Dissertação de Mestrado

Inatel
Instituto Nacional de Telecomunicações

Um Estudo de Técnica de Codificação Cooperativa em Canais com Desvanecimento Seletivo em Frequência

Tiago Reis Rufino Marins

Dezembro / 2012

Um estudo de sistema de codificação cooperativa em canais com desvanecimento seletivo em frequência

Tiago Reis Rufino Marins

Dissertação apresentada ao Instituto Nacional de Telecomunicações, como parte dos requisitos para obtenção do título de Mestre em Telecomunicações.

Orientador: Prof. Dr. Geraldo Gil Ramundo Gomes

Co-Orientador: Prof. Dr. Luciano Leonel Mendes

Santa Rita do Sapucaí
2012

Marins, Tiago Reis Rufino
M339e
Um Estudo de Sistema de Codificação Cooperativa em Canais com Desvanecimento Seletivo em Frequência. / Tiago Reis Rufino Marins. – Santa Rita do Sapucaí, 2012.
76 p.

Orientador: Prof. Ph.D. Geraldo Gil Ramundo Gomes; Prof. Dr. Luciano Leonel Mendes.
Dissertação de Mestrado – Engenharia de Telecomunicações – Instituto Nacional de Telecomunicações – INATEL.
Inclui bibliografia e anexo.

1. Codificação Cooperativa 2. Reed Solomon 3. OFDM 4. Engenharia de Telecomunicações. I. Gomes, Geraldo Gil Ramundo. II. Mendes, Luciano Leonel. III. Instituto Nacional de Telecomunicações – INATEL. IV. Título.

CDU 621.39

Dissertação defendida e aprovada em __/__/___, pela comissão julgadora:

(Prof. Dr. Geraldo Gil Ramundo Gomes (Orientador) / Instituto Nacional de Telecomunicações (INATEL)

Prof. Dr. Luciano Leonel Mendes (Co-orientador) / Instituto Nacional de Telecomunicações (INATEL)

Prof. Dr. Dayan Adionel Guimarães / Instituto Nacional de Telecomunicações (INATEL)

Prof. Dr. Fabbryccio Akkazzha C. M. Cardoso / CPqD

Prof. Dr. Luciano Leonel Mendes -

Coordenador do Curso de Mestrado

À minha esposa e filha,

razões do meu viver.

Agradecimentos

A Deus por proporcionar força de vontade para vencer mais essa etapa da minha vida. Aos meus pais e irmãos pelo apoio, incentivo e dedicação prestados em todos os momentos da minha vida.

A minha esposa Maria e minha filha Liz pela paciência e compreensão nos momentos ausentes.

Ao meu orientador Prof. Dr. Geraldo Gil Ramundo Gomes e Co-Orientador Prof. Dr. Luciano Leonel Mendes pelas orientações, sugestões, conselhos e auxílios na busca por soluções aos problemas encontrados ao longo dessa jornada.

Aos colegas do laboratório de Hardware do Inatel, em especial a meus amigos Alexandre Carvalho Ferreira, André Antônio dos Anjos, Anderson Daniel Soares, Antônio Alves Ferreira Junior, Carlos Augusto Rocha, Cleomarcio Almeida, Guilherme Pedro Aquino e Marcelo Carneiro de Paiva pelo apoio incomensurável nas horas difíceis ao longo desse período.

A todos professores e funcionários do curso de Mestrado do Instituto Nacional de Telecomunicações.

Índice

Lista de Figuras vi

Lista de Tabelas viii

Lista de Abreviaturas e Siglas x

Lista de Símbolos xii

1 Introdução 1

2 Princípios da comunicação cooperativa 5

2.1 Técnica SF 6

2.2 Técnica AF 7

2.3 Técnica DF 8

2.4 Técnica CF 8

2.5 Técnica ST-CC 9

2.5.1 Técnica CC 9

2.6 Técnica proposta de codificação cooperativa CC-RS 12

2.7 Análise de desempenho da técnica CC-RS 14

2.7.1 Cenário proposto 15

2.7.2 Estruturação do quadro na transmissão 17

2.7.3 Método não sinalizado 18

2.7.4 Método sinalizado 21

2.8 Análise dos resultados das simulações 24

2.9 Conclusões 27

3 Codificação cooperativa em canais com desvanecimento seletivo em frequência. 28

3.1 Análise do canal de comunicação 29

3.1.1 Limiar de recepção 29

3.2 Descrição do cenário para análise do esquema de codificação cooperativa CC-RS 31

3.3 Estruturação do quadro na transmissão 37

3.4 Reestruturação do quadro na ERB 37

3.5 Análise dos resultados das simulações. 39

3.6 Conclusões 46

4 Conclusões e proposta de trabalhos futuros 48

A Reed Solomon 51

A.1 Princípio de funcionamento da codificação e decodificação do Reed Solomon 51

B Ferramenta computacional utilizada nas simulações 54

C Software da unidade de controle do usuário relay 57

D Publicações oriundas da dissertação. 71

Bibliografia 72

Lista de Figuras

2.1 Ilustração de um cenário típico da técnica SF utilizado em redes. . 6

2.2 Ilustração da técnica de comunicação cooperativa AF. 7

2.3 Ilustração da técnica de comunicação cooperativa DF. 8

2.4 Ilustração das duas fases empregadas pela técnica ST-CC. 10

2.5 Estrutura dos quadros com e sem o processo de codificação cooperativa. 11

2.6 Parâmetros envolvidos no cálculo da taxa de cooperação. 13

2.7 Cenário proposto nas simulações. 15

2.8 Codificação cooperativa no sistema TDMA. 16

2.9 Diagrama em blocos do terminal do usuário. 18

2.10 Método não sinalizado: (a) Etapa de recepção do usuário fonte com o processo de cooperação. (b) Etapa de recepção do usuário fonte sem o processo de cooperação. 19

2.11 Cenário onde há maior probabilidade de não cooperação. 20

2.12 Possibilidade do resultado da decodificação do quadro do usuário fonte no sistema não sinalizado e sem o processo da cooperação. . 21

2.13 (a) Valores atribuídos ao símbolo de sinalização nos quadros com cooperação e sem cooperação para o código RS(15,5) (a) e para o código RS(255,85) (b). 22

2.14 Disposição dos quadros do *relay* com cooperação e sem cooperação. 23

2.15 Método sinalizado: (a) Etapa de recepção do usuário fonte com o processo de cooperação. (b) Etapa de recepção do usuário fonte sem o processo de cooperação. 24
2.16 Desempenho do usuário fonte em função da taxa de cooperação. . 26
2.17 Desempenho do usuário fonte em função da taxa de cooperação. . 26

3.1 Limiar utilizado para análise do canal de transmissão. 30
3.2 Fluxograma utilizado pela ERB para autorizar o usuário *relay* a cooperar. 31
3.3 Cenário proposto para as simulações com canal seletivo móvel. . . 32
3.4 Diagrama em blocos básico do transmissor do usuário. 33
3.5 (a) Resposta em frequência do canal do usuário *relay* para a ERB. (b) Resposta em frequência do canal do usuário fonte para a ERB. 35
3.6 Fluxograma de estruturação do quadro do usuário *relay*. 38
3.7 Diagrama em blocos da ERB. 39
3.8 (a) Etapa de recepção do usuário fonte com o processo de cooperação. (b) Etapa de recepção do usuário fonte sem o processo de cooperação. 40
3.9 Desempenho do usuário fonte utilizando o código RS(15,5). 41
3.10 Curva de desempenho do usuário fonte utilizando o código RS(255,85). 41
3.11 Desempenho do usuário fonte utilizando o código RS(15,5). 43
3.12 Desempenho do usuário fonte utilizando o código RS(255,85). . . 43
3.13 Desempenho do usuário *relay* utilizando o código RS(15,5). 45
3.14 Desempenho do usuário *relay* utilizando o código RS(255,85). . . 45

A.1 Codificador Reed Solomon RS(n,k). 52
A.2 Codificação RS na forma sistemática. 52

B.1 Ambiente do SIMULINK®. 55
B.2 Diagrama em blocos do terminal do usuário *relay*. 55

B.3 Bloco de codificação dos dados do usuário *relay*. 55

B.4 Diagrama em blocos da ERB. 56

Lista de Tabelas

2.1 Técnicas propostas para comunicação cooperativa. 10

2.2 Capacidade de correção do RS(15,5) em função da taxa da cooperação. 14

2.3 Taxas de cooperação e as respectivas capacidades de correção do código RS utilizado nas simulações. 25

2.4 Parâmetros utilizados nas simulações. 25

3.1 Capacidade de correção do RS(255,85) em função da taxa de cooperação. 36

3.2 Parâmetros utilizados nas simulações. 40

A.1 Características do código RS. 53

Lista de Abreviaturas e Siglas

AF	-	*Amplify and Forward*
AWGN	-	*Additive White Gaussian Noise*
BER	-	*Bit Error Rate*
BCH	-	*Bose, Chaudhuri e Hocquenghem*
BPSK	-	*Binary Phase-Shift Keying*
CAG	-	*Controle Automático de Ganho*
CC	-	*Coded Cooperation*
CF	-	*Compress and Forward*
COFDM	-	*Coded Orthogonal Frequency Division Multiplexing*
CRC	-	*Cyclic Redundancy Check*
CC-RS	-	*Codificação Cooperativa com Reed Solomon*
DF	-	*Decode and Forward*
DFT	-	*Discrete fourier Transform*
DSL	-	*Digital Subscriber Line*
ERB	-	*Estação Rádio-Base*
FEC	-	*Forward Error Control*
FFT	-	*Fast Fourier Transform*
IDFT	-	*Inverse Discrete fourier Transform*
IFFT	-	*Inverse Fast Fourier Transform*
LDPC	-	*Low-Density Parity-Check*
LTE	-	*Long Term Evolution*

MDS	-	*Maximum distance Separable*
OFDM	-	*Orthogonal Frequency Division Multiplexing*
QAM	-	*Quadrature Amplitude Modulation*
QoS	-	*Quality of Service*
RS	-	*Reed Solomon*
SNR	-	*Signal-to-Noise Ratio*
ST-CC	-	*Space Time Coded Cooperation*
TDMA	-	*Time Division Multiple Access*
WiMAX	-	*Worldwide Interoperability for Microwave Access*
Wi-Fi	-	*Wireless Fidelity*
2,5G	-	2^{nd} *Generation*
3G	-	3^{rd} *Generation*

Lista de Símbolos

δ	-	Função delta de Dirac.
ξ	-	Porcentagem de subportadoras acima do limiar de recepção.
ξ_{min}	-	Porcentagem mínima de subportadoras acima do limiar de recepção.
E_b	-	Energia média de *bit*
$(E_b/N_0)_l$	-	Limiar de recepção.
$(E_b/N_0)_f$	-	Relação sinal-ruído entre o usuário fonte e ERB.
$(E_b/N_0)_{fr}$	-	Relação sinal-ruído entre o usuário fonte e o usuário *relay*.
$(E_b/N_0)_r$	-	Relação sinal-ruído entre o usuário *relay* e a ERB.
f	-	Frequência
H	-	Resposta em frequência do canal.
H_n	-	Resposta em frequência da n-ésima subportadora.
h_f	-	Resposta impulsiva do canal do usuário fonte.
h_r	-	Resposta impulsiva do canal do usuário relay.
j	-	Índice da subportadora do símbolo OFDM.
k	-	Número de símbolos de mensagem do código RS.
m	-	Quantidade de *bits* que possui cada símbolo da palavra código.
n	-	Número de símbolos da palavra código.
N_0	-	Densidade espectral de potência do ruído AWGN.

N_s	-	Número de subportadoras que estão acima do limiar de recepção.
N_t	-	Número de subportadoras de um símbolo OFDM.
$P_{b_{cp}}$	-	Probabilidade de erro de *bit* de um sistema 16-QAM OFDM em um canal plano e variante no tempo.
$P_{b_{cs}}$	-	Probabilidade de erro de *bit* de um sistema 16-QAM OFDM em canal móvel e seletivo em frequência.
r	-	Variável aleatória com distribuição de Rayleigh.
t	-	Capacidade de correção de erros do código Reed Solomon.
t_{rsp}	-	Capacidade de correção de erros do código Reed Solomon puncionado.
T_c	-	Taxa de cooperação.
$T_{c_{max}}$	-	Máxima taxa de cooperação.
$U1$	-	Usuário *relay*.
$U2$	-	Usuário fonte.
UC	-	Unidade de Controle.
U_p	-	Quantidade de símbolos de paridadade do usuário fonte que estão contidos no quadro do usuário *relay*.
w_{ss}	-	Peso de Hamming do símbolo de sinalização.

Resumo

Este trabalho apresenta uma análise do desempenho do código Reed Solomon no sistema de codificação cooperativa. Esta análise é avaliada em canal plano e também em canal seletivo em frequência, ambos variantes no tempo. A técnica OFDM é empregada na transmissão dos dados dos usuários e tem a função de melhorar o desempenho do sistema em canais com desvanecimentos seletivo em frequência, bem como prover métodos que a ERB possa usar na estimação do canal de comunicação. O desempenho do usuário fonte é medido em função da taxa de cooperação e também da porcentagem de subportadoras do usuário *relay* que estão acima de um limiar de recepção.

Palavras-chave: Codificação Cooperativa, Diversidade Espacial, Reed Solomon, OFDM, Taxa de Cooperação, Taxa de Erro de Bit, Canal Seletivo Variante no Tempo.

Abstract

This work presents an analysis of the performance of Reed Solomon in a cooperative communication system. This analysis is evaluated in flat, time-varying and frequency selective channels. The OFDM technique is employed in the transmission of user data and has the function of improving system performance in frequency-selective fading channels, providing methods that the base station can use for the channel estimation. The source user performance is measured in terms of the cooperation ratio and in terms of the percentage of the subcarriers of the user relay that are above a reception threshold.

Keywords: Coded Cooperation, Space Diversity, Reed Solomon, OFDM, Cooperation Ratio, Bit Error Rate, Mobile Communications.

Capítulo 1

Introdução

No cenário atual das telecomunicações, é visível o crescimento acelerado das comunicações móveis [1]. Várias tecnologias estão surgindo para suprir a demanda dos usuários por serviços móveis com taxas de transmissão de dados cada vez maiores [2]. As tecnologias de 2,5G (2^{nd} *Generation*) e 3G (3^{rd} *Generation*) utilizadas pelas operadoras atualmente não conseguem oferecer serviços com qualidade aos clientes, devido as limitações na taxa de transferência de dados [3]. O LTE (*Long Term Evolution*) [4], que é uma evolução das atuais redes 3G existentes, vem se tornando uma das tecnologias mais promissoras para atender a crescente demanda do mercado consumidor [1]. Apesar destas novas tecnologias permitirem um aumento considerável das taxas de transmissão em ambientes móveis, a qualidade de serviço (QoS - *Quality of Service*) dos clientes que estão próximos da borda da área de cobertura poderá ser insatisfatória.

A busca pelo aumento da vazão dos atuais sistemas móveis nem sempre é possível empregando tecnologias com maior eficiência espectral. Cada vez mais, uma alta eficiência espectral está combinada com a utilização de canais com larguras de faixas mais amplas. O aumento da largura de faixa em ambientes de comunicações móveis resulta, inevitavelmente, em canais de comunicação com seletividade em frequência, o que degrada a qualidade dos enlaces, podendo até mesmo inviabilizar a recepção, independentemente do valor da relação sinal-ruído

de recepção.

A utilização de técnicas convencionais de diversidade [5] resulta em uma melhora no desempenho de sistemas móveis em canais seletivos em frequência e variantes no tempo. Porém, devido a limitações físicas dos sistemas móveis, a implementação de alguns esquemas de diversidade fica limitada. Um exemplo disso é a diversidade espacial, que tipicamente requer o uso de múltiplas antenas espaçadas entre si de unidades de comprimento de onda do sinal para garantir que os sinais recebidos estejam descorrelacionados [6][7].

A técnica de Comunicação Cooperativa [8]-[25] é uma alternativa para viabilizar a diversidade espacial em sistemas onde não é possível garantir o espaçamento adequado das antenas. O objetivo desta técnica em sistemas móveis é viabilizar a utilização do canal de comunicação de um usuário para auxiliar na transmissão das informações de outro usuário [11]. Em outras palavras, o usuário *relay*, ou usuário de retransmissão, cede parte de seus recursos de comunicação para a transmissão de dados do usuário fonte.

O fato do *relay* compartilhar parte dos seus recursos possibilita que uma parcela das informações do usuário fonte percorra dois caminhos diferentes, o que pode ser visto como uma diversidade espacial de transmissão, já que os canais dos usuários são independentes. Uma das técnicas de comunicação cooperativa mais empregadas utiliza o puncionamento da palavra código gerada por um codificador convolucional para proporcionar a cooperação entre os usuários. Este tipo de técnica recebe o nome de Codificação Cooperativa [16]-[18].

A codificação cooperativa é uma técnica que possibilita que a palavra código do usuário *relay*, gerada a partir de um código corretor de erro (FEC - *Forward Error Control*) [26][27], seja puncionada e estruturada antes da transmissão. O usuário *relay* descarta *bits* de paridade para que *bits* de paridade do usuário fonte sejam inseridos.

Os sistemas de codificação cooperativa convencionais não levam em consi-

deração as condições em que se encontra o canal de comunicação do usuário *relay* no processo de cooperação, impossibilitando atingir a máxima eficiência do sistema cooperativo e também para a queda proibitiva do seu desempenho. A utilização da informação sobre as condições do canal de comunicação dos usuários no processo de cooperação permite produzir ganhos mais significativos para o usuário fonte, sem que a comunicação do usuário *relay* seja proibitivamente degradada.

A técnica de multiplexação por divisão em frequências ortogonais (OFDM - *Orthogonal Frequency Division Multiplexing*) [28] é largamente utilizada nos sistemas de comunicações móveis atuais, como é o caso do LTE, WiMAX [29] e Wi-FI [30]. Uma das principais vantagens do OFDM é a elevada robustez frente a canais seletivos em frequência [31][32]. A simplicidade do processo de equalização e estimação de canal também permite que o OFDM possa ser utilizado com eficiência em conjunto com a codificação cooperativa, uma vez que a estimativa da resposta em frequência do canal pode ser facilmente obtida.

Os objetivos deste trabalho são: (i) apresentar uma análise da aplicabilidade do código Reed Solomon na técnica de codificação cooperativa e; (ii) propor a integração da técnica de codificação cooperativa com o OFDM, utilizando o processo de estimação de canal para considerar a degradação do canal do usuário *relay* na decisão de cooperação. A análise das condições do canal de comunicação entre o usuário *relay* e a ERB (Estação Rádio-Base) e o sucesso na decodificação dos dados do usuário fonte serão os parâmetros determinantes na decisão pela cooperação. O usuário *relay* é autorizado a cooperar com o usuário fonte apenas se o número de subportadoras que apresentam elevada relação sinal-ruído esteja acima de um limiar pré-definido. Este limiar é determinado em função de valores que permitam o usuário *relay* operar com uma QoS aceitável ao tipo de serviço utilizado.

Para se alcançar os objetivos propostos, este texto está organizado da seguinte forma: o Capítulo 2 apresenta a aplicabilidade do código Reed Solomon no sis-

tema de codificação cooperativa, onde é mostrado o desempenho da codificação cooperativa em função da variação da taxa de cooperação. No Capítulo 3 é feita uma análise do uso do OFDM no sistema de codificação cooperativa, mostrando como esta técnica pode aumentar a eficiência do sistema cooperativo em canais seletivos em frequência. Ainda no Capítulo 3 são explanados os parâmetros utilizados pela ERB para caracterização do canal de comunicação do usuário *relay*. Estes parâmetros são determinantes para a ERB autorizar o usuário *relay* a cooperar com o usuário fonte. Além disso, é explorado o ganho de desempenho que o usuário fonte tem com a cooperação em ambientes com desvanecimento seletivo em frequência, bem como qual a queda do desempenho que o *relay* pode sofrer com o processo de cooperação. Finalmente, o Capítulo 4 traz as conclusões dos resultados obtidos com o sistema de codificação cooperativa operando com o código Reed Solomon em ambientes com desvanecimento seletivo em frequência e algumas sugestões de temas para trabalhos futuros.

Capítulo 2

Princípios da comunicação cooperativa

O propósito da comunicação cooperativa é compartilhar os recursos de transmissão de um ou mais usuários que estejam em condições favoráveis para auxiliar outro que esteja em condições ruins. O usuário que é beneficiado pela cooperação é denominado Fonte e os usuários que o ajudam são denominados *Relay*. Deste modo, a cooperação entre os usuários possibilita que parcela da informação do usuário fonte possa ser transmitida por mais de um canal de comunicação, sendo estes estatisticamente independentes. Este trabalho se limita ao estudo de técnicas de comunicação cooperativa que utilizam *relays* como caminhos alternativos. Neste caso, o canal de comunicação por cada *relay* se torna um caminho para trafegar as informações do usuário fonte.

