

# Confiabilidade do canal em sistemas DS/CDMA com codificação turbo sujeitos ao desvanecimento lento e seletivo em frequência e interferência de múltiplo acesso

## Turbo code channel reliability for DS/CDMA systems under slow and frequency selective fading and multiple access interference

Wagner Okano<sup>1</sup>; Fernando Ciriaco<sup>2</sup>; Taufik Abrão<sup>2</sup>

### Resumo

---

Neste trabalho analisa as estratégias de codificação turbo e convolucional aplicadas ao problema da detecção uniusuário (SuD – *Single User Detection*) em sistemas de múltiplo acesso DS/CDMA (*Direct Sequence/Code Division Multiple Access*) considerando canais de multipercorso com desvanecimento lento e seletivo em frequência. Neste trabalho, propõe-se uma nova equação para o fator de confiabilidade do canal DS/CDMA, obtida a partir de resultados de desempenho de simulação Monte Carlo (MCS). Comparações com resultados da literatura indicaram superioridade de desempenho utilizando a equação proposta na região de alta,  $E_b/N_0$  e similaridade de desempenho para as demais regiões de  $E_b/N_0$  em relação aos resultados relatados na literatura. Adicionalmente, este trabalho faz uma análise do compromisso complexidade/requisito de memória versus desempenho de receptores DS/CDMA considerando decodificadores turbo (no contexto do padrão CDMA2000) e de Viterbi (no contexto do padrão IS-95).

**Palavras-chave:** Confiabilidade do Canal. Códigos Turbo. DS/CDMA. Detecção uniusuário. Canais de Multipercorso.

### Abstract

---

This work analyzes the Turbo and Convolutional codes strategies applied to the single-user detection (SuD) problem in multiple access systems DS/CDMA, in multipath slow Rayleigh channels. In this work, a new equation for the DS/CDMA channel reliability, obtained from Monte Carlo Simulations (MCS), is proposed. Comparison with the literature results and has indicated a superior performance in the high  $E_b/N_0$  regions; in other  $E_b/N_0$  regions the proposed equation has similar performance than those described in the literature. Additionally, this work makes an analysis of the complexity/memory requirements versus performance of DS/CDMA receivers considering turbo (CDMA2000) and Viterbi (IS-95) decoders.

**Key words:** Channel Reliability. Turbo-Codes. DS/CDMA. Single-user Detection. Multipath Channels.

---

<sup>1</sup> Engenheiro de Telecomunicações da Sercomtel Celular S.A, Londrina, PR. Atualmente é estudante do programa de mestrado do Departamento de Engenharia Elétrica da Universidade Estadual de Londrina (DEEL-UEL); e-mail: wjokano@sercomtel.com.br

<sup>2</sup> Docentes do Departamento de Engenharia Elétrica da Universidade Estadual de Londrina – DEEL-UEL; fciriaco@uel.br, taufik@uel.br

## Introdução

O desvanecimento e a interferência de múltiplo acesso (MAI – *Multiple Access Interference*) são os dois principais fatores que limitam o desempenho de sistemas DS/CDMA. Em particular, detectores multiusuário (MuD – *Multiuser Detection*) são utilizados para combater a MAI enquanto a diversidade espaço-temporal é utilizada no combate ao desvanecimento. A aplicação de códigos corretores de erro no receptor (FEC – *Forward Error Correction*) é utilizada para combater a MAI e o desvanecimento de forma conjunta.

A codificação é freqüentemente usada em sistemas de comunicação digitais para proteger a informação do ruído e da interferência e, desse modo, reduzir o número de erros de bits, consistindo na inserção de bits de redundância no sinal de informação. Estes bits adicionais visam a ajudar a detecção e correção de erros proporcionando uma maior confiabilidade na informação recebida. O uso da codificação implica em uma redução na taxa de transmissão ou uma expansão na largura da banda.

Em 1948, Claude E. Shannon (1948) em sua teoria matemática da comunicação, descobriu um parâmetro calculável denominado capacidade de canal, prevendo que um determinado sistema de comunicação pode operar livre de erros, desde que a taxa de transmissão não seja maior que a capacidade do canal. A codificação do canal é o processo em que a redundância controlada é adicionada à informação objetivando a detecção e a correção de erros introduzidos pelas características do canal. Essa adição controlada de redundância leva à redução da quantidade de erros na recepção, apesar da inevitável redução na taxa líquida de informação trafegada.

As técnicas de codificação do canal podem ser classificadas de acordo com dois critérios: a) quanto a utilização e b) quanto ao tipo de código. O primeiro é dividido na técnica da correção dos erros no receptor (FEC), em que o código é usado para tentar obter uma taxa de erro menor após a

decodificação no receptor e na técnica da detecção dos erros, em que somente a existência dos erros no sinal recebido é detectada. O segundo critério pode ser subdividido em: códigos de bloco, códigos convolucionais e codificação turbo.

Nos códigos de bloco, para cada bloco de bits da fonte, é gerado um novo bloco de bits codificado, necessariamente maior que o bloco da fonte. A codificação neste caso é sem memória, i.e., cada bloco de bits codificado depende somente do bloco atual de bits da fonte considerada. A codificação por blocos apresenta diversas desvantagens, tais como a necessidade do recebimento completo da palavra de código antes de iniciar o processo de decodificação, necessidade de sincronismo de quadro, degradação considerável de desempenho na região de baixa relação sinal-ruído (SNR). Os códigos de blocos são utilizados na detecção dos erros, especialmente para verificação de redundância cíclica (CRC – *Cyclic Redundancy Check*), cuja decodificação, relativamente simples, é obtida por divisão polinomial. Os códigos de blocos são usados às vezes também para a correção de erros, por exemplo a Reed-Solomon.

Por sua vez, os códigos convolucionais são usados sempre como corretores de erros, e foram introduzidos inicialmente em (ELIAS, 1955). Nos códigos convolucionais, uma seqüência contínua de bits de informação, com tamanho variável, é mapeada em uma seqüência também contínua de bits codificados, e um bit codificado depende de um ou mais bits de informação anteriores, combinados de forma linear. Tal configuração constitui um codificador com memória, e resulta em uma estrutura de codificação mais complexa que não pode ser dividida em palavras de código. Desde o advento da codificação com memória, vários algoritmos de decodificação foram propostos, entre eles destacam-se (WOZENCRAFT, 1957; WOZENCRAFT; REIFFEN, 1961), (FANO, 1963) e (MASSEY, 1963).

Uma invenção marcante para o processo de decodificação baseada na estimação da seqüência de

máxima probabilidade foi estabelecida em 1967 com o advento do algoritmo de Viterbi (VA) (VITERBI, 1967). Uma interpretação clássica do algoritmo de Viterbi pode ser encontrada em (FORNEY, 1973) e uma das primeiras aplicações foi proposta em (HELLER; JACOBS, 1971) durante o início dos anos de 1970. Entretanto, o algoritmo de Viterbi não garante que seja atingida a menor taxa de erro de bit (BER) possível, esse algoritmo foi proposto em (BAHL et al., 1974), sendo denominado algoritmo de máxima probabilidade *a posteriori* (MAP – *Maximum A Posteriori Probability*) (BAHL ET AL., 1974; HAGENAUER; OFFER; PARKE, 1996). Devido à sua alta complexidade de implementação, à época foi pouco utilizado na prática.

O algoritmo de Viterbi (LEE; MESSERSHMITT, 1994; WICKER, 1995) utiliza a decodificação suave que minimiza a probabilidade de erros na decisão sobre os bits de informação transmitidos quando aplicado na decodificação convolucional que utiliza o critério de máxima verossimilhança.

Na década de 70, os códigos FEC foram incorporados aos sistemas espaciais e de comunicação por satélites e tornaram-se comuns em sistemas de comunicação rádio móvel celular nos anos 80. Porém, por muitos anos, a modulação e os códigos FEC foram tratados como assuntos distintos em sistemas de comunicação, até que em 1987, Ungerboeck (UNGERBOECK, 1987a, 1987b) propôs a modulação codificada por treliça (TCM – *Trellis Coded Modulation*), com a qual é possível obter ganhos significativos de codificação sem exigir uma expansão de largura de banda nem a diminuição da taxa de transmissão de bits.

Em 1993, os códigos turbo (TC – Turbo-Codes) foram apresentados por Berrou, Glavieux e Thitimajshima (BERROU; GLAVIEUX; THITIMAJSHIMA, 1993; BERROU; GLAVIEUX, 1996), os quais obtiveram uma grande repercussão por apresentarem um desempenho próximo ao limite de Shannon (SHANNON, 1948). O código turbo é obtido pela concatenação paralela (PCCC – *Parallel Concatenated Convolutional Codes*) de dois códigos

convolucionais sistemáticos recursivos (RSC – *Recursive Systematic Convolutional*), separados por um entrelaçador (*interleaver*). Essa codificação com capacidade de correção de erros, apesar de possuir uma estrutura relativamente simples é capaz de transmitir a informação por meio do canal e prover uma taxa de erro de bit (BER – *Bit Error Rate*) muito reduzida. O desempenho de um código turbo é fortemente influenciado pelo *interleaver* utilizado, o qual proporciona ao código turbo uma aparência aleatória, gerando poucos códigos com distância mínima de Hamming. Outra vantagem da codificação turbo é que existem diversas estruturas capazes de decodificar eficientemente a informação.

