

Avaliação do desempenho de algoritmos de decodificação com informação de medida de canal

Rebecca C. de Albuquerque e Daniel C. Cunha

Resumo—O presente trabalho tem como objetivo apresentar uma avaliação de desempenho das três variantes do algoritmo de Chase frente ao algoritmo de decodificação por decisão abrupta, assumindo-se o uso de modulação binária e canal com ruído Gaussiano branco aditivo. Adicionalmente, é feita uma investigação da complexidade de cada algoritmo de Chase em termos do número de operações matemáticas efetuadas.

Palavras-Chave—algoritmo de decodificação, decisão suave, complexidade computacional.

Abstract—The objective of this work is to present an performance evaluation of the three variations of Chase algorithm compared to the hard decision decoding algorithm, considering the use of binary modulation and Additive White Gaussian Noise (AWGN) channel. Also, an investigation about the complexity of each Chase algorithm is realized in terms of the performed mathematical operations.

Keywords—decoding algorithm, soft decision, computational complexity.

I. INTRODUÇÃO

Os códigos corretores de erros têm como objetivo principal corrigir erros que podem ocorrer durante a transmissão de informação em um sistema de comunicações. O desempenho de um código corretor de erros está diretamente relacionado à técnica de decodificação empregada no receptor. Considerando o uso de códigos de bloco lineares, existem dois métodos de decodificação tradicionais: a decodificação por decisão abrupta (HDD, do inglês *Hard Decision Decoding*) e a decodificação por decisão suave (SDD, do inglês *Soft Decision Decoding*). Na decodificação HDD, erros podem ser introduzidos naturalmente pelo processo de decisão abrupta, enquanto na decodificação SDD, a utilização das saídas dos filtros casados ajudam a melhorar a estimação dos bits recebidos ao preço de um aumento de complexidade [1], [2].

Neste trabalho, é realizada uma avaliação do desempenho das três variantes do algoritmo SDD proposto em [3] frente ao algoritmo HDD, assumindo-se o uso de modulação binária e canal com ruído Gaussiano branco aditivo. Adicionalmente, é feita uma investigação da complexidade de cada algoritmo SDD em termos do número de operações matemáticas efetuadas. Na seção II, a decodificação com informação de medida de canal é descrita. Na seção III, resultados de simulação são apresentados. Por fim, na seção IV, são apresentadas as conclusões e perspectivas de trabalhos futuros.

Departamento de Engenharia Elétrica, Escola Politécnica, Universidade de Pernambuco, Recife-PE, Brasil, E-mails: rebeccacamile@gmail.com, dccunha@gmail.com. Este trabalho foi financiado pelo PIBIC/Polí.

II. DECODIFICAÇÃO COM INFORMAÇÃO DE MEDIDA DE CANAL

Considere um código de bloco $C(n, k, d)$, em que n , k e d representam o comprimento da palavra-código, o número de bits de informação e a distância mínima do código, respectivamente. A maioria das aplicações que utiliza códigos de bloco faz uso de uma regra de decodificação incompleta chamada decodificação BDD (do inglês, *Bounded Distance Decoding*). Esta regra consiste em decodificar todos os padrões de erro de peso menor ou igual à capacidade de correção $t = \lfloor \frac{d-1}{2} \rfloor$ do código. Para aumentar o valor de t e, consequentemente, melhorar o desempenho do decodificador, é comum a utilização de algoritmos SDD. O algoritmo considerado neste trabalho é o algoritmo de decodificação com informação de medida de canal proposto por Chase [3].

Basicamente, o algoritmo de decodificação com informação de medida de canal necessita de dois parâmetros de entrada. O primeiro deles é a sequência binária gerada por decisões abruptas das saídas dos filtros casados. O segundo parâmetro é a própria sequência de valores reais observada na saída do canal, denotada por $\rho = [\rho_1, \rho_2, \dots, \rho_n]$. Estes valores reais representam o nível de confiabilidade referente a cada bit da palavra recebida. Quanto maior for o valor de confiabilidade, menor é a probabilidade de o bit correspondente ter sido afetado fortemente pelo ruído.

