Códigos de Bloco[a)]

Dennys L. A. Rocha[1, b)] and Gabriel Adriano de Melo[1, c)]
Alunos de graduação do Instituto Tecnológico de Aeronáutica

(Dated: 17 de novembro de 2017)

Neste trabalho estudou-se a decodificação (correção de erros) de sinais que passam por canais ruidosos. Para isso, empregou-se o método de *Hamming (7,4,3)* como solicitado no pré-relatório e também um código de livre escolha, o *Golay (24,12,8)*. Os algoritmos criados foram feitos nas linguagens *Java* e *MATLAB*. Eles operam de modo a gerar textos binários aleatórios e trabalhar sobre eles adicionando ruídos e corrigindo seus erros.

Palavras-chave: Hamming, Golay, erro.

In this work we studied the decoding (error correction) of signals passing through noisy channels. For that, the Hamming method (7,4,3) was used as requested in the pre-lab and also a free choice code, the Golay (24,12,8). The algorithms were created in Java and MATLAB languages. They operate in order to generate random binary texts and work on them by adding noise and correcting their errors.

Keywords: Hamming, Golay, error.

I. INTRODUÇÃO

No estudo das comunicações digitais, nôs deparamos com métodos de correção para erros gerados na transmissão de uma mensagem. Neste relatório serão estudados os métodos de Hamming (7,4,3) e Golay (24,12,8) (por simplicidade serão chamados respectivamente de Hamming e Golay apenas).

O método de Hamming utiliza código de 4 bits (chamado de palavra) e sua palavra-código possui 7 bits. Ele é capaz de corrigir um erro e detectar até dois. Seu modo de funcionamento se baseia em, dados os 4 bits de código, adicionar mais três bits (chamados bits de paridade) para correção de erros.

De forma semelhante ao método de Hamming, o método de Golay também trabalha com bits de redundância, mas este opera com códigos de 12 bits e palavras-código de 24 bits. Ele é capaz de corrigir até três erros e detectar sete.

Os números que acompanham os nomes dos métodos são, respectivamente, o tamanho da palavra-código (n), o tamanho do código (k) e a distância mínima entre duas palavras-código (r). Define-se como taxa a razão entre o tamanho do código e o tamanho da palavra-código (Equação 1):

$$R = \frac{k}{n} \tag{1}$$

Ambos os códigos atuam de modo a corrigir erros gerados pelo canal onde a mensagem passa, então é razoável definir um parâmetro que meça o quanto de informação errônea foi transmitida, comparando-se a mensagem enviada da mensagem recebida. Esse parâmetro é chamado de *Bit Error Rate* (BER) e é definido pela Equação 2:

$$BER = \frac{N_{erros}}{N} \tag{2}$$

onde N_{erros} é a quantidade total de bits trocados e N a quantidade total de bits transmitidos.

II. DESCRIÇÃO DOS ALGORITMOS

Nesta atividade, foi utilizado um canal binário simétrico (BSC), onde a probabilidade de troca de bit independe do valor do bit. Os códigos transmitidos pela fonte são tratados ao passar por esse canal. Cada um dos códigos deve estimar, a partir de seu método de correção, quais bits foram trocados e corrigi-los.

A. Método de Hamming

Para o codificador de Hamming, devemos gerar os bits de paridade conforme ilusta a Figura 1.

De modo a gerar os bits de paridade, constrói-se uma matriz de paridade G de tal forma que todas as colunas de comprimento 4 sejam linearmente independentes. Essa matriz, ao ser multiplicada pelo vetor código $v_{1\times4}$, deve retornar um vetor $w_{1\times7}$ chamado de palavra-código, que tem a informação nos quatro primeiros bits e as paridades nos três últimos, isto é:

$$w = v \cdot G = b_1 b_2 b_3 b_4 p_1 p_2 p_3 \tag{3}$$

[a)] Auxílio matemático: Introdução à terceira atividade laboratorial de ELE-32. Disponível em `https://goo.gl/Xe8c5i`.

