Universidade Estadual Paulista "Júlio de Mesquita Filho"
Instituto de Geociências e Ciências Exatas
Campus de Rio Claro

Aplicações de Álgebra Linear aos Códigos Corretores de Erros e ao Ensino Médio

Everton Rodrigo Nicoletti

Dissertação apresentada ao Programa de Pós-Graduação – Mestrado Profissional em Matemática em Rede Nacional-PROFMAT como requisito parcial para a obtenção do grau de Mestre

Orientadora
Profa. Dra. Carina Alves

2015

512.5 Nicoletti, Everton Rodrigo
N643a Aplicações de Álgebra Linear aos Códigos Corretores de Erros e ao Ensino Médio/ Everton Rodrigo Nicoletti- Rio Claro: [s.n.], 2015.
71 f., il., figs., tabs.

Dissertação (mestrado) - Universidade Estadual Paulista, Instituto de Geociências e Ciências Exatas.
Orientadora: Carina Alves

1. Álgebra Linear. 2. Matrizes. 3. Códigos lineares. 4. Códigos de Hamming. I. Título

Ficha Catalográfica elaborada pela STATI - Biblioteca da UNESP
Campus de Rio Claro/SP

TERMO DE APROVAÇÃO

Everton Rodrigo Nicoletti

Aplicações de Álgebra Linear aos Códigos Corretores de Erros e ao Ensino Médio

Dissertação APROVADA como requisito parcial para a obtenção do grau de Mestre no Curso de Pós-Graduação Mestrado Profissional em Matemática em Rede Nacional-PROFMAT do Instituto de Geociências e Ciências Exatas da Universidade Estadual Paulista "Júlio de Mesquita Filho", pela seguinte banca examinadora:

Profa. Dra. Carina Alves
Orientadora

Profa. Dra. Marta Cilene Gadotti
IGCE - Unesp

Profa. Dra. Cintya Wink de Oliveira Benedito
IMECC - Unicamp

Rio Claro, 24 de Fevereiro de 2015

À minha mãe.

Agradecimentos

A Deus que permitiu a conclusão deste trabalho.

Aos meus pais, em especial à minha mãe, que apesar de pouca instrução sempre me incentivou.

A Sociedade Brasileira de Matemática que proporcionou, através do PROFMAT - Mestrado Profissional em Matemática em Rede Nacional, a oportunidade de aprimorar minha formação profissional.

A CAPES pelo apoio financeiro.

Aos professores do Departamento de Matemática da Universidade Estadual Paulista - Unesp/Rio Claro que de alguma forma contribuíram com minha formação acadêmica.

A minha orientadora Profa. Dra. Carina Alves que dedicou seu tempo, compartilhando seus conhecimentos e toda sua experiência acadêmica, para o enriquecimento da minha dissertação.

Aos professores Flávio R. Gazola e Taciana Belluci, que colaboraram com a revisão ortográfica da Língua Portuguesa e Inglesa.

"O impulso para descobrir segredos está profundamente enraizado na natureza humana; mesmo a mente menos curiosa é estimulada pela perspectiva de compartilhar o conhecimento oculto aos outros. [...]
Histórias de detetive e palavras cruzadas divertem a maioria. Já a quebra de códigos secretos pode ser uma tarefa para poucos".

John Chadwick

Resumo

Este trabalho aborda conceitos básicos de Álgebra Linear e suas aplicações no desenvolvimento da Teoria de Códigos Corretores de Erros. O uso desta ferramenta matemática simplifica a geração e a decodificação dos códigos lineares. Destacamos também a importância de se trabalhar com este tema na educação básica.

Palavras-chave: Álgebra Linear, Matrizes, Códigos lineares, Códigos de Hamming.

Abstract

The present work addresses basic concepts of Linear Algebra and its applications in the development of the Theory of Error Correcting Codes. The use of this mathematical tool simplifies the generation and decoding of linear codes. This dissertation also highlights the importance of working with this subject in high school.

Keywords: Linear Algebra, Matrices, Linear codes, Hamming codes.

Lista de Figuras

Lista de Tabelas

Sumário

1 Introdução

Durante milhares de anos, houve a preocupação em manter informações secretas, como por exemplo, o Código de César, usado pelo imperador romano Júlio César, para proteger mensagens de significado militar. Em contrapartida, houve também o interesse em desvendar tais informações. A descoberta de mensagens sigilosas possibilitou o surgimento dos códigos, com os quais, através de diferentes estratégias, foi possível se comunicar em segredo. A partir do momento que um determinado código perde sua finalidade, faz-se necessário a criação de códigos mais complexos e difíceis de serem decifrados. Deste modo, a matemática moderna contribui, de forma eficaz, com a elaboração de técnicas para dificultar cada vez mais a decodificação desses códigos. Apesar de suas raízes no passado, a criação dos códigos está diretamente relacionada com o mundo atual, principalmente com a invenção do computador e os avanços tecnológicos.

Atualmente, com o grande volume de informações que circulam na Internet, é de extrema importância que esse processo de comunicação seja seguro, por exemplo, ao realizarmos transações financeiras, como compras com cartão de crédito e envio de senhas na rede.

O presente trabalho tem como objetivos: mostrar que a teoria dos códigos é um campo de pesquisa atual, muito atraente, tanto do ponto de vista científico quanto tecnológico; evidenciar as características e a importância de alguns códigos; aplicar, ao estudo da teoria dos códigos, conteúdos de Álgebra básica bem como os de Álgebra Linear, articulando conceitos e técnicas com aplicações imediatas no cotidiano, sempre com a finalidade de garantir segurança ao processo de comunicação; elaborar sequências didáticas com esse tema para serem aplicadas à nível de educação básica, mais precisamente no Ensino Médio, articulando o currículo com questões que envolvam diretamente a realidade dos alunos, como por exemplo, o acesso às redes sociais na Internet através de senhas pessoais e secretas.

Como toda codificação é passível de ser decodificada, os avanços alcançados neste sentido têm como finalidade garantir cada vez mais que a troca de informações seja feita sem a ocorrência de erros e da forma mais segura possível. É neste aspecto que a aplicação de teorias matemáticas, à luz das teorias da Álgebra Linear, possibilita a compreensão da complexa decodificação de mensagens secretas.

Segundo Tamarozzi (2001), citado por Groenwald (2011, p.2), "o estudo dos códigos

possibilita o desenvolvimento de atividades didáticas envolvendo o conteúdo de matrizes que se constitui em material útil para exercícios, atividades e jogos de codificação, onde o professor pode utilizá-los para fixação de conteúdos". Além de fixá-los, as atividades aqui propostas possibilitam a construção do raciocínio, a partir do conhecimento prévio dos alunos, para compreender o conceito de matriz inversa.

Para Shokranian (2005), citado por Groenwald (2010, p.2), "enviar uma mensagem em código pode servir para dois objetivos, que são: enviar uma mensagem secreta e proteger o conteúdo da mensagem contra pessoas não autorizadas". Neste sentido, os alunos estarão desenvolvendo estratégias para quebrar um código e também para tornar a decodificação de seu código mais difícil.

As situações problema são articuladas ao cotidiano dos alunos tendo em vista sua aplicação nos processos tecnológicos que norteiam os sistemas de comunicação digital.

De acordo com o Currículo do Estado de São Paulo - Matemática e suas Tecnologias (2011, p. 21-22),

> "A educação tecnológica básica é uma das diretrizes que a Lei de Diretrizes e Bases da Educação Nacional - LDBEN estabelece para orientar o currículo do Ensino Médio. A lei ainda associa a "compreensão dos fundamentos científicos dos processos produtivos" ao relacionamento entre teoria e prática em cada disciplina do currículo. E insiste quando insere o "domínio dos princípios científicos e tecnológicos que presidem a produção moderna" entre as competências que o aluno deve demonstrar ao final da educação básica. [...] a compreensão dos fundamentos científicos e tecnológicos da produção, faz da tecnologia a chave para relacionar o currículo ao mundo da produção de bens e serviços, isto é, aos processos pelos quais a humanidade – e cada um de nós – produz os bens e serviços de que necessita para viver".

Mais especificamente, este trabalho está estruturado como segue:

No Capítulo 2, apresentamos algumas ferramentas matemáticas que serão aplicadas ao estudo da teoria dos códigos.

No Capítulo 3, apresentamos as principais características de um sistema de comunicação bem como seus elementos. Destacamos também a contribuição de Shannon para essa teoria. Definimos ainda, codificação, decodificação e códigos de bloco.

No Capítulo 4, definimos códigos que possuem a capacidade de detectar e corrigir erros de acordo com sua característica. Um código corretor de erros visa recuperar informações que, durante o processo de transmissão, tenham sofrido algum tipo de ruído. Pode-se afirmar que, hoje praticamente todo sistema de comunicação possui algum tipo de código corretor de erros, por exemplo, a telefonia digital, a transmissão de dados via satélite, a comunicação interna em computadores, armazenamento óptico de dados e armazenamento de dados em HD, pen drive e blu-ray. Enfatizamos, neste aspecto, os códigos de Hamming que serão utilizados como ferramentas de detecção e correção de erros ocorridos na transmissão de informações.

No Capítulo 5, abordamos o estudo dos códigos como um espaço vetorial, dando destaque às representações matriciais, inclusive para os códigos de Hamming.

No Capítulo 6, apresentamos três sequências didáticas relacionadas ao tema abordado, para serem utilizadas como proposta de atividades no Ensino Médio, proporcionando um estudo contextualizado sobre multiplicação de matrizes e matriz inversa.

No Capítulo 7, encontram-se as considerações finais, destacando a relevância de estudos com o tema aqui abordado e também sua articulação para o desenvolvimento de atividades de nível médio nas escolas de educação básica.

Por fim, as referências bibliográficas que foram utilizadas para o enriquecimento deste trabalho.

2 Preliminares

A Álgebra Linear é a área da Matemática que estuda todos os aspectos relacionados com um conjunto chamado Espaço Vetorial. Nele, são definidas operações e as propriedades relacionadas à essas operações dão estrutura a esse conjunto.

Neste capítulo abordaremos as principais definições e resultados da Álgebra Linear, que serão utilizados como ferramentas para o estudo da teoria dos códigos lineares nos capítulos 5 e 6. Os itens [1], [2], [3] e [4] das referências foram utilizados para o desenvolvimento deste capítulo.

2.1 Anel e Corpo

Nesta seção definiremos anel e corpo, que serão ferramentas para o desenvolvimento das próximas seções.

Definição 2.1. *Anel é um conjunto não vazio R munido de duas operações, soma $(+)$ e multiplicação $(\cdot)$ em R, em que são satisfeitas as seguintes propriedades, para quaisquer x, y, $z \in R$:*

- *comutatividade para a soma:* $x + y = y + x$.
- *associatividade para a soma:* $(x + y) + z = x + (y + z)$.
- *elemento neutro para soma, ou seja, existe* $0 \in A$ *tal que* $x + 0 = 0 + x = x$.
- *elemento oposto para a soma, isto é, existe* $(-x) \in A$ *tal que* $x + (-x) = (-x) + x = 0$.
- *associatividade para a multiplicação:* $(x \cdot y) \cdot z = x \cdot (y \cdot z)$.
- *distributividade para multiplicação em relação à soma:* $x \cdot (y + z) = x \cdot y + x \cdot z$ *e* $(y + z) \cdot x = y \cdot x + z \cdot x$.

Observação 2.1. Quando para todo x, $y \in R$ tem-se, $x \cdot y = y \cdot x$, dizemos que o anel é comutativo.

Observação 2.2. Quando para todo $x \in R$, existe $1 \in R$ tal que $x \cdot 1 = 1 \cdot x = x$, dizemos que o conjunto R é um anel com unidade.

Exemplo 2.1. O conjunto $\mathbb{Z}_2 = \{0, 1\}$ é um anel, em que $\mathbb{Z}_2$ é o quociente $\frac{\mathbb{Z}_2}{\sim}$, onde $\sim$ é a congruência módulo 2, com as operações soma $(+)$ e multiplicação $(\cdot)$ em $\mathbb{Z}_2$.

$+$	**0**	**1**
0	0	1
1	1	0

$.$	**0**	**1**
0	0	0
1	0	1

Tabela 2.1: Soma e multiplicação em $\mathbb{Z}_2$

Exemplo 2.2. O conjunto dos números inteiros, racionais, reais e complexos são anéis.

Definição 2.2. *Um corpo $\mathbb{F}$ é um anel comutativo com unidade, onde todo elemento não nulo possui inverso multiplicativo.*

Exemplo 2.3. O conjunto dos números racionais, reais e complexos são corpos.

Observação 2.3. Se o corpo $\mathbb{F}$ possuir número finito de elementos, dizemos que ele é um corpo finito, denotado por $\mathbb{F}_q$, onde q é a quantidade de elementos do corpo.

Exemplo 2.4. O conjunto $\mathbb{Z}_p = \{0, 1, \ldots p-1\}$, para todo p primo é um corpo finito.

Definição 2.3. *A característica de um corpo $\mathbb{F}$ é o menor número inteiro positivo m tal que, sendo $a \in \mathbb{F}$, $ma = \underbrace{a + a + a + \cdots + a}_{m\,vezes} = 0$. Se não existir tal número m, a característica do corpo é definida como zero.*

2.2 Matrizes e Sistemas Lineares

Nesta seção abordaremos alguns tópicos referentes ao estudo de matrizes: matriz quadrada, matriz identidade, matriz transposta, multiplicação de matrizes e matriz inversa. Também definiremos um sistema linear e apresentaremos a sua representação matricial.

Definição 2.4. *Uma matriz A de ordem $m \times n$ é uma tabela com $m \cdot n$ elementos, distribuídos em m linhas e n colunas.*

Observação 2.4. Uma matriz A de ordem $m \times n$ também pode ser denotada por $A = (a_{ij})_{m \times n}$, onde $1 \leqslant i \leqslant m$ e $1 \leqslant j \leqslant n$. Cada elemento $a_{ij} \in \mathbb{F}$ é representado pela sua posição na matriz, ou seja, é o elemento que está na i-ésima linha e na j-ésima coluna da matriz.

Exemplo 2.5. Matriz A de ordem 2×3:

$$A = \begin{pmatrix} 5 & 0 & -2 \\ 2 & 1 & 3 \end{pmatrix}.$$

O elemento que está está na primeira linha e terceira coluna é o -2, isto é, $a_{13} = -2$.

Observação 2.5. O conjunto de todas as matrizes de ordem $m \times n$ com elementos em um corpo $\mathbb{F}$ será representado por $\mathbb{M}_{m\times n}(\mathbb{F})$.

Definição 2.5. *Uma matriz A é uma matriz quadrada quando o número de linhas é igual ao número de colunas, ou seja, $m = n$.*

Exemplo 2.6. $A = \begin{pmatrix} 1 & 2 \\ 0 & 3 \end{pmatrix}$ é uma matriz quadrada de ordem 2.

Definição 2.6. *A matriz identidade de ordem n, denotada por $I_n = (a_{ij})_{n\times n}$ é dada por*

$$a_{ij} = \begin{cases} 1, se\ i = j \\ 0, se\ i \neq j \end{cases}.$$

Exemplo 2.7. Matriz identidade de ordem 3:

$$I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Definição 2.7. *Seja $A = (a_{ij})$ uma matriz de ordem $m \times n$. A matriz $A^t = (b_{ij})$ de ordem $n \times m$ cujas linhas são as colunas de A, isto é, $b_{ij} = a_{ji}$ para todo i, j com $1 \leqslant i \leqslant m$ e $1 \leqslant j \leqslant n$ é chamada transposta de A.*

Exemplo 2.8. Dada uma matriz A de ordem 2×3:

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 4 & 1 & 3 \end{pmatrix}.$$

A matriz transposta de A de ordem 3×2 é dada por

$$A^t = \begin{pmatrix} 1 & 4 \\ 2 & 1 \\ 0 & 3 \end{pmatrix}.$$

Definição 2.8. *Sejam $A = (a_{ik})$ uma matriz de ordem $m \times p$ e $B = (b_{kj})$ uma matriz de ordem $p \times n$. A matriz produto de A por B é a matriz $AB = (c_{ij})$ de ordem $m \times n$ tal que*

$$c_{ij} = \sum_{k=1}^{p} a_{ik} \cdot b_{kj},\ i = 1, ..., m;\ j = 1, ..., n.$$

Observação 2.6. Só podemos efetuar o produto AB se o número de colunas da matriz A for igual ao número de linhas da matriz B.

Observação 2.7. O elemento c_{ij} da matriz AB é obtido da soma dos produtos dos elementos da i-ésima linha da matriz A pelos elementos correspondentes da j-ésima coluna da matriz B.

Exemplo 2.9. Dadas as seguintes matrizes A de ordem 1×2 e B de ordem 2×2:

$$A = \begin{pmatrix} 1 & 0 \end{pmatrix} \qquad B = \begin{pmatrix} 2 & 3 \\ 0 & 1 \end{pmatrix}$$

A matriz produto AB de ordem 1×2 é dado por

$$AB = \begin{pmatrix} 1 \cdot 2 + 0 \cdot 0 & 1 \cdot 3 + 0 \cdot 1 \end{pmatrix} = \begin{pmatrix} 2 & 3 \end{pmatrix}.$$

Observação 2.8. O produto de duas matrizes não é comutativo, ou seja, nem sempre teremos $AB = BA$.

Exemplo 2.10. Dadas as matrizes A e B de ordem 2:

$$A = \begin{pmatrix} 1 & 2 \\ 1 & 0 \end{pmatrix} \qquad B = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$$

A matriz produto AB de ordem 2 é

$$AB = \begin{pmatrix} 4 & 7 \\ 1 & 3 \end{pmatrix}$$

A matriz produto BA de ordem 2 é

$$BA = \begin{pmatrix} 2 & 5 \\ 1 & 5 \end{pmatrix}$$

Logo, as matrizes AB e BA não são iguais.

Definição 2.9. *Dada uma matriz A de ordem n. Se existir uma matriz B de ordem n tal que $AB = BA = I_n$, a matriz A é dita invertível e a matriz B é a sua inversa. A matriz inversa de A é denotada por A^{-1}, ou seja, $B = A^{-1}$.*

Exemplo 2.11. A matriz $A^{-1} = \begin{pmatrix} 3 & -5 \\ -1 & 2 \end{pmatrix}$ é a inversa da matriz $A = \begin{pmatrix} 2 & 5 \\ 1 & 3 \end{pmatrix}$, pois $AA^{-1} = A^{-1}A = I_2$.

Definição 2.10. *Um sistema de equações lineares com m equações e n incógnitas é um conjunto de equações do tipo:*

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n = b_m \end{cases}$$

onde $a_{ij}, b_i \in \mathbb{F}$, *com* $1 \leqslant i \leqslant m$ *e* $1 \leqslant j \leqslant n$.

Uma solução desse sistema é uma n-upla $(x_1, x_2, ..., x_n)$ em $\mathbb{F} \times \cdots \times \mathbb{F} = \mathbb{F}^n$ que satisfaça simultaneamente todas as m equações.

