Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Departamento de Matemática

Raphael Cascelli dos Santos Souza

Códigos Perfeitos e Raio de Empacotamento

Juiz de Fora
2019

Raphael Cascelli dos Santos Souza

Códigos Perfeitos e Raio de Empacotamento

Monografia apresentada ao Curso de Matemática da Universidade Federal de Juiz de Fora como requisito parcial para obtenção do título de Bacharel em Matemática.

Orientadora: Beatriz Casulari da Motta Ribeiro

Juiz de Fora

2019

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

Cascelli dos Santos Souza, Raphael.
Códigos Perfeitos e Raio de Empacotamento / Raphael Cascelli dos Santos Souza. – 2019.
56 f. : il.

Orientadora: Beatriz Casulari da Motta Ribeiro
Trabalho de Conclusão de Curso – Universidade Federal de Juiz de Fora, Instituto de Ciências Exatas. Departamento de Matemática, 2019.

1. Códigos Corretores de Erros. 2. Métricas em Códigos. 3. Corpos Finitos. I. Sobrenome, Nome do orientador, orient. II. Título.

Raphael Cascelli dos Santos Souza

Códigos Perfeitos e Raio de Empacotamento

Monografia apresentada ao Curso de Matemática da Universidade Federal de Juiz de Fora como requisito parcial para obtenção do título de Bacharel em Matemática.

Aprovada em: **15 de julho de 2019**

BANCA EXAMINADORA

Profa. Dra. Beatriz Casulari da Motta Ribeiro
Orientadora
Universidade Federal de Juiz de Fora

Profa. Dra. Tatiana Aparecida Gouveia
Universidade Federal de Juiz de Fora

Prof. Dr. Frederico Sercio Feitosa
Universidade Federal de Juiz de Fora

AGRADECIMENTOS

A Deus em primeiro lugar.

Aos meus pais, Luciano Lopes e Geovana Cascelli, e irmãos, Vinícius e Thiago Cascelli, pelo incentivo constante, confiança e amor. Hoje tenho a certeza de que sem seu apoio o caminho para alcançar aquilo que sonhei seria um pouco mais difícil. Aos meus avós, tios e primos, por todo o apoio.

À professora e orientadora Beatriz Casulari da Motta Ribeiro, por ter me proporcionado as primeiras oportunidades de estudar e conhecer melhor as áreas da Matemática que mais me despertam interesse, desde a Matemática Discreta e Álgebra até os Corpos Finitos e Códigos Corretores de Erros; e principalmente, pela amizade, apoio, conselhos e incentivo durante todo esse período.

À professora Tatiana Aparecida Gouveia e ao professor Frederico Sercio Feitosa, por terem aceitado o convite para fazer parte da banca que avaliou este trabalho, colaborando para o resultado final aqui apresentado.

A cada um dos meus professores da graduação, por tudo que ensinaram e contribuíram para o meu crescimento acadêmico e pessoal.

A todos os meus amigos, os quais os nomes podem ser omitidos sem que haja desapreço por cada um, por todo suporte, companhia, incentivo e amor. Sou muito grato a cada um deles.

À Universidade Federal de Juiz de Fora e ao Departamento de Matemática.

À Propesq - UFJF, pelas bolsas de Iniciação Científica.

"*Em dois finais de semanas consecutivos eu fui e descobri que todas minhas coisas tinham sido descarregadas e nada tinha sido feito. Eu estava realmente aborrecido e irritado porque queria estas respostas e tinha perdido dois finais de semana. E então eu me disse: Maldição, se as máquinas podem detectar um erro, por que não podemos localizar a posição do erro e corrigi-lo?*"

(R.W. Hamming em entrevista em fevereiro de 1977)

RESUMO

O objetivo principal desse trabalho é estudar certos exemplos de códigos perfeitos, isto é, que fornecem empacotamentos de esferas recobrindo todo o espaço. Os exemplos estudados são os códigos de Hamming e códigos de Hamming estendidos, além da classe mais geral de códigos sobre ordens totais. A fim de atingir esse objetivo, começamos apresentando brevemente uma introdução aos corpos finitos e espaços vetoriais sobre tais corpos. Em seguida, estudamos os fundamentos da teoria dos códigos corretores de erros, incluindo matrizes geradora e verificação de paridade, equivalência de códigos e códigos duais. Além disso, apresentamos e exploramos diferentes tipos de métricas em espaços de códigos, a saber: de Hamming, de Lee e ponderadas.

Palavras-chave: Códigos corretores de erros. Empacotamento de esferas. Métricas ponderadas.

ABSTRACT

The main purpose of this work is to study certain exemples of perfect codes, that is, which give sphere packings that cover the whole space. In particular, we study Hamming and extended Hamming codes and the more general class of the codes over total orders. To achive this, we begin by presenting briefly an introduction to finite fields and vector spaces over these fields. Then, we study the base of error correcting codes, which include generator and parity check matrices, code equivalence and dual codes. Besides that, we present and explore diferent types of metrics in code spaces, such as Hamming, Lee and poset metrics.

Key-words: Error-correcting codes. Sphere packing. Poset metrics.

SUMÁRIO

1 INTRODUÇÃO

Na transmissão de dados, podem ocorrer problemas, como interferências eletromagnéticas ou erros humanos, como erros de digitação, que fazem com que a mensagem recebida seja diferente da enviada. Na teoria da informação, chamamos tais erros de ruídos e gostaríamos de estudar métodos que permitam não só detectá-los, como também corrigi-los. Essa teoria é relativamente nova, tendo tido início na metade do século XX.

Em 1947, Richard W. Hamming trabalhava no Laboratório Bell com computadores, na época máquinas caras e raras, os quais acessava apenas nos fins de semana. Os programas eram escritos em cartões perfurados e, caso houvesse algum erro, o computador parava o processamento daquele programa e ia para o de outro usuário. Hamming notou que, diversas vezes, seus programas eram interrompidos e conjecturou que deveria haver uma forma de que a máquina corrigisse pelo menos alguns dos erros. Conseguiu escrever um programa que detectava até dois erros e corrigia um, caso fosse único, e publicou no jornal do laboratório. A criação de códigos mais eficientes foi feita, então, por Claude Elwood Shannon, que também trabalhava no Laboratório Bell, em 1948. Shannon ficou então conhecido como o criador da Teoria dos Códigos Corretores de Erros, mas Hamming continuou dando contribuições inestimáveis, como veremos nesse texto.

A partir da década de 1970, com as pesquisas espaciais e a grande popularização dos computadores, essa teoria começou a interessar também aos engenheiros. Hoje em dia, os códigos corretores de erros são utilizados sempre que se deseja transmitir ou armazenar dados, garantindo a sua confiabilidade. São exemplos disso todas as comunicações via satélite, as comunicações internas de um computador, o armazenamento de dados em fitas ou disquetes magnéticos, ou o armazenamento ótico de dados.

Nesse trabalho, os códigos corretores de erros são subespaços vetoriais de dimensão k de $\mathbb{F}_q^n$. As principais características, como matrizes geradoras e verificações de paridades, além dos conceitos de distância mínima e de raio de empacotamento de um código corretor de erros serão apresentadas no capítulo 2. Essas propriedades serão mais exploradas nos capítulos seguintes, conforme descreveremos agora.

Existem vários problemas a serem considerados na teoria. Um deles é: fixados um corpo finito $\mathbb{F}_q$, um comprimento n e uma dimensão k, quantos erros esse código pode detectar? Mais ainda, quantos desses ele pode corrigir? Essa pergunta tem relação com a métrica utilizada no código, que determina qual é a distância mínima entre os elementos no espaço. Códigos com mesmos parâmetros, mas métricas distintas podem corrigir mais ou menos erros. Nessa direção, faz sentido o estudo de diferentes métricas em espaços de códigos, assunto do nosso capítulo 3.

Outro problema clássico é o de empacotamento de esferas, que consiste em distribuir esferas de mesmo raio no espaço de forma que duas esferas se toquem no máximo em um

ponto do bordo. Busca-se ainda um arranjo de esferas que ocupe o maior espaço possível do plano. No caso da relação entre esse problema e os códigos corretores de erros, estamos interessados em empacotar o plano com esferas centradas em pontos do código. Para isso, o estudo do raio de empacotamento, apresentado no capítulo 2, é retomado no capítulo 4 no caso de códigos perfeitos, isto é, códigos em que não acontece uma das situações mais problemáticas para a correção de erros: quando a mensagem recebida não pertence a nenhuma das esferas do empacotamento.

2 CÓDIGOS CORRETORES DE ERROS

O principal objetivo deste capítulo é realizar uma introdução aos corpos finitos, apresentando algumas das suas propriedades mais importantes. Estudaremos também as definições e propriedades principais dos chamados códigos corretores de erros. Nossas principais referências aqui são [2] e [5].

2.1 O QUE É UM CÓDIGO?

O exemplo mais familiar de um código corretor de erros é um idioma. Por exemplo, a língua portuguesa tem 23 letras, mas vamos considerar ainda o espaço em branco, o ç e as vogais acentuadas, fazendo com que o alfabeto A tenha mais letras e a língua portuguesa possa ser considerada como um elemento de A^{46}. Esse número 46 é o tamanho da mais longa palavra dessa língua, o termo médico *pneumoultramicroscopicossilicovulcanoconiótico*. Como conhecemos a língua, se recebemos uma mensagem "televipão" (no caso, é televipão com 37 espaços em branco em seguida), sabemos corrigir, pois a palavra que mais se assemelha é "televisão".

No entanto, esse código não é eficiente. De fato, a língua portuguesa tem palavras muito "próximas" das outras. Por exemplo, "gato", "rato", "pato", "mato", "fato", "tato" e "jato" têm apenas uma letra de diferença. Assim, ao receber "hato", sabemos que há erro, mas não sabemos corrigir; já ao recebermos "gato" não temos nem como saber se a mensagem tem erro.

Para corrigir esse tipo de problema, os códigos corretores de erros tem como função acrescentar novas informações que serão transmitidas fazendo com que estas possam ser corrigidas quando ocorrem ruídos. Vamos ver dois exemplos:

Exemplo 2.1.1. *Digamos que queremos responder a pergunta "haverá aula amanhã?" com SIM ou NÃO. Uma possibilidade seria codificar o SIM com a palavra-código* 1 *e o NÃO com a palavra-código* 0*. No entanto, havendo algum erro na transmissão, o* 1 *viraria* 0 *ou o* 0 *viraria* 1*, isto é, poderia ser recebido NÃO no lugar de SIM ou SIM no lugar de NÃO e o destinatário seria incapaz de saber se houve erro.*

Para tentar solucionar esse erro, poderíamos introduzir uma redundância e codificar o SIM com 11 *e o NÃO com* 00*. Agora, imagine que foi enviado SIM, isto é,* 11*, mas foi recebido* 10*. O receptor consegue saber que houve erro, mas não consegue corrigi-lo, por não saber se foi o primeiro ou o segundo bit que foi errado.*

Vamos tentar novamente incluir uma redundância: codificar o SIM com 111 *e o NÃO com* 000*. Digamos então que ocorreu um erro na transmissão do SIM, e o receptor recebeu o vetor* 110*. Nesse caso, ele não só saberia que houve erro, como poderia corrigi-lo. Para isso, seria necessário supor que a probabilidade de erro é pequena, assim, seria mais*

provável haver um erro do que dois, ou seja, o vetor 110 *seria associado a palavra-código* 111 *e não à* 000.

Exemplo 2.1.2. *Um robô se move sobre um tabuleiro quadriculado segundo quatro comandos de forma que o robô se desloca do centro de uma casa para o centro da casa vizinha indicada pelo comando. Os quatro comandos são codificados como elementos de* $\{0,1\} \times \{0,1\}$ *através do seguinte código da fonte:*

$$Leste \longmapsto 00$$
$$Oeste \longmapsto 01$$
$$Norte \longmapsto 10$$
$$Sul \longmapsto 11$$

Se os comandos, quando enviados, sofrerem interferências, precisamos ser capazes de corrigir o erro. Por exemplo, a mensagem 00 *possa, na chegada, ser recebida como* 01*, o que faria com que o robô, em vez de ir para Leste, fosse para Oeste. Como não é nem possível saber se* 01 *era de fato a mensagem (ou seja, identificar o erro), não temos um bom código. Assim, vamos adicionar redundâncias, de forma que o código torne-se um corretor de erros. O novo código introduzido na recodificação é chamado código de canal:*

$$Leste \longmapsto 00000$$
$$Oeste \longmapsto 01011$$
$$Norte \longmapsto 10110$$
$$Sul \longmapsto 11101$$

Agora, se a palavra 10110 *for recebida como* 11110*, podemos detectar o erro. A palavra do código mais próxima da referida mensagem (a que tem menor número de componentes diferentes) é* 10110*, que é precisamente a palavra transmitida.*

O procedimento descrito no exemplo 2.1.2 pode ser esquematizado como mostra a Figura 1. O canal pode ser, por exemplo, canal de radiofrequência, canal de micro-ondas, cabo, circuito integrado digital, fita magnética, disco de armazenamento, etc.

Nosso estudo consiste em transformar o código da fonte em código de canal, em detectar e corrigir erros na recepção e em decodificar o código de canal em código da fonte. Inspirados no exemplo 2.1.1, iremos apenas considerar neste texto canais simétricos, isto é, canais que possuem as seguintes propriedades:

- Todos os símbolos transmitidos têm a mesma probabilidade (pequena) de serem recebidos errados.
- Se um símbolo é recebido errado, a probabilidade de ser qualquer um dos outros é a mesma.

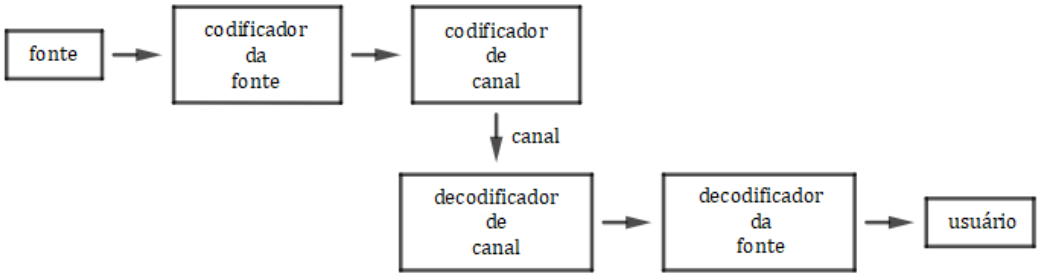

Figura 1 – Esquema de Codificação e Decodificação de uma mensagem.

2.2 CORPOS FINITOS E ESPAÇOS VETORIAIS

O ambiente que iremos trabalhar é o de espaços vetoriais sobre corpos finitos. Dessa forma, nessa seção, apresentaremos uma breve construção dessa classe de corpos. Como base para essa seção, usamos as referências [3], [4] e [5].

Definição 2.2.1. *Seja K um corpo. A característica de K, denotada por $char(K)$, é o menor inteiro positivo n tal que $n \cdot k = 0$ para todo $k \in K$. Se não existir tal inteiro, dizemos que a característica de K é zero.*

Proposição 2.2.2. *Seja K um corpo com elemento unidade 1. Considere o conjunto*

$$\Lambda_K = \{n \in \mathbb{N} \ : \ n \cdot 1 = 0\} \subset \mathbb{N}.$$

Então

1. *Se $n \cdot 1 \neq 0$ para todo n natural, então $char(K) = 0$.*
2. *Se existir n natural tal que $n \cdot 1 = 0$, então $char(K) = \min \Lambda_K$.*

Demonstração. 1) Como não existe $n \in \mathbb{N}$ tal que $n \cdot 1 = 0$, isto implica que não existe $n \in \mathbb{N}$ tal que $n \cdot a = 0$, $\forall a \in K$. Logo, $char(K) = 0$.
2) Seja $n = \min \Lambda_K$. Para todo $a \in K$ temos que como $a = 1 \times a$ então $n \cdot a = n \cdot (1 \times a) = \underbrace{1 \times a + \ldots + 1 \times a}_{n \ vezes} = \underbrace{(1 + \ldots + 1)}_{n \ vezes} \times a = (n \cdot 1) \times a = 0 \times a = 0$. Como n é mínimo em Λ_K, temos $n = char(K)$. □

Proposição 2.2.3. *Seja K um corpo. Então $char(K) = 0$ ou $char(K) = p$, onde p é um número primo.*

Demonstração. Se $\text{char}(K) = 0$, então não temos o que mostrar. Caso contrário, como $K \neq \{0\}$, existe um elemento não nulo no corpo, de modo que $\text{char}(K) \geq 2$. Suponhamos

por absurdo que $\text{char}(K)$ não é um número primo. Então, é possível escrever $\text{char}(K) = m \cdot n$, com $m, n \in \mathbb{Z}$ e $1 < m < n < \text{char}(K)$. Assim,

$$0 = \text{char}(K) \cdot 1 = (m \cdot n) \cdot 1 = (m \cdot 1)(n \cdot 1)\,.$$

Logo, temos que $m \cdot 1 = 0$ ou $n \cdot 1 = 0$, contrariando a minimalidade de $char(K)$. □

Corolário 2.2.4. *Todo corpo finito possui característica prima.*

Demonstração. Pelo fato de K ser finito, temos que existem dois inteiros $m < n$ em K tais que $m \cdot 1 = n \cdot 1$. Logo, $(n - m)1 = 0$ com $n - m > 0$ e, portanto $char(K) > 0$. Logo, a característica de K é um número primo. □

Como exemplo de corpo finito com p elementos, onde p é um número primo, temos o corpo $\mathbb{Z}_p := \mathbb{Z}/\langle p \rangle$. Ainda, é possível provar que $\mathbb{Z}_p$ é corpo se, e somente se, p é primo.

