ELE-32 Laboratório 1: Código de Bloco

Adrisson Rogério Samersla, *Graduação, ITA,* Matheus Vidal de Menezes, *Graduação, ITA,*

***Abstract*—This paper presents a theoretical analysis of the Hamming Codes in telecommunication. The mathematical structures are systematically presented leading to the Hamming Vector Space. Besides this the analytical deductions of the matrix representation are shown, allowing its generalization. Then the rate of success and efficiency of this method are discussed. In the results section, the Hamming 4/7 and a bigger one (8/14) are presented, tested and compared. Finally the questions raised during the class are discussed and answered with the theory presented and the results achieved.**

***Index Terms*—Binary Symmetric Channel (BSC), Hamming Code, Hamming Space, code rate, generalization, implementation, simulation.**

I. Introdução

Em um mundo extremamente computadorizado, nos quais os circuitos e os dados armazenados são digitais, a transmissão de bits é de suma importância para a comunicação entre sistemas. Dessa forma, a utilização de mecanismos de redundância é fundamental para garantir a validade da informação transmitida. Dentre os artifícios já consagrados pela literatura encontra-se a Codificação de Hamming. Essa estratégia emprega bits a mais, chamados bits de paridade, para identificar e corrigir erros no processo de transmissão. A análise, implementação e aperfeiçoamento dessa codificação serão os objetivos deste relatório.

II. Análise Teórica

A Codificação de Hamming mais conhecida, devido a sua simplicidade, é a Hamming(4,7), isto é, com 4 bits de informação e 3 bits de paridade, totalizando os 7 bits usados. Essa configuração em particular possui uma análise gráfica muito intuitiva, porém de difícil generalização, por isso o desenvolvimento será mais analítico, permitindo estender os resultados obtidos para códigos maiores. Inicialmente, algumas considerações sobre as estruturas matemáticas a serem usadas primam por serem feitas:

A. Estruturas Matemáticas

Os elementos a serem utilizados são bits (0 ou 1). Seja $U = \{0, 1\}$, definindo-se a Operações de Adição como xor binário ("ou exclusivo"):

$$\forall a, b \in U, a + b \triangleq a \oplus b \tag{1}$$

Uma definição alternativa pode usar aritmética modular, isto é, $a + b \triangleq a + b(mod 2)$.

O conjunto mais essa operação, $(U, \oplus)$, forma um Grupo (Abeliano, pois a operação comuta). É interessante notar que o elemento neutro da adição é o 0 e o inverso da operação (ou simétrico, por ser uma adição) é o próprio elemento, pois $1 \oplus 1 = 0 \oplus 0 = 0$.

Além da operação de adição, pode-se definir também uma operação de multiplicação usual. Assim, $(U, \times)$ é um monoide com o elemento neutro igual a 1.

Unindo essas duas definições, $(U, \oplus, \times)$ é um anel, pois é um grupo abeliano sob a operação de adição, um monoide sob a operação de multiplicação e com a operação de multiplicação distributiva sob a adição. Também é sabido que a operação de multiplicação é comutativa, logo $(U, \oplus, \times)$ é um anel comutativo, isto é, um corpo.

Além disso, as codificação podem ser definidas como ênuplas de tamanho N, de modo que a operação de soma de ênuplas herda da adição já definida (aplicado elemento por elemento) e a multiplicação por um escalar é a usual. Portanto, $(U^N, \oplus, \cdot)$ é um espaço vetorial sobre o corpo definido anteriormente (chamado Espaço de Hamming).

B. Modelagem do Canal

Para que a comunicação entre dois sistemas ocorra é necessário haver um canal pelo qual o sinal será transmitido. Justamente nessa fase mais física do processo é que podem ocorrer trocas de bits, isto é, os ruídos presentes podem perturbar o sinal de tal forma que seu significado mude de 0 para 1 (ou de 1 para 0). Para as análises a serem feitas neste artigo, serão considerados canais do tipo **BSC** (*Binary Symmetric Channel*), nos quais a probabilidade de um erro na transmissão de um bit é igual para o 1 e para o 0 (por isso simétrico).

