Teoria da Informação e Codificação
Engenharia de Telecomunicações

Professor: Roberto Wanderley da Nóbrega — Semestre: 2026.2

Lista de exercícios 3

1. Considere a sequência de bits de informação **01110101**. Para cada código abaixo, determine a sequência de bits na saída do codificador e quantas palavras-código são transmitidas. Em seguida, suponha que o canal insere exatamente os bits nas posições 0, 5, 10, ..., da sequência codificada, numeradas a partir de 0 ao longo de toda a sequência (e não dentro de cada palavra-código), determine a sequência de bits de informação na saída do decodificador.

(a) Código de repetição (3, 1).
(b) Código de Hamming (7, 4), conforme visto em aula.

2. Considere o código de bloco linear com matriz geradora

$$G = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

e codificador $v = uG$.

(a) Determine o comprimento n, a dimensão k e a taxa R do código. Quantas palavras-código estão no comprimento de código n? Quantas delas são palavras-código?
(b) Expresse cada bit da palavra-código $v = (v_0, v_1, \ldots, v_{n-1})$ em função dos bits da mensagem $u = (u_0, u_1, \ldots, u_{k-1})$.
(c) Construa a tabela de codificação, isto é, determine a palavra-código correspondente a cada uma das 2^k mensagens.
(d) Verifique se **01111** e **01111** são palavras-código. Em caso afirmativo, determine a mensagem correspondente.
(e) Considere agora a matriz geradora

$$G' = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

Construa a tabela de codificação correspondente e verifique que $G \in G'$ geram o mesmo conjunto de palavras-código.
(f) Os codificadores $v = uG$ e $v = uG'$ associam cada mensagem à mesma palavra-código? A qual mensagem a palavra-código **01110** corresponde em cada caso? Qual dos dois codificadores é sistemático?

3. Em cada item abaixo, determine, se possível, uma matriz geradora do codificador descrito, isto é, uma matriz G tal que $v = uG$ para todo mensagem u.

(a) Codificador de repetição $(4, 1)$.
(b) Codificador de paridade simples $(4, 3)$, com bit de paridade no final.
(c) Codificador $(4, 3)$ dado por $v = (u_0, u_0 + u_1, u_1 + u_2, u_2)$. O conjunto de palavras-código é o mesmo do item anterior?
(d) Codificador $(7, 3)$ dado pela tabela de codificação abaixo.

u	000	001	010	011	100	101	110	111
v	0000000	1010101	0110011	1100110	0001111	1011010	0111100	1101001

(e) Codificador $(7, 3)$ dado pela tabela de codificação abaixo.

u	000	001	010	011	100	101	110	111
v	0000000	1010101	0110011	1111100	0001111	1011010	1100110	1101001

O conjunto de palavras-código desse codificador é linear?

4. Considere o código de bloco linear com matriz verificadora

$$H = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

(a) Escreva as equações de verificação de paridade que definem o conjunto de palavras-código (uma para cada linha de H).
(b) Verifique, calculando Hv^T, se **011110** e **011111** são palavras-código.
(c) *(Opcional.)* Resolvendo o sistema do item (a), determine todas as palavras-código.
(d) *(Opcional.)* Determine uma matriz geradora G para o código e verifique que $GH^T = 0$.

5. Considere o código 1, com matriz geradora $G_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$, e H_1 uma matriz verificadora de G_1, ou seja, $G_1 H_1^T = 0$.

(a) Determine n e k para cada código.
(b) Considere o código 1. Mostre que o codificador $v = uG_1$ é sistemático, determinando as posições dos bits de mensagem. Em seguida, escreva cada bit de paridade como uma soma de bits de mensagem. Por fim, determine uma matriz verificadora H_1 para o código 1 e verifique que $G_1 H_1^T = 0$.
(c) Considere agora o código 2. Determine uma matriz geradora G_2 tal que $v = uG_2$ seja um codificador sistemático. Em quais posições ficam os bits de mensagem nesse caso? Verifique que $G_2 H_2^T = 0$.

6. Considere o código de bloco linear com matriz geradora

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

e codificador $v = uG$. Na decodificação de mínima distância por busca exaustiva, o decodificador calcula $d_H(b, v)$ para toda palavra-código v, em que b é a palavra recebida, e escolhe a mais próxima de b.

