Larissa Stefani

Construção e Decodificação de Códigos Algébrico - Geométricos



UNIVERSIDADE ESTADUAL PAULISTA
Campus de São Jóse do Rio Preto
Instituto de Biociências, Letras e Ciências Exatas

Construção e Decodificação de Códigos Algébrico-Geométricos

Larissa Stefani

Orientador

Prof. Dr. José Carmelo Interlando

Inst. de Biociências, Letras e Ciências Exatas - UNESP
BIBLIOTECA
S. J. Rio Preto

937/00

Dissertação a ser apresentada ao Departamento de Matemática/IBILCE/UNESP como parte dos requisitos para o obtenção do título de Mestre em Ciências Matemáticas - Àrea de concentração Matemática.

São José do Rio Preto - 1999

Agradecimentos

Ao Prof. José Carmelo Interlando pela orientação.

Aos Prof. Trajano Pires da Nóbrega Neto e Prof. Bartolomeu Ferreira Uchoa Filho pela disposição em ler este trabalho.

A CAPES pelo apoio financeiro.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

RESUMO

O objetivo desse trabalho é descrever os processos de codificação e decodificação de Códigos de Bloco Lineares, com a finalidade de encontrar códigos assintoticamente bons, ou seja, capazes de superar o Limitante Assintótico de Gilbert-Varshamov. Para isso, analisamos vários tipos de códigos, com complexidade crescente como Códigos Cíclicos, BCH, RS, Alternantes e Goppa, finalizando com códigos Algébrico-Geométricos, que apontam para um novo caminho na construção de bons códigos.

ABSTRACT

The objective of this dissertation is to describe the coding and the decoding processes of Linear Block Codes, with the aim of finding asymptotically good codes, ie, codes that are better than the Gilbert-Varshamov bound. For that, we study classes of codes with increasing complexity, such as cyclic, BCH, RS, Alternant, and Goppa, finalizing with Algebraic-Geometry codes which point to a new way in the construction of good codes.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

ÍNDICE

Introdução 1

Capítulo 1: **Códigos de Bloco Lineares** 2
1.1 Introdução 2
1.2 Codificação de um Código de Bloco Linear 4
1.3 Definições 8
1.4 Detecção de Erros 11
1.5 Correção de Erros 13
1.6 Limitantes 19
1.7 Entropia e Quantidade de Informação 21
1.8 Códigos de Hamming 26
1.9 Propriedades Elementares de Corpos Finitos 27

Capítulo 2: **Códigos Cíclicos e BCH** 31
2.1 Introdução 31
2.2 Códigos Cíclicos 32
2.3 Códigos BCH 43
2.4 A Distância Mínima Verdadeira de um Código BCH 47
2.5 O Número de Dígitos de Informação de um Código BCH 53
2.6 Códigos BCH Longos são Assintoticamente Ruins 57
2.7 Decodificação dos Códigos BCH 59
2.8 O Algoritmo de Peterson-Gorenstein-Zierler 72

Capítulo 3: **Curvas Algébricas** 74
3.1 Variedades Projetivas e Afins 74
3.2 Anel Local em um Ponto 80
3.3 Divisores e o Teorema de Riemann-Roch 82

Capítulo 4: **Códigos RS Generalizados, Alternantes e de Goppa** 86
4.1 Introdução 86
4.2 Códigos Reed-Solomon Estendidos 86
4.3 Introdução aos Códigos RS Geométricos 91
4.4 Códigos RS Generalizados 92
4.5 Códigos Alternantes 94
4.6 Códigos de Goppa 96

unesp

Capítulo 5: **Códigos Algébrico-Geométricos** 98
Introdução 98
Códigos Sobre Curvas Algébricas 99
Decodificação 104
O Comportamento Assintótico dos Códigos Algébrico-Geométricos 117

Conclusão 120
Bibliografia 122

INTRODUÇÃO

O objetivo deste trabalho é apresentar e estudar algumas famílias de códigos corretores de erros. Trataremos mais especificamente dos processos de codificação e decodificação de códigos lineares, com o objetivo de encontrar códigos assintoticamente bons e até capazes de superar o limitante de Gilbert-Varshamov.

Para isso, iniciaremos no Capítulo 1 definindo códigos lineares, seus principais parâmetros, suas propriedades gerais e seus limitantes. No Capítulo 2, passaremos a descrever códigos cíclicos e BCH, códigos esses que apesar de assintoticamente ruins, apresentam um algoritmo de decodificação muito eficiente a saber, o algoritmo de Berlekamp-Massey.

No Capítulo 3 apresentamos resumidamente resultados fundamentais de curvas algébricas, os quais utilizaremos na descrição dos códigos algébrico-geométricos apresentados no Capítulo 5.

Na busca de bons códigos, apresentaremos no Capítulo 4 os códigos RS, com diferentes representações. Os códigos RS constituem uma importante família não somente do ponto de vista prático (são usados em Compact Discs), mas também por serem o ponto de partida para o estudo dos códigos alternantes e sua subclasse mais importante, dada pelos códigos de Goppa.

Finalizamos este trabalho com o Capítulo 5 onde apresentamos uma breve discussão sobre o comportamento assintótico dos códigos geométricos, bem como sua construção e decodificação. Os códigos algébrico-geométricos apontam para um novo caminho na construção de bons códigos.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

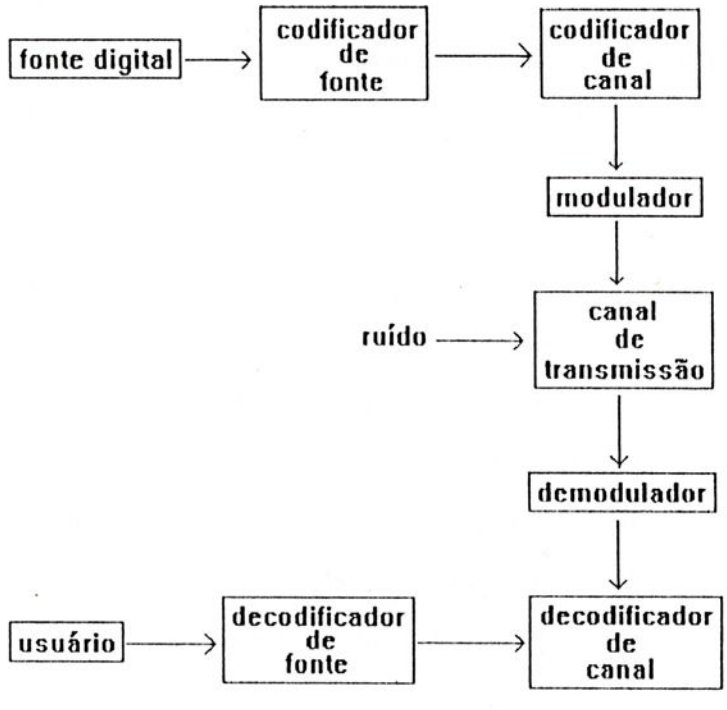
Capítulo 1

Códigos de Bloco Lineares

1.1 Introdução

A teoria dos códigos é um ramo da Matemática com inúmeras aplicações, sendo utilizada sempre que houver necessidade de se transmitir ou armazenar dados.

A transmissão de dados ocorre através de um sistema de comunicações o qual pode ser representado pelo seguinte diagrama:



Da fonte digital são enviados os dados na forma de símbolos $a_0, a_1, a_2 \ldots$, onde a_i pertence a um determinado alfabeto (conjunto finito). Em geral, a ocorrência dos símbolos $a_0, a_1, a_2, \ldots$ obedece a alguma distribuição de probabilidade.

O codificador de fonte enviará ao codificador de canal os símbolos da fonte já em forma de mensagem, ou seja, sem redundância. O estudo da codificação de fonte, isto é, representação eficiente dos símbolos que saem da fonte, foge ao escopo do presente trabalho. O texto [1] é uma boa referência para este assunto.

Nos códigos de bloco, as mensagens a serem transmitidas chegam ao codificador de canal como k-uplas, $u = (u_0, \ldots, u_{k-1})$ onde $u_i \in GF(q)$, o corpo de Galois com q elementos.

A fim de obter alguma proteção, o codificador acrescenta aos dígitos da mensagem u um número fixo de dígitos extra, os quais chamaremos de dígitos de verificação de paridade. Então à mensagem original $u = (u_0, \ldots, u_{k-1})$ corresponderá a n-upla $v = (v_0, v_1, \ldots, v_{n-1})$, onde $k < n$, a qual chamaremos de palavra-código.

Para cada mensagem u, haverá uma única palavra-código correspondente v. Portanto o conjunto C de todas as palavras-código tem cardinalidade q^k. Se C for um subespaço vetorial de $GF(q)^n$ (o conjunto de todas as n-uplas sobre $GF(q)$), então C é denominado código linear. É altamente desejável que um código tenha a propriedade de ser linear, pois neste caso os processos de codificação e decodificação são muito menos complexos que para os códigos não-lineares.

As palavras-código chegarão então ao modulador que as converterá em forma de ondas apropriadas para o canal que estiver sendo utilizado.

Ao demodulador cabe o processo inverso do modulador, ou seja, transformar as ondas recebidas nas n-uplas que chegarão ao decodificador de canal.

Observe que devido ao ruído no canal de transmissão, a n-upla recebida pelo decodificador de canal pode ser diferente da n-upla enviada. Assim cabe ao decodificador converter a n-upla recebida na palavra-código enviada. Caso o decodificador converta a n-upla recebida em uma palavra-código que não a enviada, dizemos que houve um erro



de decodificação. Um dos objetivos da teoria de códigos corretores de erros é minimizar a probabilidade de erro de decodificação.

Na sequência o decodificador de canal entrega a mensagem decodificada ao decodificador de fonte que, por sua vez, a converte em símbolos que são passados ao usuário.

Neste capítulo descreveremos os fundamentos de códigos de blocos lineares, incluindo os princípios de codificação e decodificação.

Na Seção 1.2 descrevemos o processo de codificação de mensagens num código de bloco linear usando a matriz geradora G e a matriz verificação de paridade H.

Na Seção 1.3 apresentamos as definições de padrão de erro, peso de Hamming de uma palavra-código, distância mínima e a construção do arranjo padrão, o qual será utilizado durante o processo de decodificação .

Na Seção 1.4 descreveremos o processo de detecção de erros.

Na Seção 1.5 descreveremos o processo de correção de erros, isto é, decodificação, utilizando o arranjo padrão e também será apresentado o teorema de Shannon sobre a existência de "bons" códigos.

Na Seção 1.6 serão apresentados alguns limitantes para os códigos lineares como o limitante de Singleton e o de Gilbert-Varshamov.

Na Seção 1.7 apresentaremos o Limitante de Gilbert-Varshamov Assintótico, o qual nos assegura a existência de "bons" códigos lineares quando seu comprimento tende a infinito.

Na Seção 1.8 a título de exemplo, descreveremos os códigos de Hamming binários, onde os processos de codificação e decodificação são bem simples.

Na Seção 1.9 estaremos apresentando uma breve revisão sobre corpos finitos cujas propriedades serão utilizadas nos capítulos posteriores.

1.2 Codificação de um Código de Bloco Linear

Definição 1.2.1 *Um código de bloco linear de comprimento n e dimensão k é um*

subespaço vetorial de dimensão k do espaço de todas as n-uplas sobre o corpo $GF(q)$. Quando $q = 2$, dizemos que o código é binário. Analogamente, temos códigos ternários, quaternários, etc. O corpo $GF(q)$ é dito o alfabeto sobre o qual o código está definido.

Seja então C um código de bloco linear (n, k). Como C é um k–subespaço vetorial, C possui uma base $g_0, g_1, \ldots, g_{k-1}$. Na forma matricial, temos

$$G = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix} = \begin{bmatrix} g_{00} & g_{01} & \cdots & g_{0,n-1} \\ g_{10} & g_{11} & \cdots & g_{1,n-1} \\ \vdots & \vdots & & \vdots \\ g_{k-1,0} & g_{k-1,1} & \cdots & g_{k-1,n-1} \end{bmatrix},$$

Se $u = (u_0, u_1, \ldots, u_{k-1})$ é uma mensagem a ser codificada, a palavra-código correspondente será:

$$\begin{aligned} v &= u \cdot G \\ &= (u_0, u_1, \ldots, u_{k-1}) \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{k-1} \end{bmatrix} \\ &= u_0 g_0 + u_1 g_1 + \ldots + u_{k-1} g_{k-1} \end{aligned}$$

A matriz G é dita a matriz geradora do código C. Dizemos que G está na forma sistemática se $G = [P_{k \times n-k} | I_k]$, onde $p_{ij} \in GF(q)$ representam as entradas da matriz P, $0 \leq i \leq k-1$; $0 \leq j \leq n-k-1$ e I_k representa a matriz identidade de ordem k.

Dessa forma, ao codificarmos uma dada mensagem, teremos como resultado uma palavra-código dividida em duas partes, onde os $n-k$ primeiros dígitos corresponderão aos dígitos de verificação de paridade (ou dígitos de redundância) e os k dígitos restantes corresponderão à própria mensagem. Neste caso, o código C será dito um código de bloco linear sistemático.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Assim se $u = (u_0, u_1, \ldots, u_{k-1})$ for a mensagem enviada e v sua palavra-código correspondente, teremos

$$v = \left(\underbrace{v_0, v_1, \ldots, v_{n-k-1}}_{\text{dígitos de paridade}}, \underbrace{u_0, u_1, \ldots, u_{k-1}}_{\text{dígitos de informação}} \right)$$

Exemplo 1.2.1 *Seja C um código de bloco linear com $n = 7$ e $k = 4$ sobre $GF(2)$. Tome*

$$G = \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{bmatrix} = \left[\begin{array}{ccc|cccc} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

como sendo a matriz geradora de C. Se $u = (1101)$ é a mensagem a ser codificada, temos:

$$\begin{aligned} v &= 1g_0 + 1g_1 + 0g_2 + 1g_3 \\ &= (0001101) \end{aligned}$$

Uma outra matriz associada a um código linear C do tipo (n, k), é a matriz verificação de paridade H, que é uma $(n-k) \times n$ matriz com $n-k$ linhas L.I. tal que $GH^t = 0$. Isto é uma consequência do seguinte resultado:

Proposição 1.2.1 *Para qualquer matriz G do tipo $k \times n$ onde $k < n$ e com entradas em um corpo $GF(q)$, com k linhas L.I., existe uma matriz do tipo $(n-k) \times n$, cujas $n-k$ linhas são L.I., tal que, qualquer vetor no espaço linha de G é ortogonal às linhas de H e qualquer vetor que é ortogonal às linhas de H está no espaço linha de G.*

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Demonstração: Dada

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

onde $\{v_1, v_2, \ldots, v_k\}$ é uma base de S, $S \subset V_n$ (o conjunto de todas as n-uplas de um corpo finito), mostraremos que a dimensão do espaço nulo S_d de S, isto é, $S_d = \{x \in V_n; xG = 0\}$ é igual a $n - k$.

Seja $u_1, u_2, \ldots, u_l$ uma base de S_d. Podemos completar essa base, de modo a obter uma base de V_n.

Denotaremos por $\{u_1, u_2, \ldots, u_l, w_1, w_2, \ldots, w_M\}$, uma base de V_n e mostraremos que $M = k$.

Para isso devemos provar o seguinte:

(i) $\langle w_1 G^t, w_2 G^t, \ldots, w_M G^t \rangle = V_k$ e

(ii) $\{w_1 G^t, w_2 G^t, \ldots, w_M G^t\}$ é L.I., e portanto $w_1 G^t, w_2 G^t, \ldots, w_M G^t$ é base de V_k.

Verifiquemos (i) e (ii).

(i) $\forall u \in V_k$, temos que $u = vG^t$, onde $v \in V_n$ (pois existem k colunas de G que são L.I.).
Daí

$$\begin{aligned} u &= (a_1 u_1 + \ldots + a_l u_l) G^t + (b_1 w_1 + \ldots + b_M w_M) G^t \\ &= 0 + b_1 w_1 G^t + \ldots + b_M w_M G^t \end{aligned}$$

Portanto, $u = b_1 (w_1 G^t) + \ldots + b_M (w_M G^t)$.

(ii) Suponha que $a_1 w_1 G^t + \ldots + a_M w_M G^t = 0$. Daí, $(a_1 w_1 + \ldots + a_M w_M) G^t = 0$. Segue

que $a_1 w_1 + \ldots + a_M w_M \in S_d$ e portanto

$$a_1 w_1 + \ldots + a_M w_M = b_1 u_1 + \ldots + b_l u_l$$

o que implica que

$$b_1 u_1 + \ldots + b_l u_l - (a_1 w_1 + \ldots + a_M w_M) = 0,$$

logo, $a_1 = a_2 = \ldots = a_M = 0$. □

A matriz H gera portanto um código linear com parâmetros $(n, n-k)$. Este código, que denotaremos por C_d, é o espaço nulo do código linear com parâmetros (n, k) gerado pela matriz G. C_d é dito o código dual de C. Assim, uma n-upla v é uma palavra-código de C gerado por G se, e somente se, $vH^t = 0$.

Se $G = [P_{k \times n-k} | I_k]$ é a matriz geradora de um código linear C, então a matriz verificação de paridade será dada por $H = [I_{n-k} | -P^t]$, onde P^t é a matriz transposta de P e I_{n-k} é a matriz identidade de ordem $n-k$.

1.3 Definições

Considere C um código linear (n, k) sobre $GF(q)$, com matriz geradora G e matriz verificação de paridade H. Suponha que uma palavra-código $v = (v_0, \ldots, v_{n-1})$ seja enviada e devido ao ruído no canal de comunicações o vetor recebido seja $r = (r_0, \ldots, r_{n-1})$, com $r \neq v$. Então $r = v + e$, onde $e = (e_0, \ldots, e_{n-1})$ é denominado padrão de erro (ou vetor erro) e a operação $+$ é adição em $GF(q)$.

Definição 1.3.1 *A distância de Hamming entre dois vetores $x = (x_0, \ldots, x_{n-1})$ e $y = (y_0, \ldots, y_{n-1})$, é denotada por $d(x, y)$ e representa o número de coordenadas onde eles diferem.*

Definição 1.3.2 *O peso de Hamming de um vetor* $x = (x_0, \ldots, x_{n-1})$, *é igual ao número de coordenadas não nulas de* x, *que denotamos por* $w(x)$.

Observe que a distância de Hamming é de fato uma métrica, pois satisfaz as propriedades

1) $d(x,y) \geq 0$ e $d(x,y) = 0 \Leftrightarrow x = y$,

2) $d(x,y) = d(y,x)$

3) $d(x,y) + d(y,v) \geq d(x,v)$.

Note também que $d(x,y) = w(x-y)$, onde "-" é a "subtração em $GF(q)$".

O terceiro parâmetro importante de um código linear C, além do comprimento n e da dimensão k, é a distância mínima entre suas palavras-código.

Definimos

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

Note que

$$\begin{aligned} d_{\min}(C) &= \min\{w(x-y) : x, y \in C,\ x \neq y\} \\ &= \min\{w(v) : v \in C,\ v \neq 0\} \\ &= w_{\min}(C), \end{aligned}$$

o peso mínimo de C.

Um código linear (n,k) cuja distância mínima de Hamming é $d_{\min}$, é referido como um código $(n,k,d_{\min})$. Veremos mais adiante que $d_{\min}$ caracteriza a capacidade de correção de erros de um código.

Das definições de peso de Hamming e da matriz verificação de paridade H, temos que, se existe uma palavra-código v no código linear C com peso de Hamming l, então $vH^t = 0$, e esta igualdade se verifica se, e somente se, l colunas de H forem L.D. A partir disso temos:



Teorema 1.3.1 *Seja C um código linear $(n, k, d_{\min})$ com matriz verificação de paridade H. Para cada palavra-código com peso de Hamming l, existem l colunas de H cuja soma é igual ao vetor nulo. Reciprocamente , se existem l colunas de H cujo vetor soma resulta no vetor nulo, existe uma palavra-código em C com peso de Hamming igual a l.*

Corolário 1.3.1 *A distância mínima de um código linear é dada pelo número mínimo de colunas L.D. de sua matriz verificação de paridade H.*

Definição 1.3.3 *Seja C um código linear $(n, k, d_{\min})$ sobre $GF(q)$. Para qualquer vetor a de $GF(q)^n$, o conjunto*

$$a + C = \{a + x : x \in C\}$$

é dito uma classe lateral de C, e a é o líder dessa classe lateral.

Observe que todo vetor de $GF(q)^n$ está numa única classe lateral, e que cada classe lateral contém q^k elementos.

Proposição 1.3.1 *Duas classes laterais quaisquer ou são disjuntas, ou são iguais.*

Considere C um código linear $(n, k, d_{\min})$. Sejam $v_0, v_1, \ldots, v_{q^k-1}$, as palavras-código de C. Vamos construir uma tabela que chamaremos de arranjo padrão de C da seguinte forma:

i) Na primeira linha, coloque todas as palavras-código, iniciando pela palavra-código nula;

ii) Nas demais linhas, coloque as classes laterais $a_i + C$, arranjadas na mesma ordem que a 1^a linha, iniciando pelo líder de classe lateral.

A tabela a seguir, representa um arranjo padrão, onde os v_i, $0 \leq i \leq q^k - 1$, representam as palavras-código de um código $C(n, k)$ e os a_j, $1 \leq j \leq n - k - 1$, são os líderes

das classes laterais.

1^a linha	$v_0 = 0$	v_1	v_2	$\cdots$	v_{q^k-1}
2^a linha	a_1	$a_1 + v_1$	$a_1 + v_2$		$a_1 + v_{q^k-1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
q^{n-k}-ésima linha	$a_{q^{n-k}-1}$	$a_{q^{n-k}-1} + v_1$	$a_{q^{n-k}-1} + v_2$		$a_{q^{n-k}-1} + v_{q^k-1}$

Proposição 1.3.2 *Duas n-uplas numa mesma linha de um arranjo padrão nunca são iguais. Toda n-upla aparece em exatamente uma linha e uma coluna do arranjo padrão.*

1.4 Detecção de Erros

Nessa seção, descreveremos a estratégia do decodificador para que este possa detectar e posteriormente, corrigir os erros introduzidos pelo canal.

Sejam C um código linear $(n, k, d_{\min})$, $v = (v_0, \ldots, v_{n-1})$ a palavra-código enviada através de um canal de comunicações, $e = (e_0, \ldots, e_{n-1})$ o padrão de erro ocorrido e $r = (r_0, \ldots, r_{n-1})$ a n-upla recebida.

O primeiro passo do decodificador é o cálculo da *síndrome* s, a qual é dada por

$$\begin{aligned} s &= r \cdot H^t = (v + e) H^t = vH^t + eH^t = eH^t \\ &= (s_0, s_1 \ldots, s_{n-k-1}) . \end{aligned}$$

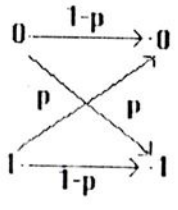
Se $s = 0$, r é aceita como a palavra-código enviada. Agora, note que se o padrão de erro e for igual a uma palavra-código, então, $r = v + e$ também é uma palavra-código e $r \neq v$. Chamamos esse tipo de erro de *erro não detectável*. Como existem $q^k - 1$ palavras-código não nulas, haverão $q^k - 1$ padrões de erro não detectáveis.

Se $s \neq 0$, então r não é palavra-código e a presença de erros é detectada. Todos os padrões de erro cujos pesos são $d_{\min} - 1$ ou menos não podem transformar uma palavra-código em outra e portanto são detectáveis. Além disso, se o padrão de erro não coincide

com uma palavra-código em C, o vetor recebido r não será palavra-código e portanto será detectável. Perceba que há um total de $q^n - q^k$ erros detectáveis e portanto, se n for um número grande comparado com k, apenas uma pequena parcela dos erros é não detectável.

Para calcular a probabilidade de erro do decodificador (evento em que o decodificador diz que não ocorreram erros quando na verdade eles ocorreram), vamos supor que os erros ocorram de forma independente com probabilidade p, de acordo com o modelo do canal.

Por exemplo, se tivermos um canal binário BSC ("Binary Symmetric Channel", do inglês), teremos o seguinte modelo:



onde os erros ocorrem de forma independente em cada variável com probabilidade p, $0 \leq p \leq \frac{1}{2}$. No caso da probabilidade p ser maior que $\frac{1}{2}$, bastaria trocar todos os símbolos de saída e novamente a probabilidade de erro em cada variável seria menor que $\frac{1}{2}$.

Nesse canal BSC, a probabilidade ocorrer o padrão de erro $e = (11000)$ é:

$$\text{Prob}\{e = 11000\} = pp(1-p)(1-p)(1-p) = p^2(1-p)^3.$$

Em geral, a probabilidade Prob $\{e\}$ de que ocorra um determinado padrão de erro de peso a é dada por:

$$p^a(1-p)^{n-a} \tag{1.1}$$

onde a é o número de dígitos não nulos do padrão de erro e, ou o peso de Hamming $w(e)$.

Agora, dado um código linear C, façamos uma distribuição dos pesos de Hamming

de todas as suas palavras-código. Para isso, tome o conjunto $\{A_0, A_1, \ldots, A_n\}$, onde A_i corresponde ao número de palavras-código com peso de Hamming igual a i, $0 \leq i \leq n$. Observe que $A_0 = 1$, pois há somente uma palavra-código toda nula.