Este processo pode ser interpretado como uma diversidade espacial, já que a ERB irá receber a informação proveniente do usuário fonte e dos demais canais de comunicação *relay*. Dentre as técnicas de comunicação cooperativas existentes, destacam-se as técnicas SF (*Store and Forward*) [10], AF (*Amplify and Forward*) [11], DF (*Decode and Forward*) [11][12], CF (*Compress and Forward*) [10][13], ST-CC (*Space Time Coded Cooperation*) [10]-[14] e CC (*Coded Cooperation*) [16]-[18]. As particularidades de cada técnica mencionada serão tratadas nas seções

seguintes.

2.1 Técnica SF

Esta técnica é empregada tipicamente em estruturas de redes Ethernet, onde há possibilidade de tráfego da informação por vários caminhos diferentes. A Figura 2.1 ilustra um cenário típico utilizado em ambientes de redes.

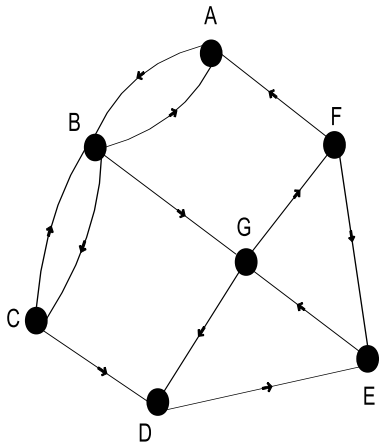


Figura 2.1: *Ilustração de um cenário típico da técnica SF utilizado em redes.*

Os nós A-G, usualmente são representados por roteadores, pontos de acesso sem fio, ou mesmo telefones celulares. As conexões entre os nós representam os canais de comunicação, como por exemplo, fibra óptica ou sistema sem fio. Cada equipamento que tem a funcionalidade de roteador ou nó da rede pode ser um *relay* nesta técnica. O *switch* por exemplo opera como um *relay* em um sistema de roteamento de rede. Ele recebe e armazena um fluxo de pacotes de dados, que são destinados a diferentes usuários. Cada pacote é analisado para verificar se não contém erros. O processo utilizado neste caso é a análise do *checksum* [33] do código CRC (*Cyclic Redundancy Check*)[33] de cada pacote. Se nesta análise for identificado que o pacote contém erros, ele é descartado. Caso contrário, ele é encaminhado pelo *relay*.

Nesta técnica, o *relay* deve receber todo o pacote de dados antes de fazer o encaminhamento, o que torna este sistema lento comparado a outras técnicas que utilizam o encaminhamento de alguns *bits*, símbolos ou mesmo parte do sinal recebido. A vantagem desta técnica, está em possibilitar que os usuários possam se cooperar mutuamente, uma vez que o pacote de dados recebido contém informações de vários usuários. Isto permite que cada usuário possa simultaneamente ser o usuário fonte e um usuário *relay* para os demais.

2.2 Técnica AF

Na técnica de cooperação AF, o *relay* recebe a forma de onda enviada pelo usuário fonte, amplifica-a e retransmite-a sem realizar nenhuma técnica de regeneração do sinal. Desta maneira, a implementação deste tipo de processo cooperativo se torna mais simples.

Entretanto, o ganho de desempenho deste sistema fica restrito a cenários onde não existam degradações severas do sinal pelo meio de comunicação, dado que não é feita nenhuma tentativa para recuperação da informação antes da retransmissão. A Figura 2.2 ilustra o princípio de funcionamento da técnica AF.

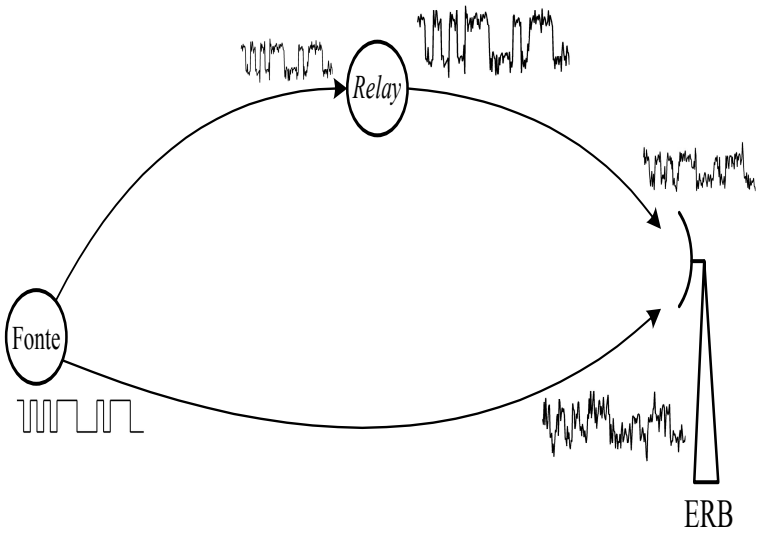


Figura 2.2: *Ilustração da técnica de comunicação cooperativa AF.*

2.3 Técnica DF

A técnica DF é semelhante a técnica AF, porém o sinal recebido pelo *relay* é regenerado antes de ser retransmitido à ERB. A Figura 2.3 ilustra o princípio de funcionamento desta técnica. O usuário fonte transmite sua informação à ERB e uma versão ruidosa desta informação também é recebida pelo *relay*. Diferentemente da técnica AF, o *relay* faz o tratamento da informação recebida antes da cooperação, ou seja, a mensagem ao ser recebida pelo *relay* é decodificada, recodificada e retransmitida pelo seu canal de comunicação.

A cooperação só é válida mediante a decodificação com sucesso das informações do usuário fonte. A estação rádio-base recebe os sinais provenientes de ambos os canais de comunicação. Neste processo é possível conseguir um desempenho superior a técnica AF, pois a informação é regenerada antes da retransmissão [11].

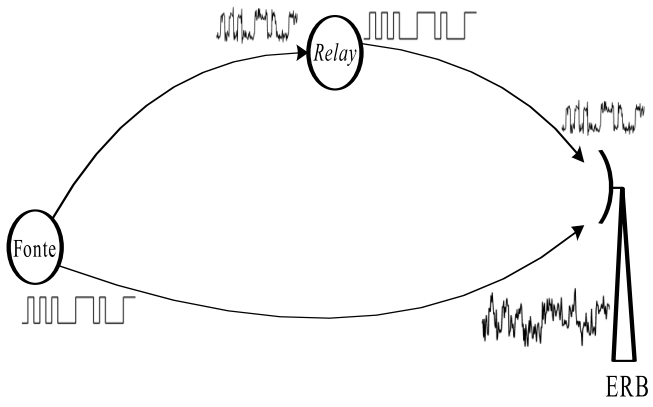


Figura 2.3: *Ilustração da técnica de comunicação cooperativa DF.*

2.4 Técnica CF

Na técnica CF, o usuário *relay*, ao receber o sinal do usuário fonte, só faz o processo de demodulação do sinal. Não é realizada a decodificação dos dados.

Os dados oriundos da demodulação são amostrados, quantizados, comprimidos e retransmitidos. Está técnica não possui como característica a regeneração da informação, mas possibilita que uma amostra digitalizada do sinal recebido do usuário fonte seja retransmitida pelo *relay*. A informação do *relay* ao chegar ao destino é demodulada para extração das informações do usuário fonte. Em seguida é realizado o processo de descompressão dos dados para serem processados em conjunto com o sinal enviado pelo usuário fonte.

2.5 Técnica ST-CC

Esta técnica utiliza a codificação cooperativa aliada às tecnologias de codificação espaço-temporal [34].

O princípio de funcionamento desta técnica pode ser dividido em duas fases. Na primeira fase o usuário fonte transmite para o destino e para os potenciais usuários *relay*. Na segunda fase os usuários *relay* transmitem para o destino, em subcanais ortogonais, no caso de algoritmos de diversidade cooperativa baseada em repetição, ou nos mesmos subcanais nos algoritmos baseados em diversidade cooperativa com codificação espaço-temporal. O desempenho do sistema utilizando está técnica pode ser maximizado pela escolha adequada dos códigos espaços-temporais [35]. Quando estes códigos são combinados com os códigos corretores de erro (FEC), ganhos de desempenho são obtidos durante o processo de cooperação. A Figura 2.4 ilustra o princípio de funcionamento desta técnica.

2.5.1 Técnica CC

A técnica CC utiliza os conceitos da regeneração dos dados do usuário fonte antes da retransmissão. Assim como na DF, a informação do usuário fonte é analisada antes da cooperação, sendo que o *relay* só coopera se a informação do usuário fonte foi decodificada corretamente. Neste caso, a utilização de FEC se faz necessária.

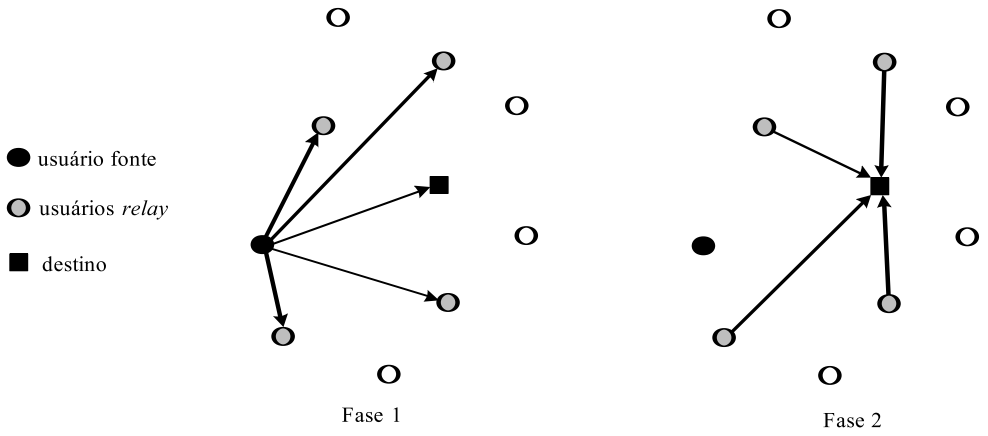


Figura 2.4: *Ilustração das duas fases empregadas pela técnica ST-CC.*

A vantagem desta técnica em relação a DF está no compartilhamento da palavra código dos usuários, o que acarreta no aumento da eficiência do processo de cooperação. A Tabela 2.1 mostra um comparativo entre as técnicas utilizadas na comunicação cooperativa.

Tabela 2.1: *Técnicas propostas para comunicação cooperativa.*

Técnicas	Regeneração da informação	Possibilita a diversidade	Esquema de codificação
Store-and-Forward (SF)	sim	não	-
Amplify-and-Forward (AF)	não	sim	-
Compress-and-Forward (CF)	não	sim	Compressão
Decode-and-Forward (DF)	sim	sim	Repetição
Coded Cooperation (CC)	sim	sim	FEC
Space Time CC (SF)	sim	sim	Space-time & FEC

A técnica de codificação cooperativa se baseia em compartilhar uma parcela dos *bits* do usuário *relay* para transmitir os *bits* do usuário fonte. O objetivo desta técnica é possibilitar que uma parcela dos *bits* do usuário fonte possa trafegar por um canal de comunicação que possui melhores condições de transmissão. Para isto, o usuário *relay* segmenta seu quadro, onde alguns *bits* são retirados para que *bits* do usuário fonte sejam inseridos. Uma das técnicas que visam explorar o uso desse sistema é apresentada em [16]-[18], e utiliza o conceito de compartilhar

os *bits* de paridade do usuário *relay* dado que os *bits* recebidos do usuário fonte foram decodificados com sucesso. A Figura 2.5 ilustra a estrutura de um quadro com e sem o processo de cooperação.

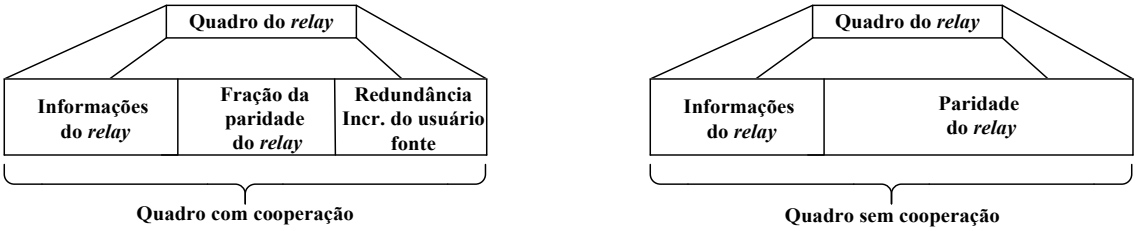


Figura 2.5: *Estrutura dos quadros com e sem o processo de codificação cooperativa.*

Em uma das possíveis formas de codificação cooperativa, a codificação da mensagem de ambos os usuários é realizada por um codificador convolucional [26][27], onde o *relay* realiza o puncionamento de sua sequência codificada, visando inserir *bits* de redundâncias do usuário fonte. Este esquema demanda a utilização de outro código verificador de paridade para permitir que o terminal do *relay* possa certificar-se de que as informações oriundas do usuário fonte foram recebidas corretamente. O código verificador utilizado é o CRC. A verificação do resultado da decodificação do usuário em [16]-[18] é obtida pela análise do *checksum* da informação recebida do usuário fonte.

O esquema de codificação cooperativa utilizado por [16]-[18] punciona a palavra código gerada por um codificador convolucional para inserir *bits* do usuário fonte. A informação que o *relay* detém para optar em cooperar ou não é o resultado da verificação do *checksum* das informações recebidas do usuário fonte. Caso o resultado seja positivo, o usuário *relay* cede parte dos seus *bits* para transmitir uma parcela dos *bits* do usuário fonte.

Embora os resultados apresentem um ganho de desempenho elevado, em alguns cenários chegando a valores em torno de 15 dB, os sistemas demonstrados em [16]-[18] apresentam algumas desvantagens, seja pelo fato de terem que utilizar um outro código concatenado com o convolucional ou pelo processo de cooperação não levar em conta as condições dos canais de comunicação dos usuários. Mo-

tivado pelos resultados alcançados com as técnicas convencionais de codificação cooperativa e ao mesmo tempo buscando alternativas para tornar o processo de cooperação mais eficiente, surgiu uma nova proposta de técnica de codificação cooperativa, à técnica CC-RS (*Codificação Cooperativa com Reed Solomon*).

2.6 Técnica proposta de codificação cooperativa CC-RS

A técnica CC-RS utiliza um único FEC tanto para codificação de canal quanto para prover métodos que permitam identificar quando uma informação recebida do usuário fonte foi decodificada com sucesso.

O código empregado neste sistema é o Reed Solomon (RS)[27], que possui características apropriadas para o sistema de codificação cooperativa, pois este código permite identificar se o número de erros por decisão abrupta na palavra-código recebida ultrapassa a sua capacidade de correção. Com isso não é necessário utilizar outro código detector de erro para que o usuário *relay* possa identificar se é possível realizar a cooperação. Isto possibilita utilizar um único código para agregar robustez aos dados do usuário e ainda extrair a informação do *status* da decodificação do vetor recebido, necessária para auxiliar o usuário *relay* durante o processo de cooperação. Outra característica interessante do RS está na facilidade em puncionar a palavra código durante o processo de cooperação [27]. Com a utilização do Reed Solomon é possível variar a taxa de cooperação, buscando uma relação de compromisso entre o número de símbolos que o usuário *relay* cede ao usuário fonte e o ganho que o usuário fonte tem com a cooperação.

A taxa de cooperação é dada por

$$T_c = \frac{U_p}{n}, \tag{2.1}$$

onde U_p é a quantidade de símbolos que foram cedidos no processo de cooperação

e n a quantidade total de símbolos do usuário *relay*. A Figura 2.6 ilustra esses parâmetros no quadro do usuário *relay*.

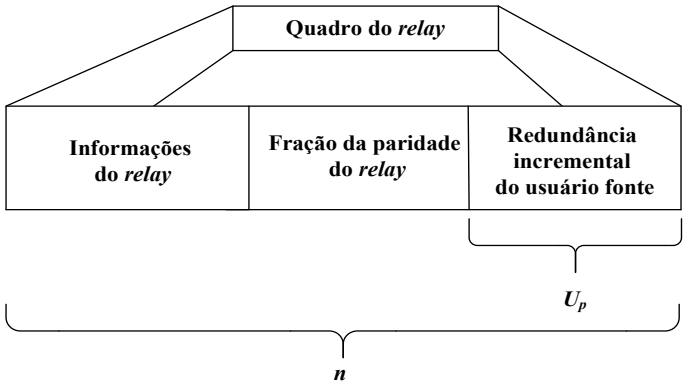


Figura 2.6: *Parâmetros envolvidos no cálculo da taxa de cooperação.*

O valor mínimo da taxa de cooperação, correspondente a não haver cooperação, é zero. Já o valor máximo é determinado por

$$T_{c_{max}} = \frac{n-k}{n}, \tag{2.2}$$

onde k é o número de símbolos da mensagem do usuário *relay*. Obviamente, o valor máximo da taxa de cooperação é obtido quando todos os símbolos de paridade do usuário *relay* são descartados para se transmitir os símbolos de paridade do usuário fonte. Os símbolos cedidos pelo usuário *relay* ao usuário fonte serão tratados como símbolos errados pelo decodificador do usuário *relay*. Isso significa que erros de símbolo são introduzidos sempre que a taxa de cooperação é maior do que zero. De fato, a capacidade de correção do código empregado pelo *relay* será dada por

$$t_{rsp} = \frac{n-k}{2} - U_p, \tag{2.3}$$

A Tabela 2.2 mostra os valores de T_c para um sistema cooperativo utilizando o RS(15,5).

A Tabela 2.2 mostra que para valores de $T_c \geq 0,333$ o decodificador já não con-

Tabela 2.2: *Capacidade de correção do RS(15,5) em função da taxa da cooperação.*

Código principal	Código Puncionado	Capacidade de correção de erros	Taxa de cooperação
RS(15,5)	sem puncionamento	5 símbolos	0
	RS(12,5)	2 símbolos	0,2
	RS(11,5)	1 símbolo	0,27
	RS(10,5)	0 símbolos	0,333
	RS(9,5)	0 símbolos	0,4
	RS(8,5)	0 símbolos	0,467
	RS(7,5)	0 símbolos	0,533
	RS(6,5)	0 símbolos	0,6
	RS(5,5)	0 símbolos	0,667

segue fazer a correção dos dados do usuário *relay*, caso venha sofrer degradações no canal de comunicação. O impacto disto no sistema CC-RS é a piora de desempenho do usuário *relay* e a melhora do desempenho do usuário fonte, já que é possível utilizar mais símbolos para cooperação. O ganho máximo de desempenho do usuário fonte é obtido quando todos os símbolos de paridade do *relay* estão sendo disponibilizados para transmissão dos seus símbolos, que neste caso corresponde a trabalhar com $T_c = 0,667$.

Portanto, utilizar valores de T_c que tornam os decodificadores incapazes de fazer a correção dos dados pode contribuir para o aumento de desempenho do usuário fonte, mas pode acabar prejudicando o desempenho do usuário *relay*. Assim, a escolha por qual T_c utilizar acaba sendo uma solução de compromisso, realizada em função das condições do canal de comunicação.

2.7 Análise de desempenho da técnica CC-RS

O objetivo desta análise é comprovar a eficácia do código Reed Solomon no sistema de codificação cooperativa. O desempenho do usuário fonte será explorado em função da taxa de cooperação utilizada pelo *relay*. Não será analisado o efeito da cooperação no desempenho do usuário *relay*.

2.7.1 Cenário proposto

O cenário proposto consiste num sistema de comunicação móvel operando com dois usuários. No sistema cooperativo, a cooperação só trará benefícios quando o usuário *relay* estiver em melhores condições de transmissão que o usuário fonte.

A relação sinal-ruído média entre o usuário *relay* e a ERB é Δ dB maior do que a relação sinal-ruído entre o usuário fonte e a ERB. A Figura 2.7 ilustra o ambiente utilizado para mensurar o desempenho do sistema cooperativo com o código RS. O software utilizado nas simulações foi o SIMULINK®, que possui algumas funcionalidades já implementadas, como o caso dos codificadores, dos decodificadores e do canal utilizado nas simulações. Os blocos responsáveis pela estruturação do quadro na transmissão, pela reestruturação dos quadros na recepção e pelo controle da cooperação no sistema foram implementados utilizando as ferramentas do próprio SIMULINK®. Mais detalhes sobre está ferramenta podem ser encontradas nos Anexos B e C.