Desde 1993, inúmeras pesquisas têm sido realizados na área de codificação turbo, buscando reduzir a complexidade associada ao decodificador. Constituem decodificadores práticos de reduzida complexidade os algoritmos Max-Log-MAP, proposto em (KOCH; BAIER, 1990) e também em (ERFANIAN; PASUPATHY; GULAK, 1994), o algoritmo Log-MAP sugerido por (ROBERTSON; VILLEBRUN; HOEHER, 1995) e o algoritmo SOVA proposto em (HAGENAUER; HOEHER, 1989; HAGENAUER, 1995). Por sua vez, em (GOFF; GLAVIEUX; BERROU, 1994), (WACHSMANN; HUBER, 1995), além de (ROBERTSON; WORZ, 1997), sugeriram o uso destes códigos junto a esquemas eficientes de modulação. Durante meados dos anos 90, o trabalho (HAGENAUER; OFFER; PARKE, 1996), além de (PYNDIAH, 1998), ampliaram o conceito dos códigos em blocos concatenados introduzindo o conceito de códigos em blocos concatenados paralelos. Em 1997, Nickl, Hagenauer e Burkett mostraram que o limite de Shannon pode ser aproximado, com uma separação de 0,27 dB, por meio do emprego de um simples código turbo de Hamming. Por sua vez, Barbulescu e Pietrobon (1994) introduziram projetos de *interleaver*. Uma visão compreensiva sobre codificação turbo pode ser encontrada em Benedetto e Montorsi (1996a, 1996b) e Perez, Seghers e Costello (1996).

Os códigos turbo saíram muito rapidamente dos laboratórios de pesquisa, para serem aplicados na prática em todo o mundo. Depois de seu anúncio em 1993, foram utilizados para diversos fins, principalmente nas comunicações sem fio, da terceira geração (3G) das comunicações móveis aos sistemas de exploração espaciais (BURR, 2001). A utilização de códigos turbo aliado a um esquema de decodificação iterativa resulta em diversas possibilidades de sistemas capazes de fornecer confiabilidade nas comunicações, com uma relação sinal-ruído próxima a algumas unidades de decibéis (dB) do limite de Shannon (SKLAR, 1997).

Os códigos convolucionais concatenados em série (SCCC – *Serial Concatenated Convolutional Code*) (BENEDETTO et al., 1997) surgiram como uma alternativa de melhoria nos ganhos de codificação quando se utilizam entrelaçadores de dimensão reduzida. Em contrapartida aos PCCC, os códigos SCCC possuem uma estrutura de concatenação em série, porém, similarmente aos PCCC podem ser decodificados por meio de estruturas iterativas. Os SCCC apresentam a característica de redução da BER mais eficaz com o aumento da dimensão dos entrelaçadores e de eliminar o patamar de BER irreduzível existente nos PCCCs (BENEDETTO et al., 1997; SOLEYMANI; GAO; VILAIORNISAWAI, 2002).

Diversos fatores devem ser considerados na escolha do esquema de codificação: complexidade, taxa de codificação, capacidade de correção de erro e a mínima relação sinal-ruído que permite uma recepção com BER desejada, entre outros.

Neste trabalho serão utilizadas as duas principais estratégias de decodificação existentes: a MAP e o algoritmo de Viterbi (VA – *Viterbi Algorithm*) (VITERBI, 1967; LIN; COSTELLO, 1983; PROAKIS; SALEHI, 1994; WICKER, 1995). Comparando-se a estratégia de decodificação turbo com a decodificação convolucional, que utiliza o algoritmo de Viterbi, nota-se que o ganho de desempenho dos códigos turbo depende de um bloco

de codificação de grande comprimento. Devido ao atraso introduzido pelo *interleaver* utilizado na codificação turbo, estes são mais apropriados para transmissões de dados, visto que as aplicações de voz e videoconferência são sensíveis a atrasos na transmissão.

Nos sistemas sem fio 3G e vindouros, o aumento da capacidade do canal direto (*downlink*) é fundamental para garantir qualidade às aplicações que necessitam de alta taxa de dados, tais como o acesso à Internet. Com isso, os códigos turbo foram adotados em vários padrões móveis 3G, tais como o WCDMA e CDMA2000, principalmente para as aplicações que requerem elevadas taxa de dados (3GPP, 2000). No entanto, existem poucas pesquisas que analisam o desempenho dos códigos turbo considerando canais seletivos em frequência (HALL; WILSON, 1998; SHIN; KIN; LEE, 2000). Este trabalho analisa o compromisso desempenho *versus* complexidade da decodificação turbo aplicada aos sistemas DS/CDMA em canais seletivos em frequência. O foco principal deste trabalho é a codificação e a decodificação turbo utilizada nos sistemas móveis 3G, e a discussão acerca de codificação e decodificação convolucional tem importância do ponto de vista de comparação com a turbo além de sua importância histórica, tendo sido utilizado para esta comparação os modelos relativos às especificações do padrão IS-95 e cdma2000 (3GPP, 2000; TIA/EIA/IS-2000-2, 1999).

Este trabalho está dividido em seis seções. A seção “Modelo de Sistema DS/CDMA” descreve o modelo do sistema DS/CDMA em canais de multipercurso com desvanecimento e também o princípio da codificação convolucional da informação. Em seguida, a seção “Decodificação e confiabilidade do canal” revisa as estratégias de decodificação turbo e o algoritmo de Viterbi, bem como propõe uma equação para a medida da confiabilidade de canal. Os resultados numéricos do desempenho para a decodificação turbo e para o algoritmo de Viterbi, em termos de BER, são apresentados na seção “Resultados Numéricos”.

A seção “Complexidade Computacional” analisa a complexidade computacional dos codificadores e decodificadores em termos do número de transições na treliça e do requisito de memória. Finalmente, na seção “Conclusões” são sintetizadas as principais conclusões deste trabalho.

### Modelo de Sistema DS/CDMA

Em um sistema DS/CDMA com modulação por chaveamento de fase binária (BPSK – *Binary Phase-Shift Keying*) o sinal transmitido pelo  $k$ -ésimo usuário em um sistema com  $K$  usuários assíncronos ativos é dado por (PROAKIS, 1995; VERDÚ, 1998):

$$z_k(t) = \sqrt{2P_k} \sum_i x_k^{(i)} s_k(t - iT_b - \tau_{k,l}) \cos(\omega_c t), \quad (1)$$

Sendo  $P_k = A_k^2/2$  representa a potência de transmissão do  $k$ -ésimo usuário e  $A_k$  a amplitude;  $x_k^{(i)}$  é o  $i$ -ésimo símbolo BPSK transmitido com período  $T_b$ , relacionado ao  $k$ -ésimo bit de informação na codificação turbo,  $\omega_c$  é a frequência angular da portadora;  $s_k(t) = \sum_{n=0}^{N-1} p(t - nT_c) \underline{s}_{k,n}$  é a seqüência de espalhamento definida no intervalo  $[0, T_b)$  e zero fora, onde  $\underline{s}_{k,n} \in \{-1; 1\}$  é o  $n$ -ésimo *chip* da seqüência de comprimento  $N$  utilizada pelo  $k$ -ésimo usuário;  $T_c$  é o período de *chip* e o ganho de processamento,  $T_b/T_c$ , é igual a  $N$ ; assume-se formatação de pulso  $p(t)$ , retangular com amplitude  $1/\sqrt{N}$  no intervalo  $[0; T_c)$  e zero fora;  $\tau_{k,l} = \Delta_{k,l} + d_k$  representa um atraso aleatório no  $[0, N-1] T_c$  sendo  $d_k$  constitui o atraso de propagação e  $\Delta_{k,l}$  o atraso do  $l$ -ésimo percurso para o  $k$ -ésimo usuário.

Assumindo um pacote de dados (*frame*) com  $I$  bits para cada usuário, propagando-se através de  $L$  percursos independentes com desvanecimento Rayleigh lento, o sinal recebido em banda base (assumindo-se filtro passa-baixa ideal) e notação vetorial é dado por (VERDÚ, 1998):

$$r(t) = \sum_{i=0}^{I-1} \mathbf{s}^T(t - iT_b) \mathbf{a} \mathbf{c}^{(i)} \mathbf{b}^{(i)} + \eta(t) \quad (2)$$

sendo:  $\mathbf{s}(t) = [s_1(t - \tau_{1,1}), s_1(t - \tau_{1,2}), \dots, s_1(t - \tau_{1,L}); \dots; s_k(t - \tau_{k,1}), \dots, s_k(t - \tau_{k,L})]^T$  é a matriz das seqüências de espalhamento,

$\mathbf{a} = \text{diag}[\sqrt{P_1} \mathbf{I}; \sqrt{P_2} \mathbf{I}; \dots; \sqrt{P_K} \mathbf{I}]$  é a matriz diagonal das amplitudes recebidas incluindo os efeitos de perda de percurso e sombreamento, onde  $\mathbf{I}_{L \times L}$  é a matriz identidade de dimensão  $L$ , e a matriz diagonal de ganho de canal é  $\mathbf{c}^{(i)} = \text{diag}[c_{1,1}^{(i)}, \dots, c_{1,L}^{(i)}; c_{2,1}^{(i)}, \dots, c_{2,L}^{(i)}; \dots; c_{K,1}^{(i)}, \dots, c_{K,L}^{(i)}]$ , e o vetor de dados é dado por  $\mathbf{b}^{(i)} = [b_1^{(i)}; b_2^{(i)}; \dots; b_K^{(i)}]$  com  $\mathbf{b}_k^{(i)}$  representando o vetor de bits com dimensão  $1 \times L$  referente ao  $k$ -ésimo usuário.