Definição 2.2.5. *Fixado um número primo p, sejam $\mathbb{F}_p$ o subconjunto $\{0, 1, \ldots, p-1\}$ dos inteiros e φ a seguinte função*

$$\begin{aligned} \varphi : \mathbb{Z}_p &\longrightarrow \mathbb{F}_p \\ \bar{a} &\longmapsto a \end{aligned}$$

Então, $\mathbb{F}_p$, com a estrutura de corpo induzida por φ, é um corpo finito, chamado corpo de Galois de ordem p.

Observação 2.2.6. *Se F é um subcorpo de um corpo finito $\mathbb{F}_p$, sendo p um número primo, então F deve conter os elementos 0 e 1 e consequentemente deve conter também todos os demais elementos de $\mathbb{F}_p$, visto que a adição é uma operação fechada em F.*

Proposição 2.2.7. *Seja F um corpo finito contendo um subcorpo K com p elementos. Então, F possui p^n elementos, onde $n = [F : K]$.*

Demonstração. Como F pode ser visto como um espaço vetorial sobre K, a dimensão do espaço vetorial F sobre o corpo K é finita F é um corpo finito.

Seja $n = [F : K]$, temos que F possui uma base sobre K com n elementos, digamos $x_1, x_2, \ldots, x_n$. Assim, todo elemento de F pode ser escrito de forma única como $a_1x_1 + a_2x_2 + \ldots + a_nx_n$, onde $a_1, a_2, \ldots, a_n \in K$.

Desta maneira, note que, como K possui p elementos, F possui exatamente p^n elementos. □

Lema 2.2.8. *Seja K um corpo finito com q elementos. Para todo $a \in K\backslash\{0\}$, temos que*

$$a^{q-1} = 1.$$

Demonstração. Seja $a \in K\backslash\{0\}$. Consideremos a aplicação

$$\begin{aligned} \varphi_a : K\backslash\{0\} &\longrightarrow K\backslash\{0\} \\ x &\longmapsto ax \end{aligned} \tag{2.1}$$

φ_a é injetora pois se $\varphi_a(x) = \varphi_a(y)$, então $ax = ay$ e, como K é corpo, podemos multiplicar à esquerda pelo elemento inverso de a, a^{-1}, de modo que

$$x = a^{-1}ax = a^{-1}ay = y\,.$$

Desta maneira, como K é finito segue que φ_a é sobrejetora e portanto, bijetora.

Se $K\backslash\{0\} = \{x_1, x_2, \ldots, x_{q-1}\}$, temos então

$$\{ax_1, ax_2, \ldots, ax_{q-1}\} = \{x_1, x_2, \ldots, x_{q-1}\},$$

e portanto,

$$ax_1ax_2 \ldots ax_{q-1} = x_1x_2 \ldots x_{q-1},$$

e, consequentemente,

$$a^{q-1} = 1.$$

□

Segue imediatamente do lema anterior o seguinte resultado:

Teorema 2.2.9. *Seja K um corpo finito com q elementos. Para todo $a \in K$ e para todo $i \in \mathbb{N}$, temos que $a^{q^i} = a$.*

Proposição 2.2.10. *Sejam K um corpo finito tal que $char(K) = p$ e $q = p^r$, para algum inteiro positivo r. O polinômio*

$$f(x) = x^q - x$$

não possui fatores irredutíveis múltiplos em $K[x]$.

Demonstração. De fato, note que o polinômio f e sua derivada, f', são primos entre si pois

$$f'(x) = qx^{q-1} - 1 = -1.$$

Portanto, concluímos que $f(x)$ não possui fatores irredutíveis múltiplos em $K[x]$. □

Proposição 2.2.11. *Se F é um corpo finito com q elementos e K é um subcorpo de F, então o polinômio $x^q - x \in K[x]$ é fatorado em $F[x]$ na forma*

$$x^q - x = \prod_{a \in F} (x - a)$$

e F é um corpo de decomposição de $x^q - x$ sobre K.

Demonstração. O polinômio $x^q - x$ de grau q possui no máximo q raízes em F. Pelo Teorema 2.2.9, para $i = 1$, temos que todos os elementos de F são raízes de $x^q - x$. Logo, o polinômio $x^q - x$ se fatora em F e não pode se fatorar em nenhum corpo menor, o que significa que F é um corpo de decomposição do polinômio $x^q - x$. □

Tomemos agora um polinômio de grau n irredutível, $f(x)$, em $\mathbb{Z}_p[x]$. Sabemos que o quociente $\mathbb{Z}_p[x]/\langle f(x)\rangle$ é um corpo com p^n elementos. O Teorema a seguir é uma das formas do teorema de existência e unicidade dos corpos finitos, cuja demonstração pode ser encontrada em [5].

Teorema 2.2.12. *Para todo primo p e para todo inteiro positivo n existe um corpo finito com p^n elementos. Além disso, todo corpo finito com $q = p^n$ elementos é isomorfo ao corpo de decomposição do polinômio $x^q - x$ sobre $\mathbb{F}_p$.*

Dessa forma, consideramos $\mathbb{F}_q$, onde $q = p^n$, o corpo de decomposição de $x^q - x$. O teorema 2.2.12 nos diz que $\mathbb{F}_q$ é único, ou seja, dado outro corpo finito K com q elementos

$$K \simeq \mathbb{F}_q \simeq \mathbb{Z}_p[x]/\langle f(x)\rangle\,,$$

para algum $f(x) \in \mathbb{Z}_p[x]$ irredutível.

Por fim, seja $\mathbb{F}_q^n$ o conjunto de todas as n-uplas de elementos de $\mathbb{F}_q$. O conjunto $\mathbb{F}_q^n$ possui uma estrutura de espaço vetorial sobre $\mathbb{F}_q$, onde a soma e o produto por escalar são definidas coordenada a coordenada, isto é, se $x, y \in \mathbb{F}_q^n$, $\lambda \in \mathbb{F}_q$ com $x = (x_1, x_2, ..., x_n)$ e $y = (y_1, y_2, ..., y_n)$, então

$$x + y = (x_1 + y_1, x_2 + y_2, ..., x_n + y_n)$$

e

$$\lambda x = (\lambda x_1, \lambda x_2, ..., \lambda x_n).$$

Nesse caso, dizer que V é subespaço vetorial de $\mathbb{F}_q^n$ significa que dados $u, v \in V$ e $\lambda \in \mathbb{F}_q$ quaisquer, então $u + v$ e λu pertencem a V. Essa é a noção que nos permitirá definir um código linear.

Nas discussões que se seguem, vamos por vezes considerar matrizes que auxiliam na detecção de erros e na geração de códigos. O conjunto das matrizes de m linhas, n colunas e coeficientes em $\mathbb{F}_q$ será denotado por $M(m, n, \mathbb{F}_q)$.

2.3 CÓDIGOS LINEARES E DETECÇÃO DE ERROS

Quando vimos a língua portuguesa como um código corretor de erros, estávamos considerando $\mathcal{C}$ o conjunto das possíveis palavras que podem ser escritas utilizando-se das letras, acentos e caracteres especiais, bem como o espaço em branco que utilizamos para que todas as palavras possuam o mesmo comprimento. Nesse sentido, definimos em geral:

Definição 2.3.1. *Um $(n; M)$ código $\mathcal{C}$ sobre $\mathbb{F}_q$ é um subconjunto de $\mathbb{F}_q^n$ com M elementos. Chamamos $\mathbb{F}_q$ de alfabeto (e seus elementos de letras) e os elementos de $\mathcal{C}$ são chamadas de palavras.*

Definição 2.3.2. *Dizemos que $\mathcal{C} \subset \mathbb{F}_q^n$ é um $[n; k]$ código linear sobre $\mathbb{F}_q$ se $\mathcal{C}$ for um subespaço vetorial de dimensão k de $\mathbb{F}_q^n$.*

A dimensão de $\mathcal{C}$ é definida como o número de elementos de uma base, ou seja, a quantidade mínima de elementos $v_1, v_2, ..., v_k \in \mathcal{C}$ tal que todo elemento $v \in \mathcal{C}$ pode ser descrito como combinação linear $v = \lambda_1 v_1 + \lambda_2 v_2 + ... + \lambda_k v_k$, com $\lambda_1, \lambda_2, ..., \lambda_k$ escalares em $\mathbb{F}_q$. Essa noção coincide, claramente, com a de dimensão de um espaço vetorial sobre $\mathbb{F}_q$.

Observe que cada um dos k escalares pode assumir q valores distintos e como estamos considerando uma base de $\mathcal{C}$, obtemos q^k combinações lineares distintas, ou seja, $\mathcal{C}$ tem q^k elementos e um $[n; k]$ código linear é um $(n; q^k)$ código sobre $\mathbb{F}_q$.

Apenas a estrutura de espaço vetorial já permite, sob certas circunstâncias, detectarmos erros. Seja $\mathcal{C}$ um $[n; k]$ código linear, com $k < n$ e seja $v \in \mathcal{C}$ uma palavra. Suponha que ao transmitirmos a palavra v a palavra recebida seja w (podendo ser $v = w$). Ao receber a mensagem w, procedemos antes de tudo com a verificação de que esta mensagem é uma palavra do nosso vocabulário, ou seja, verificamos se $w \in \mathcal{C}$. Esta verificação é simples, se lembrarmos que um subespaço vetorial é definido por um sistema de equações lineares homogêneas. Assim, podemos definir:

Definição 2.3.3. *Se considerarmos a matriz $H \in M(n-k, n, \mathbb{F}_q)$ definida pelos coeficientes do sistema linear, podemos representar este sistema matricialmente pela equação*

$$H \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

e temos que $\mathcal{C}$ é o conjunto solução deste sistema. Uma matriz H satisfazendo esta propriedade é chamada de matriz de verificação de paridade.

Pela definição 2.3.3, se $w = (w_1, w_2, \ldots, w_n)$ for uma palavra recebida, e denotarmos por w^t a matriz transposta (ou vetor coluna), basta efetuar o produto Hw^t para sabermos se w pertence a $\mathcal{C}$. Se $w \notin \mathcal{C}$, sabemos que a mensagem recebida é equivocada, ou seja, conseguimos detectar a ocorrência de um erro.

Observemos ainda que sendo $\mathcal{C}$ um código de dimensão k, o posto de H é $n - k$. Assim, a matriz H deve ter $n - k$ linhas linearmente independentes.

Exemplo 2.3.4. *Considere o código linear* $\mathcal{C} = \{0000, 1010, 0111, 1101\} \subset \mathbb{F}_2^4$. *Vamos calcular uma matriz de verificação de paridade para* $\mathcal{C}$.

Pela definição, a matriz de verificação de paridade para esse código é uma matriz $H \in M(2, 4, \mathbb{F}_2)$ *tal que*

$$H = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \end{bmatrix}$$

de forma que $Hc^t = 0$ *para todo elemento* c *do código. Assim temos o sistema linear homogêneo*

$$\begin{cases} x_1 + x_3 &= 0 \\ x_2 + x_3 + x_4 &= 0 \\ x_1 + x_2 + x_4 &= 0 \\ y_1 + y_3 &= 0 \\ y_2 + y_3 + y_4 &= 0 \\ y_1 + y_2 + y_4 &= 0, \end{cases}$$

e assim como cada x_i *e* y_j *são elementos de* $\mathbb{F}_2$, *pela primeira equação temos que*

$$x_1 = x_3 = 0 \text{ ou } x_1 = x_3 = 1.$$

Suponhamos que $x_1 = x_3 = 1$. *Daí, pela segunda equação, temos*

$$x_2 + x_4 + 1 = 0$$

então, $x_2 + x_4 = 1$, *isto é*

$$x_2 = 0 \text{ e } x_4 = 1 \text{ ou } x_2 = 1 \text{ e } x_4 = 0.$$

Façamos $x_2 = 0$ *e* $x_4 = 1$.

Assim, temos $x_1 = 1$, $x_2 = 0$, $x_3 = 1$ *e* $x_4 = 1$.

Agora, para os outros coeficientes, pela quarta equação temos

$$y_1 = y_3 = 0 \text{ ou } y_1 = y_3 = 1.$$

Suponhamos que $y_1 = y_3 = 0$. *Desta maneira, pela quinta equação, temos*

$$y_2 + y_4 = 0$$

e segue que

$$y_2 = y_4 = 0 \text{ ou } y_2 = y_4 = 1.$$

Como não queremos todos os y_j*'s iguais a zero, façamos* $y_2 = y_4 = 1$.

Assim, temos $y_1 = 0$, $y_2 = 1$, $y_3 = 0$ *e* $y_4 = 1$.

Portanto, temos

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

Observe que cada escolha dos coeficientes fornece uma matriz de verificação de paridade diferente para o código $\mathcal{C}$.

Agora, notamos que se houve um erro, podemos ter $w \neq v$ mas com $w \in \mathcal{C}$. E, ainda, como $\mathcal{C}$ possui q^k elementos, enquanto $\mathbb{F}_q^n$ possui q^n elementos, dos quais $q^n - q^k = q^k(q^{n-k} - 1)$ elementos não pertencem a $\mathcal{C}$. Se assumirmos a hipótese de que o ruído perturba a palavra transmitida v de modo que possamos receber qualquer elemento de $\mathbb{F}_q^n$, a probabilidade P de detectarmos o erro é dada por:

$$P = \frac{\#\{x \in \mathbb{F}_q^n \mid x \notin \mathcal{C}\}}{\#\mathbb{F}_q^n} = \frac{q^n - q^k}{q^n} = 1 - \frac{1}{q^{n-k}}.$$

Como $q \geq 2$ e $n - k \geq 1$, temos que $P < 1$ e esta cresce conforme q ou $n - k$ crescem. Assim, se quisermos detectar em média 999 erros a cada 1000 ocorrências, se tivermos $q = 2$, basta termos $n - k \geq 10$. Como

$$\lim_{q\to\infty} \frac{1}{q^{n-k}} = \lim_{n-k\to\infty} \frac{1}{q^{n-k}} = 0,$$

podemos detectar erros com a confiança tão grande quanto quisermos.

Encerramos essa seção com uma observação à cerca da eficiência da detecção de erros.

Observação 2.3.5. *Para detectarmos a existência de erros em um $[n; k]$ código linear precisamos efetuar o produto Hw^t. Assim, se $H = (a_{ij})$, e $w = (w_1, w_2, \ldots, w_n)$, sabemos que a j-ésima entrada de Hw^t é $a_{j1}w_1 + a_{j2}w_2 + \ldots + a_{jn}w_n$, isto é, precisamos realizar n produtos e n somas, devendo ainda verificar se $a_{j1}w_1 + a_{j2}w_2 + \ldots + a_{jn}w_n = 0$, sendo portanto necessário realizar $2n + 1$ operações. Daí, como o vetor Hw^t tem $n - k$ entradas, para verificarmos a ocorrência de erros executamos $(n-k)(2n+1)$ operações. Apesar deste número que cresce quadraticamente com n parecer relativamente grande se não tivéssemos a estrutura vetorial, teríamos que comparar w com todos os q^k elementos de $\mathcal{C}$. De forma geral, para $q > 1$, temos*

$$\lim_{n-k\to\infty} \frac{(n-k)(2n+1)}{q^{n-k}} = \lim_{n-k\to\infty} \frac{((n-k)(2n+1))^k}{q^{n-k}} = 0,$$

para todo $d \geq 0$, ou seja, para códigos com codimensão $n - d$ grande, a estrutura linear é bastante econômica para a verificação de erros.

2.3.1 Métrica em um código

No próximo capítulo vamos definir algumas métricas particulares em espaços de códigos mas antes, nesta seção, vamos introduzir a noção de uma métrica em um

conjunto que utilizaremos para calcular um parâmetro especial de um código dado, além de necessitarmos dessas ferramentas para determinarmos um algoritmo útil na geração de um código linear.

Definição 2.3.6. *Uma métrica em um conjunto X é uma função $d : X \times X \to \mathbb{R}$ satisfazendo as seguintes propriedades:*

1. $d(x, y) > 0$ *se* $x \neq y$ *e* $d(x, x) = 0$, *para quaisquer* $x, y \in X$;
2. $d(x, y) = d(y, x)$, *para quaisquer* $x, y \in X$;
3. $d(x, z) \leq d(x, y) + d(y, z)$, *para quaisquer* $x, y, z \in X$.