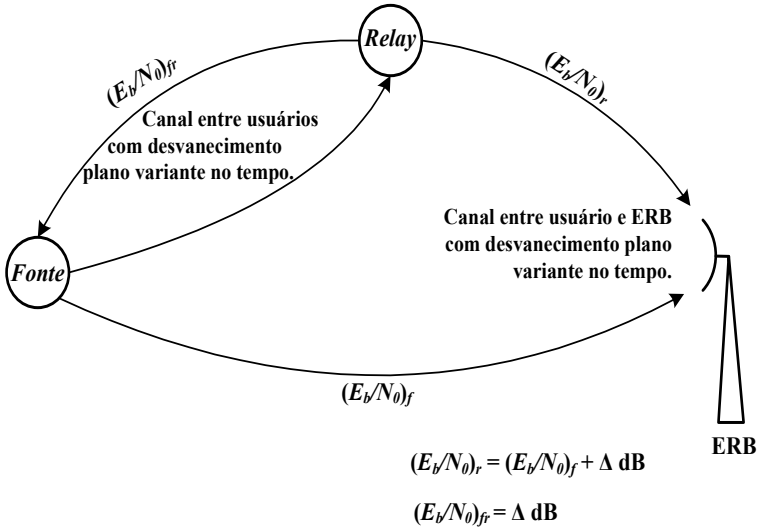


Figura 2.7: *Cenário proposto nas simulações.*

Nas simulações propostas, o valor da $(E_b/N_0)_{fr}$ média é constante. Assume-se que a tecnologia utilizada no sistema móvel é a TDMA (TDMA - *Time Division Multiple Access*) [36], onde cada usuário ocupa uma janela de tempo diferente. A Figura 2.8 ilustra a estrutura das janelas de tempo durante a transmissão dos

usuários fonte e *relay*.

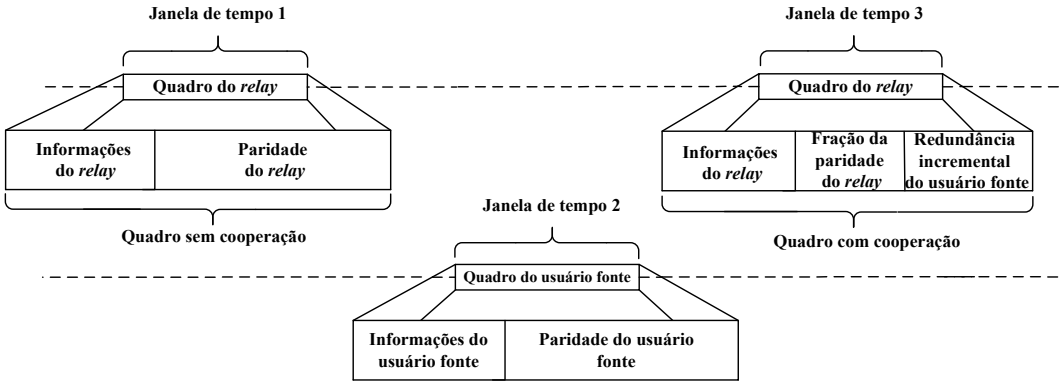


Figura 2.8: *Codificação cooperativa no sistema TDMA.*

Na Figura 2.8 pode-se visualizar que na janela de tempo onde ocorre a cooperação, o *relay* segmenta seu quadro e adiciona parcela da informação do usuário fonte. Na janela de tempo que não tem cooperação ele só envia suas informações.

Com o intuito de avaliar o desempenho desta proposta, implementou-se um sistema baseado em portadora única e mapeamento BPSK (BPSK - *Binary Phase Shift Keying*) [39] para os enlaces entre os usuários e a ERB. O valor de $(E_b/N_0)_{fr}$ foi fixado em 20 dB, visando analisar a situação em que o usuário *relay* se encontra mais próximo a ERB do que o usuário fonte. O canal foi modelado com uma distribuição Rayleigh com desvio Doppler [6] igual a

$$f_{Doppler} = \frac{1}{10 \times Ts}, \tag{2.4}$$

onde Ts é o tempo de símbolo na saída do modulador.

O código utilizado foi o RS(15,5), cujas taxas de cooperação são mostradas na Tabela 2.2. Um CAG (CAG - Controle automático de ganho) foi utilizado na recepção para prover um nível constante na entrada do demodulador, uma vez que ocorrem variações de amplitude do sinal recebido, dado pelo comportamento do sistema operando em canais móveis. Este controle não altera a relação sinal-ruído na entrada do receptor.

O *relay*, ao fazer a opção pela cooperação, deve estruturar seu quadro de

forma a permitir que sejam inseridos os símbolos do usuário fonte. A ERB, por sua vez, deve reestruturar este quadro ao recebê-lo, garantindo que os símbolos do usuário fonte sejam retirados do quadro do *relay* antes da decodificação. As subseções a seguir descrevem como são realizados este dois processos.

2.7.2 Estruturação do quadro na transmissão

O usuário *relay*, ao receber os dados do usuário fonte, irá fazer o processo de decodificação desses dados. Em caso de sucesso, o usuário *relay* segmenta o seu quadro e insere alguns símbolos de paridade do usuário fonte.

A quantidade de símbolos do usuário fonte retransmitidos pelo usuário *relay* durante a cooperação depende da T_c atribuída ao sistema. Neste cenário, a única premissa utilizada pelo *relay* para fazer a opção pela cooperação é a informação do *status* da decodificação das informações do usuário fonte. No caso de haver sucesso na decodificação, o *relay* retira parte dos seus símbolos de paridade e insere os símbolos de paridade do usuário fonte. Caso ocorra erro na decodificação, o *relay* não segmenta o quadro e transmite somente suas informações. Todo o controle do fluxo de dados recebido após a codificação é realizado pelo bloco denominado "Unidade de Controle"

A Figura 2.9 ilustra o diagrama em blocos da etapa de transmissão de ambos os usuários. O usuário *relay* ao receber os dados do usuário fonte faz a demodulação e posteriormente a remoção do símbolo de sinalização. Os dados são decodificados e uma informação sobre o *status* desta decodificação é entregue a "Unidade de Controle". No caso de sucesso a "Unidade de Controle"concatena uma parcela dos dados codificados do usuário *relay* com uma parcela dos dados codificados do usuário fonte, acrescentando ao final deste processo um símbolo a mais no quadro a ser transmitido, cujo objetivo é informar se o dados enviados pelo usuário *relay* também contém dados do usuário fonte. Após esta estruturação, os dados são modulados e transmitidos. Em função da canal de comunicação, um usuário pode

assumir em determinado instante a função de usuário fonte ou de usuário *relay*. Portanto, o diagrama em blocos dos terminais dos usuários são similares.

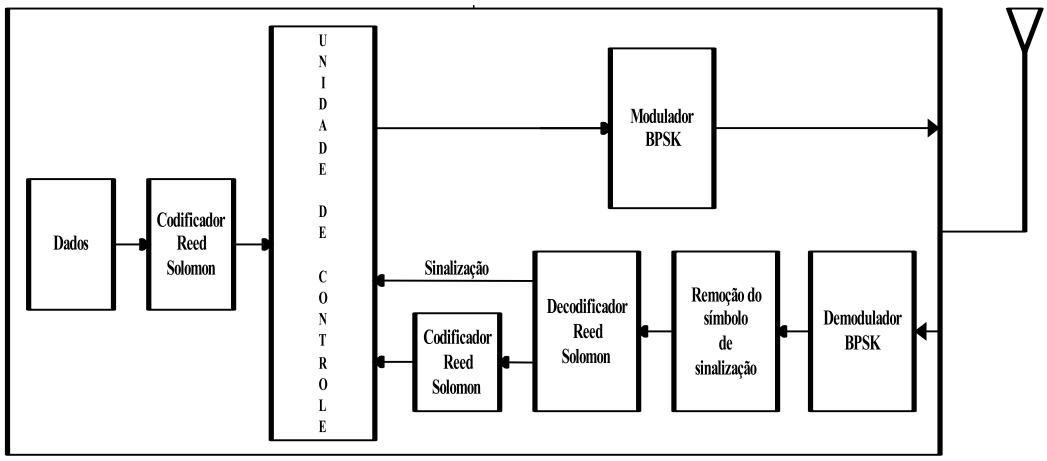


Figura 2.9: *Diagrama em blocos do terminal do usuário.*

A ERB, ao receber o quadro enviado pelo *relay* deve ser capaz de reestruturá-lo, retirando os símbolos que pertencem ao usuário fonte. Dois métodos foram explorados para possibilitar a implementação deste processos: método não sinalizado e método sinalizado. Tais métodos são descritos a seguir.

2.7.3 Método não sinalizado

Neste método não é adicionada nenhuma informação para diferenciar um quadro com cooperação de outro sem cooperação. O usuário *relay*, quando está cooperando, simplesmente segmenta seu quadro e insere os símbolos do usuário fonte. Na recepção é impossível a ERB identificar se um determinado usuário está cooperando ou não. Logo, a ERB sempre monta dois quadros distintos para decodificação, um contendo as informações provenientes do caminho principal e outro do caminho *relay*. O princípio deste método consiste em gerar sempre dois

quadros distintos de cada usuário, um contendo somente os símbolos proveniente do caminho principal e outro contendo símbolos do caminho principal mais os símbolos do caminho *relay*. No caso da não cooperação, um dos quadros será formado por uma parcela dos símbolos do usuário fonte e outra com os símbolos do usuário *relay*, como ilustrado na Figura 2.10. Nesta figura, as denotações de $U1$ e $U2$ representam os símbolos de informação dos usuários *relay* e fonte respectivamente. A denotação $U1$' representa os símbolos de paridade do usuário *relay* e as denotações $U2$' e $U2$"foram utilizadas para representar os símbolos de paridade do usuário fonte. Sendo que, $U2$' representa os símbolos de paridade do usuário fonte transmitidos por ele e $U2$"representa os símbolos de paridade do usuário fonte que foram retransmitidos pelo usuário *relay*. O bloco "UC"é quem faz o processo de reestruturação dos quadros de ambos os usuários.

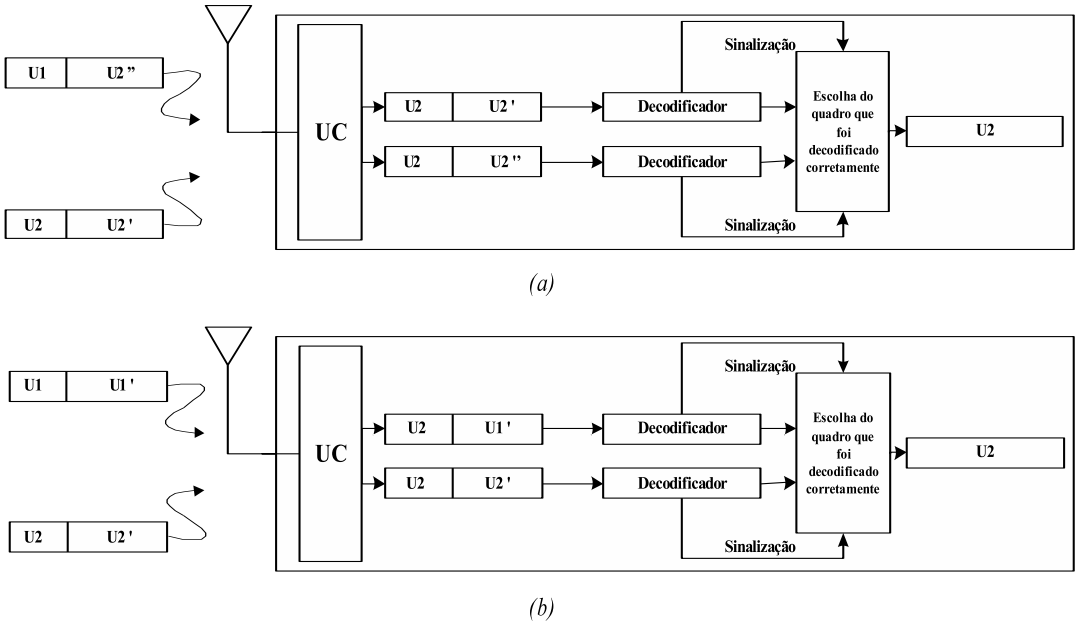


Figura 2.10: *Método não sinalizado: (a) Etapa de recepção do usuário fonte com o processo de cooperação. (b) Etapa de recepção do usuário fonte sem o processo de cooperação.*

A ausência de informações sobre qual quadro está cooperando torna idênticos

os sistemas de reestruturação. Assim como na transmissão, a única informação que a ERB tem disponível para fazer a reestruturação na recepção é a taxa de cooperação utilizada no sistema.

Devido a este tipo de reestruturação pode ocorrer aumento na taxa de erro de decodificação das informações do usuário fonte, quando o sistema opera com valores elevados de T_c e não utiliza o processo de cooperação. A consequência disto é a queda de desempenho. Isto se acentua quando o usuário *relay* está em melhores condições de transmissão que o usuário fonte, mas não coopera com ele. A Figura 2.11 ilustra um cenário onde há maior probabilidade de não cooperação. Nesse cenário, há uma grande probabilidade de que os símbolos de paridade do usuário *relay* cheguem ao receptor com menos erros do que os símbolos de paridade recebidos do usuário fonte.

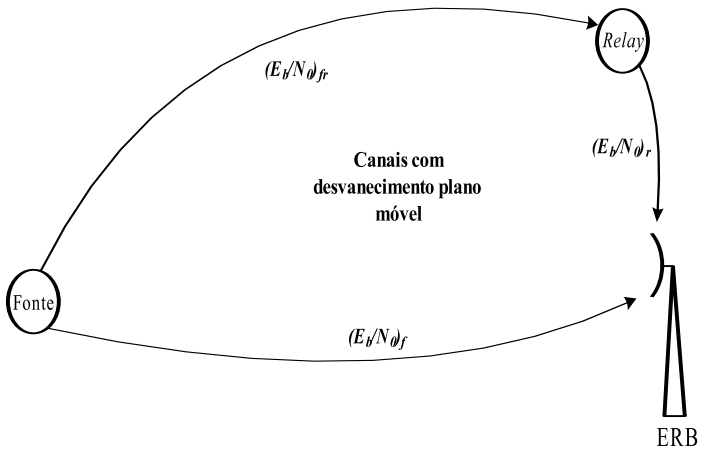


Figura 2.11: *Cenário onde há maior probabilidade de não cooperação.*

A ERB, ao decodificar o quadro do usuário fonte, que fora formado com os símbolos de paridade do usuário *relay* mais os símbolos do usuário fonte, pode encontrar símbolos de mensagens que não pertençam a nenhum dos dois usuários. Estes símbolos de mensagem são interpretados como erro, já que são resultantes da decodificação de outra palavra-código. A Figura 2.12 ilustra o processo de decodificação dos dados do usuário fonte em que se produz erro. Os usuários

relay e fonte são representados por $U1$ e $U2$ respectivamente.

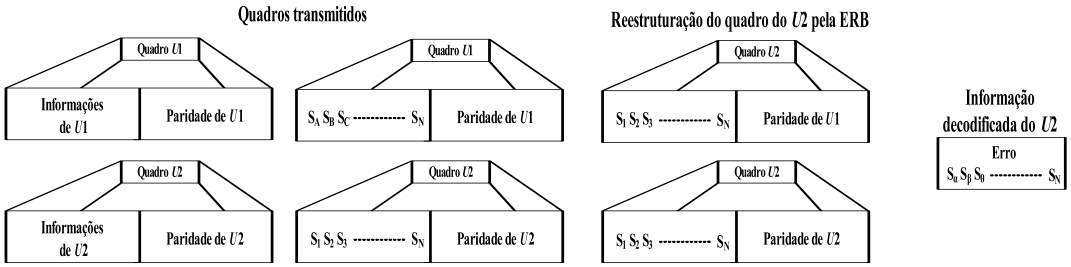


Figura 2.12: *Possibilidade do resultado da decodificação do quadro do usuário fonte no sistema não sinalizado e sem o processo da cooperação.*

No método em questão, a informação da T_c não é suficiente para que a ERB possa decodificar de forma inequívoca os dados do usuário fonte. Devido a grande ocorrência de erros que este método pode resultar quando o usuário *relay* não está cooperando com o usuário fonte, a solução foi usar um método que permitisse sinalizar à ERB quando houvesse cooperação.

2.7.4 Método sinalizado

No método sinalizado, uma informação adicional é inserida no quadro antes da transmissão. Essa informação utiliza um símbolo a mais no quadro dos usuários, permitindo que a ERB possa reestruturar os quadros de forma inequívoca. O símbolo adicionado é chamado de Símbolo de Sinalização, cuja função é estritamente informar à ERB se o *relay* está ou não está cooperando com o usuário fonte. Neste caso, a ERB passa a contar com duas informações para reestruturar os quadros recebidos, sendo uma correspondente à T_c definida no sistema e outra contendo a informação do símbolo de sinalização. O conteúdo deste símbolo está vinculado ao usuário optar ou não pela cooperação. No caso do usuário *relay* optar em fazer a cooperação, o símbolo é preenchido com *bits* uns. Caso a opção seja pela não cooperação, o símbolo é preenchido com *bits* zeros. Os valores foram escolhidos de forma a permitir que o símbolo indicando cooperação tenha a maior

distância de Hamming [27] possível de outro indicando a não cooperação. Qualquer outra combinação que maximizasse essa distância poderia ser utilizada. A Figura 2.13 ilustra como fica a estrutura do quadro com o símbolo de sinalização. Nesta Figura é visto que a quantidade de *bits* do símbolo de sinalização depende

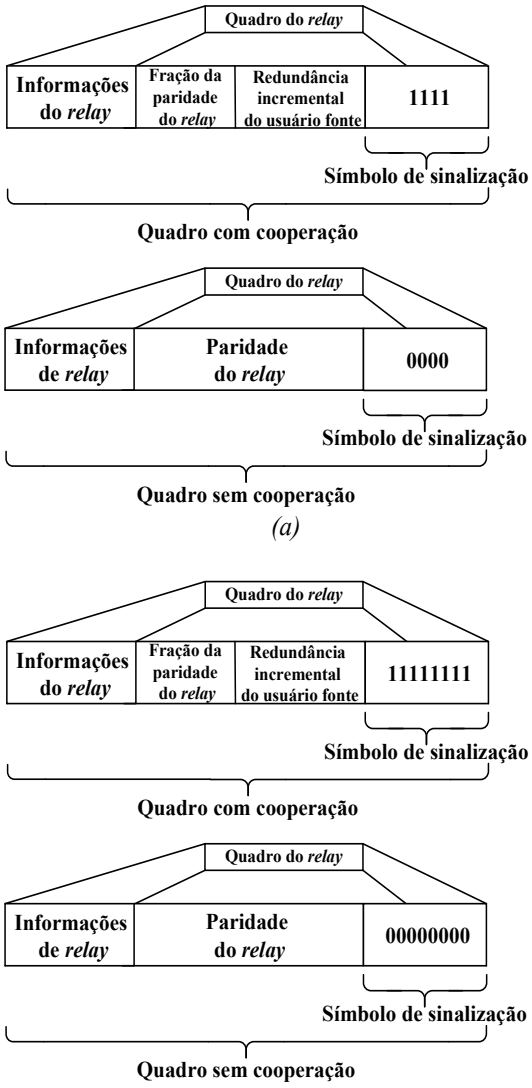


Figura 2.13: *(a) Valores atribuídos ao símbolo de sinalização nos quadros com cooperação e sem cooperação para o código RS(15,5) (a) e para o código RS(255,85) (b).*

do código RS que está sendo empregado.

Ao longo do sistema de transmissão e recepção, o símbolo de sinalização também estará sujeito a degradações provenientes do canal de comunicação. Estas

degradações podem alterar o conteúdo deste símbolo, de tal forma que ao chegar ao receptor contenha combinações diferentes das utilizadas na transmissão. Na recepção, a ERB não emprega nenhuma técnica para correção deste símbolo. O que ela faz é uma análise do conteúdo, de forma que seja possível distinguir um quadro com cooperação de outro sem cooperação. No sistema proposto, a ERB faz a distinção de um quadro com cooperação de outro sem cooperação pela análise do peso de Hamming [27] do símbolo de sinalização, denotado aqui como w_{ss}. Se $w_{ss} \geq m/2$, o quadro é com cooperação. Caso contrário, o quadro é classificado como sem cooperação. O valor de m é a quantidade de *bits* de um símbolo RS. Por exemplo, se o sistema estiver operando com um código RS(15,5) e a ERB receber um quadro cujo símbolo de sinalização seja igual a "0001", ele será classificado como sem cooperação, pois $w_{ss} = 1 < \frac{m}{2} = \frac{4}{2} = 2$.

Apesar da inserção do símbolo de sinalização resultar em um acréscimo da vazão do sistema, ele auxilia a ERB no processo de reestruturação do quadro na recepção e impede que sejam decodificadas mensagens que não correspondam as mensagens transmitidas pelo usuário fonte. A Figura 2.14 ilustra como fica a disposição final do quadro transmitido após a inserção do símbolo de sinalização.