A saída do filtro casado considera uma recepção coerente para o  $k$ -ésimo usuário e corresponde ao  $l$ -ésimo componente multipercurso amostrado ao final do  $i$ -ésimo intervalo de bit é  $\int_{-\infty}^{+\infty} r(t) s_k(t - iT_b - \tau_{k,l}) dt$ , e desse resulta:

$$u_{k,\ell}^{(i)} = \sqrt{P_k T_b} \Psi_{k,\ell}^{(i)} b_k^{(i)} + SI_{k,\ell}^{(i)} + I_{k,\ell}^{(i)} + \eta_{k,\ell}^{(i)} \quad (3)$$

sendo o primeiro termo corresponde ao sinal desejado, o segundo à auto-interferência, o terceiro à interferência de múltiplo acesso sobre o  $l$ -ésimo componente multipercurso do  $k$ -ésimo usuário e o último ao AWGN (*Additive White Gaussian Noise*) filtrado.

### Decodificação e confiabilidade do canal

A seguir, analisa-se o desempenho do critério MAP e do algoritmo de Viterbi, considerando suas respectivas complexidades, bem como propõe-se uma nova equação para a medida da confiabilidade de canal.

### Algoritmo BCJR

Bahl et al. (1974) publicaram um algoritmo de decodificação de códigos baseado em probabilidades *a posteriori* denominado algoritmo BCJR, algoritmo MAP ou ainda “*forward-backward algorithm*”. O procedimento pode ser aplicado tanto a códigos de blocos como a códigos convolucionais mas, como possui uma maior complexidade que o algoritmo de Viterbi, durante cerca de vinte anos não o utilizaram na prática, situação que foi substancialmente alterada com o advento dos códigos turbo em 1993 (BERROU; GLAVIEUX; THITIMAJSHIMA, 1993).

Existem diversas versões simplificadas do algoritmo MAP, nomeadas log-MAP (*Logarithmic Maximum a Posteriori Probability*) (ROBERTSON; VILLEBRUN; HOEHER, 1995) e max-log-MAP (*Maximum Logarithmic Maximum a Posteriori Probability*) (BENEDETTO et al., 1995, 1996). Este trabalho visa a comparar o desempenho do critério MAP e do algoritmo de Viterbi, considerando suas respectivas complexidades.

Considerando modulação BPSK, o bit de informação à entrada do codificador,  $b_k^{(i)} = \pm 1$ , possui uma probabilidade de ocorrência *a priori*  $P(b_k^{(i)})$ . Associada a esta probabilidade, define-se a relação de verossimilhança logarítmica, denominada LLR (*log-likelihood ratio*), como:

$$L(b_k^{(i)}) = \ln \frac{P(b_k^{(i)} = +1)}{P(b_k^{(i)} = -1)} \quad (4)$$

Se os bits  $b_k^{(i)} = \pm 1$ , forem equiprováveis a LLR *a priori* é nula.

A seqüência codificada  $\mathbf{y}_k$  que chega ao decodificador, considerando um canal com desvanecimento multipercurso, é expressa pela composição das diversas saídas do combinador MRC, equação (3):

$$\mathbf{y}_k(t) = [y_k^{(1)} \quad y_k^{(2)} \quad \dots \quad y_k^{(I)}] \quad (5)$$

A partir da chegada desta seqüência, o algoritmo BCJR, ou qualquer outro, procurará estimar a seqüência de bits originais  $b_k^{(i)}$ . Utiliza-se, neste contexto, a LLR *a posteriori*, definida pela razão:

$$L(b_k | \mathbf{y}_k) = \ln \frac{P(b_k = +1 | \mathbf{y}_k)}{P(b_k = -1 | \mathbf{y}_k)} \quad (6)$$

O sinal da LLR indica a estimativa do bit enviado, e o seu valor absoluto representa uma maior ou menor confiabilidade na estimativa obtida, ou seja, quanto mais afastado o valor de  $L(b_k | \mathbf{y}_k)$  estiver do limiar de decisão, maior é o grau de confiança na estimativa do bit.

A informação contida em  $L(b_k | \mathbf{y}_k)$  pode ser passada a outro elemento de decodificação, se houver, ou convertida em decisões abruptas. Note-se que o numerador e o denominador da equação (6) são probabilidades condicionais *a posteriori*, ou seja, probabilidades calculadas após a chegada completa da seqüência  $\mathbf{y}_k$ . Em outras palavras, têm-se as probabilidades de no instante  $i$  o bit à entrada do codificador ter sido  $b_k^{(i)} = +1$  ou  $b_k^{(i)} = -1$ , conhecendo toda a seqüência  $\mathbf{y}_k$ . São estas as probabilidades condicionais que o algoritmo BCJR original (BAHL et al., 1974) de fato calcula e compara.

No contexto de codificação, é comum utilizar a idéia de que a seqüência codificada percorre um caminho numa árvore de possibilidades de tamanho fixo, chamada de treliça. Visando a esclarecer o conceito de codificação, apresenta-se um exemplo de codificador convolucional de taxa 1/2, resultando em quatro estados possíveis  $\mathbf{S} = \{0, 1, 2, 3\}$  considerando um instante de tempo  $i$  arbitrário. Adotou-se que um bit  $-1$  na entrada do codificador produz um ramo de traço contínuo e um bit  $+1$  produz

um ramo tracejado e cada ramo está rotulado com a correspondente saída de dois bits  $x_k^{(i)}$ , em que 0 e 1 correspondem a  $-1$  e  $+1$ , respectivamente. A figura 1 representa uma treliça do codificador utilizado.

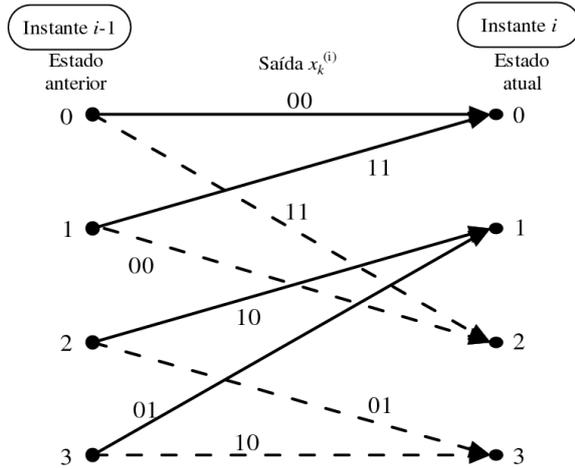


Figura 1. Diagrama em treliça do codificador utilizado.

No instante  $i$ , adota-se que o estado correspondente é  $S_i = s$ , o estado anterior é  $S_{i-1} = s'$  e o símbolo recebido no decodificador é  $y_k^{(i)}$ . Note-se que, até este instante, já foram recebidos  $i-1$  símbolos e que ainda serão recebidos  $L-i$  símbolos. Com isso, a seqüência completa  $\mathbf{y}_k$  pode ser dividida em três subsequências, uma representando o passado, outra o presente e outra o futuro:

$$\mathbf{y}_k = \left[ \underbrace{y_k^{(1)} y_k^{(2)} \dots y_k^{(i-1)}}_{y_k^{(<i)}} \underbrace{y_k^{(i)}}_{y_k^{(=i)}} \underbrace{y_k^{(1)} y_k^{(2)} \dots y_k^{(i-1)}}_{y_k^{(>i)}} \right] \quad (7)$$

$$= \left[ y_k^{(<i)} \quad y_k^{(=i)} \quad y_k^{(>i)} \right]$$

Com isso, a LLR *a posteriori* é dada por (BAHL et al., 1974):

$$L(b_k^{(i)} | \mathbf{y}_k) = \ln \frac{\sum_{R_1} P(s', s, \mathbf{y}_k)}{\sum_{R_0} P(s', s, \mathbf{y}_k)}$$

$$= \ln \frac{\sum_{R_1} \alpha_{i-1}(s') \gamma_i(s', s) \beta_i(s)}{\sum_{R_0} \alpha_{i-1}(s') \gamma_i(s', s) \beta_i(s)} \quad (8)$$

sendo  $P(s, s', \mathbf{y}_k)$  representa a probabilidade conjunta de estar no estado  $s'$  no instante  $i-1$ , estar no estado  $s$  no instante corrente  $i$ , e de a seqüência de  $L$  bits recebida ser  $\mathbf{y}_k$ . No numerador,  $R_1$  significa que o somatório se estende às transições entre estados  $s'$  e  $s$  provocadas por um bit  $b_k^{(i)} = +1$  (ramos tracejados na figura 1). Da mesma maneira, no denominador,  $R_0$  designa os outros ramos, originados por um bit  $b_k^{(i)} = -1$ . As variáveis  $\alpha$ ,  $\gamma$  e  $\beta$  são probabilidades definidas a seguir.