O algoritmo proposto em [3] consiste em uma seqüência de passos descrita a seguir:

- 1) Adicionar um padrão de teste \mathbf{T} , vetor de n bits com 1's nas posições menos confiáveis, à palavra recebida \mathbf{Y} , previamente decodificada por HDD, gerando a palavra \mathbf{Y}' :

$$\mathbf{Y}' = \mathbf{Y} \oplus \mathbf{T}, \quad (1)$$

em que \oplus representa a operação soma módulo 2.

- 2) Verificar se a palavra \mathbf{Y}' pode ser decodificada pelo decodificador BDD, gerando um padrão de erro \mathbf{Z}' .
- 3) Caso não possa, voltar ao passo 1 com um novo padrão \mathbf{T} . Se os padrões \mathbf{T} se esgotarem, saltar para o passo 8.
- 4) Adicionar o mesmo padrão de teste \mathbf{T} ao padrão de erro \mathbf{Z}' , originando \mathbf{Z}_T :

$$\mathbf{Z}_T = \mathbf{Z}' \oplus \mathbf{T}. \quad (2)$$

- 5) Calcular o peso analógico W_ρ do padrão \mathbf{Z}_T :

$$W_\rho = \sum_{i=1}^n \rho_i Z_{T_i}, \quad (3)$$

e armazenar o padrão cujo peso analógico é mínimo.

- 6) Voltar ao passo 1 com um novo padrão \mathbf{T} , caso o conjunto de padrões de teste não tenha sido esgotado. Caso contrário, ir para o passo 7.
- 7) Decodificar $\mathbf{X} = \mathbf{Y} \oplus \mathbf{Z}_T$ e encerrar o algoritmo.
- 8) Decodificar $\mathbf{X} = \mathbf{Y}$ e encerrar o algoritmo.

Existem três variantes do algoritmo de Chase baseadas no número de padrões de teste \mathbf{T} utilizados. Para o Algoritmo 1, são gerados todos os padrões de teste com $\lfloor d/2 \rfloor$ 1's. Devido ao grande número de padrões de teste, o emprego do Algoritmo 1 só se justifica para códigos com distância mínima d pequena. O Algoritmo 2 faz uso de um conjunto consideravelmente menor de padrões de teste. São consideradas todas as combinações de 1's nas $\lfloor \frac{d}{2} \rfloor$ posições menos confiáveis, produzindo assim, $2^{\lfloor d/2 \rfloor}$ padrões de teste. Por fim, o Algoritmo 3 é o mais simples de todos, pois utiliza o menor conjunto de padrões de teste: $\lfloor \frac{d}{2} \rfloor + 1$ padrões com i 1's nas i posições menos confiáveis, sendo $i = 1, 3, \dots, d - 1$ e $i = 0$ para d par e $i = 0, 2, 4, \dots, d - 1$ para d ímpar.

III. SIMULAÇÕES

Foram realizadas simulações dos algoritmos de decodificação BDD e de Chase- j ($j = 1, 2, 3$) para o código de Golay estendido $C(24, 12, 8)$. A Fig. 1 ilustra o desempenho dos algoritmos por meio da curva de probabilidade de erro de bit (P_b) versus razão sinal-ruído E_b/N_0 , em que E_b é a energia por bit de informação e N_0 , a densidade espectral de potência do ruído Gaussiano. Para uma $P_b = 10^{-3}$, observou-se ganhos de codificação dos algoritmos SDD em relação ao algoritmo HDD, cujos valores assumiram 1 dB, 1,5 dB e 1,6 dB para os algoritmos de Chase-3, 2 e 1, respectivamente.

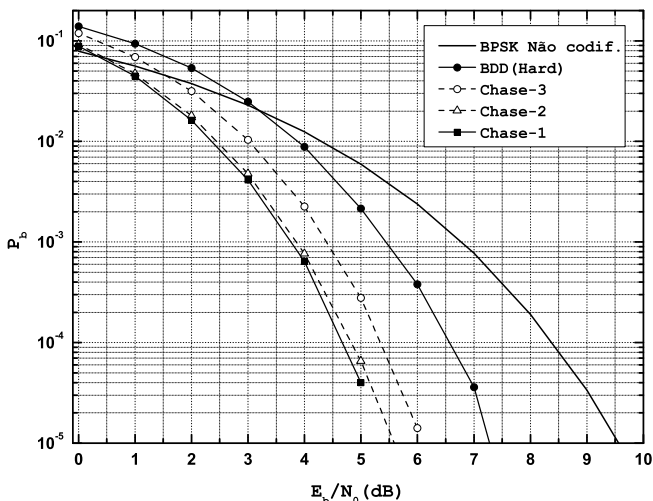


Fig. 1. Desempenho do código de Golay estendido $C(24, 12, 8)$ utilizando modulação binária em um canal com ruído Gaussiano branco aditivo.