Então, com respeito a essa distribuição $\{A_0, A_1, \ldots, A_n\}$, teremos que a probabilidade de ocorrer um padrão de erro não detectável é

$$P_u(E) = \sum_{i=1}^{n} A_i p^i (1-p)^{n-i} \tag{1.2}$$

1.5 Correção de Erros

Na Seção 1.4 vimos que o primeiro passo da decodificação é o cálculo da síndrome s. Se $s \neq 0$ (ou seja, foram detectados erros), os dígitos da síndrome $s = (s_0, \ldots, s_{n-k-1})$, são dados em função dos dígitos do padrão de erro. Assim, se a matriz geradora de um código linear binário C com parâmetros $(n, k, d_{\min})$ é dada por:

$$G = \left[\begin{array}{cccc|cccc} p_{00} & p_{01} & \cdots & p_{0,n-k-1} & 1 & 0 & \ldots & 0 \\ p_{10} & p_{11} & \cdots & p_{1,n-k-1} & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & 0 \\ p_{k-1,0} & p_{k-1,1} & \cdots & p_{k-1,n-k-1} & 0 & 0 & \ldots & 1 \end{array}\right],$$

então sua matriz verificação de paridade será

$$H = \left[\begin{array}{cccc|cccc} 1 & 0 & \ldots & 0 & p_{00} & p_{01} & \cdots & p_{k-1,0} \\ 0 & 1 & \ldots & 0 & p_{01} & p_{11} & \cdots & p_{k-1,1} \\ \vdots & \vdots & & 0 & \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots & 1 & p_{0,n-k-1} & p_{1,n-k-1} & \cdots & p_{k-1,n-k-1} \end{array}\right]$$

e se na transmissão de uma palavra-código ocorrer o padrão de erro $e = (e_0, e_1, \ldots, e_{n-1})$, ao calcularmos a síndrome $s = eH^t$, obtemos

$$\begin{aligned} s_0 &= e_0 + e_{n-k}p_{00} + \ldots + e_{n-1}p_{k-1,0} \\ s_1 &= e_1 + e_{n-k}p_{01} + \ldots + e_{n-1}p_{k-1,1} \\ &\vdots \\ s_{n-k-1} &= e_{n-k-1} + e_{n-k}p_{0,n-k-1} \ldots + e_{n-1}p_{k-1,n-k-1} \end{aligned}$$

As equações da síndrome produzem q^k possíveis soluções, ou seja, existem q^k padrões de erro que possuem a mesma síndrome.

Sabemos que $\text{Prob}\{w(e) = a\} = p^a(1-p)^{n-a}$ é a probabilidade de ocorrer um padrão de erro com peso a. Temos também que num canal BSC os erros em cada transmissão ocorrem de forma independente (das outras transmissões) e com probabilidade p, onde $0 \leq p \leq \frac{1}{2}$. Portanto $1 - p > p$. Assim,

$$(1-p)^n > p(1-p)^{n-1} > p^2(1-p)^{n-2} > \ldots > p^n.$$

Temos então que o padrão de erro mais provável de ter ocorrido é aquele de peso de Hamming mínimo e por isso o decodificador escolherá entre os q^k possíveis padrões de erro que são soluções para as equações da síndrome, o que possuir o menor peso de Hamming.

Esse tipo de decodificação, onde o vetor escolhido é o mais provável, é dito decodificação pelo vizinho mais próximo, ou ainda, decodificação de máxima verossimilhança.

Exemplo 1.5.1 *Retomemos o código (7,4) do Exemplo 1.2.1. Suponha que* $v = (1\,0\,0\,1\,0\,1\,1)$ *seja a palavra-código transmitida e que* $r = (1\,0\,0\,1\,0\,0\,1)$, *seja o vetor recebido.*

Calculando a síndrome, temos:

$$s = rH^t = (1\,1\,1)\,.$$

Os dígitos da síndrome estão relacionados com os dígitos do vetor erro $e = (e_0, e_1, \ldots, e_6)$*, através das seguintes equações lineares:*

$$1 = e_0 + e_3 + e_5 + e_6$$
$$1 = e_1 + e_3 + e_4 + e_5$$
$$1 = e_2 + e_4 + e_5 + e_6$$

Há um total de $2^4 = 16$ *padrões de erro que satisfazem as equações acima. Dentre eles,* $e = (0\,0\,0\,0\,0\,1\,0)$ *é o de menor peso de Hamming e se o canal é BSC este será o erro mais provável e temos:*

$$\widehat{v} = r + e = (1\,0\,0\,1\,0\,1\,1)\,.$$

Descreveremos agora, um esquema de decodificação baseado no arranjo padrão de um código linear C, conforme descrito na Seção 1.3.

Assim, seja C um código linear com parâmetros $(n, k, d_{\min})$ sobre $GF(q)$. C é um subespaço de V^n (V^n denota o espaço vetorial de todas as n-uplas sobre $GF(q)$).

Então,

$$V^n = C \cup (a_1 + C) \cup \ldots \cup \left(a_{q^{n-k}-1} + C\right)$$

onde $a_i + C$ são as classes laterais de C em V^n, $1 \leq i \leq q^{n-k} - 1$. Nas q^{n-k} classes laterais distintas, tome como líder de classe lateral o vetor de menor peso de Hamming em cada uma delas.

Se $v \in C$ é a palavra-código transmitida e $r = v + e$ o vetor recebido, temos que r pertence a alguma das classes laterais. Suponha que r pertença à classe $a_i + C$. Então, $r = a_i + v_j$, onde $v_j \in C$. Segue que $r = v + e = a_i + v_j$, logo $e = a_i + v_j + v$, onde $(v_j + v) = v_l \in C$ e $e = a_i + v_l$, ou seja, e pertence à classe $a_i + C$. Concluímos então

que o vetor recebido r pertence a mesma classe lateral que o padrão de erro ocorrido e.

A estratégia de decodificação utilizando o arranjo padrão será então localizar a classe lateral onde se encontra o vetor recebido r. O padrão de erro estimado e será o mais provável, ou seja, o de menor peso de Hamming, o qual é o líder da classe lateral onde está r. Assim, $r = e + v$, onde r e e são conhecidos e portanto v também o é.

Observe pela tabela a seguir que v corresponderá à palavra-código localizada no topo da coluna que contém r e o vetor a_i é o de menor peso dentre os vetores da linha i, $1 \leq i \leq q^{n-k} - 1$. Assim, basta encontrar a posição de r no arranjo padrão para sabermos a palavra-código enviada e o padrão de erro ocorrido.

0	v_2	v_3	$\cdots$	v_{q^k}
a_1	$a_1 + v_2$	$a_1 + v_3$	$\cdots$	$a_1 + v_{q^k}$
a_2	$a_2 + v_2$	$a_2 + v_3$	$\cdots$	$a_2 + v_{q^k}$
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\cdots$
$a_{q^{n-k}-1}$	$a_{q^{n-k}-1} + v_2$	$a_{q^{n-k}-1} + v_3$	$\cdots$	$a_{q^{n-k}-1} + v_{q^k}$

Arranjo Padrão de um Código Linear (n, k) sobre $GF(q)$

Assim, ao utilizarmos o arranjo padrão, o decodificador cometerá um erro se, e somente se, o vetor erro for diferente de um líder de classe lateral.

A probabilidade de erro de decodificação é a probabilidade de que a saída do decodificador seja uma palavra-código diferente da enviada e ela é dada por:

$$P_e = \left(\frac{1}{M}\right) \sum_{i=1}^{M} \text{Prob}\left\{\text{saída do decodificador } \neq \ v_i \mid v_i \text{ foi enviada}\right\},$$

onde M é o número de palavras $v_1, v_2, \ldots, v_M$, todas usadas com igual probabilidade.

Um erro de decodificação ocorre se e não é líder de classe lateral. Assim:

$$P_e = \text{Prob}\left\{e \neq \text{ líder de classe lateral}\right\}.$$

Seja α_i o número de líderes de classe lateral com peso i. Pela Equação em (1.2), temos

$$P_c = 1 - \sum_{i=0}^{n} \alpha_i p^i (1-p)^{n-i}.$$

Quanto à capacidade de correção de erros de um código C (linear ou não) de distância mínima $d_{\min}$ temos:

Teorema 1.5.1 *Se o padrão de erro ocorrido tiver peso de Hamming $w(e) \leq t = \left\lfloor \frac{d_{\min}-1}{2} \right\rfloor$, onde $\lfloor x \rfloor$ representa o maior inteiro menor ou igual a x, então o decodificador decodificará o vetor recebido na palavra-código que foi de fato enviada. Por outro lado, se $t > \left\lfloor \frac{d_{\min}-1}{2} \right\rfloor$, então o decodificador poderá cometer um erro de decodificação, ou seja, decodificar o vetor recebido em um vetor diferente da palavra-código enviada.*

Demonstração: Lembrando que todas as palavras-código são equiprováveis, a estratégia que minimiza a probabilidade de erro é aquela em que o vetor recebido é convertido na palavra-código mais próxima (no sentido de distância de Hamming). Assim, sejam v a palavra-código transmitida e r o vetor recebido. Tome $w \neq v$ uma palavra-código em C. Temos:

$$d(v,r) + d(w,r) \geq d(v,w) \tag{1.3}$$

Suponha que o peso do padrão de erro ocorrido durante a transmissão de v seja t'. Assim, r difere de v em t' posições e assim $d(v,r) = t'$.

Como v e w são palavras-código em C, temos

$$d(v,w) \geq d_{\min} \geq 2t+1 \tag{1.4}$$

De (1.3) e (1.4) e do fato de $d(v,r) = t'$ segue que

$$d(w,r) \geq 2t + 1 - t'$$

Se $t' \leq t$ então

$$d(w,r) > t.$$

Com isso, temos que se ocorrem t ou menos erros o vetor r estará mais próximo (no sentido de distância de Hamming) da palavra código transmitida v do que qualquer outra palavra-código $w \in C$.

Por outro lado, se ocorrer um erro de peso l, onde $l > t$, haverá ao menos um caso onde o vetor recebido estará mais próximo de uma palavra-código diferente da enviada.

Para mostrarmos isto, suponha v e w palavras-código em C, tais que $d(v, w) = d_{\min}$.

Sejam e_1 e e_2 dois padrões de erro que satisfaçam:

i) $e_1 + e_2 = v + w$

ii) e_1 e e_2 não possuem componentes não nulas nas mesmas posições.

Segue que

$$w(e_1) + w(e_2) = w(v + w) = d(v, w) = d_{\min}. \tag{1.5}$$

Suponha que na transmissão de v, ocorra o padrão de erro e_1 onde $w(e_1) > t$. Assim, $r = v + e_1$ é o vetor recebido e teremos:

$$d(v, r) = w(v + r) = w(e_1). \tag{1.6}$$

A distância de Hamming entre w e r é dada por

$$d(w, r) = w(w + r) = w(w + v + e_1) = w(e_2). \tag{1.7}$$

Como $w(e_1) > t$ e usando o fato de que $2t + 1 \leq d_{\min} \leq 2t + 2$, segue de (1.5) que

$$w(e_2) \leq t + 1$$

De (1.6) e (1.7) e do fato de $w(e_1) > t$ e $w(e_2) \leq t + 1$, segue que

$$d(v, r) \geq d(w, r),$$

ou seja, se ocorrer um padrão de erro com peso $l > t$ o vetor recebido r estará mais próximo de $w \in C$, onde w é diferente da palavra-código enviada v. □

Definição 1.5.1 *A taxa de um código de bloco linear $(n, k, d_{\min})$ é a razão $R = \frac{k}{n}$ de símbolos por bloco, onde k é a dimensão e n é o comprimento do código.*

Definição 1.5.2 *Definimos a capacidade de um canal binário simétrico com probabilidade de erro p como sendo*

$$C(p) = 1 + p \log_2 p + (1-p) \log_2 (1-p)$$

bits por transmissão.

Teorema 1.5.2 *(Teorema de Shannon da Codificação de Canal) [2] Dado $\varepsilon > 0$, se $R < C(p)$, existe um código binário (n, k), com n suficientemente grande, com taxa $\frac{k}{n} \geq R$ e cuja probabilidade de erro satisfaz $P_e < \varepsilon$. Ou seja, para códigos binários, dada a taxa $R' = \frac{k}{n}$, a probabilidade de erro P_e pode se tornar arbitrariamente pequena se tomarmos R' menor que a capacidade do canal.*

Este teorema é válido também para códigos q-ários.

1.6 Limitantes

Daremos agora limitantes para os parâmetros dos códigos lineares.

Teorema 1.6.1 *(Limitante de Singleton) Se C é um código linear com parâmetros $[n, k, d_{\min}]$, então $d_{\min} \leq n - k + 1$.*

Demonstração: Se H é a matriz verificação de paridade de C, o posto de H é $n-k$ e portanto o número máximo de colunas L.I. em H é também $n-k$.

Assim qualquer conjunto de $n-k+1$ colunas de H é L.D. e portanto $d_{\min} \leq n-k+1$.

□

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Os códigos $(n, k, d_{\min})$, com $d_{\min} = n - k + 1$, são ditos separáveis à máxima distância (ou MDS, "maximum distance separable", do inglês).

O próximo teorema apresenta um limitante inferior para os códigos lineares.

Teorema 1.6.2 *(O limitante de Gilbert- Varshamov) Existe um código linear binário, de comprimento n, com no máximo r dígitos de verificação de paridade e distância mínima pelo menos d se*

$$1 + \binom{n-1}{1} + \binom{n-1}{2} + \ldots + \binom{n-1}{d-2} < 2^r.$$

Demonstração: Devemos construir a matriz verificação de paridade H do tipo $r \times n$ de forma que, nenhum conjunto com $d-1$ ou menos colunas de H seja L.D. e pelo Teorema 1.3.1 verificaremos o limitante.

Escolha para a $1^{\underline{a}}$ coluna uma r-upla não nula; para $2^{\underline{a}}$ coluna tome outro vetor L.I. com relação a $1^{\underline{a}}$ coluna e assim até encontrarmos um conjunto com i colunas onde nenhum conjunto de $d-1$ ou menos colunas seja linearmente dependente.

Há um total de $\binom{i}{1} + \ldots + \binom{i}{d-2}$ combinações lineares distintas entre essas colunas, tomadas de $d-2$ a $d-2$ ou menos.

Se o total de combinações possíveis for menor que $2^r - 1$, podemos escolher mais uma coluna diferente dessas combinações lineares e obteremos que qualquer conjunto com $d-1$ colunas serão linearmente independentes.

Continuamos o processo enquanto

$$1 + \binom{i}{1} + \ldots + \binom{i}{d-2} < 2^r.$$

□

O limitante de Gilbert-Varshamov estendido para códigos lineares sobre $GF(q)$ nos diz que existe um código de comprimento n, com no máximo r dígitos de verificação de

paridade e distância mínima d se

$$\sum_{i=0}^{d-2} (q-1)^i \binom{n-1}{i} < q^r.$$

1.7 Entropia e Quantidade de Informação

Suponha que num determinado experimento, ocorra um resultado R. A esse resultado R damos o nome de evento. Se $A_1, \ldots, A_n$ é o conjunto de todos os possíveis resultados ao realizarmos um experimento, teremos então um sistema completo de eventos. Sejam $A_1, \ldots, A_n$ um sistema completo de eventos onde $A_i \cap A_j = \emptyset$, $\forall i, j \in 1, \ldots, n$ e a cada experimento ocorra um e somente um evento.

Um esquema finito A é um sistema completo de eventos juntamente com suas respectivas probabilidades e é representado por

$$A = \begin{pmatrix} A_1 & A_2 & \cdots & A_n \\ p_1 & p_2 & \cdots & p_n \end{pmatrix},$$

onde $\sum_{i=1}^{n} p_i = 1$.

Para medirmos a quantidade $H(A)$ de incerteza (ou quantidade de informação de A), gostaríamos que $H(p_1, p_2, \ldots, p_n)$ assumisse o maior valor quando

$$p_1 = p_2 = \ldots = p_n = \frac{1}{n}. \tag{1.8}$$

Agora, suponha dado um outro esquema finito B tal que:

$$B = \begin{pmatrix} B_1 & B_2 & \cdots & B_m \\ q_1 & q_2 & \cdots & q_m \end{pmatrix}.$$

Definimos o esquema AB, como sendo o conjunto de eventos $A_k B_l$, $1 \leq k \leq n$, $1 \leq l \leq m$, com as suas respectivas probabilidades.

Definimos a quantidade $H_k(B)$ como sendo a incerteza do esquema B, dado que ocorreu o evento A_k.

Assim, é razoável definirmos $H(AB) = H(A) + \sum_{k=1}^{n} p_k H_k(B)$, onde $\sum_{k=1}^{n} p_k H_k(B) = H_A(B)$.

Assim

$$H(AB) = H(A) + H_A(B). \tag{1.9}$$

A inclusão do evento impossível não deve alterar a quantidade de incerteza que temos a respeito de A. Assim,

$$H(p_1, p_2, \ldots, p_n, 0) = H(p_1, p_2, \ldots, p_n). \tag{1.10}$$

A partir de (1.8), (1.9) e (1.10) temos:

Teorema 1.7.1 *Seja $H(p_1, p_2, \ldots, p_n)$ uma função definida para qualquer $n \in \mathbb{N}$ e para todos valores $p_1, p_2, \ldots, p_n$, tais que $p_k \geq 0$, para $k = 1, \ldots, n$ e $\sum_{k=1}^{n} p_k = 1$. Se para qualquer n, essa função H é contínua com relação a todas as variáveis e além disso valem (1.8), (1.9) e (1.10), então H fica completamente determinada a menos de uma constante positiva λ, isto é,*

$$H(p_1, p_2, \ldots, p_n) = -\lambda \sum_{k=1}^{n} p_k \log p_k$$

Assim, definimos a *entropia* de uma variável aleatória discreta X, como sendo:

$$H(X) = -\sum_{x \in \chi} p(x) \log p(x)$$

onde χ é o alfabeto da variável aleatória. Note então que $H(X)$ mede a "quantidade de incerteza" que se tem a respeito de X.

Tome agora um código binário com parâmetros (n, M, d), onde n denota o seu comprimento, M é o número de palavras-código e d a distância mínima. Então a taxa de

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

informação desse código é definida por:

$$R = \frac{\log_2 M}{n} \text{ bits/transmissão.}$$

Fixados n e d, o melhor código é aquele onde M é máximo, ou seja, é aquele que possui a maior taxa de informação. Na definição de taxa para código q-ários, tomamos os logaritmos na base q.

Definição 1.7.1 $A(n,d) = \max\{M \mid \text{existe um código } (n,M,d)\}$.

O problema central da Teoria dos Códigos Combinatorial é determinar $A(n,d)$, para todos os valores possíveis de n e d.

Para fazermos a probabilidade de erro tender a zero, devemos fixar a taxa e aumentar o comprimento. (ver Teorema de Shannon da Codificação de Canal, p. 19)

Se a probabilidade de erro de cada símbolo é p, o número médio de símbolos errados em uma palavra recebida é np, e portanto d deve crescer tão rápido quanto (ou mais) do que $2np$ se desejamos corrigir todos esses erros.

Assim, surge a quantidade: $\alpha(\delta) = \lim_{n\to\infty} \sup \log_q A(n,\delta n)/n$.

A partir daqui, apresentaremos os resultados necessários para chegarmos ao Limitante Assintótico de Gilbert-Varshamov, isto é,

$$\alpha(\delta) \geq 1 - H_q(\delta),$$

se $0 \leq \delta \leq \theta$ onde $\theta = \frac{q-1}{q}$.

A função $H_q : [0,\theta] \to [0,1]$ é a função entropia dada por:

$$H_q(x) = \begin{cases} 0, \text{ se } x = 0 \\ x\log_q(q-1) - x\log_q x - (1-x)\log_q(1-x), \text{ se } 0 \leq x \leq \theta \end{cases}$$

Para provarmos o limitante de GV para o caso binário, utilizaremos o seguinte resultado:

Lema 1.7.1 $\sum_{k=0}^{\mu n} \binom{n}{k} \leq 2^{nH_2(\mu)}$, $0 \leq \mu \leq \frac{1}{2}$.

Demonstração: Suponha primeiramente que $\frac{1}{2} \leq \lambda \leq 1$ e provemos que

$$\sum_{k=\lambda n}^{n} \binom{n}{k} \leq 2^{nH_2(\lambda)}.$$

Temos:

$$2^{r\lambda n} \sum_{k=\lambda n}^{n} \binom{n}{k} \leq \sum_{k=\lambda n}^{n} 2^{rk} \binom{n}{k} \leq \sum_{k=0}^{n} 2^{rk} \binom{n}{k} = (1+2^r)^n.$$

Portanto,

$$\sum_{k=\lambda n}^{n} \binom{n}{k} \leq \left(2^{-r\lambda} + 2^{r(1-\lambda)}\right)^n.$$

Escolhendo $r = \log_2 \frac{\lambda}{1-\lambda}$, obtemos:

$$\sum_{k=\lambda n}^{n} \binom{n}{k} \leq \left(2^{-\log_2 \lambda + \lambda \log_2 (1-\lambda)} + 2^{(1-\lambda)\log_2 \lambda - (1-\lambda)\log_2 (1-\lambda)}\right)^n.$$

Lembrando que $H_2(\lambda) = -\lambda \log_2 \lambda - (1-\lambda)\log_2(1-\lambda)$, vem:

$$\sum_{k=\lambda n}^{n} \binom{n}{k} \leq 2^{-\log \lambda - (1-\lambda)\log(1-\lambda)} A,$$

onde $A = 2^{\log(1-\lambda)} + 2^{\log(\lambda)} = (1-\lambda) + \lambda = 1$.

Logo

$$\sum_{k=\lambda n}^{n} \binom{n}{k} \leq 2^{H_2(\lambda)n},$$

se $\frac{1}{2} \leq \lambda \leq 1$. □

Passemos agora a demonstração do Limitante de Gilbert-Varshamov para o caso

binário. Pelo Teorema 1.6.2 existe um código linear binário (n, k, d) para o qual vale:

$$\sum_{i=0}^{d-2} \binom{n}{i} \geq 2^{n-k}.$$

Agora, $\sum_{i=0}^{d-2} \binom{n}{i} \leq 2^{nH_2\left(\frac{d-2}{n}\right)}$ se $\frac{d-2}{n} \leq \frac{1}{2}$. Segue então que $2^{n-k} \leq 2^{nH_2\left(\frac{d-2}{n}\right)}$. Portanto,

$$\begin{aligned} n-k &\leq nH_2\left(\frac{d-2}{n}\right) \Rightarrow 1-\frac{k}{n} \leq H_2\left(\frac{d-2}{n}\right) \Rightarrow \\ &\Rightarrow 1-H_2\left(\frac{d-2}{n}\right) \leq R \end{aligned}$$

onde $R = \frac{k}{n}$.

Em outras palavras, $R \geq 1 - H_2\left(\frac{d-2}{2}\right)$. Assintoticamente $R \gtrsim 1 - H_2\left(\frac{d}{2}\right)$ para $n \to \infty$ $\left(R \geq \left(1 - H_2\left(\frac{d}{n}\right)\right)\varepsilon$, onde $\varepsilon \to 1$ para $n \to \infty\right)$.

Concluímos que $H_2\left(\frac{d}{n}\right) \geq 1 - R$.

Como $\frac{d}{n} \geq H_2^{-1}(1-R)$ e observando que H_2^{-1} é uma função crescente, temos que existem códigos (n, k, d) tais que $\frac{d}{n} \geq H_2^{-1}(1-R)$.

Com isso obtemos o seguinte resultado:

Teorema 1.7.2 (*O Limitante de Gilbert-Varshamov Assintótico) Suponha que* $0 \leq \delta \leq \frac{1}{2}$. *Então existe uma sequência infinita de códigos lineares binários com parâmetros* (n, k, d) *com* $\frac{d}{n} \geq \delta$ *e taxa* $R = \frac{k}{n}$ *satisfazendo*

$$R \geq 1 - H_2\left(\frac{d}{n}\right)$$

para todo n.

1.8 Códigos de Hamming

Os códigos de Hamming binários formam uma família de códigos de fácil codificação e decodificação.

Definição 1.8.1 *Um código de Hamming binário $\mathcal{H}_r$, de comprimento $n = 2^r - 1$ $(r \geq 2)$ tem matriz verificação de paridade H formada por todos os vetores binários não nulos de comprimento r, cada um usado apenas uma vez como uma coluna de H. $\mathcal{H}_r$ é um código com parâmetros $n = 2^r - 1$, $k = 2^r - 1 - r$ e $d_{\min} = 3$.*

Observe que como $d_{\min} = 3$, $\mathcal{H}_r$ é capaz de corrigir todos os padrões de erro contendo até um erro.

A matriz verificação de paridade de um código de Hamming binário é formada por todos os vetores binários não nulos de comprimento r. É conveniente arranjarmos as colunas de H na ordem crescente em relação à representação decimal, pois com isso a decodificação tornar-se-á realmente simples.

Suponha que r seja o vetor recebido e que ocorra um erro na l-ésima posição. Assim, o vetor erro será da forma $e = (0, 0, \ldots, 1, 0, \ldots, 0)$.

Ao calcularmos a síndrome s, teremos como resultado um vetor igual à l-ésima coluna de H, ou seja, sabemos exatamente a posição onde o erro ocorreu.

Vamos ilustar esse fato com o seguinte exemplo.

Exemplo 1.8.1 *Considere o código de Hamming de comprimento $n = 7$. Nesse caso, $r = 3$. A dimensão do código é $k = n - r = 4$, e a distância mínima é $d = 3$. A matriz verificação de paridade é dada por*

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

É conveniente arranjar as colunas de H na ordem crescente com respeito à representação decimal. Assim,

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

Se enviarmos a palavra código $x = (0000000)$ e recebermos o vetor $r = (1000000)$, teremos $s = H' r^t = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ que é a representação binária da 1^a coluna.

1.9 Propriedades Elementares de Corpos Finitos

Nesta seção trataremos das propriedades dos corpos finitos, que serão utilizadas na construção e decodificação dos códigos que serão futuramente apresentados.