#### Cálculo da probabilidade conjunta

A probabilidade conjunta  $P(s, s', \mathbf{y}_k)$  pode ser obtida pelo produto de outras três probabilidades, da forma:

$$P(s', s, \mathbf{y}_k) = \alpha_{i-1}(s') \gamma_i(s', s) \beta_i(s) \quad (9)$$

onde as probabilidades  $\alpha$ ,  $\gamma$  e  $\beta$  são definidas como:

$$\alpha_{i-1}(s') = P(s', \mathbf{y}_k^{(<i)}) \quad (10)$$

$$\gamma_i(s', s) = P(\mathbf{y}_k^{(i)}, s' | s) \quad (11)$$

$$\beta_i(s) = P(\mathbf{y}_k^{(>i)} | s) \quad (12)$$

No instante  $i$  as probabilidades  $\alpha$ ,  $\gamma$  e  $\beta$  estão associadas ao passado, ao presente e ao futuro da seqüência  $\mathbf{y}_k$ , respectivamente.

#### Cálculo de $\gamma$

A probabilidade  $\gamma_i(s', s)$  é a probabilidade condicional de receber o símbolo  $y_k^{(i)}$  no instante  $i$  e o estado atual ser  $S_i = s$ , dado que o estado anterior foi  $S_{i-1} = s'$ . Para canais AWGN, essa probabilidade é composta pelo produto de probabilidades:

$$\begin{aligned} \gamma_i(s', s) &= P(y_k^{(i)} | x_k^{(i)})P(b_k^{(i)}) \\ &= C_i \exp(b_k^{(i)} L(b_k^{(i)})/2) \exp\left(\frac{L_c}{2} \sum_{l=1}^n x_{k,l}^{(i)} y_{k,l}^{(i)}\right) \end{aligned} \quad (13)$$

sendo  $C_i$  é irrelevante no cálculo da confiabilidade da estimativa, pois aparece no numerador e no denominador da equação (8), sendo cancelado ao se calcular a LLR condicional  $L_k^{(i)}(y_k)$ . Os termos  $x_{k,l}^{(i)}$  e  $y_{k,l}^{(i)}$  representam o bit  $l$  no instante  $i$  da seqüência à saída do codificador e à entrada do decodificador, respectivamente. O termo  $L_c$  é denominado valor ou medida de confiabilidade do canal, sendo descrita na literatura como (WOODARD; HANZO, 2000; SOLEYMANI; GAO; VILAIPORENSAWAI, 2002):

$$L_c = 4\Psi \frac{E_c}{N_0} = 4\Psi R_c \frac{E_b}{N_0} \quad (14)$$

sendo  $E_c$  e  $E_b$  são as energias transmitidas por bit codificado e bit de informação, respectivamente,  $R_c$  é a taxa do código e  $\Psi$  é a amplitude de desvanecimento. No caso de canal puramente AWGN,  $\Psi = 1$ .

#### Medida da confiabilidade do canal

Resultados de desempenho via método MCS mostraram que a equação (14) resulta em bons desempenhos para a estratégia turbo apenas nas regiões de baixo e médio  $E_b/N_0$ . Para valores de médio  $E_b/N_0 > 14$  dB, o desempenho é degradado consideravelmente, tornando sua utilização inviável. Baseado na técnica de ajuste de curvas (*fitting*) de desempenho MCS, neste trabalho foi obtida, de forma não-exaustiva, uma nova expressão para a medida da confiabilidade do canal com desvanecimentos Rayleigh lentos e seletivos em frequência:

$$L_c = \frac{-16,6785}{1 + \exp\left(\frac{\frac{E_b}{N_0} (dB) - 8,8324}{2,7642}\right)} + 17,622 \quad (15)$$

Essa expressão permite a obtenção de desempenhos apreciáveis para o sistema com codificação turbo, tanto nas faixas de baixo e médio  $E_b/N_0$  (similares aos reportados na literatura) quanto na região de alto  $E_b/N_0$ . No caso da região de alta  $E_b/N_0$ , desempenhos MCS mostraram-se superiores aos relatados na literatura. Um resultado ilustrativo de desempenho, obtido via simulação MCS, comparando a eficácia da utilização da medida de confiabilidade do canal, expressa pelas equações (14) versus (15), é apresentado na seção “Resultados numéricos”, figura 4.

Ressalta-se que existem poucos estudos e resultados na literatura considerando a equação da confiabilidade do canal para alta região de  $E_b/N_0$ . Neste trabalho, considerou-se a estratégia indireta de obtenção de uma expressão fechada para a confiabilidade do canal a partir dos resultados de desempenho do sistema em termos de taxa de erro de bit.

O algoritmo MAP é ótimo para a decodificação de códigos turbo, mas é extremamente complexo. O algoritmo Log-MAP é uma simplificação do algoritmo MAP, mas, em função da mudança das operações para o domínio logarítmico, no qual as multiplicações são substituídas por adições, os problemas numéricos inerentes ao algoritmo MAP são contornados, tendo assim a complexidade reduzida. O algoritmo Max-Log-MAP resulta em redução adicional de complexidade em relação ao algoritmo Log-MAP, porém o algoritmo Max-Log-MAP tem um desempenho ligeiramente inferior quando comparado aos algoritmos do MAP e do Log-MAP.

$$\alpha_i(s) = \sum_s \gamma_i(s', s) \alpha_{i-1}(s') \quad (16)$$

$$\beta_{i-1}(s') = \sum_s \gamma_i(s', s) \beta_i(s) \quad (17)$$

Sendo as condições iniciais são:

$$\alpha_0(s) = \begin{cases} 1 & s = 0 \\ 0 & s \neq 0 \end{cases}$$

$$\beta_I(s) = \begin{cases} 1 & s = 0 \\ 0 & s \neq 0 \end{cases}$$

Note-se que para o cômputo das probabilidades  $\alpha$  e  $\beta$  necessita-se conhecer a probabilidade  $\gamma_i(s', s)$ . Com isso, inicia-se o cálculo da LLR *a posteriori* pelo cômputo de  $\gamma_i(s', s)$ .

No caso de  $\alpha_i(s)$ , os somatórios são efetuados para todos os estados anteriores  $S_i = s'$  dos quais saem ramos que convergem no estado  $s$ , enquanto que no caso de  $\beta_{i-1}(s')$  os somatórios são efetuados para todos os estados seguintes  $S_i = s$ , os quais se atingem do estado  $s'$ . Considerando códigos binários, os somatórios possuem apenas duas parcelas. A probabilidade  $\alpha$  é calculada à medida que se for recebendo a seqüência  $\mathbf{y}_k$ , ou seja, no cálculo de  $\alpha$  o sentido é do início para o fim da treliça (*forward*). A probabilidade  $\beta$  só pode ser calculada depois de ter sido recebido toda a seqüência  $\mathbf{y}_k$ , ou seja, no cálculo de  $\beta$  o sentido é do fim para o começo da treliça (*backward*). Os valores iniciais  $\alpha_0(s)$  e  $\beta_I(s)$  pressupõem que o percurso na treliça começa e termina num estado nulo<sup>3</sup>. Para isso é necessário acrescentar ao fim da mensagem ou ao início da próxima transmissão alguns bits nulos (*tail bits*) capazes de levar o codificador a produzir uma saída também nula.

<sup>3</sup> Diz-se que se trata de uma treliça terminada.

### Instabilidade numérica

Os problemas de instabilidade numérica associados ao algoritmo BCJR são devido a natureza iterativa de alguns cálculos pode conduzir a indesejáveis situações de “*underflow*” ou “*overflow*” que devem ser evitadas. Nesse sentido, as equações recursivas (16) e (17) devem ser previamente normalizadas pela soma de todos os  $\alpha$  e  $\beta$  em cada instante, respectivamente. Com isso, definem-se as probabilidades normalizadas como:

$$\alpha_i(s) = \frac{\alpha'_i(s)}{\sum_s \alpha'_i(s)} \quad (18)$$

$$\beta_{i-1}(s') = \frac{\beta'_{i-1}(s')}{\sum_s \beta'_{i-1}(s')} \quad (19)$$

sendo  $\alpha'_i(s)$  e  $\beta'_{i-1}(s')$  são as versões não normalizadas calculadas como as equações (16) e (17).

