

Técnica de decodificação híbrida iterativa para códigos LDPC

Walter Prado de Souza Guimarães e Valdemar C. da Rocha Júnior

Resumo—Um sistema de decodificação híbrida (DH) para códigos LDPC empregando a técnica min-sum *belief propagation* (BP) para a correção de erros, aliada à decodificação iterativa de apagamentos, é investigado por meio de simulação em computador. Empregando um número significativamente menor de iterações, o sistema DH se torna mais eficaz, especialmente na região de patamar de erros, e com desempenho e complexidade de implementação equivalentes ao da decodificação min-sum BP.

Palavras-Chave—Códigos LDPC, decodificação iterativa BP, decodificação de apagamentos, decodificação híbrida.

Abstract—A hybrid decoding system for LDPC codes employing min-sum belief propagation (BP) decoding to correct errors, combined with erasure iterative decoding, is investigated by means of computer simulation. The proposed hybrid decoding system by employing a reduced number of iterations becomes more efficient, especially in the error-floor region, achieving a similar performance with equivalent implementation complexity to that of min-sum BP decoding schemes.

Keywords—LDPC codes, BP iterative decoding, erasures decoding, hybrid decoding.

I. INTRODUÇÃO

O objetivo desse artigo é investigar o comportamento de um sistema de decodificação híbrida (DH) iterativa para códigos LDPC (*low density parity check*) [1], visando reduzir o número de iterações e manter desempenho equivalente ao alcançado com sistemas de decodificação iterativa do tipo soma-produto [2]. O sistema DH proposto consiste de dois estágios, sendo o primeiro deles a decodificação iterativa de erros do tipo min-sum *belief propagation* (BP) em canal com ruído aditivo Gaussiano branco (AWGN - *additive white Gaussian noise*), e o segundo estágio consistindo da decodificação iterativa em canal binário com apagamento (BEC - *binary erasure channel*) artificialmente criado, interagindo com o decodificador min-sum BP do primeiro estágio. Uma iteração no sistema DH inclui necessariamente uma passagem pelo algoritmo min-sum BP e, caso necessário, inclui também uma passagem pelo algoritmo de decodificação de apagamentos. A eficiência dessa abordagem é ilustrada através de simulações computacionais, comparando o sistema DH com o método de decodificação min-sum BP.

Nos esquemas de decodificação híbrida conhecidos na literatura, desenvolvidos para uso comercial ou científico, são considerados erros aleatórios ou em surtos, empregando decodificação com decisão suave associada à decodificação

com decisão abrupta [3]. O artifício da correção de apagamentos, aplicado após uma falha na decodificação de erros, tem recebido pouca ou nenhuma atenção tanto em canais com AWGN como em outros modelos de canal. Em [4] é proposta uma técnica de pós-processamento baseada em correção de apagamentos que é ativada após a ocorrência de falha na decodificação min-sum BP e sob certas condições de peso da síndrome. Diferentemente, neste artigo, é proposto um método que atua sempre que houver a falha da decodificação min-sum BP, sendo que os apagamentos são definidos com base nas informações de confiabilidade dos dígitos, geradas ao fim da decodificação min-sum BP. Adicionalmente, emprega-se um número fixo e pre-determinado de apagamentos a serem processados pela decodificação iterativa de apagamentos.

O restante deste artigo está organizado como descrito a seguir. Na Seção II é descrito o sistema DH proposto. Na Seção III são feitas considerações sobre o canal BEC artificial para lidar com os erros remanescentes, após a decodificação min-sum BP sem sucesso. Na Seção IV é descrita uma forma de determinação do valor adequado para o número de apagamentos artificialmente criados e o artigo é finalizado na Seção V com a apresentação dos resultados de simulação em computador e alguns comentários.