Definição 2.3.7. *Sejam d uma métrica em um conjunto X, um ponto $x_0 \in X$ e $r > 0$. Definimos a bola $B(x_0; r)$ e a esfera $S(x_0; r)$, ambas de centro em x_0 e raio r dadas por:*

$$B(x_0; r) = \{x \in X : d(x, x_0) \leq r\}$$

e

$$S(x_0; r) = \{x \in X : d(x, x_0) = r\}.$$

Definição 2.3.8. *Seja $x \in \mathbb{F}_q^n$ definimos o peso do vetor $x \in \mathbb{F}_q^n$ como*

$$\omega(x) = d(x, 0).$$

Quando estivermos trabalhando com métricas específicas, usaremos um subíndice $d_\star(\cdot;\cdot)$ para identificar a métrica. Da mesma forma, o mesmo subíndice será adotado para designar todos os conceitos derivados da métrica, tais como $B_\star(\cdot;\cdot)$, $S_\star(\cdot;\cdot)$ e $\omega_\star(\cdot)$.

Definição 2.3.9. *Considerando os pontos de um dado código $\mathcal{C}$ definimos a distância mínima entre dois destes pontos como*

$$d = d(\mathcal{C}) = \min\{d(v, w) \ : \ v, w \in \mathcal{C}, \ v \neq w\}.$$

Se lembrarmos que o código $\mathcal{C}$ é linear e que a distância é compatível com a métrica, temos que

$$d(x, y) = d(x - x, y - x) = d(0, z)$$

para quaisquer $x, y \in \mathbb{F}_q^n$, onde $z = x - y$. Ou ainda, como $v - w \in \mathcal{C}$ sempre que $v, w \in \mathcal{C}$, temos que

$$d = \min\{\omega(v) \ : \ 0 \neq v \in \mathcal{C}\}.$$

Definição 2.3.10. *Sejam o alfabeto $\mathbb{F}_q$ e $n \in \mathbb{N}$. Dizemos que uma função $f : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ é uma isometria de $\mathbb{F}_q^n$ se preserva distâncias. Isto é,*

$$d(f(x), f(y)) = d(x, y) \quad \forall x, y \in \mathbb{F}_q^n.$$

Proposição 2.3.11. *Toda isometria de $\mathbb{F}_q^n$ é uma bijeção de $\mathbb{F}_q^n$.*

Demonstração. Seja $f : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ uma isometria. Suponha que para $x, y \in \mathbb{F}_q^n$ tenhamos $f(x) = f(y)$. Logo, $d(x, y) = d(f(x), f(y)) = 0$, o que implica que $x = y$. Assim provamos que f é injetora, e como toda aplicação injetora de um conjunto finito nele próprio é sobrejetora, temos que f é uma bijeção. □

Proposição 2.3.12. *As seguintes propriedades são de fácil verificação.*

1. *A função identidade de $\mathbb{F}_q^n$ é uma isometria.*
2. *Se f é uma isometria de $\mathbb{F}_q^n$, então f^{-1} é uma isometria de $\mathbb{F}_q^n$.*
3. *Se f e g são isometrias de $\mathbb{F}_q^n$, então $f \circ g$ é uma isometria de $\mathbb{F}_q^n$.*

Vamos usar os exemplos a seguir no próximo capítulo.

Exemplo 2.3.13. *Se $f : \mathbb{F}_q \longrightarrow \mathbb{F}_q$ é uma bijeção, e k é um número inteiro tal que $1 \leq k \leq n$, a aplicação*

$$\begin{aligned} T_f^k : \mathbb{F}_q^n &\longrightarrow \mathbb{F}_q^n \\ (x_1, x_2, \ldots, x_n) &\longmapsto (x_1, x_2, \ldots, f(x_k), \ldots, x_n) \end{aligned}$$

é uma isometria.

Exemplo 2.3.14. *Se π é uma bijeção do conjunto $\{1, 2, \ldots, n\}$ nele próprio, também chamada de permutação de $\{1, 2, \ldots, n\}$, a aplicação permutação de coordenadas*

$$\begin{aligned} T_\pi : \mathbb{F}_q^n &\longrightarrow \mathbb{F}_q^n \\ (x_1, x_2, \ldots, x_n) &\longmapsto (x_{\pi(1)}, x_{\pi(2)}, \ldots, x_{\pi(n)}) \end{aligned}$$

é uma isometria.

2.3.2 Matriz geradora de um código

Consideremos o corpo finito $\mathbb{F}_q$ e seja $\mathcal{C} \subset \mathbb{F}_q^n$ um $[n; k]$ código linear.

Definição 2.3.15. *Seja $\mathcal{B} = \{v_1, v_2, \ldots, v_k\}$ uma base ordenada de $\mathcal{C}$. A matriz geradora G de $\mathcal{C}$ associada à base $\mathcal{B}$ é a matriz cujas linhas são os vetores $v_i = (v_{i1}, v_{i2}, \ldots, v_{in})$, $i = 1, 2, \ldots, k$, isto é*

$$G = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_k \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \ldots & v_{1n} \\ v_{21} & v_{22} & \ldots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{k1} & v_{k2} & \ldots & v_{kn} \end{bmatrix}.$$

Note que a matriz G para um código $\mathcal{C}$ não é única, pois ela depende da escolha da base $\mathcal{B}$. De fato, uma base de um espaço vetorial pode ser obtida de uma outra qualquer através de sequencias de operações do tipo:

1. Permutação de dois elementos da base;
2. Multiplicação de um elemento da base por um escalar não nulo e;
3. Substituição de um vetor da base por ele mesmo somado com um múltiplo escalar de outro vetor da base.

Então, duas matrizes geradoras de um mesmo código $\mathcal{C}$ podem ser obtidas uma da outra por uma sequencia de operações, que chamamos de operações elementares, do tipo:

1. Permutação de duas linhas.
2. Multiplicação de uma linha por um escalar não nulo.
3. Adição de um múltiplo escalar de uma linha a outra.

Podemos também construir códigos a partir de matrizes geradoras G. Para isso, consideramos a transformação linear:

$$\begin{aligned} T : \mathbb{F}_q^k &\longrightarrow \mathbb{F}_q^n \\ x &\longmapsto xG \end{aligned}$$

Assim, temos um código $\mathcal{C} = T(\mathbb{F}_q^k)$. Nesse caso, consideramos $\mathbb{F}_q^k$ como sendo o código da fonte, $\mathcal{C}$, o código do canal e a transformação T, uma codificação.

Exemplo 2.3.16. *Seja* $G \in M(3, 5, \mathbb{F}_2)$ *dada por*

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

Considerando a transformação linear

$$\begin{aligned} T : \mathbb{F}_2^3 &\longrightarrow \mathbb{F}_2^5 \\ x &\longmapsto xG \end{aligned}$$

obtemos um código $\mathcal{C}$ *em* $\mathbb{F}_2^5$*, imagem de* T*. A palavra* 111 *do código da fonte, por exemplo, é codificada como* 10000*.*

Suponhamos agora que seja dada a palavra 10101 *do código, e que gostaríamos de decodificá-la, isto é, achar a palavra* $x \in \mathbb{F}_2^3$ *da qual ela provém por meio de* T*. Temos então, que resolver o sistema*

$$(x_1\ x_2\ x_3)G = (10101),$$

ou seja,

$$\begin{cases} x_1 + x_2 + x_3 &= 1 \\ x_2 + x_3 &= 0 \\ x_1 + x_3 &= 1 \\ x_2 + x_3 &= 0 \\ x_1 + x_3 &= 1, \end{cases}$$

cuja solução é $x_1 = 1$, $x_2 = 0$ *e* $x_3 = 0$. *Logo, a palavra* 10101 *é decodificada como* 100.

Da definição de uma matriz geradora G de um código $\mathcal{C}$ dado, para cada escolha de k linhas linearmente independentes de G, correspondem k coordenadas que são chamadas de *coordenadas ou conjunto de informação* de $\mathcal{C}$ e as $r = n - k$ coordenadas remanescentes são chamadas de *conjunto de redundância* de $\mathcal{C}$ e r de *redundância.*

Definição 2.3.17. *Diremos que uma matriz G de um código $\mathcal{C}$ está na forma padrão se é do tipo*

$$G = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & a_{11} & a_{12} & \dots & a_{1(n-k)} \\ 0 & 1 & 0 & \dots & 0 & a_{21} & a_{22} & \dots & a_{2(n-k)} \\ 0 & 0 & 1 & \dots & 0 & a_{31} & a_{32} & \dots & a_{3(n-k)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 & a_{k1} & a_{k2} & \dots & a_{k(n-k)} \end{bmatrix} = [Id_{k\times k} \,|\, A_{k\times(n-k)}].$$

Exemplo 2.3.18. *Dado um código $\mathcal{C}$, nem sempre é possível achar uma matriz geradora de $\mathcal{C}$ na forma padrão. Consideremos, o código em $\mathbb{F}_2^5$ de matriz geradora*

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

nunca poderá ter uma matriz geradora na forma padrão pois os vetores da base de $\mathcal{C}$ não geram vetores que tenham as duas primeiras coordenadas não nulas.

Teorema 2.3.19. *Se $G = [I_k \,|\, A]$ é uma matriz geradora na forma padrão de um $[n; k]$ código $\mathcal{C}$, então $H = [-A^t \,|\, I_{n-k}]$ é uma matriz de verificação de paridade de $\mathcal{C}$.*

Demonstração. Sejam $G \in M(k, n, \mathbb{F}_q)$ e $H \in M(n - k, n, \mathbb{F}_q)$ as matrizes geradora e de verificação do código $\mathcal{C}$ na forma padrão.

Note que

$$HG^t = \begin{bmatrix} -a_{11} & -a_{21} & \dots & -a_{k1} & 1 & 0 & \dots & 0 \\ -a_{12} & -a_{22} & \dots & -a_{k2} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_{1(n-k)} & -a_{2(n-k)} & \dots & -a_{k(n-k)} & 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ a_{11} & a_{21} & \dots & a_{k1} \\ a_{12} & a_{22} & \dots & a_{k2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1(n-k)} & a_{2(n-k)} & \dots & a_{k(n-k)} \end{bmatrix}$$

$$= \begin{bmatrix} -a_{11} + a_{11} & -a_{21} + a_{21} & \dots & -a_{k1} + a_{k1} \\ -a_{12} + a_{12} & -a_{22} + a_{22} & \dots & -a_{k2} + a_{k2} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{1(n-k)} + a_{1(n-k)} & -a_{2(n-k)} + a_{2(n-k)} & \dots & -a_{k(n-k)} + a_{k(n-k)} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}.$$

Logo, os vetores linhas de G, geradores de $\mathcal{C}$, estão contidos no núcleo da transformação linear $T_H(w) = Hw^t$. Mas, como H possui posto $n - k$ segue que

$$\dim(\ker(T_H)) = k = \dim(\mathcal{C})$$

e, assim, $\mathcal{C} = \ker(T_H)$, ou seja, H é a matriz de verificação de paridade de $\mathcal{C}$. □

Exemplo 2.3.20. *Seja $\mathcal{C} = \{000, 101, 202\} \subset \mathbb{F}_3^3$ um código linear sobre $\mathbb{F}_3$ com matriz geradora na forma padrão*

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

O Teorema 2.3.19 *nos fornece um algoritmo para determinar a matriz de verificação de paridade de um código através de sua matriz geradora desde que esta esteja na forma padrão.*

Temos que $G = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} = [I_1 \,|\, A_{1\times 2}]$, onde

$$I_1 = \begin{bmatrix} 1 \end{bmatrix} \text{ e } A_{1\times 2} = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

Assim, pelo teorema 2.3.19*, temos que $H = [-A_{1\times 2}^t \,|\, I_2]$. Logo,*

$$H = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix}.$$

Vamos verificar que H é de fato a matriz de verificação de paridade de $\mathcal{C}$. Seja $x \in \mathbb{F}_3^3$. Como a primeira linha da matriz H é o vetor (010), os vetores x de $\mathbb{F}_3^3$ que não possuem a segunda coordenada igual a zero podem sem desconsiderados e assim restam os vetores

$$000,\ 100,\ 001,\ 101,\ 200,\ 002,\ 202,\ 102 \text{ e } 201\,.$$

Para tais vetores, temos

$$\begin{array}{lll} (201)(000)^t = 0 & (201)(101)^t = 0 & (201)(202)^t = 0 \\ (201)(100)^t = 2 & (201)(200)^t = 1 & (201)(102)^t = 1 \\ (201)(001)^t = 1 & (201)(002)^t = 2 & (201)(201)^t = 2 \end{array}$$

Logo, como apenas os vetores $000, 101, 202 \in \mathbb{F}_2^3$ resultam no vetor nulo quando multiplicamos H por seus vetores transpostos, e $\mathcal{C} = \{000, 101, 202\}$, a matriz H tem a propriedade da matriz de verificação de paridade para o código $\mathcal{C}$.

2.3.3 Equivalência de códigos

Definição 2.3.21. *Dados dois códigos $\mathcal{C}$ e $\mathcal{C}'$ em $\mathbb{F}_q^n$, diremos que $\mathcal{C}'$ é equivalente a $\mathcal{C}$ se existir uma isometria f de $\mathbb{F}_q^n$ tal que $f(\mathcal{C}) = \mathcal{C}'$. Nesse caso, escrevemos $\mathcal{C} \cong \mathcal{C}'$.*

Note que, pela Proposição 2.3.12, a equivalência de códigos é uma relação de equivalência, isto é, possui as seguintes propriedades:

1. É reflexiva: todo código é equivalente a si próprio.
2. É simétrica: se o código $\mathcal{C}$ é equivalente ao código $\mathcal{C}'$, então $\mathcal{C}'$ é equivalente a $\mathcal{C}$.
3. É transitiva: se o código $\mathcal{C}$ é equivalente ao código $\mathcal{C}'$ e $\mathcal{C}'$ por sua vez é equivalente ao código $\mathcal{C}''$, então $\mathcal{C}$ é equivalente a $\mathcal{C}''$.

Decorre imediatamente da definição que dois códigos equivalentes têm os mesmos parâmetros.

Exemplo 2.3.22. *Voltemos ao Exemplo* 2.3.18 *em que consideramos o código em $\mathbb{F}_2^5$ de matriz geradora*

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

Como vimos, esse código nunca poderá ter uma matriz na forma padrão. No entanto, efetuando também permutações das colunas de G, podemos obter a matriz

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

que é a matriz geradora na forma padrão de um código $\mathcal{C}'$ equivalente a $\mathcal{C}$.

De modo mais geral, efetuando sequências de operações sobre a matriz geradora G de um código linear $\mathcal{C}$, do tipo:

- permutação de duas colunas,
- multiplicação de uma coluna por um escalar não nulo,

obtemos uma matriz geradora G' de um código $\mathcal{C}'$ equivalente a $\mathcal{C}$. Ao efetuarmos as operações acima numa base de $\mathcal{C}$ estamos também efetuando-as sobre todas as palavras do código.

Se pudermos utilizar a operação de permutação de duas colunas quaisquer de uma matriz geradora além das operações elementares sobre as linhas de uma matriz temos o seguinte resultado:

Teorema 2.3.23. *Dado um código $\mathcal{C}$, existe um código equivalente $\mathcal{C}'$ com matriz geradora na forma padrão.*

2.3.4 Códigos duais

Definição 2.3.24. *Dados $x = (x_1, x_2, \ldots, x_n)$, $y = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_q^n$, o produto interno formal de x por y é definido por*

$$\langle x, y \rangle = x_1 y_1 + x_2 y_2 + \ldots + x_n y_n.$$

Dado código $\mathcal{C}$ definimos o código dual de $\mathcal{C}$ como

$$\mathcal{C}^{\perp} = \{x \in \mathbb{F}_q^n \mid \langle x, c \rangle = 0, \ \forall c \in \mathcal{C}\}.$$

Proposição 2.3.25. *Dado um código $\mathcal{C}$, as matrizes G e H são matrizes geradora e de verificação de paridade de $\mathcal{C}$ se, e somente se, H e G são matrizes geradora e de verificação de paridade de $\mathcal{C}^{\perp}$, respectivamente.*

Demonstração. Sejam $\mathcal{C}$ um código, $G \in M(k, n, \mathbb{F}_q)$ e $H \in M(n-k, n, \mathbb{F}_q)$.

($\Rightarrow$) Seja $x = (x_1, x_2, \ldots, x_n) \in \mathcal{C}^{\perp}$.

Se $G = (a_{ij})$ e $H = (b_{ij})$ são matrizes geradora e de verificação de paridade de $\mathcal{C}$, respectivamente, temos que a j-ésima entrada de Gx^t é

$$a_{j1}x_1 + a_{j2}x_2 + \ldots + a_{jn}x_n. \tag{2.2}$$

Como cada vetor linha $a_j = (a_{j1}, a_{j2}, \ldots, a_{jn}) \in \mathcal{C}$, temos que a soma em (2.2) é o produto interno formal de a_j por x, isto é, $\langle a_j, x \rangle$, que por definição é igual a zero para todo $j = 1, 2, \ldots, k$, pois $a_j \in \mathcal{C}$.