Do mesmo modo, depois de todos os  $2I$  produtos  $\alpha_{i-1}(s') \cdot \gamma_i(s', s) \cdot \beta_i(s)$  em todos os ramos da treliça terem sido calculados no instante  $i$  a sua soma,

$$\begin{aligned} \sum_{P_i} &= \sum_{R_0, R_1} \alpha_{i-1}(s') \gamma_i(s', s) \beta_i(s) \\ &= \sum_{R_0} \alpha_{i-1}(s') \gamma_i(s', s) \beta_i(s) + \\ &\quad + \sum_{R_1} \alpha_{i-1}(s') \gamma_i(s', s) \beta_i(s) \end{aligned}$$

irá normalizar  $P(s', s, \mathbf{y}_k)$ :

$$P_{\text{norm}}(s', s, \mathbf{y}_k) = \frac{P(s', s, \mathbf{y}_k)}{\sum_{P_i}}$$

Garante-se, assim, que as somas de todos os  $\alpha$  e  $\beta$  e  $P_{\text{norm}}(s', s, \mathbf{y}_k)$  são sempre iguais a 1 em cada instante  $i$ . Nota-se que nenhuma destas somas de normalização afeta o valor da LLR *a posteriori*.

$$L(b_k^{(i)} | \mathbf{y}_k) = \ln \frac{\sum_{R_1} P(s', s, \mathbf{y}_k)}{\sum_{R_0} P(s', s, \mathbf{y}_k)} = \ln \frac{\sum_{R_1} P_{\text{norm}}(s', s, \mathbf{y}_k)}{\sum_{R_0} P_{\text{norm}}(s', s, \mathbf{y}_k)} \quad (20)$$

*Decodificação turbo*

A aplicação do algoritmo BCJR à decodificação iterativa resulta na estratégia conhecida como decodificação turbo. Considerando uma seqüência codificada com taxa  $1/n$ , no qual o primeiro bit codificado,  $x_k^{(i)}$ , é igual ao bit de informação  $b_{k,l}^{(i)}$ , a LLR *a posteriori* pode ser decomposta numa soma de três parcelas (BERROU; GLAVIEUX; THITIMAJSHIMA, 1993):

$$L(b_k^{(i)} | \mathbf{y}_k) = L(b_k^{(i)}) + L_C y_{k,1}^{(i)} + L_e(b_k^{(i)}) \quad (21)$$

As primeiras duas parcelas estão relacionadas com o bit de informação  $x_k^{(i)}$ . No entanto, a terceira,  $L_e(b_k^{(i)})$ , depende somente dos bits de paridade da palavra de código, denominada de informação extrínseca. Pode-se dizer que  $L_e(b_k^{(i)})$  é uma estimativa da LLR *a priori*  $L(b_k^{(i)})$ , pois caso seja fornecido os valores de  $L(b_k^{(i)})$  e  $L_C y_{k,1}^{(i)}$ , a um decodificador MAP (ou outro), obtém-se  $L(b_k^{(i)} | \mathbf{y}_k)$  na sua saída. Por subtração, a estimativa de  $L(b_k^{(i)})$ , pode ser obtida por:

$$L_e(b_k^{(i)}) = L(b_k^{(i)} | \mathbf{y}_k) - L(b_k^{(i)}) - L_C y_{k,1}^{(i)} \quad (22)$$

Esta estimativa de  $L(b_k^{(i)})$  é, presumivelmente, um valor mais preciso da LLR *a priori*, devendo substituir o valor anterior de  $L(b_k^{(i)})$ . Caso esse procedimento seja repetido de um modo iterativo irá fornecer a um outro decodificador os valores de  $L_C y_{k,l}^{(i)}$  e a nova  $L(b_k^{(i)}) = L_e(b_k^{(i)})$ , espera-se obter uma  $L(b_k^{(i)} | \mathbf{y}_k)$  mais confiável a cada nova iteração. A utilização dessa estratégia diversas vezes (utilizando a equação (21) caracteriza a decodificação turbo.

Os inventores dos códigos turbo (BERROU; GLAVIEUX; THITIMAJSHIMA, 1993) usaram dois códigos convolucionais recursivos e sistemáticos (RSC), de taxa 1/2, concatenados em paralelo e entrelaçados, e realizaram a decodificação iterativamente com dois decodificadores MAP, figuras 2 e 3, onde  $\mathbf{P}$  e  $\mathbf{P}^{-1}$  representam o entrelaçador e o desentrelaçador, respectivamente.

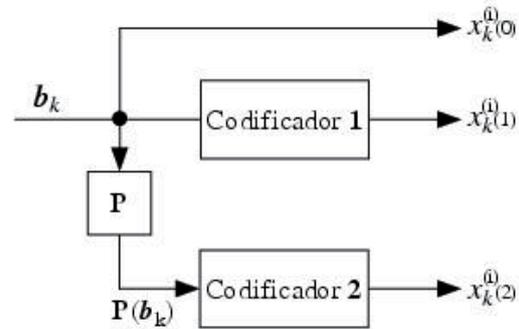


Figura 2. O codificador turbo.

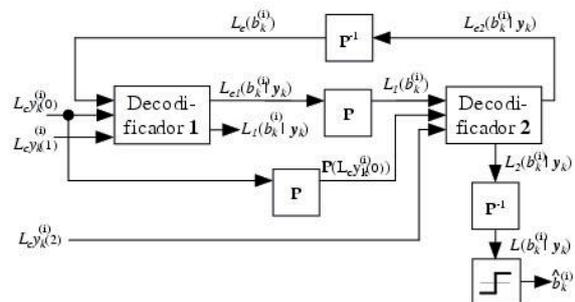


Figura 3. O decodificador turbo.

Considerando bits transmitidos equiprováveis, na primeira iteração, a LLR *a priori*  $L(b_k^{(i)})$  é nula. A informação extrínseca  $L_e(b_k^{(i)})$  que cada decodificador

fornece será usada para atualizar a  $L(b_k^{(i)})$  de iteração para iteração e desse decodificador para o outro. Dessa maneira o decodificador turbo ganha progressivamente mais confiança nas decisões que terá de tomar no fim do processo iterativo.

### Resultados numéricos

Em todas as simulações Monte Carlo, foram adotados os seguinte parâmetros de sistema: número mínimo de erros/ponto = 10; seqüências de espalhamento aleatórias com ganho de processamento  $N = 31$ ; canal com dois percursos e desvanecimento lento seguindo a distribuição Rayleigh com atrasos aleatórios ordenados e uniformemente distribuídos no intervalo  $[0; N-1]T_c$  e perfil atraso-potência com  $E[\psi_{k,1}^2] = 0,652$  e  $E[\psi_{k,2}^2] = 0,348$ , para qualquer  $k$  e carregamento variável com o número de usuários ativos  $K = [3, 7, 12, 17, 22]$ , dependendo do sistema analisado, resultando em um carregamento,  $L_N = K/N$ .

Para efeito de comparação, foi incluído o desempenho Monte Carlo para o limite analítico quando não há codificação e nem MAI no sistema (SuB – *Single User Bound*) (PROAKIS, 1995):

$$BER_{SuB} = \frac{1}{2} \sum_{\ell=1}^D \left\{ \left[ 1 - \sqrt{\frac{\bar{\gamma}_\ell}{\bar{\gamma}_\ell + 1}} \right] \prod_{i, i \neq \ell}^D \frac{\bar{\gamma}_\ell}{\bar{\gamma}_\ell + \bar{\gamma}_i} \right\} \quad (23)$$

sendo  $\gamma$  é definido pelo valor de  $E_b/N_0$  médio à entrada do receptor:

$$\bar{\gamma} = \sum_{\ell=1}^L \bar{\gamma}_\ell, \quad \text{onde} \quad \gamma_\ell = \frac{E_b}{N_0} E[\Psi_\ell^2] \quad (24)$$

sendo  $D$  é a diversidade Rake (número de ramos no receptor). Neste trabalho, adotou-se  $D = L$  e  $\gamma_\ell$  o valor médio quadrático do  $\ell$ -ésimo sinal recebido. Utilizou-se a relação sinal-ruído média no receptor (SNR – *Signal-to-Noise Ratio*),  $\bar{\gamma} = \sum_{\ell=1}^D \bar{\gamma}_\ell$  no intervalo  $\gamma = [0; 16]$  dB para decodificação Turbo e  $\gamma = [0; 14]$  dB para decodificador de Viterbi.

A cada realização MCS, assume-se que todos os  $K$  usuários apresentem velocidades constantes e uniformemente distribuídas no intervalo  $[0; v_{\max}]$ , resultando em uma frequência Doppler máxima de  $f_D = v_{\max} / \lambda_c = 222,2$  Hz para uma frequência da portadora  $f_c = 1 / \lambda_c = 2$  GHz.

A Tabela 1 sintetiza os principais parâmetros do sistema adotados nas simulações MCS.

**Tabela 1.** Principais Parâmetros do Sistema.

Sistema	K	$L_N$ (%)	$E[\psi_{k,1}^2]$	$E[\psi_{k,2}^2]$	$\bar{\gamma}$ (dB)	
					Conv.	Turbo
S1	3	6,68	0,652	0,348	[0;14]	[0;16]
S2	7	22,58	0,652	0,348	[0;14]	[0;16]
S3	12	38,71	0,652	0,348	[0;14]	[0;16]
S4	17	54,84	0,652	0,348	[0;14]	[0;16]
S5	22	70,97	0,652	0,348	[0;14]	[0;16]

Na especificação do CDMA2000 (TIA/EIA/IS-2002, 1999), o entrelaçador é cuidadosamente projetado e assim, visa-se evitar o mapeamento de bits em posições vizinhas dentro da mesma janela de processamento. As linhas são embaralhadas de acordo com uma regra, e os elementos dentro de cada linha são permutados de acordo com uma seqüência linear específica. Por fim, os endereços de saída são lidos por coluna. No entanto, neste trabalho, utilizou-se uma estrutura simplificada para o entrelaçador. O entrelaçador utilizado realiza a permutação dos bits de forma aleatória, sendo as posições sorteadas a cada quadro de  $I$  bits transmitidos.