[b)] Endereço eletrônico: dennysrocha.1994@gmail.com
[c)] Endereço eletrônico: gaadrime.melo@gmail.com

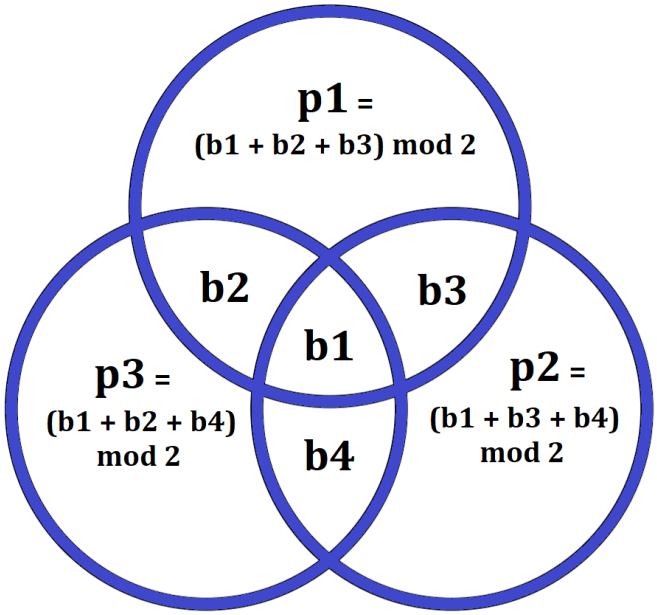

Figura 1. Diagrama ilustrativo para a construção dos bits de paridade e assim a palavra código.

A mensagem w é então enviada ao decodificador que, com uma matriz

$$H^T = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

encontra a síndrome

$$s = w \cdot H^T \tag{5}$$

que, devido à propriedade $v \cdot H^T = 0_{1\times 3}$, se simplifica a Equação 6:

$$s = e \cdot H^T \tag{6}$$

onde e é o erro gerado pelo canal.

A partir da síndrome obtida, decide-se qual é o erro fazendo a suposição de que ele terá peso de Hamming igual a 1.

A mensagem estimada será, portanto, dada pela Equação 7

$$v \equiv w + e \ (mod\, 2) \tag{7}$$

B. Método de Golay

Semelhante ao método de Hamming, no método de Golay usa-se uma matriz de paridade

$$G = \left[I_{12} \,|\, \begin{pmatrix} A_1|1 & A_2|1 & \dots & A_{12}|1 \end{pmatrix}^T \right] \tag{8}$$

cujos A_i, $i = 1, ..., 11$, são obtidos sucessivamente através do vetor

$$A_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{9}$$

mudando a posição do primeiro elemento para a última posição. Com os A_i em mãos, adiciona-se uma coluna ao final de cada vetor com o valor 1. O vetor A_{12} é um vetor com apenas a ultima coluna igual a zero. Resulta em:

$$A = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \tag{10}$$

O código $v_{1\times 12}$ se transforma então na palavra-código $w_{1\times 24}$. Para decodificá-la, utilizamos os seguintes passos:

1. Computamos a primeira síndrome:

$$s_1 \equiv w \cdot G^T \ (mod\, 2) \tag{11}$$

onde, nesse caso, $H = G^T$.

2. Se o peso de Hamming da primeira síndrome for menor ou igual a três, então o padrão de erro será

$$u = \left[s_1 \,|\, 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \right]. \tag{12}$$

3. Se o peso de Hamming da primeira síndrome for maior ou igual a três mas para algum $i = 1, ..., 12$ o peso de Hamming de $s + A_i$ for menor ou igual a dois, então o padrão de erro será

$$u_1 = [s + A_i \,|\, e_i] \tag{13}$$

onde e_i é a i-ésima linha da matriz I_{12}. A mensagem é, portanto:

$$v \equiv w + u_1 \ (mod\, 2) \tag{14}$$

4. Computamos a segunda síndrome:

$$s_2 = s \cdot A \tag{15}$$

5. Se o peso de Hamming da segunda síndrome for menor ou igual a três, então o padrão de erro será

$$u_2 = \left[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \,|\, s_2 \right]. \tag{16}$$