O sistema acima pode ser escrito na forma matricial, assim:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}.$$

Deste modo, todo sistema de equações lineares pode ser escrito na forma $A \cdot X = B$, onde A é a matriz dos coeficientes, X é a matriz das incógnitas e B é a matriz dos termos independentes.

Exemplo 2.12. Sistema de equações lineares com $a_{ij}, b_i \in \mathbb{Z}_2$:

$$\begin{cases} x_2 + x_4 + x_5 = 0 \\ x_1 + x_3 + x_4 = 0 \\ x_1 + x_2 + x_3 + x_5 = 0 \end{cases}$$

2.3 Espaço e Subespaço Vetorial

Nesta seção definiremos espaço vetorial e subespaço vetorial e abordaremos algumas características relacionadas a esses conceitos.

Definição 2.11. *Um conjunto não vazio* V *é um espaço vetorial sobre um corpo* $\mathbb{F}$ *se para quaisquer vetores* u *e* $v \in V$ *e um escalar* $\alpha \in \mathbb{F}$ *são válidos:*

- $u + v \in V$;
- $\alpha \cdot v \in V$.

E para todo u, v, $w \in V$ e α, $\beta \in \mathbb{F}$ temos:

- $u + v = v + u$.
- $(u + v) + w = u + (v + w)$.
- Existe $0 \in V$ tal que $u + 0 = u$, onde 0 é o vetor nulo.
- Existe $(-u) \in V$ tal que $u + (-u) = 0$.
- $\alpha \cdot (u + v) = \alpha \cdot u + \alpha \cdot v$.
- $(\alpha + \beta) \cdot v = \alpha \cdot v + \beta \cdot v$.
- $(\alpha \cdot \beta) \cdot v = \alpha \cdot (\beta \cdot v)$.

- $1 \cdot u = u$

Exemplo 2.13. Todo corpo $\mathbb{F}$ é um espaço vetorial sobre si mesmo.

Exemplo 2.14. $(\mathbb{Z}_2)^n$, $n \in \mathbb{N}$ é um espaço vetorial sobre $\mathbb{Z}_2$.

Exemplo 2.15. O conjunto $\mathbb{M}_{m\times n}(\mathbb{F})$ das matrizes $m \times n$ com coeficientes em $\mathbb{F}$ é um espaço vetorial sobre $\mathbb{F}$.

Definição 2.12. *Considere um espaço vetorial V. Um subconjunto S de V é um subespaço vetorial de V se S for um espaço vetorial com respeito às mesmas operações que tornam V um espaço vetorial.*

Para identificar subespaços tomaremos como critério o resultado a seguir.

Teorema 2.1. *$S \subset V$ é um subespaço vetorial de V se, e somente se,*
(i) S é não vazio,
(ii) S é fechado sob adição de vetores, isto é, $u, v \in S$ implica

$$u + v \in S.$$

(iii) S é fechado sob multiplicação por escalar: $u \in S$ implica

$$\alpha \cdot u \in S, \text{ para cada } \alpha \in \mathbb{F}.$$

Corolário 2.1. *S é um subespaço de V se, e somente se,*
(i) $0 \in S$ ou $(S \neq \varnothing)$,
(ii) $u, v \in S$ implica

$$\alpha \cdot u + \beta \cdot v \in S, \text{ para todo } \alpha, \beta \in \mathbb{F}.$$

As demonstrações destes resultados podem ser encontradas em [4].

Exemplo 2.16. Se V é um espaço vetorial qualquer, então o conjunto $\{0\}$ constituído somente do vetor nulo e também o espaço todo V são subespaços de V.

Exemplo 2.17. Seja o espaço vetorial $\mathbb{R}^3$. O conjunto S constituído dos vetores cuja terceira componente é zero, $S = \{(a, b, 0) : a, b \in \mathbb{R}\}$ é um subespaço de $\mathbb{R}^3$.

2.3.1 Bases e Dimensão

Nesta subseção definiremos combinação linear, dependência e independência linear e também base e dimensão de um espaço vetorial V.

O conceito de dependência e independência linear desempenha um papel essencial na teoria da Álgebra Linear e na Matemática em geral.

Definição 2.13. *Um vetor $v \in V$ é uma combinação linear dos vetores $v_1, ..., v_n \in V$ se existirem escalares $\alpha_1, ..., \alpha_n \in \mathbb{F}$ tais que*

$$v = \alpha_1 v_1 + ... + \alpha_n v_n.$$

Definição 2.14. *Seja V um espaço vetorial sobre um corpo $\mathbb{F}$. Diz-se que os vetores $v_1, ..., v_n \in V$ são linearmente dependentes sobre $\mathbb{F}$, se existem escalares $\alpha_1, ..., \alpha_n \in \mathbb{F}$, nem todos nulos, tais que*

$$\alpha_1 v_1 + ... + \alpha_n v_n = 0.$$

Caso contrário, diz-se que os vetores são linearmente independentes (LI) sobre $\mathbb{F}$.

Observe que se 0 é um dos vetores $v_1, ..., v_n$, digamos $v_1 = 0$, então os vetores devem ser linearmente dependentes, pois

$$1v_1 + 0v_2 + ... + 0v_n = 1 \cdot 0 + 0 + ... + 0 = 0,$$

Por outro lado, qualquer vetor não nulo v é, por si só, linearmente independente, pois

$$\alpha \cdot v = 0, v \neq 0 \text{ implica } \alpha = 0.$$

Exemplo 2.18. Os vetores $u = (1, -1, 0)$, $v = (1, 3, -1)$ e $w = (5, 3, -2)$ de $\mathbb{R}^3$ são linearmente dependentes pois $3u + 2v - w = 3(1, -1, 0) + 2(1, 3, -1) - (5, 3, -2) = (0, 0, 0)$.

Exemplo 2.19. Os vetores $v_1, v_2 \in (\mathbb{Z}_2)^2$ tais que $v_1 = (1, 0)$ e $v_2 = (0, 1)$ são linearmente independentes.

De fato, sejam $\alpha_1, \alpha_2 \in \mathbb{Z}_2$, tais que $\alpha_1 v_1 + \alpha_2 v_2 = (0, 0)$. Logo,$(\alpha_1, \alpha_2) = (0, 0)$ e portanto $\alpha_1 = \alpha_2 = 0$.

Definição 2.15. *Sejam V um espaço vetorial sobre $\mathbb{F}$ e $\mathcal{B}$ um subconjunto de V. Dizemos que o espaço vetorial V é finitamente gerado se todo vetor de V pode ser obtido como combinação linear dos vetores de $\mathcal{B}$. Neste caso, denotamos $V = [\mathcal{B}]$, isto é, V é gerado por $\mathcal{B}$.*

Exemplo 2.20. O espaço vetorial $\mathbb{R}^3$ é finitamente gerado, isto é,

$\mathbb{R}^3 = [(1, 0, 0), (0, 1, 0), (0, 0, 1)]$.

Definição 2.16. *Seja V um espaço vetorial. Um subconjunto de vetores $\mathcal{B} = \{v_1, ..., v_n\}$ é uma base de V se*

(i) $\mathcal{B}$ é um conjunto linearmente independente.

(ii) O espaço gerado por $\mathcal{B}$, isto é, o conjunto de todas as combinações lineares de $\{v_1, ..., v_n\}$ é V.

Se o espaço vetorial V for finitamente gerado, então o número de elementos de uma base de um espaço vetorial V é a dimensão desse espaço e denotada por $dimV$. Caso contrário, dizemos que V tem dimensão infinita.

Observação 2.9. O subespaço $U = \{0\}$ de V tem dimensão nula, isto é, $dimU = 0$.

Exemplo 2.21. O conjunto $\{(1,0),(0,1)\}$ é uma base de $\mathbb{R}^2$ e $dim\mathbb{R}^2 = 2$.

O conjunto $\{(1,0,0),(0,1,0),(0,0,1)\}$ é uma base de $\mathbb{R}^3$ e $dim\mathbb{R}^3 = 3$.

Proposição 2.1. *Seja V um espaço vetorial sobre $\mathbb{F}$ de dimensão finita $n \geqslant 1$ e seja $\mathcal{B} \subseteq V$. As seguintes afirmações são equivalentes:*

a) $\mathcal{B}$ é uma base de V;

b) Cada elemento de V se escreve de maneira única como combinação linear de elementos de $\mathcal{B}$.

Demonstração. a)⇒ b). Vamos supor que $\mathcal{B} = \{v_1, ..., v_n\}$ seja uma base de V. Por definição, $\mathcal{B}$ gera V e, portanto, todo elemento de V se escreve como combinação linear de $v_1, ..., v_n$. Para mostrar a unicidade, suponha que $v = \alpha_1 v_1 + ... + \alpha_n v_n$ e $v = \beta_1 v_1 + ... + \beta_n v_n$, onde $alpha_i, \beta_1 \in \mathbb{F}$, com $1 \leqslant i \leqslant n$. Então,

$$\alpha_1 v_1 + ... + \alpha_n v_n = \beta_1 v_1 + ... + \beta_n v_n \Leftrightarrow$$
$$(\alpha_1 - \beta_1)v_1 + ... + (\alpha_n - \beta_n)v_n = 0.$$

Como $\mathcal{B}$ é um conjunto linearmente independente, segue que $\alpha_i - \beta_i = 0$, para todo $i = 1, ..., n$. Logo, $\alpha_i = \beta_i$ para todo i. Portanto, v se escreve de maneira única como combinação linear de elementos de $\mathcal{B}$.

b)⇒ a). Considere que cada elemento de V se escreve de maneira única como combinação linear de elementos de $\mathcal{B}$. Por definição, $\mathcal{B}$ gera V. Para que $\mathcal{B}$ seja uma base, esse conjunto deve ser linearmente independente. Sejam $v_1, ..., v_n \in \mathcal{B}$ e $\lambda_1, ..., \lambda_n \in \mathbb{F}$ tais que $\lambda_1 v_1 + ... + \lambda_n v_n = 0$. Como $0v_1 + ... + 0v_n = 0$, segue da condição de unicidade que $\lambda_i = 0$ para todo $i = 1, ..., n$. Portanto, $\mathcal{B}$ é uma base. □

2.3.2 Produto interno, Complemento Ortogonal e Soma Direta

Nesta subseção definiremos produto interno, o qual será aplicado no conceito de complemento ortogonal e também abordaremos a definição de soma direta.

Definição 2.17. *Seja V um espaço vetorial sobre um corpo $\mathbb{F}$. Um produto interno sobre V é uma função que a cada par de vetores u e $v \in V$ associa um escalar em $\mathbb{F}$, denotado por $< u, v >$ e satisfazendo as seguintes propriedades para todos u, v e $w \in V$ e qualquer $\alpha \in \mathbb{F}$:*

(i) $< u, u > \geqslant 0$ e $< u, u >= 0$ se, e somente se $u = 0$.

(ii) $< u, v >=< v, u >$.

(iii) $< u, v + w >=< u, v > + < u, w >$.

(iv) $< \alpha u, v >= \alpha < u, v >$.

Observação 2.10. No caso de $\mathbb{F} = \mathbb{C}$ a propriedade (ii) é substituída por $< u, v >= \overline{< v, u >}$, onde $\overline{< v, u >}$ representa o conjugado de $< v, u >$.

Exemplo 2.22. Sejam os vetores $u = (u_1, ..., u_n)$ e $v = (v_1, ..., v_n) \in \mathbb{R}^n$. A função

$$< u, v >= u_1v_1 + ... + u_nv_n$$

é um produto interno sobre $\mathbb{R}^n$.

Observação 2.11. Dois vetores u e $v \in V$ são *ortogonais* e neste caso denotamos por $u \perp v$ se, e somente se, $< u, v >= 0$.

Exemplo 2.23. Sejam os vetores $u = (x, 0)$ e $v = (0, y) \in \mathbb{R}^2$, para quaisquer x e $y \in \mathbb{R}$. Então,

$$< u, v >= x \cdot 0 + 0 \cdot y = 0.$$

Logo, u e v são ortogonais.

Definição 2.18. *Sejam V um espaço vetorial e $U \subset V$ um subespaço vetorial de V. Vamos representar por $U^\perp$ o subconjunto formado pelos vetores de V que são ortogonais a todo vetor de U, isto é:*

$$U^\perp = \{v \in V; < v, u >= 0, \forall u \in U\}.$$

O subconjunto $U^\perp$ é chamado complemento ortogonal de U e é também um subespaço vetorial de V .

Exemplo 2.24. Sejam o espaço vetorial $\mathbb{R}^2$ e o subespaço vetor $U = \{(x, 0), x \in \mathbb{R}\}$. O complemento ortogonal $U^\perp$ do subespaço U é o subespaço

$$U^\perp = \{(0, y), y \in \mathbb{R}\}.$$

Definição 2.19. *Sejam U e V subespaços vetoriais de W. A soma de U e V, denotada $U + V$, consiste em todas as somas $u + v$ tais que $u \in U$ e $v \in V$, isto é*

$$U + V = \{u + v; u \in U \text{ e } v \in V\}.$$

Teorema 2.2. *Se U e V são subespaços vetoriais de W, então $U + V$ é subespaço de W.*

Demonstração. O conjunto $U + V$ é não vazio. De fato, U e V são não vazios. Em particular, $0 \in U$ e $0 \in V$. Logo, $0 = 0 + 0 \in U + V$.

Mostremos que o conjunto $U + V$ é fechado para a soma de vetores. Sejam $x_1, x_2 \in U + V$, tais que $x_1 = u_1 + v_1$ e $x_2 = u_2 + v_2$, para alguns $u_1, u_2 \in U$ e $v_1, v_2 \in V$. Logo,

$$x_1 + x_2 = (u_1 + v_1) + (u_2 + v_2) = (u_1 + u_2) + (v_1 + v_2).$$

Como $u_1 + u_2 \in U$ e $v_1 + v_2 \in V$, segue que $x_1 + x_2 \in U + V$.

Mostremos que o conjunto $U + V$ é fechado para o produto por escalar. Seja $x = u + v \in U + V$, tais que $u \in U$ e $v \in V$. Logo,

$$\alpha x = \alpha(u + v) = \alpha u + \alpha v, \text{ para todo } \alpha \in \mathbb{F}.$$

Como $\alpha u \in U$ e $\alpha v \in V$, segue que $\alpha x \in U + V$. □

Definição 2.20. *Dizemos que o espaço vetorial W é a soma direta dos subespaços vetoriais U e V, denotada $W = U \oplus V$, se todo vetor $w \in W$ pode ser escrito de modo único como $w = u + v$ tais que $u \in U$ e $v \in V$.*

Teorema 2.3. *Sejam U e V subespaços vetoriais de W. Dizemos que $W = U \oplus V$ é a soma direta de U e V, se e somente se, (i) $W = U + V$. (ii) $U \cap V = \{0\}$.*

Demonstração. Suponhamos $W = U \oplus V$. Todo vetor $w \in W$ pode ser escrito de modo único como $w = u + v$ tais que $u \in U$ e $v \in V$. Assim, em particular, $W = U + V$. Suponhamos agora que $w \in U \cap V$. Então,

$$w = w + 0, \text{ tal que } w \in U \text{ e } 0 \in V \text{ e}$$
$$w = 0 + w, \text{ tal que } 0 \in U \text{ e } w \in V.$$

Como tal soma para w deve ser única, segue que $w = 0$. Portanto, $U \cap V = \{0\}$. Por outro lado, suponhamos que $W = U + V$ e $U \cap V = \{0\}$. Dado $w \in W$, existem $u \in U$ e $v \in V$, tais que $w = u + v$. Suponha que exista outra decomposição $w = u' + v'$, com $u' \in U$ e $v' \in V$. Assim,

$$u + v = u' + v'$$
$$(u - u') + (v - v') = 0$$
$$u - u' = v - v'.$$

Mas, $u - u' \in U$ e $v - v' \in V$. Por hipótese, $U \cap V = \{0\}$. Logo,

$$u - u' = v - v' = 0$$
$$u = u' \text{ e } v = v'.$$

Portanto, a decomposição é única e $W = U \oplus V$. □

Observação 2.12. Se $U \oplus V = W$, então dizemos que U e V são complementares.

Exemplo 2.25. Sejam U e $U^{\perp}$ subespaços de $\mathbb{R}^3$ definidos assim:

$$U = \{(0, 0, z); z \in \mathbb{R}\}$$
$$U^{\perp} = \{(x, y, 0); x, y \in \mathbb{R}\}.$$

Como $\mathbb{R}^3 = U + U^{\perp}$ e $U \cap U^{\perp} = \{(0, 0, 0)\}$, conclui-se que $\mathbb{R}^3 = U \oplus U^{\perp}$.

2.4 Transformações Lineares e Operador Linear

Nesta seção definiremos transformação linear, núcleo e imagem de uma transformação linear, transformação injetora e operador linear, estes terão aplicação direta no estudo dos códigos de bloco lineares.

Definição 2.21. *Sejam V e W dois espaços vetoriais sobre $\mathbb{F}$. Uma transformação $T : V \to W$ é linear se satisfaz as seguintes condições:*
i) $T(u+v) = T(u) + T(v), \forall u, v \in V$.
ii) $T(\alpha v) = \alpha T(v), \forall \alpha \in \mathbb{F}$ *e* $v \in V$.

Exemplo 2.26. Seja $T : \mathbb{R} \to \mathbb{R}$. A transformação $T(u) = ku$, onde k é uma constante fixada é linear.

Exemplo 2.27. Seja $T : \mathbb{R}^2 \to \mathbb{R}^3$. A transformação $T(x, y) = (2x, 0, x + y)$ é linear.

Exemplo 2.28. Seja $T : V \to V$. A transformação nula, $T(v) = 0$ é linear.

Exemplo 2.29. Sejam $T_A : \mathbb{R}^n \to \mathbb{R}^m$ e A uma matriz de ordem $m \times n$. A transformação $T_A(v) = Av$, onde v é um vetor coluna $n \times 1$, é linear.

Definição 2.22. *Seja $T : V \to W$ uma transformação linear. O subconjunto de V formado por todos os vetores $v \in V$ tais que $T(v) = 0$ é chamado núcleo da transformação e é representado por $N(T)$, isto é*

$$N(T) = \{v \in V; T(v) = 0\}.$$

O subconjunto $N(T)$ é um subespaço vetorial de V.

Definição 2.23. *Seja $T : V \to W$ uma transformação linear. A imagem de T é o subconjunto de W formado pelos vetores $w \in W$ tais que existe um vetor $v \in V$ que satisfaz $T(v) = w$, ou seja,*

$$Im(T) = \{w \in W; T(v) = w \text{ para algum } v \in V\}.$$

O subconjunto $Im(T)$ é um subespaço vetorial de W.

Exemplo 2.30. Seja a transformação linear $T : \mathbb{R}^3 \to \mathbb{R}^3$ definida por $T(x, y, z) = (x, 2y, 0)$. A imagem de T é

$$\begin{aligned} Im(T) &= \{(x, 2y, 0) : x, y \in \mathbb{R}\} \\ &= \{x(1, 0, 0) + y(0, 2, 0) : x, y \in \mathbb{R}\}. \end{aligned}$$

Logo, o subespaço $Im(T)$ é gerado pelos vetores LI $(1, 0, 0)$ e $(0, 2, 0)$, portanto $dim Im(T) = 2$.