II. SISTEMA COM DECODIFICAÇÃO HÍBRIDA

No transmissor, a informação binária $\mathbf{u} = (u_1, u_2, \dots, u_K)$ é usada para gerar a palavra-código $\mathbf{c} = (c_1, c_2, \dots, c_N)$ pertencente a \mathcal{C} , que denota um código binário LDPC. Cada palavra-código é então modulada antes de ser transmitida pelo canal, usando-se um esquema de modulação digital, por exemplo BPSK, que dá origem à N -upla $\mathbf{x} = (x_1, x_2, \dots, x_N)$ de números reais correspondentes. A N -upla \mathbf{x} é enviada por um canal com AWGN cuja saída \mathbf{y} é expressa como $\mathbf{y} = \mathbf{x} + \mathbf{n}$, em que $\mathbf{n} = (n_1, n_2, \dots, n_N)$ denota um vetor cujas componentes são amostras de ruído branco Gaussiano introduzido pelo canal, com média nula e variância σ^2 . A decodificação consiste em realizar o processamento de \mathbf{y} para recuperar \mathbf{u} .

O decodificador recebe do canal a N -upla \mathbf{y} e a converte em uma sequência de valores de razão logarítmica de verossimilhança quantizada (QLLR - *quantized log-likelihood ratio*) [5], representada por $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_N)$.

Para o primeiro estágio de decodificação há a necessidade de especificar o número I_{BP} de iterações do algoritmo min-sum BP. A escolha do valor de I_{BP} deve ser tal que, caso a palavra recebida não seja decodificada em até I_{BP} iterações, provavelmente o decodificador min-sum BP encontrou um conjunto de armadilha (*trapping set*) [6]. Ao atingir um conjunto de armadilha, não é possível por meio de mais iterações

do algoritmo min-sum BP sair desta situação e conseguir decodificar com sucesso a palavra recebida.

Após cada iteração do algoritmo min-sum BP é gerada uma N -upla $\hat{\xi} = (\hat{\xi}_1, \hat{\xi}_2, \dots, \hat{\xi}_N)$ de valores de confiabilidade, a qual é convertida para a forma binária $\hat{c} = (\hat{c}_1, \hat{c}_2, \dots, \hat{c}_N)$ por decisão abrupta. A N -upla binária \hat{c} é submetida a um teste de síndrome por meio da operação $\hat{c}\mathbf{H}^T$, em que \mathbf{H}^T denota a matriz de verificação de paridade transposta. Caso ocorra $\hat{c}\mathbf{H}^T = \mathbf{0}$, \hat{c} é uma palavra código e o ciclo de DH é interrompido com sucesso. Caso contrário, uma nova iteração do algoritmo min-sum BP é realizada e o teste sobre \hat{c} é aplicado, até verificar-se $\hat{c}\mathbf{H}^T = \mathbf{0}$ ou serem atingidas I_{BP} iterações. Quando o número de iterações atinge o valor I_{BP} , sem que haja sucesso na decodificação, ao invés de ser declarada a falha de decodificação, $\hat{\xi}$ é convenientemente processada, como explicado a seguir, e convertida numa N -upla $\mathbf{z} = (z_1, z_2, \dots, z_N)$ com dígitos binários e apagamentos, a qual é então submetida à decodificação iterativa de apagamentos.

O segundo estágio de decodificação opera sobre a saída de um canal BEC artificialmente criado, somente quando há falha na decodificação no primeiro estágio. A motivação para o emprego da decodificação de apagamentos no segundo estágio de decodificação é a seguinte. É sabido que, para um determinado código de bloco com distância mínima d , é possível a correção de qualquer padrão com até $d-1$ apagamentos [7] e de uma grande quantidade de padrões contendo d ou mais apagamentos, desde que o total de apagamentos por palavra não exceda o número de dígitos de verificação de paridade do código [8].

Uma iteração da decodificação de apagamentos consiste em examinar cada equação de verificação de paridade, com relação ao número de apagamentos. Caso uma equação tenha um único apagamento entre as posições verificadas, este é corrigido. Caso uma equação não contenha apagamentos ou contenha dois ou mais apagamentos, o decodificador a ignora. Caso um ou mais apagamentos tenham sido corrigidos, é então repetido o processo de examinar todas as equações de verificação de paridade. Se ao completar a verificação de todas as equações nenhum apagamento tiver sido corrigido, a decodificação de apagamentos é encerrada [9].

A sequência binária na entrada do canal BEC consiste de 0's e 1's, obtidos por decisão abrupta tomada sobre as respectivas medidas de confiabilidade, obtidas a partir das probabilidades a posteriori na última iteração da decodificação min-sum BP, utilizando como medida a QLLR. Em caso de falha na decodificação min-sum BP, o maior percentual dos erros remanescentes está concentrado em valores de QLLR próximos a 0 [10].