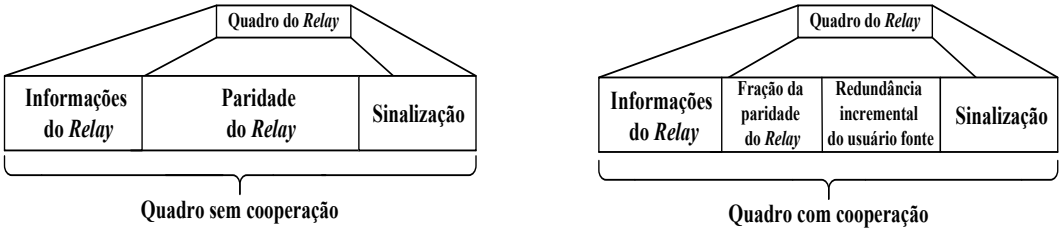


Figura 2.14: *Disposição dos quadros do relay com cooperação e sem cooperação.*

Após a análise do símbolo de sinalização, os quadros do usuário fonte são rearranjados e decodificados. Caso o *relay* esteja cooperando, a ERB disponibiliza dois quadros do usuário fonte para serem decodificados, um composto pelas informações que vieram diretamente do usuário fonte e outro formado por uma parcela das informações provenientes do usuário *relay* mais uma parcela proveniente do usuário fonte. Após este processo os quadros são decodificados de forma

que a informação escolhida do usuário fonte é a que teve sucesso após a decodificação. A informação que não foi decodificada com sucesso é descartada. No caso do *relay* não fazer a opção pela cooperação, um único quadro é decodificado. A Figura 2.15 ilustra como a ERB faz a reestruturação do quadro do usuário fonte na recepção.

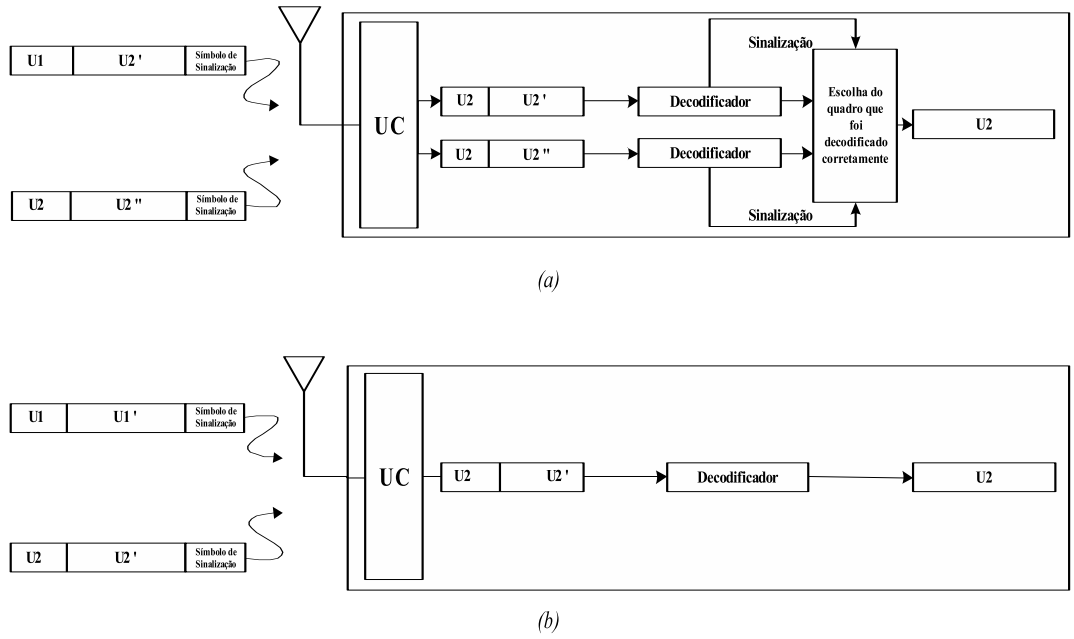


Figura 2.15: *Método sinalizado: (a) Etapa de recepção do usuário fonte com o processo de cooperação. (b) Etapa de recepção do usuário fonte sem o processo de cooperação.*

2.8 Análise dos resultados das simulações

A técnica de codificação cooperativa CC-RS apresentada neste Capítulo é analisada para diferentes taxas de cooperação. A Tabela 2.3 mostra os valores das taxas de cooperação utilizadas na análise da técnica CC-RS e as respectivas capacidades de correção do código RS empregado.

Nesta seção é apresentada a análise de desempenho em função da taxa de erro de *bit*, obtida por simulações computacionais, conforme descrito sa Seção

Tabela 2.3: *Taxas de cooperação e as respectivas capacidades de correção do código RS utilizado nas simulações.*

Código principal	Código Puncionado	Capacidade de correção de erros	Taxa de cooperação
RS(15,5)	sem puncionamento	5 símbolos	0
	RS(12,5)	2 símbolos	0,2
	RS(11,5)	1 símbolo	0,27
	RS(10,5)	0 símbolos	0,333
	RS(5,5)	0 símbolos	0,667

2.7.1. Os parâmetros das simulações são mostrados na Tabela 2.4.

A Figura 2.16 apresenta os resultados de desempenho do usuário fonte, obti-

Tabela 2.4: *Parâmetros utilizados nas simulações.*

Parâmetro	**Valor**
Modulação	B-PSK
Código	RS(15,5)
Canal	Rayleigh
$f_{Doppler}$	320 Hz
$(E_b/N_0)_r$	$(E_b/N_0)_f + 20dB$
$(E_b/N_0)_{fr}$	20 dB

dos no processo de codificação cooperativa CC-RS. Nestas simulações o método utilizado para reestruturação do quadro foi o sinalizado.

O maior ganho, em torno de 6 dB@$(E_b/N_0)_f = 10$dB, foi obtido com a máxima taxa de cooperação ($T_c = 0,667$). Códigos mais robustos podem ser empregados para aumentar o ganho de desempenho neste cenário, visto que haverá mais símbolos para serem compartilhados. A relação sinal-ruído e as condições do canal de comunicação entre os usuários resulta em uma alta probabilidade do usuário *relay* colaborar com o usuário fonte.

As condições de transmissão entre o usuário *relay* e a ERB ($(E_b/N_0)_r$), bem como, as condições de transmissão entre os usuários ($(E_b/N_0)_{fr}$) contribuíram para os resultados obtidos. Valores mais elevados de ganho podem ser alcançados em cenários onde as condições de transmissão do usuário *relay* são melhores.

A utilização do método não sinalizado na reestruturação do quadro também

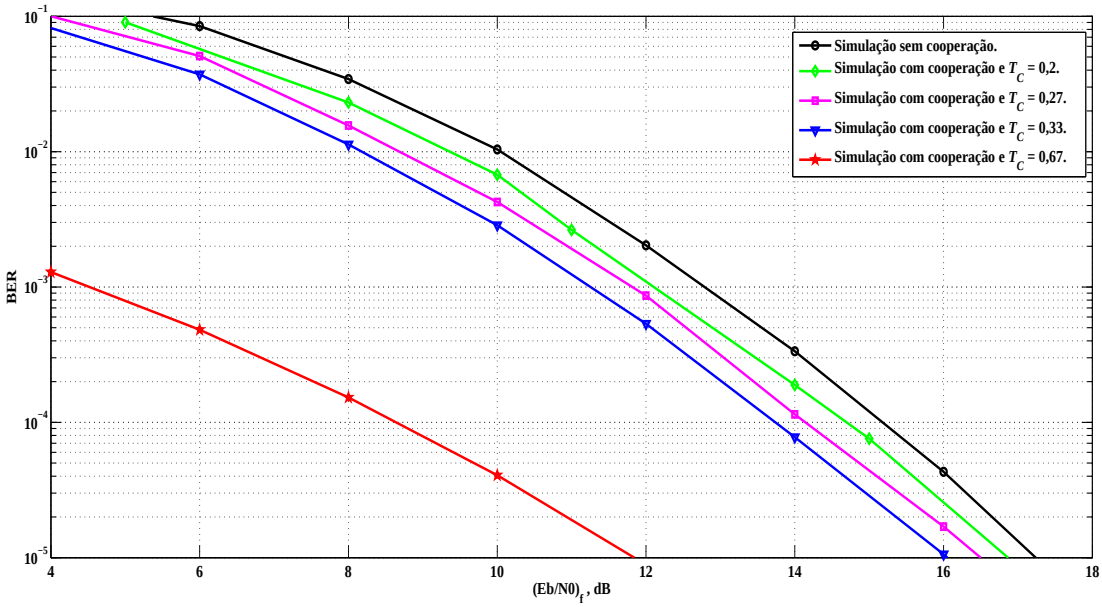


Figura 2.16: *Desempenho do usuário fonte em função da taxa de cooperação.*

foi analisada. O objetivo da análise com este método é fazer uma comparação com o método sinalizado na situação em que o usuário *relay* não coopera com o usuário fonte. A Figura 2.17 apresenta os resultados de desempenho do usuário fonte, utilizando o método sinalizado e não sinalizado.

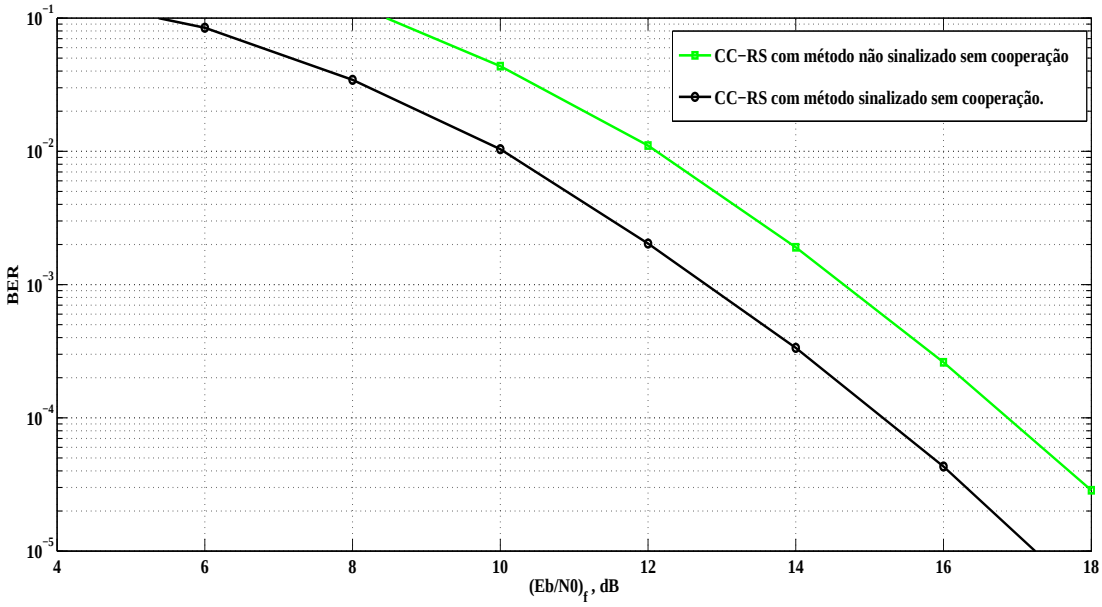


Figura 2.17: *Desempenho do usuário fonte em função da taxa de cooperação.*

O resultado obtido com o método não sinalizado mostra uma queda de desem-

penho de aproximadamente 2 dB do usuário fonte quando não há cooperação.

A técnica de codificação cooperativa CC-RS tem como principal objetivo melhorar o desempenho do usuário que esteja em condições ruins para transmissão. Desta maneira, quando não está ocorrendo a cooperação, o desempenho do usuário fonte não pode ser degradado pelo sistema cooperativo, uma vez que ele já se encontra em condições ruins para transmissão. Portanto, este motivo foi decisivo pela escolha do método sinalizado nas simulações de desempenho do usuário fonte.

2.9 Conclusões

A aplicabilidade do Reed Solomon no sistema de codificação cooperativa se mostrou uma alternativa eficiente, já que não é necessário utilizar outro código para auxiliar o usuário *relay* decidir por cooperar. A adição do símbolo de sinalização na técnica CC-RS resulta em um pequeno *overhead* ao sistema. Porém, os benefícios obtidos com esta solução, como a decodificação inequívoca dos dados do usuário fonte tornam esse *overhead* aceitável.

O uso do método não sinalizado quando não há cooperação pode ser comprovado que contribui para queda de desempenho do usuário fonte, o que pode ser constatado pela comparação entre o desempenho do usuário fonte utilizando o método sinalizado.

A eficácia do uso do código Reed Solomon no sistema de codificação cooperativa pode ser comprovada pelo ganho de desempenho obtidos nas simulações computacionais.

Os resultados mostraram que é possível aumentar o desempenho do usuário utilizando o recurso da cooperação, e sua maximização neste cenário depende da taxa de cooperação e das condições de transmissão em que se encontra o usuário *relay*.

Capítulo 3

Codificação cooperativa em canais com desvanecimento seletivo em frequência.

Uma das alternativas para viabilizar que os sistemas móveis operem com taxa de dados maior consiste em aumentar a largura de faixa dos canais de comunicação. Entretanto, esta opção pode resultar em queda de desempenho nos sistemas de recepção, uma vez que podem ocorrer severas degradações no sinal em função do desvanecimento seletivo em frequência.

O uso de técnicas que permita aumentar o desempenho dos sistemas operantes em canais com desvanecimento seletivo em frequência se tornam essenciais. A aplicação da codificação cooperativa em canais móveis vem se mostrando eficiente, como apresentado em [16]-[18] e nos resultados obtidos nas simulações do Capítulo 2.

O ganho de desempenho com o processo de codificação cooperativa é mais acentuado quando o canal de comunicação do usuário *relay* está em melhores condições de transmissão que o canal de comunicação do usuário fonte. Assim, uma análise prévia do canal de comunicação entre o usuário *relay* e a ERB se faz necessária para sinalizar em quais instantes este usuário pode cooperar, a fim de

obter a melhor eficiência do sistema cooperativo e também minimizar os efeitos que a cooperação pode acarretar no desempenho do usuário *relay*.

Os recursos da técnica OFDM, como o uso das portadoras pilotos para estimação de canal, permitem aumentar a robustez do sistema frente os efeitos causados pelos desvanecimentos seletivos em frequência e provêm informações para analisar as condições do canal de comunicação [28]. Este capítulo irá abordar o uso da codificação cooperativa aliada à tecnologia OFDM, onde o recurso da estimação de canal será utilizado como ferramenta para mensurar as condições do canal de comunicação entre o *relay* e a ERB.

3.1 Análise do canal de comunicação

O objetivo desta análise é garantir que a cooperação ocorra somente quando o canal de comunicação entre o usuário *relay* e a ERB esteja em condições propícias a isto. A análise e a decisão de quando o usuário *relay* poderá cooperar com o usuário fonte é realizada pela ERB e transmitida para o usuário *relay* no canal de descida. Neste caso, é necessário além do sucesso na decodificação dos dados do usuário fonte, uma autorização enviada pela ERB para que o usuário *relay* faça a opção pela cooperação.

O modo como a ERB estima a resposta do canal e autoriza a cooperação será apresentado a seguir.

3.1.1 Limiar de recepção

O limiar de recepção é o valor de E_b/N_0 mínimo necessário para que o sistema opere com uma BER tolerável a cada tipo de serviço. Este limiar varia de acordo com diversos fatores, como o canal de comunicação, o serviço que trafega pelo sistema, a implementação do receptor, tipo de modulação, código corretor de erro, profundidade do entrelaçador, etc..

A análise que a ERB faz para caracterizar se o canal é propício ou não à coo-

peração depende da quantidade de subportadoras que possuem um valor de E_b/N_0 maior ou igual ao limiar de recepção. O valor de E_b/N_0 de cada subportadora na recepção pode ser calculado por

$$(E_b/N_0)_{rx_n} =\mid H_n(f) \mid^2 \times (E_b/N_0)_l, \tag{3.1}$$

onde $(E_b/N_0)_l$ é a relação sinal-ruído em um canal AWGN (AWGN - *Additive White Gaussian Noise*), desconsiderando a ação seletiva do canal de comunicação. A Figura 3.1 mostra um símbolo OFDM recebido em função de um limiar estabelecido.

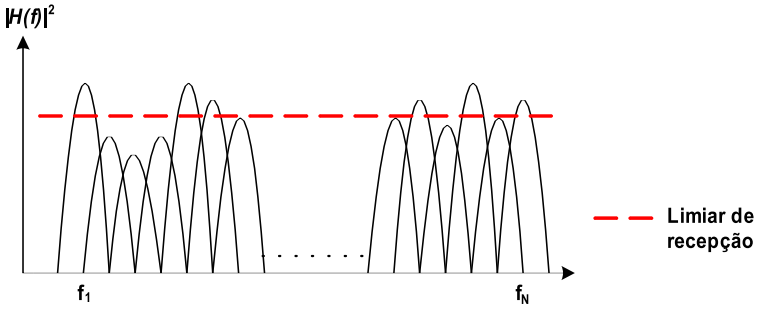


Figura 3.1: *Limiar utilizado para análise do canal de transmissão.*

O parâmetro usado para definir se o usuário *relay* poderá colaborar com o usuário fonte é dado por

$$\xi = \frac{N_s}{N_t} \times 100, \tag{3.2}$$

onde N_s é o número de subportadoras que possuem (E_b/N_0) maior que o limiar de recepção e N_t é o número total de subportadoras do símbolo OFDM. Caso $\xi \geq \xi_{min}$, o usuário *relay* é autorizado a cooperar, desde que consiga decodificar com sucesso os dados do usuário fonte. Se o $\xi < \xi_{min}$, a ERB não autoriza a cooperação. A Figura 3.2 mostra o fluxograma utilizado pela ERB para determinar

se o usuário *relay* está apto a cooperar ou não com o usuário fonte.

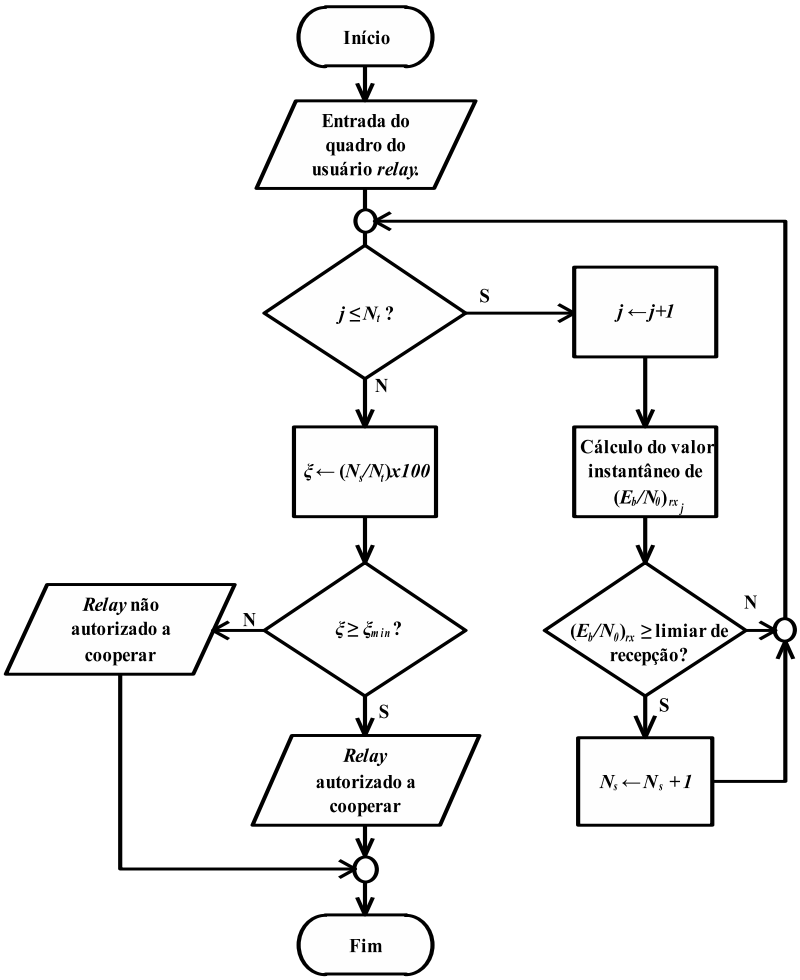


Figura 3.2: *Fluxograma utilizado pela ERB para autorizar o usuário relay a cooperar.*

3.2 Descrição do cenário para análise do esquema de codificação cooperativa CC-RS

O esquema de integração do OFDM com a codificação cooperativa tem por objetivo viabilizar o ganho de codificação em canais seletivos em frequência. Neste ambiente, a ERB faz a análise das condições dos canais de comunicação dos usuários, definindo aqueles que estão aptos a operar como *relay*. A Figura 3.3

ilustra o cenário proposto para análise do desempenho do sistema.