Assim, se p é um número primo, os inteiros módulo p, denotados por $GF(p)$ ou $\mathbb{Z}_p$, formam um corpo finito com p elementos $\{0, 1, 2, \ldots, p-1\}$, cujas operações $+, -, \times, \div$, são tomadas $\bmod p$. Dizemos que um corpo E é uma extensão de um corpo F se $F \subset E$.

Podemos construir o corpo extendido $GF(p^m)$ a partir de $GF(p)$ utilizando um polinômio $\pi(x)$ com grau m que satisfaça:

i) Os coeficientes de $\pi(x)$ estão em $GF(p)$.

ii) $\pi(x)$ é irredutível sobre $GF(p)$.

Teorema 1.9.1 *Seja $\pi(x)$ um polinômio irredutível sobre $GF(p)$ de grau m. Então o conjunto de todos os polinômios em x de grau $\leq m-1$ e coeficientes em $GF(p)$, calculados módulo $\pi(x)$, formam um corpo de ordem p^m.*

Vale observar que um elemento de $GF(p^m)$ pode ter diversas representações. Podemos representá-lo como uma m-upla de elementos de $GF(p)$, ou ainda, descrevê-los como classes residuais de polinômios em x com coeficientes em $GF(p)$, tomados $\bmod \pi(x)$.



Assim, se α é o elemento $0100\ldots0$ de $GF(p^m)$ temos que $\pi(\alpha) = 0$; ou seja, a equação $\pi(x)$ tem α como raiz, onde $\alpha \in GF(p^m)$. Neste caso, dizemos que $GF(p^m)$ foi obtido a partir de $GF(p)$ pela adição (a $GF(p)$) de um zero de $\pi(x)$.

Concluímos que $GF(p^m)$ consiste de todos os polinômios em α de grau $\leq m-1$, com coeficientes em $GF(p)$. Um elemento de $GF(p^m)$ será da forma:

$$a_0 a_1 \ldots a_{m-1} \leftrightarrow a_0 + a_1\alpha \ldots a_{m-1}\alpha^{m-1},$$

onde $a_i \in GF(p)$ e $\pi(\alpha) = 0$.

Se F é um corpo finito de ordem q e elemento unitário 1, com $\underbrace{1+1+\ldots+1=0}_{p \text{ vezes}}$ onde p é o menor número inteiro positivo tal que isso ocorre, então p é característica do corpo F. Note que p é primo, obrigatoriamente.

Se F tem característica p, então $\forall\beta \in F$, $p\beta = 0$ e os elementos $0, 1, 2, \ldots, p-1$ estão em F formando um subcorpo $GF(p)$ de F. Se $q = p$, então $F = GF(p)$.

No caso de $q > p$, escolhemos um conjunto máximo de elementos de F linearmente independente sobre $GF(p)$, digamos, $\beta_0 = 1, \beta_1, \ldots, \beta_{m-1}$ e temos que F contém todos os elementos $a_0\beta_0 + a_1\beta_1 + \ldots + a_{m-1}\beta_{m-1}$, $a_i \in GF(p)$ e não outros.

Concluímos que F é um espaço vetorial de dimensão m sobre $GF(p)$ que possui $q = p^m$ elementos (isto é, F tem ordem p^m).

Teorema 1.9.2 *Seja F um corpo de ordem p^m e F^* o conjunto dos elementos não nulos de F. Então F^* é um grupo cíclico multiplicativo de ordem $p^m - 1$.*

Definição 1.9.1 *$\alpha \in F$ é dito um elemento primitivo se tem ordem $p^m - 1$ (isto é, α gera o grupo cíclico F^*).*

Corolário 1.9.1 *Todo corpo finito contém um elemento primitivo.*

Teorema 1.9.3 *(Teorema de Fermat) $\forall\beta \in F$, onde F é um corpo de ordem p^m, temos*

$\beta^{p^m} = \beta$ *ou seja,* β *é raiz da equação* $x^{p^m} = x$. *Então,*

$$x^{p^m} - x = \prod_{\beta \in F} (x - \beta) .$$

Definição 1.9.2 *Uma classe ciclotômica módulo* $p^m - 1$ *sobre* $GF(p)$ *contendo* s *é o conjunto*

$$C_s = \left\{ s, sp, sp^2, \ldots, sp^{m_s - 1} \right\}$$

onde m_s *é o menor inteiro positivo tal que* $p^{m_s} s \equiv s \bmod (p^m - 1)$.

Definição 1.9.3 *O polinômio minimal* $M(x)$ *de* β *sobre* $GF(p)$ *é o polinômio mônico de menor grau com coeficientes em* $GF(p)$ *tal que* $M(\beta) = 0$.

Propriedades dos polinômios minimais de elementos de $GF(p^m)$:

M1) Um polinômio minimal $M(x)$ é sempre irredutível.

M2) Se $f(x)$ é um polinômio com coeficientes em $GF(p)$ tal que $f(\beta) = 0$, então $M(x) \mid f(x)$.

M3) $M(x) \mid x^{p^m} - x$.

M4) $\partial M(x) \leq m$.

M5) O polinômio minimal de um elemento primitivo de $GF(p^m)$ tem grau m e é chamado polinômio primitivo.

M6) β e β^p tem o mesmo polinômio minimal.

Elementos de um corpo que possuem o mesmo polinômio minimal são ditos conjugados.

Teorema 1.9.4 *Quaisquer dois corpos finitos de ordem* p^m *são isomorfos.*

Teorema 1.9.5 *Para todo número primo* p *e inteiro* $m \geq 1$, *existe um corpo com ordem* p^m, *chamado o corpo de Galois de ordem* p^m *e denotado por* $GF(p^m)$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Lema 1.9.1 *Se n, r, s são inteiros com $n \geq 2$, $r \geq 1$, $s \geq 1$, então*

$$n^s - 1 | n^r - 1 \Leftrightarrow s | r.$$

Teorema 1.9.6 *Seja $GF(p^r)$ um corpo de extensão de $GF(p)$. Então $GF(p^r)$ possui um subcorpo $GF(p^s)$ se, e somente se, $s|r$. Se $\beta \in GF(p^r)$, então $\beta \in GF(p^s)$ se, e somente se, $\beta^{p^s} = \beta$.*

Proposição 1.9.1 *A operação de multiplicação por p particiona os inteiros* $\bmod p^m - 1$ *em conjuntos que consistem das classes laterais ciclotômicas* $\bmod p^m - 1$.

Denotaremos por C_s a classe ciclotômica contendo s, caso s seja o menor número dessa classe.

Definiremos o polinômio $M^{(i)}(x)$ como sendo o polinômio minimal de $\alpha^i \in GF(p^m)$.

Pela propriedade M6), temos que $M^{(pi)}(x) = M^{(i)}(x)$ e se $i \in C_s$ então em $GF(p^m)$ temos $\prod\limits_{j \in C_s} (x - \alpha^j)$ divide $M^{(i)}(x)$.

M7) Se $i \in C_s$ então

$$M^{(i)}(x) = \prod_{j \in C_s} \left(x - \alpha^j\right)$$

daí pelo Teorema 1.9.3 temos

$$x^{p^m - 1} - 1 = \prod_s M^{(s)}(x)$$

onde s percorre os representantes das classes ciclotômicas $\bmod p^m - 1$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Capítulo 2

Códigos Cíclicos e BCH

2.1 Introdução

Os códigos cíclicos tem grande importância teórica (pois utilizam extensivamente conceitos de álgebra abstrata) e prática (a complexidade de codificação e decodificação é pequena quando comparada com a de um código linear geral).

Na Seção 2.2 definiremos os códigos cíclicos como sendo um ideal no anel dos polinômios módulo $x^n - 1$. Apresentaremos seu polinômio gerador $g(x)$ e o polinômio verificação de paridade $h(x)$.

Na Seção 2.3 definiremos os códigos de Bose-Chaudhuri e Hocquenghem (BCH) e seus parâmetros. Esses códigos formam uma importante subclasse dos códigos cíclicos.

Na Seção 2.4 trataremos da distância mínima verdadeira dos códigos BCH e na Seção 2.5, do seu número de dígitos de informação.

Na Seção 2.6 utilizando os resultados anteriormente obtidos, mostraremos que os códigos BCH são assintoticamente ruins, mas apesar disso, seu algoritmo de decodificação apresentado na Seção 2.7 é eficiente e de fácil implementação, daí sua importância. Esse algoritmo é o algoritmo iterativo de Berlekamp-Massey.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

2.2 Códigos Cíclicos

Iniciaremos essa seção apresentando conceitos preliminares e propriedades fundamentais que caracterizam um código cíclico sobre $GF(q)$. Assim,

Definição 2.2.1 *Um código linear C (n,k) é dito um código cíclico se toda permutação cíclica de uma palavra-código resultar numa palavra-código.*

Assim, se $v = (v_0, v_1, \ldots, v_{n-1}) \in C$ temos que $v^{(i)} = (v_{n-i}, v_{n-i-1}, \ldots, v_{n-1}, v_0, v_1, \ldots, v_{n-i+1})$ é a i-ésima permutação cíclica de v e também pertence a C.

Denotaremos por F o corpo de Galois $GF(q)$. Dado $v = (v_0, v_1, \ldots, v_{n-1}) \in F^n$, associamos a cada vetor v o polinômio $v(x) = v_0 + v_1 x + \ldots + v_{n-1}x^{n-1} \in F[x]$, onde $F[x]$ denota o anel dos polinômios em x com coeficientes em F.

Definimos $R_n = F[x]/(x^n - 1)$, como sendo o anel formado pelas classes residuais de $F[x]$ módulo $x^n - 1$. Observe que os polinômios de grau $\leq n-1$ estão em diferentes classes e serão tomados como seus representantes. R_n pode ser visto como um espaço vetorial de dimensão n sobre F. Se $v(x) = v_0 + v_1 x + \ldots + v_{n-1}x^{n-1} \in F[x]$, temos que

$$\begin{aligned} xv(x) &= v_0 x + v_1 x^2 + \ldots + v_{n-1}x^n \in F[x] \\ &\equiv v_{n-1} + v_0 x + \ldots + v_{n-2}x^{n-1} \pmod{x^n - 1} \end{aligned}$$

que corresponde a uma permutação cíclica de v.

Definição 2.2.2 *Um ideal C de R_n é um subespaço linear de R_n tal que, se $c(x) \in C$ então $r(x)c(x) \in C$, $\forall r(x) \in R_n$. Alternativamente, podemos definir um ideal C de R_n como sendo um subespaço linear de R_n tal que se $c(x) \in C$, então $xc(x) \in C$.*

Tendo em vista o exposto acima e a Definição 2.2.2, podemos definir um código cíclico da seguinte forma.

Definição 2.2.3 *Um código cíclico de comprimento n é um ideal de R_n.*

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Mostraremos a seguir que todo ideal de R_n é principal. Assim, todo código cíclico é um ideal principal de R_n e portanto terá um polinômio gerador (isto é, $C = \langle g(x) \rangle$).

Teorema 2.2.1 *Seja C um ideal não nulo de R_n (isto é, C é um código cíclico de comprimento n). Então:*

a) Existe um único polinômio mônico $g(x) \in C$ tal que, se $f(x) \in C$, então $\partial f(x) \geq \partial g(x)$;

b) $C = \langle g(x) \rangle$;

c) $g(x)$ é um fator de $x^n - 1$;

d) $\forall c(x) \in C$, $c(x)$ pode ser escrito de maneira única como $c(x) = f(x)\, g(x)$ em $F[x]$, onde $f(x) \in F[x]$, $\partial f(x) < n - r$ e $r = \partial g(x)$. A dimensão de C é $k = n - r$. Assim, a mensagem $f(x)$ será codificada na palavra-código $f(x)\, g(x)$;

e) Se $g(x) = g_0 + g_1 x + \ldots + g_r x^r$, então C será gerado (como um subespaço de F^n) pelas linhas da matriz geradora G dada por:

$$G = \begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_r & & 0 \\ & g_0 & g_1 & & g_{r-1} & g_r & \\ & & & & & & \\ 0 & & g_0 & \cdots & \cdots & \cdots & g_r \end{bmatrix}$$

$$= \begin{bmatrix} g(x) & & & & \\ & xg(x) & & & \\ & & \ddots & & \\ & & & & x^{n-r-1} g(x) \end{bmatrix}$$

Demonstração:

a) Suponha $f(x), g(x) \in C$, polinômios mônicos de menor grau em C, tais que $\partial f(x) = \partial g(x) = r$. Então, $h(x) = f(x) - g(x) \in C$ e $\partial h(x) < r$, uma contradição. Assim $f(x) = g(x)$.

b) Suponha $c(x) \in C$ e $c(x) = q(x)\,g(x) + r(x)$ em R_n, onde $\partial r(x) < r = \partial g(x)$. Como $r(x) = c(x) - q(x)\,g(x) \in C$ (devido à linearidade de C), então $r(x) = 0$. Logo, $c(x) \in \langle g(x) \rangle$.

c) $x^n - 1 = h(x)\,g(x) + r(x)$ em $F[x]$, onde $\partial r(x) < r$. Em R_n, temos:

$$r(x) = -h(x)\,g(x) \in C$$

uma contradição, a menos que $r(x) \equiv 0$.

d), e) De (b), temos que para qualquer $c(x) \in C$, $c(x) = q(x)\,g(x)$ em R_n $(\partial c(x) < n)$.

Portanto,

$$\begin{aligned} c(x) &= q(x)\,g(x) + e(x)\,(x^n - 1) \ \text{(em } F[x]\text{)} \\ &= (q(x) + e(x)\,h(x))\,g(x) \ \text{(em } F[x]\text{)} \\ &= f(x)\,g(x) \ \text{(em } F[x]\text{)}, \end{aligned}$$

donde $\partial f(x) \leq n - r - 1$.

Assim, o código é formado pelos múltiplos de $g(x)$ por polinômios de grau menor ou igual a $n - r - 1$ em $F[x]$.

Como existe um total de $n - r$ múltiplos de $g(x)$ linearmente independentes, a saber, $g(x), xg(x), \ldots, x^{n-r-1}g(x)$, estes serão as linhas de G e o código terá dimensão $n - r$. □

Pelo Teorema 2.2.1 temos que se C for um código cíclico de comprimento n e com polinômio gerador $g(x)$ então $g(x)$ é um fator de $x^n - 1$, ou seja, $g(x) \mid x^n - 1$. Mostraremos agora o seguinte resultado

Proposição 2.2.1 *Seja $g(x)$ um polinômio tal que $\partial g \leq n - 1$, se $g(x)$ divide $x^n - 1$ então $g(x)$ é um polinômio gerador de um código cíclico de comprimento n.*

Demonstração: Suponha que um polinômio $g(x) \in F_q[x] \nmid (x^n - 1)$ divida $x^n - 1$. Devemos provar que $g(x)$ tem grau mínimo em $\langle g(x) \rangle$, isto é, ele é o gerador de um código cíclico. Suponha por absurdo que exista um polinômio $b(x) \in \langle g(x) \rangle$ tal que $\partial b < \partial g$. Então:

$$a(x)\, g(x) = d(x)\,(x^n - 1) + b(x)\,,$$

para algum $a(x)$, $d(x) \in F_q[x]$. Isto é, $b(x)$ deve ser o resto da divisão de um múltiplo de $g(x)$ por $x^n - 1$. Agora em $F_q[x]$, como $g(x) \mid (d(x)\,(x^n - 1) + b(x))$ e $g(x) \mid (x^n - 1)$ por hipótese, então $g(x) \mid b(x)$, o que é uma contradição, já que $\partial g > \partial b$. Portanto em $\langle g(x) \rangle$, $g(x)$ é o polinômio de menor grau e é o gerador de um código cíclico. □

Definição 2.2.4 *Se C é um código cíclico tal que $C = \langle g(x) \rangle$ e cujos parâmetros são $[n, k = n - r]$, onde $r = \partial g(x)$, então definimos*

$$h(x) = (x^n - 1)/g(x) = \sum_{i=0}^{k} h_i x^i, \quad (h_k \neq 0) \tag{2.1}$$

como o polinômio verificação de paridade de C.

Da definição do polinômio verificação de paridade $h(x)$ e do fato de que $g(x) | x^n - 1$, temos que $x^n - 1 = g(x)\, h(x)$, onde $\partial h = k$ e $h(x)$ tem a forma

$$h(x) = h_0 + h_1 x + \ldots + h_k x^k,$$

$h_0 \neq 0$ e $h_k = 1$.

Se $v(x)$ é um polinômio código em C,

$$v(x) = f(x)\, g(x)\,. \tag{2.2}$$

Multiplicando a Equação em (2.2) por $h(x)$ obtemos:

$$v(x)\, h(x) \;=\; f(x)\, g(x)\, h(x) = f(x)\,(x^n - 1) \tag{2.3}$$

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

Como $\partial f \leq k-1$, temos que os coeficientes de $x^k, x^{k+1}, \ldots, x^{n-1}$ são todos nulos em $x^n f(x) - f(x)$. Assim, no produto $v(x)\, h(x)$ da Equação em (2.3) temos que os coeficientes de $x^k, x^{k+1}, \ldots, x^{n-1}$ também serão nulos.

Obtemos então $n-k$ equações da forma

$$\sum_{i=0}^{n-1} h_i v_{n-j-i} = 0, \tag{2.4}$$

$1 \leq j \leq n-k$. Veremos que essas equações poderão ser tomadas como as equações de verificação de paridade do código C.

Se tomarmos $h^\star(x)$ como o polinômio recíproco de $h(x)$, $h^\star(x)$ será da forma

$$h^\star(x) = x^k h(x^{-1}) = h_k + h_{k-1}x + \ldots + h_0 x^k.$$

Como

$$x^n - 1 = g(x)\, h(x),$$

temos

$$\begin{aligned}
x^{-n} - 1 &= g(x^{-1})h(x^{-1}) = 0 \\
&\Rightarrow x^k(x^{-n} - 1) = x^k h(x^{-1})g(x^{-1}) \\
&\Rightarrow x^{n-k}(x^{k-n} - x^k) = h^\star(x)x^{n-k}g(x^{-1}) \\
&\Rightarrow 1 - x^n = h^\star(x)g^\star(x) \\
&\Rightarrow -(x^n - 1) = h^\star(x)g^\star(x)
\end{aligned}$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Portanto, como $h^*(x)$ tem grau k e divide $x^n - 1$, temos que ele gera um código cíclico $(n, n-k)$, cuja matriz geradora H pode ser dada por:

$$H = \begin{bmatrix} 0 & & & h_k & \dots & h_2 & h_1 & h_0 \\ & & h_k & & \cdots & h_1 & h_0 & \vdots \\ \vdots & & & & & & & \vdots \\ h_k & \cdots & \cdots & h_2 & h_1 & h_0 & & 0 \end{bmatrix} =$$

$$= \begin{bmatrix} & & & & & & \overleftarrow{h(x)} \\ & & & & & \overleftarrow{xh(x)} & \\ & & & \cdots & & & \\ x^{n-k-1}\overleftarrow{h(x)} & & & \cdots & & & \end{bmatrix}$$

O espaço linha de H gera portanto o código dual $C^{\perp}$ (de C), uma vez que, $\forall v \in C$, v é ortogonal à qualquer linha de H pelas $n-k$ Equações em (2.4). A matriz H é portanto a matriz verificação de paridade do código C e as Equações em (2.4) são as equações de verificação de paridade de C.

Com isso, obtemos o seguinte resultado:

Teorema 2.2.2 *Seja C um código cíclico (n,k) com polinômio gerador $g(x)$. Então $C^{\perp}(n, n-k)$, o dual de C, também é um código cíclico e é gerado por $h^*(x) = x^k h(x^{-1})$ onde $h(x) = (x^n - 1)/g(x)$.*

Definição 2.2.5 *Um código cíclico C sobre $GF(q)$ cujo comprimento é $n = q^m - 1$ é denominado código cíclico primitivo.*

Os códigos cíclicos primitivos sobre $GF(q)$ são determinados a partir da fatoração de $x^{q^m-1} - 1$ sobre $GF(q)$. Do Capítulo 1, Seção 1.9, temos que

$$x^{q^m-1} - 1 = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{q^m-1})$$

onde α é um elemento primitivo de $GF(q^m)$.

O polinômio $x^{q^m-1}-1$ pode ser fatorado sobre $GF(q)$ como

$$x^{q^m-1}-1 = M_1(x)\, M_2(x) \ldots M_s(x),$$

onde os $M_i(x)$, $1 \leq i \leq s$ são os polinômios minimais sobre $GF(q)$ dos elementos não nulos de $GF(q^m)$.

Segue então que existem

$$\binom{s}{0} + \binom{s}{1} + \cdots + \binom{s}{s} = 2^s$$

códigos cíclicos primitivos, incluindo o código universal gerado por $g(x) = 1$ e o código formado apenas pela palavra toda nula, ou seja, existem $2^s - 2$ códigos cíclicos não triviais sobre F_q de comprimento $n = q^m - 1$.

O polinômio gerador de um código cíclico de comprimento n sobre $GF(q)$ é um fator de $x^n - 1$. Assumindo n e q relativamente primos, definimos m, a *ordem multiplicativa* de q módulo n, como sendo o menor inteiro tal que $n \mid q^m - 1$ (assim, $x^n - 1 \mid x^{q^m-1} - 1$, mas $x^n - 1 \nmid x^{q^s-1} - 1$ para $0 < s < m$) e assim, concluímos que o menor corpo de característica p (onde $q = p^a$, para algum $a > 0$) que contém os zeros de $x^n - 1$ (as raízes n-ésimas da unidade) é a extensão $GF(q^m)$.

Sejam $\alpha_0, \alpha_1, \ldots, \alpha_{n-1} \in GF(q^m)$ as raízes n-ésimas da unidade distintas entre si.

Então

$$x^n - 1 = \prod_{i=0}^{n-1} (x - \alpha_i)$$

e $GF(q^m)$ é dito o corpo de decomposição de $x^n - 1$.

A classe ciclotômica mod n que contém s é denotada por C_s e é dada por:

$$C_s = \left\{ s, sq, sq^2, \ldots, sq^{m_s-1} \right\},$$

onde m_s é o menor inteiro positivo tal que $sq^{m_s} \equiv s \bmod n$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Em geral, escolhemos s como sendo o menor inteiro em C_s para ser o representante dessa classe ciclotômica mod n. Assim, teremos $\{0, 1, \ldots, n-1\} = \bigcup_s C_s$, onde s percorre o conjunto dos representantes das classes mod n. Note que o número de elementos de C_1 é m_1.

Se $M^{(s)}(x)$ é o polinômio minimal de α^s, onde α é um elemento primitivo de $GF(q^m)$, temos que $M^{(s)}(x) = \prod_{i \in C_s} (x - \alpha^i)$, é o polinômio mônico de menor grau sobre $GF(q)$ tendo α^s como raiz. Então

$$x^n - 1 = \prod_s M^{(s)}(x),$$

onde s percorre o conjunto de representantes das classes mod n. Obtemos assim a fatoração de $x^n - 1$ em polinômios irredutíveis sobre $GF(q)$.

Teorema 2.2.3 *Seja $g(x)$ o polinômio gerador de um código cíclico primitivo C sobre $GF(q)$ com zeros $\alpha_1, \alpha_2, \ldots, \alpha_r$ em $GF(q^m)$. Um polinômio $v(x)$ sobre $GF(q)$, é um polinômio código em C se, e somente se, $c(\alpha_1) = c(\alpha_2) = \ldots = c(\alpha_r) = 0$, onde $c(\alpha_j)$ é avaliado em $GF(q^m)$.*

Demonstração: ($\Rightarrow$) Suponha que $c(x)$ seja um polinômio código. Então, $c(x) = a(x)\, g(x)$.

Segue que $c(\alpha_j) = a(\alpha_j)\, g(\alpha_j) = 0$, para $1 \leq j \leq r$.

($\Leftarrow$) Suponha que $c(\alpha_j) = 0$, $0 \leq j \leq r$.

Fazendo a divisão de $c(x)$ por $M_j(x)$, onde $M_j(x)$ é o polinômio minimal de α_j temos que

$$c(x) = a_j(x)\, M_j(x) + s_j(x),$$

onde $\partial s_j < \partial M_j$ para $1 \leq j \leq r$. Como

$$0 = c(\alpha_j) = a_j(\alpha_j)\, M_j(\alpha_j) + s_j(\alpha_j)$$

temos que $s_j(\alpha_j) = 0$. Mas como M_j é o polinômio minimal de α_j, segue que $s_j(x) \equiv 0$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Logo, $c(x)$ é múltiplo do mmc$\{M_1(x), \ldots, M_r(x)\} = g(x)$ e portanto $c(x)$ é palavra-código. □

Com esse resultado, construiremos uma matriz verificação de paridade em $GF(q^m)$, para um código cíclico sobre $GF(q)$.

Assim, seja C um código cíclico de comprimento n, cujos zeros do polinômio gerador $g(x)$ sejam $\alpha_1, \alpha_2, \ldots, \alpha_r$ em $GF(q^m)$. Então, se $v = (v_0, v_1, \ldots, v_{n-1})$ é uma palavra-código em C, temos que $vH^t = 0$, onde H é dada por

$$H = \begin{bmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{n-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdots & \alpha_2^{n-1} \\ \vdots & \vdots & & & \vdots \\ 1 & \alpha_r & \alpha_r^2 & & \alpha_r^{n-1} \end{bmatrix}.$$

Com essa representação, passemos ao caso dos códigos de Hamming binários introduzidos na Seção 1.8, Capítulo 1.

Se $\mathcal{H}_r$ for um código de Hamming binário, seus parâmetros são dados por $[n = 2^r - 1, k = 2^r - 1 - r, d_{\min} = 3]$.

Sabemos que a matriz verificação de paridade H desse código é formada por todas as r-uplas não nulas distintas, dispostas como colunas, ou seja, um total de $2^r - 1$ colunas.