Além desses parâmetros do sistema, adotou-se a matriz geradora dos códigos turbo da forma:

$$G(E) = \left[ 1, \frac{g_0(E)}{d(E)}, \frac{g_1(E)}{d(E)} \right] \quad (25)$$

$$= \left[ 1, \frac{1+E+E^3}{1+E^2+E^3}, \frac{1+E+E^2+E^3}{1+E^2+E^3} \right]$$

Ou ainda  $[1, 15/13, 17/13]_{\text{octal}}$ , conforme apresentado na Tabela 2.

Uma forma generalizada da codificação não recursiva, utilizada para o algoritmo de Viterbi, é dada por (SOLEINMANI; GAO; VILAIPORENSAWAI, 2002):

$$G(E) = [g_0(E), g_1(E), \dots, g_{n-1}(E)] \quad (26)$$

Para a codificação convolucional, foi adotado os polinômios geradores, dado em octal, conforme a Tabela 3.

Analisou-se o desempenho dos algoritmos de Viterbi e turbo para três taxas de codificação,  $\frac{1}{2}$ ,  $\frac{1}{3}$  e  $\frac{1}{4}$ , acarretando em uma taxa básica de transmissão e tamanho dos pacotes que podem ser observados na tabela 4.

**Tabela 2.** Polinômios da Codificação turbo.

Taxa (1/n)	Registradores (Q)	Polinômio (octal)		
		d	g <sub>0</sub>	g <sub>1</sub>
$\frac{1}{2}, \frac{1}{3}$ e $\frac{1}{4}$	3	13	15	17

**Tabela 3.** Polinômios da Codificação Convolucional.

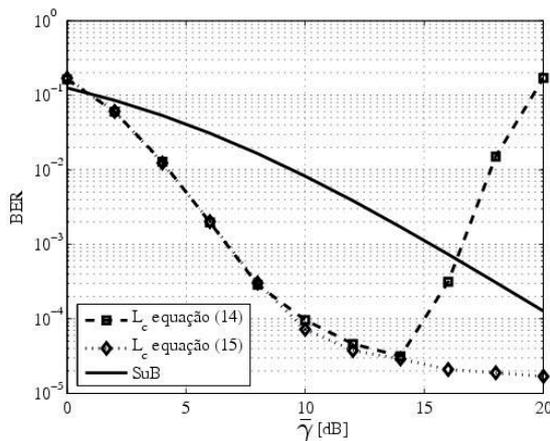
Taxa (1/n)	Registradores (Q)	Polinômio (octal)			
		g <sub>0</sub>	g <sub>1</sub>	g <sub>2</sub>	g <sub>3</sub>
$\frac{1}{2}$	8	753	561	--	--
$\frac{1}{3}$	8	557	663	711	--
$\frac{1}{4}$	8	765	671	513	473

**Tabela 4.** Parâmetros dos algoritmos de Viterbi e Turbo.

Cenário	(1/n)	$R_n$ (bps)
1	$\frac{1}{2}$	14400
2	$\frac{1}{3}$	9600
3	$\frac{1}{4}$	6800

Bits Transmitidos ( $I$ )	
Viterbi	Turbo
280	282
184	186
128	130

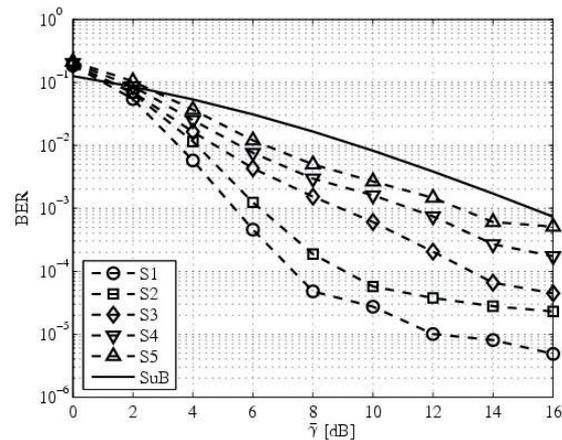
Uma comparação entre as equações (14) e (15) é apresentada na figura 4. A equação (15) apresenta desempenho similar quando da utilização da equação (14) para as faixas de baixo e médio  $\gamma$  e um melhor desempenho na faixa de alto  $\gamma$ , tornando-se uma melhor opção para a determinação da confiabilidade do canal.



**Figura 4.** Comparação das equações (14) e (15) para  $K = 5$  usuários e cenário 2 e SuB, equação (23).

A figura 5 apresenta o desempenho do decodificador turbo com 5 iterações considerando diferentes índices de carregamento (sistemas 1, 2, 3, 4 e 5) e cenário 2. Note-se que o desempenho da estratégia turbo mostra-se melhor que o SuB, considerando modulação BPSK e ausência de codificação, para toda a faixa de  $\gamma$  e número de usuários simulado, essa estratégia mostra ser capaz de garantir uma baixa BER, mesmo para pequenos valores de  $\gamma$  quando em canal multipercursos.

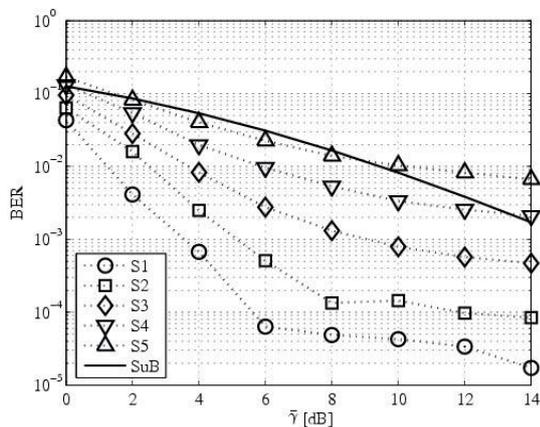
Observa-se que para garantir uma  $BER < 10^{-4}$  e considerando  $K < 12$  usuários, é necessário apenas uma relação de  $\gamma < 16$  dB. No entanto, quando se aumenta o carregamento o desempenho é degradado proporcionalmente, mostrando que a estratégia turbo é sensível ao aumento do número de usuários no sistema. Quando o carregamento aumentar, deve-se utilizar de forma combinada uma maior taxa de codificação, uma melhor estratégia para o entrelaçador e desentrelaçador e/ou um maior número de iterações, visando a atingir a BER desejada.



**Figura 5.** Desempenho para decodificação turbo em cenário 2 e  $N_{ITER} = 5$ .

A figura 6 apresenta o desempenho do decodificador de Viterbi considerando diferentes índices de carregamento (sistemas 1, 2, 3, 4 e 5) e cenário 2. Note-se que o desempenho do decodificador de Viterbi também é melhor que o

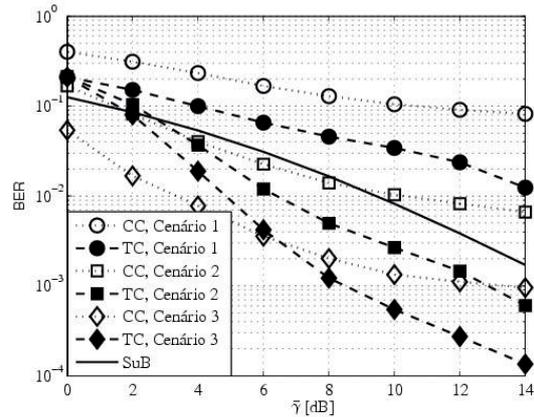
SuB, considerando modulação BPSK e ausência de codificação, para a faixa de  $\gamma < 12$  dB e número de usuários simulado, garantindo também uma baixa BER, mesmo para pequenos valores de  $\gamma$  quando em canal multipercuso. No entanto, quando se aumenta o carregamento o desempenho é degradado proporcionalmente, mostrando-se mais sensível ao aumento de carregamento do que o decodificador turbo quando na condição do cenário 2 (taxa =  $1/n = 1/3$ ).



**Figura 6.** Desempenho para o decodificador de Viterbi em cenário 2.

A figura 7 apresenta o desempenho dos decodificadores de Viterbi e turbo considerando diferentes taxas de codificação (cenários 1, 2 e 3) e carregamento do sistema S5 ( $K = 22$  usuários). Quando se compara o decodificador turbo com o de Viterbi observa-se que para taxa de codificação pequena,  $1/2$  (cenário 1), o decodificador turbo apresenta melhor desempenho para toda a faixa de  $\gamma$  simulada, para taxa de codificação de  $1/3$  (cenário 2) para a faixa de  $\gamma > 4$  dB e para taxa de codificação de  $1/4$  (cenário 3) para a faixa de  $\gamma > 6$  dB. Esse fato mostra que, à medida que a taxa de codificação aumenta, o desempenho do decodificador turbo torna-se melhor do que o obtido pelo decodificador de Viterbi. A medida que a faixa de  $\gamma$  aumenta e que, a decodificação turbo para a taxa de codificação de  $1/3$  (cenário 2) tem desempenho superior à

decodificação de Viterbi. O mesmo se verifica para taxa de codificação de  $1/4$  (cenário 3) para a faixa de  $\gamma > 13$  dB. O mesmo ocorre para sistemas S4 ( $K = 17$  usuários) e S3 ( $K = 12$  usuários).



**Figura 7.** Desempenho para decodificação turbo e Viterbi considerando carregamento do sistema 5 para diferentes cenários.