C. Codificação de Hamming

Primeiramente, é útil definir a matriz H^T: Seja um Hamming(N,n) com $k = N - n$, então

$$H^T = \left[\frac{-P}{I_{k\times k}}\right] \tag{2}$$

Onde: P é uma matriz com dimensões $n \times k$ a ser definida posteriormente; $I_{k\times k}$ é a matriz identidade de ordem k e H^T tem dimensões $N \times k$

Dessa forma, pode-se definir G como:

$$G = \left[I_{n\times n}\middle|P\right] \tag{3}$$

Onde: G tem dimensões $n \times N$

Repare que a Matriz G codifica uma palavra de n bits em um código (de Hamming) de N bits:

$$v_{1\times N} = u_{1\times n} \cdot G \tag{4}$$

A submatriz de G que é a identidade garante que os primeiros n bits do novo código sejam os mesmos e que $rank(G) = n$. Outra forma de analisar-se a matriz G é pensá-la como um operador linear de $U^n \to U^N$.

Além disso, repare que a matriz H^T codifica uma palavra de k bits, chamada **síndrome**, que está associada com o erro durante a transmissão da informação:

$$s_{1\times k} = v_{1\times N} \cdot H^T \tag{5}$$

Primeiramente, G induz um subespaço vetorial de dimensão n em U^N, logo faltam k vetores (que geram os erros) para completarem a base com N vetores de U^N, por isso a síndrome tem k elementos. Dessa forma, $s = \emptyset$ indica que o código é valido, o que justifica a escolha anterior para as matrizes G e H^T. Seja $r = v+e$, onde v é um código válido e e representa qual bit foi trocado ("xor" com 1 é uma inversão de bit):

$$r = u_{1\times n} \cdot G + e \rightarrow r \cdot H^T = u_{1\times n} \cdot G \cdot H^T + e \cdot H^T \tag{6}$$

mas

$$G \cdot H^T = \left[I_{n\times n} \middle| P\right] \cdot \left[\frac{-P}{I_{k\times k}}\right] = I_{n\times n} \cdot (-P) + P \cdot I_{k\times k} \tag{7}$$

$$G \cdot H^T = (-P) + P = \emptyset_{n\times k} \tag{8}$$

Repare que o simétrico de um elemento, nesse corpo, é ele mesmo, portanto H^T também pode ser definida com P. Repare também que as matrizes G e H^T foram definidas justamente para que o produto se anulasse. Logo:

$$s \triangleq r \cdot H^T = u_{1\times n} \cdot \emptyset_{n\times k} + e \cdot H^T = e \cdot H^T \tag{9}$$

Dada a síndrome s do sinal recebido r, deseja-se descobrir qual o erro e associado, porém esse sistema de equações não é determinado, isto é, diferentes erros podem gerar a mesma síndrome. A estratégia da Decodificação de Hamming é escolher a classe de erros mais provável: como $p \leq 0.5$, a probabilidade de ocorrer 1 erro é muito maior do que 2 ou mais, logo o algoritmo focará em descobrir erros unitários.

Dessa forma a decodificação torna-se computacionalmente eficiente, pois basta determinar qual a linha da matriz H^T é igual à síndrome obtida, já que um erro unitário filtrará essa linha (multiplicando as outras por 0). Repare que uma combinação linear das outras linhas também formariam a síndrome desejada, mas estamos interessados apenas no erro com o menor Peso de Hamming, com o código válido de menor Distância de Hamming do recebido.

Para o algoritmo a cima apresentado encontrar um erro unitário apropriado, é necessário que as linhas da matriz H^T sejam diferentes entre si. Para o caso $n+k+1 = 2^k$, que será tratado na seção II-E, a matriz P é única, exceto permutações de suas linhas.

D. Probabilidade de Acerto

Seja p a probabilidade de erro para cada bit e sabendo que a Codificação de Hamming é capaz de corrigir apenas um erro unitário, portanto a probabilidade da mensagem ser transmitida sem enganos (isto é, a palavra que entrou no codificador ser a mesma que saiu do decodificador) é dada pela probabilidade de terem ocorridos exatamente 0 ou 1 erros durante a transmissão:

$$P_{corr} = (1-p)^N + N \cdot p \cdot (1-p)^{N-1} \tag{10}$$

Repare que a probabilidade de acerto independe, a priori, da escolha da matriz P, variando apenas com a probabilidade de erro de um bit p e com o tamanho da sequência de bits N. As figuras 1 e 2 ilustram essa dependência:

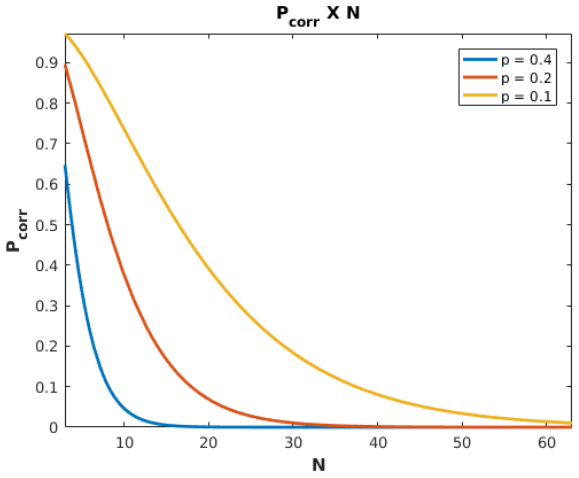

Fig. 1. Gráfico da função Pcorr(p,N) fixando-se p e variando N.