Logo, $Gx^t = 0$ e assim, G é a matriz de verificação de paridade de $\mathcal{C}^\perp$.

Pelo Teorema 2.3.19, como $HG^t = GH^t = 0$ temos ainda que H é uma matriz geradora para $\mathcal{C}^\perp$.

É imediato verificarmos a volta, mas consideremos a principio que não conhecemos as matrizes geradora e de verificação de paridade para o código $\mathcal{C}$ dado. Ao invés disto, seja dado código $\mathcal{C}^\perp$ e consideremos as matrizes $\tilde{G} \in M(n-k, n, \mathbb{F}_q)$, $\tilde{H} \in M(k, n, \mathbb{F}_q)$.

($\Leftarrow$) Seja $y = (y_1, y_2, \ldots, y_n) \in \mathcal{C}$.

Se $\tilde{G} = (b_{ij})$ e $\tilde{H} = (a_{ij})$ são matrizes geradora e de verificação de paridade de $\mathcal{C}^\perp$, respectivamente, temos que a i-ésima entrada de $\tilde{G}y^t$ é

$$b_{i1}y_1 + b_{i2}y_2 + \ldots + b_{in}y_n. \tag{2.3}$$

Como cada vetor linha $b_i = (b_{i1}, b_{i2}, \ldots, b_{in}) \in \mathcal{C}^\perp$, temos que a soma em (2.3) é o produto interno formal de b_i por y, isto é, $\langle b_i, y\rangle$, que é igual a zero para todo $i = 1, 2, \ldots, n-k$, pois $b_i \in \mathcal{C}^\perp$.

Logo, $\tilde{G}y^t = 0$ e assim, $\tilde{G}$ é uma matriz de verificação de paridade de $\mathcal{C}$.

Pelo Teorema 2.3.19, como $\tilde{H}\tilde{G}^t = \tilde{G}\tilde{H}^t = 0$ temos que $\tilde{H}$ é uma matriz geradora para $\mathcal{C}$.

Note que na equação (2.2) poderíamos ter um vetor linha

$$\mu a_k + \lambda a_j = (\mu a_{k1} + \lambda a_{j1}) + (\mu a_{k2} + \lambda a_{j2}) + \ldots + (\mu a_{kn} + \lambda a_{jn}) \in \mathcal{C}$$

em vez do vetor $a_j = (a_{j1}, a_{j2}, \ldots, a_{jn}) \in \mathcal{C}$ na j-ésima entrada de G e ainda assim, para Gx^t teríamos

$$(\mu a_{k1} + \lambda a_{j1})x_1 + (\mu a_{k2} + \lambda a_{j2})x_2 + \ldots + (\mu a_{kn} + \lambda a_{jn})x_n \tag{2.4}$$

que é o produto interno formal de $\mu a_k + \lambda a_j$ por x, tal que

$$\langle \mu a_k + \lambda a_j, x\rangle = \langle \mu a_k, x\rangle + \langle \lambda a_j, x\rangle = \mu\langle a_k, x\rangle + \lambda\langle a_j, x\rangle = \mu \cdot 0 + \lambda \cdot 0 = 0 + 0 = 0.$$

Com isto mostramos que as matrizes $\tilde{G} = H$ e $\tilde{H} = G$ para os códigos $\mathcal{C}$ e $\mathcal{C}^\perp$, a menos de operações elementares sobre suas linhas, e concluímos assim a demonstração. □

Exemplo 2.3.26. *Sejam $\mathcal{C} = \{\lambda(01) \ : \ \lambda \in \mathbb{F}_5\}$ um código linear com matrizes geradora e de verificação de paridade $G, H \in M(1, 2, \mathbb{F}_5)$, respectivamente, dadas por*

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

Por definição, o código dual a $\mathcal{C}$, $\mathcal{C}^\perp$, é dado por

$$\begin{aligned} \mathcal{C}^\perp &= \{x \in \mathbb{F}_5^2 \ : \ \langle x, c\rangle = 0, \ \forall c \in \mathcal{C}\} \\ &= \{00, 10, 20, 30, 40\} \\ &= \{\alpha(10), \ \alpha \in \mathbb{F}_5\}. \end{aligned}$$

Desta maneira, temos $\tilde{G}, \tilde{H} \in M(1,2,\mathbb{F}_5)$ *as matrizes geradora e de verificação de paridade de* $\mathcal{C}^\perp$*, respectivamente, dadas por*

$$\tilde{G} = \begin{bmatrix} 1 & 0 \end{bmatrix} \text{ e } \tilde{H} = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

Note que escolhemos $\tilde{G}$ *e* $\tilde{H}$ *desta maneira apenas por conveniência pois de fato,* $\tilde{G}$ *e* $\tilde{H}$ *poderiam ser*

$$\tilde{G} = \begin{bmatrix} \alpha & 0 \end{bmatrix} \text{ e } \tilde{H} = \begin{bmatrix} 0 & \beta \end{bmatrix}$$

para quaisquer $\alpha, \beta \in \mathbb{F}_5$ *não nulos. Portanto,* $\tilde{G} = H$ *e* $\tilde{H} = G$ *a menos de isomorfismos.*

Definição 2.3.27. *Um código é dito auto-ortogonal se* $\mathcal{C} \subset \mathcal{C}^\perp$ *e auto dual se* $\mathcal{C} = \mathcal{C}^\perp$*.*

Exemplo 2.3.28. *Considere o conjunto* $\mathcal{C} = \{0000, 1111\}$ *subespaço de* $\mathbb{F}_2^4$*.*

Dado $x = (x_1, x_2, x_3, x_4) \in \mathcal{C}^\perp$*, sabemos que*

$$\langle (x_1, x_2, x_3, x_4), (1,1,1,1) \rangle = x_1 + x_2 + x_3 + x_4 = 0.$$

Como $x_i \in \{0,1\}$ *para* $i = 1,2,3,4$*, o vetor* x *deve ter um quantidade par de coordenadas iguais. Desta maneira,*

$$\mathcal{C}^\perp = \{0000, 1111, 1010, 0101, 1001, 0110\}.$$

Como $\mathcal{C} \subset \mathcal{C}^\perp$*,* $\mathcal{C}$ *é um código auto-ortogonal.*

2.3.5 Correção de erros

Dada uma métrica d em $\mathbb{F}_q^n$, temos uma boa maneira de tentar corrigir erros. Vamos supor, como antes, que a palavra transmitida foi $v \in \mathcal{C}$ e a palavra recebida foi $w \notin \mathcal{C}$. Se a métrica em si for razoável (considerando-se as características físicas do canal de transmissão de informação), é plausível supormos que a probabilidade de w estar "longe" de v é menor que a probabilidade de estar "perto" no sentido de que para quaisquer $\alpha, \beta \in \mathbb{R}$ e $m > 0$ com $0 \leq \beta \leq \alpha$, então a probabilidade de termos $\alpha \leq d(v,w) \leq \alpha + m$ é menor ou igual que a probabilidade de termos $\beta \leq d(v,w) \leq \beta + m$. Obviamente esta probabilidade é definida pelas características físicas do canal de transmissão mas como neste texto não estamos tratando de canais específicos, vamos sempre assumir que esta hipótese é verdadeira.

Com isto, passamos a ter um mecanismo sistemático para corrigir erros: se a palavra recebida $x \notin \mathcal{C}$, escolhemos o ponto w de $\mathcal{C}$ mais próximo de x, ou seja, escolhemos $w \in \mathcal{C}$ tal que

$$d(x,w) = \inf\{d(x,u) \ : \ u \in \mathcal{C}\}.$$

Note que o ínfimo é atingido pois $\mathcal{C}$ é um conjunto finito.

Do mesmo modo que ocorre com a detecção de erros, a correção de erros pode falhar e temos na realidade que entender quais são suas limitações. Assim, acrescentar às restrições já determinadas(a estrutura vetorial do código $\mathcal{C}$ e a estrutura métrica $d(\cdot;\cdot)$ definida em $\mathbb{F}_q^n$) as seguintes condições:

1. A métrica definida é compatível com a estrutura vetorial, no sentido de ser invariante por translações, isto é, para quaisquer $x, y, z \in \mathbb{F}_q^n$
$$d(x+y, x+z) = d(y, z)\,.$$

2. A métrica é compatível com a estrutura discreta de um espaço finito, no sentido que $d(x, y) \in \mathbb{N}$ para quaisquer $x, y \in \mathbb{F}_q^n$.

Definição 2.3.29. *Seja d a distância mínima de um código $\mathcal{C}$. O raio do código é definido como o inteiro não-negativo*
$$r = \left\lfloor \frac{d-1}{2} \right\rfloor .$$

Note que dados $v, w \in \mathcal{C}$, as bolas $B(v; r)$ e $B(w; r)$ são sempre disjuntas. De fato, supondo que u pertença a interseção destas, temos pela desigualdade triangular que
$$\begin{aligned} d(v, w) &\leq d(v, u) + d(u, w) \\ &\leq \left\lfloor \frac{d-1}{2} \right\rfloor + \left\lfloor \frac{d-1}{2} \right\rfloor \\ &\leq d - 1\,, \end{aligned}$$
uma contradição, pois, por definição, $d \leq d(v, w)$.

De modo geral, dizemos que a *capacidade de correção do código* é R se podemos garantir que, caso a palavra transmitida v e a recebida w distem no máximo R, então temos certeza de estar corrigindo o código. Assim, temos que se a distância entre a mensagem enviada v e a mensagem recebida w for menor ou igual a $\left\lfloor \frac{d-1}{2} \right\rfloor$, temos que v é o ponto de $\mathcal{C}$ mais próximo de w e estamos de fato corrigindo o erro ocorrido durante a transmissão da mensagem. Neste caso, dizemos que $\mathcal{C}$ é um $[n; k; d]$ código.

Geometricamente, podemos pensar na capacidade de correção como sendo o maior natural R tal que
$$B(v; R) \cap B(w; R) = \emptyset$$
para quaisquer $v, w \in \mathcal{C}$ distintos e, assim, podemos definir o importante conceito de raio de empacotamento:

Definição 2.3.30. *O raio de empacotamento, $R_e(\mathcal{C})$, de um código $\mathcal{C}$ é o raio máximo que nos permite empacotar bolas disjuntas centradas nas palavras do código. Formalmente, definimos o raio de empacotamento como*
$$R_e(\mathcal{C}) = \max\{r \in \mathbb{N} \ : \ B(v; r) \cap B(w; r) = \emptyset,\ \forall v, w \in \mathcal{C},\ v \neq w\}\,.$$

Note que o raio $r = \left\lfloor \frac{d-1}{2} \right\rfloor$ de um código $\mathcal{C}$ é apenas um limitante inferior para o raio de empacotamento $R_e(\mathcal{C})$, este sim definindo a capacidade de correção do código.

Note que o raio de empacotamento $R_e(\mathcal{C})$ é exatamente o inteiro tal que as esferas de centro em cada elemento de $\mathcal{C}$ e raio $R_e(\mathcal{C})$ são duas a duas disjuntas. Esse raio está relacionado com o problema de empacotamento de esferas, que consiste em dispor esferas de mesmo raio no espaço de tal modo que a interseção de duas delas tenha no máximo um ponto. O objetivo do empacotamento é encontrar um arranjo de esferas idênticas de tal forma que a fração do espaço coberto por essas esferas seja o maior possível. Nesse sentido, estamos interessados em códigos com raio de empacotamento suficientemente grande de forma que consigamos empacotar todo o espaço $\mathbb{F}_q^n$.

Encerramos esse capítulo com a noção de taxa de informação do código.

As k variáveis que definem a dimensão do código e a cardinalidade q do corpo definem o tamanho q^k do alfabeto, que, por sua vez, é determinado pela natureza da informação que desejamos transmitir e, por isso, estas k variáveis são chamadas de variáveis de informação. Quando escolhemos um subespaço k-dimensional de um espaço sobre $\mathbb{F}_q$ de dimensão $n > k$, estamos introduzindo variáveis que não contém informação adicional, mas que são usadas para detectar e corrigir erros e por isto são chamadas de variáveis de controle. Assim como o tamanho do alfabeto, a capacidade de correção do código R_e também é determinada a *priori* pela natureza da informação assim como, por exemplo, dados referentes a operações bancárias necessitam de confiabilidade maior que a transmissão de televisão. Assim, tendo q^k e R_e definidos pela natureza da informação, boa parte do desafio na Teoria de Códigos Corretores de Erros é buscar códigos em que a relação entre as variáveis de informação e as de controle seja a maior possível, ou seja, em que precisemos adicionar o mínimo de variáveis de controle. Esta relação k/n entre as variáveis de informação e o total de variáveis é chamada de *taxa de informação* do código.

Obviamente, qualquer que seja a taxa de informação, não podemos ter a certeza de efetivamente corrigir todos os erros de transmissão, mas, do mesmo modo que ocorre com a detecção, podemos corrigir erros com grau de segurança tão grande quanto desejado.

3 MÉTRICAS EM ESPAÇOS DE CÓDIGOS

O principal objetivo deste capítulo é apresentar as principais métricas em espaços de códigos utilizadas para descrever como ocorre a discretização destes espaços. Nesse capítulo, usamos as referências [2] e [8].

3.1 MÉTRICA DE HAMMING

A distância de Hamming entre dois vetores $x, y \in \mathbb{F}_q^n$ é simplesmente o número de coordenadas distintas entre estes dois vetores:

Definição 3.1.1. *Dados $x, y \in \mathbb{F}_q^n$ com $x = (x_1, x_2, \ldots, x_n)$ e $y = (y_1, y_2, \ldots, y_n)$, a distância de Hamming é definida como*

$$d_H(x, y) = \#\{i \ : \ x_i - y_i \neq 0, \ i = 1, 2, \ldots, n\}\,.$$

Definimos, ainda, o peso de Hamming de x como

$$\omega_H(x) = d_H(x, 0)\,.$$

Proposição 3.1.2. *d_H é uma métrica em $\mathbb{F}_q^n$.*

Demonstração. Sejam $x, y, z \in \mathbb{F}_q^n$ com

$$x = (x_1, x_2, \ldots, x_n),\ y = (y_1, y_2, \ldots, y_n) \text{ e } z = (z_1, z_2, \ldots, z_n)\,.$$

Para mostrarmos a primeira propriedade de métrica, basta notarmos que, como $d_H(x, y)$ é a quantidade de coordenadas diferentes entre x e y, $d_H(x, y)$ é sempre maior ou igual a zero, valendo a igualdade se, e somente se, as i-ésimas coordenadas x_i e y_i são iguais para todo i.

Do mesmo modo, a segunda propriedade é satisfeita pois as i-ésimas coordenadas de x que o diferem de y também são as i-ésimas coordenadas de y que o diferem de x. Isto é, $d_H(x, y) = d_H(y, x)$.

Por fim, a desigualdade triangular é mostrada a partir do raciocínio que a contribuição das i-ésimas coordenadas de x e y para $d_H(x, y)$ é igual a zero se $x_i = y_i$, e igual a um se $x_i \neq y_i$. No caso em que a contribuição é igual a zero, certamente a contribuição das i-ésimas coordenadas a $d_H(x, y)$ é menor ou igual a das i-ésimas coordenadas a $d_H(x, z) + d_H(z, y)$ que pode assumir os valores 0, 1 ou 2. No outro caso, temos que $x_i \neq y_i$ e, portanto, não podemos ter $x_i = z_i$ e $z_i = y_i$. Consequentemente, a contribuição das i-ésimas coordenadas a $d_H(x, z) + d_H(z, y)$ é maior ou igual a um, que é a contribuição das i-ésimas coordenadas a $d_H(x, y)$. □

O exemplo a seguir mostra que podemos construir códigos usando a distância de Hamming que nos dão bons empacotamentos de esferas.

Exemplo 3.1.3. *Consideremos em* $\mathbb{F}_2^{2^2-1} = \mathbb{F}_2^3$ *o código* $\mathcal{C}$ *que é definido a partir da matriz de paridade* $H \in M(2,3,\mathbb{F}_2^3)$ *dada por*

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

Como H *é uma matriz* 2×3, $\mathcal{C}$ *é um código com dimensão* 1 *em* $\mathbb{F}_2^3$ *gerado por* 111, *bastando para isto verificar que*

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

Note ainda que

$$B_H(000;1) = \{000, 100, 010, 001\}$$

e

$$B_H(111;1) = \{111, 011, 101, 110\}.$$

Desta maneira,

$$B_H(000;1) \cap B_H(111;1) = \emptyset$$

e

$$\mathbb{F}_2^3 = B_H(000;1) \cup B_H(111;1).$$

Ou seja, não apenas podemos empacotar todas as bolas unitárias em pontos do código, como também estas recobrem todo o espaço $\mathbb{F}_2^3$.

Vimos anteriormente que se $d = d(\mathcal{C})$ for a distância mínima do código e $R_e(\mathcal{C})$ o seu raio de empacotamento, então

$$\left\lfloor \frac{d-1}{2} \right\rfloor \leq R_e(\mathcal{C}) < d\,,$$

independentemente da métrica em consideração. Vamos mostrar que, no caso de uma métrica de Hamming, a situação é bem melhor definida.