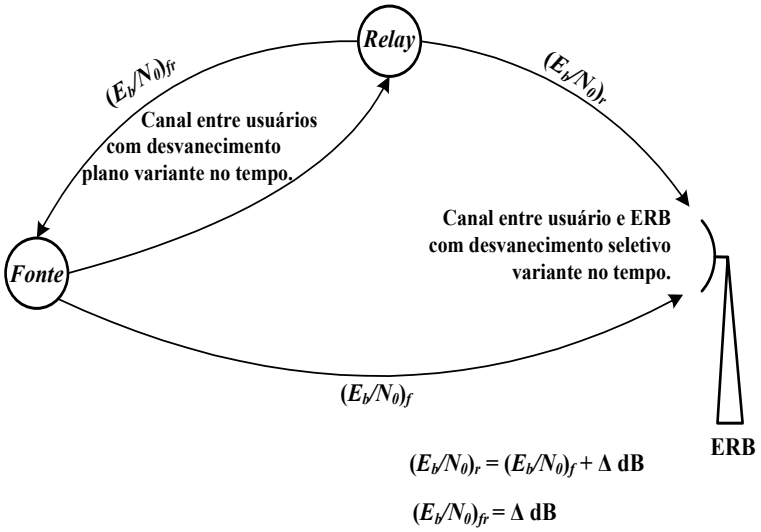


Figura 3.3: *Cenário proposto para as simulações com canal seletivo móvel.*

Nesta figura é visto que a comunicação entre usuário fonte e *relay* é bidirecional. Isto significa que em determinados instantes o usuário que está operando como fonte pode ser o *relay* e vice-versa. O que determina quais as funções que os usuários assumem no sistema é a condição do canal de comunicação destes para a ERB. Para fins de análise de desempenho, no cenário proposto as funções dos usuários não se alternam.

A técnica de multiplexação OFDM foi empregada na interface aérea do sistema. Assume-se que o intervalo de guarda seja suficiente para que a interferência intersimbólica causada por múltiplos percursos seja desprezível. Uma subportadora piloto é inserida a cada 7 subportadoras de dados, sendo que a primeira subportadora é sempre uma subportadora piloto. O objetivo desta configuração é minimizar a complexidade de implementação da geração e recepção do símbolo OFDM. O número total de subportadoras utilizadas por símbolo OFDM (N_t) é de 1024. Este valor é usado como opção de configuração de transmissão de algumas tecnologias de comunicação, como WiMAX [37] e LTE [38].

O canal entre os usuários é plano e variante no tempo, com desvio Doppler definido pela Eq.(2.4).

A modulação empregada nas portadoras piloto foi a BPSK (*Binary Phase Shift Keying*)[39], enquanto que as portadoras de dados trafegam símbolos provenientes de uma modulação 16-QAM (*Quadrature Amplitude Modulation*) [39].

A técnica empregada pela ERB no processo de interpolação das portadoras pilotos é à IDFT/DFT (*Inverse Discrete Fourier Transform / Discrete Fourier Transform*) [40], a qual apresenta desempenho satisfatório em canais lineares [31]. A informação obtida na estimação é utilizada no processo de equalização do sinal para corrigir as degradações sofridas ao longo do canal de transmissão.

A Figura 3.4 ilustra o diagrama em blocos básico do terminal dos usuários.

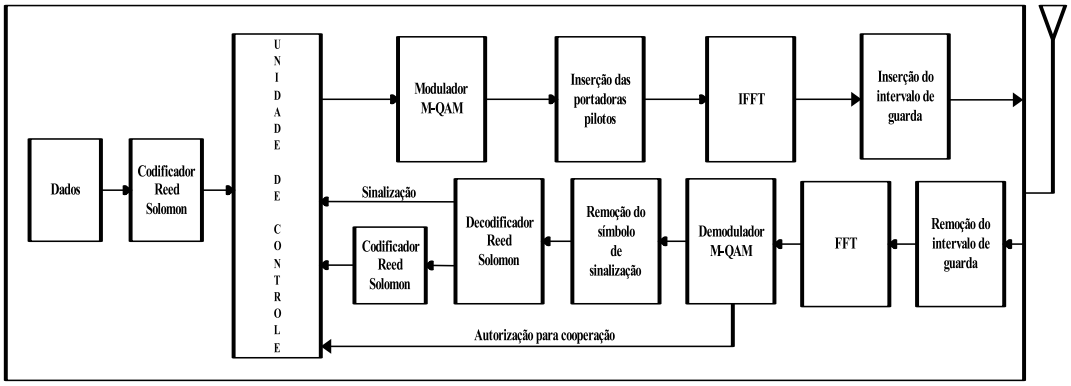


Figura 3.4: *Diagrama em blocos básico do transmissor do usuário.*

A probabilidade de erro de *bit* de um sistema 16-QAM OFDM em canal móvel e plano é dado por [32]

$$P_{b_{cp}} \approx \frac{3}{2}\left(1 - \sqrt{\frac{0,4E_b/N_0}{1+0,4E_b/N_0}}\right), \tag{3.3}$$

onde E_b é a energia média de *bit* da constelação 16-QAM, N_0 é a densidade espectral de potência do ruído AWGN. Admite-se que as variáveis aleatórias Gaussianas ortogonais empregadas para gerar o modelo de canal Rayleigh possuem variância igual à 1/2 [41].

Para se obter a probabilidade de erro de *bit* de um sistema 16-QAM OFDM em canal móvel e seletivo em frequência é necessário ponderar a relação sinal ruído pela resposta em frequência do canal. Neste caso, considerando-se as mesmas condições empregadas para obter (3.3), tem-se [32]

$$P_{b_{cs}} \approx \frac{3}{2N_t} \sum_{n=1}^{N_t} \left(1 - \sqrt{\frac{0,4|H_n|^2 E_b/N_0}{1+0,4|H_n|^2 E_b/N_0}} \right), \tag{3.4}$$

onde H_n é a resposta em frequência da n-ésima subportadora. Considera-se que o número de subportadoras seja elevado o suficiente para que a resposta em frequência para uma dada subportadora seja considerada plana [28]. A Eq. (3.4) pode ser empregada para definir se o canal de um dado usuário permite que o mesmo coopere com outro usuário em função da resposta em frequência do canal, uma vez que é possível determinar se a BER experimentada por um dado usuário é suficiente ou não para atender a QoS deste usuário.

A eficácia do esquema de codificação cooperativa é medida através do ganho obtido pelo usuário fonte. O valor do limiar de recepção $(E_b/N_0)_l$ foi definido como 10,27 dB. É possível, com este valor, que um sistema operando em canal plano e invariante no tempo apresente uma BER de 10^{-3} e 10^{-4} utilizando os códigos RS(15,5) e RS(255,85) respectivamente.

O usuário mais próximo da ERB possui uma (E_b/N_0) média Δ dB maior do que o usuário mais distante da ERB. No sistema implementado, a informação que a ERB envia aos usuários sobre as condições do canal é referente a análise dos dados de um quadro que já foi transmitido, ou seja, a estruturação de um quadro futuro depende da análise de um quadro que já foi transmitido. Dessa maneira, para que a análise do canal seja eficiente para o sistema, o tempo de coerência do canal deve ser maior que pelo menos o tempo de dois símbolos OFDM. As respostas impulsivas discretas atribuídas aos canais dos dois usuários para a ERB

são, respectivamente,

$$\begin{aligned} h_r[n] &= r \cdot (0,999\delta[n] + 0,043\delta[n-2] + 0,01\delta[n-4]) \\ h_f[n] &= r \cdot (0,85\delta[n] + 0,35\delta[n-7] + 0,39\delta[n-14]) \,, \end{aligned} \tag{3.5}$$

onde r é uma variável aleatória com distribuição Rayleigh [6] e $\delta[n]$ é um impulso unitário (delta de Kronecker). O objetivo da escolha destes canais é permitir que nulos espectrais ocorram em posições diferentes para cada percurso, de forma a explorar o recurso da diversidade. A Figura 3.5 ilustra as respostas em frequências normalizadas dos usuários *relay* e fonte respectivamente.

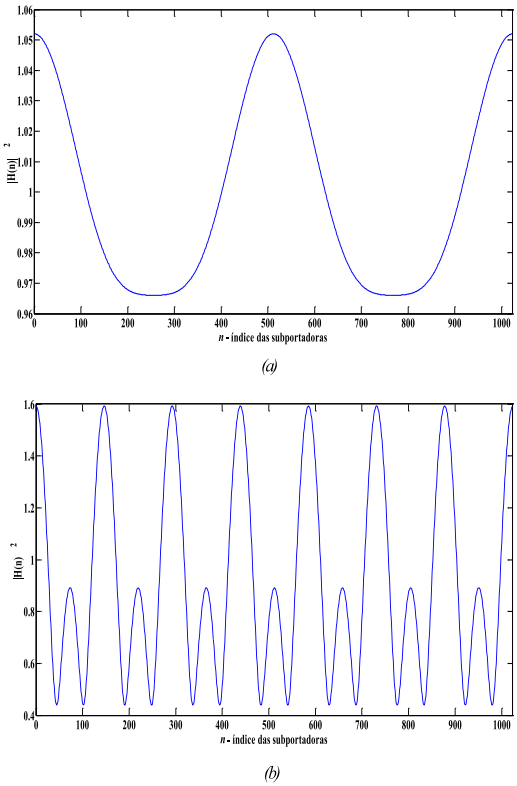


Figura 3.5: *(a) Resposta em frequência do canal do usuário relay para a ERB. (b) Resposta em frequência do canal do usuário fonte para a ERB.*

A função desempenhada por cada um dos usuários é decidida pela análise da resposta em frequência de ambos os canais e está baseada no princípio de que

a codificação cooperativa só é eficaz quando o usuário *relay* possui um canal de comunicação com melhores condições de transmissão do que o canal do usuário fonte. Assim, os canais descritos na Eq. (3.5) representam uma situação em que o usuário *relay* está mais próximo da ERB que o usuário fonte.

O desempenho do usuário fonte será analisado em função de uma única taxa de cooperação. O valor utilizado é de 0,667, ou seja, o usuário *relay* cede todos os seus símbolos de paridade para a transmissão dos símbolos de paridade do usuário fonte. O número de símbolos que serão compartilhados depende do código empregado. Neste caso, foram os códigos RS(15,5) e RS(255,85). A escolha destes dois códigos se baseou na característica de que ambos possuem o mesmo valor de $(n-k)/n$, o que significa que a taxa de codificação destes códigos é a mesma. Outra razão para a escolha consiste em possibilitar uma análise do desempenho do sistema com diferentes comprimentos de códigos. As capacidades de correção e taxas de cooperação são apresentadas nas Tabelas 2.2 e 3.1 respectivamente.

Tabela 3.1: *Capacidade de correção do RS(255,85) em função da taxa de cooperação.*

Código principal	Código Puncionado	Capacidade de correção de erros	Taxa de cooperação
RS(255,85)	sem puncionamento	85 símbolos	0
	RS(204,85)	34 símbolos	0,2
	RS(187,85)	17 símbolos	0,27
	RS(170,85)	0 símbolos	0,333
	RS(153,85)	0 símbolos	0,4
	RS(136,85)	0 símbolos	0,467
	RS(119,85)	0 símbolos	0,533
	RS(102,85)	0 símbolos	0,6
	RS(85,85)	0 símbolos	0,667

Os valores das taxas de cooperação da Tabela 3.1 não correspondem a todos os valores possíveis com o código RS(255,85). Estes valores foram escolhidos para efeito de análises comparativas com os valores da Tabela 2.2, que utiliza o código RS(15,5).

3.3 Estruturação do quadro na transmissão

A estruturação do quadro do usuário *relay* na transmissão é análoga à apresentada no Capítulo 2. Porém, a informação de sucesso na decodificação da informação do usuário fonte não é a única prerrogativa para que o *relay* faça a opção pela cooperação, pois também é necessário a autorização por parte da ERB para que isto aconteça. A cooperação, portanto, só ocorre mediante o atendimento destes dois requisitos. Caso um destes requisitos não seja atendido, o *relay* não coopera com o usuário fonte. A Figura 3.6 mostra o fluxograma do processo de estruturação do quadro realizado pelo *relay*. O método de estruturação utilizado é o sinalizado, descrito na Seção 2.7.

3.4 Reestruturação do quadro na ERB

A Figura 3.7 ilustra os blocos básicos que compõe o sistema de recepção da ERB. A ERB, ao receber a informação proveniente dos dois caminhos, deve reestruturar os quadros de ambos os usuários. Esta reestruturação é realizada pela bloco "Unidade de Controle I"e repassada para os decodificadores. Os decodificadores, por sua vez, disponibilizam os dados decodificados e uma sinalização que informa a quantidade de erros que estavam contidos na palavra código, ou se a mesma não foi decodificada com sucesso. Por conseguinte, o bloco "Unidade de Controle II"realiza a análise dos dados em função da sinalização enviada pelos decodificadores. Os dados utilizados pela ERB são os que tiveram sucesso na decodificação.

As informações que a ERB detém para reestruturação dos quadros são o símbolo de sinalização e a taxa de cooperação utilizada no sistema. Com estas duas informações a "UC"disponibiliza, no caso de haver cooperação, dois quadros ao usuário fonte, um contendo as informações provenientes do caminho principal e outro contendo parcela da informação do caminho principal mais

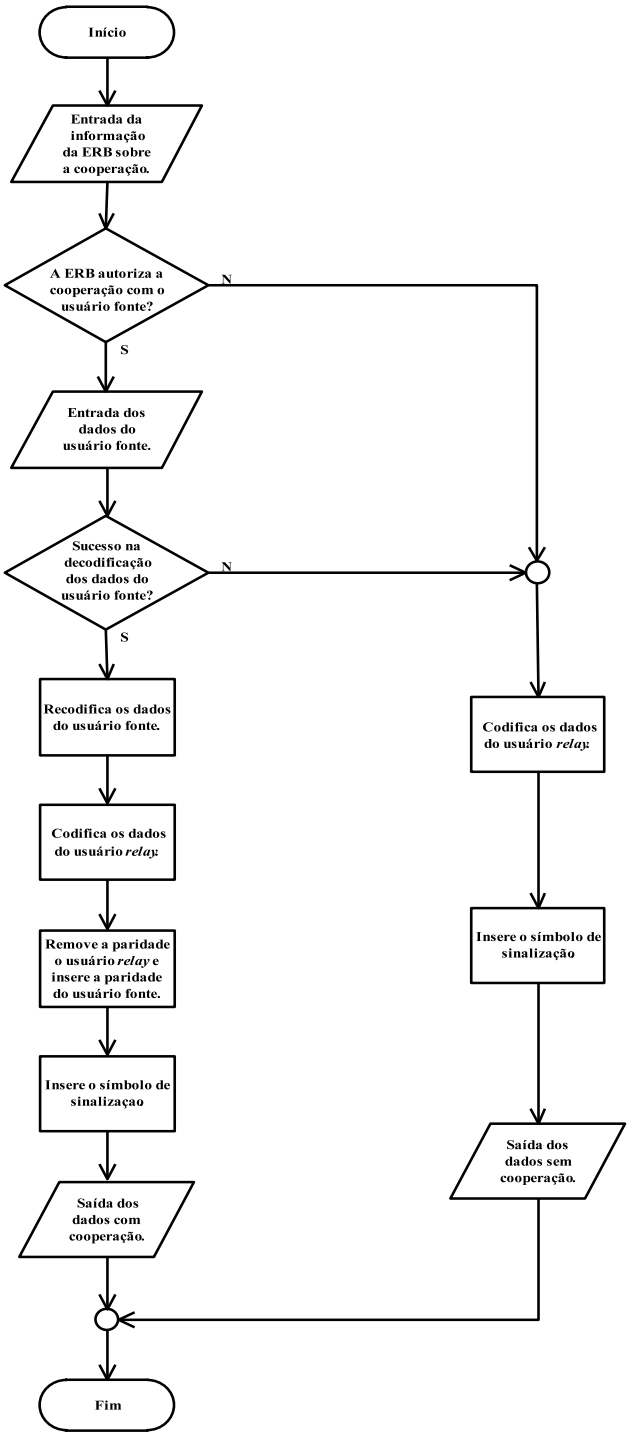


Figura 3.6: *Fluxograma de estruturação do quadro do usuário relay.*

parcela da informação do caminho *relay*. No caso de não haver cooperação, a "UC" disponibiliza somente um quadro contendo informações provenientes do ca-

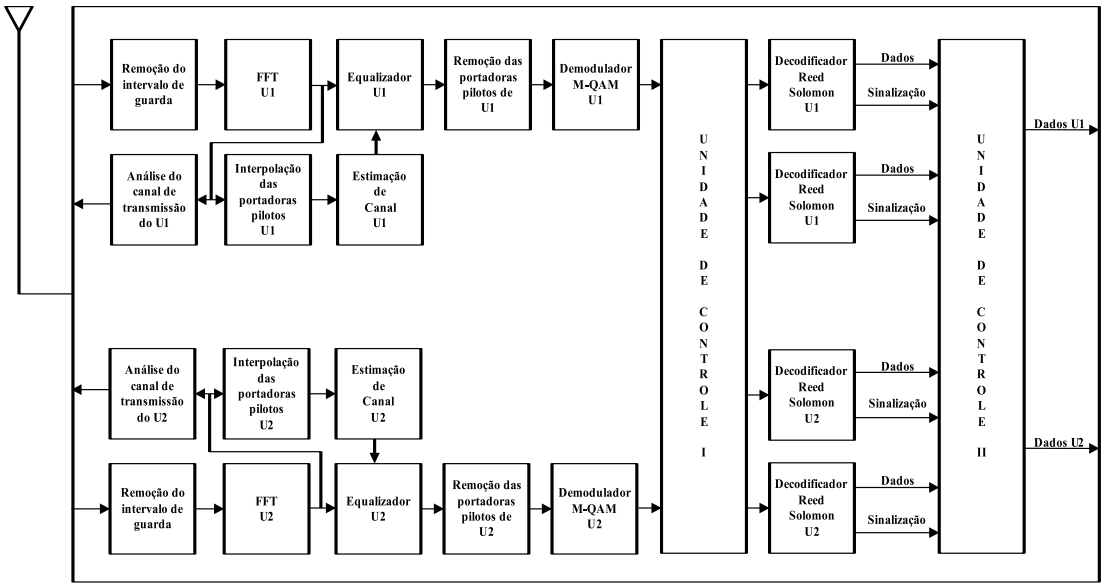


Figura 3.7: *Diagrama em blocos da ERB.*

minho principal. A Figura 3.8 ilustra como fica a reestruturação do quadro do usuário fonte após a análise do símbolo de sinalização.

3.5 Análise dos resultados das simulações.

O objetivo desta seção é apresentar as análises de desempenho da técnica de codificação cooperativa CC-RS em um canal de comunicação seletivo em frequência e variante no tempo. Estas análises exploram o ganho de desempenho do usuário para a máxima taxa de cooperação possível do sistema ($T_c = 0,667$). Também é realizada uma comparação do desempenho do sistema utilizando dois códigos RS diferentes, RS(15,5) e RS(255,85). Os parâmetros das simulações podem ser vistas na Tabela 3.2. O desempenho do usuário fonte é mensurado em função da porcentagem das subportadoras que estão acima do limiar de recepção. As Figuras 3.9 e 3.10 mostram os resultados obtidos no sistema operando com os códigos RS(15,5) e RS(255,85), respectivamente. A diferença entre a relação sinal-ruido dos canais de comunicação dos usuários é de 26 dB ($\Delta = 26$ dB). Todas as

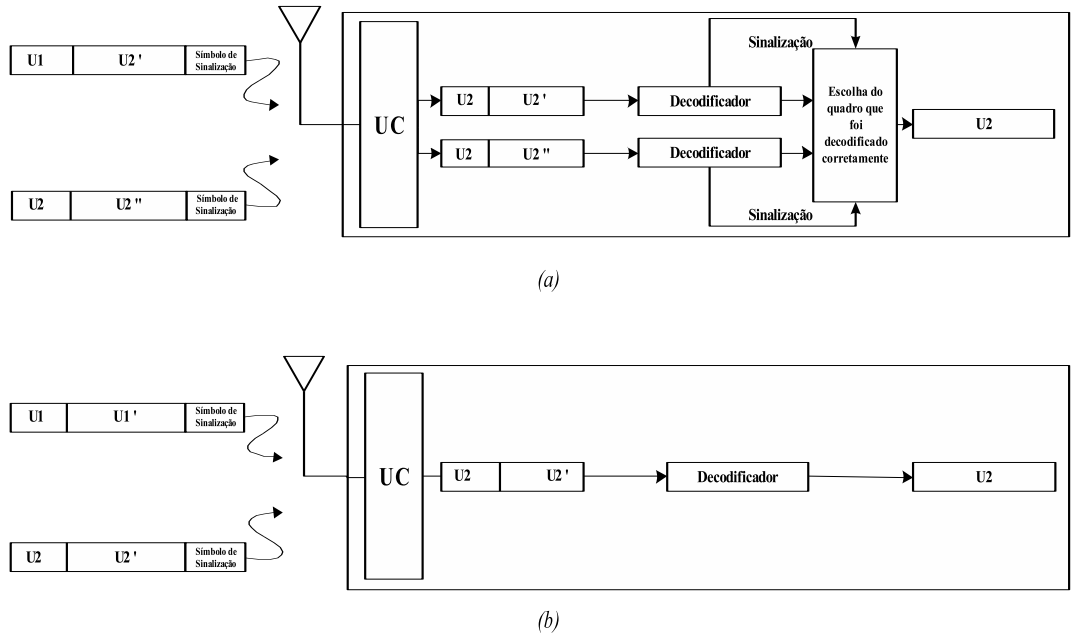


Figura 3.8: *(a) Etapa de recepção do usuário fonte com o processo de cooperação. (b) Etapa de recepção do usuário fonte sem o processo de cooperação.*

Tabela 3.2: *Parâmetros utilizados nas simulações.*

Parâmetro	**Valor**
Técnica de interpolação	IDFT/DFT
Modulação das subportadoras de dados	16-QAM
Modulação das subportadoras pilotos	B-PSK
Canal entre usuário e ERB	Seletivo e variante no tempo
Canal entre usuários	Plano e variante no tempo
Taxa de cooperação (T_c)	0,667
Código	RS(15,5)
	RS(255,85)
Limiar de recepção $(E_b/N_0)_l$	10,27 dB
$(E_b/No)_r$	$(E_b/N_0)_f + 16$ dB
	$(E_b/N_0)_f + 26$ dB
$(E_b/N_0)_{fr}$	16 dB
	26 dB

simulações com denotação de COFDM utilizaram códigos corretores de erro e estimação de canal.

