. 4.6.

Sob a hipótese de que estamos usando um acumulador grande, é razoável assumir que  $I_{s_m}^E = I_{s_n}^A$  e  $I_{s_n}^E = I_{s_m}^A$ . Dado um valor de  $I_{o_n}^A$  e  $I_{i_n}^A$ , podemos numericamente obter os valores de  $I_{s_n}^A$  e  $I_{s_m}^A$  que satisfazem a condição assumida, o que nos permite posteriormente calcular  $I_{i_n}^E = I_{AC}(I_{CA}, I_D)$  e  $I_{o_n}^E = I_{AD}(I_{CA}, I_D)$ .

## 4.5 Obtenção de curva EXIT global dos códigos

O procedimento tradicionalmente adotado para se analisar um dos códigos acima é analisar a interface entre as camadas VND e CND [28; 29]. Se há outras camadas além destas, as curvas EXIT destas são combinadas com a curva do VND ou CND de acordo com a conexão das camadas.

Alternativamente, pode-se obter uma curva global do código, combinando as curvas das camadas de forma a obter uma única curva para o código, dado um valor de informação *a priori* disponível. O objetivo é separar o detector do código para que vários códigos possam ser analisados. A vantagem deste método é que, uma vez que temos a curva EXIT do detector, podemos determinar qual classe de códigos é mais apropriada para este.

O conceito é uma extensão da idéia utilizada para a obtenção numérica da curva do acumulador. Considere por exemplo a Fig. 4.2. Para combinarmos as curvas do VND e CND, consideramos que, para um dado valor de  $I_D$  fornecido pelo detector, VND e CND vão trocar mensagens até atingirem um ponto estável. Este ponto estável é o menor ponto onde as curvas  $I_{VC}(I_{CV}, I_D)$  e  $I_{CV}(I_{VC})$  se cruzam. Isto acontece no ponto  $(a, b)$  tal que  $I_{VC}(a, I_D) = b$  e  $I_{CV}(b) = a$ , que pode ser numericamente obtido usando as equações descritas acima. Dado este ponto, podemos obter  $I_{VD}(a)$ , o valor de informação *a priori* que o código retorna ao detector. Ao realizarmos esta combinação obtemos uma curva para o maior valor de  $I_{VD}$  que o código pode retornar, dado um valor de  $I_D$ .

O mesmo procedimento pode ser realizado para códigos LDGM, onde obteremos  $I_{CD}(I_D)$ . Para códigos RA, o procedimento deve ser aplicado duas vezes: primeiro para a obtenção de  $I_{CA}(I_{AC})$  (que independe de  $I_D$ ) e posteriormente para a obtenção de  $I_{AD}(I_D)$ .

Apresentamos as curvas para alguns códigos LDPC, LDGM e RA obtidas através deste método nas Figs. 4.7 a 4.9. Os parâmetros dos códigos estão definidos nas Tabs. 4.2 a 4.4. Percebe-se que códigos LDGM não são capazes de retornar um valor de  $I^E = 1$  para  $I^A < 1$ , enquanto que códigos RA e LDPC (dependendo dos parâmetros) conseguem. Além disso, as curvas dos códigos RA são as mais íngremes. Pode-se perceber o impacto da presença do acumulador ao compararmos os códigos LDGM e RA com os mesmos parâmetros.

Para qualquer código, a área sob a curva EXIT é igual a 1 menos a taxa do código [29]. Esta propriedade nos permite avaliar a validade do método proposto. Apresentamos nas Tabs. 4.2 a 4.4 os valores das áreas e as taxas para alguns códigos. Percebe-se que algumas das áreas estão abaixo do valor esperado, principalmente para os códigos RA. Isto pode ser interpretado como um indício de que estas curvas são conservadoras: os códigos retornam um valor de  $I^E$  ligeiramente inferior ao necessário em algum ponto.

Alternativamente, a validade do método pode ser comprovada através da comparação entre as curvas EXIT obtidas pelo método proposto e as curvas EXIT dos códigos obtidas por simulação. As duas curvas estão sobrepostas na Fig. 4.10 para os códigos RA sistemáticos, que utiliza do método três

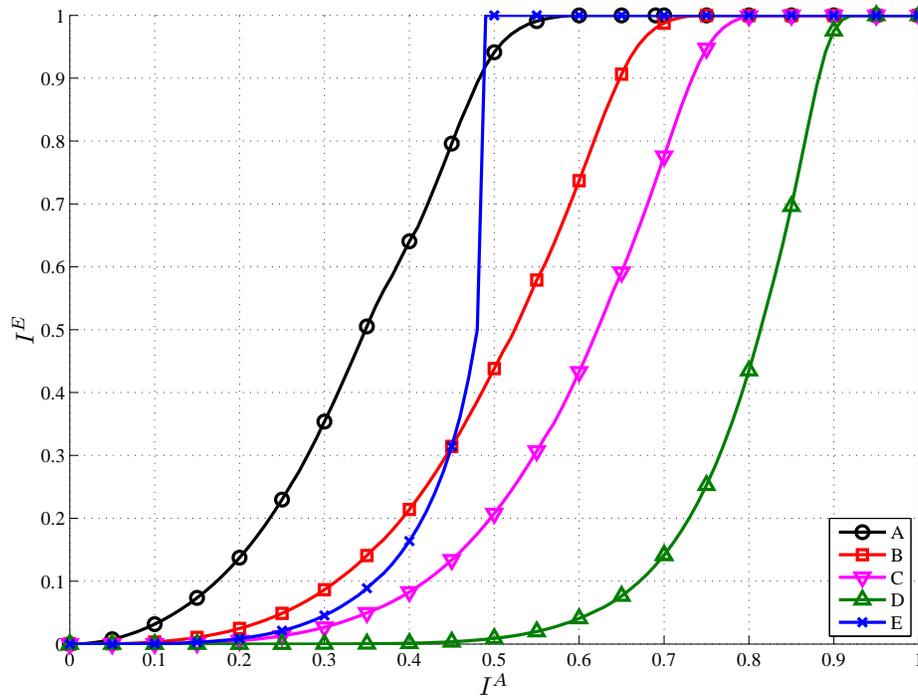


Fig. 4.7: Curvas EXIT para alguns códigos LDPC.

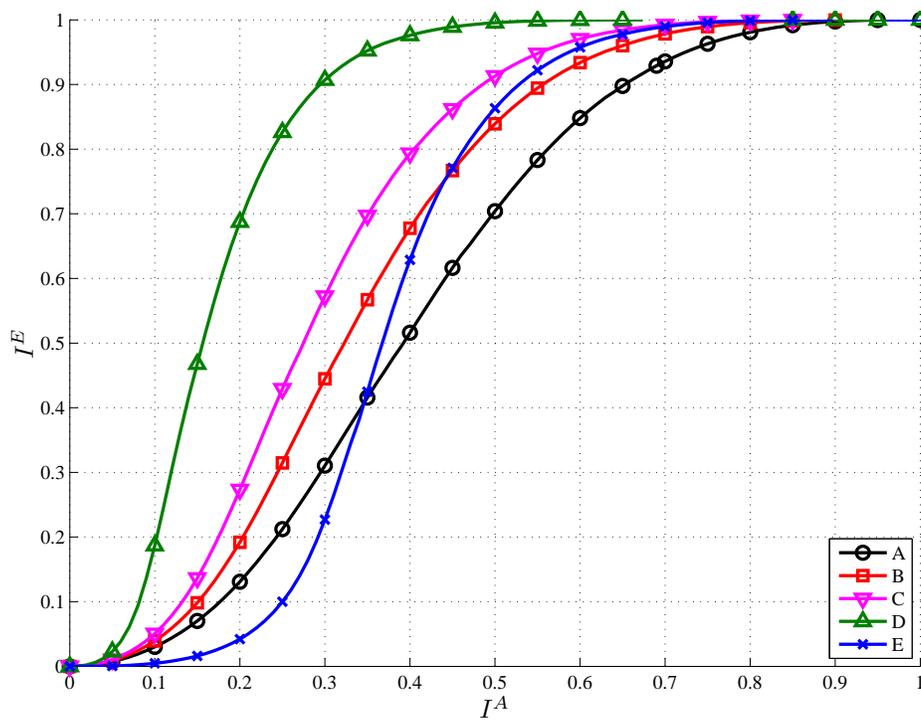


Fig. 4.8: Curvas EXIT para alguns códigos LDGM sistemáticos.

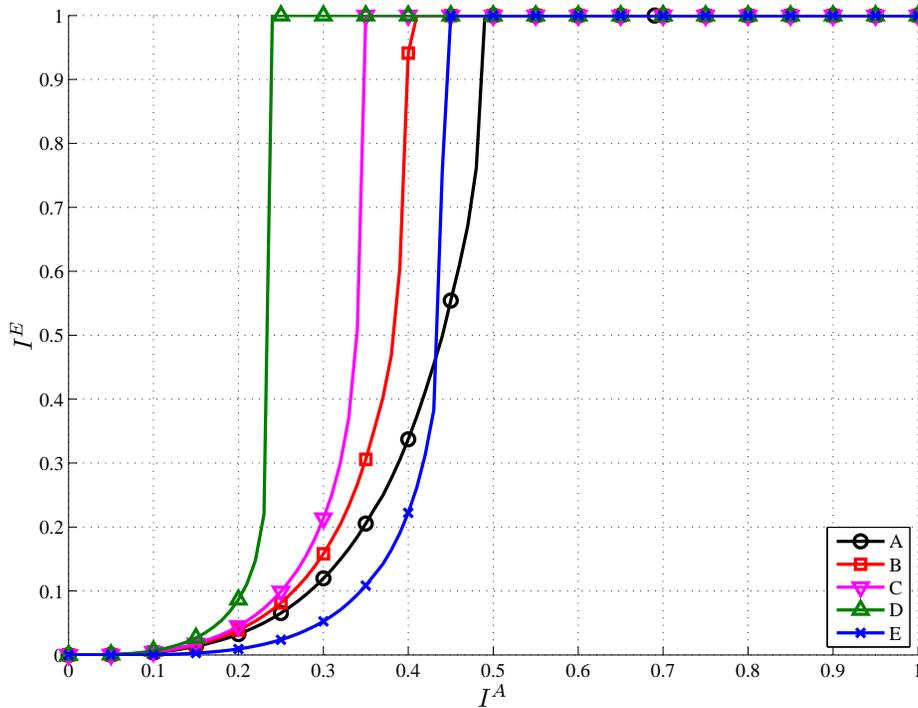


Fig. 4.9: Curvas EXIT para alguns códigos RA sistemáticos.

vezes: uma para obtermos as curvas do acumulador e duas para concatenar as curvas das camadas internas. É de se esperar que, se há alguma diferença, ela será maior nesta situação. As curvas simuladas são as tracejadas. Utilizamos um código com bloco transmitido de  $10^4$  bits e 50 iterações internas do código. Para a curva obtida por simulação utilizamos 21 pontos de  $I^A$ , enquanto que para a curva obtida numericamente utilizamos 51 pontos. Pode-se notar que as curvas estão muito próximas. A diferença entre as curvas é principalmente devida ao número de pontos utilizados para desenhar cada uma delas. Isto nos leva a crer que o método proposto é exato o suficiente e a diferença entre a área e a taxa se deve ao fato de estarmos utilizando equações para o nó-variável adequadas para um canal Gaussiano, e não um canal de apagamento, onde a propriedade da área é verdadeira.

## 4.6 Comparação entre códigos

Os códigos podem ser comparados em quatro aspectos: facilidade de projeto, facilidade de implementação, facilidade de decodificação e desempenho.

Para todos os códigos analisados, a análise depende dos parâmetros  $d_v(i)$  e  $d_c(i)$ . As equações das camadas VND e CND tem a mesma complexidade para todos os códigos. Códigos RA tem, além destas camadas, a camada do acumulador, o que torna mais complicado o processo de obtenção

da curva EXIT do código completo. Entretanto, esta complexidade adicional não torna o processo inviável.