Proposição 3.1.4. *Considerando em* $\mathbb{F}_q^n$ *a métrica de Hamming, temos que para todo código linear* $\mathcal{C} \subset \mathbb{F}_q^n$, $\left\lfloor \frac{d-1}{2} \right\rfloor = R_e(\mathcal{C})$.

Demonstração. Vamos mostrar que se $r > \left\lfloor \dfrac{d-1}{2} \right\rfloor$ então existem $u, v \in \mathcal{C}$ tais que

$$B_H(u;r) \cap B_H(v;r) \neq \emptyset\,.$$

Se $d = 2k + \varepsilon$, com $\varepsilon \in \{0, 1\}$, então

$$k + \varepsilon = \left\lfloor \frac{d-1}{2} \right\rfloor + 1 \,.$$

Como existe $u \in \mathcal{C}$, $u = (u_1, u_2, \ldots, u_n)$ tal que $\omega_H(u) = d$. Temos então que u possui exatamente $2k + \varepsilon$ coordenadas não nulas. Suponhamos que estas sejam as coordenadas no conjunto $I \cup J \cup \{l_\varepsilon\}$ onde

$$I = \{i_1, i_2, \ldots, i_k\}$$

$$J = \{j_1, j_2, \ldots, j_k\}$$

e $\{l_\varepsilon\} = \emptyset$ se $\varepsilon = 0$. Seja $x = (x_1, x_2, \ldots, x_n)$ o vetor definido por

$$x_m = \begin{cases} u_m, & \text{se} \quad m \notin I \cup \{l_\varepsilon\}, \\ 0, & \text{se} \quad m \in I \cup \{l_\varepsilon\}. \end{cases}$$

Temos então que

$$d_H(x, u) = \#(I \cup \{l_\varepsilon\}) = k + \varepsilon$$

e

$$d_H(x, 0) = \#J = k \,.$$

Assim, $x \in B_H(0; k + \varepsilon) \cap B_H(u; k + \varepsilon)$, de modo que $R_e(\mathcal{C}) < k + \varepsilon = \left\lfloor \frac{d-1}{2} \right\rfloor + 1$ e concluímos que $R_e(\mathcal{C}) = \left\lfloor \frac{d-1}{2} \right\rfloor$. □

3.2 MÉTRICA DE LEE

Se considerarmos um vetor $x = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_q^n$, o peso de Hamming $\omega_H(x)$ identifica o número de coordenadas não nulas de x, mas ignora totalmente o valor assumido por estas coordenadas quando estes não se anulam.

Se considerarmos que $0 \leq x_i \leq p - 1$ para cada $i = 1, 2, \ldots, n$ e que estamos identificando os inteiros com o resto de sua divisão por p, podemos identificar $\mathbb{F}_p \cong \mathbb{Z}_p$ com as i-ésimas raízes da unidade

$$\{e^0, e^{2\pi i/p}, e^{4\pi i/p}, \ldots, e^{2(p-1)\pi i/p}\},$$

que são vértices de um polígono regular de p lados no plano.

Definição 3.2.1. *Definimos a distância de Lee $|a - b|_L$ entre dois pontos $a, b \in \mathbb{F}_p \cong \mathbb{Z}_p$ como sendo o menor número de arestas de um polígono regular de p lados que precisamos percorrer para ligar os vértices $e^{2a\pi i/p}$ e $e^{2b\pi i/p}$, isto é,*

$$|a - b|_L = \min\{|a - b|, p - |a - b|\}$$

onde $|\cdot|$ é o valor absoluto usual.

Definimos a métrica de Lee $d_L(x,y)$ entre dois pontos $x, y \in \mathbb{F}_p^n$ como

$$d_L(x,y) = d_L((x_1, x_2, \ldots, x_n), (y_1, y_2, \ldots, y_n)) = \sum_{i=1}^{n} |x_i - y_i|_L.$$

Ainda, definimos o peso de Lee de um elemento $x \in \mathbb{F}_p^n$ como

$$\omega_L(x) = d_L(x, 0)\,.$$

Observação 3.2.2. *Como usamos o valor absoluto usual para definir a distância de Lee, temos diretamente que essa é uma métrica.*

Exemplo 3.2.3. *Em $\mathbb{F}_7^4$, consideremos os vetores $u = (3,1,4,0)$ e $w = (2,5,4,3)$. As distâncias entre u e w pelas métricas de Hamming e de Lee, respectivamente, são:*

$$\begin{aligned}
d_H(u,w) &= \#\{i : u_i \neq w_i\} = 3 \text{ e} \\
d_L(u,w) &= \sum_{i=1}^{4} |u_i - w_i|_L = \sum_{i=1}^{4} \min\{|u_i - w_i|, 7 - |u_i - w_i|\} \\
&= \min\{|3-2|, 7-|3-2|\} + \min\{|1-5|, 7-|1-5|\} \\
&\quad + \min\{|4-4|, 7-|4-4|\} + \min\{|0-3|, 7-|0-3|\} \\
&= \min\{1,6\} + \min\{4,3\} + \min\{0,7\} + \min\{3,4\} \\
&= 1 + 3 + 0 + 3 = 7.
\end{aligned}$$

Portanto, a distância de Lee, em alguns casos, é mais eficiente no sentido de que pode separar mais os vetores do espaço, fazendo com que a detecção e a correção de erros em um código com métrica de Lee possam ser otimizadas.

Observação 3.2.4. *Note que se $p = 2$ ou $p = 3$, a métrica de Lee coincide com a métrica de Hamming. Com efeito, se $a, b \in \mathbb{Z}_2$ então*

$$|a-b|_L = \begin{cases} \min\{0,2\} = 0, & \text{se } a = b \\ \min\{1,1\} = 1, & \text{se } a \neq b\,. \end{cases}$$

Agora, se $a, b \in \mathbb{Z}_3$, então se

$$|a-b|_L = \begin{cases} \min\{0,3\} = 0, & \text{se } a = b \\ \min\{1,2\} = 1, & \text{se } a \neq b\,. \end{cases}$$

Logo, para $a, b \in \mathbb{Z}_p$, com $p = 2$ ou $p = 3$ temos

$$|a-b|_L = \begin{cases} 0, & \text{se } a = b \\ 1, & \text{se } a \neq b \end{cases}\,.$$

A métrica de Lee também nos dá bons recobrimentos de esferas.

Exemplo 3.2.5. *Consideremos em* $\mathbb{F}_5^2$ *o código* $\mathcal{C} = \{\lambda(12) \ : \ \lambda \in \mathbb{F}_5\}$.

Primeiramente, vamos calcular o raio do código r*, como na definição* 2.3.29*:*

$$\begin{aligned} d_L(12, 00) &= \omega_L(12) \\ &= \sum_{i=1}^{2} |x_i - 0|_L \\ &= |1| + |2| = 3. \end{aligned}$$

Note que $d_L(12, 00) = d_L(\lambda(12), 00)$ *para todo* $\lambda \in \mathbb{F}_5$ *não nulo pois*

$$|1|_L = |4|_L = \min\{1, 4\} = 1\,,$$

$$|2|_L = |3|_L = \min\{2, 3\} = 2\,.$$

Logo,

$$r = \left\lfloor \frac{d_L(\mathcal{C}) - 1}{2} \right\rfloor = \left\lfloor \frac{3-1}{2} \right\rfloor = 1\,.$$

Agora, vamos descrever as bolas unitárias, relativas à métrica de Lee, centradas nas palavras do código. Primeiro, centrada em 00*, temos*

$$B_L(00; 1) = \{00, 10, 40, 01, 04\}.$$

As outras bolas são obtidas através de uma translação de seu centro:

$$\begin{aligned} B_L(\lambda(12); 1) &= \lambda(12) + B_L(00; 1) \\ &= \{\lambda(12) + x \ : \ x \in B_L(00; 1)\}. \end{aligned}$$

Assim,

$$B_L(\lambda(12); 1) \cap B_L(\alpha(12); 1) = \emptyset,$$

quando $\lambda \neq \alpha$*. Portanto, como* $\#(B_L(\lambda(12); 1)) = 5$*, para cada* $\lambda \in \mathbb{F}_5$*, temos*

$$\# \left(\sum_{\lambda \in \mathbb{F}_5} B_L(\lambda(12); 1)) \right) = 5 \cdot 5 = \#(\mathbb{F}_5^2),$$

e assim, as bolas unitárias centradas nos elementos do código recobrem todo o espaço $\mathbb{F}_5^2$*.*

Observação 3.2.6. *Na definição* 3.2.1*, utilizamos um corpo finito* $\mathbb{F}_p \cong \mathbb{Z}_p$*, onde* p *é primo. Em geral, podemos definir, de forma análoga, uma métrica de Lee para vetores com entrada no grupo aditivo* $\mathbb{Z}_q$*, isto é, dos inteiros módulo* q*, com* q *não necessariamente primo.*

3.3 MÉTRICAS PONDERADAS

Nessa seção, estudaremos ordens parciais, apresentando alguns exemplos e, em seguida, a noção de ideal necessária para a definição das métricas ponderadas.

3.3.1 **Ordens parciais**

Considere um conjunto finito com n elementos. Sem perda de generalidade vamos assumir que este é o conjunto que contém os naturais $1, 2, \ldots, n$ e denotá-lo por

$$[n] := \{1, 2, \ldots, n\}\,.$$

Definição 3.3.1. *Uma ordem parcial P em um conjunto X é um subconjunto $R \subset X \times X$ satisfazendo as seguintes condições:*

1. $(x, x) \in R,\ \forall x \in X$;
2. *Dados $x, y \in X$, se $(x, y) \in R$ e $(y, x) \in R$, então $x = y$;*
3. *Se $(x, y) \in R$ e $(y, z) \in R$, então $(x, z) \in R$, $\forall x, y, z \in X$.*

Neste caso, dizemos que X é ordenado por R e usamos a notação $P = (X, R)$. Ainda, P é chamado de conjunto parcialmente ordenado (abreviadamente, poset - do inglês partial ordered set).

A fim de simplificar a notação, denotaremos a relação R por $\leq$ quando não houver ambiguidade. Se um elemento $a \in X$ se relaciona com um elemento $b \in X$, isto é $a \leq b$ ou $b \leq a$, dizemos que a e b são *comparáveis*, caso contrário eles são ditos *incomparáveis*.

Dado um poset $(X, \leq)$ tal que o conjunto X é finito dizemos que o poset é um *poset finito*. A cardinalidade do conjunto X é chamada de *comprimento do poset*.

Definição 3.3.2. *Dado um poset finito $(X, \leq)$ a representação gráfica deste poset é chamada de diagrama de Hasse do poset. Os elementos de X são representados por vértices e as comparações entre dois elementos $a, b \in X$ são representadas por arestas, onde se convenciona que um elemento a está abaixo de b se, e somente se, $a \leq b$ e não existe $c \neq a$ e $c \neq b$ tal que $a \leq c \leq b$.*

Na figura 2, por exemplo, temos representada uma ordem parcial $\leq$ tal que

$$a \leq b\,,\ c \leq b \text{ e } d \leq e\,.$$

Assim, os pares de elementos (a, b), (b, c) e (d, e) são comparáveis; já os pares de elementos (a, c), (a, d), (a, e), (b, d), (b, e), (c, d) e (c, e) são incomparáveis.

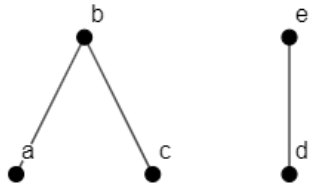

Figura 2 – Exemplo de diagrama de Hasse.

3.3.2 **Exemplos de posets**

Definição 3.3.3. *O poset dado pela ordem total (ou cadeia) é definido como*

$$R = \{(x, y) \in [n] \times [n] \ : \ x \leq y\},$$

onde $\leq$ *é a ordem usual dos reais.*

Definição 3.3.4. *O poset anti-cadeia é definido como*

$$R = \{(x, y) \in [n] \times [n] \ : \ x = y\}.$$

Note que na ordem anti-cadeia cada elemento é comparável consigo mesmo e consequentemente, quaisquer dois elementos distintos são incomparáveis.

Os diagramas de Hasse para as ordens cadeia e anti-cadeia no conjunto dos naturais $\{1, 2, \ldots, n\}$ estão representados na figura 3:

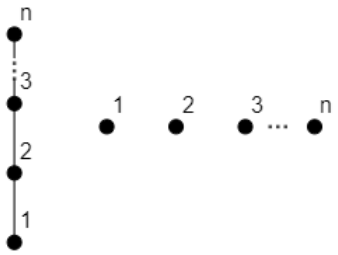

Figura 3 – À esquerda, o poset cadeia. À direita, o poset anti-cadeia.

Definição 3.3.5. *Se* $n = 2k$*, definimos o poset coroa* R *a partir das desigualdades:*

$$\begin{aligned} x \leq x \ &para \ todo \ x \in [n], \\ i \leq k + i \ &para \ todo \ i = 1, 2, \ldots, k-1, \\ i + 1 \leq k + i \ &para \ todo \ i = 1, 2, \ldots, k-1, \\ 1 \leq 2k \ &e \ k \leq 2k. \end{aligned}$$

Exemplo 3.3.6. *Seja* $n = 8$ *na definição* 3.3.5*. Temos que:*

- $1 \leq 1,\ 2 \leq 2,\ \ldots,\ 8 \leq 8$;
- $1 \leq 5,\ 2 \leq 6$ *e* $3 \leq 7$;
- $2 \leq 5,\ 3 \leq 6$ *e* $4 \leq 7$;
- $1 \leq 8$ *e* $4 \leq 8$,

e, assim, obtemos o diagrama de Hasse para a ordem coroa em [8] *na figura* 4.

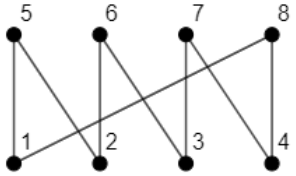

Figura 4 – Poset coroa em [8].

Definição 3.3.7. *Sejam* $0 = n_0 < n_1 < \cdots < n_{k-1} < n_k = n$. *Dados* $x, y \in [n]$, *existem únicos* $i = i(x), j = j(y) \in \{0, 1, \ldots, k\}$ *tais que* $n_i + 1 \leq x \leq n_{i+1}$ *e* $n_j + 1 \leq y \leq n_{j+1}$. *Se* $x \neq y$ *dizemos que* $x \leq y$ *se, e somente se,* $i < j$. *Denotamos esta ordem por* $P = [n_1, n_2, \ldots, n_k]$ *e a chamamos de ordem* $[n_1, n_2, \ldots, n_k]$*-semi-fraca.*

Exemplo 3.3.8. *Vejamos como proceder com a definição de uma ordem semi-fraca no caso de um poset com ordem* $[1, 4]-$*semi-fraca no conjunto* [4].

Primeiramente devemos identificar os índices n_i *que descrevem as relações de ordem. Sejam* $n_0 = 0$, $n_1 = 1$ *e* $n_2 = 4$ *de forma que* $0 = n_0 < n_1 < n_2 = 4$.

Agora, seja dado $1 \in [4]$.

Devemos determinar o índice $i = i(1)$ *tal que* $n_i + 1 \leq 1 \leq n_{i+1}$, *que é único pela definição.*

Ora, pela primeira desigualdade, como $n_i + 1 \leq 1$ *e* $n_i \geq 0$ *temos* $n_i = 0$ *e assim* $i = 0$. *Desta maneira, qualquer que seja* $y \in [4]$ *com* $y \neq 1$, $j = j(y) > i = 0$. *Logo,* $1 \leq y$ *para qualquer* $y = 2, 3, 4$.

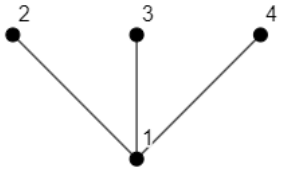

Figura 5 – Poset em [4] usando a ordem semi-fraca.

Note que tomamos o elemento $1 \in [4]$ *pois* $1 = n_1$ *e poderíamos compará-lo com os demais* n_i*'s e de fato obtemos* $1 = n_0 + 1 \leq 1 \leq n_{0+1} = 1$*. Observe também que sempre que tivermos uma ordem* $[1, n]$*-semi-fraca em* $[n]$ *o poset mantém essa mesma estrutura vista na figura* 5*.*

Exemplo 3.3.9. *Vejamos agora um outro exemplo considerando um poset* $[3, 4]$*-semi-fraco no conjunto* $[4]$*.*

Novamente, identificamos $n_0 = 0$*,* $n_1 = 3$ *e* $n_2 = 4$ *de forma que* $0 = n_0 < n_1 < n_2 = 4$*.*

Agora, seja dado $4 \in [4]$*.*

Queremos determinar o índice $j = j(4)$ *tal que* $n_j + 1 \leq 4 \leq n_{j+1}$*, e além disso, este* j *é único pela definição.*

Caso $j = 0$*, temos* $1 = 0 + 1 = n_0 + 1 \leq 4 \leq n_{0+1} = 3$ *que é uma contradição com a definição. Logo,* $j = 1$*. Note que* j *não pode ser igual a dois pois* n_3 *não foi definido então não sabemos se de fato* $4 \leq n_3$*.*

Desta maneira, qualquer que seja $x \in [4]$ *com* $x \neq 4$*,* $i = i(x) \leq j = 1$*. Logo,* $x \leq 4$ *para qualquer* $x = 1, 2, 3$*.*

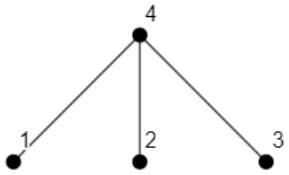

Figura 6 – Poset em [4] usando a ordem semi-fraca.