Definição 1: $\mathbb{S}_{\mathcal{X}}$ denota o conjunto de cardinalidade \mathcal{X} , um número inteiro positivo, que contém as coordenadas dos \mathcal{X} dígitos de menor confiabilidade da entrada do canal BEC, os quais são escolhidos para apagamentos.

Desta forma, os dígitos associados às posições em $\mathbb{S}_{\mathcal{X}}$ correspondem àqueles com os menores valores de QLLR em módulo. A N -upla \mathbf{z} na saída do canal BEC, contendo \mathcal{X} dígitos apagados, tem seus componentes especificados do

seguinte modo:

$$z_i = \begin{cases} \Delta, & \text{para } i \in \mathbb{S}_{\mathcal{X}} \\ \hat{c}_i, & \text{para } i \notin \mathbb{S}_{\mathcal{X}}, \end{cases}$$

em que Δ representa um apagamento.

Caso nenhum apagamento tenha sido corrigido na primeira iteração o algoritmo DH é encerrado, com falha na decodificação. Entretanto, se pelo menos um apagamento tiver sido corrigido, ao ser encerrada a etapa de decodificação de apagamentos a N -upla \mathbf{z} é transformada na N -upla $\mathbf{w} = (w_1, w_2, \dots, w_N)$, conforme explicado a seguir. As i -ésimas coordenadas binárias em \mathbf{z} , em que $i \notin \mathbb{S}_{\mathcal{X}}$, terão os seus valores mantidos em \mathbf{w} . Para as demais coordenadas, a N -upla \mathbf{w} assume valores binários estimados pela decodificação de apagamentos nas posições em que os apagamentos forem corrigidos ou, em caso contrário, assume os valores binários de \hat{c} estimados ao fim da decodificação min-sum BP. É associada a \mathbf{w} a N -upla $\xi = (\xi_1, \xi_2, \dots, \xi_N)$ de valores QLLR, da seguinte forma:

$$\xi_j = \begin{cases} \hat{\xi}_j, & w_j = \Delta \text{ ou } w_j = \hat{c}_j, \\ -\hat{\xi}_j, & w_j \neq \hat{c}_j, \end{cases}$$

para $j \in \mathbb{S}_{\mathcal{X}}$. Para as demais coordenadas, $\xi_j = \hat{\xi}_j$. Observe-se na N -upla ξ que os valores de QLLR resultantes da decodificação min-sum BP anterior serão mantidos se, ao final da decodificação de apagamentos, os dígitos permanecerem apagados ou se os seus valores corresponderem aos mesmos que foram estimados pela decodificação min-sum BP anterior. Caso contrário, apenas o sinal do valor QLLR correspondente ao dígito é invertido.

Caso o número máximo de iterações I_{DH} do algoritmo DH não tenha sido atingido, a N -upla ξ é usada como entrada no algoritmo min-sum BP e assim é iniciada uma nova iteração do algoritmo DH. Seja $\mathbf{b} = (b_1, b_2, \dots, b_N)$ a N -upla binária obtida por decisão abrupta sobre os valores de ξ . Para o caso do número máximo de iterações I_{DH} ter sido atingido, a ocorrência de $\mathbf{b} \in \mathcal{C}$, indica que o ciclo de decodificação híbrida é finalizado com sucesso, enquanto a condição $\mathbf{b} \notin \mathcal{C}$ indica que a decodificação é encerrada sem sucesso.

III. CONSIDERAÇÕES SOBRE O CANAL BEC ARTIFICIAL

Um dos aspectos críticos da abordagem aqui apresentada é a regra a ser empregada para a atribuição de apagamentos na N -upla \mathbf{z} produzida pelo algoritmo min-sum BP, ao alcançar I_{BP} iterações sem sucesso. Tal atribuição de apagamentos é baseada em estimativas de probabilidade geradas pela decodificação iterativa min-sum BP. Em um canal BEC, os símbolos na saída são corretos (0's e 1's) ou desconhecidos (Δ 's). No caso do canal BEC artificial aqui introduzido, as saídas geradas continuam sendo 0's, 1's e Δ 's, porém a regra usada para gerá-las deve ser tal que a confiabilidade dos 0's e dos 1's seja a maior possível e que o número \mathcal{X} de apagamentos seja o maior possível, desde que não inviabilize a operação de correção de apagamentos do decodificador. Obviamente, o que se deseja com esse canal BEC artificial é que o conjunto $\mathbb{S}_{\mathcal{X}}$ contenha todas as posições que foram responsáveis pela falha do algoritmo min-sum BP. Isso só será