Se α é um elemento primitivo de $GF(2^r)$, então $1, \alpha, \alpha^2, \ldots, \alpha^{2^r-2}$ são distintos e podem ser representados por r-uplas distintas e não nulas sobre $GF(2)$.

Assim, tomaremos H como sendo

$$H = \left(1, \alpha, \alpha^2, \ldots, \alpha^{2^r-2}\right)$$

onde cada entrada corresponde a um vetor coluna de comprimento r.

Um vetor

$$c = (c_0, c_1, \ldots, c_{n-1}) \in \mathcal{H}_r \Leftrightarrow$$

$$
\begin{aligned}
&\Leftrightarrow \quad Hc^t = 0 \\
&\Leftrightarrow \quad \sum_{i=0}^{n-1} c_i \alpha^i = 0 \\
&\Leftrightarrow \quad c(\alpha) = 0
\end{aligned}
$$

onde $c(x) = c_0 + c_1 x + \ldots + c_{n-1} x^{n-1}$.

Da propriedade M2) dos polinômios minimais dada na Seção 1.9 do Capítulo 1, temos que $c(x) \in \mathcal{H}_r \Leftrightarrow M^{(1)}(x) \mid c(x)$, onde $M^{(1)}(x)$ é um polinômio minimal. Em outras palavras,

Teorema 2.2.4 *Os códigos de Hamming como definimos acima são códigos cíclicos tais que* $\mathcal{H}_r = \langle g(x) \rangle = \left\langle M^{(1)}(x) \right\rangle$, onde $M^{(1)}(x)$ é o polinômio minimal de $\alpha \in GF(2^r)$.

Então a matriz geradora de $\mathcal{H}_r$ será da forma:

$$
G = \begin{bmatrix}
M^{(1)}(x) & & & \\
& x M^{(1)}(x) & & \\
& \ldots\ldots\ldots\ldots & & \\
& & & x^{n-r-1} M^{(1)}(x)
\end{bmatrix}
$$

Exemplo 2.2.1 *Seja* $\mathcal{H}_4$ *o código de Hamming sobre* $GF(2^4)$ *com parâmetros* $[n = 15, k = 11, d = 3]$ *e* α *um elemento primitivo de* $GF(2^4)$ *que é raiz de* $\pi(x) = x^4 + x + 1$. *Os elementos de* $GF(16)$ *podem ser descritos de várias maneiras conforme apresentado na Tabela 2.1.*

Como uma 4-upla	Como um polinômio	Como uma potência de α
0000	0	0
1000	1	1
0100	α	α
0010	α^2	α^2
0001	α^3	α^3
1100	$1+\alpha$	α^4
0110	$\alpha+\alpha^2$	α^5
0011	$\alpha^2+\alpha^3$	α^6
1101	$1+\alpha+\alpha^3$	α^7
0101	$1+\alpha^2$	α^8
0111	$\alpha+\alpha^3$	α^9
1110	$1+\alpha+\alpha^2$	α^{10}
0111	$\alpha+\alpha^2+\alpha^3$	α^{11}
1111	$1+\alpha+\alpha^2+\alpha^3$	α^{12}
1011	$1+\alpha^2+\alpha^3$	α^{13}
1001	$1+\alpha^3$	α^{14}

Tabela 2.1 - $GF(2^4)$ **gerado por** α **tal que** $\alpha^4+\alpha+1=0$

A matriz verificação de paridade do código $\mathcal{H}_4$ *é dada por:*

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \cdots & \alpha^{15} \end{bmatrix}$$

e onde de acordo com a Tabela 2.1, cada entrada α^i, $0 \leq i \leq 15$, *pode ser escrita como uma 4-upla sobre* $GF(2)$ *em forma de coluna.*

Sua matriz geradora é dada por

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

Observe também que da Tabela 2.1 obtemos o polinômio minimal de cada elemento de $GF(16)$. *Assim:*

elemento	polinômio minimal
0	x
1	$x+1$
$\alpha, \alpha^2, \alpha^4, \alpha^8$	x^4+x+1
$\alpha^3, \alpha^6, \alpha^9, \alpha^{12}$	$x^4+x^3+x^2+x+1$
α^5, α^{10}	x^2+x+1
$\alpha^7, \alpha^{11}, \alpha^{13}, \alpha^{14}$	x^4+x^3+1

Tabela 2.2 - Polinômios Minimais

Portanto $x^{15}-1 = (x+1)(x^2+x+1)(x^4+x+1)(x^4+x^3+1)(x^4+x^3+x^2+x+1)$, *donde temos que existem* $2^5-2=30$ *códigos cíclicos (não triviais) de comprimento* 15.

2.3 Códigos BCH

Nesta seção descreveremos códigos BCH. Esses códigos tem capacidade de correção de erros múltiplos. Isso implicará em códigos com altas distâncias mínimas porém com taxas baixas. Apesar disso, os códigos BCH são importantes, pois sua decodificação é feita de forma muito eficiente, através do uso do Algoritmo de Berlekamp-Massey.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Iniciaremos agora definindo um código BCH utilizando o termo "distância de projeto" o qual será justificado mais adiante.

Definição 2.3.1 *Um código cíclico de comprimento n sobre $GF(q)$, é um código BCH com distância de projeto δ, se para algum inteiro $l \geq 0$, o seu polinômio gerador for dado por*

$$g(x) = mmc\left\{M^{(l)}(x), M^{(l+1)}(x), \ldots, M^{(l+\delta-2)}(x)\right\}$$

onde $M^{(i)}(x)$ é o polinômio minimal de α^i, $l \leq i \leq l+\delta-2$, onde α é uma raiz primitiva n-ésima da unidade, em alguma extensão $GF(q^m)$ contendo $GF(q)$.

Na Definição 2.3.1, note que $mdc(n, q) = 1$, para que $n \mid q^m - 1$, para algum $m \geq 1$.

Assim, $g(x)$ é o polinômio mônico de menor grau sobre $GF(q)$, tendo $\alpha^l, \alpha^{l+1}, \ldots, \alpha^{l+\delta-2}$ como raízes. Portanto, $c = (c_0, c_1, \ldots, c_{n-1}) \in GF(q)^n$ será uma palavra-código se, e somente se, o polinômio $c(x) = c_0 + c_1 x + \ldots + c_{n-1}x^{n-1}$ tem $\alpha^l, \alpha^{l+1}, \ldots, \alpha^{l+\delta-2}$ como raízes.

A matriz verificação de paridade de um código BCH é dada por

$$H = \begin{pmatrix} 1 & \alpha^l & \alpha^{2l} & \cdots & \alpha^{(n-1)l} \\ 1 & \alpha^{l+1} & \alpha^{2(l+1)} & & \alpha^{(n-1)(l+1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & \alpha^{l+\delta-2} & \alpha^{2(l+\delta-2)} & \cdots & \alpha^{(n-1)(l+\delta-2)} \end{pmatrix},$$

onde cada entrada corresponde a uma coluna de m elementos sobre $GF(q)$. Obtemos assim um conjunto de $m(\delta-1)$ linhas em H não necessariamente linearmente independente em $GF(q)$.

Com relação à definição de códigos BCH, temos que os elementos $\alpha^l, \alpha^{l+1}, \ldots, \alpha^{l+\delta-2}$ devem ser todos distintos entre si. Usualmente, toma-se $l = 1$ (código BCH no sentido estrito). Se $n = q^m - 1$, isto é, se α for um elemento primitivo em $GF(q^m)$, então o código BCH é denominado primitivo.

O teorema a seguir nos dá os limitantes para os parâmetros do código BCH e justifica o termo "distância de projeto" δ do código, que é um limitante para a distância mínima.

Teorema 2.3.1 *Um código BCH sobre $GF(q)$ de comprimento n e distância de projeto δ tem distância mínima $d \geq \delta$ e dimensão $k \geq n - m(\delta - 1)$.*

Demonstração: Seja $v = (v_0, v_1, \ldots, v_{n-1})$ uma palavra-código com peso de Hamming $w \leq \delta - 1$ e suponha que $v_{j_1}, v_{j_2}, \ldots, v_{j_w}$ sejam as componentes não nulas de v. Calculando a síndrome, obtemos:

$$S = vH^t = (v_{j_1}, v_{j_2}, \ldots, v_{j_w}) \begin{pmatrix} (\alpha^l)^{j_1} & (\alpha^{l+1})^{j_1} & \cdots & (\alpha^{l+\delta-2})^{j_1} \\ (\alpha^l)^{j_2} & \left(\alpha^{l+1}\right)^{j_2} & & (\alpha^{l+\delta-2})^{j_2} \\ \cdots & \cdots & \cdots & \cdots \\ (\alpha^l)^{j_w} & (\alpha^{l+1})^{j_w} & \cdots & (\alpha^{(l+\delta-2)})^{j_w} \end{pmatrix} = 0.$$

Portanto,

$$(v_{j_1}, v_{j_2}, \ldots, v_{j_w}) \begin{pmatrix} (\alpha^{j_1})^l & (\alpha^{j_1})^{l+1} & \cdots & (\alpha^{j_1})^{l+\delta-2} \\ (\alpha^{j_2})^l & (\alpha^{j_2})^{l+1} & & (\alpha^{j_2})^{l+\delta-2} \\ \cdots & \cdots & \cdots & \cdots \\ (\alpha^{j_w})^l & (\alpha^{j_w})^{l+1} & \cdots & (\alpha^{j_w})^{l+\delta-2} \end{pmatrix} = 0.$$

Esta última relação implica a seguinte igualdade:

$$(v_{j_1}, v_{j_2}, \ldots, v_{j_w}) \begin{pmatrix} (\alpha^{j_1})^l & (\alpha^{j_1})^{l+1} & \cdots & (\alpha^{j_1})^{l+w-1} \\ (\alpha^{j_2})^l & (\alpha^{j_2})^{l+1} & & (\alpha^{j_2})^{l+w-1} \\ \vdots & & \cdots & \vdots \\ (\alpha^{j_w})^l & (\alpha^{j_w})^{l+1} & \cdots & (\alpha^{j_w})^{l+w-1} \end{pmatrix}_{w \times w} = 0.$$

Essa igualdade será satisfeita se o determinante da matriz quadrada $w \times w$ acima for

nulo. Colocando os termos comuns em evidência, obtemos:

$$\alpha^{(j_1+j_2+\cdots+j_w)t}\begin{vmatrix} 1 & \alpha^{j_1} & \cdots & \left(\alpha^{j_1}\right)^{w-1} \\ 1 & \alpha^{j_2} & & \left(\alpha^{j_2}\right)^{w-1} \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \alpha^{j_w} & \cdots & \left(\alpha^{j_w}\right)^{w-1} \end{vmatrix} = 0$$

O determinante nesta última igualdade é um determinante de Vandermonde e portanto é não nulo. Em conclusão, não existe palavra-código com peso de Hamming menor que δ e a distância mínima de um código BCH é pelo menos δ (a distância de projeto δ é chamada de limitante do BCH).

Como os códigos BCH são cíclicos, temos $k = n - \partial g(x)$, o que implica que $\partial g(x) = n - k$.

Pela Definição 2.3.1 e do fato de que $\partial M^{(i)}(x) \leq m$ temos $\partial g(x) = n - k \leq m(\delta - 1)$. □

Dizemos que um código BCH é maximalmente redundante se $n - k = m(\delta - 1)$. Códigos Reed-Solomon são um exemplo de códigos BCH maximalmente redundantes. Adiantamos a definição dos códigos Reed-Solomon aos quais voltaremos mais adiante.

Definição 2.3.2 *Um código Reed-Solomon (ou código RS) é um código BCH de comprimento $n = q - 1$ sobre $GF(q)$ que tem como polinômio gerador*

$$g(x) = \prod_{i=1}^{d-1} \left(x - \alpha^i\right),$$

onde α é um elemento primitivo de $GF(q)$.

Sabemos pelo Limitante de Singleton que para um código linear com parâmetros (n, k, d) onde d é a distância mínima, temos $d \leq n - k + 1$ e pelo Limitante do BCH, $d \geq \delta$. Assim $n - k + 1 \geq d \geq \delta$ e como a dimensão de um código RS é dada por

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

$k = n - \delta + 1$ (de acordo com o Teorema 2.3.1) segue que para um código RS,

$$n - (n - \delta + 1) + 1 \geq d \geq \delta$$

donde concluímos que $d = \delta$. A matriz verificação de paridade de um código RS terá a forma

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & (\alpha^2)^2 & \cdots & (\alpha^2)^{n-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & \alpha^{d-1} & (\alpha^{d-1})^2 & \cdots & (\alpha^{d-1})^{n-1} \end{bmatrix}$$

2.4 A Distância Mínima Verdadeira de um Código BCH

Na Seção 2.3, ao descrevermos um código BCH com distância de projeto δ, apresentamos um limitante inferior para a sua distância mínima d. Assim, pelo Teorema 2.3.1, temos que $d \geq \delta$. Nesta seção, analisaremos com mais cuidado esse parâmetro, com a finalidade de apontar o valor verdadeiro para a distância mínima de um código BCH.

Primeiramente, introduziremos alguma notação:

C denotará um código BCH primitivo sobre $GF(q)$, com distância de projeto δ, sendo $\alpha \in GF(q^m)$ uma raiz primitiva n-ésima de unidade, onde m é a ordem multiplicativa de $q \bmod n$, ou seja, m é o menor inteiro tal que n divide $q^m - 1$.

Dado um vetor $a = (a_0, \ldots, a_{n-1})$ em $GF(q)^n$, com peso de Hamming w, definimos:

$$a(x) = \sum_{i=0}^{n-1} a_i x^i$$

e $a_{i_1}, \ldots, a_{i_w}$ representam as componentes não nulas do vetor a. Definimos os localizadores de a como sendo $X_r = \alpha^{i_r}$, $r = 1, \ldots, w$.

O polinômio localizador de a é dado por:

$$\sigma(z) = \prod_{r=1}^{w} (1 - X_r z) = \sum_{r=0}^{w} \sigma_r z^r, \ \sigma_0 = 1. \tag{2.5}$$

Definimos as somas de potências como sendo

$$A_l = a(\alpha^l) = \sum_{r=1}^{w} Y_r X_r^l, \quad l = 0, 1, \ldots \tag{2.6}$$

onde $Y_r = a_{i_r}$, $r = 1, \ldots, w$. Em particular, se $a \in C$, então $A_l = 0$, para $1 \leq l \leq \delta - 1$.

Os seguintes resultados serão úteis:

Identidades de Newton:

Sejam $X_1, X_2, \ldots, X_w$ indeterminadas e $\sigma(z) = \prod_{i=1}^{w} (1 - X_i z) = \sum_{i=0}^{w} \sigma_i z^i$, $\sigma_0 = 1$, onde σ_i é uma função simétrica elementar de X_i, dada por

$$\begin{aligned}
&\sigma_0 = 1 \\
&\sigma_1 = -(X_1 + X_2 + \ldots + X_w) \\
&\sigma_2 = X_1X_2 + X_1X_3 + \ldots + X_{w-1}X_w \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
&\sigma_w = (-1)^w X_1X_2X_3 \ldots X_w \text{ e } \sigma_i = 0 \text{ se } i > w.
\end{aligned}$$

a) Definimos $P_i = \sum_{r=1}^{w} (X_r)^i$, $i = 1, 2, \ldots, w, \ldots$. Então:

$$\begin{aligned}
&P_1 + \sigma_1 = 0 \\
&P_2 + \sigma_1 P_1 + 2\sigma_2 = 0 \\
&\vdots \\
&P_w + \sigma_1 P_{w-1} + \ldots + \sigma_{w-1} P_1 + w\sigma_w = 0
\end{aligned}$$

e para $i > w$, $P_i + \sigma_1 P_{i-1} + \ldots + \sigma_w P_{i-w} = 0$.

Este resultado pode ser provado usando o fato de que $\sigma(z) P(z) + z\sigma'(z) = 0$, onde $P(z) = \sum_{i=1}^{\infty} P_i z^i$.

b) Suponha que X_i pertença a um corpo de característica p. Temos que $P_l = 0$ para $1 \leq l \leq s$ (s fixo) se $\sigma_l = 0$ para todo l (no intervalo $1 \leq l \leq s$) que não seja divisível por p.

c) Seja $\{\beta_0, \ldots, \beta_{m-1}\}$ uma base para $GF(q^m)$ sobre $GF(q)$. Então a matriz

$$A = \begin{bmatrix} \beta_0 & \beta_0^q & \cdots & \beta_0^{q^{m-1}} \\ \beta_1 & \beta_1^q & \cdots & \beta_1^{q^{m-1}} \\ \vdots & \vdots & \cdots & \vdots \\ \beta_{m-1} & \beta_{m-1}^q & \cdots & \beta_{m-1}^{q^{m-1}} \end{bmatrix}$$

é invertível.

Proposição 2.4.1 *Seja V um subespaço vetorial h-dimensional de $GF(q^m)$. Então, $\prod_{\beta \in V} (z - \beta) = z^{q^h} + l_{h-1} z^{q^{h-1}} + \ldots + l_0 z$. Este polinômio, denotado por $L(z)$, recebe o nome de polinômio linearizador sobre $GF(q^m)$.*

Demonstração: Seja $\{\beta_0, \beta_1, \ldots, \beta_{h-1}\}$, $\beta_i \in GF(q^m)$, $i = 0, 1, \ldots, h-1$, uma base do espaço vetorial V e a matriz

$$A = \begin{bmatrix} \beta_0 & \beta_0^q & \beta_0^{q^2} & \cdots & \beta_0^{q^{h-1}} \\ \beta_1 & \beta_1^q & \beta_1^{q^2} & \cdots & \beta_1^{q^{h-1}} \\ \vdots & \vdots & & \cdots & \vdots \\ \beta_{h-1} & \beta_{h-1}^q & \beta_{h-1}^{q^2} & \cdots & \beta_{h-1}^{q^{h-1}} \end{bmatrix}. \tag{2.7}$$

Temos (por c)) que A é invertível e então existe uma única solução $l_0, l_1, \ldots, l_{h-1}$, em $GF(q^m)$ para as equações

$$\beta_i^{q^h} + \sum_{j=0}^{h-1} l_j \beta_i^{q^j} = 0, \quad i = 0, 1, \ldots, h-1 \tag{2.8}$$

e portanto $\beta_0, \beta_1, \ldots, \beta_{h-1}$ são zeros do polinômio

$$L(z) = z^{q^h} + \sum_{j=0}^{h-1} l_j z^{q^j}.$$

Note que se

$$\gamma = a_0\beta_0 + a_1\beta_1 + \ldots + a_{h-1}\beta_{h-1},$$

onde $a_i \in GF(q)$, $0 \leq i \leq h-1$, temos

$$\begin{aligned} L(\gamma) &= \gamma^{q^h} + \sum_{j=0}^{h-1} l_j \gamma^{q^j} = \\ &= (a_0\beta_0 + \ldots + a_{h-1}\beta_{h-1})^{q^h} + \sum_{j=0}^{h-1} l_j (a_0\beta_0 + \ldots + a_{h-1}\beta_{h-1})^{q^j} = \\ &= a_0 \left(\beta_0^{q^h} + \sum_{j=0}^{h-1} l_j \beta_0^{q^j} \right) + \ldots + a_{h-1} \left(\beta_{h-1}^{q^h} + \sum_{j=0}^{h-1} l_j \beta_{h-1}^{q^j} \right) \end{aligned}$$

e por (2.8) temos que $L(\gamma) = 0$.

Portanto, para todo $\gamma \in V$, γ é raiz de $L(z) = \prod_{\beta \in V} (z - \beta)$. □

Lema 2.4.1 *Seja $\sigma(z) = \sum_{i=0}^{w} \sigma_i z^i$ um polinômio sobre $GF(q^m)$. Então $\sigma(z)$ é o polinômio localizador de uma palavra-código formada por 0's e 1's num código BCH sobre $GF(q)$ com distância de projeto δ se, e somente se,*

i) Os zeros de $\sigma(z)$ são raízes n-ésimas da unidade e distintas entre si;

ii) $\sigma_i = 0$ para todo i (no intervalo $1 \leq i \leq \delta - 1$) que não seja divisível por p, onde q é uma potência do número primo p.

Demonstração: ($\Rightarrow$) Seja a uma palavra-código de 0's e 1's em um código BCH sobre $GF(q)$ com distância de projeto δ e seja $\sigma(z)$ seu polinômio localizador.

Seja $\alpha \in GF(q^m)$ uma raiz primitiva n-ésima da unidade.

Os localizadores de a são dados por $X_1 = \alpha^{i_1}, X_2 = \alpha^{i_2}, \ldots, X_w = \alpha^{i_w}$ e como

$$\sigma(z) = \prod_{i=1}^{w} (1 - X_i z),$$

temos que X_i^{-1} são raízes de $\sigma(z)$, para $1 \leq i \leq w$ e a condição i) segue do fato de que $(\alpha^{-i_k})^n = (\alpha^n)^{-i_k} = 1^{-i_k} = 1$. Agora, por definição, $A_l = a(\alpha^l) = \sum_{r=1}^{w} Y_r X_r^l$ para $1 \leq l \leq \delta - 1$. Como a é palavra-código, segue que $a(\alpha^l) = 0$ se $1 \leq l \leq \delta - 1$ e assim temos que

$$A_l = \sum_{r=1}^{w} Y_r X_r^l = \sum_{r=1}^{w} X_r^l = P_l = 0$$

Portanto, $P_l = 0$ e pelo resultado b), temos que $\sigma_l = 0$ no intervalo $1 \leq l \leq \delta - 1$ se $p \nmid l$, o que verifica a condição ii).

($\Leftarrow$) Suponha que os zeros de $\sigma(z)$ sejam raízes n-ésimas da unidade e $\sigma_l = 0$, para todo l, $1 \leq l \leq \delta - 1$ que não seja divisível por p.

Sejam $X_1, X_2, \ldots, X_w$ os recíprocos dos zeros de $\sigma(z) = \prod_{i=1}^{w} (1 - X_i z)$.

Se a é um vetor com 1's nessas localizações, então $A_l = P_l = 0$. Assim, $a(\alpha^l) = 0$ e portanto a pertence ao código. □

Teorema 2.4.1 *Um código BCH de comprimento $n = q^m - 1$ e distância de projeto $\delta = q^h - 1$ sobre $GF(q)$ tem distância mínima verdadeira δ.*

Demonstração: Seja C um código BCH com distância de projeto δ. Pelo Teorema 2.3.1, temos que sua distância mínima $d \geq \delta$.

Para provarmos que $d = \delta$ basta encontrarmos uma palavra-código de peso δ em C.

Para isso, seja V um subespaço vetorial h-dimensional de $GF(q^m)$ e $L(z) = \prod_{\beta \in V} (z - \beta)$. Pela Proposição 2.4.1 sabemos que

$$L(z) = z^{q^h} + l_{h-1} z^{q^{h-1}} + \ldots + l_0 z.$$

Considere um vetor $a \in C$ que tenha como localizadores todos os $\beta \in V$ não nulos e tenha 1's nessas localizações.

Por definição, $\sigma(z) = \prod_{\substack{\beta \neq 0 \\ \beta \in V}} (1 - \beta z)$ e portanto,

$$\sigma(z) = z^{q^h} L(z^{-1}).$$

Assim

$$\sigma(z) = 1 + l_{h-1} z^{q^h - q^{h-1}} + \ldots + l_1 z^{q^h - q} + l_0 z^{q^h - 1}.$$

Os coeficientes σ_i que não aparecem, isto é, são iguais a zero, são aqueles para os quais $i \neq q^h - q^j$, $j = 0, 1, 2, \ldots, h$ e concluímos pelo Lema 2.4.1 que a é uma palavra-código de 0's e 1's e assim encontramos uma palavra-código de peso $q^h - 1 = \delta$ em C. □

Seja C um código BCH com distância de projeto δ. O seu polinômio gerador é dado por $g(x) = \text{mmc}\left\{M^{(l)}(x), M^{(l+1)}(x), \ldots, M^{(l+\delta-2)}(x)\right\}$, onde l é um inteiro positivo e $M^{(i)}(x)$ é o polinômio minimal de α^i, $l \leq i \leq l + \delta - 2$.

Seja δ_0, tal que $\delta \leq \delta_0$, a distância de projeto de um código BCH primitivo C_0, tal que C_0 tenha polinômio gerador $g_0(x) = \text{mmc}\left\{M^{(l)}(x), M^{(l+1)}(x), \ldots, M^{(l+\delta_0-2)}(x)\right\}$.

Temos então que $g(x) | g_0(x)$ e assim $\langle g_0(x) \rangle \subseteq \langle g(x) \rangle$. Se C_0 e C são códigos quaisquer tais que $C_0 \subseteq C$, então a distância mínima do código C_0, que denotaremos por d_0, é maior ou igual a d, a distância mínima do código C.

Portanto, um código BCH com distância de projeto δ contém um código BCH com distância de projeto δ_0 se, e somente se, $\delta \leq \delta_0$. Isso caracteriza os códigos BCH como sendo encaixados.

Apresentaremos a seguir um limitante superior para a distância mínima de um código BCH primitivo.

Teorema 2.4.2 *A distância mínima d de um código BCH primitivo C com distância de projeto δ sobre $GF(q)$ é no máximo $q\delta - 1$.*

Demonstração: Para algum inteiro positivo h, temos:

$$q^{h-1} \leq \delta < \delta_0 = q^h - 1.$$

Decorre do fato de os códigos BCH serem encaixados que δ_0 é a distância de projeto de um código BCH C_0 tal que $C_0 \subset C$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Multiplicando $q^{h-1} \leq \delta < q^h - 1$ por q obtemos:

$$q^h \leq q\delta < q(q^h - 1)$$

ou seja,

$$q^h - 1 \leq q\delta - 1 < q(q^h - 1) - 1.$$

Assim, $d \leq d_0 \leq q\delta - 1$. □

2.5 O Número de Dígitos de Informação de um Código BCH

Suponha que C seja um código BCH sobre $GF(q)$ de comprimento $n = q^m - 1$ e distância de projeto δ. Pelo Teorema 2.3.1, o número de símbolos de informação (que nesta seção denotaremos por $I(n, \delta)$) é ao menos $n - m(\delta - 1)$, mas o número preciso depende do grau do polinômio gerador $g(x)$ do código C, ou seja,

$$k = I(n, \delta) = n - \partial g(x),$$

e assim, para sabermos exatamente o valor de $I(n, \delta)$, precisamos saber o grau do polinômio gerador $g(x)$ de C.