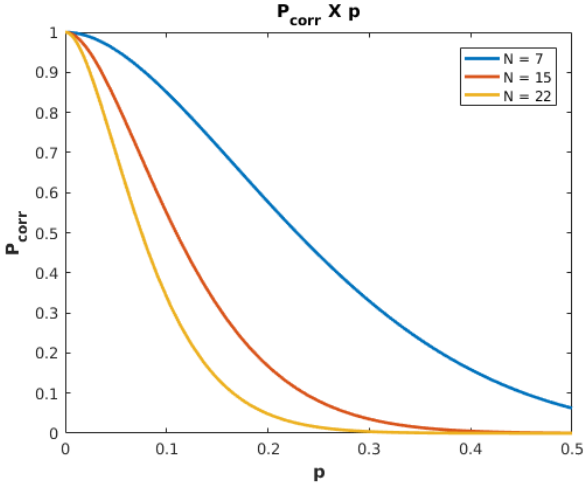

Fig. 2. Gráfico da função Pcorr(p,N) fixando-se N e variando p.

A probabilidade de erro na transmissão, aqui denotada por P_b é dada por:

$$P_b = 1 - P_{corr} = 1 - (1-p)^N - N \cdot p \cdot (1-p)^{N-1} \tag{11}$$

E. Eficiência da Transmissão

A medida de eficiência a ser utilizada será a taxa de bits de informação: quantos bits de informação são transmitidos sobre o tamanho total da sequência. Assim, seja uma Codificação

de Hamming com n bits de informação e k bits de paridade totalizando $N = n + k$ bits, a taxa será:

$$\zeta = \frac{n}{N} \tag{12}$$

A priori, não existe uma única relação entre k e n, porém minimizar N diminui a probabilidade de erros (figuras 1 e 2). Logo, é fundamental escolher sabiamente a quantidade de bits de paridade.

A matriz H^T deve ter linhas diferentes entre si, portanto são $n + k$ linhas diferentes, sendo k linhas fixas da matriz identidade. Além disso, a linha só com zeros não pode aparecer, pois representa um código válido, portanto as 2^k possibilidades de código de síndrome devem suprir as $n + k$ linhas mais a linha nula:

$$2^k \geqslant n + k + 1 \therefore 2^k - k \geqslant n + 1 \tag{13}$$

O menor valor de k para um n será dado pela igualdade (Hamming(4,7) é um desses casos). Quando isso ocorrer, a matriz P será única (exceto permutações de linha), pois tirando as k linhas da matriz identidade e a linha de zeros, há exatamente n síndromes diferentes para as n linhas.

III. Implementação

Com base na teoria explicada, foram implementadas as codificações requisitadas.

A. Codificador/Decodificador de Hamming 4/7

Primeiramente, foi implementado um codificador de canal para o código de Hamming 4/7, *i.e.*, 4 bits de informação e 3 de paridade, estes que servem para determinar e corrigir erros de transmissão através de um canal **BSC**. A matriz de paridade P, neste caso, fica definida como:

$$P_{4/7} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \tag{14}$$

Essa matriz é utilizada tanto na codificação, quanto na decodificação. Ela, inclusive pode ser enxergada como um grafo, conforme na Figura 3.

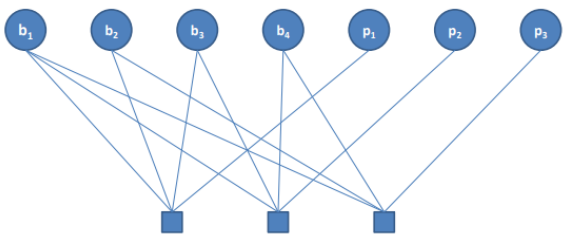

Fig. 3. Visualização em forma de grafo da relação bit de mensagem com bit de paridade no código Hamming 4/7. Note que cada linha da matriz de paridade diz quais bits de paridade um bit de informação se conecta por meio dos quadrados inferiores.