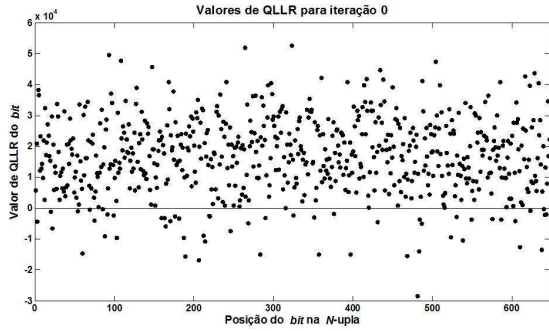


Fig. 1. Valores iniciais de QLLR.

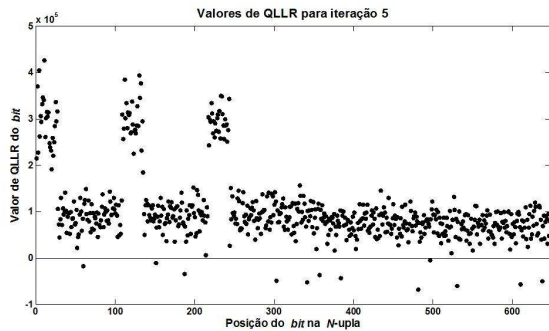


Fig. 2. Valores de QLLR após 5 iterações.

verdade se forem acertadas as alocações feitas para 0's e 1's da saída deste canal BEC.

No processo de decodificação iterativa de erros, usando o algoritmo min-sum BP, à medida em que são realizadas mais iterações, em geral uma quantidade maior de valores de QLLR correspondentes aos dígitos da N -upla recebida se distancia mais do valor nulo, especialmente para os dígitos em que a decodificação tem um maior grau de certeza [10]. Tais conclusões foram verificadas nos resultados de simulação de valores de QLLR que estão mostrados na Figura 1 e na Figura 2. Tais valores de QLLR foram obtidos usando um algoritmo min-sum BP para um código LDPC com comprimento de bloco $N = 648$ e taxa igual a $1/2$, denotado por LDPC (648,324), do padrão IEEE802.11n [11], em canal com AWGN e uma relação sinal-ruído (SNR) de 3,3 dB, transmitindo uma palavra-código toda nula, sem que haja perda de generalidade devido à linearidade do código.

A representação usada nas Figuras 1 e 2 mostra as posições dos dígitos do código no eixo horizontal (valores de 1 a N) e os valores de QLLR estimados no eixo vertical. A Figura 1 mostra os valores de QLLR recebidos antes da decodificação ter início e a Figura 2 mostra os valores estimados de QLLR após 5 iterações. A premissa de concentração de erros nos menores valores de QLLR é confirmada. No entanto, em razão da natureza do canal (AWGN) e da presença de ciclos no grafo de Tanner [12] do referido código LDPC, algumas posições contendo erro podem assumir valores elevados de QLLR.

IV. ESCOLHA DE UM VALOR PARA \mathcal{X}

Como já mencionado anteriormente, o parâmetro \mathcal{X} representa a quantidade de *bits* a serem apagados ao fim de uma

decodificação min-sum BP sem sucesso, usando-se o critério de apagar os dígitos correspondentes aos menores valores absolutos de QLLR. A determinação de \mathcal{X} é essencial para que seja obtido um desempenho satisfatório com esta abordagem, uma vez que no processo de decodificação iterativa de apagamentos, à medida em que se aumenta o número de apagamentos, se torna pouco provável encontrar equações de verificação de paridade que tenham apenas um único *bit* apagado e que, portanto, possam ser resolvidas. O conjunto contendo *bits* apagados e que não podem ser decodificados pelas equações de verificação de paridade é chamado de *conjunto de parada* [13].

Definição 2 (Referência [13]): Um conjunto de parada S em um código linear C é um subconjunto dos nós de variáveis (nós de *bits*) do grafo de Tanner deste código, tal que todos os vizinhos de S (nós de verificação de paridade) são conectados a S pelo menos duas vezes.