O núcleo de T é dado por

$$\begin{aligned} N(T) &= \{(x, y, z) : T(x, y, z) = (0, 0, 0)\} \\ &= \{(x, y, z) : (x, 2y, 0) = (0, 0, 0)\} \\ &= \{(0, 0, z) : z \in \mathbb{R}\} \\ &= \{z(0, 0, 1) : z \in \mathbb{R}\}. \end{aligned}$$

Logo, o $N(T)$ é gerado pelo vetor $(0, 0, 1)$ e assim $dim N(T) = 1$.

Definição 2.24. *Seja uma transformação $T : V \to W$. Dizemos que T é injetora se dados $u, v \in V$ com $T(u) = T(v)$, então $u = v$.*

Teorema 2.4. *Seja $T : V \to W$ uma transformação linear. Então, $N(T) = 0$ se, e somente se, T é injetora.*

Demonstração. Mostremos que se $N(T) = 0$, então T é injetora.

Suponhamos que $u, v \in V$ com $T(u) = T(v)$. Então, $T(u) - T(v) = T(u - v) = 0$, isto é, $u - v \in N(T)$. Por hipótese, $N(T) = 0$, assim $u - v = 0$. Logo, $u = v$. Portanto, T é injetora.

Agora mostremos que se T é injetora, então $N(T) = 0$.

Seja $v \in N(T)$, isto é, $T(v) = 0$. Como necessariamente $T(0) = 0$, $T(v) = T(0)$. Logo, $v = 0$, pois T é injetora. Portanto, $N(T) = 0$. □

Teorema 2.5. *Sejam V e W espaços vetoriais de dimensão finita. Seja $T : V \to W$ uma transformação linear. Então,*

$$dimN(T) + dimIm(T) = dimV.$$

Demonstração. Vamos supor inicialmente que $N(T) \neq \{0\}$ e seja $\{v_1, ..., v_n\}$ uma base de $N(T)$. Como $N(T) \subset V$ é subespaço de V podemos completar este conjunto de modo a obter uma base de V. Seja então, $\{v_1, ..., v_n, w_1, ..., w_m\}$ uma base de V. Queremos mostrar que $\{T(w_1), ..., T(w_m)\}$ é uma base de $Im(T)$, isto é,

i) Os vetores $T(w_1), ..., T(w_m)$ geram a imagem de T. De fato, dado $w \in Im(T)$, existe $u \in V$ tal que $T(u) = w$. Se $u \in V$, então $u = \alpha_1 v_1 + ... + \alpha_n v_n + \beta_1 w_1 + ... + \beta_m w_m$. Mas,

$$\begin{aligned} w = T(u) &= T(\alpha_1 v_1 + ... + \alpha_n v_n + \beta_1 w_1 + ... + \beta_m w_m) \\ &= \alpha_1 T(v_1) + ... + \alpha_n T(v_n) + \beta_1 T(w_1) + ... + \beta_m T(w_m). \end{aligned}$$

Como os vetores $v_1, ..., v_n$ pertencem ao $N(T)$, $T(v_1) = ... = T(v_n) = 0$. Assim,

$$w = \beta_1 T(w_1) + ... + \beta_m T(w_m)$$

e a imagem de T é gerada pelos vetores $\{T(w_1), ..., T(w_m)\}$.

ii) $\{T(w_1), ..., T(w_m)\}$ é linearmente independente. Consideremos a combinação linear

$$\alpha_1 T(w_1) + ... + \alpha_m T(w_m) = 0.$$

e mostremos que os α_i são todos nulos. Como T é linear, $T(\alpha_1 w_1 + ... + \alpha_m w_m) = 0$ segue que $\alpha_1 w_1 + ... + \alpha_m w_m \in N(T)$. Logo, $\alpha_1 w_1 + ... + \alpha_m w_m$ pode ser escrito como combinação linear da base $\{v_1, ..., v_n\}$ de $N(T)$, isto é, existem $\beta_1, ..., \beta_n$ tais que

$$\alpha_1 w_1 + ... + \alpha_m w_m = \beta_1 v_1 + ... + \beta_n v_n, \text{ ou ainda,}$$
$$\alpha_1 w_1 + ... + \alpha_m w_m - \beta_1 v_1 - ... - \beta_n v_n = 0.$$

Mas $\{v_1, ..., v_n, w_1, ..., w_m\}$ é uma base de V e temos então $\alpha_1 = ... = \alpha_m = \beta_1 = ... = \beta_n = 0$. Portanto, $dimV = n + m = dimN(T) + dimIm(T)$.

Se $N(T) = \{0\}$, considere $\{v_1, ..., v_n\}$ uma base de V e de maneira análoga à feita acima, pode-se mostrar que $\{T(v_1), ..., T(v_n)\}$ é uma base de $Im(T)$.

Se $N(T) = V$, então para todo $v \in V$, $T(v) \in N(T)$, isto é, $T(v) = 0$. Logo, $Im(T) = \{0\}$ e $dimIm(T) = 0$. Portanto, $dimV = dimN(T) + 0 = dimN(T) + dimIm(T)$. □

Exemplo 2.31. No exemplo 2.28 concluímos que a dimensão da imagem da transformação linear $T : \mathbb{R}^3 \to \mathbb{R}^3$ definida por $T(x, y, z) = (x, 2y, 0)$ é 2. Pelo teorema 2.5 $dim\mathbb{R}^3 = dimN(T) + dimIm(T)$. Logo, $dimN(T) = 3 - 2 = 1$.

Definição 2.25. *Sejam U e V espaços vetoriais sobre $\mathbb{F}$, tais que $dimU = n$ e $dimV = m$. Seja $T : U \to V$ uma transformação linear. Consideremos $\{u_1, u_2, ..., u_n\}$ uma base de U e $\{v_1, v_2, ..., v_m\}$ uma base de V. Para cada $u_i \in U$, existe $T(u_i) \in V$ de modo que os vetores $T(u_i)$ são combinações linear dos vetores da base de V. Assim, teremos:*

$$\begin{cases} T(u_1) = \alpha_{11}v_1 + \alpha_{21}v_2 + ... + \alpha_{m1}v_m \\ T(u_2) = \alpha_{12}v_1 + \alpha_{22}v_2 + ... + \alpha_{m2}v_m \\ \vdots \\ T(u_n) = \alpha_{1n}v_1 + \alpha_{2n}v_2 + ... + \alpha_{mn}v_m \end{cases}$$

onde $\alpha_{ij} \in \mathbb{F}$, com $1 \leqslant i \leqslant m$ e $1 \leqslant j \leqslant n$.

A matriz de ordem $m \times n$ que se obtém dos coeficientes $\alpha_{ij} \in \mathbb{F}$:

$$\begin{pmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2n} \\ \vdots & \vdots & & \vdots \\ \alpha_{m1} & \alpha_{m2} & \cdots & \alpha_{mn} \end{pmatrix}.$$

é chamada matriz da transformação linear.

Exemplo 2.32. Seja $T : \mathbb{R}^3 \to \mathbb{R}^2$ uma transformação linear tal que $T(x, y, z) = (2x + y - z, 3x - 2y + 4z)$. Sejam as bases $\{(1, 1, 1), (1, 1, 0), (1, 0, 0)\}$ de $\mathbb{R}^3$ e $\{(1, 3), (1, 4)\}$ de $\mathbb{R}^2$. Fazendo:

$$\begin{aligned} T(1, 1, 1) &= (2, 5) = 3(1, 3) - 1(1, 4), \\ T(1, 1, 0) &= (3, 1) = 11(1, 3) - 8(1, 4), \\ T(1, 0, 0) &= (2, 3) = 5(1, 3) - 3(1, 4). \end{aligned}$$

Então, a matriz de ordem 2×3 desta transformação é

$$\begin{pmatrix} 3 & 11 & 5 \\ -1 & -8 & -3 \end{pmatrix}$$

Definição 2.26. *Sejam U, V e W espaços vetoriais sobre $\mathbb{F}$, $T : V \to U$ e $S : U \to W$ transformações lineares. A transformação composta $S \circ T$ é definida por*

$$\begin{aligned} S \circ T : V &\to W \\ v &\mapsto S(T(v)). \end{aligned}$$

Teorema 2.6. *Sejam U, V e W espaços vetoriais sobre $\mathbb{F}$, $T : V \to U$ e $S : U \to W$ transformações lineares. A transformação composta $S \circ T$ é linear.*

Demonstração. Sejam $u, v \in V$ e $\alpha \in \mathbb{F}$. Então,

(i) $(S \circ T)(u + v) = S(T(u + v)) = S(T(u) + T(v)) = S(T(u)) + S(T(v)) = (S \circ T)(u) + (S \circ T)(v)$.

(ii) $(S \circ T)(\alpha u) = S(T(\alpha u)) = S(\alpha(T(u)) = \alpha S(T(u)) = \alpha(S \circ T)(u)$.

Logo, a transformação composta $S \circ T$ é linear. □

Exemplo 2.33. Sejam $T : \mathbb{R}^2 \to \mathbb{R}$ e $S : \mathbb{R} \to \mathbb{R}$ transformações lineares tais que $T(x, y) = x + 2y$ e $S(x) = 2x$. A transformação composta $S \circ T : \mathbb{R}^2 \to \mathbb{R}$ definida por:

$$S(T(x, y)) = 2(x + 2y) = 2x + 4y.$$

é linear.

Definição 2.27. *Seja V um espaço vetorial sobre $\mathbb{F}$. Uma transformação linear de V em V, isto é, $T : V \to V$ é chamada de operador linear.*

Definição 2.28. *Seja V um espaço vetorial sobre $\mathbb{F}$. Um operador linear $T : V \to V$ é invertível se existe $T^{-1} : V \to V$, tal que $T \circ T^{-1} = T^{-1} \circ T = I_V$, em que $I_V : V \to V$ é a função identidade.*

Exemplo 2.34. O operador linear $T : \mathbb{R}^2 \to \mathbb{R}^2$ definido por $T(x, y) = (x + y, x + 2y)$ é invertível e $T^{-1} : \mathbb{R}^2 \to \mathbb{R}^2$, tal que $T \circ T^{-1} = T^{-1} \circ T = I_{\mathbb{R}^2}$ é definida por $T^{-1}(x, y) = (2x + y, -x + y)$.

Neste capítulo destacamos alguns tópicos da Álgebra Linear que serão aplicados no desenvolvimento da teoria dos códigos lineares no capítulo 5. Aplicaremos também a teoria de matrizes, no capítulo 6. No próximo capítulo introduziremos o estudo da teoria da informação, onde destacaremos a funcionalidade e características dos sistemas de comunicação, bem como sua relação com o tema proposto.

3 Teoria da Informação

Para o desenvolvimento deste capítulo foram utilizados os itens [9], [12], [14] e [16] das referências.

O advento de novas tecnologias proporciona sistemas de comunicação cada vez mais modernos, porém durante a transmissão de informações pode haver algum tipo de perturbação ou algum erro pode ocorrer. Neste sentido, estudar o tema Teoria da Informação é fundamental, pois nestes sistemas de comunicação serão aplicados os códigos corretores de erros, objetos de estudo do capítulo 4, para detectar e corrigir possíveis erros.

O uso de sistemas de comunicação digital nas mais diversas áreas tem possibilitado o estudo e desenvolvimento de teorias matemáticas que sirvam de suporte às novas tecnologias digitais, os quais integram a teoria da informação. Esta trata dos aspectos quantitativos de armazenamento e transmissão das mensagens e tem como um de seus objetivos principais garantir que os dados enviados, através de um canal, não sofra nenhum tipo de perturbação. Durante esse processo, pode-se detectar dois problemas:

- falta de capacidade no armazenamento ou transmissão das mensagens enviadas;
- ruído na transmissão, ou seja, podem ocorrer erros nas mensagens enviadas.

Shannon, em seu trabalho "A Mathematical Theory of Communication" publicado em 1948, afirma que através de uma codificação adequada da informação, erros introduzidos pelo ruído do canal podem ser reduzidos a qualquer nível desejado sem sacrificar a taxa de transmissão da informação. Na figura 3.1 podemos observar os seguintes elementos de um sistema geral de comunicação:

- **Fonte de informação**: produz a mensagem que será enviada;
- **Transmissor**: possibilita enviar a mensagem com um sinal adequado;
- **Canal**: meio utilizado para enviar a mensagem do transmissor para o receptor;
- **Receptor**: realiza as ações inversas ao que foi feito pelo transmissor, reconstruindo a mensagem;
- **Destino**: para quem/onde a mensagem foi destinada.

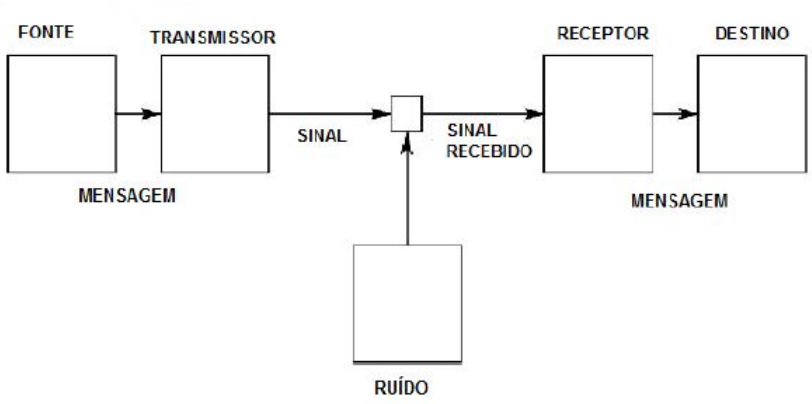


Figura 3.1: Diagrama - Sistema de Comunicação

3.1 Codificação

Código é um conjunto de símbolos usados na transmissão e recepção de mensagens. Os elementos básicos para se construir um código são:

- **Alfabeto**: Um conjunto finito de elementos. Cada um desses elementos, chama-se dígito. Quando o número de elementos do alfabeto é q, diz-se que o código é *q-ário*.
- **Palavra-código**: É uma sequência finita de dígitos. O número de dígitos de uma palavra-código é o seu comprimento.

Definição 3.1. *Codificação é o processo de mapeamento, ou seja, é uma conversão de uma dada sequência de dígitos (alfabeto fonte) em uma outra sequência de dígitos (alfabeto do código).*

Matematicamente, uma codificação é uma função injetiva que a cada símbolo do alfabeto fonte faz corresponder uma palavra-código. Dada uma codificação, f, com alfabeto fonte $F = \{s_1, s_2, ..., s_n\}$, as palavras $f(s_1), f(s_2), ..., f(s_n)$ são palavras-código e o código, C, é o conjunto dessas palavras-código,

$$C = \{f(s_1), f(s_2), ..., f(s_n)\}.$$

Exemplo 3.1. O código de César, utilizado pelo imperador romano Júlio César, consiste em fazer um deslocamento de 3 posições nas letras do alfabeto. Observe a tabela 3.1.

A	B	C	D	E	F	G	H	I	J	K	L	M
D	E	F	G	H	I	J	K	L	M	N	O	P

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Tabela 3.1: Alfabeto fonte e alfabeto código para o código de César

Ao recebermos a mensagem codificada DOJHEUD, convertemos essa sequência de dígitos na sequência ALGEBRA, o que nos permitirá ler a mensagem recebida.

Exemplo 3.2. Na codificação de Morse o alfabeto fonte é $\{A, B, C, ..., Y, Z\}$ e o alfabeto do código é $\{\cdot, -, \ \}$. A imagem do símbolo "A" do alfabeto fonte é a palavra "$\cdot, -, \ $", ou seja, $f(A) = \cdot -\ $.

3.2 Decodificação Única e Decodificação Instantânea

Os códigos devem ser unicamente decodificáveis para permitir o mapeamento inverso para o símbolo original do alfabeto no receptor. Dada uma codificação, f, com alfabeto fonte $F = \{s_1, s_2, ..., s_n\}$, as palavras no alfabeto F são codificadas definindo-se a função,

$$f^*(x_1, x_2, ..., x_m) = f(x_1)f(x_2), ..., f(x_m),$$

onde $x_i \in F, i = \{1, \ldots, m\}$.

Uma codificação f diz-se unicamente decodificável se a função f^*, definida no conjunto das palavras-código no alfabeto fonte, é injetora.

Exemplo 3.3. Seja f a codificação de César.

$$f^*(ALGEBRA) = f(A)f(L)f(G)f(E)f(B)f(R)f(A) = DOJHEUD.$$

Neste caso, a codificação de César é unicamente decodificável.

Exemplo 3.4. Considere a codificação binária,f, cujo alfabeto do código é $A = \{0, 1\}$, definida no alfabeto fonte, $F = \{a, b, c, d\}$ por

$$f(a) = 00 \quad f(b) = 110 \quad f(c) = 100 \quad f(d) = 1101.$$

Esta codificação não é unicamente decodificável. De fato,

$$f^*(bc) = f(b)f(c) = 110100 = f(d)f(a) = f^*(da).$$

Definição 3.2. *Uma codificação é instantânea se nenhuma palavra-código é prefixo de outra palavra-código.*

Exemplo 3.5. Considere a codificação binária,f, cujo alfabeto do código é $A = \{0, 1\}$, definida no alfabeto fonte, $F = \{a, b, c\}$ por

$$f(a) = 1 \quad f(b) = 10 \quad f(c) = 100.$$

Esta codificação não é instantânea, pois a palavra-código $(1, 0)$ é prefixo da palavra-código $(1, 0, 0)$.

3.3 Códigos de Bloco

Os códigos de bloco se caracterizam pelo fato do processo de codificação ser feito sobre blocos de bits ou blocos de símbolos. Isso quer dizer que um feixe de bits ou símbolos é segmentado em blocos de k bits ou símbolos (dimensão), a partir dos quais são geradas palavras-código com n bits ou símbolos (comprimento). Assim, a notação que caracteriza um código de bloco é $C(n, k)$. Suponhamos que queremos codificar uma mensagem constituída por uma sequência de símbolos extraídos de um alfabeto com q elementos. Primeiramente, o codificador "parte", ou secciona, a mensagem em "blocos" com um certo número, k, de dígitos e a cada um desses blocos vai acrescentar alguns dígitos extras, redundantes, chamados *dígitos de verificação de paridade*, de modo que os blocos fiquem mais longos, com n dígitos. Se k bits estão contidos em um bloco de n bits, então a quantidade de bits de redundância introduzidos no processo de codificação é $n - k$.

O segredo de uma boa codificação está na escolha adequada dos dígitos de paridade, ou seja, o desempenho do código perante a existência de erros e a necessidade de os detectar ou corrigir depende da qualidade da redundância que for introduzida na mensagem.

A Figura 3.2 ilustra o acréscimo dos dígitos de verificação de paridade pelo codificador.