3.3.3 Ideais em ordens parciais e métricas

Definição 3.3.10. *Um ideal em uma ordem* $P = (X, \leq)$ *é um subconjunto* $I \subset X$ *que contém todos os elementos de* X *menores que algum elemento de* I*, ou seja, se* $x \in X, y \in I$ *e* $x \leq y$*, então* $x \in I$*. Dado* $J = \{x_1, x_2, \ldots, x_r\} \subset X$*, chamamos de ideal gerado por* J *o menor ideal de* X *contendo* J*, usando qualquer uma das notações* $\langle J \rangle$ *ou* $\langle x_1, x_2, \ldots, x_r \rangle$*.*

Como X é um ideal contendo J e a interseção de ideais é um ideal, temos que o ideal gerado por um conjunto sempre existe e

$$\langle J \rangle = \bigcap_{\substack{\text{I é ideal} \\ J \subset I}} I.$$

Usando o conceito de ideais em uma ordem parcial, podemos induzir uma métrica no espaço vetorial $\mathbb{F}_q^n$.

Definição 3.3.11. *Dado $x = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_q^n$, definimos suporte de x como sendo o conjunto de coordenadas não nulas de x, ou seja,*

$$supp(x) := \{i \in [n] \ : \ x_i \neq 0\}.$$

Dada uma ordem P em $[n]$ definimos o P-peso ponderado de x como sendo a cardinalidade do ideal gerado pelo suporte de x:

$$\omega_P(x) = \#(\langle supp(x)\rangle)\,.$$

Definição 3.3.12. *Usando o P-peso, definimos a métrica ponderada (por P) de modo similar a relação estabelecida entre peso e métrica nos casos de Hamming e Lee:*

$$d_P(x, y) := \omega_P(x - y).$$

Exemplo 3.3.13. *Em $\mathbb{F}_{101}^4$, consideremos os vetores $u = (18, 4, 97, 100)$ e $w = (18, 4, 97, 78)$. As distâncias entre u e w pelas métricas de Hamming e ponderada pela ordem $[3, 4]$-semi-fraca (exemplo 3.3.9 e figura 6), respectivamente, são:*

$$\begin{aligned} d_H(u, w) &= \#\{i : u_i \neq w_i\} = 1 \ e \\ d_P(u, w) &= \omega_P(u - w) \\ &= \omega_P((18, 4, 97, 100) - (18, 4, 97, 78)) \\ &= \omega_P(0, 0, 0, 22) \\ &= \#(\langle supp(0, 0, 0, 22)\rangle) \\ &= \#\langle 4\rangle = 4. \end{aligned}$$

Portanto, a distância ponderada por esta ordem, nesse caso, é mais eficiente que a métrica de Hamming, da mesma forma que comparamos as métricas de Hamming e de Lee no exemplo 3.2.3.

Proposição 3.3.14. *Se P é uma ordem em $[n]$, então d_P é uma métrica em $\mathbb{F}_q^n$.*

Demonstração. Para mostrarmos a primeira propriedade de métrica basta notarmos que como $d_P(x, y)$ é a cardinalidade do ideal gerado pelo suporte de $x - y$, $d_P(x, y)$ é sempre maior ou igual a zero, sendo igual a zero se, e somente se, o ideal é gerado pelo vetor nulo, ou seja, $y = x$.

Da mesma maneira, a segunda propriedade é satisfeita pois como $\text{supp}(x - y)$ é o conjunto de coordenadas não nulas de $x - y$, o vetor $y - x$ possui exatamente as mesmas coordenadas não nulas, isto é, $\text{supp}(x - y) = \text{supp}(y - x)$ resultando em $\langle \text{supp}(x - y)\rangle = \langle \text{supp}(y - x)\rangle$. Portanto, possuem a mesma cardinalidade e assim $d_P(x, y) = d_P(y, x)$.

Por fim, para mostrarmos que $d_P(x, y) \leq d_P(x, z) + d_P(z, y)$ começamos observando que $d_P(x + z, y + z) = d_P(x, y)$ para quaisquer $x, y, z \in \mathbb{F}_q^n$ já que $(x + z) - (y + z) = x - y$, de modo que ambos os conjuntos tem o mesmo suporte.

Assim, substituindo (x, y), (x, z) e (z, y) respectivamente por

$$(x - y, y - y), (x - z, z - z) \text{ e } (z - y, y - y)$$

vemos que basta demonstrar que

$$d_P(0, x - y) \leq d_P(0, x - z) + d_P(0, z - y),$$

e como $x - y = (x - z) + (z - y)$, basta mostrar que

$$\omega_P(u + v) \leq \omega_P(u) + \omega_P(v)$$

para quaisquer $u, v \in \mathbb{F}_q^n$.

Mas

$$\operatorname{supp}(x + y) \subset \operatorname{supp}(x) \cup \operatorname{supp}(y)$$

e

$$\langle I \cup J \rangle = \langle I \rangle \cup \langle J \rangle$$

donde temos que

$$\begin{aligned} \omega_P(u + v) &= \#(\langle \operatorname{supp}(u + v) \rangle) \\ &\leq \#(\langle \operatorname{supp}(u) \cup \operatorname{supp}(v) \rangle) \\ &= \#(\langle \operatorname{supp}(u) \rangle \cup \langle \operatorname{supp}(v) \rangle) \\ &\leq \omega_P(u) + \omega_P(v). \end{aligned}$$

□

Observação 3.3.15. *Note que as métricas ponderadas abrangem a métrica de Hamming, pois quando P é uma ordem anti-cadeia, isto é, cada elemento é comparável apenas consigo mesmo, temos que $d_P = d_H$.*

Exemplo 3.3.16. *Considerando as métricas cadeia (P_1), anti-cadeia (P_2), coroa (P_3), a ordem $[3, 4]$-semi-fraca (P_4) e a ordem $[1, 4]$-semi-fraca (P_5) no conjunto $[4]$ vamos descrever as esferas de $\mathbb{F}_2^4$ relativas as métricas dadas. Como métricas ponderadas são invariantes por translações, basta descrever as bolas, ou esferas, centradas na origem $0 = 0000 \in \mathbb{F}_2^4$.*

Para calcularmos $S_{P_j}(0; r) = \{v \in \mathbb{F}_2^4 \ : \ d_{P_j}(v, 0) = r\}$ vamos usar a definição de P_j-peso de v onde $\omega_{P_j}(v) = r$ para $r = 1, 2, 3, 4$, quando consideramos cada uma das métricas P_j acima.

Primeiramente, para a métrica cadeia temos o poset da figura 7.

Como $d_{P_1}(v, 0) = \omega_{P_1}(v - 0) = \omega_{P_1}(v) = 1$, caso $v \neq 1000$ teríamos $\omega_{P_1}(v) > 1$ pela definição de ideal em uma ordem. Logo,

$$S_{P_1}(0; 1) = \{1000\}.$$

Figura 7 – Poset cadeia em [4].

Se $d_{P_1}(v,0) = \omega_{P_1}(v) = 2$ *então* $\langle supp(v)\rangle = \{1,2\}$ *e assim* $\langle supp(v)\rangle = \langle 2\rangle$ *ou* $\langle 1,2\rangle$ *pela definição de ideal em uma ordem. Logo,*

$$S_{P_1}(0;2) = \{0100, 1100\}.$$

Se $d_{P_1}(v,0) = \omega_{P_1}(v) = 3$ *então* $\langle supp(v)\rangle = \{1,2,3\}$ *e assim* $\langle supp(v)\rangle = \langle 3\rangle$ *ou* $\langle 1,3\rangle$ *ou* $\langle 2,3\rangle$ *ou* $\langle 1,2,3\rangle$*. Logo,*

$$S_{P_1}(0;3) = \{0010, 1010, 0110, 1110\}.$$

Se $d_{P_1}(v,0) = \omega_{P_1}(v) = 4$ *então* $\langle supp(v)\rangle = \{1,2,3,4\}$ *e assim* $\langle supp(v)\rangle = \langle 4\rangle$ *ou* $\langle 1,4\rangle$ *ou* $\langle 2,4\rangle$ *ou* $\langle 3,4\rangle$ *ou* $\langle 1,2,4\rangle$ *ou* $\langle 1,3,4\rangle$ *ou* $\langle 2,3,4\rangle$ *ou* $\langle 1,2,3,4\rangle$*. Logo,*

$$S_{P_1}(0;4) = \{0001, 1001, 0101, 0011, 1101, 1011, 0111, 1111\}.$$

Agora, para a métrica anti-cadeia temos o poset da figura 8*.*

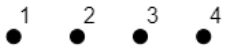

Figura 8 – Poset anti-cadeia em [4].

Se $d_{P_2}(v,0) = \omega_{P_2}(v) = 1$ *então* $\langle supp(v)\rangle = \{1\}$ *ou* $\langle supp(v)\rangle = \{2\}$ *ou* $\langle supp(v)\rangle = \{3\}$ *ou* $\langle supp(v)\rangle = \{4\}$ *e assim* $\langle supp(v)\rangle = \langle 1\rangle$ *ou* $\langle 2\rangle$ *ou* $\langle 3\rangle$ *ou* $\langle 4\rangle$*. Logo,*

$$S_{P_2}(0;1) = \{1000, 0100, 0010, 0001\}.$$

Se $d_{P_2}(v,0) = \omega_{P_2}(v) = 2$ *então* $\langle supp(v)\rangle = \{1,2\}$ *ou* $\{1,3\}$ *ou* $\{1,4\}$ *ou* $\{2,3\}$ *ou* $\{2,4\}$ *ou* $\{3,4\}$ *e assim* $\langle supp(v)\rangle = \langle 1,2\rangle$ *ou* $\langle 1,3\rangle$ *ou* $\langle 1,4\rangle$ *ou* $\langle 2,3\rangle$ *ou* $\langle 2,4\rangle$ *ou* $\langle 3,4\rangle$*. Logo,*

$$S_{P_2}(0;2) = \{1100, 1010, 1001, 0110, 0101, 0011\}.$$

Se $d_{P_2}(v,0) = \omega_{P_2}(v) = 3$ *então* $\langle supp(v)\rangle = \langle 1,2,3\rangle$ *ou* $\langle 1,2,4\rangle$ *ou* $\langle 1,3,4\rangle$ *ou* $\langle 2,3,4\rangle$*. Logo,*

$$S_{P_2}(0;3) = \{1110, 1101, 1011, 0111\}.$$

Se $d_{P_2}(v,0) = \omega_{P_2}(v) = 4$ *então* $\langle supp(v)\rangle = \langle 1,2,3,4\rangle$. *Logo,*

$$S_{P_2}(0;4) = \{1111\}.$$

Agora, para a métrica coroa temos o poset da figura 9.

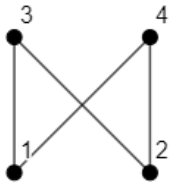

Figura 9 – Poset coroa em [4].

Se $d_{P_3}(v,0) = \omega_{P_3}(v) = 1$ *então* $\langle supp(v)\rangle = \langle 1\rangle$ *ou* $\langle 2\rangle$. *Logo,*

$$S_{P_3}(0;1) = \{1000, 0100\}.$$

Se $d_{P_3}(v,0) = \omega_{P_3}(v) = 2$ *então* $\langle supp(v)\rangle = \langle 1,2\rangle$. *Logo,*

$$S_{P_3}(0;2) = \{1100\}.$$

Se $d_{P_3}(v,0) = \omega_{P_3}(v) = 3$ *então* $\langle supp(v)\rangle = \langle 3\rangle$ *ou* $\langle 4\rangle$ *ou* $\langle 1,3\rangle$ *ou* $\langle 1,4\rangle$ *ou* $\langle 2,3\rangle$ *ou* $\langle 2,4\rangle$ *ou* $\langle 1,2,3\rangle$ *ou* $\langle 1,2,4\rangle$. *Logo,*

$$S_{P_3}(0;3) = \{0010, 0001, 1010, 1001, 0110, 0101, 1110, 1101\}.$$

Se $d_{P_3}(v,0) = \omega_{P_3}(v) = 4$ *então* $\langle supp(v)\rangle = \langle 3,4\rangle$ *ou* $\langle 1,3,4\rangle$ *ou* $\langle 2,3,4\rangle$ *ou* $\langle 1,2,3,4\rangle$. *Logo,*

$$S_{P_3}(0;4) = \{0011, 1011, 0111, 1111\}.$$

Agora, para a métrica $[3,4]$*-semi-fraca temos o poset da figura* 10.

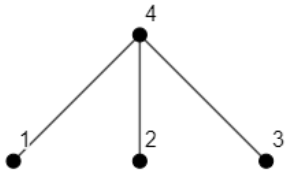

Figura 10 – Poset $[3,4]$-semi-fraco em [4].

Se $d_{P_4}(v,0) = \omega_{P_4}(v) = 1$ *então* $\langle supp(v)\rangle = \langle 1\rangle$ *ou* $\langle 2\rangle$ *ou* $\langle 3\rangle$. *Logo,*

$$S_{P_4}(0;1) = \{1000, 0100, 0010\}.$$

Se $d_{P_4}(v,0) = \omega_{P_4}(v) = 2$ *então* $\langle supp(v)\rangle = \langle 1,2\rangle$ *ou* $\langle 1,3\rangle$ *ou* $\langle 2,3\rangle$. *Logo,*

$$S_{P_4}(0;2) = \{1100, 1010, 0110\}.$$

Se $d_{P_4}(v,0) = \omega_{P_4}(v) = 3$ *então* $\langle supp(v)\rangle = \langle 1,2,3\rangle$. *Logo,*

$$S_{P_4}(0;3) = \{1110\}.$$

Se $d_{P_4}(v,0) = \omega_{P_4}(v) = 4$ *então* $\langle supp(v)\rangle = \langle 4\rangle$ *ou* $\langle 1,4\rangle$ *ou* $\langle 2,4\rangle$ *ou* $\langle 3,4\rangle$ *ou* $\langle 1,2,4\rangle$ *ou* $\langle 1,3,4\rangle$ *ou* $\langle 2,3,4\rangle$ *ou* $\langle 1,2,3,4\rangle$.

Logo,

$$S_{P_4}(0;4) = \{0001, 1001, 0101, 0011, 1101, 1011, 0111, 1111\}.$$

Por fim, para a métrica $[1,4]$*-semi-fraca temos o poset da figura* 11.

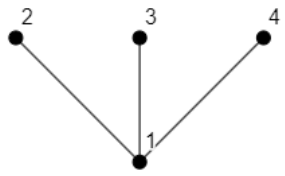

Figura 11 – Poset $[1,4]$-semi-fraco em $[4]$.

Se $d_{P_5}(v,0) = \omega_{P_5}(v) = 1$ *então* $\langle supp(v)\rangle = \langle 1\rangle$. *Logo,*

$$S_{P_5}(0;1) = \{1000\}.$$

Se $d_{P_5}(v,0) = \omega_{P_5}(v) = 2$ *então* $\langle supp(v)\rangle = \langle 2\rangle$ *ou* $\langle 3\rangle$ *ou* $\langle 4\rangle$ *ou* $\langle 1,2\rangle$ *ou* $\langle 1,3\rangle$ *ou* $\langle 1,4\rangle$. *Logo,*

$$S_{P_5}(0;2) = \{0100, 0010, 0001, 1100, 1010, 1001\}.$$

Se $d_{P_5}(v,0) = \omega_{P_5}(v) = 3$ *então* $\langle supp(v)\rangle = \langle 2,3\rangle$ *ou* $\langle 2,4\rangle$ *ou* $\langle 3,4\rangle$ *ou* $\langle 1,2,3\rangle$ *ou* $\langle 1,2,4\rangle$ *ou* $\langle 1,3,4\rangle$. *Logo,*

$$S_{P_5}(0;3) = \{0110, 0101, 0011, 1110, 1101, 1011\}.$$

Se $d_{P_5}(v,0) = \omega_{P_5}(v) = 4$ *então* $\langle supp(v)\rangle = \langle 2,3,4\rangle$ *ou* $\langle 1,2,3,4\rangle$. *Logo,*

$$S_{P_5}(0;4) = \{0111, 1111\}.$$

4 CÓDIGOS PERFEITOS E RAIO DE EMPACOTAMENTO

Neste capítulo, seguindo a abordagem de [2], vamos introduzir o conceito de código perfeito e apresentar os principais exemplos de códigos perfeitos. Em seguida, estudaremos os critérios para otimizar o raio em torno dos elementos do código de modo que consigamos contemplar todas as palavras do alfabeto e ainda assim as bolas não se intersectem, obtendo um bom empacotamento de esferas.