Para a decodificação, bastou-se percorrer as linhas da matriz H^t até achar a síndrome correspondente ao erro, o que é sempre possível, pois o número total de síndromes (três bits) a serem detectados, excetuando o caso que não há erro, é igual ao comprimento da palavra código, que também é o número de linhas da matriz H^t, *i.e.*, $2^k - 1 = n + k$, com $n = 4$ e $k = 3$. **Em suma, a maior dificuldade de se implementar o decodificador para o código de Hamming estaria em procurar, dentre os padrões de erros de transmissão, aquela de menor peso. Usando, porém, o fato visto na seção II-C de que a probabilidade de apenas 1 erro de bit é muito mais provável, esta dificuldade é diminuída. Ao invés de testar todas as combinações, usando força bruta, pode-se procurar aquelas com erro unitário, reduzindo a complexidade a $\boldsymbol{\Theta(n \cdot k)}$.**[1]

B. Codificador/Decodificador Maior 8/14

Em seguida, foi implementado um codificador/decodificador diferente do Hamming, porém com taxa de código semelhante. Para tanto, foi escolhida a mesma taxa, de modo que se tenha 8 bits de informação e 6 de paridade, totalizando 14 bits de palavra código.

Embora tenham a mesma taxa de código, as escolhas de como cada bit de paridade se relaciona com os bits de informação não está bem definida. Isso, porque há muito mais possibilidades de $2^6 - 1 = 63 > 14$. Com efeito, a menos de ordem, há mais de 1 possibilidade de escolha para a matriz de paridade P, consequentemente, as matrizes G e H^t também não são únicas. Vale ressaltar, entretanto, que a probabilidade não deve se comportar de modo diferente, conforme visto na Eq. 10. Logo, **o método utilizado para encontrar o código maior foi baseado na lógica da matriz de paridade do Hamming 4/7, *i.e.*, basicamente fez-se o 1º bit de mensagem estar conectado com todos os 6 bits de paridade, o 2º e os demais até o 7º bit se relacionam com 5 bits de paridade. Dessa forma, resta se fazer o relacionamento do 8º bit de informação com os de paridade. Neste caso, escolheu-se um qualquer, relacionando-se com $\mathbf{k - 2 = 4}$ bits de paridade, dado que o comportamento probabilidade de erro de 1 bit não deve depender da escolha da matriz de $\mathbf{H^T}$.**[2] Assim, a matriz P de paridade ficou escolhida como:

$$P_{8/14} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \tag{15}$$

Assim, para a codificação, a complexidade fica determinada pelo produto de matrizes, conforme a Eq. 4, *i.e.*, $\boldsymbol{\Theta(n^2 + n \cdot k)}$. [3]

[1]Resposta à pergunta 1 requisitada no roteiro desta atividade laboratorial.
[2]Resposta à pergunta 2a requisitada no roteiro desta atividade laboratorial.
[3]Resposta à pergunta 4a requisitada no roteiro desta atividade laboratorial.

Mais genericamente, pode-se enunciar tal método de extensão como:

- $\binom{k}{k}$ **bits de informação se relacionam com k bits de paridade**
- $\binom{k}{k-1}$ **bits de informação se relacionam com k-1 bits de paridade**
- **... até que todos os bits de informação tenham sido utilizados.**

Nesse sentido, vale salientar que adotando este método, é possível fazer a extensão para outros valores de tamanho de bloco. [4]

Para a decodificação, utilizando novamente o fato de que o erro unitário é muito maior que 2 ou mais erros, bastou-se percorrer as linhas da matriz H^t até achar a síndrome correspondente ao erro, garantindo complexidade $\mathrm{O}(\mathrm{n}\cdot\mathrm{k})$, pois para cada linha, cada bit é comparado.[5] Dessa forma, os casos de síndrome relacionados a 2 ou mais erros serão considerados improváveis e com probabilidade zero, de modo que o transmitido será igual ao recebido nesses casos. Isso pode parecer errado, mas é muito improvável de ocorrer, pois dependeria de p^w, com $w \geq 2$.

IV. Resultados e Discussão

De posse das implementações, foram obtidas as curvas de probabilidade de erro de bit de informação, em função do parâmetro p, que é a probabilidade de transmitir 0, receber 1 ou vice-versa. Foram adotados valores: $p = 0.5, 0.2, 0.1, 0.05, 0.02, 0.01, 0.005, ..., 2\cdot10^{-6}$. Assim, é trivial ver que $P_{erro} = p$, para um sistema não codificado. O resultado da simulação encontra-se na Figura 4.