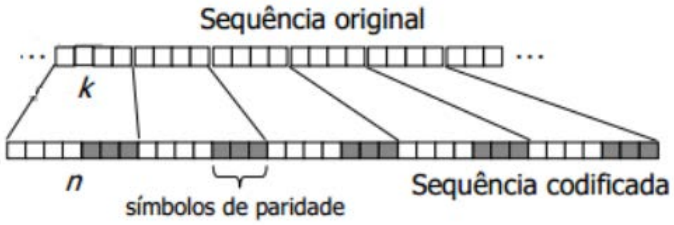


Figura 3.2: Dígitos de paridade

A quantidade de palavras-código diferentes que podem ser geradas é q^k. Para

códigos binários, teremos 2^k palavras-códigos. Daqui em diante, enfatizaremos durante todo o texto o uso do código binário.

Definição 3.3. *A taxa de codificação R_c de um código de bloco é a razão entre o número de bits de informação k e o número de bits da palavra-código n, isto é,*

$$R_c = \frac{k}{n}.$$

Como o número de bits da informação está entre zero e n, temos que a taxa de codificação está entre zero e um.

Exemplo 3.6. O código de bloco binário $C(5,3)$ possui $2^3 = 8$ palavras-código. A mensagem $(0,1,0)$ pode gerar a palavra-código $(0,1,0,\mathbf{1,0})$, onde os dois últimos símbolos são os dígitos de paridade.

Exemplo 3.7. Vamos determinar o comprimento, a dimensão e a taxa de codificação do código $C = \{(0,0,0,0),(0,1,1,0),(1,1,1,1),(1,0,0,1)\}$.

Como cada palavra-código possui 4 bits, o comprimento do código é 4, ou seja, $n = 4$. O código C tem 4 palavras-código, isto é, $2^2 = 4$. Logo, sua dimensão é 2, ou seja, $k = 2$. Assim sendo, a taxa de codificação é $R_c = \frac{2}{4} = 0,5$.

3.4 Detecção de erro

Os códigos corretores de erros codificam a informação inicial, adicionando informação redundante, de modo que, ao receber o sinal modificado pelo "ruído", seja possível, de alguma forma, recuperar a mensagem original.

Considere uma informação que será transmitida através de um canal ruidoso. Para se ter certeza que essa informação foi enviada sem erro algum, utiliza-se algum método de detecção de erro. Abordaremos o método da paridade, que possibilita a análise de duas maneiras:

- **Paridade Par**: A quantidade de bits 1 na palavra-código deve ser par.
- **Paridade Ímpar**: A quantidade de bits 1 na palavra-código deve ser ímpar.

Exemplo 3.8. Vamos verificar se houve erro durante a transmissão de dados ao receber a mensagem $(1,0,1,1,0,1,0,1)$. Para isso, considere que o bit de paridade está no início da palavra e que a paridade utilizada é par. Para detectar se houve erro na transmissão retira-se o dígito de paridade da mensagem recebida, que neste caso é o bit 1, obtendo $(0,1,1,0,1,0,1)$. Em seguida, calcula-se o bit de paridade da palavra $(0,1,1,0,1,0,1)$, que é o bit 0, pois a palavra possui quatro bits 1 e a paridade em questão deve ser par. Comparando-se o bit de paridade recebido com o calculado, nota-se que são diferentes. Portanto, houve erro durante a transmissão.

Destaquemos também o método que utiliza um código polinomial ou método CRC (Cyclic Redundancy Check).

O emissor e o receptor no processo de transmissão das informações escolhem um polinômio com coeficientes binários para representar uma sequência de bits. Este polinômio é chamado de polinômio gerador e é indicado por $G(X)$. Quanto maior o grau de $G(X)$ maior será a capacidade de detecção de erros. Neste polinômio o bit de maior ordem e o termo independente devem ser iguais a 1. O grau do polinômio $G(X)$ determina a quantidade de dígitos de verificação de paridade que serão acrescentados na mensagem inicial. Se a informação que se deseja transmitir possui k bits, então ela é representada por um polinômio de grau $k-1$.

Considere que o emissor deseja enviar uma mensagem. O processo de codificação consiste em:

- Acrescentar à mensagem inicial tantos zeros quanto for o grau do polinômio gerador $G(X)$.
- Escrever o polinômio $p(X)$ que representa a mensagem obtida no item anterior. O polinômio $p(X)$ deve ser divisível pelo polinômio $G(X)$.
- Dividir o polinômio $p(X)$ por $G(X)$, obtendo o polinômio resto $R(X)$.
- Somar o polinômio $p(X)$ com o polinômio $R(X)$.

Os coeficientes do polinômio $R(X)$ determinam os dígitos de verificação de paridade. A mensagem recebida será formada pelos coeficientes do polinômio $p(X) + R(X)$.

Exemplo 3.9. Sejam $(1,0,1,1,1,0)$ a mensagem a ser enviada e o polinômio gerador $G(X) = X^3 + 1$. Como $G(X)$ possui grau 3, devemos acrescentar à mensagem inicial três dígitos de verificação de paridade, obtendo a informação $(1,0,1,1,1,0,0,0,0)$. Deste modo, determinamos o polinômio $p(X)$, de modo que cada bit desta informação seja o coeficiente de um termo de $p(X)$, isto é, $p(X) = X^8 + X^6 + X^5 + X^4$.

Fazendo $\dfrac{p(X)}{G(X)}$, obtemos o polinômio resto $R(X) = 0 \cdot X^2 + X + 1$.

Logo, $p(X) + R(X) = X^8 + X^6 + X^5 + X^4 + X + 1$.

Portanto, a mensagem recebida é $(1,0,1,1,1,0,0,1,1)$.

A Figura 3.3 ilustra a divisão binária feita com os coeficientes dos polinômios $p(X)$ e $G(X)$.

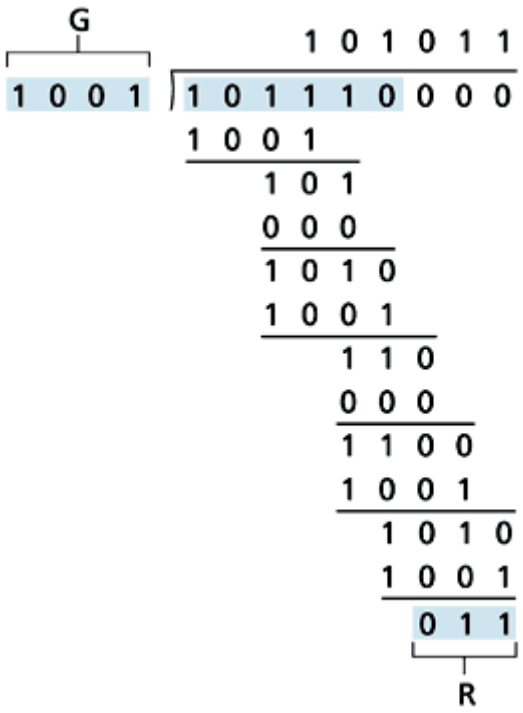


Figura 3.3: Divisão binária entre $p(X)$ e $G(X)$

O processo de decodificação consiste em dividir o polinômio, cujos coeficientes são os bits da mensagem recebida, pelo polinômio gerador $G(X)$. Se o polinômio resto $R(X)$ não for o polinômio nulo, então ocorreu erro durante a transmissão da mensagem.

Exemplo 3.10. Seja $(1, 0, 1, 0, 1, 0, 0, 1, 1)$ a mensagem recebida através de um canal ruidoso. Verifiquemos se houve erro durante a transmissão da mensagem utilizando o polinômio gerador $G(X) = X^3 + 1$.

A mensagem em questão é representada pelo polinômio $m(X) = X^8 + X^6 + X^4 + X + 1$.

Fazendo $\dfrac{m(X)}{G(X)}$, obtemos o polinômio resto $R(X) = X^2$, que não é o polinômio nulo.

Logo, a mensagem foi recebida com erro.

Neste capítulo apresentamos dois métodos de detecção de erro, mas será que após detectar o erro é sempre possível corrigi-lo? No próximo capítulo abordaremos os códigos corretores de erros, com o objetivo de responder este questionamento.

4 Códigos Corretores de Erros

Este capítulo foi desenvolvido de acordo com os itens [6], [7], [8], [9], [14], [16] e [17] das referências bibliográficas.

No capítulo anterior vimos que é possível detectar erros na transmissão de informações. Vamos agora, verificar a possibilidade de correção destes erros com a finalidade de garantir o recebimento da informação enviada. Durante a transmissão de dados, algumas vezes ocorrem erros no processo de codificação. O ruído faz com que a mensagem recebida não seja aquela que foi enviada. O objetivo dos códigos corretores de erro é detectar irregularidades e possibilitar a sua correção.

Figura 4.1: Canal ruidoso

Por volta de 1947, Richard W. Hamming trabalhava no Laboratório Bell de Tecnologia, onde analisava erros ocorridos na transmissão de informações. Lá ele pôde perceber que, se era realmente possível detectar o erro então porquê não seria possível localizá-lo e corrigi-lo. A partir deste questionamento, Hamming passou a desenvolver uma teoria que possibilitaria tal intento.

Ressaltemos que os estudos subsequentes de Shannon (1948), que de certa forma aprimorou a teoria de Hamming, foi de fundamental importância para o desenvolvimento da teoria dos códigos. Já em 1949, Marcel J. E. Golay, ao ler sobre o código $(7, 4)$ de Hamming estendeu o resultado para um código corretor de erro único, cujo comprimento é um número primo. Os primeiros trabalhos com códigos corretores de erros feitos por Golay, Hamming e Shannon possibilitaram desenvolver estudos e ideias que são usadas em nosso dia a dia, como por exemplo a comunicação móvel (telefones celulares), aparelhos de armazenamentos de dados (gravador, compact disk, DVD), além de comunicações via satélite, processamento de imagens digitais, internet e rádio, entre outras.

4.1 Códigos de Hammimg

O código de Hamming é um código corretor de erro utilizado no processamento de sinal e em telecomunicações. A sua utilização permite a transferência e armazenamento de dados de forma segura e eficiente. Hamming desenvolveu um código capaz de detectar até dois erros e corrigir um erro, caso este seja único. Conforme seus estudos avançavam sempre ocorriam novos questionamentos no sentido de elaborar códigos cada vez mais eficazes. Para construir um código corretor de erro de Hamming, deve-se primeiramente determinar o número de bits de paridade b necessários, de acordo com a relação

$$2^b \geqslant k + b + 1,\ b \in \mathbb{N},$$

onde k é o número de dígitos da informação a ser codificada e b é o primeiro número natural que satisfaz a esta relação. Em seguida, arranjamos os bits de paridade na informação colocando-os da esquerda para a direita na posição onde se encontram as potências de 2. Na sequência devemos atribuir adequadamente o valor 0 ou 1 para cada bit de paridade. Por fim, determinamos a palavra-código resultante.

Exemplo 4.1. Vamos determinar o código de Hamming para a informação $(1, 0, 1, 0)$, usando paridade par. Note que a informação possui 4 bits, ou seja, $k = 4$. Usando a relação $2^b \geqslant 4 + b + 1$, para $b = 1$, temos que $2 \geqslant 4 + 1 + 1$ não se verifica. Para $b = 2$, $4 \geqslant 4 + 2 + 1$ também não se verifica. E para $b = 3$, $8 \geqslant 4 + 3 + 1$ se verifica. Logo, a palavra-código deverá ter 3 bits de paridade, totalizando assim uma mensagem com 7 bits. Portanto, teremos o código de Hamming conhecido como código $(7, 4)$.

Bits	b_1	b_2	k_1	b_3	k_2	k_3	k_4
Posição do bit	1	2	3	4	5	6	7
Posição em binário	001	010	011	100	101	110	111
Bits de dados (k_n)			1		0	1	0
Bits de paridade (b_n)							

Tabela 4.1: Código de Hamming (7,4) - Bits de dados

Vamos agora determinar os bits de paridade.

O bit b_1 verifica os bits das posições 1, 3, 5 e 7. Como temos um bit 1 nessas posições e a paridade é par devemos ter $b_1 = 1$.

O bit b_2 verifica os bits das posições 2, 3, 6 e 7. Como temos dois bits 1 nessas posições e a paridade é par devemos ter $b_2 = 0$.

O bit b_3 verifica os bits das posições 4, 5, 6 e 7. Como temos um bit 1 nessas posições e a paridade é par devemos ter $b_3 = 1$.

Para determinar a palavra-código basta acrescentar os bits de paridade ao bits de dados.

Bits	b_1	b_2	k_1	b_3	k_2	k_3	k_4
Posição do bit	1	2	3	4	5	6	7
Posição em binário	001	010	011	100	101	110	111
Bits de dados (k_n)			1		0	1	0
Bits de paridade (b_n)	**1**	**0**		**1**			

Tabela 4.2: Código de Hamming (7,4) - Bits de paridade

Portanto, a palavra-código é $(1, 0, 1, 1, 0, 1, 0)$.

4.1.1 A métrica de Hamming

Sejam A um conjunto finito (alfabeto código) e $C \subsetneq A^n = A \times A \times \cdot \times A$ um código onde todas as palavras-código possuem comprimento n.

Definição 4.1. *A distância de Hamming entre dois vetores x e y, denotada por $d(x, y)$, é o número de posições onde os dígitos correspondentes são diferentes.*

Matematicamente, sejam $x = (x_1, \cdots, x_n) \in A^n$ e $y = (y_1, \cdots, y_n) \in A^n$. A distância de Hamming entre x e y é definida por

$$d(x, y) =\mid \{i \in \{1, 2, \ldots, n\} : x_i \neq y_i\} \mid,$$

onde $\mid ... \mid$ representa a quantidade de índices i.

Exemplo 4.2. Sejam as palavras-código $x = (1, 0, 1, 0, 0, 1, 1)$ e $y = (0, 1, 0, 0, 0, 1, 1)$. A distância de Hamming entre x e y é $d(x, y) = 3$.

Para $x \in C$ e $y \in A^n$, $d(x, y)$ é o número de erros cometidos se, ao ser transmitida a palavra x é recebida a palavra y.

Proposição 4.1. *A distância de Hamming é uma métrica em A^n, isto é, dados $x, y, z \in A^n$, temos*

- $d(x, y) \geqslant 0$ *e* $d(x, y) = 0 \Leftrightarrow x = y$.
- $d(x, y) = d(y, x)$.
- *Desigualdade triangular:* $d(x, z) \leqslant d(x, y) + d(y, z)$.

A demonstração desta proposição encontra-se na referência [17].

Definição 4.2. *A distância mínima de um código, denotado por d_{min} é dada por*

$$d_{min} = min\{d(x, y) : x, y \in C \text{ e } x \neq y\}.$$

Exemplo 4.3. A distância mínima do código

$$C = \{(0,0,0,0,0), (0,1,0,1,1), (1,0,1,1,0), (1,1,1,0,1)\}$$

é 3.

De fato, temos que:

$$d((0,0,0,0,0),(0,1,0,1,1)) = 3,\ d((0,0,0,0,0),(1,0,1,1,0)) = 3,$$
$$d((0,0,0,0,0),(1,1,1,0,1)) = 4,\ d((0,1,0,1,1),(1,0,1,1,0) = 4,$$
$$d((0,1,0,1,1),(1,1,1,0,1)) = 3 \text{ e } d((1,0,1,1,0),(1,1,1,0,1)) = 3$$

Logo, $d_{min} = min\{3,4\} = 3$.

Definição 4.3. *O peso de Hamming de um vetor $x = (x_1, \cdots, x_n) \in A^n$, denotado por $w(x)$, é a quantidade de dígitos x_i diferentes de zero.*

Exemplo 4.4. Sejam as palavras-código $x = (1,0,1,0,0,1,1)$ e $y = (0,1,0,0,0,1,1)$. O peso de Hamming dessas palavras-código é $w(x) = 4$ e $w(y) = 3$.

Proposição 4.2. *Se x e y são palavras-código de um código binário qualquer, então $d(x,y) = w(x+y)$.*

Demonstração. Sejam $x = (x_1, \cdots, x_n)$ e $y = (y_1, \cdots, y_n)$ as palavras-códigos de um código binário. Pelo princípio da indução finita sobre n, temos que $d(x,y) = w(x+y)$.

De fato, se $n = 1$, então $x = 1$ e $y = 0$ e $d(x,y) = w(x+y) = 1$.

Suponhamos que $d(x,y) = w(x+y) = z$, para algum n e mostremos que a igualdade é válida para $n+1$.

Tomemos $x = (x_1, \cdots, x_n, x_{n+1})$ e $y = (y_1, \cdots, y_n, y_{n+1})$. Temos dois casos:

a) se $x_{n+1} = y_{n+1}$, então, pela hipótese de indução finita, $d(x,y) = w(x+y) = z$.

b) se $x_{n+1} \neq y_{n+1}$, então $d(x,y) = z+1$. Por outro lado, $x+y = (x_1+y_1, \cdots, x_n+y_n, x_{n+1}+y_{n+1})$. Assim, $w(x+y) = z+1$, pois $x_{n+1}+y_{n+1} \neq 0$. Logo, $d(x,y) = w(x+y) = z+1$.

Portanto, pelo princípio de indução finita, $d(x,y) = w(x+y)$, para todo n. □

Exemplo 4.5. Sejam as palavras-código $x = (1,0,1,1,1)$ e $y = (0,0,1,0,1)$.

Temos que $d(x,y) = 2$. Como $x+y = (1,0,0,1,0)$, segue que $w(x+y) = 2$. Portanto, $d(x,y) = w(x+y)$.

4.2 Detecção de erros

Nesta seção destacaremos a importância da distância de Hamming no processo de decodificação de mensagens.

Definição 4.4. *Sejam $x = (x_1, \cdots, x_n) \in A^n$ e $r \in \mathbb{R}^+$.*

A bola com centro em x e raio r é o subconjunto de A^n

$$B(x,r) = \{y \in A^n : d(x,y) \leqslant r\}.$$

Um código $C \subsetneq A^n$ diz-se *t-detector de erros* se, para todo $x \in C$, havendo pelo menos 1 erro e no máximo t erros na transmissão de x, então a mensagem recebida não é uma palavra-código.

Matematicamente, C é t-detector de erros se, para qualquer $x \in C$,

$$B(x,t) \cap C = \{x\}.$$

Teorema 4.1. *Um código $C \subsetneq A^n$ é t-detector de erros, se e somente se, $d_{min} > t$.*

Demonstração. Suponha que C detecta t erros e que $d_{min} \leqslant t$. Assim sendo, existem $x \in C$ e $y \in C$ tais que

$$0 < d(x,y) = d_{min} \leqslant t.$$

Logo, $y \in B(x,t) \cap C$ e $x \neq y$, o que contradiz a hipótese.

Reciprocamente, por hipótese temos que $d_{min} > t$. Tomemos $x \in C$ qualquer. Se $y \in B(x,t) \cap C$, então

$$d(x,y) \leqslant t < d_{min}$$

Logo, $x = y$. □

Observação 4.1. Uma vez detectado um erro, adotamos um critério de correção que irá substituir o elemento y recebido, pelo elemento x do código que está mais próximo de y. Para que a correção seja possível será necessário então que não haja ambiguidades quanto à determinação de um tal elemento.