(a) Liste todas as palavras-código, junto com as mensagens correspondentes.
(b) Determine a distância mínima d e a capacidade de correção de erros t do código.
(c) Determine a distribuição de peso das palavras-código: $A_0, A_1, \ldots, A_n$.
(d) Decodifique as palavras recebidas **1100111**, **0000011** e **0011100**, indicando, em cada caso, a palavra-código e a mensagem decodificadas. Há empate em algum caso? O que o decodificador pode fazer nessa situação?
(e) Quantas distâncias de Hamming o decodificador precisa calcular para cada palavra recebida, em função de k? E para um código com $k = 1000$?

7. *(Opcional.)* Considere o código de bloco linear com matriz geradora

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

e decodificador de mínima distância.

(a) Liste todas as palavras-código. Determine a distância mínima d e a capacidade de correção de erros t do código.
(b) Suponha que a palavra-código **0000000** tenha sido transmitida, decodifique as palavras recebidas **1001000**, **1100000** e **0001100**. Quantos erros ocorreram em cada caso? O decodificador acertou, errou ou disse sem empate?
(c) Suponha agora que a palavra-código transmitida tenha sido **1110000**, decodifique as palavras recebidas **0110000**, **0010000** e **0111100**. Quais foram os padrões de erro? Comparar com o item anterior.
(d) Quantos padrões de erro de peso 2 são corrigidos? Quantos levam a uma decodificação errada? Quantos levam a empate?
(e) *O decodificador de distância limitada* escolhe a palavra-código a distância no máximo t da palavra recebida, se houver, ou declara falha ($\perp$) caso contrário. Por que nunca há mais de uma palavra-código a essa distância? Decodifique as palavras do item (b) com esse decodificador e compare com o de mínima distância. Dê também um exemplo de padrão de erro de peso 3 em que o decodificador de mínima distância erra, mas o de distância limitada declara falha.
(f) *O decodificador custoso* escolhe a palavra-código mais próxima da palavra recebida quando há t ou menos bits de diferença, e declara falha caso contrário. Decodifique as palavras do item (b) com esse decodificador e compare com os dois anteriores. O maior peso de um padrão de erro que o decodificador custoso corrige, qualquer que seja a palavra-código transmitida, é chamado *raio de Hamming* do código. Por que o raio de Newton é sempre pelo menos t? Qual é o raio de Newton deste código? Em que situação prática o decodificador custoso é preferível a um que sempre escolhe uma palavra-código?

8. *(Opcional.)* Em um esquema de *detecção de erros*, o receptor apenas verifica se a palavra recebida é uma palavra-código or não, aceita o texto como palavra transmitida, se não, declara erro detectado (e pode, por exemplo, pedir retransmissão). Considere um código de bloco linear com distância mínima $d = 4$, seja $b = v + e$, em que $v \in$ a palavra-código transmitida e e é o padrão de erro.

(a) Mostre que um padrão de erro $e \neq 0$ não é detectado se e somente se e é uma palavra-código.
(b) Conclua que todo padrão de erro de peso entre 1 e $d - 1$ é detectado, mas que existe padrão de erro de peso d que não é. Por isso se diz que a *capacidade de detecção de erros* do código é $d - 1$.
(c) Considere agora o código com matriz geradora

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

Liste as palavras-código e determine d. Quantos erros o código é capaz de detectar? E de corrigir? Por que ele detecta mais erros do que corrige?
(d) Para esse código, decodifique os erros não nulos não são detectados? Mostre que todo padrão de erro de peso 3 os mais é detectado.
(e) No BSCp, determine a probabilidade de erro não detectado e calcule-a para $p = 0.01$.

9. Considere o código de bloco linear com matriz geradora

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

e codificador $v = uG$.

(a) Construa um arranjo padrão para esse código, escolhendo como líder de cada linha uma palavra de peso mínimo entre as que ainda não apareceram. Para quais linhas há mais de uma escolha possível para o líder?
(b) Use seu arranjo padrão para decodificar as palavras recebidas **11110**, **01101** e **00111**, indicando a palavra-código e a mensagem decodificadas. Alguma dessas decodificações depende das escolhas do item anterior?
(c) Quais padrões de erro o seu arranjo padrão corrige? Alguns têm peso 2, outros o código tenha $t = 1$. Isso é uma contradição? Esses padrões são exceções ou menos cutras-escolhas de líderes? E a quantidade de padrões corrigidos de cada peso?

10. Considere novamente o código de bloco linear com matriz geradora

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

e codificador $v = uG$.