Nos resultados obtidos é visto que o processo de estimação resultou em uma

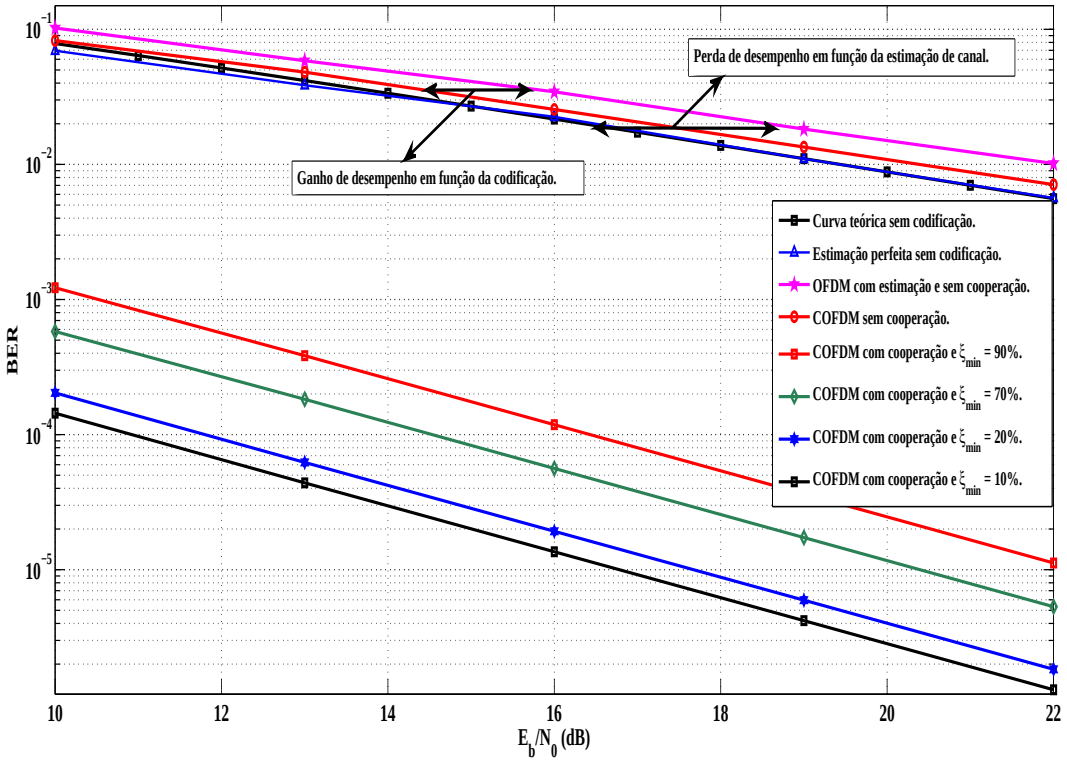


Figura 3.9: *Desempenho do usuário fonte utilizando o código RS(15,5).*

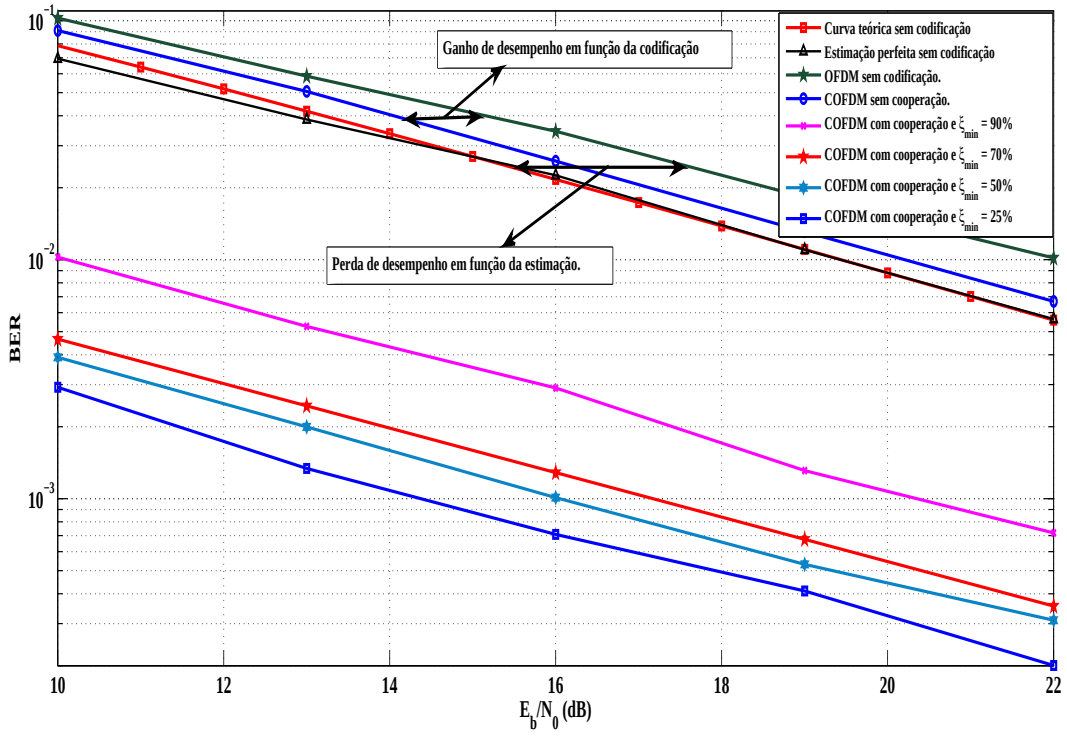


Figura 3.10: *Curva de desempenho do usuário fonte utilizando o código RS(255,85).*

queda de desempenho do sistema. O maior ganho explorado nas simulações foi para os valores de ξ_{min} igual à 10% com o código RS(15,5) e de 25% com o código RS(255,85), os quais não correspondem ao valor máximo que se pode atingir com este sistema. O decréscimo de ξ_{min} resulta em aumento do ganho de desempenho do sistema, sendo que este ganho não aumenta linearmente com o aumento de ξ_{min}.

A análise do ganho em função do valor de ξ_{min} reflete nas condições de transmissão do canal do *relay*, pois com um valor elevado de ξ_{min} já é possível melhorar o desempenho do usuário fonte em aproximadamente 10 dB. Este ganho é obtido quando pelo menos 90% das suas subportadoras estão acima do limiar de recepção.

As condições de transmissão do usuário *relay* estão diretamente relacionadas ao valor de ξ_{min} que resulta no aumento do desempenho do usuário fonte levando-o a patamares de QoS aceitáveis. Em outras palavras, é esperado que o ganho de desempenho do sistema operando em condições inferiores seja menor. As Figuras 3.11 e 3.12 mostram os resultados obtidos nas simulações com os mesmos códigos, porém a diferença entre a relação sinal-ruido dos canais de comunicação dos usuários é de 16 dB ($\Delta = 16$ dB), ou seja, uma diferença de 10 dB em relação ao valor utilizado nas simulações que geraram os resultados das Figuras 3.9 e 3.10.

Observa-se nos resultados das Figuras 3.11 e 3.12 que realmente há uma perda de desempenho quando comparados com os resultados das Figuras 3.9 e 3.10. O fato do *relay* piorar em 10 dB o seu canal implicou em uma redução de aproximadamente 10 dB no desempenho do usuário fonte para o mesmo valor de ξ_{min}.

O ganho de desempenho do sistema COFDM com estimação sem o recurso da cooperação apresentou valor inferior a 2 dB. Este resultado se deve às condições do canal e também o fato de que não foi utilizada nenhuma técnica de entrelaçamento neste sistema. O uso do entrelaçador aumentaria a robustez do sistema perante

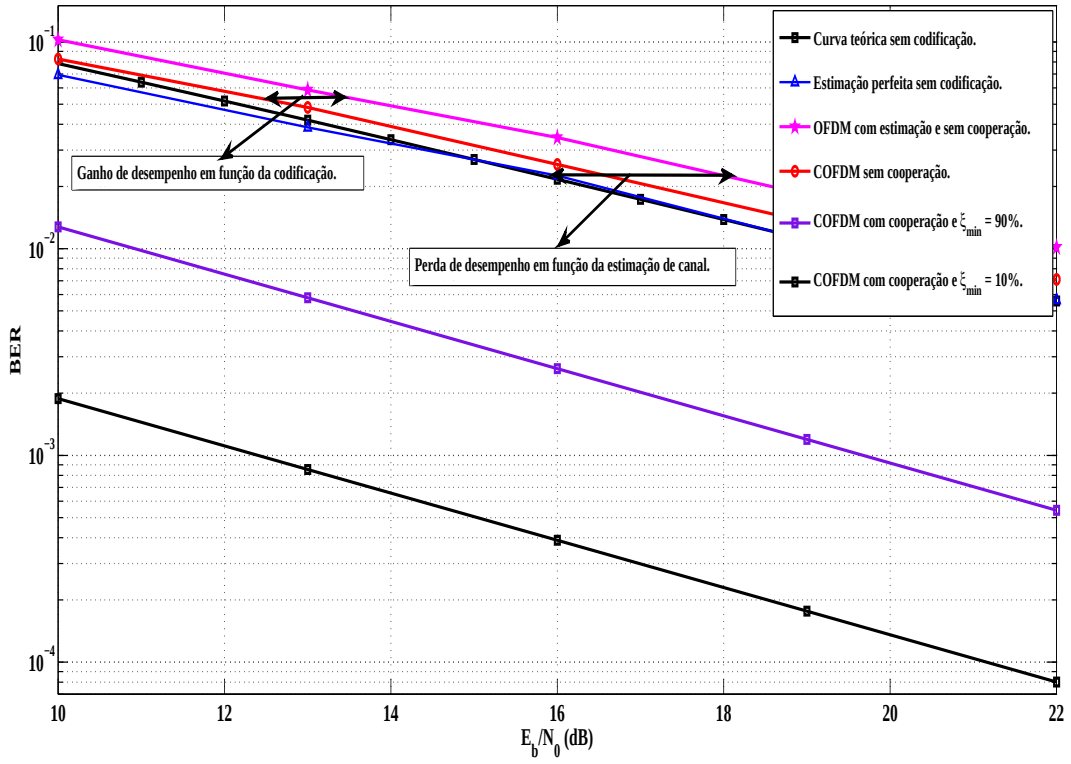


Figura 3.11: *Desempenho do usuário fonte utilizando o código RS(15,5).*

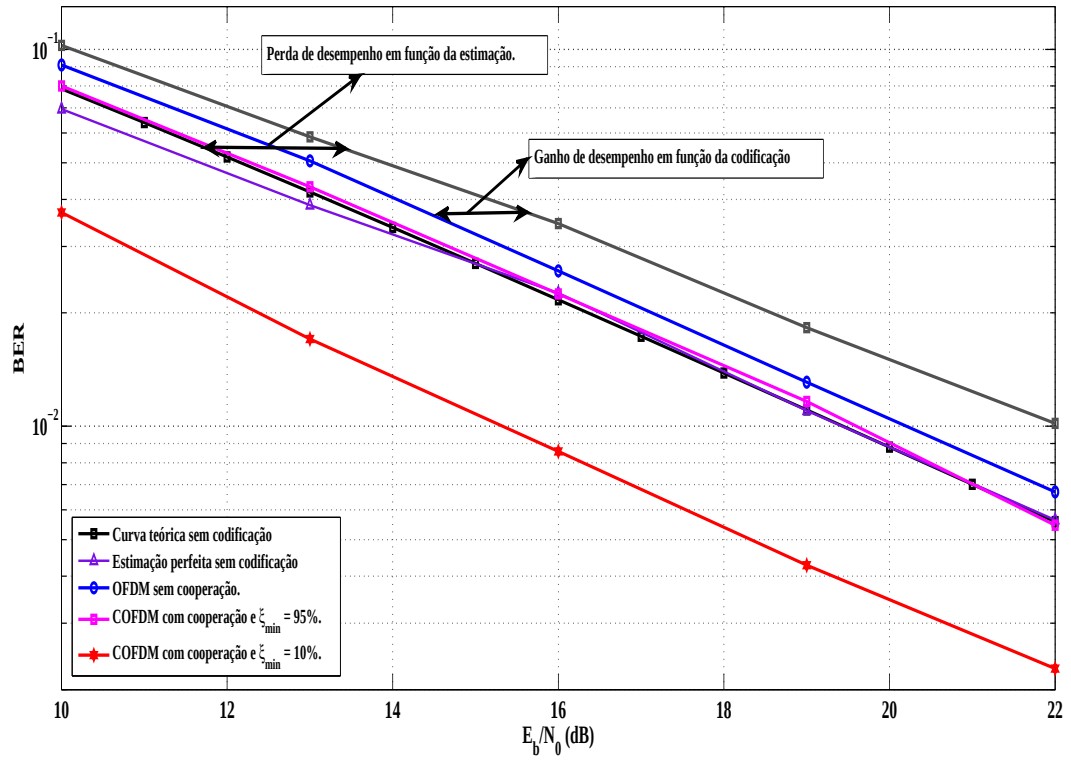


Figura 3.12: *Desempenho do usuário fonte utilizando o código RS(255,85).*

erros em rajadas introduzidos pelo canal que excedem a capacidade de correção do código.

Outra análise também a ser considerada é a comparação do desempenho entre os sistemas operando com código RS(15,5) e o RS(255,85). Nos valores obtidos nas simulações é observado que o desempenho do sistema operando com um código mais curto, RS(15,5), é superior ao desempenho do sistema operando com código mais longo, RS(255,85). Isto por conjuctura é devido a variações do canal de comunicação ao longo da palavra código transmitida.

O usuário *relay*, ao ceder parte dos seus recursos para auxiliar outro usuário durante a cooperação, pode perder desempenho. O fato de compartilhar os símbolos de paridade reduz a capacidade de correção do decodificador, o que pode acarretar a não correção de uma informação que tenha sido corrompida durante a transmissão. Assim, o desempenho deste usuário está relacionado com a taxa de cooperação do sistema, quanto maior a taxa de cooperação, maior a probabilidade de ocorrência de erros na decodificação dos seus dados.

Nos resultados mostrados nas Figuras 3.9 - 3.12, a condição que apresenta a maior probabilidade de cooperação entre os usuários e consequentemente a maior queda de desempenho do usuário *relay*, é quando o sistema opera com uma diferença entre as relações sinal-ruído igual à 26 dB ($\Delta = 26$ dB) e $\xi_{min} = 10\%$. Neste caso, a análise do desempenho deste usuário é obtido comparando um cenário onde não há cooperação com outro cenário onde ocorre a cooperação. As Figuras 3.13 e 3.14 mostram os resultados das simulações de desempenho do usuário *relay*.

O fato do usuário *relay* cooperar com o usuário fonte implica na redução do seu desempenho, como pode ser comprovado nos resultados obtidos nas Figuras 3.13 e 3.14. Esta redução é mais acentuada no sistema operando com o código RS(15,5), uma vez que, a incidência de cooperação com os usuários utilizando este código é maior.

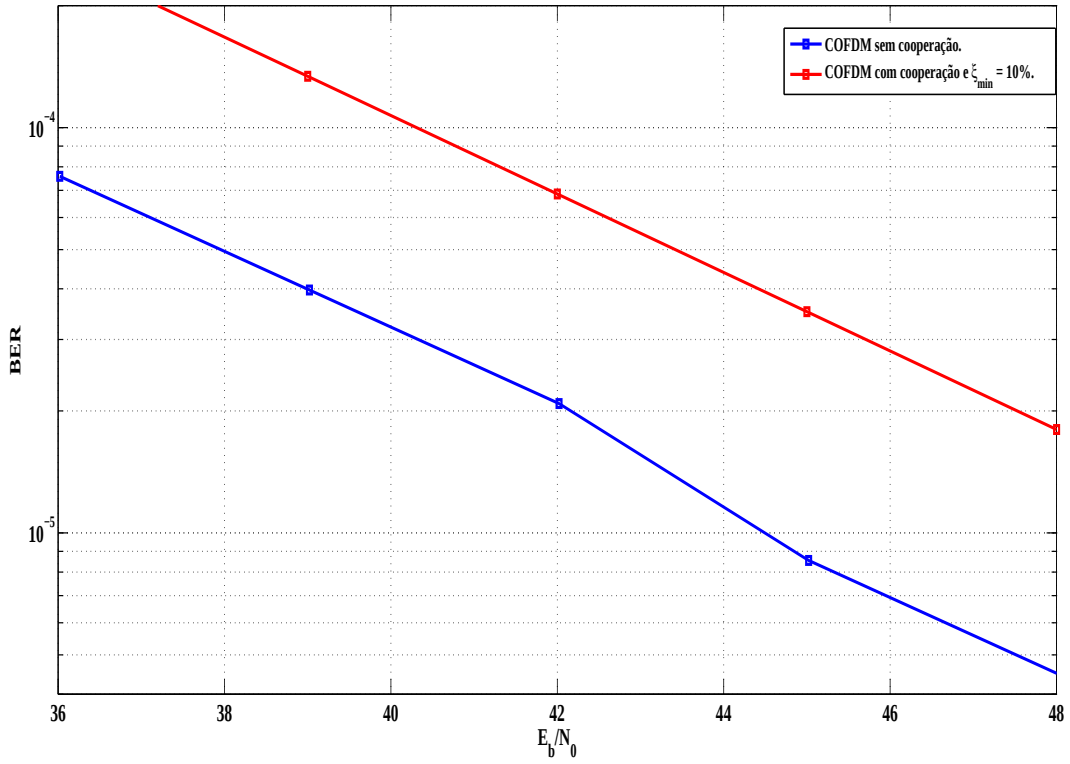


Figura 3.13: *Desempenho do usuário relay utilizando o código RS(15,5).*

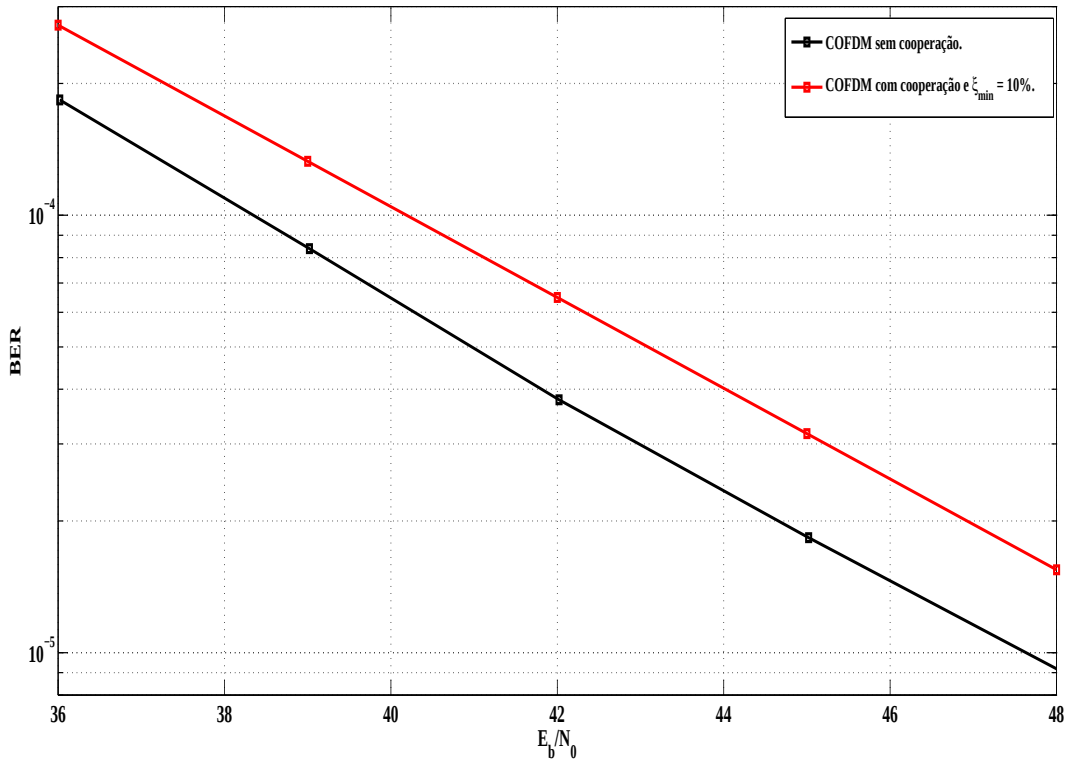


Figura 3.14: *Desempenho do usuário relay utilizando o código RS(255,85).*

3.6 Conclusões

A técnica de codificação cooperativa operando em canais com desvanecimento seletivo em frequência melhorou o desempenho do usuário fonte. A condição de transmissão do canal do *relay* está diretamente relacionada ao aumento do desempenho do usuário fonte.