A decodificação pode ser feita utilizando-se a distância de Hamming, ou seja, decodificando pela palavra mais próxima.

Exemplo 4.6. Consideremos o código

$$C = \{(0,0,0,0,0), (0,1,0,1,1), (1,0,1,1,0), (1,1,1,0,1)\}.$$

Ao receber a mensagem $(0,0,1,1,0)$, o receptor detecta que houve um erro, pois a mensagem $(0,0,1,1,0)$ não pertence ao código. Vamos então, corrigir este erro.

Temos que,

$$d((0,0,1,1,0),(0,0,0,0,0)) = 2,\ d((0,0,1,1,0),(0,1,0,1,1)) = 3,$$
$$d((0,0,1,1,0),(1,0,1,1,0)) = 1 \text{ e } d((0,0,1,1,0),(1,1,1,0,1)) = 4$$

Como as menores distâncias são 1 e 2, tomemos as bolas com centro em $(0,0,1,1,0)$ e raios 1 e 2. Assim,

$$B((0,0,1,1,0),2) \cap C = \{(0,0,0,0,0),(1,0,1,1,0)\} \text{ e}$$
$$B((0,0,1,1,0),1) \cap C = \{(1,0,1,1,0)\}.$$

A palavra-código que está mais próxima da mensagem recebida $(0,0,1,1,0)$ é $(1,0,1,1,0)$.

Logo, se houver um erro na transmissão da mensagem $(1,0,1,1,0)$, e for recebida a mensagem $(0,0,1,1,0)$ que não pertence ao código, o erro pode ser corrigido, substituindo $(0,0,1,1,0)$ pela palavra-código x, tal que $d(x,(0,0,1,1,0))$ seja mínima. Contudo, se houver dois erros e for recebida a mensagem $(1,1,0,1,0)$, que não pertence ao código, ao invés de $(1,0,1,1,0)$, o erro é detectado, mas não pode ser corrigido, uma vez que há duas palavras-código que estão à mesma distância mínima de $(1,1,0,1,0)$.

Observação 4.2. A obtenção de códigos eficientes, com a finalidade de melhorar a relação entre comprimento, número de palavras-código e capacidade de correção (distância entre as palavras-código) é o principal problema da teoria de códigos.

4.3 Correção de erros

Um código $C \subsetneq A^n$ diz-se *t-corretor de erros* se, para todo $x \in C$, havendo pelo menos 1 erro e no máximo t erros na transmissão de x, então ao receber uma mensagem y, a mensagem x é a única mensagem de C que está à distância mínima de y.

Matematicamente, C é t-corretor de erros se, para todo $x \in C$ e todo $y \in A^n$, tais que $1 \leqslant d(x,y) \leqslant t$, tem-se

$$d(x,y) < d(z,y), \forall z \in C \setminus \{x\}.$$

Observação 4.3. Utilizaremos as seguintes notações: dado um número real x, denotaremos por $\lfloor x \rfloor$ o maior inteiro menor ou igual a x e representaremos $\lceil x \rceil$ como o menor inteiro maior ou igual a x.

Teorema 4.2. *Um código $C \subsetneq A^n$ é t-corretor de erros, se e somente se, $d_{min} > 2t$.*

Demonstração. Suponha que $d_{min} \leqslant 2t$ e mostremos que C não é t-corretor de erros construindo $x \in C$, $y \in A^n$ e $z \in C \setminus \{x\}$, tais que $1 \leqslant d(x,y) \leqslant t$ e $d(x,y) \geqslant d(z,y)$.

Sejam $x, z \in C$ tais que $d(x,z) = d_{min} = r$ e $i_1, i_2, ..., i_r \in \{1,2,...,n\}$ os índices i para os quais $x_i \neq z_i$. Assim sendo, $1 \leqslant r = d_{min} \leqslant 2t$.

Seja $y \in A^n$ definido por

$y_i = x_i$, se $i = i_j$ com $j \in \{1,...,r\}$ par.

$y_i = z_i$, se $i = i_j$ com $j \in \{1,...,r\}$ ímpar.

$y_i = x_i = z_i$, se $i \notin \{i_1,...,i_r\}$.

$1 \leqslant d(x,y) =\mid \{j \in \{1,...,r\} : j \text{ é ímpar } \} \mid = \lceil \frac{r}{2} \rceil \leqslant t$.

$d(z,y) =\mid \{j \in \{1,...,r\} : j \text{ é par } \} \mid = \lfloor \frac{r}{2} \rfloor \leqslant d(x,y)$.

Reciprocamente, suponha que $d_{min} > 2t$.

Sejam $x \in C$ e $y \in A^n$ tais que $1 \leqslant d(x,y) \leqslant t$. Se $z \in C \setminus \{x\}$, então $d(x,z) \geqslant d_{min} > 2t$.

Logo, pela desigualdade triangular,

$$d(x,z) \leqslant d(x,y) + d(y,z).$$

Portanto,

$$d(y,z) > 2t - d(x,y) \geqslant t \geqslant d(x,y).$$

□

Corolário 4.1. *Um código C com distância mínima d_{min} pode detectar até $d_{min} - 1$ erros e corrigir até $\lfloor \frac{d_{min}-1}{2} \rfloor$ erros.*

A demonstração deste corolário encontra-se na referência [16].

Exemplo 4.7. Seja um código C com distância mínima d_{min}. Esse código pode detectar e corrigir a seguinte quantidade de erros:

d_{min}	Número de erros detectados por C	Número de erros corrigidos por C
1	0	0
2	1	0
3	2	1
4	3	1

Tabela 4.3: Relação entre distância mínima e número de erros

Exemplo 4.8. Determine a distância mínima do código

$$C = \{(0,1,0,1,0,1,1,0),(0,0,1,1,0,0,1,1),(0,1,0,1,0,0,1,1),(0,0,1,1,0,1,1,0)\}$$

e verifique sua capacidade de detecção e correção de erros.

Analisemos a distância entre cada uma das palavras-código, assim:
$d((0,1,0,1,0,1,1,0),(0,0,1,1,0,0,1,1)) = 4,$
$d((0,1,0,1,0,1,1,0),(0,1,0,1,0,0,1,1)) = 2,$
$d((0,1,0,1,0,1,1,0),(0,0,1,1,0,1,1,0)) = 2,$
$d((0,0,1,1,0,0,1,1),(0,1,0,1,0,0,1,1)) = 2,$
$d((0,0,1,1,0,0,1,1),(0,0,1,1,0,1,1,0)) = 2,$
$d((0,1,0,1,0,0,1,1),(0,0,1,1,0,1,1,0)) = 4.$

Logo, conclui-se que a distância mínima do código C é 2 e portanto, pelo exemplo anterior, esse código tem capacidade para detectar um erro e não possui capacidade de correção de tal erro.

4.4 Isometrias em A^n

Seja A um alfabeto e n um número natural. Dizemos que uma função $F : A^n \longrightarrow A^n$ é uma isometria de A^n se ela preserva distância, ou seja,

$$d(F(x), F(y)) = d(x, y), \forall x, y \in A^n.$$

Exemplo 4.9. Sejam $A = \{0, 1\}$ e $n = 3$. A aplicação

$$\begin{array}{rccc} F: & A^3 & \longrightarrow & A^3 \\ & (a_1, a_2, a_3) & \longmapsto & (1 - a_1, a_2, a_3). \end{array}$$

é uma isometria, pois de $1 - a_1 = 1 - b_1 \Leftrightarrow a_1 = b_1$, conclui-se que
$d(F(a_1, a_2, a_3), F(b_1, b_2, b_3)) = d((1 - a_1, a_2, a_3), (1 - b_1, b_2, b_3)) =$
$d((a_1, a_2, a_3), (b_1, b_2, b_3))$.

Exemplo 4.10. Se f é uma aplicação bijetora em A e $i \in \{1, 2, ..., n\}$, então

$$\begin{array}{rccc} T_f^i: & A^n & \longrightarrow & A^n \\ & (a_1, ..., a_n) & \longmapsto & (a_1, ..., f(a_i), ..., a_n). \end{array}$$

é uma isometria.

Exemplo 4.11. Se π é uma permutação de $\{1, 2, ..., n\}$, então

$$\begin{array}{rccc} T_\pi: & A^n & \longrightarrow & A^n \\ & (a_1, ..., a_n) & \longmapsto & (a_{\pi(1)}, ..., a_{\pi(n)}). \end{array}$$

é uma isometria.

4.5 Códigos Equivalentes

Sejam $C \subsetneq A^n$ e $C' \subsetneq A^n$ dois códigos. Se existe uma isometria F de A^n tal que $F(C) = C'$, dizemos que C é equivalente a C'.

Teorema 4.3. *Sejam $C \subsetneq A^n$ e $C' \subsetneq A^n$ dois códigos e $(a_1, ..., a_n) \in C$. Os códigos C e C' são equivalentes, se e somente se, existe uma permutação π de $\{1, 2, ..., n\}$ e $f_1, ..., f_n$ bijeções em A tais que*

$$C' = \{(f_{\pi(1)}(a_{\pi(1)}), ..., f_{\pi(n)}(a_{\pi(n)}))\}.$$

Dois códigos de comprimento n sobre o alfabeto A são equivalentes se, e só se, um deles se obtém do outro fazendo as seguintes operações:

- Substituir o elemento do alfabeto A que está na posição i de cada palavra, segundo uma bijeção f_i em A, para $i = 1, 2, ..., n$.
- Permutar as posições dos símbolos em cada palavra segundo uma permutação π de $\{1, 2, ..., n\}$.

Exemplo 4.12. Sejam o alfabeto $A = \{a, b, c, d, e\}$ e o código $C = \{aab, abc, cde, bbd\}$. Considere as bijeções f_1, f_2 e f_3 em A tais que

$$\begin{array}{rcl} f_1 : A & \longrightarrow & A \\ a & \longmapsto & a \\ b & \longmapsto & b \\ c & \longmapsto & c \\ d & \longmapsto & d \\ e & \longmapsto & e \end{array} \qquad \begin{array}{rcl} f_2 : A & \longrightarrow & A \\ a & \longmapsto & a \\ b & \longmapsto & c \\ c & \longmapsto & d \\ d & \longmapsto & b \\ e & \longmapsto & e \end{array} \qquad \begin{array}{rcl} f_3 : A & \longrightarrow & A \\ a & \longmapsto & b \\ b & \longmapsto & c \\ c & \longmapsto & a \\ d & \longmapsto & e \\ e & \longmapsto & d \end{array}$$

Tomemos a permutação de $\{1, 2, 3\}$, $\pi = (1\ \ 2)$.

Logo,

$$aab \dashrightarrow f_1(a)f_2(a)f_3(b) = aac \dashrightarrow aac$$

$$abc \dashrightarrow f_1(a)f_2(b)f_3(c) = aca \dashrightarrow caa$$

$$cde \dashrightarrow f_1(c)f_2(d)f_3(e) = cbd \dashrightarrow bcd$$

$$bbd \dashrightarrow f_1(b)f_2(b)f_3(d) = bce \dashrightarrow cbe$$

Portanto, obtemos o código $C' = \{aac, caa, bcd, cbe\}$ equivalente a C.

5 Códigos de Bloco Lineares

Os itens [12], [14] e [16] das referências foram utilizados no desenvolvimento deste capítulo.

Sejam p um número primo e $m \in \mathbb{N}$ e $q = p^m$. Representa-se por $\mathbb{F}_q$ o corpo finito com q elementos. Nos códigos de bloco lineares o alfabeto considerado é $A = \mathbb{F}_q$, para algum p primo e algum $m \in \mathbb{N}$.

Definição 5.1. *Um código linear $C(n, k)$ é um subespaço vetorial de dimensão k do espaço vetorial $\mathbb{F}_q$.*

Um código linear satisfaz as seguintes propriedades:

i) A palavra-código nula pertence ao conjunto de palavras-código.

ii) A soma de duas palavras-código é também uma palavra-código.

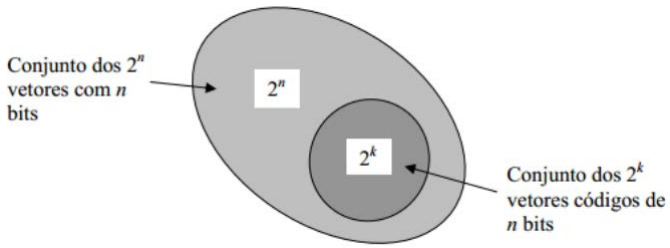


Figura 5.1: Representação dos códigos binários lineares como um subespaço vetorial

Exemplo 5.1. O código de bloco linear $C(4, 2)$ sobre $\mathbb{Z}_2$ possui 4 palavras-código.

Dada a mensagem (x_1, x_2), cada palavra-código será da forma $(x_1, x_2, x_2, x_1 + x_2)$.

Mensagem	Palavras-código
00	0000
01	0111
10	1001
11	1110

Tabela 5.1: Código C(4,2)

Observação 5.1. No exemplo 5.1, observe que a palavra-código $(1, 1, 1, 0)$ pode ser vista, por exemplo, como a soma, em $\mathbb{Z}_2$, das palavras-código $(0, 1, 1, 1)$ e $(1, 0, 0, 1)$.

Observação 5.2. A última palavra-código da tabela 5.1 foi obtida a partir das $k = 2$ palavras cuja componente de mensagem tem apenas um bit 1: $(0, 1)$ e $(1, 0)$, as quais são linearmente independentes, ou seja, nenhuma delas é combinação linear da outra.

Para um código linear qualquer $C(n, k)$, as $2^k - (k + 1)$ palavras-código restantes (excluindo a palavra nula) são obtidas por combinação linear de k palavras-código linearmente independentes.

Adotando a notação de vetores, podemos associar as k entradas da informação com o vetor $X = (x_1, x_2, ..., x_k)$ e também associar as n entradas da palavra-código com o vetor $Y = (y_1, y_2, ..., y_n)$. Vamos supor sem perda de generalidade que os dígitos de paridade ficam todos juntos no final da palavra-código.

Exemplo 5.2. Vamos construir um código binário de comprimento 5 de modo que as três primeiras componentes x_1, x_2 e x_3 de cada palavra-código sejam de informação e possuam dois dígitos de paridade, x_4 e x_5.

Definimos os dígitos de paridade assim:

$$x_4 = x_1 + x_2$$

$$x_5 = x_1 + x_3$$

Usando a representação vetorial, cada palavra-código será da forma

$$(x_1, x_2, x_3, x_1 + x_2, x_1 + x_3)$$

Para cada informação, de acordo com as coordenadas do vetor descrito acima, obtém-se cada uma das 8 palavras-código do código $C(5, 3)$.

Mensagem	Palavras código
000	00000
001	00101
010	01010
100	10011
011	01111
101	10110
110	11001
111	11100

Tabela 5.2: Código C(5,3)

Proposição 5.1. *Seja $C \subsetneq (\mathbb{F}_q)^n$ um código linear. Então*

$$d_{min} = min\{d(x,0) : x \in C \setminus \{0\}\}.$$

Demonstração. Seja $d' = min\{d(x,0) : x \in C \setminus \{0\}\}$.

Sejam $x, y \in C$ tais que $d_{min} = d(x,y)$. Como C é um espaço vetorial, $x-y \in C\setminus\{0\}$ e $d' = d(x-y,0)$.

Logo,

$$d' \leqslant |\{i \in \{1,2,...,n\} : x_i - y_i \neq 0\}| = d(x,y) = d_{min}.$$

Por outro lado, como $0 \in C$, para qualquer $x \in C \setminus \{0\}$ temos $d(x,0) \geqslant d_{min}$.

Logo, $d' \geqslant d_{min}$.

Portanto, $d' = d_{min}$, ou seja, $d_{min} = min\{d(x,0) : x \in C \setminus \{0\}\}$. □

Um código linear de comprimento n, dimensão k e distância mínima d será chamado *código* $[n,k,d]$.

Exemplo 5.3. O código $C = \{(0,0,0,0),(1,0,1,1),(0,1,1,0),(1,1,0,1)\}$ é um subespaço vetorial de $(\mathbb{Z}_2)^4$. O conjunto $\mathcal{B} = \{(1,0,1,1),(1,1,0,1)\}$ é uma base de C. Temos que,

$$w(1,0,1,1) = 3,\ w(1,1,0,1) = 3 \text{ e } w(0,1,1,0) = 2.$$

Logo, a distância mínima do código C é 2.

Portanto, temos um código $[4,2,2]$.

5.1 Representação de um código linear como imagem de uma transformação linear

Seja $C(n,k)$ um código linear sobre $\mathbb{F}_q$ e $\{v_1, v_2, ..., v_k\}$ uma base de C. A aplicação

$$\begin{array}{rccc} T: & (\mathbb{F}_q)^k & \longrightarrow & (\mathbb{F}_q)^n \\ & (x_1, x_2, ..., x_k) & \longmapsto & x_1v_1 + x_2v_2 + \cdots + x_kv_k \end{array}$$

é linear, injetora e $Im(T) = C$.

Assim, um código linear $C(n,k)$ sobre $\mathbb{F}_q$ pode ser representado por uma aplicação injetora $T : (\mathbb{F}_q)^k \longrightarrow (\mathbb{F}_q)^n$ com característica k.

Seja T uma transformação linear que representa um código linear. Para determinar se $v \in (\mathbb{F}_q)^n$ é ou não uma palavra-código é preciso obter uma base $\{v_1, v_2, ..., v_k\}$ de $Im(T)$ e resolver o sistema de equações $x_1v_1 + x_2v_2 + \cdots + x_kv_k = v$.

Exemplo 5.4. Dados $A = \{0,1\} = \mathbb{Z}_2$, $n = 5$ e o código linear $C(5,2)$, tal que

$$C = \{(0,0,0,0,0), (0,1,0,1,1), (1,0,1,1,0), (1,1,1,0,1)\}.$$

O conjunto $\{(1,0,1,1,0), (0,1,0,1,1)\}$ é uma base de C e tomando a aplicação linear

$$\begin{array}{rccc} T: & (\mathbb{Z}_2)^2 & \longrightarrow & (\mathbb{Z}_2)^5 \\ & (x_1, x_2) & \longmapsto & (x_1, x_2, x_1, x_1 + x_2, x_2) \end{array}$$

obtemos $C = Im(T)$.

Esta é uma representação do código C como subespaço imagem de uma transformação linear.

Observação 5.3. A codificação da mensagem (x_1, x_2) do código fonte é obtida calculando $T(x_1, x_2)$ para quaisquer $x_1, x_2 \in \mathbb{Z}_2$.