## Complexidade computacional

No intuito de expressar e comparar a complexidade das estratégias de decodificação turbo e Viterbi, considerou-se a análise da ordem do número de transições na treliça computadas e o requisito de alocação de memória. Nessa análise, considerou-se apenas a complexidade das estratégias de decodificação turbo e de Viterbi, ou seja, a complexidade associada ao modulador e ao demodulador (Rake) não foram consideradas, pois tornam-se insignificantes quando comparadas com a complexidade das estratégias de decodificação.

Note-se que a complexidade dos decodificadores dependem diretamente do número de transições na treliça por bit de informação. Para o codificador convolucional binário, existem apenas duas transições possíveis para cada um dos  $2^Q$  estados. Com isso, pode-se aproximar a complexidade do decodificador de Viterbi como (LIEW; HANZO, 2001):

$$O\{CC(n, Q)\} = nK2^Q \quad (27)$$

Considera-se que o número de transições na treliça no algoritmo de decodificação Log-MAP é cerca de três vezes maior que a do algoritmo de Viterbi, pois percorre-se a treliça na direção direta (*forward*) e na direção reversa (*backward*) além de realizar o cálculo logarítmico de saída suave (*soft output*). Para códigos turbo, é utilizada uma estratégia de decodificação que utiliza dois decodificadores (Figura 3), com isso, a complexidade computacional da decodificação turbo, utilizando-se do algoritmo Log-MAP, pode ser aproximada por (LIEW; HANZO, 2001):

$$O\{TC(n, Q)\} = 6N_{\text{iter}}nK2^Q \quad (28)$$

Para a análise do requisito de memória utilizada pelos decodificadores deve-se considerar o número de estados da treliça por bloco codificado. Para o codificador convolucional binário, observações realizadas para o algoritmo de Viterbi mostraram que todos os percursos sobreviventes dos estados correntes da treliça surgem, no máximo, dos  $5 \times (Q+1)$  estados anteriores da treliça. Com isso, a cada instante da decodificação, apenas  $5 \times (Q+1)$  transições são armazenadas na memória. Portanto, para o algoritmo de Viterbi, o requisito de memória é dado por (LIEW; HANZO, 2001):

$$Mem\{CC(n, Q)\} = 5(Q+1)k2^Q \quad (29)$$

O algoritmo Log-MAP requer armazenar as variáveis  $\gamma$ ,  $\alpha$  e  $\beta$ . O requisito de memória para o algoritmo Log-MAP é também cerca de três vezes o requisito do algoritmo de Viterbi. Consequentemente, o requisito de memória da estratégia turbo é de (LIEW; HANZO, 2001):

$$Mem\{TC(n, Q)\} = 3IK2^Q \quad (30)$$

Utilizando os valores dos parâmetros adotados para obtenção dos resultados de simulação apresentados, pode-se determinar o custo computacional dos algoritmos de Viterbi e turbo. A tabela 5 apresenta esses resultados considerando sistema S3.

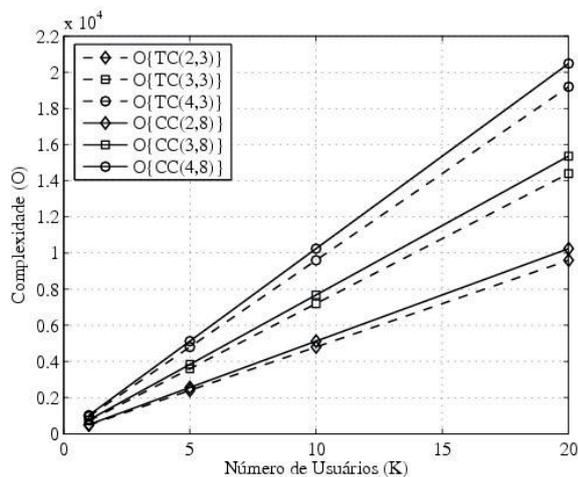
Note-se, na tabela 5, que o decodificador turbo apresenta uma menor complexidade em termos do número de transições na treliça por bit transmitido e também um menor requisito de memória quando comparado ao decodificador de Viterbi. Por meio das equações (27) e (28) e considerando o número de registradores  $Q = 8$  e  $Q = 3$  para os decodificadores de Viterbi e turbo, respectivamente, obtém-se uma relação indicando que ao se utilizar um número de iterações  $N_{\text{iter}} \leq 5$  para a estratégia turbo, esta apresentará uma menor complexidade. Além disso, deve-se considerar que o requisito de memória da estratégia turbo diminui com o aumento da taxa de codificação, enquanto para a decodificação convolucional, este requisito mantém-se constante, independentemente da taxa.

**Tabela 5.** Caracterização da Complexidade (O) e Requisito de Memória (Mem) dos Diferentes Decodificadores de Canal.

(n,Q)	(2 <sup>Q</sup> )	Decodif.	N <sub>iter</sub>	K	I	O	Mem
Código Convolutacional (CC – Convolutional Code)							
CC(2,8)	256	VA	--	10	280	5120	115200
CC(3,8)	256	VA	--	10	184	7680	115200
CC(4,8)	256	VA	--	10	128	10240	115200
Código Turbo (TC – Turbo Code)							
TC(2,3)	8	Log-MAP	5	10	282	4800	67680
TC(3,3)	8	Log-MAP	5	10	186	7200	44640
TC(4,3)	8	Log-MAP	5	10	130	9600	31200

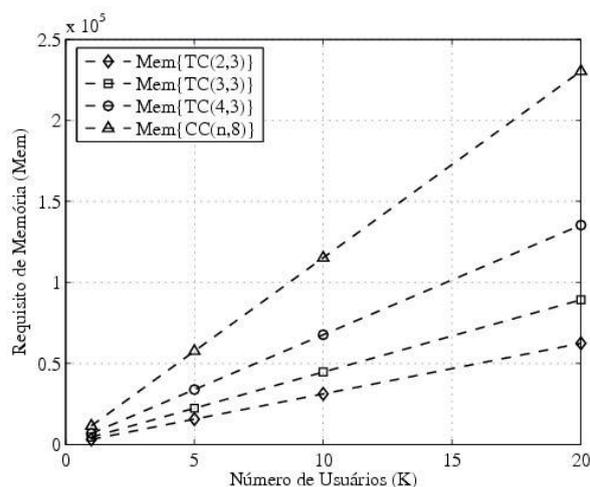
Visando comparar a complexidade e o requisito de memória com o aumento do número de usuários, utilizaram-se valores obtidos considerando os 4 sistemas simulados (Tabela 1).

A figura 8 indica que o aumento da complexidade é linear com o aumento do número de usuários e taxa de codificação para ambos os algoritmos de detecção. Essa afirmação pode ser verificada por meio das equações (27) e (28).

**Figura 8.** Complexidade em termos do número de transições na treliça por bit de informação.

A figura 9 também indica que o aumento do requisito de memória é linear com o aumento do

número de usuários e taxa de codificação para ambos os algoritmos de detecção. Além disso, observa-se que o requisito de memória decresce com o aumento da taxa de codificação para o decodificador turbo.

**Figura 9.** Requisito de memória em termos do número de estados da treliça por bloco codificado.

Considerando os resultados apresentados na tabela 5 e nas figuras 8 e 9, percebe-se que a estratégia turbo mostra-se mais eficiente, pois apresenta uma menor complexidade e menor requisito de memória em relação ao decodificador convolutacional para todos os sistemas simulados.

## Conclusões

Este trabalho faz uma revisão das estratégias de codificação e decodificação turbo e Viterbi aplicadas ao problema SuD em sistemas de múltiplo acesso DS/CDMA, sujeitos a canais de multipercurso com desvanecimento lento e seletivo em frequência.

Propôs-se uma nova equação baseada no ajuste de curva do tipo sigmóide para a medida de confiabilidade do canal multipercurso com desvanecimento lento e seletivo em frequência. Como resultado, obteve-se um melhor desempenho nas regiões de alta  $E_b/N_0$  em sistemas de múltiplo acesso DS/CDMA e detecção uniusuário (SuD), mostrando ser uma escolha mais apropriada para a medida de confiabilidade do canal, em comparação às equações sugeridas na literatura até o momento.

Compararam-se as estratégias de decodificação, considerando como métricas o desempenho (BER), a complexidade e o requisito de memória. Resultados de desempenho MCS, de complexidade em termos de transições treliças por bit de informação e requisitos de memória mostraram que a decodificação turbo é superior à convolucional (menor complexidade e menor requisito de memória), quando considerada a mesma taxa de codificação. Além disso, ela apresenta melhor desempenho (BER) quando a quantidade de bits de informação é maior, desde que o número de iterações não seja superior a 5 e para um número de registradores específicos para a (de)codificação convolucional do padrão IS-95 e da (de)codificação turbo do padrão do CDMA2000, respectivamente.

Assim, a estratégia de decodificação turbo mostra-se mais atrativa para a implementação na estação rádio-base de sistemas 3G e 4G, pois apresentou um melhor compromisso desempenho e complexidade, quando comparado ao decodificador de Viterbi.