Seja então $g(x)$ o polinômio mônico de menor grau tendo $\alpha, \alpha^2, \ldots, \alpha^{\delta-1}$ como raízes, onde α é um elemento primitivo de $GF(q^m)$. Se α^i é raiz de $g(x)$, então todos os seus conjugados, dados por $\alpha^{iq}, \alpha^{iq^2}, \ldots$ também o são. Denotaremos por m_i o menor inteiro tal que $iq^{m_i} \equiv i \pmod{q^m - 1}$, ou seja, m_i representa o número de diferentes conjugados de α^i (m_i representa o número de elementos de uma classe ciclotômica contendo i).

Para encontrarmos m_i, basta escrever i na base q como

$$i = i_{m-1}q^{m-1} + i_{m-2}q^{m-2} + \ldots + i_1 q + i_0 \tag{2.9}$$

onde $0 \leq i_v \leq q-1$.

Então

$$qi \equiv i_{m-2}q^{m-1} + \ldots + i_1 q^2 + i_0 q + i_{m-1} \pmod{q^m - 1},$$

ou seja, ao fazermos o produto da Equação em (2.9) por q, produzimos uma permutação cíclica da m-upla inicial e obtemos $(i_{m-2}, i_{m-3}, \ldots, i_1, i_0, i_{m-1})$.

Lema 2.5.1 *O grau de $g(x)$ é dado pelo número dos inteiros i, com $1 \leq i \leq q^m - 1$, tais que alguma permutação cíclica da expansão q-ária de i (escrita como uma m-upla) é menor ou igual a $\delta - 1$.*

Demonstração: Esse resultado segue imediatamente do fato de que α^i é raiz de $g(x)$ se, e somente se, α^i tem como conjugado α^{iq^r} com $1 \leq iq^r \leq \delta - 1$. □

Doravante assumiremos a distância de projeto do código C como sendo $\delta = q^\lambda$ e encontraremos um limitante superior para $I(n, \delta)$ quando $n = q^m - 1$ e $\delta = q^\lambda$.

Veja que

$$\frac{\delta}{n} = \frac{q^\lambda}{q^m - 1} \sim q^{-r},$$

quando $\lambda \to \infty$ e onde $r = m - \lambda$.

Definimos uma corrida como sendo uma sequência maximal de zeros consecutivos em uma m-upla $i_0, i_1, \ldots, i_{m-1}$. A corrida pode ser direta ou circular.

Uma corrida é dita de comprimento l, se há uma sequência consecutiva de l zeros nas posições $i_k, i_{k+1}, \ldots, i_{k+l-1}$, com $i_{k-1} \neq 0$ e $i_{k+l} \neq 0$, de uma m-upla. Caso os dígitos finais de uma m-upla sejam zeros e os primeiros também, temos uma corrida circular. Caso contrário, teremos uma corrida direta.

Por exemplo, na 9-upla (001000100) temos duas corridas diretas de comprimento 2, uma corrida direta de comprimento 3 e uma corrida circular de comprimento 4.

Do Lema 2.5.1 e do exposto acima temos:

Lema 2.5.2 *O grau de $g(x)$ é dado pelo número de inteiros i, $1 \leq i \leq q^m - 1$, tais que a expansão q-ária de i (escrita como uma m-upla) contém uma corrida de comprimento maior ou igual a r.*

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Demonstração: Escrevendo i na base q temos:

$$i = i_{m-1}q^{m-1} + i_{m-2}q^{m-2} + \ldots + i_1 q + i_0.$$

Note agora que $i \leq q^\lambda - 1$ se, e somente se, $i_\lambda = i_{\lambda+1} = \ldots = i_{m-1} = 0$, pois $0 \leq i_j \leq q-1$, $\forall j \in \{0, 1, \ldots m-1\}$.

Segue então que $i \leq q^\lambda - 1$ se, e somente se, a m-upla $(i_{m-1}, \ldots, i_1, i_0)$ começa com uma corrida de comprimento no mínimo $r = m - \lambda$. Portanto, alguma permutação cíclica da expansão q-ária de i será menor ou igual a $\delta - 1$, se e somente se, essa expansão q-ária contiver uma corrida de comprimento maior ou igual a r. □

Suponhamos agora r fixo. Denotaremos por $s(m)$ o número de m-uplas q-árias que contenham uma corrida direta de comprimento maior ou igual a r. Denotaremos por $c(m)$ o número de m-uplas q-árias que contenham uma corrida circular de comprimento maior ou igual a r, mas não possuam corridas diretas de comprimento maior ou igual a r.

Pelo Lema 2.5.2 temos que

$$\partial g(x) = s(m) + c(m) - 1,$$

onde o último termo vem do fato de estarmos excluindo $i = 0$. Logo

$$I(q^m - 1, q^\lambda) = q^m - 1 - \partial g(x) = q^m - s(m) - c(m)$$

e como $c(m) \geq 0$,

$$I(q^m - 1, q^\lambda) \leq q^m - s(m). \tag{2.10}$$

O resultado seguinte ajuda na determinação de $s(m)$.

Lema 2.5.3 $s(k) = 0$, *se* $0 \leq k < r$

$$s(r) = 1$$

$$s(m) = qs(m-1) + (q-1)\left\{q^{m-r-1} - s(m-r-1)\right\},\ m > r.$$

Demonstração: $s(k)$ denota o número de k-uplas de q-árias que contêm uma corrida direta de comprimento $\geq r$. Se $0 \leq k < r$, então $s(k) = 0$.

Claramente $s(r) = 1$. Agora, para $s(m)$, onde $m > r$ denotemos por $i = (i_1, i_2, \ldots, i_m)$ uma m-upla onde $i_j \in \{0, 1, \ldots, q-1\}$, $1 \leq j \leq m$.

Uma m-upla contada em $s(m)$:

i) ou contém uma corrida de comprimento pelo menos r nas últimas $m-1$ posições, o que justifica o termo dado por $qs(m-1)$;

ii) ou é da forma $\underbrace{00\ldots0}_{r\text{-dígitos}}\ \underbrace{i_{m-r-1}i_{m-r-2}\ldots i_0}_{m-r\text{-dígitos}}$ onde $i_{m-r-1} \neq 0$ e não possui corridas de comprimento r nas últimas $m-r-1$ posições, o que justifica o termo dado por $q-1\{q^{m-r-1} - s(m-r-1)\}$. □

Agora, se tomarmos $\varphi(m) = q^m - s(m)$, da Equação em (2.10), temos

$$I(q^m - 1, q^\lambda) \leq \varphi(m).$$

Do Lema 2.5.3 seguem as seguintes relações

$$\varphi(k) = q^k \text{ para } 0 \leq k < r;\ \varphi(r) = q^r - 1 \tag{2.11}$$

$$\varphi(m) = q\varphi(m-1) - (q-1)\varphi(m-r-1),\ m > r. \tag{2.12}$$

A solução geral para a recorrência (2.12) é dada por

$$\varphi(m) = \sum_{i=0}^{r} a_i \rho_i^m \tag{2.13}$$

onde a_i, $i = 0, 1, \ldots, r$ são determinados por (2.11) e $\rho_0, \ldots, \rho_r$ são as raízes do polinômio dado por

$$\rho^{r+1} = q\rho^r - (q-1). \tag{2.14}$$

Todas as raízes da equação acima satisfazem $|\rho_i| < q$, $0 \leq i \leq r$. Isso é claro para as

raízes reais. No caso de raízes complexas, temos $\rho = be^{i\theta}$, $\theta \neq 0$ e onde $b < 1$, pois se supormos $b \geq 1$, a partir da Equação em (2.14) e usando a desigualdade triangular, vem

$$b^{r+1} + q - 1 > qb^r.$$

Por outro lado, a Equação em (2.14) também implica que

$$\begin{aligned} \rho^r &= (q-1)\left(\rho^{r-1} + \cdots + 1\right), \\ b^r &\leq (q-1)\left(b^{r-1} + \cdots + 1\right), \\ b^{r+1} &\leq qb^r - (q-1), \end{aligned}$$

o que contradiz a desigualdade anterior. Com o exposto acima, acabamos de provar o seguinte

Teorema 2.5.1 *O número de dígitos de informação $I\left(q^m - 1, q^\lambda\right)$ de um código BCH sobre $GF(q)$ de comprimento $q^m - 1$ e distância de projeto q^λ satisfaz*

$$I(q^m - 1, q^\lambda) \leq \sum_{i=0}^{r} a_i \rho_i^m$$

onde $r = m - \lambda$ e $a_0, \ldots, a_r, \rho_0, \ldots, \rho_r$ dependem de r (mas não de m) e satisfazem $|\rho_i| < q$.

2.6 Códigos BCH Longos são Assintoticamente Ruins

Definição 2.6.1 *Uma família de códigos sobre $GF(q)$, q fixo, é boa, se ela contém uma sequência infinita de códigos $C_1, C_2, \ldots$, onde C_i representa um código com parâmetros (n_i, k_i, d_i) e $R_i = \frac{k_i}{n_i}$ e $\delta_i = \frac{d_i}{n_i}$ aproximam-se de limites não nulos quando $i \to \infty$, isto é, quando $n_i \to \infty$.*

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Sabemos pelo Limitante de Gilbert-Varshamov assintótico que fixado R, $0 \leq R \leq 1$, existem códigos binários $[n, k, d]$ com $\frac{k}{n} \geq R$ e $\frac{d}{n} \geq H_2{}^{-1}(1-R)$, onde $H_2{}^{-1}(x)$ é inversa da função entropia.

Mostraremos agora que os códigos BCH primitivos são assintoticamente ruins.

Teorema 2.6.1 *Não existe uma sequência infinita de códigos BCH primitivos sobre* $GF(q)$ *tais que* d/n *e* k/n *tendam ambos a limites não nulos.*

Demonstração: Suponha que $C_1, C_2, \ldots$ formem uma sequência de códigos BCH onde C_i é do tipo $(n_i = q^{m_i} - 1, k_i, d_i)$ e $\frac{k_i}{n_i} \geq B_1 > 0$; $\frac{d_i}{n_i} \geq B_2 > 0$, $\forall i \geq 1$.

Seja C_i' um código BCH de comprimento n_i e distância de projeto $\delta_i = \left\lfloor \frac{d_i+1}{q} \right\rfloor$, onde $\lfloor x \rfloor$ denota o maior inteiro que não supera x.

Então a distância verdadeira d_i' de C_i' é menor ou igual a d_i pois pelo Teorema 2.4.2 temos que $d_i' \leq q\delta_i - 1$, onde $\delta_i \leq \frac{d_i+1}{q}$ e portanto $C_i' \supseteq C_i$.

Agora, se tomarmos λ_i tal que $q^{\lambda_i+1} > \delta_i \geq q^{\lambda_i}$ e considerarmos C_i'' o código BCH de comprimento n_i e distância de projeto q^{λ_i}, temos $C_i'' \supseteq C_i' \supseteq C_i$ e portanto $I(q^{m_i} - 1, q^{\lambda_i}) \geq k_i$.

Temos que

$$\frac{q^{\lambda_i}}{n_i} > \frac{\delta_i}{qn_i}$$

quando $n_i \to \infty$.

Como $\delta_i > \frac{d_i - q}{q}$ temos

$$\frac{\delta_i}{qn_i} > \frac{d_i}{q^2 n_i} - \frac{1}{qn_i} \geq \frac{B_2}{q^2} - \frac{1}{qn_i} > 0$$

quando $n_i \to \infty$.

Seja $r_i = m_i - \lambda_i$. Como

$$\frac{q^{\lambda_i}}{n_i} = \frac{q^{\lambda_i}}{q^{m_i} - 1} \cong q^{-r_i},$$

temos que r_i não pode crescer indefinidamente, pois $\frac{q^{\lambda_i}}{n_i} > 0$ quando $n_i \to \infty$. Então, $r_i \leq A_1$ (onde A_1 é uma constante) $\forall i \geq 1$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Do Teorema 2.5.1 sabemos que

$$I(q^{m_i}-1, q^{\lambda_i}) \leq \sum_{j=0}^{r_i} a_j^{(i)} (p_j^{(i)})^{m_i}.$$

Como $0 \leq r_i \leq A_1$, existe um número finito de $a_j^{(i)}s$. Seja $A_2 = \max \left| a_j^{(i)} \right|$.

Do mesmo modo, $A_3 = \max \left| p_j^{(i)} \right| < q$ e então

$$I(q^{m_i}-1, q^{\lambda_i}) \leq (A_1 + 1)\, A_2 A_3^{m_i}.$$

Portanto,

$$S_i = \frac{I(q^{m_i}-1, q^{\lambda_i})}{q^{m_i}-1}$$

tende a zero quando $i \to \infty$.

Mas $S_i \geq \frac{k_i}{n_i} \geq B_1 > 0$, uma contradição.

Logo não existe uma família boa de códigos BCH sobre $GF(q)$. □

2.7 Decodificação dos Códigos BCH

Iniciaremos agora a descrição do processo de decodificação de máxima verossimilhança dos códigos BCH, utilizando a métrica de Hamming.

Para isso, suponha que $v(x) = v_0 + v_1 x + \ldots + v_{n-1}x^{n-1}$ seja a palavra código transmitida através de um canal de comunicações e $r(x) = v(x) + e(x) = r_0 + r_1 x + \ldots + r_{n-1}x^{n-1}$ o vetor recebido, onde $e(x)$ é padrão de erro.

O primeiro passo na decodificação será o cálculo da síndrome $S = Hr^t$, onde H é a matriz verificação de paridade do código.

Com o objetivo de facilitar a notação e o entendimento, consideraremos códigos BCH

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

no sentido estrito, isto é, $l = 1$. Daí teremos que sua matriz verificação de paridade é:

$$H = \begin{pmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & (\alpha^2)^2 & & (\alpha^2)^{(n-1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & \alpha^{\delta-1} & (\alpha^{(\delta-1)})^2 & \cdots & (\alpha^{\delta-1})^{(n-1)} \end{pmatrix},$$

e $S_i = c(\alpha^i) = c_0 + c_1\alpha^i + \ldots + c_{n-1}\alpha^{(n-1)i}$, $1 \leq i \leq \delta - 1$, são as componentes do vetor síndrome.

Consideremos o padrão de erro com ν erros como sendo,

$$e(x) = Y_1 x^{j_1} + Y_2 x^{j_2} + \ldots + Y_\nu x^{j_\nu}, \tag{2.15}$$

onde $x^{j_1}, \ldots, x^{j_\nu}$ dão as localizações dos erros, com $0 \leq j_1 < j_2 < \ldots < j_\nu \leq n-1$ e $Y_1, \ldots, Y_\nu$ denotam suas respectivas magnitudes. Então:

$$S_j = Y_1(\alpha^{j_1})^j + Y_2(\alpha^{j_2})^j + \ldots + Y_\nu(\alpha^{j_\nu})^j, \quad j = 1, \ldots, \delta - 1 \tag{2.16}$$

é um conjunto de equações onde $\alpha^{j_1}, \alpha^{j_2}, \ldots, \alpha^{j_\nu}$ e $Y_1, Y_2, \ldots, Y_\nu$ são incógnitas. Qualquer método de resolução destas equações é um algoritmo de decodificação para os códigos BCH. Observe que uma vez encontrados $\alpha^{j_1}, \alpha^{j_2}, \ldots, \alpha^{j_\nu}$, as potências $j_1, j_2, \ldots, j_\nu$ nos fornecem as localizações dos erros.

Defina $X_i = \alpha^{j_i}$, para $1 \leq i \leq \nu$, como sendo os números de localização de erro. Então, as Equações em (2.16) podem ser expressas na forma

$$S_j = \sum_{i=1}^{\nu} Y_i X_i^j. \tag{2.17}$$

onde $1 \leq j \leq \delta - 1$ e ν representa o número de erros ocorridos.

As $\delta - 1$ Equações em (2.17) possuem q^k soluções (onde k é a dimensão do código) e cada uma das soluções resulta em um padrão de erro diferente. Se o número de erros for

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

menor ou igual a $t = \left\lfloor \frac{(\delta-1)}{2} \right\rfloor$, então a solução que produz o padrão de erro com o menor peso de Hamming é a solução correta.

Agora defina os valores $\sigma_1, \ldots, \sigma_\nu$ através da seguinte equação:

$$(X - X_1)(X - X_2)\ldots(X - X_\nu) = X^\nu + \sigma_1 X^{\nu-1} + \sigma_{\nu-1} X + \sigma_\nu. \tag{2.18}$$

Esses valores são conhecidos como funções simétricas elementares dos X_i e o polinômio de grau ν em (2.18) é dito o *polinômio localizador de erros.*

Multiplicando ambos os membros da Equação em (2.18) por $Y_i X_i^j$ e substituindo X por X_i obtemos:

$$Y_i X_i^{j+\nu} + Y_i X_i^{j+\nu-1}\sigma_1 + \ldots + Y_i X_i^{j+1}\sigma_{\nu-1} + Y_i X_i^j \sigma_\nu = 0 \tag{2.19}$$

Agora somando-as para $1 \leq i \leq \nu$ e usando o fato de que $S_j = \sum_{i=1}^{\nu} Y_i X_i^j$ obtemos:

$$S_{j+\nu} + S_{j+\nu-1}\sigma_1 + \ldots + S_{j+1}\sigma_{\nu-1} + S_j\sigma_\nu = 0 \tag{2.20}$$

onde todos os S_j são conhecidos se $1 \leq j \leq \delta - 1 - \nu$.

Dentre as possíveis soluções do sistema, a que nos interessa é a que tem menor valor de ν, pois o vetor erro assim obtido terá o menor peso de Hamming possível (e será portanto o mais provável ter ocorrido).

O algoritmo mais eficiente que se conhece para resolver esse sistema é o algoritmo de Berlekamp-Massey. Antes de descrevermos o mesmo, listamos a seguir os passos do procedimento para decodificação dos códigos BCH:

Passo 1 Cálculo do vetor síndrome $S = (S_1, S_2, \ldots, S_{\delta-1})$, a partir do vetor recebido r;

Passo 2 Cálculo das funções simétricas elementares $\sigma_1, \sigma_2 \ldots, \sigma_\nu$ a partir de S;

Passo 3 Cálculo dos números de localização de erro $X_1, X_2, \ldots, X_\nu$ a partir dos σ_i;

Passo 4 Cálculo das magnitudes dos erros Y_i a partir dos X_i e S.



Analisaremos agora detalhadamente cada um desses quatro passos.

Passo 1 Cálculo do Vetor Síndrome

$$S = rH^T.$$

Passo 2 Cálculo das Funções Simétricas Elementares.

O problema que precisamos resolver neste estágio é o seguinte: dada uma sequência de elementos $S_1, S_2, \ldots, S_{\delta-1}$ (as componentes do vetor síndrome), determine a solução do Sistema Linear em (2.20) nas incógnitas σ_i $(1 \leq i \leq \nu)$ tal que ν seja mínimo. Conforme já mencionado o algoritmo mais eficiente para se resolver tal sistema, é o Algoritmo de Berlekamp-Massey, o qual passaremos a descrever agora.

Este algoritmo é iterativo no sentido que no n-ésimo passo, o decodificador procura determinar um conjunto de l_n valores $\sigma_i^{(n)}$, tal que, as $n - l_n$ equações abaixo, conhecidas como *somas de potência*, sejam satisfeitas com o menor l_n possível e onde $\sigma_0^{(n)} = 1$. Assim

$$\begin{aligned}
&S_n\sigma_0^{(n)} + S_{n-1}\sigma_1^{(n)} + \ldots + S_{n-l_n}\sigma_{l_n}^{(n)} = 0 \\
&S_{n-1}\sigma_0^{(n)} + S_{n-2}\sigma_1^{(n)} + \ldots + S_{n-l_n-1}\sigma_{l_n}^{(n)} = 0 \\
&\vdots \\
&S_{l_n+1}\sigma_0^{(n)} + S_{l_n}\sigma_1^{(n)} + \ldots + S_1\sigma_{l_n}^{(n)} = 0
\end{aligned} \tag{2.21}$$

Escrevendo $\sigma^{(n)}$ na forma polinomial temos:

$$\sigma^{(n)}(X) = \sigma_0^{(n)} + \sigma_1^{(n)}X + \ldots + \sigma_{l_n}^{(n)}X^{l_n}$$

que é a representação da solução no n-ésimo estágio.

A n-ésima discrepância será denotada por d_n e é definida por

$$d_n = S_{n+1}\sigma_0^{(n)} + S_n\sigma_1^{(n)} + \ldots + S_{n+1-l_n}\sigma_{l_n}^{(n)}$$

Apresentaremos agora dois lemas relacionados com a determinação de $\sigma^{(n+1)}(X)$ (não

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

necessariamente com o menor valor de l_{n+1}) a partir de $\sigma^{(n)}(X)$.

Lema 2.7.1 *Suponha que $\sigma^{(n)}(X)$ seja um polinômio solução mínimal para as n primeiras somas de potência (isto é, satisfazem (2.21), com l_n mínimo) e tenha a próxima discrepância $d_n \neq 0$. Seja*

$$\sigma^{(m)}(X) = 1 + \sigma_1^{(m)} X + \ldots + \sigma_{l_m}^{(m)} X^{l_m}$$

qualquer polinômio solução para as m primeiras somas de potência, com $1 \leq m < n$ e tal que $d_m \neq 0$. Então o polinômio

$$\sigma^{(n+1)}(X) = \sigma^{(n)}(X) - d_n d_m^{-1} X^{n-m} \sigma^{(m)}(X)$$

é uma solução para as $n+1$ primeiras somas de potência. Mais ainda,

$$l_{n+1} = \max\{l_n, l_m + n - m\}.$$

Demonstração: $\sigma^{(n)}(X)$ é uma solução para as Equações em (2.17), isto é,

$$\sum_{i=0}^{l_n} S_{j-i}\sigma_i^{(n)}(X) = \begin{cases} 0 \text{ se } l_{n+1} \leq j \leq n \\ d_n \neq 0 \text{ se } j = n+1 \end{cases} \tag{2.22}$$

Do mesmo modo, como $\sigma^{(m)}(X)$ é uma solução para as m primeiras somas de potência, então:

$$\sum_{i=0}^{l_n} S_{j-i}\sigma^{(m)}(X) = \begin{cases} 0 \text{ se } l_{m+1} \leq j \leq m < n \\ d_m \neq 0 \text{ se } j = m+1 \end{cases} \tag{2.23}$$

Se $\sigma^{(n+1)}(X) = \sigma^{(m)}(X) - d_n d_m^{-1} X^{n-m} \sigma^{(m)}(X)$ for de fato uma solução para as $n+1$ primeiras somas de potência então deve-se ter que:

$$\sum_{i=0}^{l_{n+1}} S_{j-i}\sigma_i^{(n+1)} = 0 \tag{2.24}$$

para $l_{n+1}+1 \leq j \leq n+1$. Esta soma tem a forma

$$\sum_{i=0}^{l_{n+1}} S_{j-i}\left(\sigma_i^{(n)} - d_n d_m^{-1}\sigma_{i-(n-m)}^{(m)}\right) \tag{2.25}$$

Como $\sigma_i^{(n)} = 0$ para $i < 0$ e $i > l_n$ e $\sigma_i^{(m)} = 0$ para $i < 0$ e $i > l_m$, então, a soma de (2.25) pode ser escrita como

$$\sum_{i=0}^{l_n} S_{j-i}\sigma_i^{(n)} - d_n d_m^{-1} \sum_{i=n-m}^{l_n+n-m} S_{j-i}\sigma_{i-(n-m)}^{(m)} \tag{2.26}$$

Note que para $j = n+1$, a primeira soma de (2.26) é igual a d_n e a segunda é igual a d_m. A Equação em (2.26) reduz-se a $d_n - d_n d_m^{-1} d_m = 0$ e portanto é verificada. Pela Equação em (2.22) a primeira soma na Equação em (2.26) é igual a zero desde que $l_n + 1 \leq j \leq n$. Pela Equação em (2.23) a segunda soma na Equação em (2.26) é igual a zero desde que

$$l_m + 1 \leq j - (n-m) \leq m$$

ou equivalentemente, desde que $n - m + l_m + 1 \leq j \leq n$.

Em resumo, a Equação em (2.26) é satisfeita desde que

$$\max\{l_n, l_m + n - m\} + 1 \leq j \leq n+1$$

Como $(n+1)-\max\{l_n, l_m+n-m\}$ Equações em (2.21) são satisfeitas por $\sigma^{(n+1)}(X)$, então seu grau é dado formalmente por $l_{n+1} = \max\{l_n, l_m+n-m\}$. □

Lema 2.7.2 *Sejam $\sigma^{(n)}(X)$, l_n e $d_n \neq 0$ como definidos no Lema 2.7.1. Suponha que $\sigma^{(n+1)}(X)$ seja um polinômio solução das Equações em (2.21) satisfazendo $n+1-l_{n+1}$ equações e que*

$$\sigma^{(n+1)}(X) = \sigma^{(n)}(X) - aX^{n-m}\sigma^{(m)}(X),$$

onde $a \neq 0$ e $\sigma_0^{(m)} = 1$. Então o polinômio $\sigma^{(m)}(X)$ é uma solução para as $(m-l_m)$ primeiras Equações em (2.21) cuja próxima discrepância d_m satisfaz $d_m = d_n a^{-1}$ e $l_m =$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

$l_{n+1} - (n-m)$.