A facilidade de implementação dos códigos varia. Para códigos LDGM, basta gerar a matriz geradora  $\mathbf{G}$  de acordo com o tamanho e parâmetros escolhidos. Esta matriz é suficiente tanto para gerar a palavra-código como para gerar o grafo a ser utilizado para decodificação. A implementação do acumulador para os códigos RA é simples, dada a estrutura do acumulador, mas é um adicional de complexidade. Códigos LDPC, por outro lado, são mais complicados de serem implementados. Embora a matriz  $\mathbf{H}$  de um código LDPC seja esparsa, nada garante, a princípio, que a matriz geradora do código LDPC também será. Pode-se obter a matriz geradora correspondente a partir da matriz  $\mathbf{H}$  através da equação  $\mathbf{G} \cdot \mathbf{H} = 0$ , mas esta operação pode se tornar proibitiva para códigos grandes. Uma solução proposta foi a de utilizar códigos sistemáticos

A complexidade de decodificação depende do número de nós de cada tipo, do número de ramos em cada nó e da quantidade de iterações necessárias. Estes números podem ser vistos na Tab. 4.5 em função do tamanho do bloco transmitido  $B$ , da taxa do código  $Tx$  e do número médio de ramos da camada  $CND \tilde{d}_c$ . Pode-se perceber que códigos LDGM tem a menor complexidade dos três, seguido pelos códigos LDPC e por último os códigos RA. Além disso, códigos sistemáticos são mais simples

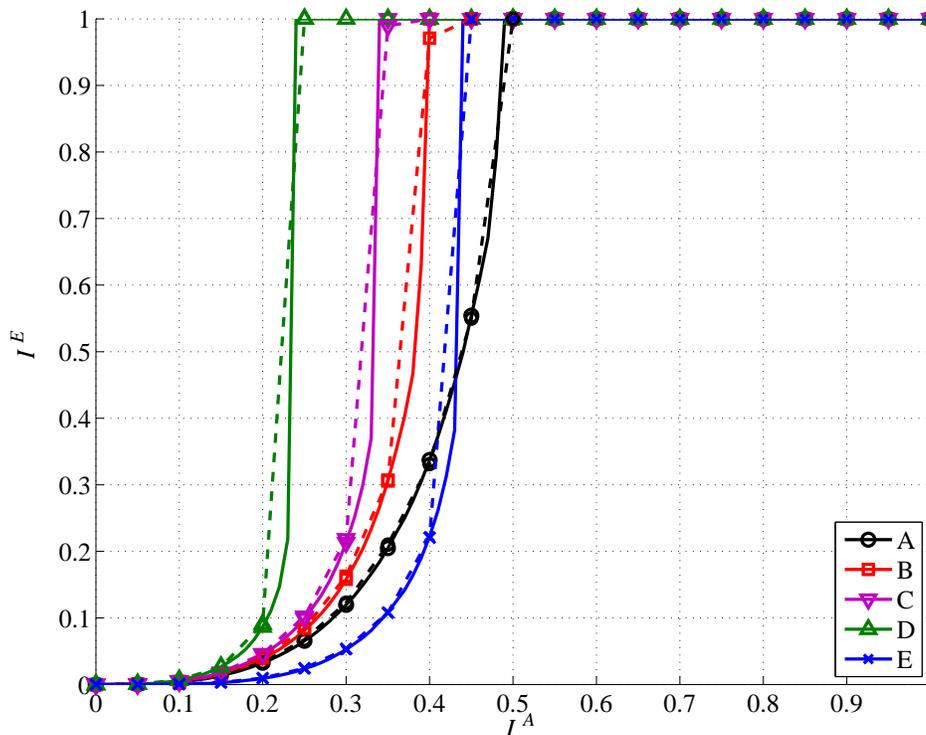


Fig. 4.10: Curvas EXIT para códigos RA sistemáticos obtidas por simulação.

Tab. 4.2: Comparação entre área e taxas de códigos LDPC

Códigos	$d_v$	$d_c$	Taxa	1-Área
A	2	3	0,333	0,3396
B	2	4	0,5	0,5006
C	2	5	0,6	0,6036
D	2	10	0,8	0,7924
E	3	5	0,4	0,4455

Tab. 4.3: Comparação entre área e taxas de códigos LDGM sistemáticos

Código	$d_c$	$d_v$	Taxa	1-Área
A	2	3	0,4	0,4055
B	2	4	0,333	0,3401
C	2	5	0,2857	0,2936
D	2	10	0,1667	0,1745
E	3	5	0,375	0,379

Tab. 4.4: Comparação entre área e taxas de códigos RA sistemáticos

Código	$d_c$	$d_v$	Taxa	1-Área
A	2	3	0,4	0,4099
B	2	4	0,333	0,3532
C	2	5	0,2857	0,3136
D	2	10	0,1667	0,2236
E	3	5	0,375	0,4081

Tab. 4.5: Valores que definem a complexidade de decodificação

Código		Nós de paridade	Nós-variável	Total de ramos
LDPC		$B(1 - Tx)$	$B$	$B[1 + (1 - Tx)\tilde{d}_c]$
LDGM	Não sistemático	$B$	$B \cdot Tx$	$B(\tilde{d}_c + 1)$
	Sistemático	$B(1 - Tx)$	$B \cdot Tx$	$B[1 + \tilde{d}_c(1 + Tx)]$
RA	Não sistemático	$2B$	$B \cdot Tx$	$B(\tilde{d}_c + 4)$
	Sistemático	$2B(1 - Tx)$	$B(1 + Tx)$	$B(1 - Tx)(\tilde{d}_c + 4)$

do que códigos não sistemáticos.

O desempenho dos códigos depende do sistema onde eles serão utilizados. Observando as curvas EXIT do detector, percebe-se que o detector nunca retorna um valor de informação extrínseca igual a um, mesmo que a entrada forneça um valor de informação *a priori* igual a um. Desta forma, para este sistema, é essencial que o código seja capaz de fornecer um valor de  $I^E = 1$  para algum valor de  $I^A < 1$ . Observando as curvas EXIT dos códigos apresentados na seção anterior, percebe-se que os códigos LDPC e RA são capazes de atender esta condição, enquanto que os códigos LDGM não.

Sendo necessário escolher uma das classes para o projeto do sistema, escolhemos os códigos RA sistemáticos pelos seguintes motivos:

- Dentre os códigos, somente as categorias LDPC e RA geram curvas EXIT adequadas pois retornam  $I^E = 1$  para algum valor de  $I^A < 1$ );
- Códigos RA tem implementação mais simples do que códigos LDPC pois são definidos por uma matriz geradora em vez de uma matriz de verificação de paridade;
- Códigos RA sistemáticos tem complexidade menor do que códigos RA não sistemáticos.

## 4.7 Resumo

Neste capítulo mostramos como algumas classes de códigos de bloco podem ser modeladas como grafo de fatores. Duas equações que relacionam o valor da informação *a priori* com a informação extrínseca para os nós fundamentais dos grafos que representam os códigos foram mostradas. Através destas equações consegue-se obter curvas EXIT para cada uma das camadas que compõem os grafos dos códigos analisados, sejam elas regulares ou não. A curva EXIT do acumulador que integra os códigos RA foi obtida numericamente, o que evitou a necessidade de simulação do acumulador.

Uma extensão do método utilizado para obter a curva do acumulador permite obter as curvas EXIT globais dos códigos. A principal idéia deste método é que as camadas do código podem, dado um valor de informação *a priori*  $I_D$  externo, trocar informação entre si até atingirem um ponto de equilíbrio. A resposta do código, neste ponto de equilíbrio, é o valor da curva EXIT global do código para o valor de  $I_D$  fornecido a priori.

A escolha do código a ser utilizado no projeto se baseou na facilidade de projeto, facilidade de implementação, complexidade de decodificação e desempenho esperado, tendo em vista o sistema onde o código seria usado. Com base nas características dos códigos e nas curvas EXIT globais, os códigos RA são a melhor escolha para o projeto do sistema multiusuário.

# Capítulo 5

## Projeto de Sistemas

Neste capítulo vamos utilizar os cálculos e ferramentas apresentadas nos capítulos anteriores para projetar um sistema multiusuário que utiliza o canal M-FSK ruidoso descrito no Capítulo 2. O objetivo é atingir taxas próximas a capacidade do canal utilizando os códigos apresentados no Capítulo 4. O sistema será dividido em duas partes: o detector multiusuário apresentado no Capítulo 3 e nos decodificadores dos usuários.

Utilizaremos as curvas EXIT das partes do sistema para prever limites de funcionamento em função da razão sinal ruído. Este método nos diz que o sistema terá baixa probabilidade de erro se houver um túnel entre as curvas EXIT das partes envolvidas. A curva EXIT do detector para uma dada relação sinal-ruído é fixa, pois o seu grafo é inalterável. As curvas dos códigos são manipuláveis através dos parâmetros que os definem, isto é, a distribuição dos graus.

Neste contexto, projetar o código é maximizar a taxa do código, manipulando as distribuições de graus das camadas, sujeito à restrição de que haja um túnel entre as curvas EXIT do código e do detector.

Consideramos o esquema de detecção paralela, onde todos os usuários são detectados ao mesmo tempo. O projeto é feito com base em uma única curva EXIT do detector, o que faz com que todos os usuários tenham a mesma taxa.

A validade deste método pode ser confirmada através de simulações de sistemas para estimar a probabilidade de erro de bit (BER, do inglês: *Bit Error Rate*) em torno do ponto de projeto, que conclui este capítulo.

### 5.1 Otimização das curvas EXIT dos códigos

No sistema iterativo utilizado, a informação extrínseca de um módulo se torna a informação a priori do outro. Isto quer dizer que a análise é feita com a curva EXIT de um dos módulos desenhada

com os eixos trocados, como mostra a figura 5.1. A área abaixo da curva EXIT do detector deve ser igual à capacidade do sistema. Com os eixos trocados, a área abaixo da curva EXIT do decodificador é igual à taxa do código. Desta forma, manter a curva do decodificador abaixo da curva do detector é equivalente a dizer que a taxa de transmissão do código utilizado é abaixo da capacidade do sistema como mostrado em [34], condição necessária para a correta transmissão de informação. A área entre a curva do detector e a curva do decodificador é a diferença entre a taxa de transmissão e a capacidade do sistema.

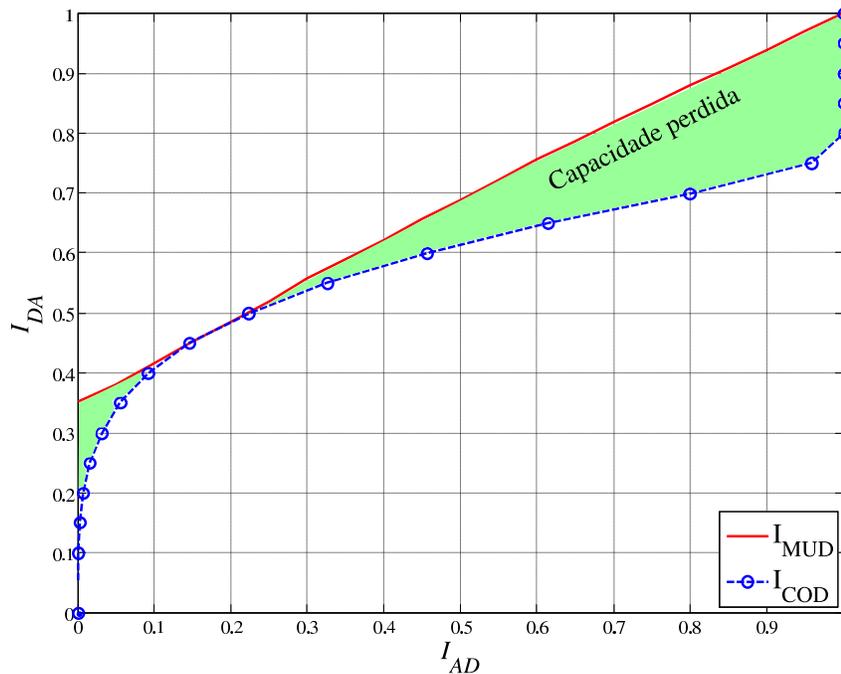


Fig. 5.1: Capacidade perdida devido a falta de ajuste das curvas.

O projeto consiste em adequar a curva do código à curva do detector. Idealmente as duas curvas coincidiriam em todos os pontos. Desta forma, um pequeno aumento na relação sinal-ruído elevaria a curva EXIT do detector suficientemente para que haja um túnel entre as duas curvas. O problema pode ser definido de duas formas:

- maximizar a taxa do código, sujeito à restrição de que a curva do código deve ficar abaixo da curva do detector
- minimizar a área entre as duas curvas, indiretamente aumentando a taxa do código.

Não obtivemos equações analíticas para a definição das curvas EXIT do detector em função de  $N$ ,  $T$  e  $E_t/N_0$  ou código em função dos seus parâmetros, de forma que a área entre as duas curvas

precisaria ser calculada numericamente. Por outro lado, as taxas dos códigos em função da distribuição de graus é rapidamente calculada através das equações apresentadas no capítulo anterior. Além disso, as derivadas parciais da taxa em função da distribuição de graus também possui uma forma analítica, o que permite o cálculo eficiente do gradiente e Hessiano da função objetivo (taxa). Isto permite utilizar os algoritmos de otimização não linear prontamente disponíveis em programas como MATLAB, o que torna a primeira alternativa de definição do problema uma escolha apropriada.