A análise das condições do canal de comunicação do usuário *relay* se torna fundamental para aumentar a eficiência do sistema cooperativo e minimizar a queda do seu desempenho, uma vez que ele só é autorizado a cooperar quando as condições do seu canal de comunicação são propícias a isto. O fato do usuário *relay* optar pela cooperação traz como consequência a queda de seu desempenho, já que no processo de compartilhar os símbolos de paridade com o usuário fonte, o decodificador pode não ter capacidade suficiente para realizar as devidas correções nos dados que foram corrompidos durante a transmissão.

Os resultados das simulação mostram que o desempenho do usuário fonte é mais eficiente quando ele utiliza o código RS(15,5) no cenário proposto. Isto por conjuctura é devido ao fato de ocorrerem variações do canal de comunicação ao longo da palavra código transmitida quando é utilizado um código mais longo. Entretanto, devido a maior incidência de cooperação com o código RS(15,5), a queda de desempenho do usuário *relay* é mais acentuada do que no sistema operando com o código RS(255,85).

A maximização do desempenho do sistema cooperativo operando em canal com desvanecimento seletivo em frequência está associada: (i) às condições do canal de comunicação. Canais com condições favoráveis a transmissão resultam numa menor probabilidade de erros durante a transmissão; (ii) à escolha do limiar de recepção. Deve-se admitir o uso de valores de QoS aceitáveis a cada tipo de serviço; (iii) a taxa de cooperação. A quantidade de símbolos compartilhados está diretamente relacionada ao aumento da probabilidade de sucesso na decodificação dos dados do usuário fonte; (iv) o valor de ξ_{min}. A determinação deste

valor está associado ao número de subportadoras que devem estar acima do limiar de recepção e; (v) ao código RS empregado. A escolha do código deve levar em consideração as características do canal, a ordem de modulação e do número de subportadoras utilizada no sistema.

Nos resultados das simulações é visto que quanto menor o valor de ξ_{min} maior é o ganho obtido pelo usuário fonte e consequentemente maior a queda do desempenho do usuário *relay*. Portanto, a escolha deste valor deve ser tomada com o objetivo de atingir o limitante que leva o desempenho do usuário fonte à apresentar uma taxa de erro arbitrariamente baixa.

Capítulo 4

Conclusões e proposta de trabalhos futuros

A utilização da técnica de codificação cooperativa para melhorar a eficiência do sistema de transmissão já foi comprovada nos estudos apresentados em [16]-[18].

O uso do código Reed Solomon como solução para o sistema de codificação cooperativa cooperativo se mostrou eficiente, dado pelos recursos oferecidos ao sistema no processo de validação dos dados do usuário fonte e na facilidade em puncionar a palavra código durante a estruturação do quadro do *relay*.

A taxa de cooperação é um dos parâmetros que permite melhorar o desempenho do usuário fonte no processo de codificação cooperativa, conforme mostrado na Figura 2.16. O ganho máximo se deu quando o sistema de codificação cooperativa operava com a máxima taxa de cooperação. A máxima taxa de cooperação no sistema CC-RS acontece quando o usuário *relay* compartilha todos os seus símbolos de paridade. Neste caso, o usuário fonte consegue utilizar os recursos de transmissão do *relay* para trafegar todos os seus símbolos de paridade por um canal que está em condições de transmissão mais favorável do que o seu.

A robustez oferecida pela tecnologia OFDM frente a canais com desvanecimentos seletivos em frequência somada aos recursos que permitem mensurar as condições de transmissão do usuário *relay* tornam o uso desta tecnologia uma

ferramenta importante para aumentar o desempenho do sistema CC-RS.

O limiar de recepção utilizado nas simulações garante uma QoS aceitável para o sistema. Este limiar possibilita conseguir uma BER da ordem de 10^{-3} e 10^{-4} em sistemas operando em canal plano e invariante no tempo, com códigos RS(15,5) e RS(255,85) respectivamente. A porcentagem mínima de subportadoras acima do limiar de recepção, ξ_{min}, estabelecida no sistema está associada ao aumento do desempenho do usuário fonte. Entretanto, se faz necessária uma análise prévia do melhor valor de ξ_{min} que a ERB deve utilizar, pois seu valor pode não oferecer tantos benefícios ao usuário fonte e ainda prejudicar o desempenho o *relay*. A escolha deste valor deve ser tomada em função das condições do canal de comunicação do usuário *relay*, uma vez que, quanto melhor as condições do canal de comunicação, maior o número de subportadoras que estarão acima do limiar de recepção.

As análises realizadas na técnica CC-RS não tinham como objetivo explorar qual a consequência da codificação cooperativa no consumo de energia do usuário *relay*, parâmetro que é de suma importância para o sistema móvel portátil, como o caso de celulares. O cenário utilizado nas análises explorou a utilização de um único *relay* no sistema móvel, embora este sistema possa operar com mais de um *relay*.

A aplicabilidade da codificação cooperativa só apresenta vantagens quando o usuário *relay* está em melhores condições de transmissão que o usuário fonte. Caso contrário, não há benefício nenhum em utilizar o recurso da cooperação e o desempenho do usuário *relay* poderá cair drasticamente, chegando a patamares inaceitáveis de QoS.

Os resultados alcançados com a utilização da técnica CC-RS mostraram que é possível melhorar o desempenho do usuário que está em condições ruins de transmissão. A maximização do desempenho desta técnica, conforme mostrado no Capítulo 3, está associada a diversos fatores, como a taxa de cooperação, o

valor de ξ_{min}, as condições do canal de transmissão do usuário *relay* e do limiar de recepção estabelecido a cada tipo de serviço.

Diante destes resultados, novas frentes de estudos podem ser exploradas para analisar o uso desta técnica em outros cenários::

- Análise da técnica CC-RS em ambientes operando com mais de um usuário *relay*.
- Análise do comportamento do sistema CC-RS em ambiente dinâmico, com o usuário atuando como fonte ou *relay* dependendo das condições do canal.
- Análise de desempenho do sistema CC-RS com a taxa de cooperação variando em função das condições do canal.
- Análise do consumo de energia que o usuário *relay* tem com o processo de cooperação, dado pelo aumento do processamento no terminal de cada usuário.
- Um estudo para determinar qual o melhor código Reed Solomon que deve ser empregado na técnica CC-RS, em função do tipo de canal, do número de subportadoras do símbolo OFDM e do tipo de modulação.

Anexo A

Reed Solomon

O código Reed Solomon (RS) pertence a uma subclasse importante dos códigos BCH não binários e suas características o incluem no grupo dos códigos chamados MDS (*Maximum Distance Separable*), atribuídas aos códigos que possuem distância mínima maior que a quantidade de símbolos de paridade. Ele é largamente empregado em diversos sistemas de armazenamento e transmissão de dados como telefonia móvel, TV digital, sistemas xDSL de comunicação por cabo, etc.. Os códigos Reed Solomon se baseiam em operações aritméticas sobre os Campos de Galouis ou Campos Finitos para codificação e decodificação.

A.1 Princípio de funcionamento da codificação e decodificação do Reed Solomon

O processo de codificação de um código RS(n,k) consiste em pegar uma sequência com k símbolos de entradas e adicionar a esta sequência $(n - k)$ símbolos de redundância, de forma que a palavra código final contenha n símbolos.

O polinômio gerador de um código RS(n,k) é determinado por [27]

$$g(X) = g_0 + g_1 X + g_2 X^2 + \cdots + g_{2t-1} X^{2t-1} + X^{2t}, \tag{A.1}$$

$$g(X) = (X - \alpha)(X - \alpha^2) \cdots (X - \alpha^{2t}), \tag{A.2}$$

onde t é a capacidade de correção de erros do código RS e α^i são as raízes de $g(X)$. A Figura A.1 ilustra o circuito de um codificador RS para o polinômio descrito na Equação A.1.

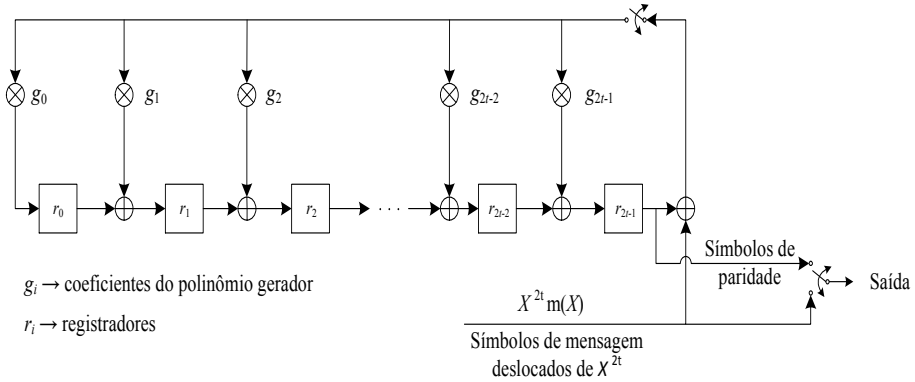


Figura A.1: *Codificador Reed Solomon RS(n,k).*

Os símbolos de mensagem e paridade são facilmente identificados após a codificação, dado pelo método de codificação na forma sistemática. A Figura A.2 ilustra a codificação RS na forma sistemática.

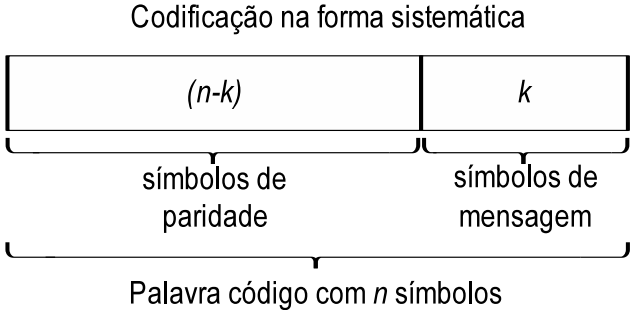


Figura A.2: *Codificação RS na forma sistemática.*

Ao ser transmitida, esta informação pode sofrer severas degradações em função

do canal de comunicação, o que acaba acarretando na inserção de erros na informação transmitida.

O decodificador ao receber essa informação com erros vai decodificar e consequentemente corrigir os erros causados no processo de transmissão, de tal forma a recuperar a informação original. A quantidade de erros passíveis de correção depende da capacidade de correção do código utilizado. Na Tabela A.1 são apresentadas as principais características do código RS [27].

Tabela A.1: *Características do código RS.*

Comprimento do código:	$n = 2^m - 1$
Número de *bits* de informação:	$k = 2^m - 1 - 2t$
Número de *bits* de paridade:	$n - k = 2t$
Distância mínima:	$d_{min} = n - k + 1$
Capacidade de correção:	$t = \left\lfloor \frac{(n-k)}{2} \right\rfloor$

onde n é o número de símbolos da palavra código, k é o número de símbolos da mensagem a ser codificada e m a quantidade de *bits* que é composto cada símbolo da mensagem.

A probabilidade de erro de símbolo deste código [27] pode ser descrita como ,

$$P_E \approx \frac{1}{2^m - 1} \sum_{j=t+1}^{2^m-1} j \binom{2^m - 1}{j} p^j (1-p)^{2^m-1-j}, \tag{A.3}$$

onde p é a probabilidade de erro do canal e t a capacidade de correção do código em questão.

Anexo B

Ferramenta computacional utilizada nas simulações

O software utilizado como ferramenta computacional neste trabalho foi o SIMULINK®, que possibilita modelagens, análises e simulações de sistemas nas mais diversas áreas da engenharia. A principal vantagem em relação a outras ferramentas é o fato de possuir uma extensa biblioteca com algumas funções pré-definidas, reduzindo tempo e complexidade na implementação das simulações. Sua interface com o usuário é simples, o que permite modelar sistemas mesmo sem um conhecimento profundo da ferramenta.

Os principais blocos como: o terminal do usuário, o canal de comunicação e a ERB, que compõem o sistema utilizado nas simulações do Capítulo 3 são ilustrados na Figura B.1. Cada bloco apresentado é composto por outros blocos com suas respectivas funções.

As Figuras B.2 e B.3 ilustram os blocos contidos no terminal do usuário *relay*. Na Figura B.3 é visto que todos os blocos com exceção do bloco "Template_U2_TO_U1"são blocos com funções específicas e portanto, disponíveis para uso na biblioteca do SIMULINK®.

O bloco designado como "Template_U2_TO_U1"é um bloco genérico, cuja funcionalidade é descrita por uma linguagem de programação implementada pelo

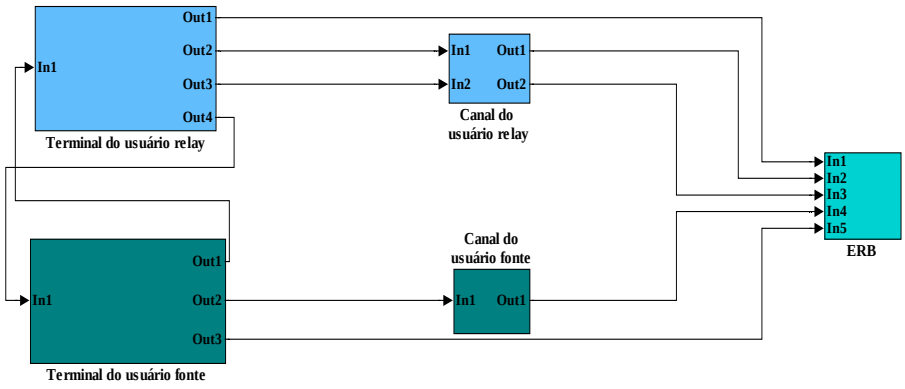


Figura B.1: *Ambiente do SIMULINK®.*

usuário. Neste caso, sua funcionalidade foi programada utilizando a linguagem de programação C. Um exemplo desta programação é ilustrado no Anexo C, que descreve a funcionalidade do bloco da Figura B.2 "Unidade de controle" do usuário *relay*.

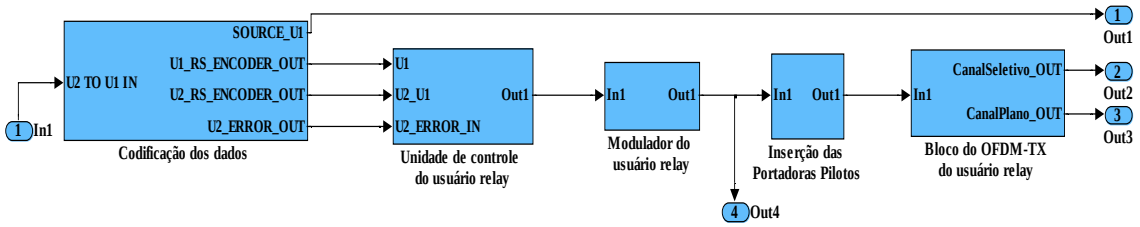


Figura B.2: *Diagrama em blocos do terminal do usuário relay.*

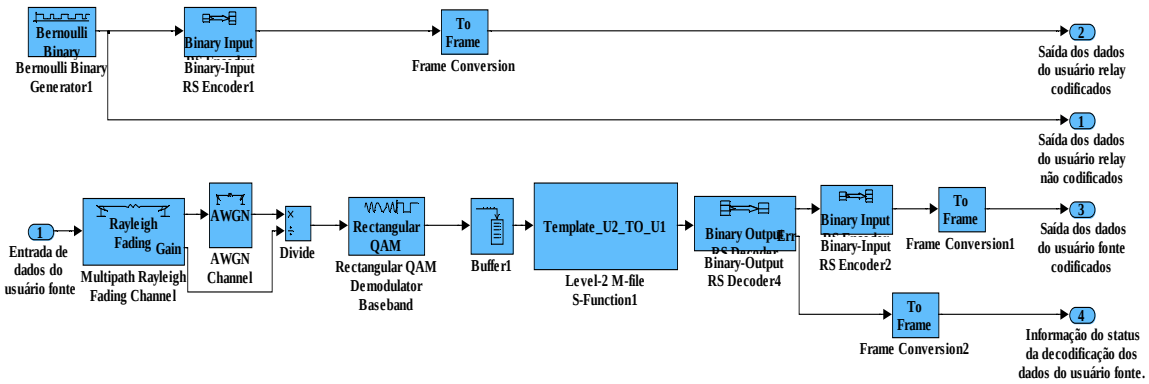


Figura B.3: *Bloco de codificação dos dados do usuário relay.*

Os blocos que compõe a ERB são ilustrados na Figura B.4. É visto nesta figura que existem duas unidades de controle, sendo a primeira responsável por

reestruturar os quadros recebidos dos usuários e a segunda por analisar os dados após a decodificação. Ambos os blocos não existem na biblioteca, o que acarretou também na necessidade das implementações de suas funcionalidades.

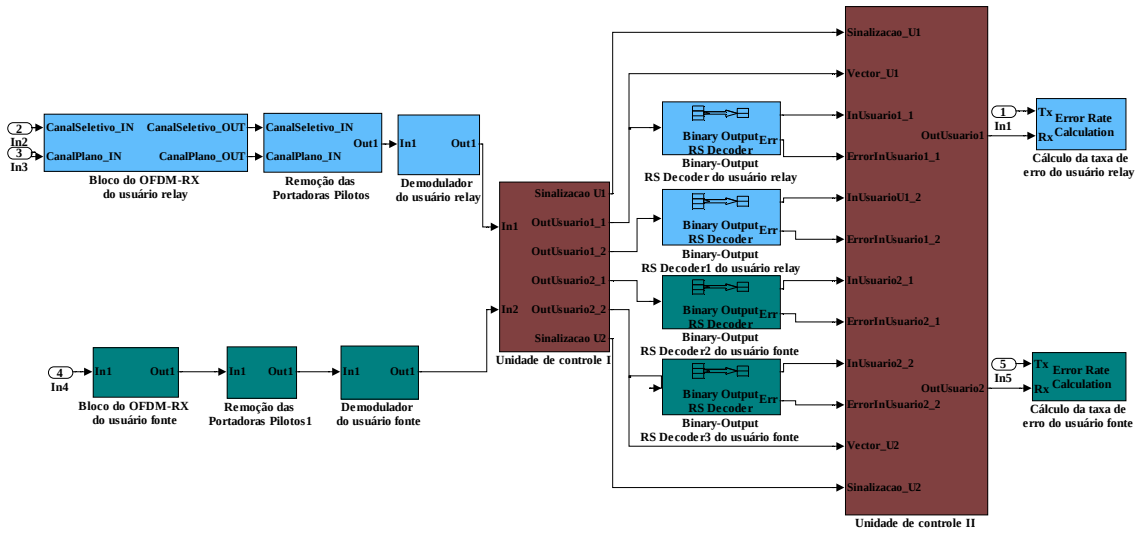


Figura B.4: *Diagrama em blocos da ERB.*

As figuras apresentadas neste anexo ilustram os blocos que compõem as simulações realizadas no Capítulo 3, enfatizando os blocos utilizados no terminal do usuário *relay* e na ERB.