5.2 Representação de um código linear como núcleo de uma transformação linear

Dado um código linear $C(n,k)$ sobre $\mathbb{F}_q$. Seja C' um subespaço de $(\mathbb{F}_q)^n$ tal que $C \oplus C' = (\mathbb{F}_q)^n$. Tomemos $w \in (\mathbb{F}_q)^n$. Então existem um único $x \in C$ e um único $y \in C'$, tais que $x + y = w$. Escreve-se $w = x \oplus y$. A aplicação

$$\begin{array}{rccc} T: & C \oplus C' & \longrightarrow & (\mathbb{F}_q)^n \\ & x \oplus y & \longmapsto & w. \end{array}$$

é linear e $N(T) = C$.

Dado $w \in (\mathbb{F}_q)^n$. Se $w \in C$ então, $T(w) = 0$.

Exemplo 5.5. Vamos obter uma representação do código linear $C(5,2)$ dado por

$$C = \{(0,0,0,0,0),(0,1,0,1,1),(1,0,1,1,0),(1,1,1,0,1)\}$$

como núcleo de uma transformação linear.

Para isso, tomemos os vetores linearmente independentes de $(\mathbb{Z}_2)^5$:

$$(1,0,1,1,0),(0,1,0,1,1),(0,0,1,0,0),(0,0,0,1,0),(0,0,0,0,1).$$

Assim, o subespaço C' de $(\mathbb{Z}_2)^5$ gerado por $\{(0,0,1,0,0),(0,0,0,1,0),(0,0,0,0,1)\}$ é um subespaço complementar de C.

Para qualquer $(x_1,x_2,x_3,x_4,x_5) \in (\mathbb{Z}_2)^5$, existem escalares $\alpha_1,\alpha_2,\alpha_3,\alpha_4,\alpha_5 \in \mathbb{Z}_2$, tal que

$$(x_1,x_2,x_3,x_4,x_5) = \alpha_1(1,0,1,1,0) + \alpha_2(0,1,0,1,1) + \alpha_3(0,0,1,0,0) + \\ \alpha_4(0,0,0,1,0) + \alpha_5(0,0,0,0,1).$$

Daí segue que $(x_1,x_2,x_3,x_4,x_5) = (\alpha_1,\alpha_2,\alpha_1+\alpha_3,\alpha_1+\alpha_2+\alpha_4,\alpha_2+\alpha_5)$. Concluindo que,

$\alpha_1 = x_1$

$\alpha_2 = x_2$

$\alpha_3 = x_3 - x_1$

$\alpha_4 = x_4 - x_1 - x_2$

$\alpha_5 = x_5 - x_2$

Logo,

$$(x_1,x_2,x_3,x_4,x_5) = x_1(1,0,1,1,0) + x_2(0,1,0,1,1) + (x_3-x_1)(0,0,1,0,0) + \\ (x_4-x_1-x_2)(0,0,0,1,0) + (x_5-x_2)(0,0,0,0,1).$$

Aplicando a transformação linear T na expressão anterior, temos

$$T(x_1,x_2,x_3,x_4,x_5) = x_1T(1,0,1,1,0) + x_2T(0,1,0,1,1) + (x_3-x_1)T(0,0,1,0,0) + \\ (x_4-x_1-x_2)T(0,0,0,1,0) + (x_5-x_2)T(0,0,0,0,1).$$

Como queremos uma transformação linear cujo núcleo é o código, segue que $T(w) = 0$, se $w \in C$ e tomando $T(w) = w$, se $w \notin C$, temos

$$T(x_1,x_2,x_3,x_4,x_5) = x_1(0,0,0,0,0) + x_2(0,0,0,0,0) + (x_3-x_1)(0,0,1,0,0) + \\ (x_4-x_1-x_2)(0,0,0,1,0) + (x_5-x_2)(0,0,0,0,1).$$

Portanto, sendo $T : (\mathbb{Z}_2)^5 \longrightarrow (\mathbb{Z}_2)^5$ a transformação linear definida por

$$T(x_1,x_2,x_3,x_4,x_5) = (0,0,x_3-x_1,x_4-x_1-x_2,x_5-x_2).$$

teremos $C = N(T)$.

Definição 5.2. *Seja $C(n,k)$ um código linear sobre $\mathbb{F}_q$. O código dual de C é o código linear $C^{\perp}(n,n-k)$ sobre $\mathbb{F}_q$.*

Exemplo 5.6. Considere o código

$$C = \{(0,0,0,0,0),(0,1,0,1,1),(1,0,1,1,0),(1,1,1,0,1)\} \subsetneq (\mathbb{Z}_2)^5.$$

O código $C^{\perp}(5,3)$ é linear.

De fato, tomemos $\{(0,1,0,1,1),(1,0,1,1,0),(1,1,1,0,1)\}$ uma base de C. Assim, a mensagem $(x_1,x_2,x_3,x_4,x_5) \in (\mathbb{Z}_2)^5$ pertence a $C^{\perp}$, se e só se,

$$\begin{cases} x_2 + x_4 + x_5 = 0 \\ x_1 + x_3 + x_4 = 0 \\ x_1 + x_2 + x_3 + x_5 = 0 \end{cases}$$

Logo, $C^{\perp} = \{(0,0,0,0,0),(1,0,1,0,0),(1,1,0,1,0),(0,1,1,1,0),(0,1,0,0,1),$ $(1,0,0,1,1),(1,1,1,0,1),(0,0,1,1,1)\}$.

5.3 Matriz geradora de um código linear

As palavras-código de um código linear podem ser obtidas a partir de uma matriz especial, a qual apresentaremos a seguir.

Definição 5.3. *Sejam $C(n,k)$ um código linear sobre $\mathbb{F}_q$ e $\mathcal{B} = \{v_1, v_2, ..., v_k\}$ uma base de C. A matriz G de ordem $k \times n$ cujas linhas são $v_1, v_2, ..., v_k$ é denominada matriz geradora do código linear C.*

Observação 5.4. Identifica-se $(x_1, x_2, ..., x_k) \in (\mathbb{F}_q)^k$ com a matriz linha $\begin{pmatrix} x_1 & x_2 & ... & x_k \end{pmatrix}$.

Proposição 5.2. *Se G é uma matriz geradora de um código linear $C(n,k)$ sobre $\mathbb{F}_q$, então*

$$C = \{x \cdot G : x \in (\mathbb{F}_q)^k\}.$$

Exemplo 5.7. Dados $A = \{0,1\} = \mathbb{Z}_2$, $n = 5$. Considere o código linear $C(5,2)$, tal que

$$C = \{(0,0,0,0,0),(0,1,0,1,1),(1,0,1,1,0),(1,1,1,0,1)\}.$$

O conjunto $\{(1,0,1,1,0),(0,1,0,1,1)\}$ é uma base de C.

Logo, a matriz geradora de C é

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

Pela proposição 5.2 as palavras-código são obtidas assim:

$$\begin{pmatrix} 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$
$$\begin{pmatrix} 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \end{pmatrix},$$
$$\begin{pmatrix} 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Observação 5.5. No exemplo 5.7 para qualquer $x = (x_1, x_2) \in (\mathbb{Z}_2)^2$ e dado $y = (y_1, y_2, y_3, y_4, y_5) \in (\mathbb{Z}_2)^5$, para averiguar se y é uma palavra-código e, caso seja, decodificar y basta resolver o sistema

$x \cdot G = y \Leftrightarrow$

$$\begin{cases} x_1 = y_1 \\ x_2 = y_2 \\ x_1 = y_3 \\ x_1 + x_2 = y_4 \\ x_2 = y_5 \end{cases}$$

Exemplo 5.8. Dado o corpo finito $\mathbb{Z}_2$, considere a transformação linear e injetora

$$\begin{aligned} T: \quad (\mathbb{Z}_2)^3 &\longrightarrow (\mathbb{Z}_2)^5 \\ (x_1, x_2, x_3) &\longmapsto (x_1, x_3, x_1 + x_2, x_2 + x_3, x_2). \end{aligned}$$

Tomemos o código linear $C(5, 3)$ tal que $C = Im(T)$.

Sejam $\{e_1, e_2, e_3\}$ a base canônica de $(\mathbb{Z}_2)^3$ e $\{f_1, f_2, f_3, f_4, f_5\}$ a base canônica de $(\mathbb{Z}_2)^5$.

Vamos determinar a matriz geradora G do código C. Para isso, fazemos

$$\begin{aligned} T(1,0,0) &= (1,0,1,0,0) = 1f_1 + 0f_2 + 1f_3 + 0f_4 + 0f_5, \\ T(0,1,0) &= (0,0,1,1,1) = 0f_1 + 0f_2 + 1f_3 + 1f_4 + 1f_5, \\ T(0,0,1) &= (0,1,0,1,0) = 0f_1 + 1f_2 + 0f_3 + 1f_4 + 0f_5. \end{aligned}$$

Logo, o conjunto $\{(1,0,1,0,0), (0,0,1,1,1), (0,1,0,1,0)\}$ é uma base de C e a matriz geradora é

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

As palavras-código são obtidas assim:

$$\begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\begin{pmatrix}0 & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}0 & 1 & 0 & 1 & 0\end{pmatrix},$$
$$\begin{pmatrix}0 & 1 & 0\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}0 & 0 & 1 & 1 & 1\end{pmatrix},$$
$$\begin{pmatrix}0 & 1 & 1\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}0 & 1 & 1 & 0 & 1\end{pmatrix},$$
$$\begin{pmatrix}1 & 0 & 0\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}1 & 0 & 1 & 0 & 0\end{pmatrix},$$
$$\begin{pmatrix}1 & 0 & 1\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}1 & 1 & 1 & 1 & 0\end{pmatrix},$$
$$\begin{pmatrix}1 & 1 & 0\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}1 & 0 & 0 & 1 & 1\end{pmatrix},$$
$$\begin{pmatrix}1 & 1 & 1\end{pmatrix}\cdot\begin{pmatrix}1 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}1 & 1 & 0 & 0 & 1\end{pmatrix}.$$

Portanto, obtemos o código $C=\{(0,0,0,0,0),(0,1,0,1,0),(0,0,1,1,1),(0,1,1,0,1),(1,0,1,0,0),(1,1,1,1,0),(1,0,0,1,1),(1,1,0,0,1)\}$.

Exemplo 5.9. Dado o corpo finito $\mathbb{Z}_2$, considere o código linear $C\subset(\mathbb{Z}_2)^5$ definido pela transformação linear injetora

$$\begin{array}{rccc} T: & (\mathbb{Z}_2)^3 & \longrightarrow & (\mathbb{Z}_2)^5\\ & (x_1,x_2,x_3) & \longmapsto & (x_1,x_2,x_3,x_1+x_3,x_1+x_2).\end{array}$$

Sejam $\{e_1,e_2,e_3\}$ a base canônica de $(\mathbb{Z}_2)^3$ e $\{f_1,f_2,f_3,f_4,f_5\}$ a base canônica de $(\mathbb{Z}_2)^5$.

Vamos determinar a matriz geradora G do código C. Para isso, fazemos

$$\begin{aligned}T(1,0,0)&=(1,0,0,1,1)=1f_1+0f_2+0f_3+1f_4+1f_5\\ T(0,1,0)&=(0,1,0,0,1)=0f_1+1f_2+0f_3+0f_4+1f_5\\ T(0,0,1)&=(0,0,1,1,0)=0f_1+0f_2+1f_3+1f_4+0f_5\end{aligned}$$

Logo, o conjunto $\{(1,0,0,1,1),(0,1,0,0,1),(0,0,1,1,0)\}$ é uma base de C e a matriz geradora é

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

Note que a matriz geradora desse código, apresenta-se de forma especial. As três primeiras linhas e as três primeiras colunas formam a matriz identidade de ordem 3.

As palavras-código são obtidas assim:

$$\begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \end{pmatrix},$$

$$\begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \end{pmatrix},$$

$$\begin{pmatrix} 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \end{pmatrix},$$

$$\begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \end{pmatrix},$$

$$\begin{pmatrix} 1 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \end{pmatrix},$$

$$\begin{pmatrix} 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \end{pmatrix},$$

$$\begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Portanto, obtemos o código $C = \{(0,0,0,0,0), (0,0,1,1,0), (0,1,0,0,1), (0,1,1,1,1), (1,0,0,1,1), (1,0,1,0,1), (1,1,0,1,0), (1,1,1,0,0)\}$.

Dada uma palavra-código, observe que as três primeiras coordenadas são os bits de informação, e portanto as duas últimas coordenadas os bits de verificação de paridade. Em casos como esse, fica muito mais fácil interpretar a informação enviada. Se recebermos a palavra-código $(1,1,0,1,0)$, então a mensagem enviada foi $(1,1,0)$.

Definição 5.4. *Uma matriz geradora de um código linear $C(n,k)$ está na forma padrão se tiver a forma $(I_k|P)$, onde I_k é a matriz identidade de ordem k e P é a submatriz de paridade de ordem $(n-k) \times k$.*

Definição 5.5. *Um código linear é sistemático se possui uma matriz geradora na forma padrão.*

Exemplo 5.10. Considere o código linear sistemático $C(7,4)$, cuja matriz geradora G é dada por

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Dada a informação $(1,1,0,1)$, fazendo

$$\begin{pmatrix} 1 & 1 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix},$$

obtemos a palavra-código $(1,1,0,1,0,0,0)$.

Observação 5.6. Dado um código C nem sempre é possível obter uma matriz geradora de C na forma padrão.

Exemplo 5.11. Considere o código linear $C(5,3)$ cuja matriz geradora é

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Não é possível, efetuando-se operações elementares nas linhas de G, obter uma matriz na forma $(I_3|P)$. Logo, não existe uma matriz geradora de C na forma padrão.

5.4 Matriz de verificação de paridade

Ao recebermos uma mensagem, podemos utilizar uma outra matriz, que será apresentada a seguir, para verificar se a mensagem recebida é ou não uma palavra-código.

Definição 5.6. *Seja $C(n,k)$ um código linear sobre $\mathbb{F}_q$. Uma matriz H de ordem $(n-k) \times n$ com $n-k$ linhas linearmente independentes é denominada matriz de verificação de paridade do código C se*

$$C = \{y \in (\mathbb{F}_q)^n : H \cdot y^t = 0\}.$$

Em outras palavras, para saber se uma mensagem y é uma palavra-código devemos verificar se suas respectivas entradas satisfazem as condições impostas para os dígitos de paridade, isto é, devemos ter $H \cdot y^t = 0$.

Proposição 5.3. *Seja $C(n,k)$ um código linear com matriz geradora G. Uma matriz H com n colunas e $n-k$ linhas linearmente independentes, é uma matriz de verificação de paridade para C se, e somente se, $H \cdot G^t = 0$.*

Demonstração. Se H é uma matriz de verificação de paridade para C, então $H \cdot G^t \cdot x^t = 0$, para todo $x \in (\mathbb{F}_q)^k$. Fazendo x percorrer a base canônica de $(\mathbb{F}_q)^k$ conclui-se que as colunas de $H \cdot G^t$ são nulas, ou seja, $H \cdot G^t = 0$.

Reciprocamente, suponha que $H \cdot G^t = 0$.

Dado $y \in C$, existe $x \in (\mathbb{F}_q)^k$ tal que $y = x \cdot G$. Então, $H \cdot y^t = H \cdot G^t \cdot x^t = 0$. Logo,

$$C \subseteq \{y \in (\mathbb{F}_q)^n : H \cdot y^t = 0\}.$$

Por outro lado, já que a matriz H possui $n-k$ linhas linearmente independentes, o sistema homogêneo $H \cdot y^t = 0$ tem k variáveis livres e, por isso, temos

$$|\{y \in (\mathbb{F}_q)^n : H \cdot y^t = 0\}| = q^n.$$

Como $C(n,k)$ um código linear sobre $\mathbb{F}_q$, segue que $|C| = q^n$.

Portanto, $C = \{y \in (\mathbb{F}_q)^n : H \cdot y^t = 0\}$ e H é uma matriz de verificação de paridade para C. □

Exemplo 5.12. Para o código linear $C(5,3)$ do exemplo 5.9 definimos os dígitos de paridade assim:

$$x_4 = x_1 + x_3 \text{ e } x_5 = x_1 + x_2.$$

Reescrevendo as duas expressões obtemos:

$$x_1 + x_3 + x_4 = 0 \text{ e } x_1 + x_2 + x_5 = 0.$$

Daí teremos a matriz verificação de paridade

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

Assim sendo, note que $y = (1,0,0,1,1)$ é uma palavra-código e $y = (1,0,1,0,1)$ não é uma palavra-código, pois

$$\begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

$$\begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Exemplo 5.13. A matriz verificação de paridade

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

define o código linear $C(4,2)$. A mensagem (x_1, x_2) é codificada como a palavra-código $y = (y_1, y_2, y_3, y_4)$, onde

$$x_1 = y_1 \text{ e } x_2 = y_2.$$

Os dígitos de verificação de paridade y_3 e y_4 são obtidos da expressão $H \cdot y^t = 0$, ou seja,

$$\begin{aligned} y_1 + y_3 &= 0 \\ y_1 + y_2 + y_4 &= 0. \end{aligned}$$

Para esse código teremos $2^2 = 4$ palavras-código. São elas: $(0,0,0,0)$, $(0,1,0,1)$, $(1,0,1,1)$ e $(1,1,1,0)$.

Proposição 5.4. *Seja $C(n,k)$ um código linear. Se x e y são palavras-código, então $x + y$ e cx também são palavras-código.*

Demonstração. Se x e y são palavras-código, então $H \cdot x^t = 0$ e $H \cdot y^t = 0$.

Logo, $H \cdot (x+y)^t = H \cdot (x^t + y^t) = H \cdot x^t + H \cdot y^t = 0 + 0 = 0$ e $H \cdot (c \cdot x)^t = H \cdot c \cdot x^t = c \cdot H \cdot x^t = c \cdot 0 = 0$. □

Definição 5.7. *Seja $C(n,k)$ um código binário linear sistemático com matriz geradora $G = (I_k|P)$. A matriz $H = (P^t|I_{n-k})$ é uma matriz de verificação de paridade para C.*

Exemplo 5.14. Considere o código linear $C(5,2)$

$C = \{(0,0,0,0,0), (0,1,0,1,1), (1,0,1,1,0), (1,1,1,0,1)\}$.

O conjunto $\{(1,0,1,1,0), (0,1,0,1,1)\}$ é uma base de C.

Logo, uma matriz geradora de C é

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

Portanto, a matriz

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

é uma matriz de verificação de paridade para C.

Exemplo 5.15. Considere o corpo finito $\mathbb{Z}_2$. Dada a transformação linear

$$\begin{aligned} T: \quad (\mathbb{Z}_2)^3 &\longrightarrow (\mathbb{Z}_2)^2 \\ (x_1, x_2, x_3) &\longmapsto (x_1 + x_2, x_3). \end{aligned}$$

cujo núcleo é $C = N(T) = \{(x_1, x_1, 0); x_1 \in \mathbb{Z}_2\}$.

Agora considere as bases canônicas $\{e_1, e_2, e_3\}$ e $\{f_1, f_2\}$ de $(\mathbb{Z}_2)^3$ e $(\mathbb{Z}_2)^2$ respectivamente.

Vamos determinar a matriz H que representa a transformação linear T nessas bases.