Seja  $I_{COD}(x)$  a curva EXIT do código e  $I_{MUD}(x)$  a curva EXIT do detector. A restrição de que a curva do código fique sempre abaixo da curva do detector pode ser escrita como:

$$I_{COD}(I_{MUD}(x)) \geq x, \quad \forall 0 \leq x \leq 1. \quad (5.1)$$

Não é possível garantir que a restrição  $I_{COD}(I_{MUD}(x)) \geq x$  será satisfeita para todo  $x$  pois há infinitos valores de  $x$  e não obtivemos formas analíticas para as funções  $I_{COD}(x)$  e  $I_{MUD}(x)$ . Este problema pode ser contornado de duas formas: aproximar as funções  $I_{COD}(x)$  e  $I_{MUD}(x)$  ou escolher um número finito de valores de  $x$  entre 0 e 1 e testar se a condição é satisfeita. A segunda opção foi escolhida por ser mais simples de implementar, sendo que 31 valores equidistantes de  $x$  entre 0 e 1 foram utilizados. A restrição (5.1) pode ser substituída por uma função  $\Delta(\lambda)$  definida como:

$$\Delta(d_v(i), d_c(i), I_{MUD}(x)) = \min_x [I_{COD}(I_{MUD}(x)) - x], \quad (5.2)$$

onde  $x$  pertence ao conjunto de pontos utilizados.

A formulação do problema para maximizar a taxa  $R(d_v(i), d_c(i))$  do código pode então ser dada por:

$$\begin{aligned} \max R(d_v(i), d_c(i)), \\ \text{sujeito a} \\ \Delta(d_v(i), d_c(i), I_{MUD}(x)) \geq 0. \end{aligned} \quad (5.3)$$

onde a maximização é feita manipulando a distribuição de graus  $d_v(i)$  e  $d_c(i)$ .

Dado que estamos utilizando códigos RA sistemáticos, qualquer combinação de distribuição de graus é possível. Na prática é necessário limitar os valores de  $d_c^{MAX}$  e  $d_v^{MAX}$ , os maiores valores de  $i$  tais que  $d_c(i) > 0$  e  $d_v(i) > 0$  respectivamente. Escolhemos  $d_c^{MAX} = 12$  e  $d_v^{MAX} = 8$ . Além disso, restringimos os valores de  $d_c^{MIN}$  e  $d_v^{MIN}$  (os menores valores de  $i$  tais que  $d_c(i) > 0$  e  $d_v(i) > 0$ ) em 1 e 3, respectivamente. Para reduzir a complexidade da otimização, limitamos os graus possíveis dos nós de paridade a 1, 2, 3, 5, 8 e 12. Isto permite obter códigos com taxas entre 0, 11 e 0, 8, o que engloba as capacidades normalizadas das combinações de  $N$ ,  $T$  e  $E_t/N_0$  analisados (apresentados nas Tabs. A.4 a A.6 do apêndice).

As curvas que obtivemos para o detector e para os códigos não são exatamente iguais à capacidade e a taxa, respectivamente. No caso do detector, o processo numérico utilizado gerou curvas que estão em torno da capacidade. No caso do código, as curvas indicam uma taxa ligeiramente menor do que a taxa do código. O resultado é que, ao utilizarmos estas curvas, obteremos aproximações para o projeto do sistema. A validade destas aproximações pode ser testada ao simularmos a probabilidade de erro de bit do sistema: se a probabilidade de erro for baixa para valores de  $E_t/N_0$  próximos do valor de projeto, as aproximações são boas.

## 5.2 Construção dos códigos

Para construir um código RA sistemático, precisamos saber como os bits da palavra código se relacionam com os bits de informação. Esta relação pode ser dada pela matriz geradora  $\mathbf{G}$  ou pela conexão equivalente dos nós das camadas VND e CND de um grafo de fatores que representa o código.

As conexões podem ser geradas a partir do tamanho do bloco transmitido  $B$  e dos parâmetros  $d_v(i)$  e  $d_c(i)$ . O número de nós da camada VND é  $K$ , que é o número de bits de informação transmitidos por palavra-código, enquanto que o número de nós da camada CND e o tamanho do acumulador valem  $B - K$ . Idealmente o número total de ramos  $\rho$  que ligam as camadas VND a CND deveria ser:

$$\rho = K \sum_i i \cdot d_v(i) = (B - K) \sum_i i \cdot d_c(i), \quad (5.4)$$

Entretanto, as distribuições  $d_v(i)$  e  $d_c(i)$  ótimas não necessariamente resultam num valor inteiro quando multiplicado por  $K$  ou  $B - K$ , respectivamente. Como o número de ramos deve ser inteiro, o valor de cada termo da somatória é aproximado para o inteiro mais próximo. Mesmo com estas aproximações, o resultado dos dois somatórios da equação acima deve coincidir, pois todos os ramos que saem da camada VND devem estar conectados aos ramos que saem da camada CND. Caso isto não aconteça devido às aproximações, ramos são adicionados aleatoriamente a algum nó (de VND ou CND, dependendo do caso), até que esta condição esteja satisfeita.

Dado o número de nós de cada camada e a quantidade de ramos de cada um dos nós, os ramos podem ser conectados aleatoriamente. Para isso, numeramos todos os ramos de 1 a  $\rho$ . Os nós da camada VND são conectados aos ramos de forma sequencial. Por exemplo, se o código fosse regular com parâmetro  $d_v = 3$ , o primeiro nó de VND seria conectado aos ramos 1, 2 e 3; o segundo aos ramos 4, 5 e 6; e assim por diante até o último nó. Os nós da camada CND são conectados sequencialmente com uma função  $\pi(i)$  dos ramos, que é uma função bijetora cujo domínio e imagem são os números inteiros de 1 a  $\rho$ . Assim, se o código fosse regular com parâmetro  $d_c = 3$ , por exemplo, o primeiro

nó da camada VND estaria conectada aos ramos  $\pi(1)$ ,  $\pi(2)$  e  $\pi(3)$ ; o segundo aos ramos  $\pi(4)$ ,  $\pi(5)$  e  $\pi(6)$ ; e assim por diante até o último nó. Este processo pode ser visualizado graficamente na Fig. 5.2.

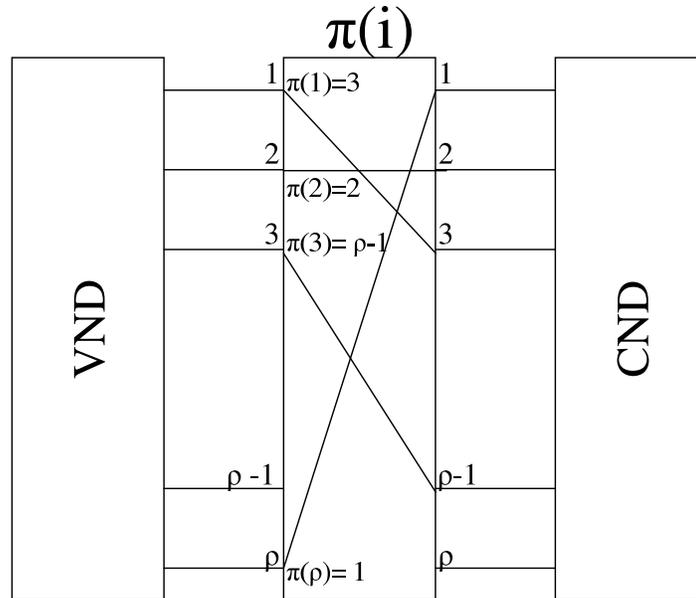


Fig. 5.2: Ligação de ramos através da função  $\pi(i)$ .

A função  $\pi(i)$  pode ser obtida através de um processo aleatório que forneça um vetor de dimensão  $\rho$  que contém todos os valores inteiros de 1 a  $\rho$ . Esta função, juntamente com o acumulador, é que define como as sequências de informação estão relacionadas com as palavras-código. Cada usuário utiliza um conjunto de  $2^K$  palavras-código dentro de um conjunto de  $2^B$  palavras possíveis. Se os conjuntos fossem os mesmos para pelo menos dois usuários, não seria possível distinguir qual deles transmitiu qual mensagem. Para evitar estas situações, a função  $\pi(i)$  deve ser única para cada usuário. Isto não garante que os conjuntos de palavras-código dos usuários serão distintos. Gerando as funções  $\pi(i)$  de forma aleatória, a probabilidade de uma palavra-código ser compartilhada por dois usuários vale  $2^{K-B}$ , que se reduz rapidamente com um aumento no tamanho do bloco transmitido, mantendo-se a taxa.

Deve-se ter o cuidado ao gerar  $\pi(i)$  para que, com isso, o mesmo nó da camada CND não esteja conectado ao mesmo nó da camada VND por mais de um ramo, pois isto faria com que o grafo do código tenha um ciclo de baixa ordem, o que é detrimental para o desempenho do sistema.

Através de  $\pi(i)$ , a implementação do código pode ser feita por vetores que armazenam as mensagens que passam da camada VND para CND e vice-versa. A camada VND escreve/lê nestes vetores de forma sequencial, enquanto que a camada CND lê/escreve nestes vetores utilizando  $\pi(i)$ , com  $i$  variando sequencialmente de 1 a  $\rho$ .

### 5.3 Simulação de probabilidade de erro de bit

O sistema completo a ser simulado está mostrado na Fig. 5.3. Para simular a probabilidade de erro de bit, é necessário:

- gerar as mensagens binárias dos usuários,
- obter as palavras-código a partir das mensagens,
- obter o sinal transmitido a partir das palavras código,
- obter o sinal recebido de acordo com as *pdf*'s que definem o canal,
- recuperar os bits de informação a partir do sinal recebido utilizando o sistema iterativo,
- comparar os bits recebidos com os bits transmitidos para identificar os erros de transmissão.

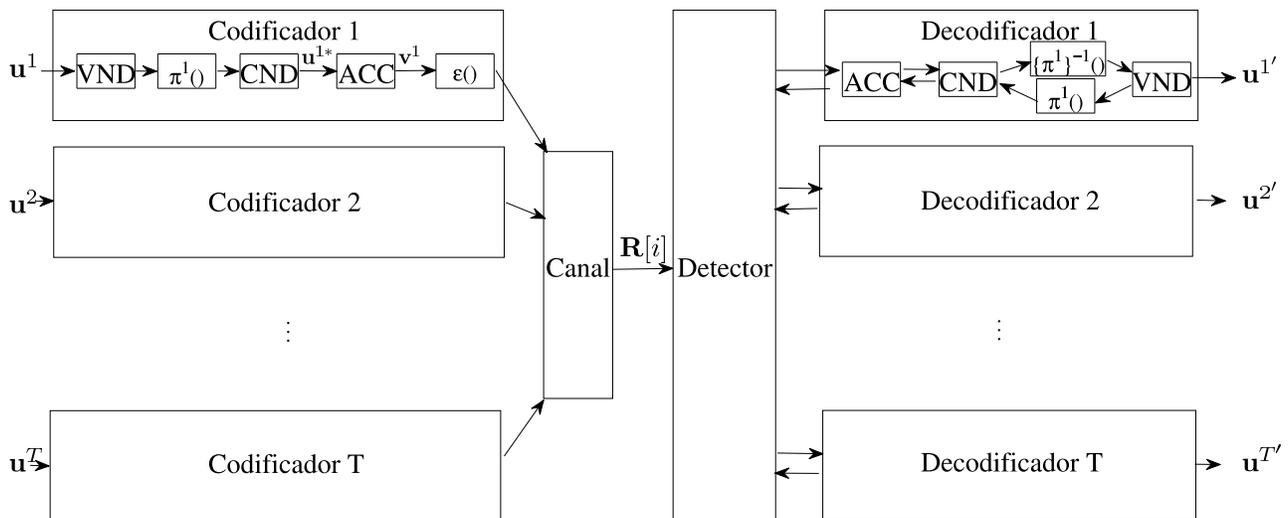


Fig. 5.3: Sistema multiusuário completo, detalhado para o primeiro usuário.

Cada um dos usuários deseja transmitir uma mensagem  $\mathbf{u}^t$ , que é um vetor binário de comprimento  $K$ . Como a interferência causada entre os usuários depende das palavras a serem transmitidas, as mensagens devem ser geradas de forma independente para cada usuário e devem ser trocadas a cada palavra-código transmitida. Isto faz com que o resultado obtido seja válido para a média dos casos de interferência, e não para um caso em particular.