## Referências

- (3GPP) 3rd Generation Partnership Project (3GPP). *Multiplexing and Channel Coding (FDD)*. 3GTS 25.212, v3.5.0, Dec. 2000. Especificação Técnica disponível em <http://www.3gpp.org/>
- BAHL, L. R.; COCKE, J.; JELINEK, F.; RAVIV, J. Optimal decoding of linear codes for minimizing symbol error rate. *IEEE Transactions on Information Theory*, New York, v. 20, n. 2, p. 284-287, Mar. 1974.
- BARBULESCU, A.; PIETROBON, S. S. Interleaver design for turbo codes. *IEEE Electronics Letters*, New York, v. 30, n. 25, p. 2107-2108, Dec.1994.
- BENEDETTO, S.; DIVSALAR, D.; MONTORSI, G.; POLLARA, F. Soft-output decoding algorithms in iterative decoding of turbo codes. *The Telecommunications and Data Acquisition Progress Report*, Pasadena, v. 124, n. 1, p. 63-87, Oct. 1995.
- BENEDETTO, S.; DIVSALAR, D.; MONTORSI, G.; POLLARA, F. A Soft-input soft-output maximum a posteriori (map) module to decode parallel and serial concatenated codes. *The Telecommunications and Data Acquisition Progress Report*, Pasadena, v. 127, n. 1, p. 1-20, Jul. 1996.
- BENEDETTO, S.; DIVSALAR, D.; MONTORSI, G.; POLLARA, F. Design of serially concatenated interleaved codes. *IEEE International Conference on Communications*, New York, v. 2, p. 710-714, Feb. 1997.
- BENEDETTO, S.; MONTORSI, G. Design of parallel concatenated convolutional codes. *IEEE Transactions on Communications*, New York, v. 44, n. 5, p. 591-600, May 1996.
- BENEDETTO, S.; MONTORSI, G. Unveiling turbo codes: some results on parallel concatenated coding schemes. *IEEE Transactions on Information Theory*, New York, v. 42, n. 2, p. 409-428, Mar. 1996.
- BERROU, C.; GLAVIEUX, A. Near optimum error correcting coding and decoding: turbo codes. *IEEE Transactions on Communications*, New York, v. 44, n. 10, p. 1261-1271, Oct. 1996.
- BERROU, C.; GLAVIEUX, A.; THITIMAJSHIMA, P. Near shannon limit error-correcting coding and decoding. *Proceedings of ICC'93*, Switzerland, v. 2, p. 1064-1070, May 1993.

- BURR, A. Turbo-codes: the ultimate error control codes? *Electronics & Communication Engineering Journal*, London, v. 13, n. 4, p. 155–165, Aug. 2001.
- ELIAS, P. Coding for noisy channels. *IRE Convention Record Part 4*, New York, v. 3, p.37-47, Mar. 1955.
- ERFANIAN, J. A.; PASUPATHY, S.; GULAK, G. Reduced complexity symbol detectors with parallel structures for isi channels. *IEEE Transactions on Communications*, New York, v. 42, n. 2/4, p. 1661-1671, Feb. 1994.
- FANO, R. M. A Heuristic discussion of probabilistic decoding. *IEEE Transactions on Information Theory*, New York, v. 9, n. 2, p. 64–74, Apr. 1963.
- FORNEY, G. D. The viterbi algorithm. *Proceedings of the IEEE*, New York, v. 61, n. 3, p. 268–278, Mar. 1973.
- GOFF, S. L.; GLAVIEUX, A.; BERROU, C. Turbo-Codes and High Spectral Efficiency Modulation. In: PROCEEDINGS OF IEEE INTERNATIONAL CONFERENCE ON COMMUNICATIONS (ICC 94), New Orleans, May 1994. v. 2, p. 645-649.
- HAGENAUER, J. Source-Controlled Channel Decoding. *IEEE Transactions on Communications*, New York, v. 43, n. 9, p. 2449–2457, Sep. 1995.
- HAGENAUER, J.; HOEHER, P. A Viterbi algorithm with soft-decision outputs and its applications. In: PROCEEDINGS OF IEEE GLOBAL TELECOMMUNICATIONS CONFERENCE (GLOBECOM'89), Dallas, 1989. p. 4711-4717.
- HAGENAUER, J.; OFFER, E.; PAPKE, L. Iterative decoding of binary block and convolutional codes. *IEEE Transactions on Information Theory*, New York, v. 42, n. 2, p. 429–445, Mar. 1996.
- HALL, E. K.; WILSON, S. G. Design and analysis of turbo codes on rayleigh fading channels. *IEEE Journal on Selected Areas in Communications*, New York, v. 16, n. 2, p. 160–174, Feb. 1998.
- HELLER, J.; JACOBS, I. Viterbi decoding for satellite and space communication. *IEEE Transactions on Communication Technology*, San Diego, v. 19, n. 5, p. 835–848, Oct. 1971.
- KOCH, W.; BAIER, A. Optimum and sub-optimum detection of coded data disturbed by time-varying inter-symbol interference. *IEEE GLOBECOM'90*, v. 3, p. 1679–1684, Dec. 1990.
- LEE, E. A.; MESSERSCHMITT, D. G. Digital communication: second edition. Howell: Kluwer Academic Publishers, 1994.
- LIEW, T. H.; HANZO, L. Space-time block codes and concatenated channel codes: a historical perspective and comparative study. In: PROCEEDINGS OF THE IEEE, New York, 2001.
- LIN, S.; COSTELLO, D. J. *Error control coding: fundamentals and applications*. New Jersey: Prentice-Hall, 1983.
- MASSEY, J. L. *Threshold decoding*. Massachusetts: Press Cambridge, 1963.
- NICKL, H.; HAGENAUER, J.; BURKETT, F. Approaching Shannon's capacity limit by 0.27 db using simple hamming codes. *IEEE Communications Letters*, New York, v. 1, n. 1, p. 130–132, Sep. 1997.
- PEREZ, L. C.; SEGHERS, J.; COSTELLO, D. J. A distance spectrum interpretation of turbo codes. *IEEE Transactions on Information Theory*, New York, v. 42, n. 6, p. 1698-1709, Nov. 1996.
- PROAKIS, J. G. *Digital communications*. Second. New York: McGraw-Hill, 1995.
- PROAKIS, J. G.; SALEHI, M. *Communication systems engineering*. New Jersey: Prentice-Hall, 1994.
- PYNDIAH, R. M. Near-optimum decoding of product codes: block turbo codes. *IEEE Transactions on Communications*, New York, v. 46, n. 8, p. 1003-1010, Aug. 1998.
- ROBERTSON, P.; VILLEBRUN, E.; HOEHER, P. A compararison of optimal and sub-optimal map decoding algorithms operating in the log domain. In: PROCEEDINGS INTERNATIONAL CONFERENCE COMMUNICATIONS (ICC'95), Seattle, 1995. v. 2, p. 1009-1013.
- ROBERTSON, P.; WORZ, T. Bandwidth-efficient turbo trellis-coded modulation using punctured component codes. *IEEE Journal on Selected Areas in Communications*, New York, v. 16, n. 2, p. 206–218, Feb. 1997.
- SHANNON, C. E. A mathematical theory of communication. *The Bell System Technical Journal*, Wiley, v. 27, n. 1/2, p. 379-423 e 623-656, Oct 1948.
- SHIN, H.; KIM, S.; LEE, J. H. Turbo decoding in a rayleigh fading channel with estimated channel state information. In: PROCEEDINGS OF VTC'2000, Boston, 2000. v. 3, p. 1358-1363.
- SKLAR, B. A primer on turbo codes concepts. *IEEE Communications Magazine*, New York, v. 35, n. 12, p. 94–102, Dec. 1997.
- SOLEYMANI, M. R.; GAO, Y.; VILAIPORENSAWAI, U. *Turbo coding for satellite and wireless communications*. Norwell: Kluwer Academic Publishers, 2002.

- TIA/EIA/IS-2000-2. *Physical layer standard for cdma2000 spread spectrum systems*. Arlington: Telecommunications Industry Association, 1999, 466p. Padrão de Especificação Técnica disponível em <http://www.tiaonline.org/standards/technology/cdma2000/documents/TIA-EIA-IS-2000-2-A-1.pdf>
- UNGERBOECK, G. Trellis-coded modulation with redundant signal sets part I: introduction. *IEEE Communications Magazine*, New York, v. 25, n. 2, p. 5-11, Feb. 1987a.
- \_\_\_\_\_. Trellis-coded modulation with redundant signal sets part II: state of the art. *IEEE Communications Magazine*, New York, v. 25, n. 2, p. 12-21, Feb. 1987b.
- VERDÚ, S. *Multiuser detection*. New York: Cambridge University Press, 1998.
- VITERBI, A. J. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, New York, v. 13, p. 260-269, Apr. 1967.
- WACHSMANN, U.; HUBER, J. Power and bandwidth efficient digital communications using turbo codes in multilevel codes. *European Transactions on Telecommunications (ETT)*, London, v. 6, p. 557-567, Sep/Oct. 1995.
- WICKER, S. B. *Error control systems for digital communications and storage*. First. Upper Saddle River: Englewood Cliffs: Prentice-Hall, 1995.
- WOODARD, J. P.; HANZO, L. Comparative study of turbo decoding techniques: an overview. *IEEE Transactions on Vehicular Technology*, New York, v. 49, n. 6, p. 2208-2233, Nov. 2000.
- WOZENCRAF, J. M. Sequential decoding for reliable communication. *IRE Conventional Record*, v. 5, n. 2, p. 11-25, 1957.
- WOZENCRAFT, J. M.; REIFFEN, B. *Sequence decoding*. Cambridge: MIT Press, 1961.