Logo,

$$\begin{aligned} T(1,0,0) &= (1,0) = 1f_1 + 0f_2, \\ T(0,1,0) &= (1,0) = 1f_1 + 0f_2, \\ T(0,0,1) &= (0,1) = 0f_1 + 1f_2. \end{aligned}$$

Portanto, a matriz verificação de paridade é

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

Qualquer $y \in (\mathbb{Z}_2)^3$ pertence ao código C se a condição $H \cdot y^t = 0$ é válida.

Dados $x = (1,1,1)$ e $y = (1,1,0) \in (\mathbb{Z}_2)^3$, como

$$H \cdot x^t = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

e

$$H \cdot y^t = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

temos que $x \notin C$ e $y \in C$.

5.5 Códigos de Hamming sob o ponto de vista matricial

Na seção 4.1 apresentamos os códigos de Hamming como um código corretor de erro enfatizando a distância mínima como ferramenta de detecção de erros. Nesta seção abordaremos esses códigos novamente, porém utilizando a matriz verificação de paridade para detectar um único erro na transmissão de informações.

Um código de Hamming de ordem $m \geqslant 2$ é um $C(2^m - 1, 2^m - m - 1)$ código linear que possui uma matriz de verificação de paridade com m linhas, H_m, cujas colunas são todos os elementos não nulos de $(\mathbb{Z}_2)^m$, dispostos em qualquer ordem.

Exemplo 5.16. Considere o código de Hamming de ordem 3, ou seja, o código $C(7,4)$ linear, cuja matriz verificação de paridade é

$$H_3 = (P_{3\times 4}|I_3) = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Logo, a matriz geradora correspondente será

$$(I_4|P_{4\times 3}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Portanto, dado $x \in (\mathbb{Z}_2)^4$, a sua codificação fica sendo $(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$, onde

$$\begin{aligned} x_5 &= x_2 + x_3 + x_4 \\ x_6 &= x_1 + x_3 + x_4 \\ x_7 &= x_1 + x_2 + x_4 \end{aligned}$$

Vamos agora utilizar a matriz verificação de paridade H para identificar um possível erro em uma das posições dos bits na mensagem recebida e assim, recuperarmos a mensagem enviada.

Suponha que a palavra-código x é enviada e a mensagem r é recebida e que ocorreu um erro na i-ésima componente de x, trocando zero por um ou vice-versa. Então podemos escrever

$$r = x + e_i$$

onde o vetor e_i possui zeros em todas as componentes exceto na i-ésima posição.

Exemplo 5.17. Se $x = (0,1,1,1,1,0,0)$ e $r = (0,0,1,1,1,0,0)$, então

$$e_i = x + r = (0,1,1,1,1,0,0) + (0,0,1,1,1,0,0) = (0,1,0,0,0,0,0).$$

Neste caso o erro ocorreu na 2ª posição.

O problema agora consiste em saber se é possível recuperarmos a mensagem enviada x quando conhecemos a mensagem recebida r. Observe que saber em qual posição ocorreu o erro é suficiente para determinar a mensagem x, pois $r = x + e_i$ implica em $x = r + e_i$, já que nossos códigos são binários. De fato,

$$H \cdot r^t = H \cdot (x + e_i)^t = H \cdot x^t + H \cdot {e_i}^t.$$

Sendo x uma palavra-código, temos que $H \cdot x^t = 0$. Logo,

$$H \cdot r^t = 0 + H \cdot {e_i}^t = H \cdot {e_i}^t.$$

Portanto, se temos a mensagem recebida r, pode-se obter $H \cdot {e_i}^t$. Por outro lado,

$$H \cdot {e_i}^t = H \cdot \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Assim, $H \cdot {e_i}^t$ é a i-ésima coluna da matriz H. E daí pode-se saber a localização do único erro e consequentemente recuperar x a partir de r.

Exemplo 5.18. Código $C(7,4)$. Considere a matriz verificação de paridade de ordem 3×7 dada por

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

Como o código é construído de modo que as palavras-código satisfaçam $H \cdot x^t = 0$ e essa matriz possui 3 colunas linearmente independentes, segue que $dim C = 7 - 3 = 4$. Assim sendo, temos o código $C(7,4)$ com $2^4 = 16$ palavras-código.

Consideremos que um único erro ocorre na transmissão de uma mensagem e que o vetor recebido é $r = (0,0,1,1,1,0,0)$.

Logo, $H \cdot r^t = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, que é a 2ª coluna da matriz H, ou seja, o erro se encontra na 2ª posição. Assim sendo, $e_2 = (0,1,0,0,0,0,0)$.

Portanto, o vetor recebido pode ser decodificado como a palavra-código

$$x = r + e_2 = (0,1,1,1,1,0,0).$$

Neste capítulo vimos que os códigos lineares podem ser estudados sob o ponto de vista matricial, fazendo uso da matriz geradora para obtermos todas as palavras-código e também utilizando a matriz de paridade para verificar se uma determinada informação pertence ao código. No próximo capítulo também abordaremos o estudo das matrizes como aplicação na codificação e decodificação de mensagens secretas.

6 Aplicações no Ensino Médio

Neste capítulo abordaremos a relação do estudo da codificação com alguns conteúdos do Ensino Médio: multiplicação de matrizes, matriz inversa e sistemas de equações lineares. Essa abordagem evidencia o aspecto motivacional ao trabalhar com esses conteúdos, uma vez que os alunos estarão contextualizando a aprendizagem com situações reais de seu cotidiano ligadas à área da tecnologia, tais como: troca de informações através da internet, transações financeiras, telecomunicações, sinais digitais, armazenamento de informações em computadores, pen drives, entre outros.

6.1 Mensagens secretas com matrizes

Para iniciarmos um processo de codificação precisamos do alfabeto fonte e do alfabeto código. Para nossa primeira atividade usaremos os alfabetos da figura 6.1.

A	B	C	D	E	F	G	H	I	J
$\begin{pmatrix}0\\0\end{pmatrix}$	$\begin{pmatrix}1\\0\end{pmatrix}$	$\begin{pmatrix}2\\0\end{pmatrix}$	$\begin{pmatrix}3\\0\end{pmatrix}$	$\begin{pmatrix}4\\0\end{pmatrix}$	$\begin{pmatrix}0\\1\end{pmatrix}$	$\begin{pmatrix}1\\1\end{pmatrix}$	$\begin{pmatrix}2\\1\end{pmatrix}$	$\begin{pmatrix}3\\1\end{pmatrix}$	$\begin{pmatrix}4\\1\end{pmatrix}$
K	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**
$\begin{pmatrix}0\\2\end{pmatrix}$	$\begin{pmatrix}1\\2\end{pmatrix}$	$\begin{pmatrix}2\\2\end{pmatrix}$	$\begin{pmatrix}3\\2\end{pmatrix}$	$\begin{pmatrix}4\\2\end{pmatrix}$	$\begin{pmatrix}0\\3\end{pmatrix}$	$\begin{pmatrix}1\\3\end{pmatrix}$	$\begin{pmatrix}2\\3\end{pmatrix}$	$\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}4\\3\end{pmatrix}$
U	**V**	**W**	**X**	**Y**	**Z**	**espaço**	**.**	**,**	**?**
$\begin{pmatrix}0\\4\end{pmatrix}$	$\begin{pmatrix}1\\4\end{pmatrix}$	$\begin{pmatrix}2\\4\end{pmatrix}$	$\begin{pmatrix}3\\4\end{pmatrix}$	$\begin{pmatrix}4\\4\end{pmatrix}$	$\begin{pmatrix}0\\5\end{pmatrix}$	$\begin{pmatrix}1\\5\end{pmatrix}$	$\begin{pmatrix}2\\5\end{pmatrix}$	$\begin{pmatrix}3\\5\end{pmatrix}$	$\begin{pmatrix}4\\5\end{pmatrix}$

Figura 6.1: Alfabeto fonte e alfabeto código para códigos com matrizes

Observação 6.1. Cada letra do alfabeto fonte corresponde a um par de números (matriz coluna) no alfabeto código. Cada par de números será, portanto uma coluna da matriz mensagem.

Sequência Didática I

A sequência didática que se segue é uma adaptação do conteúdo apresentado na referência [13], onde utilizo as ideias apresentadas para fazer a codificação de mensagens, porém enfatizo no processo de decodificação a construção do raciocínio dos alunos sobre matriz inversa, uma vez que eles ainda não sabem nada a respeito de inversão de matrizes.

I) Processo de codificação

O professor deverá escolher uma mensagem para enviar aos seus alunos.

Para codificar essa mensagem utilizando matrizes seguimos os passos:

1) Escreva a matriz M que representa a mensagem que deseja enviar de acordo com a figura 6.1.

2) Escolher uma matriz quadrada C invertível. Considere a matriz C quadrada de ordem 2 dada por

$$C = \begin{pmatrix} 3 & 1 \\ 2 & 1 \end{pmatrix}.$$

Essa matriz C será chamada de *matriz codificadora*.

3) A mensagem codificada M_C será representada pelo produto da matriz C pela matriz M, ou seja,

$$M_C = C \cdot M.$$

Os alunos receberão a mensagem codificada e precisará saber como reverter o processo descrito anteriormente para poder decifrá-la. Nesse momento eles ainda não tem conhecimento sobre matriz inversa e o professor deverá deixar seus alunos fazerem tentativas, levantar hipóteses e estratégias diferenciadas para conseguir decifrar a mensagem. Durante esta etapa, o professor pode fazer a seguinte colocação: *"Encontre uma matriz quadrada de ordem 2 que quando multiplicada pela mensagem codificada M_C resulte na mensagem M"*. Deste modo, os alunos estarão resolvendo sistemas de equações lineares de ordem 2.

Assim que os alunos encontrarem a matriz inversa de C, que representaremos por C^{-1} e conseguirem decodificar a mensagem, o professor pede para os alunos analisarem a relação que existe entre essas duas matrizes, fazendo os produtos $C \cdot C^{-1}$ e $C^{-1} \cdot C$. A partir dessa relação, o aluno terá a definição de matriz inversa e saberá, então que precisa da matriz inversa para decodificar a mensagem codificada.

I) Processo de decodificação

Os alunos receberão a mensagem codificada de seu professor.

Para decodificar essa mensagem os alunos deverão multiplicar a matriz C^{-1} pela matriz M_C, obtendo assim a mensagem M.

A matriz C^{-1} será chamada de *matriz decodificadora* e neste exemplo sugerido ela é dada por

$$C^{-1} = \begin{pmatrix} 1 & -1 \\ -2 & 3 \end{pmatrix}.$$

De modo geral, podemos justificar matematicamente o processo de decodificação, fazendo

$$M = (C^{-1} \cdot C) \cdot M = C^{-1} \cdot (C \cdot M) = C^{-1} \cdot M_C.$$

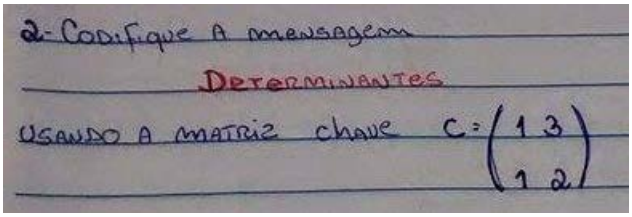
Na segunda etapa dessa sequência didática, o professor orienta seus alunos, que deverão estar divididos em duplas, a codificarem uma mensagem e trocarem com outra dupla de alunos para fazerem a decodificação. Neste momento pode-se questionar se qualquer matriz quadrada de ordem 2 pode ser usada como matriz codificadora, ou seja, questionando se toda matriz é invertível.

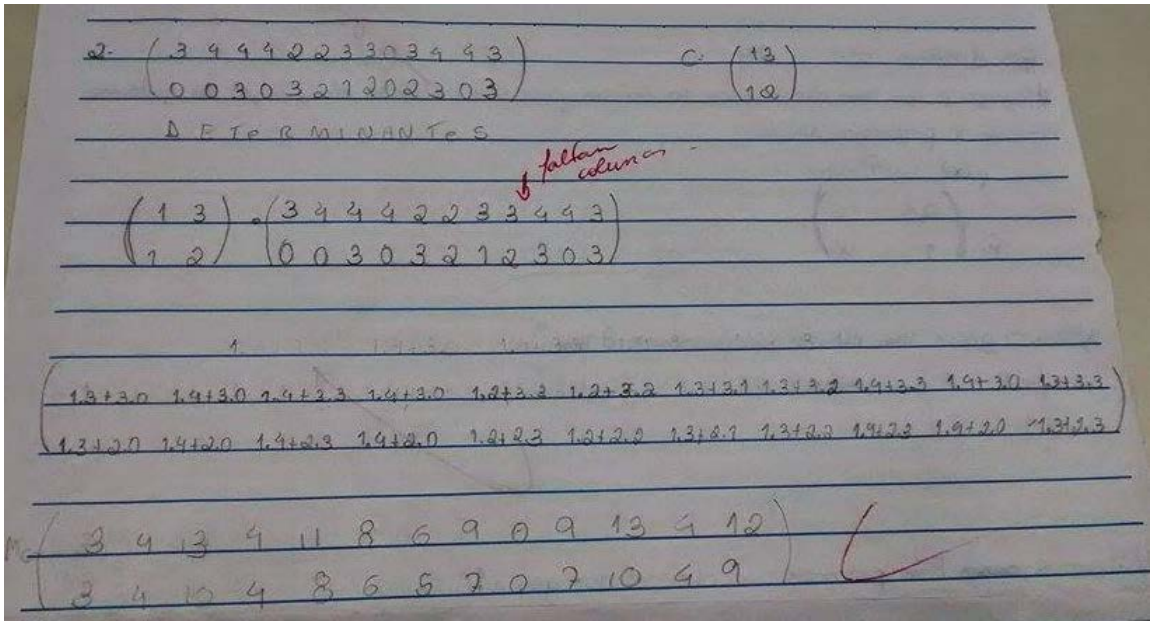
Relatos da sequência didática I: Aplicação em sala de aula

A sequência didática I foi aplicada, por mim, aos alunos da segunda série do ensino médio, na escola onde leciono desde o ano de 2008.

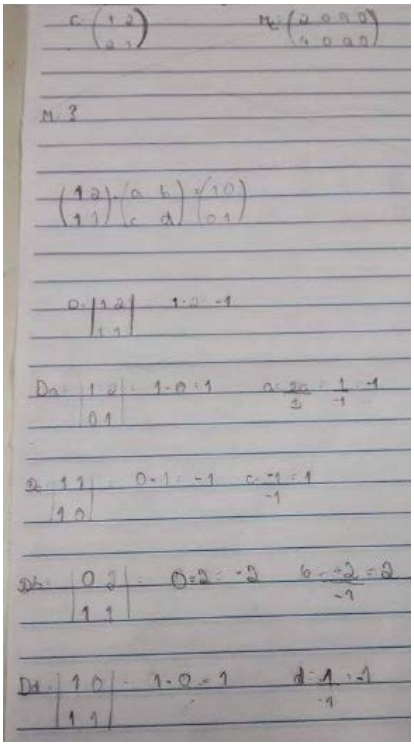
As atividades foram propostas em etapas, de modo que os alunos tivessem condições de construir o raciocínio para a compreensão do estudo da matriz inversa. Primeiramente, eles escolheram uma mensagem para codificar, utilizando como chave a matriz escolhida por mim. Neste momento, os educandos perceberam que a multiplicação de matrizes foi utilizada como ferramenta de aplicação para o processo de codificação das mensagens. Em seguida, eles receberam uma mensagem codificada por mim com o objetivo de decifrá-la. Como os alunos ainda não conheciam matriz inversa, eles levantaram hipóteses e definiram estratégias para tentar encontrar uma matriz que pudesse ser a chave decodificadora. Após a socialização das estratégias realizadas nos pequenos grupos e as orientações feitas por mim, eles conseguiram encontrar, coletivamente, uma matriz que desempenhasse o papel de decifrar a mensagem que eles haviam recebido. A partir daí, fui questionando-os da relação que há entre as duas matrizes utilizadas como chave codificadora e decodificadora, e então compreender a necessidade de aprofundar os estudos sobre matriz inversa. Como utilizamos matrizes de ordem 2, estudamos como encontrar a matriz inversa resolvendo sistema de equações lineares e também pela Regra de Crammer. Por fim, os alunos escolheram uma matriz chave qualquer e trocaram as mensagens entre si, para que pudessem obter a matriz inversa e decifrar a mensagem. Nesta etapa, os alunos puderam perceber que determinar a matriz inversa pode ser uma tarefa nem sempre fácil, principalmente quando eles perceberam que precisariam utilizar números racionais. Notou-se também que nem toda matriz possui inversa. Tudo isso, foi realizado dando enfoque ao educando como protagonista de seu aprendizado, ou seja, um aprendizado significativo e relevante para suas argumentações em sala de aula.

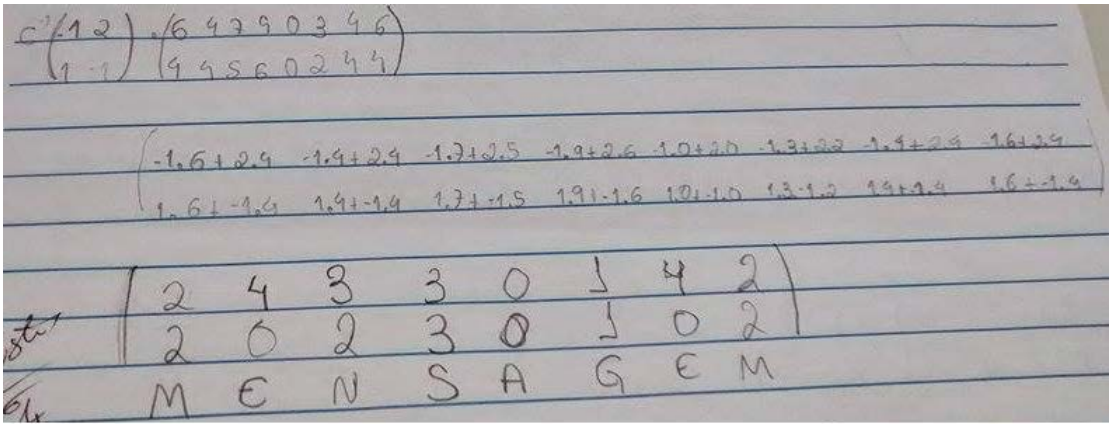
As figuras a seguir são registros das atividades realizadas pelos alunos na etapa codificando com matrizes. Nesta etapa os alunos multiplicaram a matriz codificadora pela matriz mensagem obtendo a matriz codificada. Com essa atividade os alunos estão aprimorando as habilidades relacionadas à multiplicação de matrizes, uma vez que eles já tinham conhecimentos prévios sobre este conteúdo.





As figuras a seguir são registros das atividades realizadas pelos alunos na etapa decodificando com matrizes. Neta os alunos ainda não tinham conhecimento sobre matriz inversa. Deste modo, eles foram orientados a elaborar estratégias, levantar hipóteses de como obter uma matriz que pudesse decodificar a mensagem recebida. Com essa atividade os alunos também desenvolveram a habilidade de multiplicação de matrizes, operações com números inteiros, determinantes e resolução de sistemas de equações lineares de ordem 2 pela Regra de Cramer.