A palavra-código  $\mathbf{v}^t$  a ser transmitida pelo  $t$ -ésimo usuário depende da mensagem  $\mathbf{u}^t$  que este deseja transmitir e da matriz geradora  $\mathbf{G}^t$  inerente ao código RA. Dada uma função  $\pi^t()$  que conecta

a camada VND à camada CND, o termo  $(x, y)$  da matriz  $\mathbf{G}^t$  vale 1 se o  $x$ -ésimo nó da camada VND estiver conectado ao  $y$ -ésimo nó da camada CND através de  $\pi^t()$ . A palavra código pode ser obtida com dois passos. No primeiro, os bits de informação geram uma palavra intermediária  $\mathbf{u}^{t*}$  através da matriz geradora  $\mathbf{G}^t$ . No segundo passo a palavra intermediária é processada pelo acumulador, gerando a palavra-código binária  $\mathbf{v}^t$ . Matematicamente, a relação entre estes três vetores é:

$$\begin{aligned} \mathbf{u}^{t*} &= \mathbf{u}^t \cdot \mathbf{G}^t \\ v_1^t &= u_1^{t*}, \\ v_j^t &= u_j^{t*} \oplus v_{j-1}^t. \end{aligned} \quad (5.5)$$

A palavra-código  $\mathbf{v}^t$  é binária, mas os usuários transmitem sequências de símbolos  $N$ -ários. Para uma sequência de  $B$  bits, são necessários  $B/k$  símbolos, onde  $k$  vale  $\log_2 N$ . Como os símbolos do canal são ortogonais entre si, qualquer mapeamento bijetor entre bits e símbolos produzirá os mesmos resultados. O mapeamento escolhido é a função  $\epsilon()$  apresentada na seção 3.4.1. A ação deste mapeamento, atuando a cada  $k$  bits, faz com que o símbolo  $m^t[i]$  transmitido pelo  $t$ -ésimo usuário na  $i$ -ésima utilização do canal seja igual a:

$$m^t[i] = \sum_{l=0}^{k-1} 2^l v_{(i-1)k+l+1}^t, \quad (5.6)$$

onde  $i$  vai de 1 a  $B/k$ .

A relação entre as mensagens transmitidas pelos usuário por utilização de canal e os vetores de sinais recebidos é obtida utilizando as mesmas equações da seção 2.1, incluindo às variáveis um índice  $[i]$  que indica o instante de utilização do canal. O conjunto de mensagens transmitidas por todos os usuários na  $i$ -ésima utilização do canal vale  $\mathbf{m}[i] = \{m^1[i], m^2[i], \dots, m^T[i]\}$ . A partir deste vetor obtém-se deterministicamente o valor do vetor  $\mathbf{c}[i]$ . Os valores de  $\mathbf{R}[i]$  são gerados de forma aleatória respeitando a distribuição exponencial que os caracteriza, com parâmetros que dependem de  $\mathbf{c}[i]$ .

O processo iterativo se inicia pelo detector multiusuário, que utiliza os valores  $\mathbf{R}[i]$  recebidos. Para cada utilização de canal há uma instância do grafo da Fig. 3.9. Como, a princípio, não se sabe nada a respeito dos bits transmitidos, o valor da informação *a priori* passada pelos decodificadores ao detector vale zero, o que é equivalente a dizer que as mensagens LLR's passadas também valem zero. Uma vez que os valores são processados pela instância do grafo (como descrito no capítulo 3), há mensagens disponíveis para os decodificadores de todos os usuários. Entretanto, para que este decodificadores possam realizar o seu processamento corretamente, é necessário que todas as instâncias do detector multiusuário tenham gerado mensagens, ou seja, todos os valores de  $\mathbf{R}[i]$  devem ser recebidos e processados pelo detector.

Os decodificadores realizam o seu processamento como descrito no capítulo 4. Os decodificadores processam de forma paralela entre si, gerando novas mensagens para o detector. Mensagens são trocadas entre o detector e os decodificadores até que um critério de parada seja atingido. Dois critérios foram utilizados. No primeiro, o receptor interrompe o processamento se um número de iterações máximo foi atingido e os decodificadores decidem por uma mensagem  $\mathbf{u}^{t'}$ . O segundo critério é semelhante ao apresentado em [44]: um decodificador interrompe o processamento se a decisão sobre os bits de informação se mantiver constante por pelo menos 5 iterações internas consecutivas. Caso isto ocorra, o decodificador escolhe pela mensagem  $\mathbf{u}^{t'}$ . Após a parada, o decodificador “re-codifica”  $\mathbf{u}^{t'}$  e transmite ao detector multiusuário mensagens LLR com alto valor absoluto e sinal de acordo com o bit transmitido. Se todos os decodificadores tiverem satisfeito o segundo critério, o processamento do sistema se encerra para este conjunto de palavras transmitidas.

O número de erros de recepção é obtido pela comparação entre a mensagem  $\mathbf{u}^{t'}$  e  $\mathbf{u}^t$ , para todos os usuários. A probabilidade de erro de bit simulada é obtida pela razão entre o número de erros e o número de bits de informação transmitidos. A precisão depende do número de amostras utilizadas. Seja  $BER_i$  a estimativa de probabilidade de erro de bit para a  $i$ -ésima palavra transmitida e  $\overline{BER}_i = \sum_{j=1}^i BER_j / i$  a estimativa de probabilidade de erro de bit após a simulação de  $i$  palavras. Se  $BER$  é a probabilidade de erro de bit do sistema, tem-se, pela desigualdade de Chebyshev [45] que:

$$P(|\overline{BER}_i - BER| \geq \varpi) \leq \frac{var(\overline{BER}_i)}{i\varpi^2}, \quad (5.7)$$

onde  $var(\overline{BER}_i)$  é a variância de  $BER_i$  e  $\varpi$  é um valor arbitrariamente pequeno. Esta expressão relaciona a probabilidade de  $\overline{BER}$  estar dentro de uma faixa de  $\pm\varpi$  de  $BER$  com o número  $i$  de observações. Para as simulações foi estabelecido que  $\overline{BER}_i$  deve estar numa faixa de  $\pm 2\%$  de  $BER$  com probabilidade (confiabilidade) de 96%, o que definiu o número de palavras simuladas necessárias. Além disso, restringimos o número de bits transmitidos em  $10^7$ , para evitar simulações excessivamente longas. Esta restrição fez com que valores baixos da probabilidade de erro de bit precisassem de uma faixa relativamente maior (mas nominalmente baixas) para manter a confiabilidade de 96%.

## 5.4 Resultados

Resultados da otimização para algumas combinações de  $N$ ,  $T$  e  $E_t/N_0$  de projeto igual a  $(E_t/N_0)^*$  estão na Tab. 5.1 e graficamente nas Figs. 5.4 a 5.7. As linhas cheias são as curvas EXIT do detector e as linhas tracejadas são as curvas EXIT do código RA sistemático otimizado. A área entre a curva do detector ( $I_{MUD}$ ) e a curva do código ( $I_{COD}$ ) é uma perda de taxa em relação à capacidade do canal não recuperável.

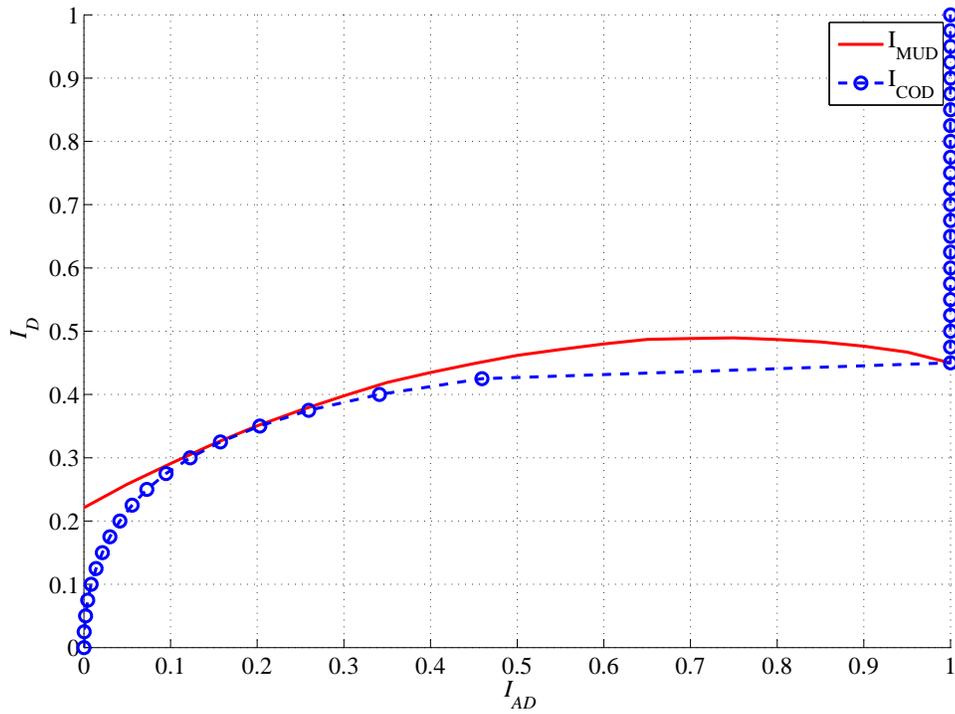


Fig. 5.4: Ajuste de curvas EXIT de códigos RA sistemáticos para  $N = 4, T = 2, (E_t/N_0)^* = 5dB$ .

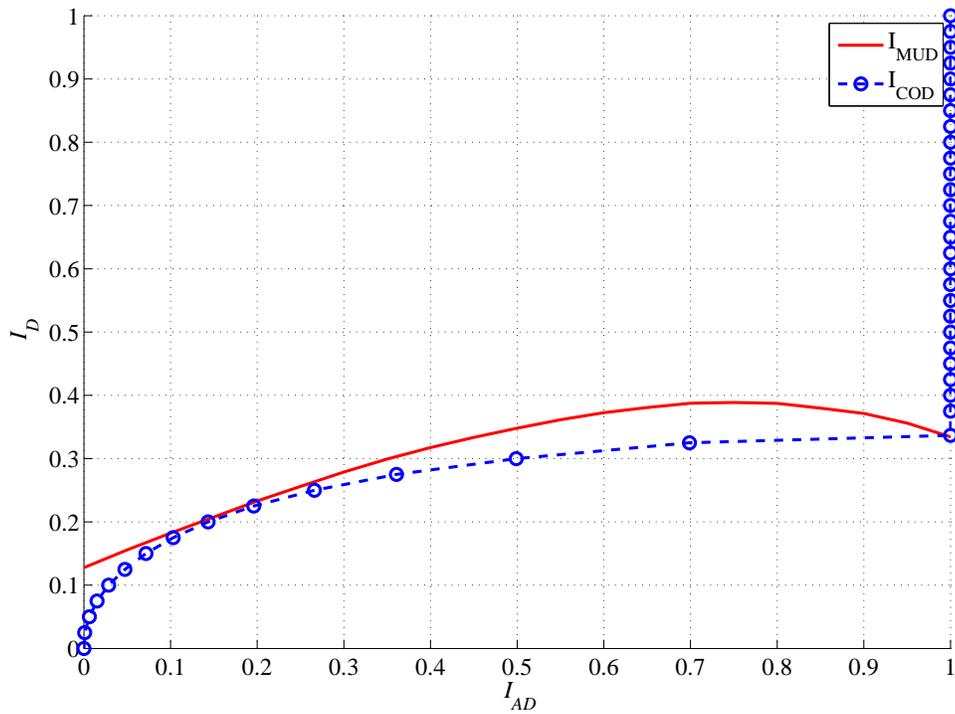


Fig. 5.5: Ajuste de curvas EXIT de códigos RA sistemáticos para  $N = 4, T = 3, (E_t/N_0)^* = 5dB$ .

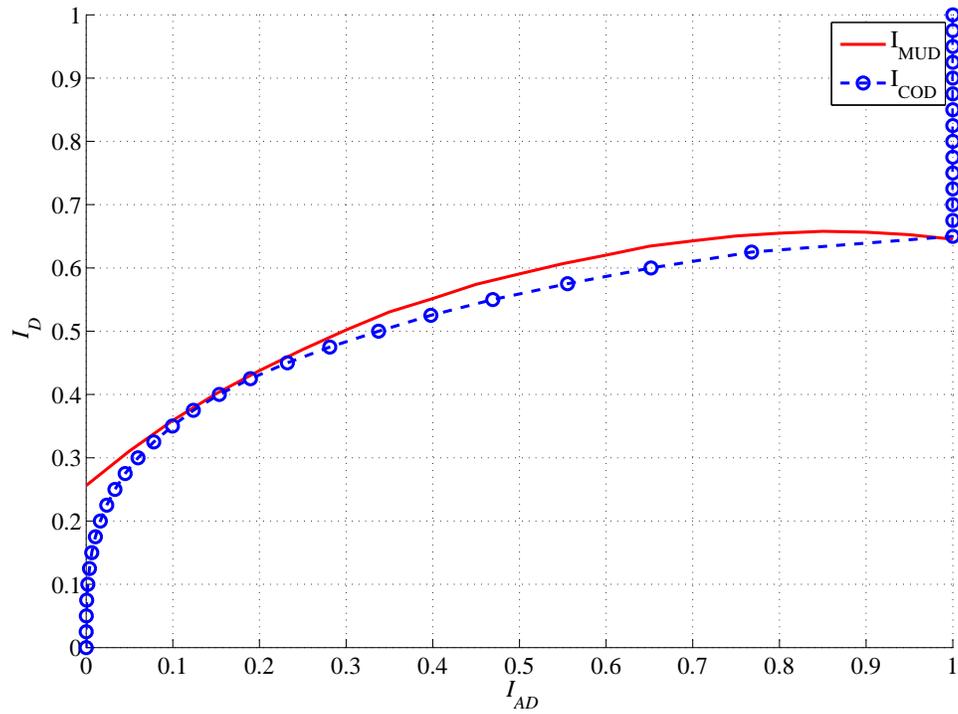


Fig. 5.6: Ajuste de curvas EXIT de códigos RA sistemáticos para  $N = 8$ ,  $T = 2$ ,  $(E_t/N_0)^* = 5dB$ .