Anexo C

Software da unidade de controle do usuário relay

```
function Template_unidade_controle_usuario_relay(block)
%   MSFUNTMPL A template for an M-file S-function
%   The M-file S-function is written as a MATLAB function with the
%   same name as the S-function. Replace 'msfuntmpl' with the name
%   of your S-function.
%
%   It should be noted that the M-file S-function is very similar
%   to Level-2 C-Mex S-functions. You should be able to get more
%   information for each of the block methods by referring to the
%   documentation for C-Mex S-functions.
%
%   Copyright 2003-2004 The MathWorks, Inc.
%   $Revision: 1.1.6.7 $

%%
%% The setup method is used to setup the basic attributes of the
%% S-function such as ports, parameters, etc. Do not add any other
```

```
%% calls to the main body of the function.
%%
setup(block);


%endfunction


%% Function: setup ===================================================
%% Abstract:
%%   Set up the S-function block's basic characteristics such as:
%%   - Input ports
%%   - Output ports
%%   - Dialog parameters
%%   - Options
%%
%%   C-Mex S-function counterpart: mdlInitializeSizes
%%
function setup(block)


[N] = evalin('base','N');


% Register number of ports
block.NumInputPorts  = 3;
block.NumOutputPorts = 1;


% Setup port properties to be inherited or dynamic
block.SetPreCompInpPortInfoToDynamic;
block.SetPreCompOutPortInfoToDynamic;
```

```
% Override input port properties
block.InputPort(1).DatatypeID  = 0;
block.InputPort(1).Dimensions = [(N*(log2(N+1))) 1];
block.InputPort(1).SamplingMode = 'Frame';
block.InputPort(2).DatatypeID  = 0;
block.InputPort(2).Dimensions = [(N*(log2(N+1))) 1];
block.InputPort(2).SamplingMode = 'Frame';
block.InputPort(3).DatatypeID  = 0;
block.InputPort(3).Dimensions = [1 1];
block.InputPort(3).SamplingMode = 'Frame';

% Override output port properties
block.OutputPort(1).DatatypeID  = 0;
block.OutputPort(1).Dimensions = [((N+1)*(log2(N+1))) 1];
block.OutputPort(1).SamplingMode = 'Frame';

% Register parameters
%   block.NumDialogPrms = 3;
%   block.DialogPrmsTunable = {'Tunable','Nontunable','SimOnlyTunable'};

% Register sample times
%  [0 offset]            : Continuous sample time
%  [positive_num offset] : Discrete sample time
%
%  [-1, 0]               : Port-based sample time = Inherent!
%  [-2, 0]               : Variable sample time
block.InputPort(1).DirectFeedthrough = true;
block.SampleTimes = [-1 0];
```

```
%% ------------------------------------------------------------------
%% Options
%% ------------------------------------------------------------------
% Specify if Accelerator should use TLC or call back into
% M-file
block.SetAccelRunOnTLC(false);

%% ------------------------------------------------------------------
%% The M-file S-function uses an internal registry for all
%% block methods. You should register all relevant methods
%% (optional and required) as illustrated below. You may choose
%% any suitable name for the methods and implement these methods
%% as local functions within the same file.
%% ------------------------------------------------------------------

%% ------------------------------------------------------------------
%% Register methods called during update diagram/compilation
%% ------------------------------------------------------------------

%%
%% CheckParameters:
%%   Required         : No
%%   Functionality    : Called in order to allow validation of
%%                      block's dialog parameters. User is
%%                      responsible for calling this method
%%                      explicitly at the start of the setup method
%%   C-Mex counterpart: mdlCheckParameters
```

```
%%
%block.RegBlockMethod('CheckParameters', @CheckPrms);


%%
%% SetInputPortSamplingMode:
%%   Required         : No
%%   Functionality    : Check and set input and output port
%%                      attributes specifying if port is operating
%%                      in sample-based or frame-based mode
%%   C-Mex counterpart: mdlSetInputPortFrameData
%%   (Signal Processing Blockset is required in order to set a port
%%    to be frame-based)
%%
block.RegBlockMethod('SetInputPortSamplingMode', @SetInpPortFrameData);


%%
%% SetInputPortDimensions:
%%   Required         : No
%%   Functionality    : Check and set input and optionally output
%%                      port dimensions
%%   C-Mex counterpart: mdlSetInputPortDimensionInfo
%%
block.RegBlockMethod('SetInputPortDimensions', @SetInpPortDims);


%%
%% SetOutputPortDimensions:
%%   Required         : No
%%   Functionality    : Check and set output and optionally input
```

```
%%                      port dimensions
%%   C-Mex counterpart: mdlSetOutputPortDimensionInfo
%%
block.RegBlockMethod('SetOutputPortDimensions', @SetOutPortDims);


%%
%% SetInputPortDatatype:
%%   Required         : No
%%   Functionality    : Check and set input and optionally output
%%                      port datatypes
%%   C-Mex counterpart: mdlSetInputPortDataType
%%
%block.RegBlockMethod('SetInputPortDataType', @SetInpPortDataType);


%%
%% SetOutputPortDatatype:
%%   Required         : No
%%   Functionality    : Check and set output and optionally input
%%                      port datatypes
%%   C-Mex counterpart: mdlSetOutputPortDataType
%%
%block.RegBlockMethod('SetOutputPortDataType', @SetOutPortDataType);


%%
%% SetInputPortComplexSignal:
%%   Required         : No
%%   Functionality    : Check and set input and optionally output
%%                      port complexity attributes
```

```
%%   C-Mex counterpart: mdlSetInputPortComplexSignal
%%
%block.RegBlockMethod('SetInputPortComplexSignal', @SetInpPortComplexSig);


%%
%% SetOutputPortComplexSignal:
%%   Required         : No
%%   Functionality    : Check and set output and optionally input
%%                      port complexity attributes
%%   C-Mex counterpart: mdlSetOutputPortComplexSignal
%%
%block.RegBlockMethod('SetOutputPortComplexSignal',@SetOutPortComplexSig);


%%
%% PostPropagationSetup:
%%   Required         : No
%%   Functionality    : Setup work areas and state variables. Can
%%                      also register run-time methods here
%%   C-Mex counterpart: mdlSetWorkWidths
%%
%block.RegBlockMethod('PostPropagationSetup', @DoPostPropSetup);


%% -----------------------------------------------------------------
%% Register methods called at run-time
%% -----------------------------------------------------------------


%%
%% ProcessParameters:
```

```
%%   Required         : No
%%   Functionality    : Called in order to allow update of run-time
%%                      parameters
%%   C-Mex counterpart: mdlProcessParameters
%%
%block.RegBlockMethod('ProcessParameters', @ProcessPrms);

%%
%% InitializeConditions:
%%   Required         : No
%%   Functionality    : Called in order to initalize state and work
%%                      area values
%%   C-Mex counterpart: mdlInitializeConditions
%%
%block.RegBlockMethod('InitializeConditions', @InitializeConditions);

%%
%% Start:
%%   Required         : No
%%   Functionality    : Called in order to initalize state and work
%%                      area values
%%   C-Mex counterpart: mdlStart
%%
%block.RegBlockMethod('Start', @Start);

%%
%% Outputs:
%%   Required         : Yes
```

```
%%   Functionality    : Called to generate block outputs in
%%                      simulation step
%%   C-Mex counterpart: mdlOutputs
%%
block.RegBlockMethod('Outputs', @Outputs);


%%
%% Update:
%%   Required         : No
%%   Functionality    : Called to update discrete states
%%                      during simulation step
%%   C-Mex counterpart: mdlUpdate
%%
%block.RegBlockMethod('Update', @Update);


%%
%% Derivatives:
%%   Required         : No
%%   Functionality    : Called to update derivatives of
%%                      continuous states during simulation step
%%   C-Mex counterpart: mdlDerivatives
%%
%block.RegBlockMethod('Derivatives', @Derivatives);


%%
%% Projection:
%%   Required         : No
%%   Functionality    : Called to update projections during
```

```
%%                      simulation step
%%   C-Mex counterpart: mdlProjections
%%
%block.RegBlockMethod('Projection', @Projection);


%%
%% ZeroCrossings:
%%   Required         : No
%%   Functionality    : For S-functions with continuous sample
%%                      time, if input signals have discontinuities,
%%                      this method is called to detect
%%                      discontinuities
%%   C-Mex counterpart: mdlZeroCrossings
%%
%block.RegBlockMethod('ZeroCrossings', @ZeroCrosssings);


%%
%% SimStatusChange:
%%   Required         : No
%%   Functionality    : Called when simulation goes to pause mode
%%                      or continnues from pause mode
%%   C-Mex counterpart: mdlSimStatusChange
%%
%block.RegBlockMethod('SimStatusChange', @SimStatusChange);


%%
%% Terminate:
%%   Required         : Yes
```

```
%%   Functionality    : Called at the end of simulation for cleanup
%%   C-Mex counterpart: mdlTerminate
%%
%block.RegBlockMethod('Terminate', @Terminate);

%% ------------------------------------------------------------------
%% Register methods called during code generation
%% ------------------------------------------------------------------

%%
%% WriteRTW:
%%   Required         : No
%%   Functionality    : Write specific information to RTW file
%%   C-Mex counterpart: mdlRTW
%%
%block.RegBlockMethod('WriteRTW', @WriteRTW);
%endfunction

%% ------------------------------------------------------------------
%% The local functions below are provided for illustrative purposes
%% to show how you may implement the various block methods listed
%% above.
%% ------------------------------------------------------------------

% function ProcessPrms(block)
%
%  block.AutoUpdateRuntimePrms;
%
```

```
% endfunction

function SetInpPortFrameData(block, idx, fd)

block.InputPort(1).SamplingMode = fd;
block.InputPort(2).SamplingMode = fd;
block.InputPort(3).SamplingMode = fd;
block.OutputPort(1).SamplingMode  = fd;

%endfunction

function SetInpPortDims(block, idx, di)
[N] = evalin('base','N');
block.InputPort(1).Dimensions = [(N*(log2(N+1))) 1];
block.InputPort(2).Dimensions = [(N*(log2(N+1))) 1];
block.InputPort(3).Dimensions = [1 1];

%endfunction

function SetOutPortDims(block, idx, di)
[N] = evalin('base','N');
block.OutputPort(1).Dimensions = [((N+1)*(log2(N+1))) 1];


function Outputs(block)

[N] = evalin('base','N');
[NumeroSimbolosCooperacao] = evalin('base','NumeroSimbolosCooperacao');
```

```
NumeroBitsCooperacao = NumeroSimbolosCooperacao*log2(N+1);

VectorNullTx = zeros(4,1);
VectorUmTx = ones(4,1);

VectorInU1 = [];
VectorDadosU1 = [];
VectorDadosU1_U2 = [];
VectorOutU1 = [];
VectorInU1 = block.InputPort(1).Data;
VectorDadosU1 = VectorInU1(1:((N*log2(N+1))-NumeroBitsCooperacao));
VectorDadosU1_U2 = VectorInU1(((N*log2(N+1))-NumeroBitsCooperacao+1):
                    (N*log2(N+1)));

U2_ERROR_IN = block.InputPort(3).Data;

VectorInU2 = [];
VectorDadosU2 = [];
VectorDadosU2_U1 = [];
VectorOutU2 = [];
VectorInU2 = block.InputPort(2).Data;
VectorDadosU2 = VectorInU2(1:((N*log2(N+1))-NumeroBitsCooperacao));
VectorDadosU2_U1 = VectorInU2(((N*log2(N+1))-NumeroBitsCooperacao+1):
                    (N*log2(N+1)));

[AuxControleCooperacao_U1] = evalin('base','ControleCooperacao_U1');

if ((U2_ERROR_IN ~= -1) && (AuxControleCooperacao_U1 == 1) )
```

```
    VectorOutU1 = vertcat(VectorDadosU1,VectorDadosU2_U1,
    ones((log2(N+1)),1));
    block.OutputPort(1).Data = VectorOutU1;
else
    block.OutputPort(1).Data = vertcat(VectorInU1,
    zeros((log2(N+1)),1));
end
```

Anexo D

Publicações oriundas da dissertação.

Artigos publicados

- *On the applicability of Reed-Solomon codes in the cooperative coding.*
  Tiago Reis Rufino Marins, Geraldo Gil Ramundo Gomes e Luciano Leonel Mendes. International Workshop Telecommunication. March, 2011. Rio de Janeiro, Rio de Janeiro.

Artigos em desenvolvimento

- *Um Esquema de Codificação Cooperativa para Sistemas OFDM em Canais Seletivos em Frequência.*
  Tiago Reis Rufino Marins, Luciano Leonel Mendes e Geraldo Gil Ramundo Gomes.

Referências Bibliográficas

[1] White paper 4G Americas, 4G Mobile Broadband Evolution: Release 10, Release 11 and Beyond - HSPA+, SAE/LTE and LTE-Advanced, October 2012. [Online]. Disponível em: http://:www.4gamericas.org/documents/4G%20Mobile%20Broadband%20Evolution-Rel%2010%20Rel%2011%20and%20Beyond%20October%202012.pdf. Acesso em: 11/11/12.

[2] GLISIC, S.. *Advanced wireless communications: 4G technologies.* New Delhi: John Wiley & Sons, 2004.

[3] HALONEN, T.; ROMERO, J.; MELERO, J.. *GSM, GPRS, and EDGE performance: Evolution towards 3G/UMTS.* New Delhi: John Wiley & Sons, 2nd ed., 2003.

[4] HOLMA, H.; TOSKALA, A.. *LTE for UMTS-OFDMA and SC-FDMA based radio access.* New Delhi: JohnWiley & Sons, 1st ed., 2009.

[5] YACOUB, M.. *Foundation of Mobile Radio Engineergin*, CRC Press, Year 1993.

[6] BENEDETTO, S. ;BIGLIERI, E.. Principles of Digital Transmission With Wireless Applications, Plenum Pub Corp, 1999.

[7] SIMON, M. K.; ALOUINI, M. S.. *Digital Communication over Fading Channels. A Unified Approach to Performance Analysis*. 1st ed., Wiley, 2000.

[8] HOUSSAIN, Ekran; KIM, Dong In; BHARGAVA, Vijay K..*Cooperative Cellular Wireless Networks*. Cambridge University Press, 2011.

[9] LIU, K. J. R. & SADEK, Ahmed K.&SU, Weifeng &KWASINSK, Andres *Cooperative Communications and Networking*. Cambridge University Press, 2009.

[10] VALENTIN, Stefan; LICHTE, Hermann S.; KARL Holger, VIVIER, Guillaume, SIMOENS, S´ebastien, VIDAL, Josep, AGUSTIN, Adrian & AAD, Imad. *Cooperative wireless networking beyond store-and-forward: Perspectives for PHY and MAC design*, WWRF WG3, 2006.

[11] NOSRATINIA, Aria; HUNTER, Todd E.; HEDAYAT, Ahmadreza. *Cooperative Communication in Wireless Networks*. IEEE Communications Magazine, October 2004.

[12] HUANG, Wan-Jen; HONG, Yao-Win; KUO, C. C. Jay. *Decode-and-Forward Cooperative Relay with Multi-User*. Global Telecommunications Conference, pp. 4397-4401, November 2007.

[13] KRAMER, Gerhard; GASTPAR, Michael; GUPTA, Piyush. *Cooperative Strategies and Capacity Theorems for Relay Networks*. IEEE Transaction on Information Theory, Vol. 51, No. 9, September 2005.

[14] CALDAS, Francisco M. C.; BANDEIRA, Carlos I. R.; FREITAS, Walter C. Jr.. *Transceptor MIMO em Sistemas de Comunicações Móveis Sem Fio com Multipontos Coordenados*. Simpósio Brasileiro de Telecomunicações 2012, Brasília, Brasil, Setembro 2012.

[15] HUNTER, T.; NOSRATINIA, A..*Diversity through coded cooperation*. IEEE Transaction on Wireless Communications, Vol. 5, No. 2, February 2006.

[16] NOSRATINIA, A.; HUNTER, T. E.; HEDAYAT, A..*Cooperative communication in wireless networks*. IEEE Communications Magazine, October 2004.

[17] STEFANOV, A.; ERKIP, E.. *Cooperative coding for wireless networks*. In Proc. IEEE Transactions on Communications, Vol. 52, NO. 9, September 2004.

[18] HUNTER, T. E.; NOSRATINIA, A..*Cooperation Diversity Through Coding*. In Proc. IEEE Int. Symp. Information Information Theory, Lausanne, Switzerland, June 2002.

[19] LANEMAN, J. N.; TSE, D. N. C.;WORNELL, G. W.. *Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior*. IEEE Trans. Inform. Theory, vol. 50, no. 12, pp. 3062-3080, Dec. 2004.

[20] SENDONARIS, A.; ERKIP, E.; AAZHANG B.. *User Cooperation Diversity - Part I: System Description*. IEEE Trans. on Communications, vol. 51, pp. 1927-1938, November 2003.

[21] SENDONARIS, A.; ERKIP, E.; AAZHANG, B.. *User Cooperation Diversity - Part II: Implementation Aspects and Performance Analysis*. IEEE Trans. on Communications, vol. 51, pp. 1939-1948, November 2003.

[22] SENDONARIS, A.. *Techniques for Next-Generation Wireless Systems*. Ph.D. thesis, Rice University, May 1999.

[23] LANEMAN, J. N.; WORNELL; TSE, D. n. C..*An efficient protocol for realizing cooperative diversity in wireless networks*. IEEE International Symposium on Information Theory (ISIT), p. 294, June 2001.

[24] HOST-MADSEN, A.; ZHANG, J..*Capacity bounds and power allocation for wireless relay channels*. IEEE Transaction on Information Theory, vol. 51, no. 6, pp. 2020-2040, June 2005.

[25] COVER, T. M.; GAMAL, A. A. E..*Capacity Theorems for the Relay Channel*. IEEE Trans. Info. Theory, vol. 25, no. 5, Sept. 1979, pp 572-84.

[26] GALLAGER, R.. *Communications and Cryptography: Two Sides of One Tapestry*. ser. in Engineering & Computer Science. Kluwer, 1994.

[27] LIN, S.; COSTELLO, Daniel J.. *Error Control Coding*. 2nd Edition, Prentice Hall, 2004.

[28] BAHAI, A. R. S.; SINGH, M.; GOLDSMITH, A. J.; SALTZBERG, B. R.. *Multi-Carrier Digital Communications - Theory and Applications of OFDM*. Kluwer Academic, 1999.

[29] IEEE. Part 16: Air Interface for Fixed Broadband Wireless Access Systems - Amendment 2: Medium Access Control Modifications and Additional Physical Layer Specefications for 2-11GHz. Jan. 2003. Padrão IEEE Std 802.16a-2003, suplementar ao IEEE Std 802.16-2001, publicado pelo IEEE.

[30] IEEE. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-Speed Physical Layer in the 5GHz Band. Set. 1999. Padrão IEEE Std 802.11a-1999, suplementar ao IEEE Std 802.11-1999, publicado pelo IEEE.

[31] MENDES, Luciano L.; BALDINI, Renato. *On Performance of Channel Estimation Algorithms for STC-OFDM System in Non Linear Channels*. Proceding of ISCTA 2009, Ambleside, UK, 2009.

[32] MENDES, Luciano L.; BALDINI, Renato. *Performance of WHT-STC-OFDM in Mobile Frequency Selective Channel*. Proceding of ITS 2010, Manaus, Brazil, 2010.

[33] RAMABADRAN, T.V.; GAITONDE, S.S..*A tutorial on CRC computations*. IEEE Micro 8 (4): 62-75, 1988.

[34] PAULRAJ, A.; NABAR, R.; GORE, D..*Introduction to Space-Time Wireless Communications*. Cambridge University Press, May 29, 2003

[35] LANEMAN, J. N.; WORNELL, G. W..*Distributed Space-Time Coded Protocols for Exploiting Cooperative Diversity in Wireless Networks*. IEEE Transactions on Information Theory, vol. 49, no. 10, October 2003.

[36] REDL, Siegmund H.; WEBER, Mathias K.; OLIPHANT, Malcolm W.. *An Introduction to Gsm*. Artech House, 1995.

[37] AHMADI, Sassan. *Mobile WiMAX: A Systems Approach to Understanding IEEE 802.16m Radio Access Technology*. 1st Edition, academic Press, 2010.

[38] HOLMA, Harri; TOSKALA, Antti. *LTE for UMTS*. 2nd Edition, John Wiley & Sons, 2011.

[39] HAYKIN, S.. *Communication System*. 5th Edition, John Wiley & Sons, 2010.

[40] AHMED, N.; RAO, B. R.. *Orthogonal Transforms for Digital Signal Processing*. Springer Verlag,1975.

[41] YATE, R. D.; GOODMAN, D. J..*Probability and Stochastic Processes: A Friendly Introduction for Electrical and Computer Engineers*. John Wiley,1999.