O número de *bits* que forma um conjunto de parada é denominado a cardinalidade do conjunto de parada.

Definição 3 (Referência [14]): A menor cardinalidade dentre todos os conjuntos de parada, denotada por $s(\mathbf{H})$, é definida como a distância de parada de C .

O número de conjuntos de parada distintos de um código linear C é o que determina o seu desempenho em decodificação iterativa de apagamentos [14]. Este desempenho pode ser medido pela probabilidade $P_{\mathbf{H}}(\epsilon)$, expressa em função de uma dada matriz de verificação de paridade \mathbf{H} , de um código com comprimento de bloco N , e em função da probabilidade de apagamento ϵ do canal BEC, como [9]

$$P_{\mathbf{H}}(\epsilon) = \sum_{\delta=0}^N \binom{N}{\delta} \epsilon^{\delta} (1-\epsilon)^{N-\delta} \left(\frac{S(\delta)}{T(\delta)} \right),$$

na qual o parâmetro $T(\delta)$ denota o número total de combinações distintas de δ posições com apagamentos e $S(\delta)$ denota o número de combinações, dentre as possíveis $T(\delta)$ combinações, que resultam em um conjunto de parada. Portanto a razão $S(\delta)/T(\delta)$ pode ser interpretada como a probabilidade de um dado conjunto de δ dígitos apagados ser um conjunto de parada. Todos os padrões com um número de apagamentos inferior à distância de parada $s(\mathbf{H})$, poderão ser corrigidos pelo decodificador iterativo de apagamentos. Logo, uma possível escolha de um valor para \mathcal{X} seria $\mathcal{X} < s(\mathbf{H})$, para o qual todos os padrões de apagamentos seriam aparentemente corrigíveis. No entanto, um valor baixo de $s(\mathbf{H})$ pode ser insuficiente para que os possíveis $\mathbb{S}_{\mathcal{X}}$ contêmam a maioria dos padrões de erros presentes nas N -uplas $\hat{\mathbf{x}}$, ainda que as coordenadas dos dígitos pertencentes a estes padrões exibam valores de QLLR próximos ao 0. Para ilustrar, o valor estimado para $s(\mathbf{H})$ para o código LDPC (648,324), obtido por simulação computacional, é $s(\mathbf{H}) = 20$. Caso seja adotado $\mathcal{X} = s(\mathbf{H}) = 20$, resulta que os possíveis $\mathbb{S}_{\mathcal{X}}$ abrigarão um percentual inferior a 3% dos erros remanescentes, ao fim de uma decodificação sem sucesso com o algoritmo min-sum BP. Este resultado sugere que, em geral, o valor de \mathcal{X} deva ser maior do que $s(\mathbf{H})$.

Definição 4: Uma N -upla contendo erros, resultante da decodificação min-sum BP, é definida como *detectada* se as

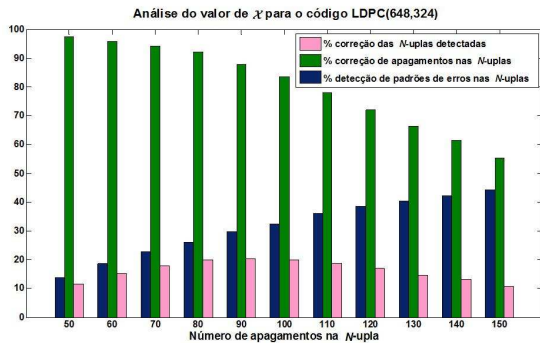


Fig. 3. Percentuais de detecção, correção de apagamentos e correção de erros em função do valor de \mathcal{X} para o código LDPC (648, 324).

posições de todos os seus erros estiverem contidas em $\mathcal{S}_{\mathcal{X}}$.

Definição 5: Uma N -upla contendo erros, resultante da decodificação min-sum BP, é definida como *corrigida* caso possa ser *detectada* e a palavra binária \hat{c} resultante da decodificação iterativa de apagamentos pertencer ao código \mathcal{C} , i.e., se $\hat{c} \in \mathcal{C}$.