Demonstração: Por hipótese,

$$\sum_{i=0}^{l_{n+1}} S_{j-i}\sigma_i^{(n+1)} = 0 \tag{2.27}$$

para $l_{n+1} + 1 \leq j \leq n+1$ e

$$\sum_{i=0}^{l_n} S_{j-i}\sigma_i^{(n)} = \begin{cases} 0 \text{ se } l_n + 1 \leq j \leq n \\ d_n \neq 0 \text{ se } j = n+1 \end{cases} \tag{2.28}$$

Como $\sigma^{(n)}(X)$ é uma solução minimal, $l_{n+1} \geq l_n$. Subtraindo as Equações em (2.28) das Equações em (2.27) para $l_{n+1} + 1 \leq j \leq n+1$ obtém-se que

$$\sum_{i=0}^{l_{n+1}} S_{j-i}\left(\sigma_i^{(n+1)} - \sigma_i^{(n)}\right) = \begin{cases} 0 \text{ se } l_{n+1} + 1 \leq j \leq n \\ -d_n \text{ se } j = n+1 \end{cases} \tag{2.29}$$

Suponha agora que os $n-m$ primeiros coeficientes de S_{j-i} sejam nulos (note que como $\sigma_0^{(n+1)} = \sigma_0^{(n)} = 1$, então $n - m > 0$ isto é, $n > m$). Então, a Equação em (2.29) reduz-se a

$$\sum_{i=n-m}^{l_{n+1}} S_{j-i}\left(\sigma_i^{(n+1)} - \sigma_i^{(n)}\right) = \begin{cases} 0 \text{ se } l_{n+1} + 1 \leq j \leq n \\ -d_n \text{ se } j = n+1 \end{cases} \tag{2.30}$$

Fazendo $l_m = l_{n+1} - (n-m)$, a Equação em (2.30) pode ser escrita como

$$\sum_{i=0}^{l_m} S_{j-i}\left(\sigma_{i+n-m}^{(n+1)} - \sigma_{i+n-m}^{(n)}\right) = \begin{cases} 0 \text{ se } l_m + 1 \leq j \leq m \\ -d_n \text{ se } j = m+1 \end{cases} \tag{2.31}$$

Defina agora o polinômio $\sigma^{(m)}(X)$ por $\sigma_i^{(m)} = \left(\sigma_{i+n-m}^{(n+1)} - \sigma_{i+n-m}^{(n)}\right) a^{-1}$ para $0 \leq i \leq l_m$ e assim,

$$\sum_{i=0}^{l_n} S_{j-i}\sigma_i^{(m)} = \begin{cases} 0 \text{ se } l_m + 1 \leq j \leq m \\ -d_n a^{-1} = d_m \text{ se } j = n+1 \end{cases} \tag{2.32}$$

unesp

Então, pelas Equações em (2.32), $\sigma^{(m)}(X)$ é uma solução para as primeiras $m - l_m$ Equações em (2.21). A próxima discrepância será $d_m = a^{-1}d_n$ e o grau de $\sigma^{(m)}(X)$ é dado formalmente por $l_m = l_{n+1} - (n - m)$. □

Em vista dos Lemas 2.7.1 e 2.7.2 temos o seguinte.

Teorema 2.7.1 *Seja $\sigma^{(n)}(X)$ uma solução minimal no n-ésimo estágio de $\sigma^{(m)}(X)$ uma das soluções minimais anteriores, $1 \leq m < n$, tal que $d_m \neq 0$ e $m - l_m$ tenha o máximo valor. Então, uma solução minimal no estágio $n+1$ é $\sigma^{(n+1)}(X)$, onde:*
se $d_n = 0$, então

$$\sigma^{(n+1)}(X) = \sigma^{(n)}(X) \text{ e } l_{n+1} = l_n; \tag{2.33}$$

se $d_n \neq 0$ então

$$\sigma^{(n+1)}(X) = \sigma^{(n)}(X) - d_n d_m^{-1} X^{n-m} \sigma^{(n)}(X)$$

e

$$l_{n+1} = \max\{l_n, l_m + n - m\}. \tag{2.34}$$

Demonstração: Se $d_n = 0$, então $\sigma^{(n+1)} = \sigma^{(n)}(X)$ é claramente uma solução minimal já que $\sigma^{(n)}(X)$ o é. Já no caso $d_n \neq 0$, como $\sigma^{(m)}(X)$ e $\sigma^{(n)}(X)$ são conhecidos, obtemos $\sigma^{(n+1)}(X)$ por (2.34). Pelo Lema 2.7.1, $\sigma^{(n+1)}(X)$ é um polinômio solução de grau dado por $l_{n+1} = \max\{l_n, l_m + n - m\}$. Resta mostrar que essa solução é minimal.

- Se $m - l_m \geq n - l_n$, então $l_{n+1} = l_n$ pelo Lema 2.7.1 e daí $\sigma^{(n+1)}(X)$ será uma solução minimal no estágio $n+1$;
- Por outro lado, se $m - l_m < n - l_n$, então

$$l_{n+1} = \max\{l_n, l_m + n - m\} = l_m + n - m > l_n.$$

Mostraremos que $\sigma^{(n+1)}(X)$ é solução minimal. Para isso, suponha por absurdo que exista $D^{(n+1)}(X)$, um polinômio com grau d, tal que $l_n \leq d < l_m + n - m$. Existem dois casos a serem considerados:

i) Se $d = l_n$, pelo Lema 2.7.2 existe uma solução $\sigma^{(m')}(X)$ tal que $l_{m'} = d - (n - m')$, isto é, tal que $m' - l_{m'} = n - l_n$. Como por hipótese $m - l_m < n - l_n$, então $m - l_m < m' - l_{m'}$. Mas $m - l_m$ foi escolhido como o maior dos valores $k - l_k$ das soluções prévias. Contradição;

ii) Se $d > l_n$, então pelo Lema 2.7.2, $d = l_{m'} + n - m'$. Mas $m - l_m < m' - l_{m'}$. Então $d = n - (m' - l_{m'}) \geq n - (m - l_m) = l_{n+1} > d$, ou seja, $d > d$, uma contradição.

Portanto, $\sigma^{(n+1)}(X)$ é de fato uma solução minimal no estágio $n + 1$. □

Descreveremos agora o algoritmo de Berlekamp-Massey.

Inicie com a tabela

n	$\delta^{(n)}(X)$	d_n	l_n	$n - l_n$
-1	1	1	0	-1
0	1	S_1	0	0
0				
2				
$\vdots$				
$\delta - 1$				

Tabela 2.1: Iterações do Algoritmo de BM

Complete-a da seguinte maneira:

(1) $n \leftarrow 0$;

(2) se $d_n = 0$ então

$$\sigma^{(n+1)}(X) \longleftarrow \sigma^{(n)}(X) \text{ e } l_{n+1} \longleftarrow l_n$$

e vá para 5.

(3) se $d_n \neq 0$ então encontre um $m \leq n - 1$ tal que $d_m \neq 0$. Então:

$$\sigma^{(n+1)}(X) \longleftarrow \sigma^{(n)}(X) - d_n d_m^{-1} X^{n-m} \sigma^{(m)}(X)$$

e

$$l_{n+1} \longleftarrow \max\{l_n, l_m + n - m\};$$

(4) se $n < \delta - 2$, então

$$d_{n+1} \longleftarrow S_{n+2} + S_{n+1}\sigma_1^{(n+1)} + \ldots + S_{n+2-l_{n+1}}\sigma_{l_{n+1}}^{(n+1)};$$

(5) $n \longleftarrow n+1$; se $n < \delta - 1$ vá para o Passo **(2)**; caso contrário, fim.

Passo 3 Cálculo dos Números de Localização de Erro

Uma vez determinado os valores $\sigma_1, \sigma_2, \ldots, \sigma_\nu$ no **Passo 2** o decodificador deverá determinar os números de localização de erro, ou seja, encontrar as raízes do polinômio

$$\sigma(X) = X^\nu + \sigma_1 X^{\nu-1} + \ldots + \sigma_{\nu-1}X + \sigma_\nu.$$

Como o número de elementos de $GF(q^m)$ é finito, basta realizarmos uma busca para encontramos as raízes de $\sigma(X)$. Esse procedimento é conhecido como *busca de Chien* e consiste em se testar primeiramente se α^{n-1} é uma raiz de $\sigma(X)$. Em caso afirmativo o dígito r_{n-1} está incorreto. De forma análoga, as outras potências menores de α vão sendo testadas (se α^{n-l} for uma raiz, então o dígito r_{n-l} está incorreto, onde $1 \leq l \leq n$) até que ν delas sejam determinadas como sendo as raízes de $\sigma(X)$.

Uma maneira eficiente de se avaliar o polinômio $\sigma(X)$ em $X = X_0$ é através da *regra de Horn* isto é,

$$\sigma(X_0) = ((((X_0 + \sigma_1)X_0 + \sigma_2)X_0 + \sigma_3)X_0 + \ldots + \sigma_{\nu-1})X_0 + \sigma_\nu.$$

Passo 4 Cálculo das Magnitudes dos Erros

Para a completa decodificação, resta-nos agora descrever um método para o cálculo das magnitudes dos erros, ou seja, determinar Y_i, $1 \leq i \leq \nu$, a partir das ν equações



$S_j = \sum_{i=1}^{\nu} Y_i X_i^j$, $j = 1, \ldots, \delta - 1$. Temos:

$$\begin{bmatrix} X_1 & X_2 & \cdots & X_\nu \\ X_1^2 & X_2^2 & \cdots & X_\nu^2 \\ \cdots & \cdots & & \\ X_1^\nu & X_2^\nu & \cdots & X_\nu^\nu \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_\nu \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_\nu \end{bmatrix} \tag{2.35}$$

onde o determinante da matriz $\nu \times \nu$ é não-nulo. Assim $Y = (Y_1, Y_2, \ldots, Y_\nu)$ fica univocamente determinado. Ao invés de invertermos a matriz em (2.35) para determinarmos o vetor Y, descreveremos o procedimento proposto por Forney, o qual é mais eficiente.

Para isso definimos as funções simétricas elementares σ_{jl} dos números de localização de erro $X_1, X_2, \ldots, X_{j-1}, X_{j+1}, \ldots, X_\nu$ pela relação:

$$\prod_{i \neq j} (X - X_i) = \sum_{l=0}^{\nu-1} \sigma_{jl} X^{\nu-1-l} \tag{2.36}$$

Da Equação em (2.18), temos:

$$\prod_{i=1}^{\nu} (X - X_i) = \sum_{i=0}^{\nu-1} \sigma_i X^{\nu-i}$$

onde σ_0 e $\sigma_{j,0}$ são iguais a 1, a unidade de $GF(q^m)$. A partir das duas últimas equações obtemos

$$(X - X_j) \sum_{l=0}^{\nu-1} \sigma_{jl} X^{\nu-1-l} = \sum_{i=0}^{\nu} \sigma_i X^{\nu-i}.$$

Isto implica que:

$$\sum_{l=0}^{\nu-1} \sigma_{jl} X^{\nu-l} - \sum_{i=0}^{\nu-1} \sigma_{jl} X_j X^{\nu-1-l} = \sum_{i=0}^{\nu} \sigma_i X^{\nu-i}$$

e assim podemos concluir que os coeficientes σ_{jl} podem ser obtidos recursivamente pela relação

$$\sigma_{ji} = \sigma_i + X_j \sigma_{j,i-1}, \tag{2.37}$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

para $0 \leq i \leq \nu - 1$ e com $\sigma_0 = \sigma_{j,0} = 1$. Note que os X_i e os σ_i já são conhecidos neste ponto.

Denotando a magnitude de cada erro por Y_j temos

$$\sum_{l=0}^{\nu-1} \sigma_{jl} S_{\nu-l} = \sum_{l=0}^{\nu-1} \sigma_{jl} \sum_{i=1}^{\nu} Y_i X_i^{\nu-l} = \sum_{i=1}^{\nu} Y_i X_i \sum_{l=0}^{\nu-1} \sigma_{jl} X_i^{\nu-1-l}$$

e por (2.36), isto implica que

$$\sum_{l=0}^{\nu-1} \sigma_{jl} S_{\nu-l} = \sum_{i=1}^{\nu} Y_i X_i \prod_{m \neq j} (X_i - X_m) = Y_j X_j \prod_{m \neq j} (X_j - X_m). \tag{2.38}$$

A última igualdade em (2.36), segue do fato de que o somatório em questão só não se anula se $i = j$. Pela Equação em (2.38) podemos concluir que:

$$\sum_{l=0}^{\nu-1} \sigma_{jl} S_{\nu-l} = Y_j \sum_{l=0}^{\nu-1} \sigma_{jl} X_j^{\nu-l}.$$

E disto segue que cada Y_j $(1 \leq j \leq \nu)$ é dado pela expressão

$$Y_j = \frac{\sum_{l=0}^{\nu-1} \sigma_{jl} S_{\nu-l}}{\sum_{l=0}^{\nu-1} \sigma_{jl} S_j^{\nu-l}} \tag{2.39}$$

E com isso concluímos o processo de decodificação dos códigos BCH.

Exemplo 2.7.1 *Considere o código Reed-Solomon* $(15, 7)$ *definido sobre* $GF(16)$, *capaz de corrigir até três erros. A matriz verificação de paridade é dada por*

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{14} \\ 1 & \alpha^2 & (\alpha^2)^2 & & (\alpha^2)^{14} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & \alpha^6 & (\alpha^6)^2 & \cdots & (\alpha^6)^{14} \end{bmatrix}$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Assuma que a palavra código $v = (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)$ *seja transmitida e que o vetor* $r = (0,0,0,\alpha^7,0,0,\alpha^3,0,0,0,0,0,\alpha^4,0,0)$ *seja recebido. Calculando a síndrome* S*, obtemos*

$$S = rH^t = \left(\alpha^{12}, 1, \alpha^{14}, \alpha^{10}, 0, \alpha^{12}\right).$$

O cálculo das funções simétricas elementares é feito utilizando o algoritmo de Berlekamp-Massey. Assim,

n	$\sigma^{(n)}(x)$	d_n	l_n	$n - l_n$
-1	1	1	0	-1
0	1	α^{12}	0	0
1	$1 + \alpha^{12}X$	α^7	1	0
2	$1 + \alpha^3 X$	1	1	1
3	$1 + \alpha^3 X + \alpha^3 X^2$	α^7	2	1
4	$1 + \alpha^4 X + \alpha^{12} X^2$	α^{10}	2	2
5	$1 + \alpha^7 X + \alpha^4 X^2 + \alpha^6 X^3$	0	3	2
6	$1 + \alpha^7 X + \alpha^4 X^2 + \alpha^6 X^3$	$-$	$-$	$-$

Portanto $\sigma(X) = 1 + \alpha^7 X + \alpha^4 X^2 + \alpha^6 X^3$ *é o polinômio localizador de erros, o que significa que o padrão de erro introduzido pelo canal contém três erros;* $\sigma_1 = \alpha^7$, $\sigma_2 = \alpha^4$ *e* $\sigma_3 = \alpha^6$*. Assim, para calcularmos os números de localização de erros precisamos resolver a equação polinomial* $\rho(x) = X^3 + \alpha^7 X^2 + \alpha^4 X + \alpha^6 = 0$ *em* X*. Suas raízes são* $X_1 = \alpha^3$, $X_2 = \alpha^6$ *e* $X_3 = \alpha^{12}$*, encontradas por inspeção direta em* $GF(16)$*. Assim, concluímos que ocorreram erros nas posições* X^3, X^6 *e* X^{12}*. Podemos então passar para o quarto passo da decodificação. Fazendo uso da Equação em (2.21) obtemos:*

$$\begin{array}{lll} \sigma_{10} = 1 & \sigma_{20} = 1 & \sigma_{30} = 1 \\ \sigma_{11} = \alpha^4 & \sigma_{21} = \alpha_{10} & \sigma_{31} = \alpha^2 \\ \sigma_{12} = \alpha^3 & \sigma_{22} = 1 & \sigma_{32} = \alpha^9 \end{array}$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Da Equação em (2.37) obtemos as magnitudes dos três erros: $Y_1 = \alpha^7$, $Y_2 = \alpha^3$, $Y_3 = \alpha^4$.

2.8 O Algoritmo de Peterson-Gorenstein-Zierler

Suponha que o padrão de erro introduzido pelo canal de comunicações contenha v erros. Iremos mostrar que a solução do sistema linear (nas incógnitas $\sigma_1, \sigma_2, \ldots, \sigma_\mu$) dado por

$$M = \begin{bmatrix} S_1 & S_2 & \cdots & S_\mu \\ S_2 & S_3 & \cdots & S_{\mu+1} \\ \vdots & \vdots & \cdots & \vdots \\ S_{\delta-\mu-1} & S_{\delta-\mu} & \cdots & S_{\delta-2} \end{bmatrix} \begin{bmatrix} \sigma_\mu \\ \sigma_{\mu-1} \\ \vdots \\ \sigma_1 \end{bmatrix} = \begin{bmatrix} -S_{\mu+1} \\ -S_{\mu+2} \\ \vdots \\ -S_{\delta-1} \end{bmatrix} \tag{2.40}$$

é única se $\mu = v$. Se $\mu > \nu$, o sistema é singular.

Por construção, sabemos que se $M = \nu$, então este Sistema em (2.40) admite ao menos uma solução a qual é representada pelas próprias funções simétricas elementares dos X_i. Seja $Y_j = X_j = 0$, se $j > \nu$. Considere agora a submatriz M associada ν às primeiras Equações em (2.40). Então,

$$M = \begin{bmatrix} S_1 & S_2 & \cdots & S_\mu \\ S_2 & S_3 & \cdots & S_{\mu+1} \\ \vdots & \vdots & \cdots & \vdots \\ S_\mu & S_{\mu+1} & \cdots & S_{2\mu-1} \end{bmatrix},$$

e podemos usar as Equações em (2.17) para constatarmos que $M = VDV^t$, onde:

$$V = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ X_1 & X_2 & \cdots & X_\mu \\ \vdots & \vdots & \cdots & \vdots \\ X_1^{\mu-1} & X_2^{\mu-1} & \cdots & X_\mu^{\mu-1} \end{bmatrix} \text{ e } D = \begin{bmatrix} Y_1X_1 & 0 & \cdots & 0 \\ 0 & Y_2X_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Y_\mu X_\mu \end{bmatrix}.$$

A matriz V é uma matriz de Vandermonde e seu determinante é expresso por $\det V =$

unesp

$\det V^t = \prod_{i>j} (X_i - X_j) \neq 0$, se $\mu = \nu$; se $\mu > \nu$, então $V = \det V^t = 0$. A matriz D tem determinante $\det D = \prod_{i=1}^{\mu} Y_i X_i \neq 0$, se $\mu = \nu$; se $\mu > \nu$, então $\det M = 0$.

Portanto, se $\mu = \nu$, podemos afirmar que $\det M = (\det V)^2 (\det D) \neq 0$ e assim a solução para o sistema de Equações em (2.40) ou (2.20) terá solução minimal única. Se $\mu > \nu$, então o sistema de Equações em (2.40) ou (2.20) é singular.

A partir desta discussão, chegamos ao seguinte algoritmo para se resolver as Equações em (2.20):

- Algoritmo de Peterson-Gorenstein-Zierler

1) $\mu \leftarrow \lfloor (\delta - 1)/2 \rfloor$;

2) se $\det M \neq 0$, fim.

3) $\mu \leftarrow \mu - 1$; vá para 2).

O número de erros é dado por $\nu = \mu$ e a solução $(\sigma_1, \sigma_2, \ldots, \sigma_\nu)$ de (2.40) é encontrada multiplicando-se a matriz M^{-1} por $(-S_{\mu+1}, -S_{\mu+2}, \ldots, -S_{2\mu})^T$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Capítulo 3

Curvas Algébricas

Neste capítulo serão apresentados os conceitos e resultados de Geometria Algébrica que se farão necessários para o estudo dos códigos Algébrico-Geométricos, a serem apresentados no Capítulo 5. Uma boa referência sobre o assunto é [4].

3.1 Variedades Projetivas e Afins

Denotaremos por F_q um corpo finito com q elementos e por $\overline{F}_q$ o seu fecho algébrico.

$\mathbb{A}^n = \left\{(a_1, \ldots, a_n) \mid a_i \in \overline{F}_q\right\}$ é o espaço afim n-dimensional. Se P é um ponto de $\mathbb{A}^n$, P é dito um ponto afim.

Se $\mathbb{G}$ é um subcorpo de $\overline{F}_q$ tal que $F_q \subset \mathbb{G}$ e $P = (a_1, \ldots, a_n)$ é tal que $a_i \in \mathbb{G}$, $i = 1, \ldots, n$, então P é dito um ponto $\mathbb{G}$-racional de $\mathbb{A}^n$. Denotaremos por $\mathbb{A}^n(\mathbb{G})$ o conjunto de todos os pontos $\mathbb{G}$-racionais.

$\mathbb{P}^n = \left\{(a_0 : a_1 : \ldots : a_n) \mid a_i \in \overline{F}_q, \text{ não todos nulos}\right\}$ define o espaço projetivo n-dimensional, onde $(a_0 : a_1 : \ldots : a_n)$ denota a classe de equivalência de $(a_0, a_1, \ldots, a_n)$, definida pela seguinte relação (de equivalência):

$$(a_0, a_1, \ldots, a_n) \equiv (b_0, b_1, \ldots, b_n) \Leftrightarrow \exists \lambda \in \overline{F}_q \backslash \{0\}\text{ , tal que } b_i = \lambda a_i, i = 0, 1, \ldots, n.$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Se $\mathbb{G}$ é um subcorpo de $\overline{F}_q$, com $F_q \subset \mathbb{G}$ e $P = (a_0 : a_1 : \ldots : a_n)$, tal que $a_i \in \mathbb{G}$, $i = 1, \ldots, n$, então P é dito um ponto $\mathbb{G}$-racional de $\mathbb{P}^n$ e o conjunto de todos os pontos $\mathbb{G}$-racionais de $\mathbb{P}^n$ é denotado por $\mathbb{P}^n(\mathbb{G})$.

Um ponto em $\mathbb{P}^n$ da forma $(0 : a_1 : \ldots : a_n)$ é dito um ponto no infinito. O conjunto de todos os pontos no infinito é dito um hiperplano no infinito e é dado por $H = \{(0 : a_1 : \ldots : a_n) \in \mathbb{P}^n\}$. Denotaremos por Q um ponto no infinito.

A função φ dada por $\varphi : \begin{array}{ccc} \mathbb{A}^n & \to & \mathbb{P}^n \backslash H \\ (a_1, \ldots, a_n) & \mapsto & (1 : a_1 : \ldots : a_n) \end{array}$ é a inclusão de $\mathbb{A}^n$ em $\mathbb{P}^n$.

Exemplo 3.1.1 *O espaço projetivo de dimensão 1 (a reta projetiva) é formado pelos pontos da forma* $(1 : a_1)$, $a_1 \in \overline{F}_q$, *mais o ponto no infinito* $(0 : 1)$.

Um polinômio $f \in \overline{F}_q[x_1, \ldots x_n]$ pode ser considerado pela função $f : \begin{array}{ccc} \mathbb{A}^n & \to & \overline{F}_q \\ P & \mapsto & f(P) \end{array}$,

onde P é um ponto de A^n.

Se $f(P) = 0$, P é dito um zero de f.

Definição 3.1.1 $Z(T) = \{P \in \mathbb{A}^n \mid f(P) = 0, \forall f \in T\}$, *onde* $T \subseteq \overline{F}_q[x_1, \ldots, x_n]$, *é o conjunto dos zeros de* T.

Definição 3.1.2 $V \subset \mathbb{A}^n$ *é dito um conjunto algébrico em* $\mathbb{A}^n$ *se existir* $T \subseteq \overline{F}_q[x_1, \ldots, x_n]$ *tal que* $V = Z(T)$.

Se I é um ideal de $\overline{F}_q[x_1, \ldots, x_n]$ gerado por T então $Z(T) = V(I)$ e então todo conjunto algébrico é igual a $V(I)$ para algum ideal I.

Definição 3.1.3 *Se* $V \subset \mathbb{A}^n$ *é um conjunto algébrico, então o conjunto*

$$I(V) = \left\{ f \in \overline{F}_q[x_1, \ldots, x_n] \mid f(P) = 0, \ \forall P \in V \right\}$$

é dito o ideal de V.

Dados os polinômios $f, g \in I(V)$, se $f(P) = 0$ e $g(P) = 0$, $\forall P \in V$, então $(f \pm g)(P) = f(P) \pm g(P) = 0, \forall P \in V$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Agora se $f \in I(V)$ e $g \in \overline{F}_q[x_1, \ldots, x_n]$ então $(fg)(P) = f(P)\,g(P) = 0$, pois $f(P) = 0, \forall P \in V$.

Com isso, vemos que $I(V)$ é de fato um ideal de $\overline{F}_q[x_1, \ldots, x_n]$. Como $\overline{F}_q[x_1, \ldots, x_n]$ é um domínio Noetheriano, ou seja, todo ideal de $\overline{F}_q[x_1, \ldots, x_n]$ é finitamente gerado e pelo seguinte

Lema 3.1.1 *(Hilbert-Nullstellensatz): Todo ideal maximal de $\overline{F}_q[x_1, \ldots, x_n]$ é da forma $\langle x_1 - a_1, \ldots, x_n - a_n \rangle$, $a_i \in \overline{F}_q$, $i = 1, \ldots, n$ e para todo $P \in \mathbb{A}^n$ dado, tem-se que $\{P\}$ é um conjunto algébrico, com ideal $I(P) = \langle x_1 - a_1, \ldots, x_n - a_n \rangle$, temos:*

Definição 3.1.4 *Uma variedade afim V em $\mathbb{A}^n$ é um conjunto algébrico, onde $I(V)$ é um ideal primo.*

Seja $V(I)^c = A^n \backslash V(I)$ e $Z = \left\{ V(I)^c \mid I \text{ é um ideal de } \overline{F}_q[x_1, \ldots, x_n] \right\}$. Z denota um espaço topológico cujos abertos são os complementares de conjuntos algébricos, pois satisfaz:

1) $(V(0))^c = \emptyset$

$(V(1))^c = \mathbb{A}^n$ e assim $\emptyset$ e $\mathbb{A}^n$ estão em Z.