A grosso modo, um código $\mathcal{C} \subset \mathbb{F}_p^n$ é perfeito se, como vimos na introdução desse trabalho, não ocorre o caso de uma palavra errada cair fora de todas as bolas centradas nas palavras do código, ou seja, se existir $r \in \mathbb{N}$ tal que a união disjunta de todas as bolas de raio r centradas nos elementos de $\mathcal{C}$ é igual a $\mathbb{F}_p^n$. Formalmente, definimos:

Definição 4.0.1. *Dizemos que um código linear $\mathcal{C} \subset \mathbb{F}_p^n$ é um código perfeito se*

$$\bigcup_{u \in \mathcal{C}} B(u; r) = \mathbb{F}_p^n$$

e

$$B(u; r) \cap B(v; r) = \emptyset$$

qualquer que seja $u, v \in \mathcal{C}$ com $u \neq v$.

Neste caso, $r = R_e(\mathcal{C})$.

Note que a definição de código perfeito implica que o empacotamento em esferas centradas nas palavras do código é máximo no sentido que cobre todo o espaço.

4.1 CÓDIGOS DE HAMMING

Sejam $n = 2^r - 1$, com $r \geq 2$, e H_r a matriz de ordem $r \times (2^r - 1)$ cujas colunas são todos os vetores não nulos de $\mathbb{F}_2^r$.

Observe que H_r contém um máximo de r linhas linearmente independentes. De fato, o número de linhas e colunas linearmente independentes é sempre igual. As colunas de H_r contém uma base de $\mathbb{F}_2^r$, portanto tem ao menos r colunas linearmente independentes e não pode conter mais do que r colunas linearmente independentes pois todo subconjunto de um espaço vetorial contendo mais elementos do que a dimensão é linearmente dependente.

Definição 4.1.1. *O código linear*

$$\mathcal{H}_r = \{x \in \mathbb{F}_2^{2^r - 1} \ : \ H_r \cdot x^t = 0\}$$

que tem H_r como matriz de verificação de paridade, é chamado de Código de Hamming. Temos que $\mathcal{H}_r$ é um $[2^r - 1; 2^r - r - 1]$ código linear.

Exemplo 4.1.2. *Como um exemplo da definição* 4.1.1*, consideremos* $r = 4$*. Então,* $n = 2^r - 1 = 15$ *e a matriz* H_4*, de ordem* 4×15*, é:*

$$H_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

O código $\mathcal{H}_4$ *é o* $[15, 11]$ *código linear cuja matriz verificação de paridade é* H_4*. Observe que todo par de colunas de* H_4 *é linearmente independente, mas existem* 3 *colunas linearmente dependentes. De fato, a soma das colunas* 1 *e* 14 *é a coluna* 15*.*

Antes de enunciarmos e demonstrarmos o teorema que determina a distância mínima de um código de Hamming, vejamos um lema auxiliar, válido para qualquer código linear, que será fundamental na demonstração deste teorema:

Lema 4.1.3. *Seja* H *a matriz teste de paridade de um código linear* $\mathcal{C}$*. Temos que o peso de* $\mathcal{C}$ *é maior do que ou igual a* s *se, e somente se, quaisquer* $s - 1$ *colunas de* H *são linearmente independentes.*

Demonstração. Primeiramente, suponhamos que cada conjunto de $s - 1$ colunas de H é linearmente independente.

Seja $c = (c_1, c_2, \ldots, c_n)$ uma palavra não nula de $\mathcal{C}$, e sejam $h^1, h^2, \ldots, h^n$ as colunas de H. Como $Hc^t = 0$, temos que

$$0 = H \cdot c^t = \sum_{i=1}^{n} c_i h^i. \tag{4.1}$$

Uma vez que $\omega(c)$ é o número de componentes não nulas de c, segue que se $\omega(c) \leq s - 1$, teríamos por 4.1 uma combinação resultando no vetor nulo de um número l de colunas de H, com $1 \leq l \leq s - 1$, o que é um contradição.

Logo, $\omega(c) \geq s$ e, portanto, $\omega(\mathcal{C}) \geq s$.

Reciprocamente, suponhamos que $\omega(\mathcal{C}) \geq s$. Suponhamos também, por absurdo, que H tenha $s - 1$ colunas linearmente dependentes, digamos $h^{i_1}, h^{i_2}, \ldots, h^{i_{s-1}}$.

Daí, existiriam $c_{i_1}, c_{i_2}, \ldots, c_{i_{s-1}}$, não todos nulos, no corpo tais que

$$c_{i_1} h^{i_1} + c_{i_2} h^{i_2} + \ldots + c_{i_{s-1}} h^{i_{s-1}}.$$

Portanto, $c = (0, \ldots, c_{i_1}, 0, \ldots, c_{i_2}, 0, \ldots, c_{i_{s-1}}, 0, \ldots, 0) \in \mathcal{C}$ e, assim,

$$\omega(c) \leq s - 1 < s,$$

o que seria uma contradição. □

Corolário 4.1.4. *Seja H a matriz teste de paridade de um código linear $\mathcal{C}$. Temos que o peso de $\mathcal{C}$ é igual a s se, e somente se, quaisquer $s-1$ colunas de H são linearmente independentes e existem s colunas de H são linearmente dependentes.*

Demonstração. Com efeito, suponhamos que $\omega(\mathcal{C}) = s$, logo, todo conjunto de $s-1$ colunas de H é linearmente independente. Por outro lado, existem s colunas de H linearmente dependentes pois, caso contrário, pelo Lema 4.1.3, teríamos $\omega(\mathcal{C}) \geq s+1$.

Reciprocamente, suponhamos que todo conjunto de $s-1$ vetores colunas de H é linearmente independente e existem s colunas linearmente dependentes.

Logo, pelo Lema 4.1.3, temos que $\omega(\mathcal{C}) \geq s$. Mas $\omega(\mathcal{C})$ não pode ser maior que s pois, neste caso, novamente o Lema 4.1.3 nos diria que todo conjunto com s colunas de H é linearmente independente, o que é uma contradição. □

Teorema 4.1.5. *Um código de Hamming $\mathcal{H}_r$ tem distância mínima* 3.

Demonstração. Seja H_r a matriz de verificação de paridade de $\mathcal{H}_r$.

Pelo Corolário 4.1.4 para mostrarmos que a distância mínima d do código é igual a 3 basta mostrarmos que quaisquer pares de colunas de H_r são linearmente independentes e existem 3 colunas de H_r que são linearmente dependentes.

De fato, quaisquer 2 colunas são sempre linearmente independentes pois as colunas de H_r são os $2^r - 1$ vetores não nulos de $\mathbb{F}_2^r$ e, desta maneira suas colunas são duas a duas linearmente independentes. Agora, podemos afirmar que existem 3 colunas de H_r que são linearmente dependentes pois o conjunto $\{h_1, h_2, h_3\}$, onde $h_1, h_2 \in \mathbb{F}_2^r$ e $h_3 = h_1 + h_2 \in \mathbb{F}_2^r$, é linearmente dependente e assim, segue o resultado. □

Exemplo 4.1.6. *Temos que $\mathcal{H}_2 = \{000, 111\}$ é um $[3; 1; 3]$ código de Hamming com matriz de verificação de paridade*

$$H_2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}.$$

Este é exatamente o mesmo código $\mathcal{C}$ do Exemplo 3.1.3*, onde vimos que*

$$B_H(000; 1) \cap B_H(111; 1) = \emptyset$$

e

$$B_H(000; 1) \cup B_H(111; 1) = \mathbb{F}_2^3,$$

de modo que o código é perfeito e seja qual for a mensagem $x \in \mathbb{F}_2^3$ recebida, sempre saberemos o que fazer com ela, isto é, trocamos x pelo centro da bola que x pertence. Observe que $\mathcal{H}_2$ tem a capacidade de corrigir $R_e(\mathcal{H}_2) = \left\lfloor \frac{3-1}{2} \right\rfloor = 1$ erro.

O lema a seguir caracteriza completamente o número de elementos em cada bola dos códigos de Hamming na métrica de Hamming.

Lema 4.1.7. *Sejam $x \in \mathbb{F}_p^n$ e $n \in \mathbb{N}$. Então,*

$$\#(B_H(x;r)) = \sum_{i=0}^{r} \binom{n}{i}(p-1)^i.$$

Demonstração. Inicialmente, note que $\#(B_H(x;r)) = \#(B_H(0;r))$, pois a translação do centro de uma bola da origem para um ponto qualquer estabelece uma bijeção entre $\#(B_H(x;r))$ e $\#(B_H(0;r))$.

Se $y \in B_H(0;r)$ então y possui exatamente i coordenadas não nulas para algum $i \leq r$. Desta maneira, temos $\binom{n}{i}$ possíveis escolhas para as coordenadas de y. E ainda, como cada coordenada não nula de y pode assumir os valores $1, 2, \ldots, p-1$, temos um total de $\binom{n}{i}(p-1)^i$ vetores que distam i de um ponto.

Portanto,

$$\#(B_H(x;r)) = \sum_{i=0}^{r} \binom{n}{i}(p-1)^i$$

como queríamos demonstrar. □

Teorema 4.1.8. *O código de Hamming $\mathcal{H}_r$ é um código perfeito considerando a métrica de Hamming.*

Demonstração. Primeiramente, pelo Teorema 4.1.5, sabemos que a distância mínima de $\mathcal{H}_r$ é 3. Portanto, o raio de empacotamento de $\mathcal{H}_r$ é igual a $\left\lfloor \frac{3-1}{2} \right\rfloor = 1$.

Afirmamos que as bolas de raio 1 centradas nos elementos de $\mathcal{H}_r$ cobrem $\mathbb{F}_2^{2^r-1}$. De fato, pelo Lema 4.1.7 como

$$\#(B_H(u;1)) = \binom{2^r-1}{0} + \binom{2^r-1}{1} = 1 + (2^r - 1) = 2^r$$

para todo $u \in \mathcal{H}_r$ e

$$2^r \cdot \#\mathcal{C} = 2^r \cdot (2^{2^r-r-1}) = 2^{2^r-1} = \#(\mathbb{F}_2^{2^r-1})$$

concluímos que as bolas $B_H(u;1)$ cobrem $\mathbb{F}_2^{2^r-1}$. Portanto $\mathcal{H}_r$ é um código perfeito. □

4.2 CÓDIGOS DE HAMMING ESTENDIDOS

Dado o código de Hamming $\mathcal{H}_r$ com matriz de verificação de paridade

$$H_r = \begin{bmatrix} h_{11} & h_{12} & h_{13} & \ldots & h_{1(2^r-1)} \\ h_{21} & h_{22} & h_{23} & \ldots & h_{2(2^r-1)} \\ h_{31} & h_{32} & h_{33} & \ldots & h_{3(2^r-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{r1} & h_{r2} & h_{r3} & \ldots & h_{r(2^r-1)} \end{bmatrix},$$

acrescentamos a esta uma linha com todas as entradas iguais a 1 e completamos a última coluna com entradas iguais 0, obtendo

$$\hat{H}_r = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 & 1 \\ h_{11} & h_{12} & h_{13} & \dots & h_{1(2^r-1)} & 0 \\ h_{21} & h_{22} & h_{23} & \dots & h_{2(2^r-1)} & 0 \\ h_{31} & h_{32} & h_{33} & \dots & h_{3(2^r-1)} & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ h_{r1} & h_{r2} & h_{r3} & \dots & h_{r(2^r-1)} & 0 \end{bmatrix}.$$

Definição 4.2.1. *O código de Hamming estendido $\hat{\mathcal{H}}_r$ é definido como o código linear que tem $\hat{H}_r$ como matriz de paridade, ou seja*

$$\hat{\mathcal{H}}_r = \{x \in \mathbb{F}_2^{2^r} \ : \ \hat{H}_r \cdot x^t = 0\}.$$

Exemplo 4.2.2. *O código $\hat{\mathcal{H}}_4$, que é o código de Hamming estendido do código $\mathcal{H}_4$, por definição tem matriz verificação de paridade $\hat{H}_4$ da forma*

$$\hat{H}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}.$$

Novamente, destacamos que todo trio de colunas é linearmente independente e existem 4 *colunas linearmente dependentes. De fato, a soma das colunas* 1, 2 *e* 3 *é a coluna* 11*, por exemplo.*

Se lembrarmos que H_r é uma matriz com $2^r - 1$ colunas e um máximo de r linhas linearmente independentes, é imediato verificar que $\hat{H}_r$ tem 2^r colunas e $r + 1$ linhas linearmente independentes, de modo que $\mathcal{H}_r$ é um $[2^r; 2^r - r - 1]$ código linear.

Teorema 4.2.3. *Um código de Hamming estendido tem distância mínima* 4*.*

Demonstração. Seja $\hat{H}_r$ a matriz de verificação de paridade de um código de Hamming estendido $\hat{\mathcal{H}}_r$.

Pelo Corolário 4.1.4 para mostrarmos que a distância mínima, d, do código é igual a 4 basta mostrarmos que quaisquer ternos de colunas de $\hat{H}_r$ são linearmente independentes e existem 4 colunas de $\hat{H}_r$ que são linearmente dependentes.

Primeiramente, note que as colunas de $\hat{H}_r$ são os 2^r vetores do conjunto

$$X = \{(1, x_2, x_3, \dots, x_{r+1}) \ : \ x_i \in \mathbb{F}_2\} \subset \mathbb{F}_2^{r+1}.$$

Desta forma, suponhamos que exista um conjunto $\{h_1, h_2, h_3\} \subset X$, onde os elementos são distintos, linearmente dependentes. Então, algum dos elementos, digamos h_3, é uma combinação linear dos outros dois, isto é, $h_3 = a \cdot h_1 + b \cdot h_2$, $a, b \in \mathbb{F}_2$. Temos então 3 possibilidade:

1. $a = b = 0$: Então $h_3 = 0 \cdot h_1 + 0 \cdot h_2 = 0$ que é um absurdo pois $0 \notin X$;

2. $a = 0$ ou $b = 0$: Digamos que $a = 0$. Então $h_3 = 0 \cdot h_1 + 1 \cdot h_2 = h_2$ que é um absurdo. Chegamos à mesma contradição quando $b = 0$.

3. $a = b = 1$: Sejam $h_1 = (1, x_{21}, x_{31}, \ldots, x_{(r+1)1}), h_2 = (1, x_{22}, x_{32}, \ldots, x_{(r+1)2})$. Daí, segue que

$$\begin{aligned} h_3 &= 1 \cdot h_1 + 1 \cdot h_2 \\ &= h_1 + h_2 \\ &= (1+1, x_{21} + x_{22}, x_{31} + x_{32}, \ldots, x_{(r+1)1} + x_{(r+1)2}) \\ &= (0, x_{21} + x_{22}, x_{31} + x_{32}, \ldots, x_{(r+1)1} + x_{(r+1)2}) \,, \end{aligned}$$

   o que é uma contradição, pois nenhum elemento de X tem a primeira coordenada igual a zero.

Dessa forma, todo conjunto contendo três elementos distintos de X é linearmente independente.

Agora, observe que considerando os elementos de X apenas a partir da segunda coordenada, temos todos os 2^r vetores de $\mathbb{F}_2^r$. Como no Teorema 4.1.5, consideremos o conjunto $\{h_1, h_2, h_3\} \subset X$, onde $h_3 = h_1 + h_2$ a partir da segunda coordenada. Como $h_1 + h_2$ tem a primeira coordenada nula, o conjunto $\{1, h_1, h_2, h_3\} \subset X$, onde $1 = (1, 0, 0, \ldots, 0)$, é um conjunto linearmente dependente pois $h_3 = 1 + h_1 + h_2$ e assim, segue o resultado. □

Conforme vimos no Lema 4.1.7, em $\mathbb{F}_q^n$, considerando a métrica de Hamming, temos

$$\#(B_H(0;1)) = n(q-1) + 1 \,.$$

Assim, considerando $\hat{\mathcal{H}}_r \subset \mathbb{F}_2^{2^r}$ temos

$$\#(B_H(0;1)) = 2^r + 1 \,.$$

Observe ainda que $\hat{\mathcal{H}}_r$ tem dimensão $2^r - r - 1$, de modo que possui $2^{2^r - r - 1}$ elementos. As bolas de raio centradas nestes pontos são disjuntas, mas a união destas têm $(2^r + 1) \cdot 2^{2^r - r - 1}$ pontos e, como $2^r + 1$ é ímpar, temos

$$\#(B_H(u;1)) \cdot \#(\hat{\mathcal{H}}_r) = (2^r + 1) \cdot 2^{2^r - r - 1} < 2^{2^r} = \#(\mathbb{F}_2^{2^r}),$$

ou seja, estas bolas não cobrem o espaço $\mathbb{F}_2^{2^r}$. Portanto, considerando a métrica de Hamming, $\hat{\mathcal{H}}_r$ não é perfeito.