Adicionalmente, foi computado o número de somas módulo 2, multiplicações e comparações para cada algoritmo de Chase. A quantidade de somas e multiplicações registrada para os algoritmos de Chase-3, 2 e 1 foi da ordem de $0,4 \cdot 10^3$, $1,3 \cdot 10^6$ e $8,6 \cdot 10^8$ operações, respectivamente. Tais valores estão diretamente ligados ao número de padrões de teste gerados em cada algoritmo. Diante do desempenho

apresentado pelos algoritmos de Chase, percebe-se que o aumento da complexidade do Algoritmo 1 não proporciona uma melhora de desempenho considerável em relação ao Algoritmo 2. Logo, o Algoritmo 2 é o que melhor relaciona complexidade computacional e desempenho.

A Fig. 2 ilustra o número de comparações realizadas em função de E_b/N_0 . Além de estar associado à geração de padrões de teste, o número de comparações também se refere à busca pelo peso analógico mínimo. Encontrar um padrão de teste \mathbf{T} que gere uma palavra \mathbf{Y}' decodificável implica na realização de uma comparação. Enquanto o número de comparações se manteve praticamente constante nos Algoritmos 3 e 2, no Algoritmo 1 ocorreu uma diminuição do valor com o aumento de E_b/N_0 .

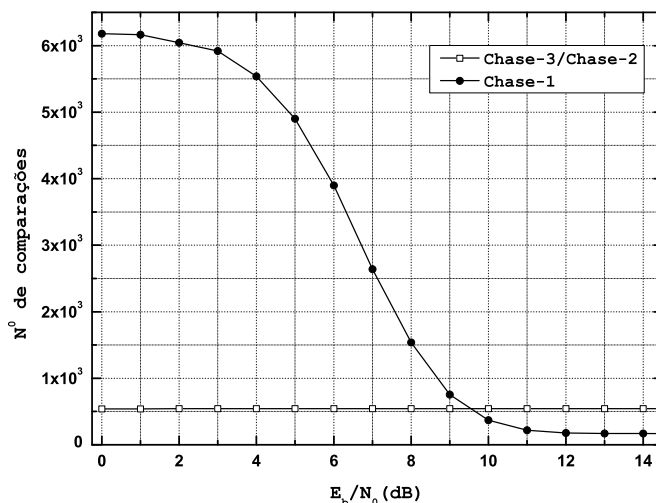


Fig. 2. Número de comparações realizadas em cada algoritmo de Chase.

Tal comportamento pode ser justificado pelo fato de que para um valor de E_b/N_0 alto, é pouco provável que o ruído consiga produzir erros na decodificação HDD. Consequentemente, quando um padrão de teste \mathbf{T} é somado à palavra \mathbf{Y} supostamente correta, uma palavra \mathbf{Y}' não decodificável é gerada e, por isso, a comparação não é realizada.

IV. CONCLUSÕES

Este trabalho apresentou uma avaliação do desempenho das três variantes do algoritmo de Chase frente ao algoritmo de decodificação HDD. Também foi realizada uma investigação da complexidade de tais algoritmos em termos do número de operações matemáticas realizadas. O estudo mostrou que o algoritmo de Chase-2 é o que apresenta a melhor relação entre complexidade e desempenho. Como perspectiva de trabalhos futuros, sugere-se a aplicação dos algoritmos abordados em outros canais típicos de comunicações, como o canal com ruído impulsivo.

REFERÊNCIAS

- [1] R. H. Morellos-Zaragoza, *The Art of Error Correcting Coding*. John Wiley & Sons, 2002.
- [2] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. John Wiley & Sons, 2005.
- [3] D. Chase, "A Class of Algorithms for Decoding Block Codes With Channel Measurement Information". *IEEE Transactions on Information Theory*, vol. IT - 18, n. 1, pp 170-182, Janeiro 1972.