2) $U_\alpha = (V(I_\alpha))^c$

$$\bigcup_\alpha U_\alpha = \bigcup_\alpha (V(I_\alpha))^c = \bigcap_\alpha (V(I_\alpha))^c \in Z.$$

3) $U_1 = V(I)^c, U_2 = V(J)^c$

$$\begin{aligned} U_1 \cap U_2 &= V(I)^c \cap V(J)^c \\ &= (V(I) \cup V(J))^c \\ &\quad (V(I \cap J)^c) \in Z \end{aligned}$$

Essa topologia é denominada topologia de Zariski.

Definição 3.1.5 *Se V é uma variedade afim, o anel quociente*

$$\overline{F}_q[V] = \overline{F}_q[x_1, \ldots, x_n]/I(V)$$

é dito o anel das coordenadas de V.

Se V é uma variedade então $I(V)$ é um ideal primo e portanto o anel das coordenadas $\overline{F}_q[V]$ é um domínio. Assim temos:

Definição 3.1.6 *Seja V uma variedade afim. O corpo das frações de $\overline{F}_q[V]$ é denotado por $\overline{F}_q(V)$ e é chamado de corpo das funções de V.*

O corpo das funções $\overline{F}_q(V)$ é então uma extensão finitamente gerada de $\overline{F}_q$ e assim existem $x_1, \ldots, x_k \in \overline{F}_q(V)$ tais que $\overline{F}_q(V) = \overline{F}_q(x_1, \ldots, x_k)$.

A dimensão de uma variedade afim é o grau de transcendência de $\overline{F}_q(V)$ sobre F_q.

A partir de agora denotaremos por X uma curva afim que é uma variedade de dimensão 1 em $\mathbb{A}^n$. Se um polinômio define uma curva afim X, este será denotado por F_X, ou simplesmente por F.

Exemplo 3.1.2 *Seja $F \in \overline{F}_q[x, y]$ um polinômio irredutível. Então*

$$X = \left\{ P \in \mathbb{A}^2 \mid F(P) = 0 \right\}$$

é uma variedade e o corpo das funções $\overline{F}_q(X)$ tem grau de transcedência 1. Portanto X é uma curva afim em $\mathbb{A}^2$ chamada curva plana afim.

Seja X uma curva dada por $F(x, y) = 0$ e seja $P = (x_0, y_0) \in X$. P é dito um ponto não-singular se pelo menos uma das derivadas parciais F_x ou F_y é não nula em P. Neste caso, a equação da reta tangente ao ponto P será dada pelo polinômio linear (isto é, um polinômio de grau 1) com equação

$$t_{(x_0,y_0)}(x, y) = F_x(x_0, y_0)(x - x_0) + F_y(x_0, y_0)(y - y_0) = 0.$$

Definição 3.1.7 *A curva X é dita não-singular (ou lisa, ou regular) se todos seus pontos são não-singulares (ou simples). Caso contrário, diz-se que X é uma curva singular.*

Um outro modo de sabermos se uma curva é singular ou não é através da existência ou não de zeros comuns nas derivadas parciais.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Exemplo 3.1.3 *Considere o corpo F_q onde $q = r^2 = p^{2m}$ (p um número primo). A curva Hermitiana é descrita pelo polinômio $F(x,y) = y^r + y - x^{r+1}$. Suas derivadas parciais são dadas por*

$$\begin{aligned} F_x(x,y) &= -(r+1)x^r = -x^r \quad (r = p = 0 \text{ em } F_q) \\ F_y(x,y) &= ry^{r-1} + 1 = 1, \end{aligned}$$

que não possuem raízes em comum (F_y não tem raízes). Logo, trata-se de uma curva lisa.

Da mesma forma que chegamos à definição de uma curva plana afim, passaremos agora à definição de uma curva plana projetiva em $\mathbb{P}^m$. Para isso, utilizaremos as seguintes notações.

Denotaremos por $G \in \overline{F}_q[x_0, x_1, \ldots, x_n]$ o monômio de grau d dado por

$$G = a\prod x_i^{d_i},$$

com $a \neq 0$ e $\sum_{i=0}^{n} d_i = d$. Um polinômio homogêneo F é formado pela soma de monômios de mesmo grau.

Se T denota um conjunto de polinômios homogêneos sobre $\overline{F}_q[x_0, x_1, \ldots, x_n]$, definimos o conjunto dos zeros de T como sendo

$$Z(T) = \{P \in \mathbb{P}^n \mid f(P) = 0,\ \forall f \in T\}.$$

Definição 3.1.8 *Um subconjunto $V \subset \mathbb{P}^m$ é um conjunto algébrico projetivo se existe um conjunto de polinômios homogêneos T tais que $V = Z(T)$. Se denotarmos por I o ideal gerado por T então $V(I) = Z(T)$.*

Definição 3.1.9 *Se $V \subseteq \mathbb{P}^n$ é um conjunto algébrico projetivo, então o ideal em $\overline{F}_q[x_0, \ldots, x_n]$ que é gerado pelos polinômios homogêneos F com $F(P) = 0$ para todo*

$P \in V$ é denominado ideal de V e é denotado por $I(V)$.

Definição 3.1.10 *Uma variedade projetiva V em $\mathbb{P}^n$ é um conjunto algébrico projetivo tal que $I(V)$ é um ideal primo.*

Definição 3.1.11 *Se $V \subseteq \mathbb{P}^n$ é uma variedade projetiva não vazia então, o anel quociente*

$$\Gamma_h(V) = \overline{F}_q[x_0, \ldots, x_n] / I(V)$$

é dito o anel das coordenadas homogêneas de V.

Um elemento $f \in \Gamma_h(V)$ é denominado uma forma de grau d se $f = F + I(V)$, onde F é um polinômio homogêneo de grau d.

O corpo das funções de V é dado por

$$\overline{F}_q(V) = \left\{ \frac{g}{h} \mid g, h \in \Gamma_h(V) \text{ são formas de mesmo grau e } h \neq 0 \right\}.$$

A dimensão da variedade projetiva V é o grau de transcendência de $\overline{F}_q(V)$ sobre F_q.

Definição 3.1.12 *Uma curva projetiva $X \subseteq \mathbb{P}^n$ é uma variedade projetiva de dimensão 1.*

Exemplo 3.1.4 *Sejam $F \in \overline{F}_q[x, y, z]$ um polinômio irredutível homogêneo e*

$$X = \{F = 0\} = \left\{ P \in \mathbb{P}^2 \mid F(P) = 0 \right\}$$

uma variedade. Então X define uma curva projetiva.

A relação entre variedades projetivas e afins se dá da seguinte forma:

Se $F = F(x_1, \ldots, x_n) \in \overline{F}_q[x_1, \ldots, x_n]$ tem grau d, colocamos

$$F^* = x_0^d F\left(\frac{x_1}{x_0}, \ldots, \frac{x_n}{x_0}\right) \in \overline{F}_q[x_0, x_1, \ldots, x_n].$$

Então F^* é um polinômio homogêneo de grau d em $n+1$ variáveis.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Se $V \in \mathbb{A}^n$, com ideal $I(V) \subset \overline{F}_q[x_1, \ldots, x_n]$, o fecho projetivo de V é dado por

$$\overline{V} = \{P \in \mathbb{P}^n \mid F^*(P) = 0,\ \forall F \in I(V)\}.$$

Por outro lado, se $\overline{V} \subset \mathbb{P}^m$ é uma variedade projetiva e $W = \overline{V} \cap \{(c_0 : \ldots : c_n) \in \mathbb{P}^n \mid c_0 \neq 0\} \neq \emptyset$, definimos $\varphi : \begin{array}{ccc} \mathbb{A}^n & \to & \mathbb{P}^m \\ (a_1, \ldots, a_n) & \mapsto & (1 : a_1 : \ldots : a_n) \end{array}$. Então $V = \varphi^{-1}(W)$, é uma variedade afim e $I(V) = \left\{F(1 : x_1 : \ldots : x_n) \mid F \in I\left(\overline{V}\right)\right\}$, onde o fecho projetivo de V é $\overline{V}$.

Se V é uma variedade afim e $\overline{V}$ é seu fecho projetivo, o corpo das funções $\overline{F}_q(V)$ é isomorfo a $\overline{F}_q\left(\overline{V}\right)$ e, além disso, V e $\overline{V}$ tem a mesma dimensão.

Exemplo 3.1.5 *O fecho projetivo da curva Hermitiana dada no Exemplo 3.1.3 é descrito pela equação* $y^r z + yz^r - x^{r+1} = 0$ *e esta curva tem somente um ponto no infinito, a saber,* $Q = (0 : 1 : 0)$.

Seja P um ponto numa variedade V. Se $f \in \overline{F}_q(V)$ então $f = g/h$, com $g, h \in \overline{F}_q[V]$ para uma variedade afim e $f = g/h$ com $g, h \in \Gamma_h(V)$ para uma variedade projetiva.

Dizemos que f está definida (é regular) em P, se existir um representante de $f = g/h$, com $h(P) \neq 0$. Se f não está definida em P, dizemos que P é um pólo de f.

3.2 Anel Local em um Ponto

Definimos o anel $O_P(V) = \left\{f \in \overline{F}_q(V) \mid f \text{ é regular em } P\right\}$, como sendo o anel local de V em P.

A avaliação de um elemento $f \in O_P(V)$ é $f(P) = g(P)/h(P)$ no caso afim e no caso projetivo teremos $g = G + I(V)$, $h = H + I(V) \in \Gamma_h(V)$ onde G e H são polinômios homogêneos de grau d. Seja $P = (a_0 : a_1 : \ldots : a_n)$, então

$$\frac{G(\lambda a_0, \ldots, \lambda a_n)}{H(\lambda a_0, \ldots, \lambda a_n)} = \frac{G(a_0, \ldots, a_n)}{H(a_0, \ldots, a_n)}$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

e portanto

$$f(P) = G(a_0, \ldots, a_n) / H(a_0, \ldots, a_n)$$

se $H(P) \neq 0$.

Pode-se verificar que de fato $O_P(V)$ é um anel local, onde

$$M_P(V) = \left\{ f = \frac{g}{h} \in O_P(V) \mid f(P) = 0 \right\}$$

é o seu único ideal maximal.

Definição 3.2.1 *O anel de valorização do corpo das funções $\overline{F}_q(V)$ é um anel O que satisfaz:*

- $\overline{F}_q \subset O \subset \overline{F}_q(V)$;
- *Para todo $z \in \overline{F}_q(V)$, $z \in O$ ou $z^{-1} \in O$.*

Proposição 3.2.1 *Seja $P \in V$. O anel $O_P(V)$ é um domínio local noetheriano com ideal maximal $M_P(V)$. O ideal $M_P(V)$ é formado pelos elementos não invertíveis do anel $O_P(V)$.*

Definição 3.2.2 *Se $M_P(V) = \langle t \rangle$, então t é chamado de parâmetro local ou parâmetro uniformizante para P.*

Agora, se X é uma curva plana e P um ponto regular de X, temos $M_P(X) = \langle t \rangle$ e

$$\overline{F}_q[X] = \bigcap_{P \in X} O_P(X) \subset O_P(X) \subset \overline{F}_q(X).$$

Se $z \in \overline{F}_q[X]$, $z \neq 0$

- $z \in M_P(X) \Rightarrow z = t^n u$ onde u é invertível em $O_P(X)$, ou
- $z \notin M_P(X) \Rightarrow z$ é invertível em $O_P(X) \Rightarrow z(P) \neq 0$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Defina $v_P: \overline{F}_q[X] \to \mathbb{N}$

$$z \mapsto v_P(z) = \begin{cases} 0 \text{ se } z \text{ é invertível em } O_P(X) \\ n \text{ se } z \in M_P(X) \text{ e } z = t^n u \end{cases}.$$

Se $z \in \overline{F}_q(X)$, então $z = \frac{g}{h}$ com $g, h \in \overline{F}_q[X]$ e $h \neq 0$ e $v_P(z) = v_P(g) - v_P(h)$

Com isso, defina $v_P: \overline{F}_q(X) \to \mathbb{Z}$

$$z = \frac{g}{h} \mapsto v_P(g) - v_P(h)$$

Proposição 3.2.1 *$v_P(z)$ não depende do parâmetro local t.*

Sejam $z, w \in \overline{F}_q(X)$ e P um ponto regular da curva X valem as seguintes propriedades:

- $z = 0 \Leftrightarrow v_P(z) = \infty$,
- $z = a \in \overline{F}_q$; onde $a \neq 0 \Rightarrow v_P(z) = 0$,
- $z,\ w \neq 0$; $v_P(zw) = v_P(z) + v_P(w)$,
- $z \neq 0$, $v_P(z^{-1}) = -v_P(z)$,
- $v_P(z + w) \geq \min\{v_P(z), v_P(w)\}$.

Podemos também definir a valorização de z em P, $v_P(z)$, como sendo a ordem de anulamento de z em P, a qual é denotada por $\text{ord}_P(z)$.

Exemplo 3.2.1 *Considere a curva projetiva X, com equação $zy^2 + yz^2 = x^3$ sobre $\overline{F}_2$. No ponto $Q = (0:1:1)$, tome $t = x/z$ como parâmetro local e seja $f = \frac{x}{y+z}$. Temos:*

$$\frac{x}{y+z} = \frac{x^3}{x^2y + x^2z} = \frac{zy^2 + z^2y}{x^2y + x^2z} = \frac{zy}{x^2} = t^{-2}\frac{y}{z}$$

onde y/z é uma unidade em $O_Q(X)$. Então $v_Q(f) = -2$.

3.3 Divisores e Teorema de Riemann-Roch

Seja X uma curva projetiva regular sobre $\overline{F}_q$. Temos então o seguinte

Definição 3.3.1 *1) Um divisor de X é uma soma formal $D := \sum_{P \in X} n_P P$ onde $n_P \in \mathbb{Z}$ e $n_P = 0$ exceto para um número finito de pontos P em X.*

2) O grau do divisor D é dado por $\deg(D) = \sum n_P$.

3) Se $D \geq 0$ para todo n_P, D é dito um divisor efetivo (notação $D \succ 0$).

Os divisores de X denotados por $Div(X) := \{D \mid D \text{ é um divisor de } X\}$ formam um grupo abeliano.

Seja $v_P = \text{ord}\ P$ a valorização em P para funções sobre X como visto anteriormente.

Seja $f \in \overline{F}_q(X)$. A ordem de f no ponto P de X é dada por $v_P(f)$. Se $v_P(f) > 0$, dizemos que f tem um zero em P e se $v_P(f) < 0$, f terá um pólo em P.

Definição 3.3.2 *Se f é uma função racional de X não nula, definimos o divisor de f como sendo*

$$(f) := \sum_{P \in X} v_P(f)\, P.$$

Assim, o divisor de f nos dirá onde estão os zeros e os pólos de f com suas multiplicidades e ordens. Como f é função ~~racional~~, ou seja, possui como numerador e denominador polinômios de mesmo grau, segue que f tem o mesmo número de zeros e pólos contados propriamente. Assim,

Teorema 3.3.1 *O grau do divisor de uma função racional é 0.*

Dizemos que os divisores D e D' são linearmente equivalentes se, e somente se, $D - D'$ é divisor de uma função racional (notação $D \equiv D'$)

Definição 3.3.3 *Seja G um divisor da curva X, ou seja, $G \in Div(X)$. O espaço vetorial $\mathfrak{L}(G)$ sobre F_q é dado por:*

$$\mathfrak{L}(G) := \left\{ f \in \overline{F}_q(X) \mid (f) + G \geq 0 \right\} \cup \{0\}.$$

$\mathfrak{L}(G)$ é o conjunto de funções racionais com pólos apenas nos zeros do divisor G e tendo zeros nos pólos de G. $\mathfrak{L}(G)$ é espaço vetorial de dimensão finita sobre F_q o qual doravante denotaremos por $l(G)$ sua dimensão.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Para determinarmos $l(G)$ partiremos da definição de diferenciais que são objetos da forma fdh, onde f e h são funções racionais com as relações $d(f+g) = df + dg$ e $d(fg) = fdg + gdf$ e $da = 0$ para $a \in \overline{F}_q$.

Para estendermos o conceito acima para formas diferenciáveis racionais, basta fazermos como no caso das funções racionais observando que $d(f/g) = (gdf - fdg)/g^2$.

Chamaremos as formas racionais diferenciáveis sobre X de diferenciáveis e denotaremos o espaço dessas formas por $\Omega(X)$, onde $\Omega(X)$ tem dimensão 1 sobre $\overline{F}_q(X)$.

Sabemos que para todo ponto P existe um parâmetro local, ou seja, uma função u tal que $v_P(u) = 1$ e para toda diferencial $w \in \Omega(X)$ existe uma função f tal que $w = fdu$. Assim, a valorização $v_P(w)$ é agora por definição $v_P(f)$ e podemos dizer que w tem um zero de ordem ρ se $\rho = v_P(f) > 0$ e w tem um pólo de ordem ρ se $\rho = -v_P(f) > 0$. Com isso temos,

Definição 3.3.4 *O divisor (w) da diferencial w é dado por*

$$(w) := \sum_{P \in X} v_P(u)\, P,$$

onde $w = fdu$ é a representação local de w e v_P é a ordem em O_P.

O divisor de uma diferencial é dito canônico e tem grau $2g-2$, onde g é o gênero (ou gênus) da curva que determinaremos a seguir.

Definição 3.3.5 *Seja X uma curva projetiva não singular sobre $\overline{F}_q$. Definimos o gênus g de X por:*

$$g := l(W),$$

onde $W = (w)$ denota o divisor canônico da diferencial w.

Teorema 3.3.2 *Se X é uma curva projetiva não singular de grau d em $\mathbb{P}^2$, então*

$$g = \frac{1}{2}(d-1)(d-2).$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Apresentaremos agora um resultado fundamental no desenvolvimento da teoria dos códigos algébrico-geométricos.

Teorema 3.3.3 *(Teorema de Riemann-Roch) Seja D um divisor em uma curva projetiva não singular com gênus g. Então para um divisor canônico W,*

$$l(D) - l(W - D) = \deg(D) - g + 1.$$

Corolário 3.3.1 *Seja D um divisor da curva X com gênus g e seja* $\deg(D) > 2g - 2$. *Então,*

$$l(D) = \deg(D) - g + 1.$$

Seja w uma diferencial e P um ponto de grau m em X sendo u um parâmetro local em P. Então existe uma função racional f tal que $w = fdu$. f é dada por

$$\sum_{i=\rho}^{\infty} a_i u^i,$$

onde $a_i \in F_{q^m}$, $\rho = v_P(w)$ e $a_P \neq 0$. O resíduo de w em P é por definição $\mathrm{Tr}(a_{-1})$ e é denotado por $\mathrm{Res}_P(w)$, onde Tr é a transformação traço de F_{q^m} em F_q definida pela soma

$$T_m(\beta) = \beta + \beta^q + \beta^{q^2} + \ldots + \beta^{q^{m-1}} = \sum_{j=0}^{m-1} \beta^{q^j}$$

onde $\beta \in F_{q^m}$. Assim,

Teorema 3.3.4 *Se w é a diferencial da curva projetiva não singular X então*

$$\sum_{P \in X} Res_P(w) = 0.$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Capítulo 4

Códigos RS Generalizados, Alternantes e de Goppa

4.1 Introdução

Esse capítulo tem por finalidade apresentar uma introdução aos códigos geométricos. Para chegarmos a estes de forma natural, faremos algumas considerações iniciais.

Na Seção 4.2, retomaremos a definição de códigos RS vista no Capítulo 2 e apresentaremos a definição de códigos RS estendidos. A Seção 4.3 fornece uma breve introdução sobre códigos RS geométricos.

A Seção 4.4 trata dos códigos RS generalizados que serão utilizados na Seção 4.5 a fim de construir os códigos alternantes. Na Seção 4.6, apresentaremos a subclasse mais importante dos códigos alternantes dada pelos códigos de Goppa.

4.2 Códigos Reed-Solomon Estendidos

Iniciaremos esta seção retomando a definição de códigos de Reed-Solomon vista no Capítulo 2. Assim

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

Definição 4.2.1 *Um código RS é um código BCH de comprimento* $n = q - 1$ *sobre* $GF(q)$ *e polinômio gerador* $g(x) = \prod_{i=l}^{l+\delta-2} (x - \alpha^i)$, *onde* α *é um elemento primitivo de* $GF(q)$ *e* δ *é a distância de projeto.*

A dimensão desse código será $k = n - \delta + 1$ e sua distância mínima, $d_{\min} = \delta$. Observando a relação entre seus parâmetros, temos que os códigos Reed-Solomon são separáveis à máxima distância (ou MDS).

Apresentaremos agora a definição do polinômio Mattson-Solomon (ou polinômio MS), denotado por $A(z)$, que será utilizado da seguinte forma:

Seja $a = (a_0, a_1, \ldots, a_{n-1})$, $a_i \in GF(q^m)$, $i = 0, \ldots, n-1$ um vetor com representação polinomial

$$a(x) = a_0 + a_1 x + \ldots a_{n-1} x^{n-1}.$$

Associaremos o polinômio MS ao vetor a. Para isso, consideremos $GF(q)$ e sua extensão $GF(q^m)$, com $\alpha \in GF(q^m)$ uma raiz n-ésima primitiva da unidade e passemos a seguinte definição:

Definição 4.2.2 *O polinômio MS associado ao vetor* $a = (a_0, a_1, \ldots, a_{n-1})$; $a_i \in GF(q^m)$ *é dado por* $A(z) = \sum_{j=1}^{n} A_j z^{n-j}$ *em* $GF(q^m)[z]$, *onde*

$$A_j = a(\alpha^j) = \sum_{i=0}^{n-1} a_i \alpha^{ij},$$

$j = 0, \pm 1, \pm 2, \ldots,$. *Alternativamente,* $A(z) = \sum_{j=0}^{n-1} A_{-j} z^j = \sum_{i=0}^{n-1} a_i \sum_{j=0}^{n-1} (\alpha^{-i} z)^j$.

Assim, seja C um código RS como na Definição 4.2.1. Se $u = (u_0, u_1, \ldots, u_{k-1})$; $u_i \in GF(q)$, é a mensagem a ser codificada, seja

$$u(z) = \sum_{i=0}^{k-1} u_i z^i.$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

A palavra-código correspondente a u será dada pelo vetor c, cujo polinômio MS é $nu(z)$, onde $n = q - 1$. Assim,

$$c = (u(1), u(\alpha), \ldots, u(\alpha^{n-1})). \tag{4.1}$$

(Mais adiante, veremos que a generalização da Equação em (4.1) originará os códigos RS generalizados).

Temos que $c \in C$ pois $c(x) = \sum_{i=0}^{n-1} c_i x^i$ tem $\alpha, \alpha^2, \ldots, \alpha^{\delta-1}$ como raízes. A justificativa desse fato é a seguinte:

O polinômio MS de c é dado por

$$nu(z) = \sum_{j=0}^{n-1} A_{-j} z^j$$

onde $A_{-j} = c(\alpha^{-j})$. Pela Definição 4.2.2 e usando que $n = -1$ em $GF(q)$, temos: $c(1) = -u_0, c(\alpha^{-1}) = -u_1, \ldots, c(\alpha^{-k+1}) = -u_{k-1}$ e $c(\alpha^j) = 0$, para $1 \leq j \leq n - k = \delta - 1$. Então, $c \in C$ e a Equação em (4.1) representa um método de codificação da mensagem $u(z)$.

A Equação em (4.1) nos sugere um novo modo de olharmos um código RS. Observe que $u(\alpha^i)$ é o resto quando a mensagem $u(x)$, com grau menor que k, é dividida por $M^{(i)}(x) = x - \alpha^i$.

Podemos então reformular a Equação em (4.1) dizendo que $u(x)$ será codificada em $(\mathrm{Res}_0, \mathrm{Res}_1, \ldots, \mathrm{Res}_{n-1})$, onde Res_i é o resíduo de $u(x)$ módulo $M^{(i)}(x)$. O fato de que existe exatamente um polinômio $u(x)$ que é codificado em $(\mathrm{Res}_0, \mathrm{Res}_1, \ldots, \mathrm{Res}_{n-1})$ é garantido pelo seguinte:

Teorema 4.2.1 *(Teorema Chinês do Resto para polinômios) Sejam $m_0(x), \ldots, m_{k-1}(x)$ polinômios sobre $GF(q)$ relativamente primos entre si e seja $M(x) = m_0(x) m_1(x) \ldots m_{k-1}(x)$. Se $\mathrm{Res}_0(x), \mathrm{Res}_1(x), \ldots, \mathrm{Res}_{k-1}(x)$ são polinômios quaisquer sobre $GF(q)$,*

então existe apenas um polinômio $u(x)$ *com* $\partial u(x) < \partial M(x)$ *tal que*

$$u(x) \equiv Res_i(x) \pmod{m_i(x)}$$

para todo $i = 0, \ldots, k-1$.

Assim, pelo teorema anterior, $\mathrm{Res}_0, \mathrm{Res}_1, \ldots, \mathrm{Res}_{k-1}$ são suficientes para encontrarmos $u(x)$ e $\mathrm{Res}_k(x), \ldots, \mathrm{Res}_{n-1}(x)$ não resultam de redundância que serão incluídos na palavra-código. Todo código desse tipo será dito código residual redundante.