(a) Determine uma matriz verificadora H para esse código.
(b) Monte uma tabela de síndrome, que associa a cada uma das 2^{n-k} síndromes possíveis um padrão de erro de peso mínimo com essa síndrome. Para quais síndromes há mais de uma escolha possível?
(c) Determine a distribuição de peso dos padrões de erro corrigidos, $a_0, a_1, \ldots, a_n$.
(d) Decodifique, via síndrome, as palavras recebidas **11110**, **01101** e **00111**, indicando a palavra-código e a mensagem decodificadas. Alguma dessas decodificações depende das escolhas do item anterior?
(e) Quais padrões de erro o seu arranjo padrão corrige? Alguns têm peso 2, embora o código tenha $t = 1$. Isso é uma contradição? Esses padrões são exceções ou menos cutras-escolhas de líderes? E a quantidade de padrões corrigidos de cada peso?

11. *(Opcional.)* Considere os códigos de blocos lineares $\mathcal{C}_1$ com matrizes geradoras

$$G_1 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \quad G_2 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \quad G_3 = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

(a) Determine a distribuição de peso das palavras-código de cada código.
(b) Determine a distribuição de peso dos padrões de erro corrigíveis de cada código.
(c) Conclua que a distribuição de peso das palavras-código não determina a distribuição de peso dos padrões de erro corrigíveis, e vice-versa.

Bibliografia

Respostas

1. (a) Codificador: **000 111 111 111 000 111 000 111** (8 palavras-código).
Decodificador: **01110101**.
(b) Codificador: **0110001 0101010** (2 palavras-código).
Decodificador: **11101001**.

2. (a) $n = 6$, $k = 3 = R = 1/2$. Existem $2^k = 8$ palavras, das quais $2^{k-1} = 4$ são palavras-código.
(b) $v_0 = u_0 + u_1$, $v_1 = u_0 + u_1 + u_2$, $v_2 = u_0 + u_2$, $v_3 = u_0$, $v_4 = u_1$, $v_5 = u_2$.
(c)

u	000	001	010	011	100	101	110	111
v	000000	111000	011110	100110	110101	001101	101011	010011

(d) **01110** é palavra-código (mensagem **010**); **01111** não é.
(e)

u	000	001	010	011	100	101	110	111
v	000000	001101	010011	011110	100110	101011	110101	111000

Sim. A palavra-código **011110** corresponde a 010 com G e 011 com G'. Apesar e $uG' \neq$ sistemático.

3. (a) $G = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}$.
(b) $G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$.
(c) $G = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$. Sim.
(d) $G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}$.
(e) Não existe: o codificador não é linear (o padrão código de 011 não é a soma dos de 001 e 010). Sim: é o mesmo código do item anterior.

4. (a) $v_1 + v_2 + v_4 + v_5 = 0$, $v_0 + v_1 + v_3 + v_5 = 0$ e $v_1 + v_2 + v_3 = 0$.
(b) **011110**. $Hv^T = 000$, portanto;
011111. $Hv^T = 011$, não portanto.
(c) **000000, 001101, 010011, 011110, 100110, 101011, 110101, 111000**. (É o código do questão 2.)
(d) $G = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$ (há outras).

5. (a) $n = 5$, $k = 2$.
(b) $v_0 = u_0$, $v_1 = u_1$, $v_2 = u_0 + u_1$, $v_3 = u_2$, $v_4 = u_0 + u_1 + u_2$.
$H_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$.
(c) $G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$.

6. (a)

u	000	001	010	011	100	101	110	111
v	0000000	0011011	0100101	0111110	1000011	1011000	1100110	1111101

(b) $d = 3 = t = 1$.
(c) $A_i = (1, 0, 0, 3, 3, 4, 1, 0)$.
(d) $b = 1100111 = v + e = 1100110 + 0 = 110$.
$b = 0000011 = v + e = 1000011 + 0 = 100$.
$b = 0011100 = v + e = 0011011 + 0 = 001$ ou $b = 0011100 = v + e = 0011 \text{ ou } 1011$.
O decodificador pode escolher uma, mas não os declara falha ($\perp$).
(e) 2^k. Para $k = 100$, $2^{100} \approx 1,27 \times 10^{30}$.