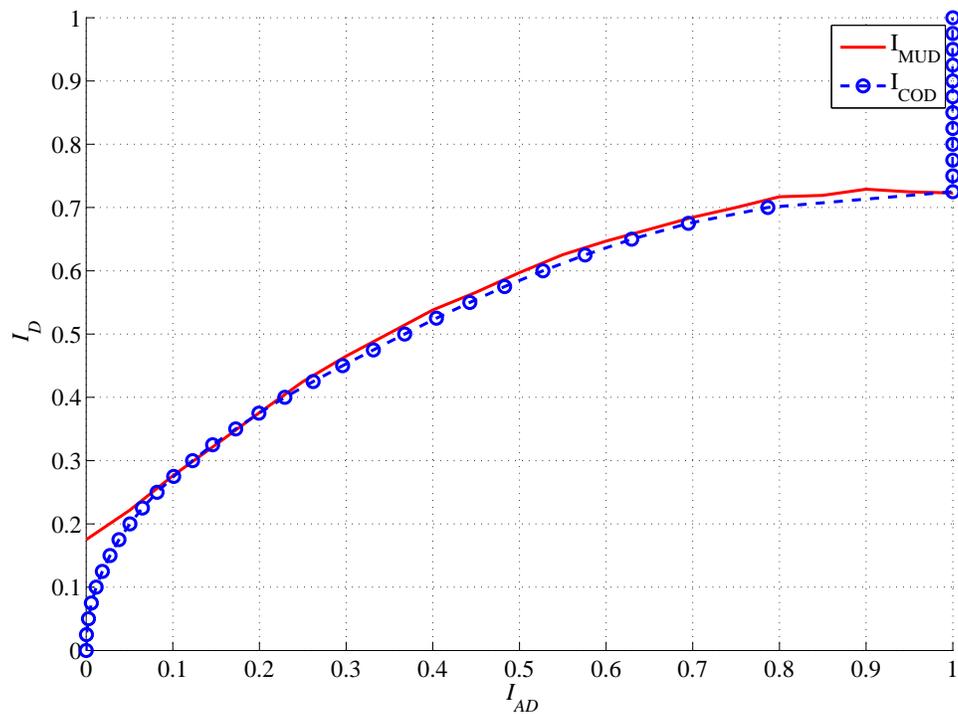


Fig. 5.7: Ajuste de curvas EXIT de códigos RA sistemáticos para  $N = 16$ ,  $T = 3$ ,  $(E_t/N_0)^* = 5dB$ .

Tab. 5.1: Parâmetros de códigos otimizados

Código	N	T	$(E_t/N_0)^*$	Taxa	Distr. de $d_v$			Distr. de $d_c$			
					3	4	6	1	2	3	12
A	4	2	5	0,3711	0	1	0	0,32	0	0,68	0
B	4	3	5	0,2509	0,9992	0	0,0008	0,9868	0	0	0,0132
C	8	2	5	0,5156	0,9	0,1	0	0	0,6	0,3	0,1
D	16	3	5	0,5267	0,9	0,1	0	0,55	0,25	0	0,2

O desempenho dos códigos está na Tab. 5.2. A tabela indica:

- o código analisado (vide Tab. 5.1);
- a taxa para a classe de códigos RA sistemáticos encontrados;
- a capacidade normalizada do canal para os valores de  $N$ ,  $T$  e  $(E_t/N_0)^*$  utilizados;
- o ponto MUD, que é o valor de  $E_t/N_0$  onde a capacidade soma MUD do canal é igual à taxa do código encontrado;
- a distância em  $dB$  entre o ponto MUD e o ponto de projeto;
- o ponto de operação do código, que é o valor de  $E_t/N_0$  necessário para que a probabilidade de erro de bit simulada seja inferior a  $10^{-5}$ , para o código com bloco de tamanho  $10^5$  bits;
- a distância em  $dB$  do ponto de operação para o ponto de capacidade;
- o ponto SUD, que é o valor de  $E_t/N_0$  necessário para que a capacidade soma SUD do canal seja igual à taxa do código encontrado.

Tab. 5.2: Desempenho de códigos encontrados

Código	Taxa	Capacidade	Ponto MUD	Dist. de projeto	Ponto de operação	Dist. da capacidade	Ponto SUD
A	0,3711	0,4198	3,9	1,1	6,7	2,8	6,5
B	0,2633	0,2958	3,9	1,1	7,1	3,2	8,2
C	0,5156	0,5388	4,5	0,5	7,7	3,2	7,1
D	0,5267	0,5352	4,8	0,2	8,6	3,8	9,4

Mostramos os resultados de simulações para a probabilidade de erro destes sistemas nas Figs. 5.8 a 5.11. Para cada combinação de  $N$ ,  $T$  e  $(E_t/N_0)^*$  utilizamos três tamanhos de bloco transmitido  $B$ :  $10^3$ ,  $10^4$  e  $10^5$  bits por bloco por usuário, aproximadamente. Para cada ponto foram contados pelo menos 100 erros de bits de informação para se chegar ao valor. Embora os resultados das curvas EXIT tenham sido para 10 iterações internas do MUD e em teoria infinitas iterações internas dos decodificadores, utilizamos 2 iterações internas do MUD, 2 iterações internas do código e 120 iterações entre o MUD e o decodificador.

A probabilidade de erro pode depender da implementação do código, mais especificamente da função  $\pi()$  utilizada. Uma implementação ruim pode causar ciclos de ordem baixa no grafo do decodificador, o que pode resultar em uma probabilidade de erro residual (“*error floor*”) alto. Para evitar que uma implementação particularmente ruim de um código afetasse os resultados, as funções  $\pi()$  de todos os usuários foram trocadas a cada  $10^5$  bits transmitidos por usuário, para os casos onde  $B = 10^2$  e  $10^3$ , e a cada  $5 \cdot 10^5$  bits transmitidos por usuário quando  $B = 10^5$ . Isto implica numa troca da função  $\pi()$  a cada 1000, 100 ou 5 palavras transmitidas, respectivamente.

Percebe-se das curvas que o comprimento do bloco influencia a probabilidade de erro. O melhor sistema encontrado trabalha a  $2.8dB$  da capacidade com probabilidade de erro menor do que  $10^{-5}$ , quando  $N = 4$ ,  $T = 2$  e  $(E_t/N_0)^* = 5dB$ . O valor de  $E_t/N_0$  necessário para que os sistemas

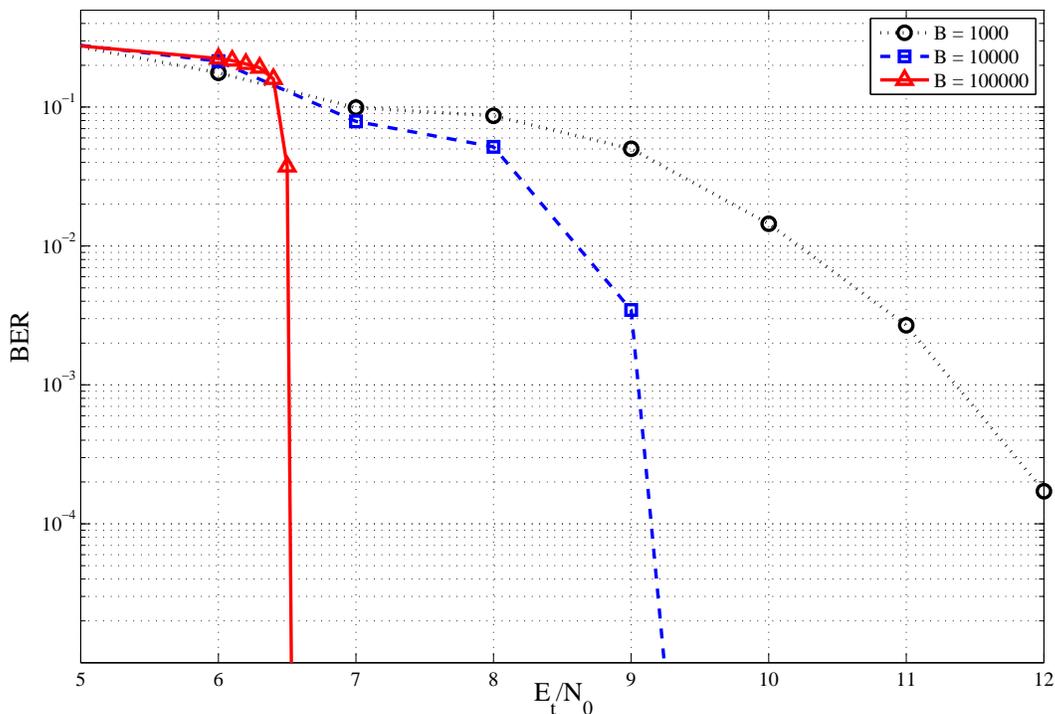


Fig. 5.8: BER simulada do código RA otimizado para  $N = 4$ ,  $T = 2$ ,  $(E_t/N_0)^* = 5dB$ .

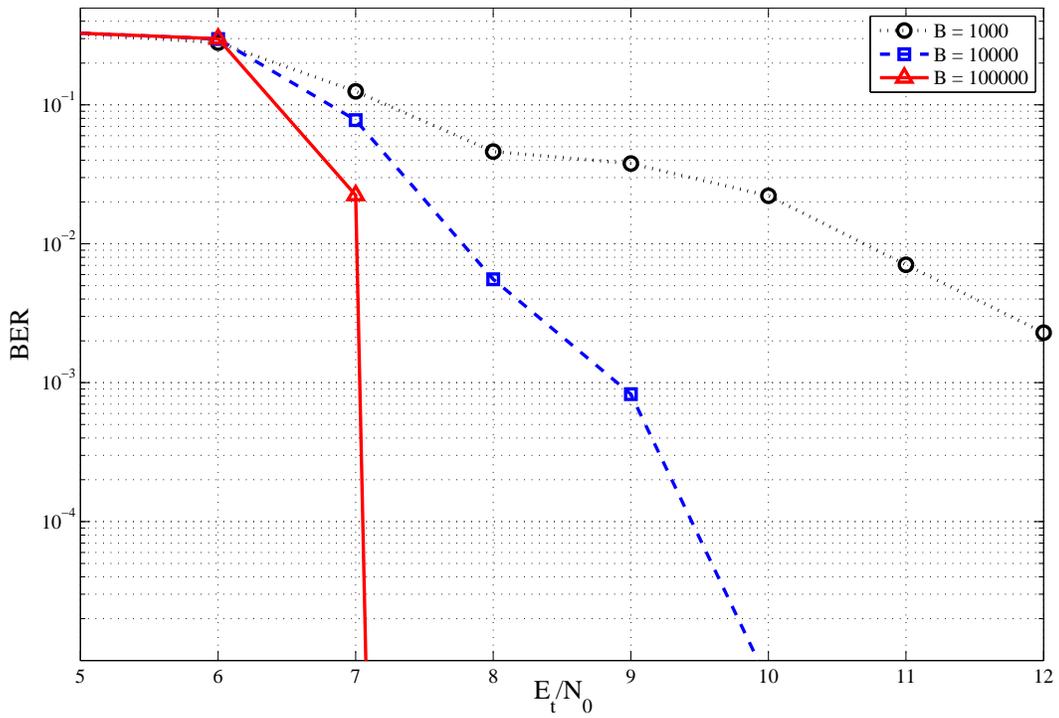


Fig. 5.9: BER simulada do código RA otimizado para  $N = 4, T = 3, (E_t/N_0)^* = 5dB$ .