Sequência Didática II

Nesta sequência didática vamos abordar os processos de codificação e decodificação com matrizes da mesma forma que foi realizado na atividade anterior, porém utilizaremos outros alfabetos fonte e código. Assim, podemos abordar esta temática utilizando alfabetos diferentes, possibilitando também que os próprios alunos construam seus alfabetos para a codificação. Vamos, então exemplificar o processo de codificação com matrizes utilizando o alfabeto fonte e o alfabeto código representado na tabela a seguir:

A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	2	3	4	5	6	7	8	9	10	11	12	13	14

O	P	Q	R	S	T	U	V	W	X	Y	Z	.	/
15	16	17	18	19	20	21	22	23	24	25	26	27	28

Tabela 6.1: Alfabeto fonte e alfabeto código para códigos com matrizes

Observação 6.2. O símbolo / será utilizado para separar as palavras na mensagem.

Vamos codificar a mensagem: OS NÚMEROS GOVERNAM O MUNDO. Fazendo a correspondência de acordo com a tabela 6.1 teremos:

O	S	/	N	U	M	E	R	O	S	/	G	O	V
15	19	28	14	21	13	5	18	15	19	28	7	15	22
E	R	N	A	M	/	O	/	M	U	N	D	O	.
5	18	14	1	13	28	15	28	13	21	14	4	15	27

Utilizaremos a seguinte matriz codificadora C quadrada de ordem 2:

$$C = \begin{pmatrix} 3 & 1 \\ 2 & 1 \end{pmatrix}.$$

Como cada símbolo da mensagem corresponde a um número e a matriz codificadora possui ordem 2, colocaremos esses números de 2 em 2, por colunas, em uma outra matriz, obtendo-se assim uma matriz mensagem M de ordem 2×14:

$$M = \begin{pmatrix} 15 & 28 & 21 & 5 & 15 & 28 & 15 & 5 & 14 & 13 & 15 & 13 & 14 & 15 \\ 19 & 14 & 13 & 18 & 19 & 7 & 22 & 18 & 1 & 28 & 28 & 21 & 4 & 27 \end{pmatrix}.$$

Para obtermos a mensagem codificada M_C fazemos o produto $C \cdot M$.

Logo,

$$M_C = \begin{pmatrix} 64 & 98 & 76 & 33 & 64 & 91 & 67 & 33 & 43 & 67 & 73 & 60 & 46 & 72 \\ 49 & 70 & 55 & 28 & 49 & 63 & 52 & 28 & 29 & 54 & 58 & 47 & 32 & 57 \end{pmatrix}.$$

Portanto, a mensagem codificada é:

$64 - 49 - 98 - 70 - 76 - 55 - 33 - 28 - 64 - 49 - 91 - 63 - 67 - 52 - 33 - 28 - 43 - 29 - 67 - 54 - 73 - 58 - 60 - 47 - 46 - 32 - 72 - 57.$

Ao receber essa mensagem, fazemos o produto $C^{-1} \cdot M_C$ para decodificá-la, isto é, $\begin{pmatrix} 1 & -1 \\ -2 & 3 \end{pmatrix} \cdot \begin{pmatrix} 64 & 98 & 76 & 33 & 64 & 91 & 67 & 33 & 43 & 67 & 73 & 60 & 46 & 72 \\ 49 & 70 & 55 & 28 & 49 & 63 & 52 & 28 & 29 & 54 & 58 & 47 & 32 & 57 \end{pmatrix}$.

Logo, obtemos a matriz

$$M = \begin{pmatrix} 15 & 28 & 21 & 5 & 15 & 28 & 15 & 5 & 14 & 13 & 15 & 13 & 14 & 15 \\ 19 & 14 & 13 & 18 & 19 & 7 & 22 & 18 & 1 & 28 & 28 & 21 & 4 & 27 \end{pmatrix},$$

ou seja, a mensagem decodificada é:

$15 - 19 - 28 - 14 - 21 - 13 - 5 - 18 - 15 - 19 - 28 - 7 - 15 - 22 - 5 - 18 - 14 - 1 - 13 - 28 - 15 - 28 - 13 - 21 - 14 - 4 - 15 - 27.$

Novamente de acordo com a tabela 6.1 teremos a mensagem: OS NÚMEROS GOVERNAM O MUNDO.

Os objetivos traçados para essas sequências didáticas consistem no reconhecimento da aplicação de multiplicação de matrizes para o processo de codificação e também das técnicas utilizadas para obter a matriz inversa.

A sequência didática II foi aplicada na forma de avaliação para os mesmos alunos que apliquei a sequência didática I, onde avaliei se o aluno adquiriu a habilidade de multiplicar matrizes. A maioria dos alunos conseguiu fazer a multiplicação de matrizes e decodificar a mensagem. Na avaliação desta habilidade o que leva alguns alunos a cometerem erros é não seguir a ordem correta das operações ao multiplicar matrizes, ou seja, operar cada linha da primeira matriz com todas as colunas da segunda matriz.

Abaixo segue o registro da avaliação realizada.

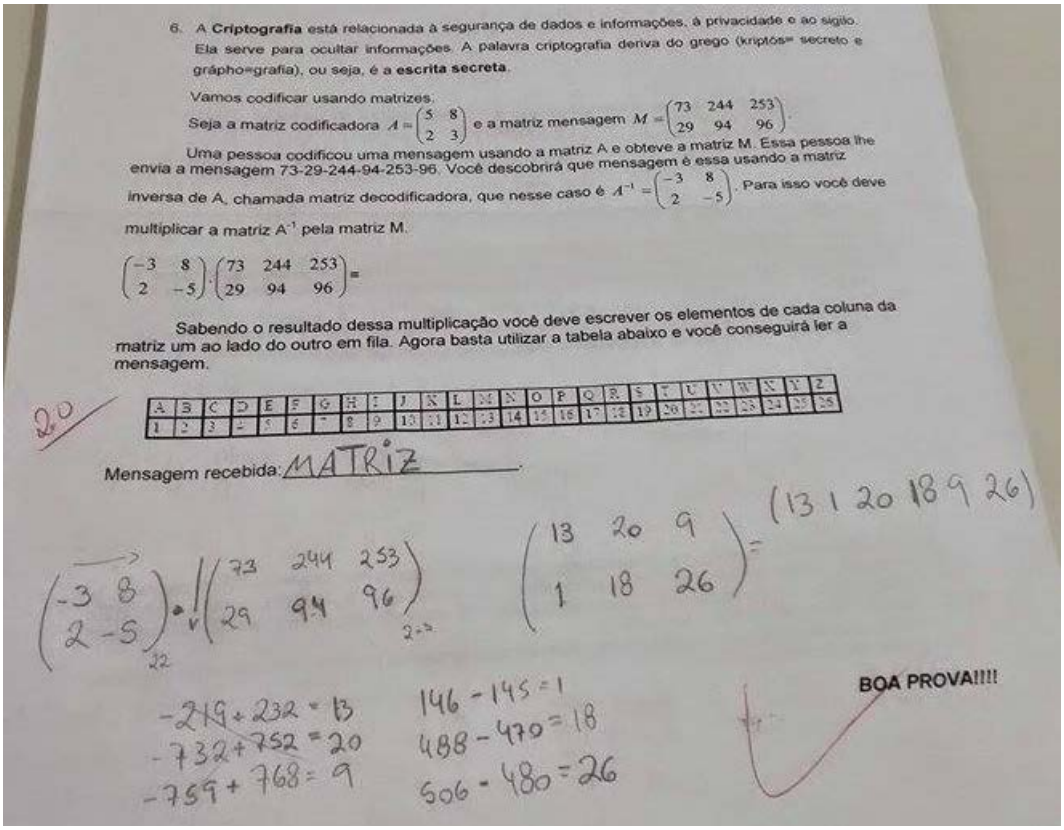
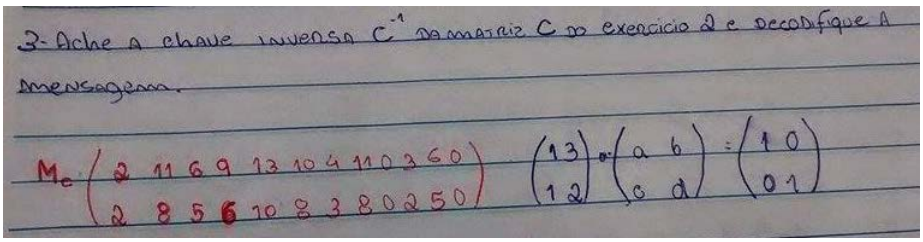


Figura 6.2: Avaliação: multiplicação de matrizes

Na atividade de decodificação a maior dificuldade por parte dos alunos é determinar a matriz decodificadora, ou seja, conseguir obter a matriz inversa. Entre outras coisas, podemos destacar a o fato deles não possuírem a habilidade de resolver sistemas de equações lineares de ordem 2. Eles preferem utilizar a regra de Crammer para solucionar tais sistemas, porém quando surgem números racionais fica mais difícil continuar ou finalizar o processo da inversão de matrizes.

Para avaliar se o aluno adquiriu a habilidade de determinar a matriz inversa, apliquei a atividade a seguir.



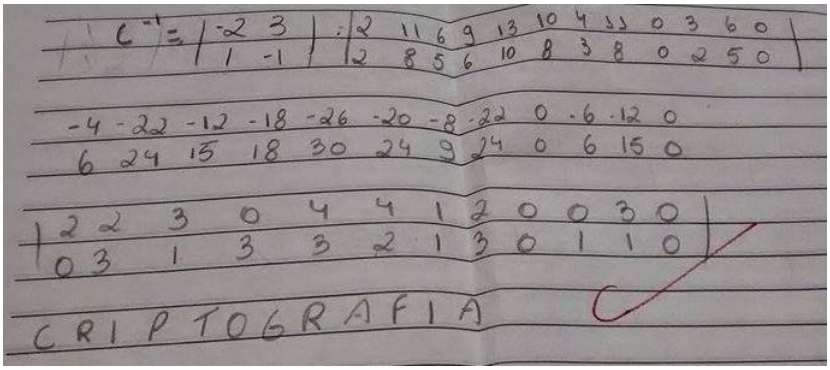


Figura 6.3: Avaliação: matriz inversa

Sequência Didática III

Para dificultar ainda mais a decodificação de mensagens secretas, podemos utilizar matrizes codificadoras quadradas de ordem 3.

Esta atividade segue os mesmos procedimentos descritos nas sequências didática I e II. Ela não foi aplicada em sala de aula, visto a dificuldade dos alunos em determinr a matriz inversa de uma matriz de ordem 2. Contudo, fica registrada aqui como uma sugestão para aprofundar os conhecimentos dos alunos quanto à resolução de sistemas de equações lineares de ordem 3 e consequentemente determinar a matriz inversa de uma matriz de ordem 3.

Vamos exemplificar o processo de decodificação de mensagens secretas usando a seguinte matriz codificadora de ordem 3.

$$C = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 0 & 1 & 2 \end{pmatrix}.$$

Suponhamos que as senhas numéricas de seis dígitos dos clientes de um determinado banco são representadas como uma matriz S de ordem 2×3, em que os três primeiros dígitos são a primeira linha da matriz e os três últimos dígitos são a segunda linha. O banco usa uma matriz invertível C, para manter o sigilo das senhas de seus clientes. Por questões de segurança, o banco gera uma nova matriz $S_C = S \cdot C$, com a senha codificada. Para recuperar a senha de um cliente, o banco utiliza a matriz inversa C^{-1}, que neste caso, é:

$$C^{-1} = \begin{pmatrix} 0 & 0,5 & -0,5 \\ 2 & -1 & -2 \\ -1 & 0,5 & 1,5 \end{pmatrix}.$$

Multiplicando-se a matriz S_C pela matriz C^{-1} obtém-se a matriz S, ou seja, $S = S_C \cdot C^{-1}$.

Sabendo que a senha codificada de um determinado cliente é $7-13-23-8-20-34$, temos:

$$S_C = \begin{pmatrix} 7 & 13 & 23 \\ 8 & 20 & 34 \end{pmatrix}.$$

Então,

$$S = \begin{pmatrix} 7 & 13 & 23 \\ 8 & 20 & 34 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0,5 & -0,5 \\ 2 & -1 & -2 \\ -1 & 0,5 & 1,5 \end{pmatrix}.$$

Logo,

$$S = \begin{pmatrix} 3 & 2 & 5 \\ 6 & 1 & 7 \end{pmatrix}.$$

Portanto, a senha desse cliente é 325617.

Após a realização destas sequências didáticas, a expectativa é que os problemas propostos tenham permitido um bom nível de discussão, em que os argumentos, as análises de situações, os levantamentos de hipóteses e as comparações das soluções tenham fortificado o grupo de alunos como um coletivo gerador de conhecimento.

7 Considerações Finais

O presente trabalho constitui aspectos da teoria dos códigos corretores de erros. A teoria dos códigos corretores de erros surgiu no início da década de 50. O trabalho inicial para a obtenção dos primeiros tipos eficientes de códigos corretores de erros foi árduo, pois exigia, entre outros aspectos, um profundo conhecimento de Álgebra Abstrata. Décadas se passaram e hoje se tem conhecimento de várias classes de bons códigos corretores de erros que podem ser perfeitamente entendidas por engenheiros e cientistas de computação, graças ao esforço de alguns pesquisadores que souberam apresentar esse material com um mínimo rigor matemático.

Primeiramente, enfocamos conceitos básicos de Álgebra Linear, os quais foram os principais pré-requisitos para o desenvolvimento dos demais capítulos. Abordamos os códigos corretores de erros, que é um tema de grande relevância devido sua aplicação em diversas situações da vida prática. No entanto, nosso objetivo maior, neste trabalho, foi relacionar a teoria dos códigos corretores de erros com conteúdos matemáticos, como os de Álgebra Linear. Neste sentido, a articulação dos conhecimentos matemáticos com os avanços tecnológicos, principalmente na área computacional, permite o aprimoramento e maior eficiência destes códigos.

A Matemática está presente na vida humana desde os primórdios das civilizações. Situações que utilizam uma linguagem de códigos e problemas que envolvem mensagens ocultas existem há milhares de anos. Com a evolução dessas civilizações, a forma de registrar estes códigos e também os processos de transmissão de informações que os envolvem vão sendo aperfeiçoados.

Diante disso, destacamos a importância de abordar este estudo no Ensino Médio, proporcionando atividades didáticas sobre codificação, aplicando e ampliando os estudos com Matrizes. Esta importância é evidente no Currículo do Estado de São Paulo, que enfatiza o estudo através da resolução de problemas no contexto da área da tecnologia e que sejam significativos aos alunos. Assim, estes vivenciam situações problema que demonstram a articulação do currículo com o seu cotidiano, aspecto relevante no desenvolvimento do aprendizado pautado em habilidades e competências. Assim sendo, através das sequências didáticas propostas neste trabalho e aplicadas em sala de aula, podemos destacar a participação efetiva dos educandos de modo que se tornem os protagonistas de seu aprendizado.

Por fim, destacamos que a realização de atividades didáticas com estes aspectos norteados pelo currículo, possibilitou ao estudo de multiplicação de matrizes e matriz inversa, uma significativa compreensão dos educandos, de modo que percebessem que os conteúdos matemáticos podem estar relacionados às situações que estão presentes em suas vidas, mesmo que de alguma forma, não sejam tão evidentes. Logo, a sala de aula é um descortinador de possibilidades para esses educandos ampliarem sua visão de mundo e principalmente seus pré-conceitos sobre a matemática, que normalmente é vista como uma grande vilã nas escolas de educação básica.

Referências

[1] BOLDRINI, José Luiz et al. *Álgebra Linear*. 3. ed. São Paulo: Editora HARBRA ltda, 1986. p. 411.

[2] CALLIOLI, C. A.; DOMINGUES, H. H.; COSTA, R. C. F. *Álgebra Linear e Aplicações*. 6. ed. São Paulo: Editora Atual, 2003. p. 352.

[3] COELHO, F. U.; LOURENÇO, M. L. *Um Curso de Álgebra Linear*. 2. ed. São Paulo: Editora EDUSP, 2005. p. 272.

[4] LIPSCHUTZ, S.; LIPSON, M. *Álgebra Linear*. Coleção Schaum. 4. ed. Editora Bookman, 2011. p. 432.

[5] HEFEZ, A.; VILLELA, M.L.T. *Códigos Corretores de Erros*. 1. ed. Rio de Janeiro: IMPA, 2008. p. 206.

[6] ADÁMEK, J. *Foundations of Coding: Theory and Applications of Error-Correcting Codes with an Introduction to Cryptography and Information Theory*. John Wiley & Sons, Inc, 1991.

[7] MILIES, C. P. *Introdução à Teoria dos Códigos Corretores de Erros*. Colóquio de Matemática da Região Centro-Oeste: SBM, 2009.

[8] HAMMING, R. W. *Error-Detecting and Error-Correcting Codes*, Bell Systems Technical Journal, No. 2, Vol. XXIX, 1950. pp. 147-160. Disponível em: <http://www.lee.eng.uerj.br/ gil/redesII/hamming.pdf>.

[9] SHANON, C. E. *A Mathematical Theory of Communication*, Bell Systems Technical Journal, Vol. XXVII, 1948. pp. 379-423. Disponível em: <http://cm.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf>.

[10] MACHADO, N. J. *Currículo do Estado de São Paulo-Matemática e suas Tecnologias*. 1. ed. São Paulo, 2011. p. 21-22.

[11] TAMAROZZI, A. C. *Codificando e decifrando mensagens*. Revista do Professor de Matemática 45, São Paulo: Sociedade Brasileira de Matemática.

[12] http://www.cesarkallas.net/arquivos/faculdade-pos/TP301-codificacao-fonte/ 2_ Bloco_V2011_Rev4.pdf. Último acesso em 18/fev/2015.

[13] http://www.m3.ime.unicamp.br/recursos/1020. Último acesso em 15/jan/2015.

[14] http://www.mat.uc.pt/∼caldeira/CodigosCriptografia_07_08/PDFs_ indisponiveis/Codigos_07_08/PDFs_handout.pdf. Último acesso em 10/dez/2014.

[15] http://www.ufsj.edu.br/portal2-repositorio/File/i-ermac/anais/minicursos/mc8.pdf. Último acesso em 10/dez/2014.

[16] http://www.sbm.org.br/docs/coloquios/CO-1-09.pdf. Último acesso em 10/dez/2014.

[17] http://www.ime.unicamp.br/∼mfirer/3NotasFoz2006.pdf. Último acesso em 16/mar/2015.

[18] http://www.gente.eti.br/lematec/CDS/XIIICIAEM/artigos/691.pdf. Último acesso em 18/mar/2015.

[19] http://www.lematec.no-ip.org/CDS/ENEM10/artigos/CC/T17_CC555.pdf. Último acesso em 16/mar/2015.