Para a determinação do melhor valor de \mathcal{X} para o código LDPC (648, 324), foi realizada a simulação computacional do algoritmo DH para diversos valores de \mathcal{X} maiores do que $s(\mathbf{H})$ e foram calculados o percentual de detecção de padrões de erros, o percentual de correção de apagamentos e o percentual de correção dos erros na N -upla recebida. A Figura 3 exibe estes três percentuais. Observa-se, para um aumento do valor de \mathcal{X} , um aumento no percentual de detecção dos padrões de erros, que vem porém acompanhado de uma queda no percentual de correção de apagamentos, em função principalmente da presença do conjunto de parada. No que se refere ao percentual de correção de erros, este atinge um valor máximo de aproximadamente 20% para $\mathcal{X} = 90$, associado a um percentual aproximado de 90% para a correção de apagamentos e a um percentual aproximado de 30% para a detecção de erros. Este ensaio foi realizado para uma relação sinal-ruído de 3,2dB em canal com AWGN. O valor $\mathcal{X} = 90$ corresponde a $4,5s(\mathbf{H})$.

V. SIMULAÇÃO, RESULTADOS E COMENTÁRIOS

O algoritmo DH foi testado para o código LDPC (648, 324) em um canal com AWGN e modulação BPSK, empregando o melhor valor encontrado para \mathcal{X} que foi igual a 90, e um decodificador min-sum BP com um número reduzido de iterações. O desempenho do algoritmo DH proposto é comparado ao de um decodificador min-sum BP que funciona com um número máximo de iterações igual a 200. A etapa de decodificação min-sum BP do algoritmo DH foi analisada para duas situações, numa delas para no máximo 25 iterações e numa outra para no máximo 50 iterações. Deste modo, foi verificado que com $I_{DH} = 2$, ou seja, aplicando o algoritmo DH com no máximo duas iterações, obtém-se desempenho semelhante ao do decodificador min-sum BP, necessitando no entanto um número global de iterações na etapa de decodificação min-sum BP bem inferior ao do que usa apenas min-sum BP.

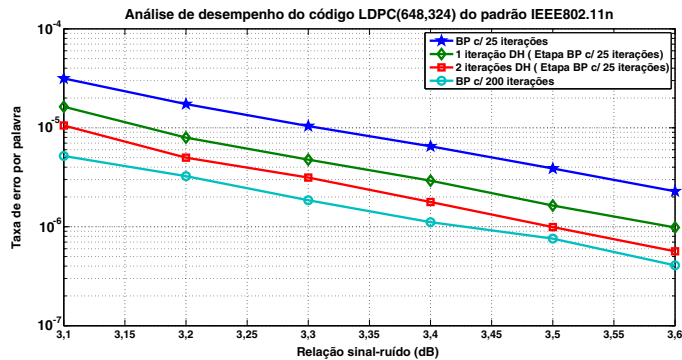


Fig. 4. Curvas de desempenho para o código LDPC (648, 324) na região de patamar de erros, para DH usando min-sum BP com 25 iterações.

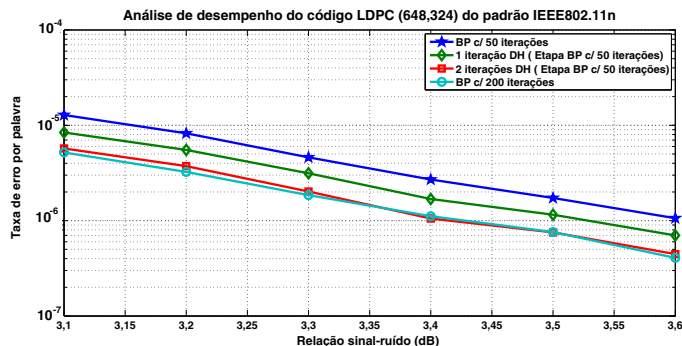


Fig. 5. Curvas de desempenho para o código LDPC (648, 324) na região de patamar de erros, para DH usando min-sum BP com 50 iterações.