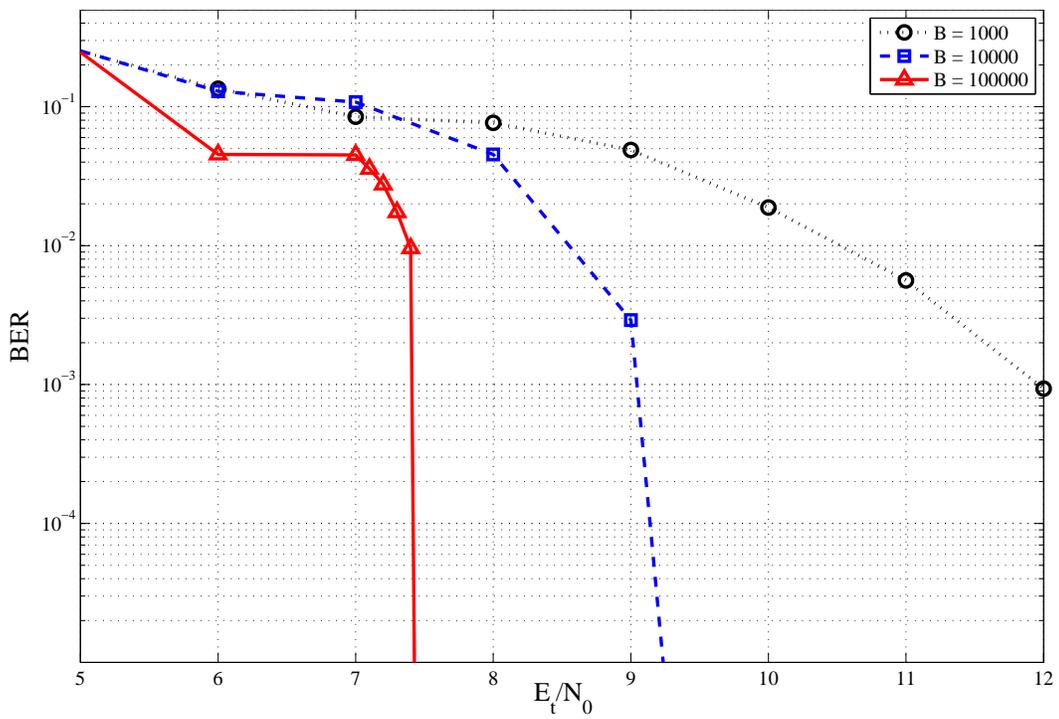


Fig. 5.10: BER simulada do código RA otimizado para  $N = 8, T = 2, (E_t/N_0)^* = 5dB$ .

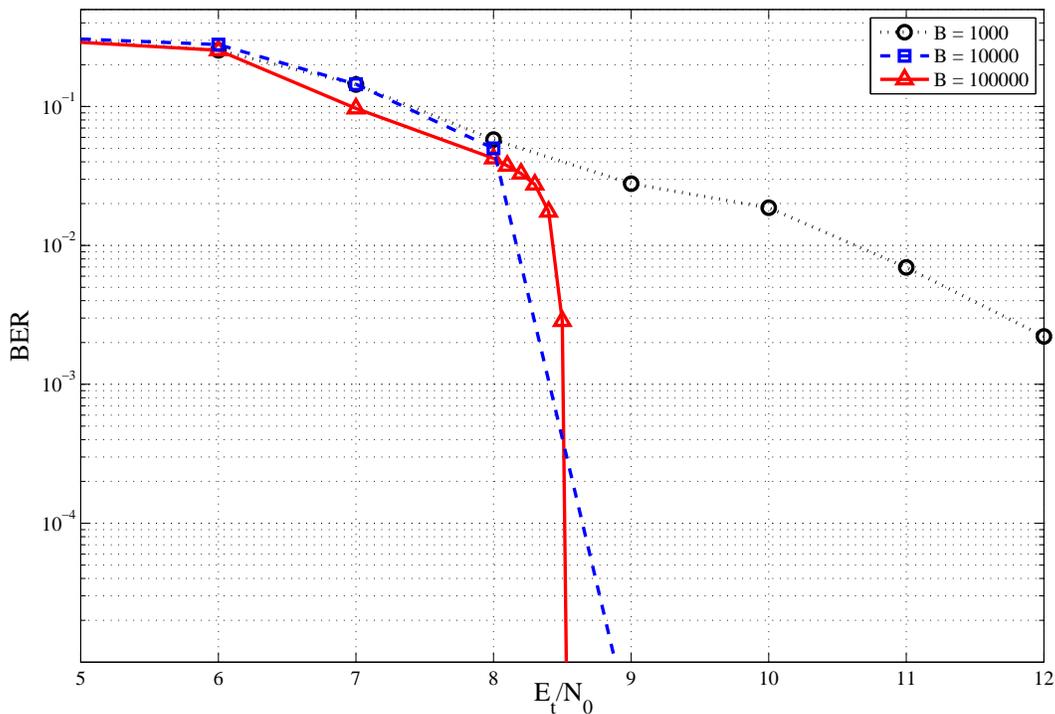


Fig. 5.11: BER simulada do código RA otimizado para  $N = 16$ ,  $T = 3$ ,  $(E_t/N_0)^* = 5dB$ .

B e D operem com probabilidade de erro de bit menor do que  $10^{-5}$  é menor do que o valor de  $E_t/N_0$  necessário para que a capacidade soma para detecção individual seja igual à taxa do respectivo sistema, isto é, estes sistemas operam satisfatoriamente num valor de  $E_t/N_0$  que é teoricamente impossível para a detecção individual.

Os resultados mais próximos dos obtidos neste trabalho foram apresentados em [18] e [19]. Em [18], códigos LDPC foram utilizados na tentativa de se obter sistemas para um canal de acesso múltiplo. O canal utilizado é do tipo binário aditivo (BAC, do inglês: *Binary Adder Channel*) com ruído Gaussiano e detecção coerente. Sistemas com dois usuários atingiram taxas de transmissão próximas da capacidade do canal quando as taxas individuais eram diferentes. Tentativas com taxas individuais idênticas ficaram mais distantes da capacidade do sistema. A solução para este problema foi mostrada em [19], onde se obteve sistemas com taxas igualmente distribuídas e próximas da capacidade também através de códigos LDPC.

As diferenças dos resultados anteriores para os encontrados neste trabalho são que os resultados apresentados aqui valem para um canal com desvanecimento seletivo em frequência e com detecção não coerente, o que implica numa dificuldade maior de projeto de sistemas. Mesmo assim, foi possível obter sistemas com mais de dois usuários com taxas idênticas e taxa soma próxima da capacidade do canal, sem a necessidade de um código de acesso. Os códigos RA utilizados pelos usuários

possuem os mesmos parâmetros, que precisam ser definidos por projeto somente uma vez. A única diferença entre os usuários é a forma como as camadas VND e CND foram conectadas através da função  $\pi()$ . A diferenciação desta função é que permite identificar qual usuário transmitiu qual mensagem e ao mesmo tempo corrigir erros de transmissão que venham a ocorrer.

## 5.5 Resumo

Neste capítulo obtivemos alguns sistemas para a transmissão de dados multiusuário num canal M-FSK ruidoso com desvanecimento Rayleigh. Para isso utilizamos as ferramentas EXIT desenvolvidas nos capítulos anteriores. O algoritmo utilizado foi simplificado para que o projeto dos códigos fosse feito rapidamente.

Resultados foram obtidos para decodificação paralela e serial. A distância entre a taxa do projeto e a capacidade do sistema no ponto de projeto está relacionada com a área entre as curvas. Esta área depende do formato das curvas EXIT do detector multiusuário e do código. Mesmo assim, os resultados da otimização mostram que a ferramenta EXIT fornece projetos próximos da capacidade do sistema. O desempenho dos codificadores gerados se mostrou próximo do previsto pelas curvas EXIT. Consequentemente, o desempenho dos codificadores deve estar próximo da capacidade do sistema.



# Capítulo 6

## Considerações Finais

Neste trabalho estudamos como receptores iterativos para canais de acesso múltiplo podem ser projetados implementados através de grafos de fatores. Para isso, tomamos como base o sistema FFH-CDMA, que tem sido considerado como uma opção para gerar sistemas UWB. As principais contribuições deste trabalho são resumidas a seguir.

### **Definição e cálculo da capacidade para detecção conjunta**

A grande maioria dos métodos anteriores utiliza uma combinação de utilização de um código de acesso, quantização das saídas do receptor e detecção individual, o que limita as taxas permissíveis. Para determinar qual é a capacidade do sistema, eliminamos o padrão de acesso utilizado pelo sistema FFH-CDMA, que pode ser interpretado como um código com taxa equivalente a um código de repetição. Obtivemos assim um canal M-FSK ruidoso, modelado como a concatenação de um canal não ruidoso com um canal ruidoso. Além disso, admitimos que as saídas do receptor pudessem assumir valores contínuos. Nestas condições, obtivemos valores para a informação mútua entre as mensagens transmitidas e os valores recebidos. Quando detecta-se somente um usuário (tratando os outros como interferência uniformemente distribuída), provamos que este usuário deve transmitir com símbolos de entrada uniformemente distribuídos para atingir a capacidade.

Um conjunto de argumentos indica que um produto de distribuições uniformes dos símbolos de entrada fornece um valor de informação mútua próximo da capacidade para a detecção conjunta, o que sugere a utilização desta distribuição. O valor da informação mútua com esta distribuição foi denominado capacidade efetiva. Mostramos que a capacidade efetiva atinge um máximo em função do número de usuários no sistema. A capacidade soma é um limite para a soma das taxas de todos os usuários. É possível identificar conjuntos de taxas individuais permissíveis que atingiriam a capacidade soma.

## Modelagem de um receptor multiusuário por grafos de fatores e análise EXIT

O receptor foi implementado através de um grafo de fatores representando a *pdf* conjunta dos bits transmitidos por todos os usuários e dos valores recebidos através do canal. Esta implementação permite o cálculo das probabilidades marginais dos bits transmitidos, dada uma realização do canal, através do algoritmo soma produto. O grafo também permite que sejam levados em consideração no cálculo da probabilidade de um bit valores conhecidos *a priori* sobre a probabilidade dos outros bits transmitidos. Os valores conhecidos *a priori* e os valores calculados são passados como mensagens de entrada e saída do grafo, respectivamente. Esta implementação permitiu que o receptor fosse analisado através de curvas EXIT, que relacionam o valor da informação *a priori* que as mensagens de entrada fornecem com o valor de informação extrínseca que o receptor retorna. As curvas EXIT podem ser interpretadas como uma função de transferência do receptor, sendo necessária uma contraparte para interagir com ele com o objetivo de determinar quais foram os bits transmitidos.

## Análise comparativa entre classes de códigos baseados em matrizes esparsas

A contraparte utilizada para interagir com o detector foram os decodificadores individuais dos usuários. A princípio qualquer tipo de código poderia ser utilizado. Concentramos nossa análise em códigos LDPC, LDGM e RA. Estes códigos permitem uma análise EXIT eficiente pois eles podem ser divididos em camadas com equações que definem analiticamente a relação entre o valor da informação *a priori* e o valor da informação extrínseca. Nossa principal contribuição nesta área foi mostrar que as equações das camadas podem ser numericamente analisadas para gerar uma curva EXIT global do código. Este procedimento pode ser utilizado em qualquer situação onde se deseja utilizar um código. Entre as classes de código analisadas, escolhemos os códigos RA sistemáticos baseado em características de desempenho esperado, complexidade de codificação e decodificação.

## Projeto de sistemas iterativos

O projeto do sistema consiste em escolher os parâmetros do código para que a sua curva EXIT esteja o mais próximo da curva EXIT do detector mas abaixo desta, o que garante que o processo irá convergir para a solução. Como o objetivo é obter o sistema com a maior taxa possível, o problema do projeto é maximizar a taxa dos códigos (o que também depende dos parâmetros deste) sujeito à restrição de que a curva do código esteja abaixo da curva do detector. Apresentamos algumas simplificações para este processo e o aplicamos para o caso estudado.

Resultados de simulação de probabilidade de erro mostram que os sistemas projetados pelo método utilizado podem ter desempenho muito próximo da capacidade. É possível que resultados ainda melhores possam ser obtidos com outras classes de código.

## 6.1 Estudos futuros

Os resultados obtidos neste trabalho indicam algumas possibilidades de estudos futuros, listadas a seguir:

### Aplicação a outros tipos de canais

Toda a análise apresentada neste trabalho pode ser feita para outros tipos de canais de acesso múltiplo. A estrutura do grafo apresentado pode ser utilizado em outros tipos de canais diferentes do apresentado, bastando para isso alterar as funções locais e o significado de algumas variáveis. Para analisar canais com estatísticas de ruído e desvanecimento diferentes, basta alterar a função  $p(R_n|c_n)$ . Canais de acesso múltiplo baseados em PPM ou DS-CDMA também podem ser considerados. Em qualquer destes casos, o mesmo tipo de análise do detector e subsequente otimização dos códigos pode ser feita.