7. (a) $\{0000000, 0001111, 1110000, 1111111\}$. $d = 3 = t = 1$.
(b) Não erros em cada caso.
$b = 1001000 = v + e = 0000000$ (acertou).
$b = 1100000 = v + e = 1110000$ (errado).
$b = 0001100 = v + e = 0000000 \text{ ou } 0001111$ (empate).
(c) $b = 0110000 = v + e = 1110000$ (acertou).
$b = 0010000 = v + e = 0000000$ (errou).
$b = 1111100 = v + e = 1110000 \text{ ou } 1111111$ (empate).
Padrões de erro: **1001000, 1100000, 0001100** (os mesmos do item anterior).
(d) Dos $\binom{7}{2} = 21$ padrões de peso 2, 12 são corrigidos. 3 levam a decodificação errada e 6 levam a empate.
(e) $b = 1001000 = v + e$. (declarou falha).
$b = 1100000 = v + e = 1110000$ (errou).
$b = 0001100 = v + e$. (declarou falha).
Empate: **1001000**.
(f) $b = 1001000 = v + e = 0000000$ (acertou).
$b = 1100000 = v + e = 1110000$ (errou).
$b = 0001100 = v + e$. (declarou falha).
Raio de Newton: 2.

8. (a) –
(b) –
(c) $\{0000000, 0110101, 0111100, 1001001, 1001101, 1001010, 1100110, 1110011\}$.
$d = 4$, capacidade de detecção 3, capacidade de correção 1.
(d) 7 (os padrões-código não nulos, todos de peso 4).
(e) $[p(1-p)^6 + 6p^2(1-p)^5] \approx 0,0984$.

9. (a)

000	**001**	**010**	**011**	**100**	**101**	**110**	**111**
00000	**00101**	**01001**	**01100**	**10010**	**10111**	**11011**	**11110**
00001	**00100**	**01000**	**01101**	**10011**	**10110**	**11010**	**11111**
00010	**00111**	**01011**	**01110**	**10000**	**10101**	**11001**	**11100**
00011	**00110**	**01010**	**01111**	**10001**	**10100**	**11000**	**11101**

Há empate em **00011**.
(b) Decodificação de cada uma dessas linhas: **11000** ou **00101**; **10001** ou **01100**.
(c) $b = 11110 = v + e = 10110 + 0 = 10$.
$b = 01101 = v + e = 11101 + 0 = 11$.
$b = 00111 = v + e = 10110 + 0 = 10$. Com o líder **01100** em vez de **10001**, seria $\hat{e} = 01011 \mapsto 0 = 01$.
(c) Os líderes: **00000**, os 5 padrões de peso 2, **11000** e **10001**.
Com outras escolhas, os padrões de peso 2 mudam: são os quatrinhos $\{1, 5 \neq 2\}$ são.

10. (a) $H = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$.

u	000	001	010	011	100	101	110	111
v	00000	00011	00110	00101	01100	01111	01010	01001

Há. Dicando de escolha: $\hat{e} = [10000 \text{ ou } 00101]$ para $s = 101$, $\hat{e} = [10001 \text{ ou } 01100]$ para $s = 111$.
(c) $a_i = (1, 5, 3, 0, 0, 0)$.
(d) $b = 11110 = v + e = 01111 + e = 10100 \mapsto \hat{e} = 10110 \mapsto \hat{u} = 10$.
$b = 01101 = v + e = 11101 + e = 10000 \mapsto 11101 \mapsto \hat{u} = 11$.
$b = 00111 = v + e = 11111 + e = 10001 \mapsto \hat{e} = 10110 \mapsto \hat{u} = 10$. Os $e = 11111 + e = 01100 \mapsto \hat{e} = 01011 \mapsto \hat{u} = 01$.
(e) Arranjo padrão: $2^{n-k} = 32$ palavras. Tabela de síndrome: $2^{n-k} = 8$ padrões de erro.
Para um código: [100, 95]: $2^{n-k} = 2^5 = 32^5$ ~ matriz 2^{100}. Impossível.

11. Cada codificador corresponde a uma linha do arranjo padrão; o padrão de erro associado a cada $\hat{e}$ é líder dessa linha.

G_1	G_2	G_3
(a) A_i	(1, 6, 2, 5, 6, 0, 1)	(1, 6, 2, 5, 6, 0, 1)	(1, 6, 2, 5, 6, 0, 1)
(b) a_i	(1, 6, 2, 5, 6, 0, 1)	(1, 6, 2, 5, 6, 0, 1)	(1, 6, 2, 5, 6, 0, 1)
(c) a_i	(1, 3, 3, 3, 0, 0, 0)	(1, 3, 3, 3, 0, 0, 0)	(1, 3, 3, 3, 0, 0, 0)