A etapa adicional de decodificação iterativa de apagamentos, usada pelo algoritmo DH, permite alcançar um desempenho equivalente ao do decodificador iterativo min-sum BP em termos de taxa de erro por palavra (WER - *Word-error rate*), porém sendo executado de forma mais rápida. Esta abordagem atua prioritariamente na região de patamar de erros (*error floor*), com base nas seguintes premissas:

- Apresenta um melhor desempenho porque, em geral, os erros remanescentes de uma decodificação min-sum BP sem sucesso fazem parte de conjuntos de armadilha de pequena cardinalidade [6], podendo ser mais facilmente detectados ainda que seja adotado um valor baixo para \mathcal{X} .
- Apresenta maior eficiência computacional, uma vez que a quantidade de erros nesta região é significativamente baixa e a inclusão do processo de decodificação iterativa de apagamentos não implicará em custo computacional significativo.

As Figuras 4 e 5 mostram curvas de desempenho para o código LDPC (648, 324) na região de patamar de erros, sendo que o eixo horizontal representa a relação sinal-ruído (SNR), em dB, e o eixo vertical representa a taxa de erro por palavra (WER). Na Figura 4 são mostradas quatro curvas de desempenho sendo que uma corresponde à decodificação min-sum BP com 200 iterações, que serve de referência, e as demais representam respectivamente o desempenho para a decodificação min-sum BP com 25 iterações, o desempenho do algoritmo DH com uma iteração, contendo uma etapa de

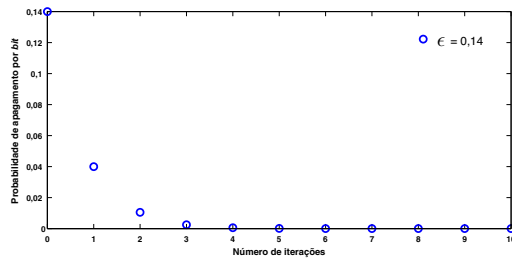


Fig. 6. Gráfico da probabilidade de apagamento por *bit* em função do número de iterações, para o código LDPC (648, 324).

decodificação min-sum BP com 25 iterações, e o desempenho do algoritmo DH com duas iterações, contendo uma etapa de decodificação min-sum BP com 25 iterações. Tomando-se por base um valor de WER de 3×10^{-6} observa-se que para um decodificador min-sum BP com 200 iterações este desempenho é obtido para uma relação sinal-ruído de aproximadamente 3,22 dB ao passo que para o decodificador min-sum BP com apenas 25 iterações isto é obtido para 3,55 dB de relação sinal-ruído. Para o algoritmo DH com uma iteração, usando uma etapa de decodificação min-sum BP com 25 iterações, obtém-se WER de 3×10^{-6} para uma relação sinal-ruído de um pouco menos de 3,4 dB e para duas iterações do algoritmo DH isto é obtido para uma relação sinal-ruído de um pouco menos de 3,31 dB. Este último resultado é de apenas 0,1 dB inferior ao resultado obtido para a decodificação min-sum BP com 200 iterações, utilizando-se, no entanto, um valor máximo global de 50 iterações na etapa de decodificação min-sum BP.

Na Figura 5 são mostradas quatro curvas de desempenho sendo que uma corresponde à decodificação min-sum BP com 200 iterações, que serve de referência, e as demais representam respectivamente o desempenho para a decodificação min-sum BP com 50 iterações, o desempenho do algoritmo DH com uma iteração, contendo uma etapa de decodificação min-sum BP com 50 iterações, e o desempenho do algoritmo DH com duas iterações, contendo uma etapa de decodificação min-sum BP com 50 iterações. Tomando-se por base um valor de WER de 3×10^{-6} , observa-se para um decodificador min-sum BP com 200 iterações que este desempenho é obtido para uma relação sinal-ruído de aproximadamente 3,22 dB, ao passo que para o decodificador min-sum BP com 50 iterações isto é obtido para 3,37 dB de relação sinal-ruído. Para o algoritmo DH com uma iteração, usando uma etapa de decodificação min-sum BP com 50 iterações, obtém-se este desempenho para uma relação sinal-ruído de pouco menos que 3,3 dB e para duas iterações do algoritmo DH isto é obtido para uma relação sinal-ruído de pouco menos que 3,23 dB. Este último resultado é equivalente ao resultado obtido para a decodificação min-sum BP com 200 iterações, utilizando-se, no entanto, de no máximo 100 iterações na etapa de decodificação min-sum BP.