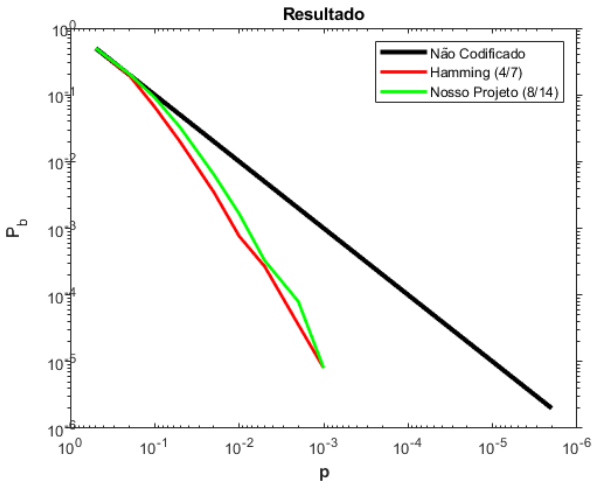

Fig. 4. Resultados da simulação, por meio do *software* MATLAB, em escala logarítmica para se observar melhor a comparação entre o codificador de Hamming (4/7) e o codificador particular (8/14).

Da figura 4, é possível observar, primeiramente que todas as probabilidades de erro de bit de transmissão caem, à medida que a probabilidade p diminui, o que já era esperado conforme a Eq. 10. Além disso, é possível também observar que em geral, a probabilidade de erro de transmissão é menor para a correspondente ao Hamming 4/7 (Figura 4), **o que também pode ser apontado pela Eq. 10, mantendo p fixo: quanto maior $\mathrm{N} = \mathrm{n} + \mathrm{k}$, menor a probabilidade de acerto (Figura 1), maior será a probabilidade de erro, logo mais acima estará a curva gerada. De modo geral, quanto maior o tamanho do bloco transmitido, maior será a probabilidade de erro de bit de informação e, portanto, menor será o desempenho do sistema de comunicação como um todo.** [6]

Para entender melhor o comportamento da probabilidade de erro, também implementou-se, de modo extra, um codificador/decodificador de Hamming 11/15. Note que $2^k - 1 = n + k$, com $n = 11$ e $k = 4$. O resultado da simulação com todas as codificações implementadas pode ser observada na Figura 5.

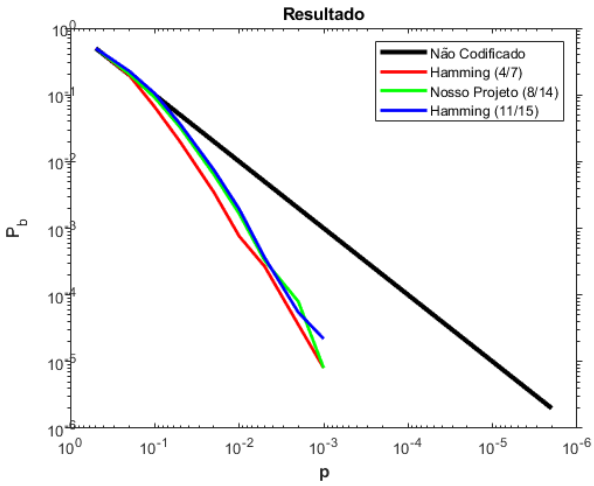

Fig. 5. Resultados da simulação, por meio do *software* MATLAB, em escala logarítmica para se observar melhor a comparação entre os codificadores implementados.

Como esperado, códigos com $N = n + k$ maiores ficaram mais próximos da reta que representa a não-codificação. As curvas mais próximas desta reta, apesar de, em geral, maiores taxas, possuem menores desempenhos.

V. Conclusão

Visando entender mais sobre a codificação de blocos de informações e seus efeitos em sistemas de comunicação, foram implementados e comparados codificadores/decodificadores, tanto de Hamming 4/7 e 11/15, quanto um codificador particular com taxa 8/14. Assim, como visto no gráfico representado na Figura 5, quanto maior o tamanho do bloco, maior a probabilidade de erro e, consequentemente, menor será o desempenho, embora maior taxa. Por isso, em ordem de desempenho crescente, chegou-se em: 11/15, 8/14 e 4/7. Nesse contexto, saber escolher codificações, segundo esses parâmetros, é um questionamento importante a se fazer em telecomunicações. De fato, ao escolher menores taxas de bit, maior será a proteção e, portanto, mais elaborado e caro pode ser o sistema.

[4]Resposta à pergunta 2b requisitada no roteiro desta atividade laboratorial.
[5]Resposta à pergunta 4b requisitada no roteiro desta atividade laboratorial.
[6]Resposta à pergunta 3 requisitada no roteiro desta atividade laboratorial.