Exemplo 4.2.4. *Consideremos a matriz de verificação de paridade do código de Hamming estendido*

$$\hat{H}_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

donde $\hat{\mathcal{H}}_2 = \{0000, 1111\}$. *Neste caso, temos*

$$B_H(0000; 1) = \{0000, 1000, 0100, 0010, 0001\}$$

e

$$B_H(1111; 1) = \{1111, 0111, 1011, 1101, 1110\}$$

de modo que

$$\mathbb{F}_2^4 \backslash (B_H(0000; 1) \cup B_H(1111; 1)) = \{1100, 1010, 1001, 0110, 0101, 0011\} \tag{4.2}$$

e assim o código não é perfeito. Mais ainda, caso a palavra recebida seja qualquer uma das palavras em (4.2)*, esta não poderá ser decodificada.*

Como o código $\hat{\mathcal{H}}_r$ não é perfeito considerando a métrica de Hamming, nosso trabalho agora se resume a exibirmos uma métrica d_P ponderada por uma ordem parcial P que torna $\hat{\mathcal{H}}_r$ um código perfeito.

Seja $[2^r] = \{1, 2, \ldots, 2^r\}$ munido com a ordem $[1, 2^r]$-semi-fraca. Como vimos anteriormente, nesta ordem as únicas relações existentes são $i \leq i$ e $1 \leq i$ para todo $i = 1, 2, \ldots, 2^r$.

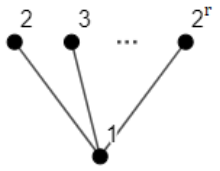

Figura 12 – Poset $[1, 2^r]$-semi-fraco em $[2^r]$.

Primeiramente vamos determinar a cardinalidade das P-bolas em $\mathbb{F}_2^{2^r}$, isto é, $\#(B_P(0; r))$.

Lema 4.2.5. *Dado um código de Hamming estendido* $\hat{\mathcal{H}}_r$*, então, com a ordem* $[1, 2^r]-$*semi-fraca em* $[2^r]$*, temos*

$$\#(B_P(0; r)) = 2 + \sum_{i=2}^{r} 2 \cdot \binom{2^r - 1}{i - 1}.$$

Demonstração. Seja $x \in \#(B_P(0;r))$.

Então, $\omega_P(x) = i$ com $i \leq r$. Se $i = 0$ ou $i = 1$, temos exatamente um vetor em $\mathbb{F}_2^{2^r}$ com peso 0 e um com peso 1, a saber

$$(0,0,\ldots,0) \text{ e } (1,0,\ldots,0)\,,$$

respectivamente. Se $i > 1$ então temos em $[2^r]$ um total de $\binom{2^r-1}{i-1}$ ideais com cardinalidade i, determinados a partir da escolha de $i-1$ elementos diferentes de 1, que são os elementos maximais.

Em $\mathbb{F}_2^{2^r}$, cada coordenada associada a um dos $i-1$ elementos maximais escolhidos em $[2^r]$ deve ser igual a 1, enquanto as associadas aos outros elementos maximais devem ser nulas. A primeira coordenada, cujo suporte está contido no ideal gerado por qualquer elemento de $[2^r]$, pode assumir qualquer um dos dois valores possíveis, 0 ou 1. Assim, para $i > 1$, temos

$$2 \cdot \#(S_P(0;i)) = 2 \cdot \binom{2^r-1}{i-1}$$

possibilidades para x. Portanto

$$\#(B_P(0;r)) = 2 + \sum_{i=2}^{r} \#(S_P(0;i)) = 2 + \sum_{i=2}^{r} 2 \cdot \binom{2^r-1}{i-1}.$$

□

Podemos então utilizar o lema 4.2.5 para provar que $\hat{\mathcal{H}}_r$ é perfeito considerando-se tal métrica ponderada.

Teorema 4.2.6. *O código de Hamming estendido $\hat{\mathcal{H}}_r$ é um código perfeito considerando a P-métrica ponderada pela ordem $[1, 2^r]$-semi-fraca e seu raio de empacotamento, $R_e(\hat{\mathcal{H}}_r)$ é igual a 2.*

Demonstração. Primeiramente, vamos mostrar que $B_P(0;2) \cap B_P(v;2) = \emptyset$ para qualquer $v \in \hat{\mathcal{H}}_r$. Observe que, mostrando isso, como uma métrica é invariante por translações, temos $B_P(u;2) \cap B_P(v;2) = \emptyset$ para quaisquer $u, v \in \hat{\mathcal{H}}_r$ com $u \neq v$.

Seja $v \in \hat{\mathcal{H}}_r$ e suponhamos que exista $x \in \mathbb{F}_2^{2^r}$ tal que

$$x \in B_P(0;2) \cap B_P(v;2).$$

Já sabemos que o único elemento de peso 1 é $100\ldots0$ enquanto os elementos de peso 2 são aqueles que têm exatamente uma coordenada não nula a partir da segunda posição.

Como $\omega_P(v) \geq 4$ então v possui pelo menos três posições entre $2, 3, \ldots, 2^r$ iguais a 1. Como x tem no máximo uma destas posições não nulas, temos que a diferença $v - x$

tem no mínimo duas dentre estas posições não nulas, de modo que

$$d_P(x, v) = \omega_P(v - x) \geq 3\,,$$

o que contradiz a hipótese de termos $x \in B_P(0; 2)$.

Logo, $B_P(0; 2) \cap B_P(v; 2) = \emptyset$ para todo $v \in \hat{\mathcal{H}}_r$, $v \neq 0$.

Ainda, pelo lema 4.2.5, o número de elementos de cada bola de raio 2 é

$$\#(B_P(v; r)) = 2 + 2 \cdot \binom{2^r - 1}{1} = 2 + 2 \cdot (2^r - 1) = 2^{r+1}$$

Como $\hat{\mathcal{H}}_r$ tem $2^{2^r - r - 1}$ elementos, concluímos então que

$$\#(B_P(v; 2)) \cdot \#(\hat{\mathcal{H}}_r) = 2^{r+1} \cdot 2^{2^r - r - 1} = 2^{2^r} = \#(\mathbb{F}_2^{2^r}),$$

e assim, $\hat{\mathcal{H}}_r$ é um P-código perfeito com raio de empacotamento igual a 2. □

4.3 CÓDIGOS SOBRE ORDENS TOTAIS

Vimos na Proposição 3.1.4 que, considerando a métrica de Hamming, temos que para qualquer código linear $\mathcal{C}$ o raio de empacotamento é determinado pela distância mínima do código

$$R_e(\mathcal{C}) = \left\lfloor \frac{d_H - 1}{2} \right\rfloor .$$

Em um espaço métrico (X, d), usando apenas a desigualdade triangular, e independentemente de qualquer estrutura adicional, dados $x, y \in X$ e $r = d(x, y)$, temos

$$B(x; s) \cap B(y; s) = \emptyset\,, \text{ se } s < \frac{r}{2}$$

e

$$x, y \in B(x; s) \cap B(y; s)\,, \text{ se } s \geq r\,.$$

Portanto, em particular, dado um código $\mathcal{C} \in \mathbb{F}_q^n$, independentemente da métrica adotada, temos

$$\left\lfloor \frac{d_P - 1}{2} \right\rfloor \leq R_e(\mathcal{C}) \leq d_P - 1. \tag{4.3}$$

A *priori* não podemos afirmar nada sobre a interseção destas bolas se $\frac{r}{2} \leq s < r$, mas sabemos que a primeira das desigualdades em (4.3), não é necessariamente justa pois, como mostramos na seção anterior, se considerarmos P a ordem $[1, 2^r]$-semi-fraca, temos que o código de Hamming estendido tem distância mínima $d_P(\hat{\mathcal{H}}_r) = 4$ e raio de empacotamento $R_e(\hat{\mathcal{H}}_r) = 2$, e assim

$$\left\lfloor \frac{d_P - 1}{2} \right\rfloor < R_e(\hat{\mathcal{H}}_r).$$

Vamos agora, através de um exemplo, mostrar que a segunda das desigualdades em (4.3) não é estrita, ou seja, vamos fornecer um código $\mathcal{C}$ e uma ordem P tais que $R_e(\mathcal{C}) = d_P(\mathcal{C}) - 1$.

Exemplo 4.3.1. *Seja P a ordem total no conjunto $[3]$.*

Afirmamos que o raio de empacotamento do código $\mathcal{C} = \{000, 101\}$ é 2, que coincide com $d_P(\mathcal{C}) - 1$ já que $\omega_P(101) = 3 = d_P(\mathcal{C})$.

Primeiramente vamos mostrar que as bolas de raio 2 centradas nos elementos de $\mathcal{C}$ são disjuntas.

$$\begin{aligned} B_P(000; 2) &= \{x \in \mathbb{F}_2^3 \ ; \ d_P(000, x) \leq 2\} \\ &= \{x \in \mathbb{F}_2^3 \ ; \ \omega_P(x) \leq 2\} \\ &= \{000, 100, 010, 110\} \end{aligned}$$

e

$$\begin{aligned} B_P(101; 2) &= \{x \in \mathbb{F}_2^3 \ ; \ d_P(101, x) \leq 2\} \\ &= \{x \in \mathbb{F}_2^3 \ ; \ \omega_P(101 - x) \leq 2\} \\ &= \{101, 001, 011, 111\}. \end{aligned}$$

O mesmo não ocorre se aumentarmos o tamanho do raio das bolas, pois

$$\begin{aligned} B_P(000; 3) &= \{x \in \mathbb{F}_2^3 \ ; \ d_P(000, x) \leq 3\} \\ &= \{x \in \mathbb{F}_2^3 \ ; \ \omega_P(x) \leq 3\} \\ &= \{000, 100, 010, 110, 001, 101, 011, 111\} \end{aligned}$$

e

$$\begin{aligned} B_P(101; 3) &= \{x \in \mathbb{F}_2^3 \ ; \ d_P(101, x) \leq 3\} \\ &= \{x \in \mathbb{F}_2^3 \ ; \ \omega_P(101 - x) \leq 3\} \\ &= \{101, 001, 011, 111, 000, 100, 010, 110\}, \end{aligned}$$

e desta forma $B_P(000; 3) \cap B_P(101; 3) = \mathbb{F}_2^3$.

Logo, $R_e(\mathcal{C}) = 2$.

Podemos generalizar o exemplo acima para um conjunto $[n]$ com n natural, mas antes façamos um lema auxiliar:

Lema 4.3.2. *Seja o conjunto $[n] = \{1, 2, \ldots, n\}$ munido com a P-ordem total*

$$1 \leq 2 \leq \ldots \leq n \, .$$

Se $u \in B_P(v,r)$ *e* $\omega_P(v) > r$*, então*

$$\langle supp(u)\rangle = \langle supp(v)\rangle,$$

ou seja,

$$\max\{supp(u)\} = \max\{supp(v)\}.$$

Demonstração. Seja $u \in B_P(v;r)$.

Se $\langle \mathrm{supp}(u)\rangle \subset \langle \mathrm{supp}(v)\rangle$, então $\langle \mathrm{supp}(v)\rangle = \langle \mathrm{supp}(v-u)\rangle$.

Desta maneira, temos $d_P(u,v) = \omega_P(v) > r$, o que é um absurdo.

Suponha agora que $\langle \mathrm{supp}(u)\rangle \supset \langle \mathrm{supp}(v)\rangle$. Então $\langle \mathrm{supp}(u)\rangle = \langle \mathrm{supp}(v-u)\rangle$ donde segue que $d_P(u,v) = \omega_P(u)$.

Como $\langle \mathrm{supp}(u)\rangle \supset \langle \mathrm{supp}(v)\rangle$ e por hipótese $\omega_P(v) > r$, temos $\omega_P(u) > r$. Logo, $d_P(u,v) = \omega_P(u) > r$ o que contraria o fato de u pertencer a $B_P(v;r)$.

Como as únicas possibilidades são $\langle \mathrm{supp}(u)\rangle \subset \langle \mathrm{supp}(v)\rangle$, $\langle \mathrm{supp}(u)\rangle \supset \langle \mathrm{supp}(v)\rangle$ ou $\langle \mathrm{supp}(u)\rangle = \langle \mathrm{supp}(v)\rangle$, concluímos que $\langle \mathrm{supp}(u)\rangle = \langle \mathrm{supp}(v)\rangle$. □

Agora, utilizando o lema 4.3.2 podemos mostrar que nas ordens totais o raio de empacotamento é sempre o máximo possível.

Teorema 4.3.3. *Seja o conjunto* $[n] = \{1, 2, \ldots, n\}$ *totalmente ordenado. Se* $\mathcal{C} \subset \mathbb{F}_q^n$ *é um código com distância mínima* $d_P(\mathcal{C})$*, então* $\mathcal{C}$ *tem a capacidade de corrigir* $d_P - 1$ *erros.*

Demonstração. Queremos mostrar que

$$B_P(0; d_P(\mathcal{C}) - 1) \cap B_P(u; d_P(\mathcal{C}) - 1) = \emptyset$$

para qualquer $u \in \mathcal{C}$ não nulo.

Se $x \in B_P(0; d_P(\mathcal{C}) - 1) \cap B_P(u; d_P(\mathcal{C}) - 1)$, como $\omega_P(u) \geq d_P(\mathcal{C})$, segue do Lema 4.3.2 que

$$\langle \mathrm{supp}(u)\rangle = \langle \mathrm{supp}(x)\rangle$$

pois $x \in B_P(u; d_P(\mathcal{C}) - 1)$.

Desta maneira, temos que $\omega_P(x) = \omega_P(u) \geq d_P(\mathcal{C})$, o que é um absurdo pois x também pertence a $B_P(0; d_P(\mathcal{C}) - 1)$, isto é, $\omega_P(x) \leq d_P(\mathcal{C}) - 1$.

Portanto, $B_P(0; d_P(\mathcal{C}) - 1) \cap B_P(u; d_P(\mathcal{C}) - 1) = \emptyset$, como queríamos demonstrar. □

5 CONCLUSÃO E PERSPECTIVAS

Como vimos no decorrer desse trabalho, os códigos perfeitos são mais eficientes no sentido de que os erros sempre produzem palavras que não se distanciam demais das palavras do código, possibilitando a identificação e correção mais eficientes de erros. Salientamos que, mesmo fora do escopo desse trabalho, essa identificação e correção de erros pode ter um custo computacional alto, porém propiciam mais confidencialidade, integridade e autenticidade da mensagem, que são alguns dos pilares que sustentam a segurança da Tecnologia da Informação.

Concluímos, então, esse texto notando que os Teoremas 4.1.8, 4.2.6 e 4.3.3 nos fornecem uma vasta gama de códigos perfeitos utilizando algumas das métricas que vimos no capítulo 3. Tais códigos, por sua vez, nos fornecem empacotamentos ótimos do espaço centrados nas palavras dos códigos. Notamos, no entanto, que não estudamos nenhum código perfeito sob a métrica de Lee ou a P-métrica Coroa. Podemos nos perguntar se existem tais códigos. Ou, ainda mais, se podemos estender o que foi feito para os códigos de Hamming estendidos, isto é: será que dado qualquer código existe uma métrica de forma que o código seja perfeito? Essas perguntas, entre outras, mostram que ainda há campo para continuar o estudo aqui proposto.

REFERÊNCIAS

[1] D'OLIVEIRA, R. G. L. *Raio de Empacotamento de Códigos Poset*. Dissertação de Mestrado. UNICAMP (2012). Disponível em: `http://repositorio.unicamp.br/bitstream/REPOSIP/305914/1/LucasD%27Oliveira_RafaelGregorio_M.pdf`. Acesso em: 29 de março de 2019.

[2] FIRER, M. *Códigos Corretores de Erros - Notas de Aula*. Disponível em: `https://www.ime.unicamp.br/~mfirer/3NotasFoz2006.pdf`. Acesso em: 29 de março de 2019.

[3] GARCIA, A. L. P.; LEQUAIN, Y. A. E. *Elementos de Álgebra*. 6. ed. Rio de Janeiro: IMPA, 2012.

[4] GONÇALVES, A. *Introdução à Álgebra*. 5. ed. Rio de Janeiro: IMPA, 2011.

[5] HEFEZ, A.; VILLELA, M. L. T. *Códigos Corretores de Erros*. 2. ed. Rio de Janeiro: IMPA, 2008.

[6] HUFFMAN, W. C.; PLESS, V. *Fundamentals of Error-Correcting Codes*. Cambridge: Cambridge University Press, 2003.

[7] MILES, C. P. *Breve Introdução à Teoria dos Códigos Corretores de Erros*. Disponível em: `https://www.sbm.org.br/docs/coloquios/CO-1-09.pdf`. Acesso em: 29 de março de 2019.

[8] MOURA, A. *Dualidade em Espaços Poset*. Tese de Doutorado, UNICAMP (2010). Disponível em: `http://repositorio.unicamp.br/bitstream/REPOSIP/305916/1/Moura_AllandeOliveira1_D.pdf`. Acesso em: 23 de maio de 2019.