### Caso multiantena

É possível melhorar o desempenho do receptor simplesmente pela adição de antenas. Se suficientemente espaçadas, não haverá correlação entre o desvanecimento e fase dos sinais recebidos em cada uma das antenas. Havendo  $L$  antenas, isto permitiria obter  $L$  observações para cada frequência. Em vez de receber um vetor com dimensão  $N$ , teríamos uma matriz  $N \times L$  com termos  $R_n^l$  de valores recebidos.

No grafo de fatores, isto implica em alterar a conexão das variáveis  $c_n$  com os valores recebidos. O fator local teria o formato da Fig. 6.1. As mensagens de saída do nó  $c_n$  precisariam ser calculadas pelo produto das mensagens de entrada, normalizados.

A adição de antenas alteraria o cálculo da capacidade e as curvas EXIT do detector, possivelmente elevando-as.

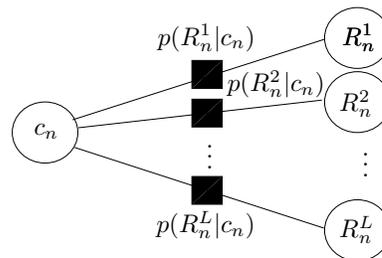


Fig. 6.1: Subgrafo relacionando as variáveis  $c_n$  e  $R_n^1, R_n^2, \dots, R_n^L$  para o caso multiantena.

### Controle de potência e caso multipotência

Neste trabalho consideramos que, no receptor, a potência média recebida de todos os usuários é a mesma. Isto exigiria um controle de potência que não foi investigado neste trabalho. Sendo a potência recebida inversamente proporcional ao quadrado da distância entre o transmissor e o receptor, usuários mais distantes do receptor deveriam transmitir com potência maior. O sistema proposto permite que o receptor identifique (após decodificação) quando cada um dos usuários está transmitindo sozinho numa dada frequência. Isto permitiria levantar estatísticas sobre a potência recebida do usuário e posteriormente ajustar a sua potência transmitida. Nada impede que este processo também seja feito através de grafos de fatores, o que seria uma evolução do detector proposto neste trabalho.

Um outro cenário possível é que a potência recebida não seja a mesma para todos os usuários. Neste caso, um controle de potência não é necessário. Em seu lugar, o receptor deve considerar esta diferença. Seja  $w^i$  um número entre 0 e 1 que indica para o  $i$ -ésimo usuário a relação entre a potência recebida a maior potência recebida para todos os usuários. Retomando a notação da Seção 2.1, o sinal recebido para a  $n$ -ésima frequência teria o seguinte valor:

$$r_n = \sum_{j=1}^T \sqrt{w^j} \cdot c_n^j \cdot \sqrt{2E_c \alpha_n^j} \cdot [x_n(t) \cdot \cos(\phi_n^j) + y_n(t) \cdot \sin(\phi_n^j)]. \quad (6.1)$$

A potência total transmitida na frequência  $n$  seria igual a  $w_n = \sum_{j=1}^T w^j \cdot c_n^j$ . Seguindo o raciocínio da Seção 2.1, a função densidade de probabilidade de  $R_n$  teria como parâmetro  $w_n$  em vez de  $c_n$ , como indicam as equações a seguir:

$$p(R_n | w_n) = \frac{1}{w_n + d} \exp\left(-\frac{R_n}{w_n + d}\right) \quad (6.2)$$

$$p(R_n) = \sum_{w_n} p(R_n | w_n) P(w_n),$$

onde o somatório é feito para todos os valores de  $w_n$  possíveis.

Se, para qualquer valor de  $w_n$ , existir no máximo um conjunto de  $c_n^1, c_n^2, \dots, c_n^T$  tal que  $w_n = \sum_{i=1}^T w^i \cdot c_n^i$ , então há  $2^T$  valores possíveis de  $w_n$ . A complexidade de calcular a equação (6.2) seria muito maior do que calcular a equação (2.8). Por outro lado, identificar o valor de  $w_n$  neste caso seria o mesmo que identificar não quantos mas sim quais usuários estão transmitindo nesta frequência, o que pode ser vantajoso para o processo de decodificação.

### Capacidade do canal multiusuário

Embora tenhamos demonstrado que a distribuição uniforme nos símbolos de entrada faz com que a informação mútua entre a mensagem de um usuário e o sinal recebido atinja a capacidade, uma prova concreta de que a mesma distribuição atinge a capacidade para o caso multiusuário não foi encontrada.

Uma prova parcial seria provar que a informação mútua  $\mathcal{I}(R, M)$  é uma função côncava da distribuição de entrada comum a todos os usuários para alguns valores de  $T$ . Neste caso, o mesmo raciocínio aplicado ao caso de  $\mathcal{I}(R, m_i)$  pode ser posteriormente aplicado com o mesmo resultado.

Além da capacidade soma, uma análise mais detalhada das taxas de transmissão individuais é necessária. Uma pergunta seria saber se todos os usuários podem transmitir com a mesma taxa e atingirem conjuntamente a capacidade soma.

### Downlink

Se o objetivo do sistema é ter um canal de comunicação bidirecional, a construção do sistema responsável pelo downlink é necessária. Neste caso, um único transmissor deseja transmitir mensagens diferentes para receptores diferentes.

As principais diferenças deste sistema com o uplink é que o transmissor tem conhecimento de todas as mensagens e pode codificá-las conjuntamente. O receptor, por outro lado, só poderia ter acesso a mensagem destinada a ele. Além disso, os sinais (se mais de um) podem ser transmitidos em sincronia temporal e percorrem o mesmo percurso, o que implica que todos os sinais dentro de uma mesma frequência sofrem o mesmo desvanecimento e possuem a mesma fase.

A natureza deste canal é radicalmente diferente do canal estudado neste trabalho, de forma que pouco (ou nada) do que foi desenvolvido aqui possa ser utilizado para o desenvolvimento deste sistema.



# Apêndice A

## Tabelas de Capacidade

As tabelas a seguir apresentam os valores nominais calculados para a  $C^{SOMA}$  do sistema (como definida no capítulo 2 para alguns valores de  $N$ ,  $T$  e  $E_t/N_0$ ).

Tab. A.1: Valores para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD),  $N = 4$ .

$E_t/N_0[dB]$	T = 2		T = 3	
	SUD	MUD	SUD	MUD
5	1,2898	1,6795	1,2284	1,7752
6	1,4182	1,8516	1,3424	1,9570
7	1,5400	2,0134	1,4530	2,1315
8	1,6536	2,1631	1,5590	2,2964
9	1,7577	2,2995	1,6594	2,4503
10	1,8517	2,4223	1,7539	2,5924
11	1,9355	2,5317	1,8422	2,7223
12	2,0091	2,6283	1,9237	2,8400
13	2,0733	2,7133	1,9981	2,9458
14	2,1292	2,7877	2,0656	3,0404
15	2,1781	2,8530	2,1262	3,1245
16	2,2209	2,9101	2,1805	3,1987
17	2,2584	2,9599	2,2289	3,2641
18	2,2912	3,0032	2,2719	3,3213
19	2,3197	3,0406	2,3099	3,3713
20	2,3446	3,0728	2,3434	3,4149
21	2,3662	3,1007	2,3730	3,4529
22	2,3850	3,1246	2,3991	3,4859
23	2,4014	3,1452	2,4218	3,5144
24	2,4156	3,1628	2,4414	3,5387
25	2,4281	3,1780	2,4583	3,5595

Tab. A.2: Valores para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD),  $N = 8$ .

$E_t/N_0[dB]$	T = 2		T = 4		T = 6	
	SUD	MUD	SUD	MUD	SUD	MUD
5	2,6619	3,2331	3,0334	4,2845	2,9744	4,5248
6	2,8766	3,4883	3,2740	4,6349	3,2028	4,8894
7	3,0713	3,7179	3,4998	4,9599	3,4213	5,2316
8	3,2440	3,9210	3,7090	5,2574	3,6280	5,5490
9	3,3947	4,0986	3,9007	5,5266	3,8214	5,8402
10	3,5246	4,2523	4,0742	5,7677	4,0008	6,1050
11	3,6357	4,3846	4,2296	5,9819	4,1660	6,3438
12	3,7303	4,4978	4,3673	6,1708	4,3169	6,5574
13	3,8104	4,5943	4,4884	6,3364	4,4533	6,7472
14	3,8778	4,6762	4,5942	6,4807	4,5756	6,9146
15	3,9345	4,7457	4,6862	6,6059	4,6842	7,0614
16	3,9821	4,8048	4,7660	6,7143	4,7802	7,1896
17	4,0221	4,8551	4,8356	6,8080	4,8650	7,3011
18	4,0560	4,8980	4,8961	6,8888	4,9398	7,3979
19	4,0849	4,9346	4,9482	6,9582	5,0062	7,4815
20	4,1096	4,9658	4,9928	7,0177	5,0648	7,5536
21	4,1304	4,9923	5,0311	7,0686	5,1160	7,6158
22	4,1479	5,0146	5,0644	7,1122	5,1605	7,6693
23	4,1630	5,0335	5,0936	7,1496	5,1990	7,7156
24	4,1763	5,0495	5,1193	7,1818	5,2318	7,7555
25	4,1880	5,0632	5,1413	7,2092	5,2594	7,7897

Tab. A.3: Valores para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD),  $N = 16$ .

$E_t/N_0[dB]$	T = 3		T = 4		T = 8		T = 12	
	SUD	MUD	SUD	MUD	SUD	MUD	SUD	MUD
5	5,0364	6,4231	5,5788	7,4922	6,5912	9,7487	6,6363	10,3179
6	5,3880	6,8562	5,9663	8,0072	7,0341	10,4213	7,0760	11,0132
7	5,7053	7,2416	6,3198	8,4705	7,4453	11,0389	7,4914	11,6563
8	5,9864	7,5799	6,6369	8,8820	7,8236	11,5997	7,8803	12,2446
9	6,2316	7,8738	6,9184	9,2438	8,1693	12,1042	8,2410	12,7777
10	6,4426	8,1269	7,1664	9,5592	8,4829	12,5541	8,5731	13,2565
11	6,6224	8,3430	7,3829	9,8322	8,7652	12,9521	8,8764	13,6834
12	6,7744	8,5266	7,5700	10,0667	9,0175	13,3016	9,1511	14,0615
13	6,9027	8,6820	7,7303	10,2673	9,2418	13,6067	9,3982	14,3945
14	7,0108	8,8133	7,8670	10,4383	9,4406	13,8722	9,6197	14,6865
15	7,1015	8,9239	7,9831	10,5841	9,6157	14,1025	9,8176	14,9414
16	7,1777	9,0170	8,0815	10,7082	9,7692	14,3018	9,9939	15,1630
17	7,2417	9,0952	8,1648	10,8138	9,9025	14,4733	10,1497	15,3549
18	7,2953	9,1612	8,2353	10,9038	10,0174	14,6203	10,2862	15,5206
19	7,3403	9,2168	8,2950	10,9804	10,1165	14,7459	10,4043	15,6635
20	7,3781	9,2638	8,3454	11,0455	10,2025	14,8531	10,5052	15,7861
21	7,4099	9,3035	8,3882	11,1007	10,2774	14,9443	10,5916	15,8910
22	7,4366	9,3369	8,4247	11,1476	10,3425	15,0217	10,6669	15,9807
23	7,4592	9,3651	8,4556	11,1873	10,3990	15,0873	10,7333	16,0573
24	7,4784	9,3891	8,4818	11,2209	10,4476	15,1430	10,7907	16,1228
25	7,4944	9,4093	8,5042	11,2493	10,4894	15,1900	10,8393	16,1785

Apresentamos nas tabelas a seguir os valores normalizados para a capacidade soma para alguns valores de  $N$ ,  $T$  e  $E_t/N_0$ . Estes valores são definidos como  $C^{SOMA}/(T \cdot \log_2 N)$ , onde  $C^{SOMA}$  é a capacidade soma nominal do sistema, apresentado nas tabelas anteriores

Tab. A.4: Valores normalizados para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD),  $N = 4$ .

$E_t/N_0[dB]$	T = 2		T = 3	
	SUD	MUD	SUD	MUD
5	0,3224	0,4199	0,2047	0,2959
6	0,3546	0,4629	0,2237	0,3262
7	0,3850	0,5033	0,2422	0,3553
8	0,4134	0,5408	0,2598	0,3827
9	0,4394	0,5749	0,2766	0,4084
10	0,4629	0,6056	0,2923	0,4321
11	0,4839	0,6329	0,3070	0,4537
12	0,5023	0,6571	0,3206	0,4733
13	0,5183	0,6783	0,3330	0,4910
14	0,5323	0,6969	0,3443	0,5067
15	0,5445	0,7132	0,3544	0,5207
16	0,5552	0,7275	0,3634	0,5331
17	0,5646	0,7400	0,3715	0,5440
18	0,5728	0,7508	0,3786	0,5536
19	0,5799	0,7601	0,3850	0,5619
20	0,5861	0,7682	0,3906	0,5691
21	0,5915	0,7752	0,3955	0,5755
22	0,5962	0,7811	0,3999	0,5810
23	0,6003	0,7863	0,4036	0,5857
24	0,6039	0,7907	0,4069	0,5898
25	0,6070	0,7945	0,4097	0,5932

Tab. A.5: Valores normalizados para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD),  $N = 8$ .