A introdução da decodificação iterativa de apagamentos, além de ser mais simples do que a decodificação min-sum BP, permite ao algoritmo DH, em vários casos, efetuar a correção de erros com um número global menor de iterações. A técnica de evolução de densidade (*density evolution*) pode ser empregada para determinar o número mínimo necessário de

iterações do decodificador para a correção dos apagamentos, em função da probabilidade de apagamento ϵ e das funções de distribuição de graus [9]. O código LDPC (648, 324) analisado possui a função de distribuição que representa a fração de ramos que é conectada a nós de *bits* dada por $\lambda(x) = 0.25x + 0.341x^2 + 0.409x^{11}$ e a função de distribuição que representa a fração de ramos que é conectada a nós de verificação de paridade dada por $\rho(x) = 0.637x^6 + 0.363x^7$. A Figura 6 mostra que, para este código LDPC, em aproximadamente três iterações o decodificador iterativo de apagamentos converge. Nesta análise considera-se uma estimativa de probabilidade de apagamentos do canal igual a 0,14 que é referente à razão do número de apagamentos \mathcal{X} pelo comprimento de bloco do código N .

AGRADECIMENTOS

W. P. S. Guimarães agradece à Coordenação de Meteorologia da Universidade do Estado do Amazonas - UEA por disponibilizar a sua rede de computadores em *cluster* para a realização das simulações apresentadas neste artigo. V. C. da Rocha Jr. agradece ao Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq pelo apoio parcial recebido através do Projeto 306612/2007-0. Os autores agradecem ao doutorando José Sampaio de Lemos Neto pelos comentários e sugestões.

REFERÊNCIAS

- [1] R. G. Gallager, "Low-density parity-check codes". *IRE Trans. on Info. Theory*, v. 8, pp. 21-28, 1962.
- [2] J. C. Moreira and P. G. Farrell, *Essentials of Error-Control Codes*. New York, USA, John Wiley & Sons Ltd., 2006.
- [3] A. H. Banihashemi, A. Nough and P. Zarrinkat, "Hybrid (multistage) decoding of low-density parity-check codes". *40th Annual Allerton Conf. on Communications, Control and Computing* (Oct. 2002: Illinois, USA). Proceedings. Illinois, USA, pp. 1437-1438, 2002.
- [4] Y. Han and W. Ryan, "Low-floor decoders for LDPC codes". *IEEE Transactions on Communications*, v.57, pp. 1663-1673, Jun. 2009.
- [5] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier and X. -Y. Hu, "Reduced-complexity decoding of LDPC codes", *IEEE Trans. on Communications*, v. 53, n. 8, pp. 1288-1299, 2005.
- [6] T. Richardson, "Error floors of LDPC codes", *Proc. 41st Allerton Conf. Comm., Control, and Comput.*, Monticello, IL, 2003.
- [7] S. Lin and D. J. Costello Jr., *Error Control Coding: Fundamentals and Applications*. Prentice Hall, Inc. Englewood Cliffs, New Jersey, 2004.
- [8] P. R. Freitas, V. C. da Rocha Jr. and J. S. Lemos-Neto, "On the iterative decoding of binary product codes over the binary erasure channel", *Int. Symp. on Wireless Comm. Systems*, Aachen, Germany, pp. 126 - 130, 2011.
- [9] S. Johnson, *Iterative Error Correction Turbo, Low-Density Parity-Check and Repeat-Accumulate Codes*. New York, USA: Cambridge University Press, 2010.
- [10] Z. Ai-Min and Y. Sen-Jie, "A modified belief propagation decoding algorithm of LDPC codes for fast convergence". *International Conference on Communication Software and Networks* (Feb. 2009: Macau). Proceedings. Macau, pp. 518-522, 2009.
- [11] IEEE 802.11n Wireless LAN Medium Access Control MAC and Physical Layer PHY specifications, IEEE 802.11n-D1.0, 2006.
- [12] R. M. Tanner, "A recursive approach to low complexity codes". *IEEE Trans. on Info. Theory*, v. 27, pp. 533-547, 1981.
- [13] C. Di, D. Proietti, I. E. Telatar, T. J. Richardson and R. L. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel". *IEEE Trans. on Info. Theory*, v. 48, n. 6, pp. 1570-1579, 2002.
- [14] M. Schwarz and A. Vardy, "On the stopping distance and the stopping redundancy of codes", *IEEE Trans. on Info. Theory*, v. 52, n. 3, pp. 922-932, 2006.