A mensagem é, portanto:

$$v \equiv w + u_2 \ (mod\, 2) \tag{17}$$

6. Se o peso de Hamming da segunda síndrome for maior ou igual a três mas para algum $i = 1, ..., 12$ o peso de Hamming de $s_2 + A_i$ for menor ou igual a dois, então o padrão de erro será

$$u_3 = [e_i \mid s_2 + A_i] \tag{18}$$

A mensagem é, portanto:

$$v \equiv w + u_3 \ (mod\, 2) \tag{19}$$

7. Caso contrário, reenviar a mensagem da forma que foi recebida ou então, se possível, pedir retransmissão.

III. RESULTADOS

1. Qual foi a maior dificuldade de implementar o decodificador para o código de Hamming?

Para a implementação do código de Hamming, a matriz H^T foi escrita na forma de $2^n - 1$ linhas por n colunas, de forma que todas as combinações de vetores independentes possíveis fossem escritas, a menos do vetor nulo. Para facilitar na decodificação, escreveu-se os vetores em ordem decrescente (a partir do vetor todo 1), a exceção dos vetores com peso de Hamming unitário, que foram escritos na forma de uma matriz identidade nas últimas linhas. Assim, a maior dificuldade, foi que ao contar a síndrome e relacionar com o erro, fazia-se uma contagem decrescente a partir do vetor todo 1, decrementando-o, e dever-se-ia pular os vetores de Hamming com peso um nessa contagem.

2. Qual foi o método utilizado para encontrar o código maior? Este método é extensível para qualquer tamanho de bloco?

O primeiro método consistia em determinar um conjunto de vetores de tamanho n e distância de Hamming mínima k entre eles. Nesse método, partia-se de um vetor inicial de tamanho n com todos os elementos unitários e decrementava-se, verificando se o resultado tinha distância de Hamming k entre todos os vetores da lista em construção (cujo primeiro elemento era o vetor inicial), e em caso afirmativo, adicionava-o à lista. Com a lista concluída, fazia-se uma outra seleção, construindo-se uma lista de todas as síndromes possíveis, com elementos iniciais de peso de Hamming 1 (erros de apenas um bit), em seguida realizando todas as combinações de até Q elementos da lista anterior. Assim, gerava-se uma relação entre as síndromes e os erros nos bits. Utilizou-se o tipo unsigned long long int para essas operações de vetores de bit, representando um vetor de bit de tamanho até 64. A princípio, utilizando-se uma representação maior que 64 bits, esse método é extensível para qualquer tamanho de bloco n, sendo selecionados vetores com distâncias mínima k, realizando-se Q combinações entre eles.

3. Qual é a relação medida entre o tamanho do bloco e o desempenho?

Para o código de Hamming, que corrige apenas um erro, um aumento no tamanho do bloco significava uma melhor taxa de transmissão, porém com uma maior taxa de erro de bits. Para o algoritmo de seleção de códigos com distância mínima k, verificou-se mais correções de erros de bits com o aumento de k.

4. Como o seu código se compara com a capacidade do canal? Utilize uma probabilidade de erro de bit de 10^-5.

A capacidade do canal binário simétrico pode ser calculada por $1 + p \cdot log_2 p + (1 - p) \cdot log_2(1 - p)$, o que resulta em 0.99982 bit por uso de canal, para a probabilidade de erro de bit de 10^-5. A nossa implementação do código de Golay utiliza uma taxa de $50 percento$, corrigindo até 3 erros em um bloco, diminuindo a taxa de erro de bits de 10^-5 para 0.00000. Assim, temos uma capacidade de aproximadamente 0.5 bit por uso de canal.

5. Qual é a complexidade de codificação e decodificação do seu sistema?

Para o primeiro método de seleção por força bruta, encontrados os v elementos de tamanho n e de distância k, a codificação se reduz a uma multiplicação de matrizes, com ordem $O(n * v)$ e a decodificação se resumia a uma outra multiplicação de matrizes de ordem $O(n * v)$ e uma busca de síndrome na tabela, de $O(1)$. A complexidade estava em achar tais elementos (construir a matriz g), de ordem $O(2^n)$ e na construção da tabela de ordem $O(v^Q)$. Para a nossa implementação de Golay 24, tanto o algoritmo da codificação quando o da decodificação apresentam ordem $O(12)$, provenientes da construção da matriz de paridade G e análise das síndromes geradas, respectivamente. A construção do canal BSC requeriu uma ordem $O(24)$ para tratar cada bit individualmente.

IV. CONCLUSÃO

REFERÊNCIAS

[1]Wikipédia, "Teorema de codificação da fonte — wikipédia, a enciclopédia livre," 2017. [Online; accessed 18-junho-2017].

[2]Gutenberg, "Free ebooks - project gutenberg," 2017. [Online; accessed 16-agosto-2017].

[3]R. Pavão, "Teoria da informação," 2017. [Online; accessed 15-agosto-2017].