$E_t/N_0[dB]$	T = 2		T = 4		T = 6	
	SUD	MUD	SUD	MUD	SUD	MUD
5	0,4436	0,5388	0,2528	0,3570	0,1652	0,2514
6	0,4794	0,5814	0,2728	0,3862	0,1779	0,2716
7	0,5119	0,6196	0,2916	0,4133	0,1901	0,2906
8	0,5407	0,6535	0,3091	0,4381	0,2016	0,3083
9	0,5658	0,6831	0,3251	0,4605	0,2123	0,3245
10	0,5874	0,7087	0,3395	0,4806	0,2223	0,3392
11	0,6060	0,7308	0,3525	0,4985	0,2314	0,3524
12	0,6217	0,7496	0,3639	0,5142	0,2398	0,3643
13	0,6351	0,7657	0,3740	0,5280	0,2474	0,3748
14	0,6463	0,7794	0,3829	0,5401	0,2542	0,3841
15	0,6557	0,7910	0,3905	0,5505	0,2602	0,3923
16	0,6637	0,8008	0,3972	0,5595	0,2656	0,3994
17	0,6703	0,8092	0,4030	0,5673	0,2703	0,4056
18	0,6760	0,8163	0,4080	0,5741	0,2744	0,4110
19	0,6808	0,8224	0,4124	0,5799	0,2781	0,4156
20	0,6849	0,8276	0,4161	0,5848	0,2814	0,4196
21	0,6884	0,8320	0,4193	0,5891	0,2842	0,4231
22	0,6913	0,8358	0,4220	0,5927	0,2867	0,4261
23	0,6938	0,8389	0,4245	0,5958	0,2888	0,4286
24	0,6961	0,8416	0,4266	0,5985	0,2907	0,4309
25	0,6980	0,8439	0,4284	0,6008	0,2922	0,4328

Tab. A.6: Valores normalizados para capacidade soma para detecção individual(SUD) e capacidade soma efetiva para detecção conjunta(MUD),  $N = 16$ .

$E_t/N_0[dB]$	T = 3		T = 4		T = 8		T = 12	
	SUD	MUD	SUD	MUD	SUD	MUD	SUD	MUD
5	0,4197	0,5353	0,3487	0,4683	0,2060	0,3046	0,1383	0,2150
6	0,4490	0,5714	0,3729	0,5005	0,2198	0,3257	0,1474	0,2294
7	0,4754	0,6035	0,3950	0,5294	0,2327	0,3450	0,1561	0,2428
8	0,4989	0,6317	0,4148	0,5551	0,2445	0,3625	0,1642	0,2551
9	0,5193	0,6562	0,4324	0,5777	0,2553	0,3783	0,1717	0,2662
10	0,5369	0,6772	0,4479	0,5975	0,2651	0,3923	0,1786	0,2762
11	0,5519	0,6953	0,4614	0,6145	0,2739	0,4048	0,1849	0,2851
12	0,5645	0,7106	0,4731	0,6292	0,2818	0,4157	0,1906	0,2929
13	0,5752	0,7235	0,4831	0,6417	0,2888	0,4252	0,1958	0,2999
14	0,5842	0,7344	0,4917	0,6524	0,2950	0,4335	0,2004	0,3060
15	0,5918	0,7437	0,4989	0,6615	0,3005	0,4407	0,2045	0,3113
16	0,5981	0,7514	0,5051	0,6693	0,3053	0,4469	0,2082	0,3159
17	0,6035	0,7579	0,5103	0,6759	0,3095	0,4523	0,2115	0,3199
18	0,6079	0,7634	0,5147	0,6815	0,3130	0,4569	0,2143	0,3233
19	0,6117	0,7681	0,5184	0,6863	0,3161	0,4608	0,2168	0,3263
20	0,6148	0,7720	0,5216	0,6903	0,3188	0,4642	0,2189	0,3289
21	0,6175	0,7753	0,5243	0,6938	0,3212	0,4670	0,2207	0,3311
22	0,6197	0,7781	0,5265	0,6967	0,3232	0,4694	0,2222	0,3329
23	0,6216	0,7804	0,5285	0,6992	0,3250	0,4715	0,2236	0,3345
24	0,6232	0,7824	0,5301	0,7013	0,3265	0,4732	0,2248	0,3359
25	0,6245	0,7841	0,5315	0,7031	0,3278	0,4747	0,2258	0,3371



# Bibliografia

- [1] <http://www.ieee802.org/>. IEEE 802 LAN/MAN Standards Committee, acessado em julho de 2010.
- [2] <https://www.bluetooth.org/>. Special Interest Group, acessado em julho de 2010.
- [3] <http://www.usb.org/developers/wusb/>. Wireless USB Promoter Group, acessado em julho de 2010.
- [4] C.E. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:623–656, 1948.
- [5] S. Haykin. *Communication Systems, Fourth Edition*. Wiley, 2001.
- [6] D. J. Goodman, P. S. Henry e V. K. Prabhu. Frequency-hopped multilevel FSK for mobile radio. *Bell Systems Technical Journal*, 59(7):1257–1275, Setembro de 1980.
- [7] S. ten Brink. Convergence of iterative decoding. *Electronic Letters*, 35(10):806–808, Maio de 1999.
- [8] U. Timor. Improved decoding scheme for frequency-hopped multilevel FSK system. *Bell Systems Technical Journal*, 59(10):1839–1855, Dezembro de 1980.
- [9] T. Mabuchi, R. Khono e H. Imai. Multiuser detection scheme based on canceling cochannel interference for MFSK/FH-SSMA systems. *IEEE Journal on Selected Areas in Communications*, 12(4):593–604, Maio de 1994.
- [10] U.-C.G. Fiebig. An algorithm for joint detection in fast frequency hopping systems. *Conference Record, 1996 IEEE International Conference on Converging Technologies for Tomorrow's Applications*, 1:90–95, Junho de 1996.
- [11] U.-C.G. Fiebig. Iterative interference cancellation for FFH/MFSK MA systems. *IEE Proceedings-Communications*, 143(6):380–388, Dezembro de 1996.

- [12] Xia Wang, Shihua Zhu e Delong Sun. Cochannel interference cancellation for frequency hopped multiple access systems. *2004 IEEE Eighth International Symposium on Spread Spectrum Techniques and Applications*, Agosto de 2004.
- [13] On-Ching Yue. Maximum likelihood combining for noncoherent and differentially coherent frequency-hopping multiple-access systems. *IEEE Transactions on Information Theory*, 28(4):631–639, Julho de 1982.
- [14] G. A. de Deus Junior. A utilização de redes neurais artificiais no projeto de receptores FH-CDMA. Dissertação de mestrado, Universidade Estadual de Campinas, Campinas, SP, Brasil, Fevereiro de 1998.
- [15] G. A. de Deus Junior. *Sistemas FFH-CDMA codificados*. Tese de doutorado, Universidade Estadual de Campinas, Campinas, SP, Brasil, Agosto de 2002.
- [16] G. Kramer. Topics in multi-user information theory. *Foundations and Trends® in Communications and Information Theory*, 4(4-5):265–444, 2007.
- [17] B. Rimoldi e R. Urbanke. A rate-splitting approach to the Gaussian multiple-access channel. *IEEE Transactions on Information Theory*, 42(2):364–375, 1996.
- [18] R. Palanki. *Iterative decoding for wireless networks*. Tese de doutorado, California Institute of Technology, Pasadena, CA, EUA, Maio de 2004.
- [19] A. Roumy e D. Declercq. Characterization and optimization of LDPC codes for the 2-user Gaussian multiple access channel. *EURASIP Journal on Wireless Communications and Networking*, 2007. Article ID 74890.
- [20] Shih-Chun Chang e J. Wolf. On the T-user M-frequency noiseless multiple-access channel with and without intensity information. *IEEE Transactions on Information Theory*, 27(1):41–48, Janeiro de 1981.
- [21] G. Einarsson. Channel capacity and error probability for a model of code-division multiple access system. *Archiv für Elektronik und Übertragungstechnik*, 38(3):157–163, Maio/Junho de 1984.
- [22] J. G. Goh e S. V. Maric. The capacities of frequency-hopped code-division multiple-access channels. *IEEE Transactions on Information Theory*, 44(3):1204–1211, Maio de 1998.
- [23] Thomas M. Cover e Joy A. Thomas. *Elements of Information Theory*. John Wiley, 1991.

- [24] Shu Lin e Daniel J. Costello Jr. *Error Control Coding: Fundamentals and Applications*. Prentice-Hall, 1983.
- [25] On-Ching Yue. Performance of frequency-hopping multiple-access multilevel FSK with hard-limited and linear combining. *IEEE Transactions on Communications*, 29(11):1687–1694, Novembro de 1981.
- [26] Lie-Liang Yang e L. Hanzo. Residue number system assisted fast frequency-hopped synchronous ultra-wideband spread-spectrum multiple-access: a design alternative to impulse radio. *IEEE Journal on Selected Areas in Communications*, 20(9):1652–1663, Dezembro de 2002.
- [27] S. ten Brink. Convergence behavior of iteratively decoded parallel concatenated codes. *IEEE Transactions on Communications*, 49(10):1727–1737, Outubro de 2001.
- [28] S. ten Brink, G. Kramer e A. Ashikhmin. Design of low-density parity check codes for modulation and detection. *IEEE Transactions on Communications*, 52(4):670–678, Abril de 2004.
- [29] S. ten Brink e G. Kramer. Design of repeat-accumulate codes for iterative detection and decoding. *IEEE Transactions on Signal Processing*, 51(11):2764–2772, Novembro de 2003.
- [30] J. Hagenauer, E. Offer e L. Papke. Iterative decoding of binary block and convolutional codes. *IEEE Transactions on Information Theory*, 42(2):429–445, Março de 1996.
- [31] M. Tüchler e J. Hagenauer. Exit charts of irregular codes. *Proceedings CISS*, pp. 748–753, 2002.
- [32] F.R. Kschischang, B. J. Frey e H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, Fevereiro de 2001.
- [33] Sheng Tong e Xinmei Wang. A simple convergence comparison of gallager codes under two message-passing schedules. *IEEE Communications Letters*, 9(3):249–251, Março de 2005.
- [34] A. Ashikhmin, G. Kramer e S. ten Brink. Extrinsic information transfer functions: A model and two properties. *Proceedings Conference Information Sciences and Systems*, pp. 742–747, Março de 2002.
- [35] R. Gallager. Low-density parity-check codes. *IEEE Transactions on Information Theory*, 8(1):21–28, Janeiro de 1962.
- [36] J. Garcia-Frias e Wei Zhong. Approaching shannon performance by iterative decoding of linear codes with low-density generator matrix. *IEEE Communications Letters*, 7(6):266 – 268, Junho de 2003.

- [37] D. Divsalar, H. Jin e R. J. McEliece. Coding theorems for ‘turbo-like’ codes. *Proceedings 36<sup>th</sup> Annual Allerton Conference on Communication, Control and Computing*, pp. 201–210, Setembro de 1998.
- [38] X.-Y. Hu, E. Eleftheriou, D.-M. Arnold e A. Dholakia. Efficient implementations of the sum-product algorithm for decoding LDPC codes. *2001 IEEE Global Telecommunications Conference*, 2:1036–1036, 2001.
- [39] E. Sharon, A. Ashikhmin e S. Litsyn. Exit functions for the Gaussian channel. *Proceedings 40<sup>th</sup> Annual Allerton Conference Communication*, pp. 972–981, Outubro de 2003.
- [40] Jung fu Cheng e R. J. McEliece. Some high-rate near capacity codecs for the Gaussian channel. *34<sup>th</sup> Allerton Conference on Communications, Control and Computing*, 1996.
- [41] H. Jin, A. Khandekar e R. McEliece. Irregular repeat-accumulate codes. *Proceedings 2<sup>nd</sup> International Symposium on Turbo codes and Related Topics*, pp. 1–8, Janeiro de 2000.
- [42] L. R. Bahl, J. Cocke, F. Jelinek e J. Raviv. Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Transactions on Information Theory*, 20(2):284–287, Março de 1974.
- [43] Li Ping, W.K. Leung e Nam Phamdo. Low density parity check codes with semi-random parity check matrix. *Electronic Letters*, 35(1):38–39, Janeiro de 1999.
- [44] R.Y. Shao, Shu Lin e M.P.C. Fossorier. Two simple stopping criteria for turbo decoding. *IEEE Transactions on Communications*, 47(8):1117–1120, Agosto de 1999.
- [45] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 1991.