Definição 4.2.3 *Seja* C *um código de comprimento* n. *Definimos o código estendido* $\overline{C}$ *como sendo:*

$$\overline{C} = \left\{ (c_1, c_2, \ldots, c_n, c_{n+1}) \mid (c_1, \ldots, c_n) \in C, \ \sum_{i=1}^{n+1} c_i = 0 \right\}.$$

O seguinte teorema nos fornece os parâmetros de $\overline{C}$.

Teorema 4.2.2 *Seja* C *um código RS com parâmetros* $[n = q-1, k, \delta]$ *e polinômio gerador* $g(x) = (x-\alpha)(x-\alpha^2)\ldots(x-\alpha^{\delta-1})$. *Então introduzindo em cada palavra-código* $c = (c_1, c_2, \ldots, c_n)$ *de* C *a componente* c_{n+1} *dada por*

$$c_{n+1} = -\sum_{i=1}^{n} c_i,$$

produziremos um código com parâmetros $[n+1, k, \delta+1]$.

Demonstração: Seja C o código definido acima e $c = (c_1, c_2, \ldots, c_n) \in C$, uma palavra-código com peso de Hamming $w(c) = \delta$. O peso de Hamming de c será $\delta + 1$ desde que

$$-c_{n+1} = \sum_{i=1}^{n} c_i = c(1) \neq 0.$$

Como $c(x) = a(x)\, g(x)$, temos que $c(1) = a(1)\, g(1)$. Observe que $g(1) \neq 0$.

Além disso, $a(1) \neq 0$ ou então $c(x)$ é um dos múltiplos de $(x-1)\, g(x)$, cada um dos quais com peso $\geq \delta + 1$ pelo limitante dos códigos BCH. □

Os códigos RS descritos acima são ditos códigos RS estendidos.

Exemplo 4.2.1 *Tome* $GF(4) = \{0, 1, \alpha, \beta = \alpha^2\}$, *com* $\alpha^2 + \alpha + 1 = 0$.

Um código RS sobre $GF(4)$ *com comprimento* $n = 3$, *distância de projeto* $\delta = 2$ *e* $k = 2$ *tem* $g(x) = x - \beta$ *como polinômio gerador.*

Temos um total de 16 *palavras-código:*

$$\begin{array}{cccc} 000 & 1\alpha 0 & \beta 0\alpha & \beta\alpha 1 \\ 01\alpha & \alpha\beta 0 & 10\beta & 111 \\ 0\alpha\beta & \beta 10 & 1\beta\alpha & \alpha\alpha\alpha \\ 0\beta 1 & \alpha 01 & \alpha 1\beta & \beta\beta\beta \end{array}$$

Para conseguirmos o código RS estendido $[4, 2, 3]$, *aplicamos o Teorema 4.2.2 ao código acima e obtemos:*

$$\begin{array}{cccc} 0000 & 1\alpha 0\beta & \beta 0\alpha 1 & \beta\alpha 10 \\ 01\alpha\beta & \alpha\beta 01 & 10\beta\alpha & 1111 \\ 0\alpha\beta 1 & \beta 10\alpha & 1\beta\alpha 0 & \alpha\alpha\alpha\alpha \\ 0\beta 1\alpha & \alpha 01\beta & \alpha 1\beta 0 & \beta\beta\beta\beta \end{array}$$

Passemos agora a uma nova descrição para os códigos RS estendidos.

Para isso, seja k a dimensão e $n = q$ o comprimento, o qual corresponde ao número de elementos de $GF(q)$, dados por $\alpha_i = \alpha^i$, para $0 \leq i \leq q-2$ e $\alpha_{q-1} = 0$.

Denotaremos por L o conjunto de todos os polinômios de grau menor do que k em $GF(q)[x]$, o anel dos polinômios com coeficientes em $GF(q)$.

Definimos o código C de comprimento $n = q$ por:

$$C := \{(f(\alpha_0), f(\alpha_1), \ldots, f(\alpha_{q-1})) \mid f \in L\}. \tag{4.2}$$

Uma palavra-código de C será da forma $c = (c_0, c_1, \ldots, c_{q-1})$, com $c_i = f(\alpha_i)$ e $f(x) = \sum_{j=0}^{k-1} a_j x^j$, onde $a_i \in GF(q)$. Note que C é linear e tem q^k palavras-código.

Como os polinômios em L têm grau menor que k, estes possuem no máximo $k-1$ raízes e assim, teremos $w_{\min}(C) \geq n - k + 1$. Portanto C será um código MDS.

4.3 Introdução aos Códigos RS Geométricos

Apresentaremos agora uma introdução aos códigos RS geométricos. Para isto, faremos algumas considerações iniciais.

Seja F_q um corpo finito e $\overline{F}_q$ seu fecho algébrico. Considere a reta projetiva $\mathbb{P}^1$ sobre $\overline{F}_q$ dada pelas coordenadas homogêneas (x, y), isto é, (x, y) e $(\lambda x, \lambda y)$ denotam o mesmo ponto sempre que $\lambda \in F_q$ e $\lambda \neq 0$. Os pontos em $\mathbb{P}^1$ com coordenadas em F_q são ditos pontos racionais e são da forma $P_i = (\alpha_i, 1)$, $\alpha_i \in F_q$, $0 \leq i \leq q-1$ e $Q = (1, 0)$ (chamado de ponto no infinito).

Uma função racional em $\mathbb{P}^1$ tem a forma $\frac{a(x,y)}{b(x,y)}$ onde $a(x, y)$ e $b(x, y)$ são polinômios homogêneos de mesmo grau. Um pólo de função racional ocorre quando $b(x, y)$ é zero num ponto.

Seja $\mathfrak{L}$ o espaço vetorial das funções racionais em $\mathbb{P}^1$ que estejam definidas em cada P_i com coeficientes em F_q e que tenham como restrição o fato de não possuirem pólos em $\mathbb{P}^1$ exceto possivelmente em Q. No caso de haver um pólo em Q este deverá ter ordem menor que k.

Definimos o código C sobre os pontos racionais $P_0, \ldots, P_{q-1}$ de $\mathbb{P}^1$ por:

$$C := \{(f(P_0), f(P_1), \ldots, f(P_{q-1})) \mid f \in \mathfrak{L}\}.$$

Em resumo, tomamos um conjunto de q pontos em $\mathbb{P}^1$ e um espaço vetorial de funções. O código é então formado por q-uplas que são os valores dessas funções em cada P_i. Observe que a restrição feita sobre os pólos das funções racionais $\frac{a(x,y)}{b(x,y)}$ faz com que $b(x, y) = y^l$, $l \leq k$ e portanto $a(x, y)$ será um polinômio homogêneo de grau l e C define

o mesmo código que o dado em (4.2).

4.4 Códigos RS Generalizados

Considere agora o corpo $GF(q^m)$ e escolha n elementos distintos dele, digamos $\alpha_1, \alpha_2, \ldots, \alpha_n$. Então se $v = (v_1, v_2, \ldots, v_n)$ é um vetor em $GF(q^m)^n$ com todas suas coordenadas não nulas e se definirmos $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ temos

Definição 4.4.1 *O código Reed-Solomon generalizado (ou $GRS_k(\alpha, v)$) tem como palavras-código todos os vetores da forma $(v_1 f(\alpha_1), v_2 f(\alpha_2), \ldots, v_n f(\alpha_n))$, onde f percorre o conjunto dos polinômios de grau menor que k em $GF(q^m)[x]$.*

Assim, o código $GRS_k(\alpha, v)$ terá parâmetros $[n, k, d]$ sobre $GF(q^m)$, onde $d = n-k+1$ uma vez que cada f tem no máximo $k-1$ raízes, sendo portanto um código MDS.

Quanto ao código dual de um código $GRS_k(\alpha, v)$ temos o seguinte resultado:

Teorema 4.4.1 *O código dual de um código $GRS_k(\alpha, v)$ é o código $GRS_{n-k}(\alpha, y)$, para algum y.*

Demonstração: Vamos supor $k = n-1$ e denotar por D, o código dual do código $GRS_k(\alpha, v)$. Temos então que $n-k = 1$, ou seja, a dimensão de D é 1 e este será formado por todos os múltiplos escalares de algum vetor $y = (y_1, y_2, \ldots, y_n)$. Mostraremos que $y_i \neq 0$ para $i = 1, 2, \ldots, n$.

Temos que

$$\begin{aligned}
&v_1 y_1 + \ldots + v_n y_n = 0 \\
&\alpha_1 v_1 y_1 + \ldots + \alpha_n v_n y_n = 0 \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
&\alpha_1^{n-2} v_1 y_1 + \ldots + \alpha_n^{n-2} v_n y_n = 0
\end{aligned}$$

Na forma matricial

$$\begin{bmatrix} 1 & \cdots & 1 \\ \alpha_1 & \cdots & \alpha_n \\ \vdots & \cdots & \vdots \\ \alpha_1^{n-2} & \cdots & \alpha_n^{n-2} \end{bmatrix} \begin{bmatrix} v_1 y_1 \\ \vdots \\ \vdots \\ v_n y_n \end{bmatrix} = 0. \tag{4.3}$$

Se nas equações acima, algum dos y_i for zero, então os demais produtos da forma $v_i y_i$ também o são. Como por definição, $v_i \neq 0$ para todo i, segue que $y_i = 0$, para $i = 1, 2, \ldots, n$, o que é impossível.

Então das Equações em (4.3), $GRS_k(\alpha, v)$ é dual de $GRS_{n-k}(\alpha, y)$ para todo $k < n-1$, pois:

$$\sum_{i=1}^{n} (\alpha_i^s v_i)(\alpha_i^t y_i) = \sum_{i=1}^{n} \alpha_i^{s+t} v_i y_i = 0$$

para $s \leq k-1$, $t \leq n-k-1$, por (4.3). □

Segue do teorema anterior que a matriz verificação de paridade do código $GRS_k(\alpha, v)$ é igual a matriz geradora do seu código dual $GRS_{n-k}(\alpha, y)$, dada por

$$\begin{aligned} H &= \begin{bmatrix} y_1 & y_2 & \cdots & y_n \\ \alpha_1 y_1 & \alpha_2 y_2 & \cdots & \alpha_n y_n \\ \alpha_1^2 y_1 & \alpha_2^2 y_2 & \cdots & \alpha_n^2 y_n \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_1^{r-1} y_1 & \alpha_2^{r-1} y_2 & \cdots & \alpha_n^{r-1} y_n \end{bmatrix} = \\ &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_1^{r-1} & \alpha_2^{r-1} & \cdots & \alpha_n^{r-1} \end{bmatrix} \begin{bmatrix} y_1 & 0 & \cdots & 0 \\ 0 & y_2 & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ \vdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & y_n \end{bmatrix} = XY \end{aligned} \tag{4.4}$$

onde $y = (y_1, \ldots y_n)$, com $y_i \in GF(q^m)$ e $r = n - k$.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

4.5 Códigos Alternantes

Os códigos alternantes estão estreitamente relacionados com os códigos $GRS_k(\alpha, v)$ definidos na Seção 4.4. Assim

Definição 4.5.1 *Um código alternante $A(\alpha, y)$ é formado por todas as palavras-código de um código $GRS_k(\alpha, v)$ que tenham componentes sobre $GF(q)$, isto é, $A(\alpha, y)$ é a restrição do código $GRS_k(\alpha, y)$ a $GF(q)$.*

Assim, $A(\alpha, y)$ é formado por todos os vetores a sobre $GF(q)$, tais que $Ha^t = 0$, onde H é a matriz verificação de paridade do código $GRS_k(\alpha, v)$, a qual aparece nas Equações em (4.4).

A matriz verificação de paridade H' com elementos de $GF(q)$, pode ser obtida ao substituirmos cada entrada de H por um vetor coluna de comprimento m sobre $GF(q)$.

Da definição acima e tomando $r = n - k$, temos que $A(\alpha, y)$ é um código com parâmetros $[n, K, d]$, onde $n - mr \leq K \leq n - r$ e sua distância mínima d segue do seguinte resultado:

Teorema 4.5.1 *$A(\alpha, y)$ tem distância mínima $d \geq r + 1$, onde $r = n - k$.*

Demonstração: Suponha que a seja uma palavra-código em $A(\alpha, y)$, com $w(a) \leq r$.

Então $Ha^t = XYa^t = 0$ e denotemos $b^t = Ya^t$. Temos que $w(b) = w(a)$, uma vez que Y é uma matriz diagonal e invertível. Segue que $Xb^t = 0$, uma contradição, pois X é uma matriz de Vandermonde.

Logo, para a distância mínima d de um código $A(\alpha, y)$, temos $d > r$. □

Existem códigos alternantes longos que são bons. Essa informação é suportada pelo seguinte

Teorema 4.5.2 *a) Dados n, h e δ, seja m um número tal que $m \mid n - h$. Então se,*

$$\sum_{w=1}^{\delta-1} (q-1)^w \binom{n}{w} < (q^m - 1)^{\frac{(n-h)}{m}},$$

existe um código alternante $A(\alpha, y)$ *sobre* $GF(q)$ *com parâmetros* $[n, K > h, d_{\min} \geq \delta]$.

b) Existem códigos alternantes longos que atingem o limitante de Gilbert-Varshamov.

Demonstração:

i) Seja a um vetor sobre $GF(q)$. Para valores k e α fixos, temos que o número de códigos $GRS_k(\alpha, v)$ contendo a, quando v é variável, é no máximo $(q^m - 1)^k$.

Essa afirmação segue da definição do código $GRS_k(\alpha, v)$, uma vez que $a \in GRS_k(\alpha, v)$ se $a_i = v_i f(\alpha_i)$, $i = 1, \ldots, n$; ou então, $\frac{a_i}{v_i} = f(\alpha_i)$ para algum polinômio $f(z)$ de grau menor que k. Como $f(z)$ é determinado pelos valores destes k ou menos pontos, podemos escolher no máximo k $v_i's$ para determinarmos $f(z)$. Existem $q^m - 1$ possibilidades para cada um dos $v_i's$.

ii) Considere agora uma família $\mathfrak{F}$ de códigos alternantes $A(\alpha, y)$, que são por definição, restrições de códigos $GRS_k(\alpha, v)$ a $GF(q)$, onde $k = n - (n - h)/m$.

Então $A(\alpha, y) \in \mathfrak{F}$, tem dimensão $K \geq n - m(n - k) = h$ e por (i), o número de $A(\alpha, y) \in \mathfrak{F}$ que contêm a, é no máximo $(q^m - 1)^{n-(n-h)/m}$.

Portanto, o número de $A(\alpha, y) \in \mathfrak{F}$ com distância mínima menor do que δ, é no máximo

$$(q^m - 1)^{n-(n-h)/m} \sum_{w=1}^{\delta-1} (q-1)^w \binom{n}{w}. \tag{4.6}$$

iii) O total de códigos $A(\alpha, y)$ em $\mathfrak{F}$ é igual ao número de possibilidades para v, que é $(q^m - 1)^n$. Se este resultado for maior que o resultado da Equação em (4.6), então existe um código $A(\alpha, y)$ com dimensão maior ou igual a h e distância mínima maior ou igual a δ, donde segue (a).

Assintoticamente, quando $n \to \infty$ e h/n é fixo, teremos o Limitante de Gilbert-Varshamov. □

O Teorema 4.5.2 não diz quais são os melhores códigos alternantes, mas apenas que eles existem. Como esses códigos formam uma classe muito grande, apresentaremos a seguir uma subclasse dos mesmos conhecida como códigos de Goppa.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

4.6 Códigos Goppa

Para chegarmos aos códigos de Goppa vamos considerar um polinômio $g(z)$ com coeficientes em $GF(q^m)$ para m fixo que chamaremos de polinômio de Goppa e um subconjunto $L = \{\gamma_0, \ldots, \gamma_{n-1}\}$ de $GF(q^m)$ tal que $g(\gamma_i) \neq 0$, para todo $\gamma_i \in L$.

Agora seja $c = (c_0, \ldots, c_{n-1})$ um vetor sobre $GF(q)$ para o qual associamos a seguinte função racional

$$R_c(z) = \sum_{i=0}^{n-1} \frac{c_i}{z - \gamma_i}.$$

Com isso

Definição 4.6.1 *Seja $L = \{\gamma_0, \gamma_1, \ldots, \gamma_{n-1}\}$ um subconjunto de $GF(q^m)$ com n elementos e seja $g(z)$ um polinômio mônico em $GF(q^m)[z]$ tal que $g(\gamma_i) \neq 0$, $0 \leq i \leq n-1$. O código de Goppa $\Gamma(L, g)$, com polinômio de Goppa $g(z)$, é o conjunto de palavras-código $(c_0, c_1, \ldots, c_{n-1})$ em $GF(q)^n$ tais que*

$$R_c(z) = \sum_{i=0}^{n-1} \frac{c_i}{z - \gamma_i} \equiv 0 \pmod{g(z)}. \tag{4.7}$$

Observe que na Definição 4.6.1 o fato de $R_c(z) \equiv 0 \pmod{g(z)}$ é equivalente a dizer que $R_c(z) \equiv 0$ no anel dos polinômios $GF(q^m)/g(z)$.

Obteremos a matriz verificação de paridade do código $\Gamma(L, g)$ da seguinte forma:

No anel dos polinômios $\bmod g(z)$ temos que $z - \gamma_i$ não divide $g(z)$ e portanto tem inversa dada por

$$(z - \gamma_i)^{-1} = -\frac{g(z) - g(\gamma_i)}{z - \gamma_i} g(\gamma_i)^{-1},$$

assim

$$-(z - \gamma_i)^{-1} \frac{g(z) - g(\gamma_i)}{z - \gamma_i} g(\gamma_i)^{-1} \equiv 1 \pmod{g(z)}.$$

Com isso temos que uma palavra-código $c \in \Gamma(L, g)$ se e somente se

$$\sum_{i=0}^{n-1} c_i \frac{g(z) - g(\gamma_i)}{z - \gamma_i} g(\gamma_i)^{-1} = 0. \tag{4.8}$$

unesp

Se $g(z) = \sum_{i=0}^{r} g_i z^i$ com $g_i \in GF(q^m)$ e $g_r \neq 0$, então

$$\frac{g(z) - g(\gamma_i)}{z - \gamma_i} = g_r\left(z^{r-1} + z^{r-2}\gamma_i + \ldots + \gamma_i^{r-1}\right) + g_{r-1}\left(z^{r-2} + \ldots + \gamma_i^{r-2}\right) + \tag{4.9}$$
$$+ \ldots + g_2(z + \gamma_i) + g_1.$$

Igualando os coeficientes de $z^{r-1}, z^{r-2}, \ldots, 1$ a zero nas Equações em (4.8) temos que $c \in \Gamma(L, g)$ se e somente se $Hc^t = 0$ onde

$$H = \begin{bmatrix} g_r g(\gamma_1)^{-1} & \cdots & g_r g(\gamma_n)^{-1} \\ (g_{r-1} + \gamma_1 g_r)\, g(\gamma_1)^{-1} & \cdots & (g_{r-1} + \gamma_n g_r)\, g(\gamma_n)^{-1} \\ \vdots & & \vdots \\ \left(g_1 + \gamma_1 g_2 + \ldots + \gamma_1^{r-1} g_r\right) g(\gamma_1)^{-1} & \cdots & (g_1 + \gamma_n g_2 + \ldots + \gamma_n^{r-1} g_r)\, g(\gamma_n)^{-1} \end{bmatrix}$$

$$= \begin{bmatrix} g_r & 0 & \cdots & 0 \\ g_{r-1} & g_r & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ g_1 & g_2 & \cdots & g_r \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \gamma_1 & \gamma_2 & \cdots & \gamma_n \\ \vdots & \vdots & & \vdots \\ \gamma_1^{r-1} & \gamma_2^{r-1} & \cdots & \gamma_n^{r-1} \end{bmatrix}$$

$$\cdot \begin{bmatrix} g(\gamma_i)^{-1} & \cdots & \cdots & 0 \\ \vdots & g(\gamma_2)^{-1} & & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \ldots & g(\gamma_n)^{-1} \end{bmatrix} = CXY \tag{4.10}$$

é a matriz verificação de paridade de $\Gamma(L, g)$.

Da forma como foi definida a matriz H e da definição de $\Gamma(L, g)$ temos que um código de Goppa é um código alternante $A(\gamma, y)$ com $\gamma = (\gamma_1, \ldots, \gamma_n)$ e $y = (g(\gamma_1)^{-1}, \ldots, g(\gamma_n)^{-1})$ e portanto $\Gamma(L, g)$ terá parâmetros n, $k \geq n - rm$ e $d_{\min} \geq r + 1$, onde r é o grau de g.

Capítulo 5

Códigos Sobre Curvas Algébricas

5.1 Introdução

Esse capítulo tem por finalidade apresentar os códigos algébrico-geométricos.

Na Seção 5.2 construiremos códigos sobre curvas algébricas de uma forma elementar, utilizando apenas polinômios e pontos de uma curva plana. O processo de decodificação será apresentado na Seção 5.3 e é dado por um algoritmo simples que de certa forma resume-se apenas na solução de um sistema de equações lineares.

A construção dos códigos algébrico-geométricos apontam uma maneira sistemática para a obtenção de bons códigos, sendo possível encontrar sequências destes códigos sobre $GF(q)$ que superam o limitante de Gilbert-Varshamov. O comportamento assintótico dos códigos geométricos será discutido na Seção 5.4. Apenas mencionamos o resultado que garante a existência de bons códigos algébrico-geométricos uma vez que para obtê-lo em detalhes necessitamos de resultados profundos de Geometria Algébrica, fugindo portanto ao objetivo deste trabalho, o qual consiste principalmente em descrever o que seria uma introdução aos processos de codificação e decodificação dos códigos algébrico-geométricos.

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

5.2 Códigos Sobre Curvas Algébricas

Nesta seção retornaremos à teoria dos códigos utilizando o jargão e a terminologia de geometria algébrica, apresentados no Capítulo 3.

Seja X uma curva regular irredutível de grau m sobre F_q e $P_1, P_2, \ldots, P_n$ seus pontos racionais (ou seja , $P_i = (x_i : y_i : z_i)$, com $X(x_i, y_i, z_i) = 0$, onde x_i, y_i, z_i estão em F_q).

Denotaremos por $P_1', P_2', \ldots, P_n'$ estes mesmos pontos racionais usando a seguinte representação: a primeira coordenada não nula de P_i é 1 para $1 \leq i \leq n$.

Tome $j < q$ e seja V_j o espaço vetorial sobre F_q formado pelo polinômio nulo e por todos os polinômios homogêneos com grau j e coeficientes em F_q.

Definimos agora dois códigos lineares sobre F_q, dados por:

$$C_X(j) = \{(f(P_1'), \ldots, f(P_n')) \mid f \in V_j\}$$

e

$$H_X(j) := C_X(j)^{\perp}.$$

Passaremos agora à determinação dos parâmetros desses códigos, sendo que para isso, utilizaremos o seguinte resultado de geometria algébrica.

Teorema 5.2.1 *(Teorema de Bezout [4, p. 112]) Sejam X e Y curvas planas projetivas de graus m e n, respectivamente. Assumindo que X e Y não possuem componentes em comum, então o número de pontos em $X \cap Y$ (contados com multiplicidades) é mn.*

Teorema 5.2.2 *Seja $C_X(j)$ um código linear como definido anteriormente. A distância mínima $d_{\min}$ de $C_X(j)$ satisfaz:*

$$d_{\min} \geq n - mj.$$

Supondo $n > mj$, se $j < m$, então a dimensão de $C_X(j)$ é

$$k = \binom{j+2}{2};$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

caso $j \geq m$, *então*

$$k = \binom{j+2}{2} - \binom{j-m+2}{2}.$$

Demonstração: Tome $f \in V_j$. Se X é fator de f, então a palavra-código correspondente a f será 0. Agora, se f e X não possuírem componentes em comum, então pelo Teorema de Bezout haverá no máximo mj pontos em comum entre as curvas. Logo teremos no máximo mj componentes nulas em uma palavra-código e portanto $n - mj$ é o peso de Hamming mínimo de uma palavra-código em $C_X(j)$. Assim $d_{\min} \geq n - mj$.

Quanto à dimensão, observe primeiramente que V_j é um espaço vetorial formado por polinômios homogêneos de grau j. Uma base para V_j é então formada por todos os monômios da forma $x^{j_1}y^{j_2}z^{j_3}$, onde $j_1 + j_2 + j_3 = j$.

Assim,

$$\dim V_j = \binom{j+2}{2}. \tag{5.1}$$

Considerando a transformação linear α dada por

$$\begin{array}{rccl} \alpha: & V_j & \to & C_X(j) \\ & f & \mapsto & \alpha(f) = (f(P_1'), \ldots, f(P_n')) \end{array} \tag{5.2}$$

e como $n > mj$, segue novamente do Teorema de Bezout que $\alpha(f) = 0$ (palavra toda nula) se, e somente se, X é fator de f.

Assim se $j < m$, $X \nmid f, \forall f \in V_j$ e segue das Equações em (5.1) que $k = \binom{j+2}{2}$. No caso de $j \geq m$, $\alpha(f) = 0$ se e somente se, $X \mid f$, ou seja, $\exists\, g \in V_j$ com $\partial g = j - m$ tal que $f = gX$ e a dimensão do $\ker(\alpha)$ é $\binom{j-m+2}{2}$. Como $\dim V_j = \dim \ker(\alpha) + \dim \mathrm{Im}(\alpha)$ temos que $k = \dim V_j - \dim \ker(\alpha)$. Portanto, de (5.1) temos que $k = \binom{j+2}{2} - \binom{j-m+2}{2}$. □

Podemos calcular para a curva regular X o gênero g que é dado por $g = (1/2)(m-1)$

de Biociências, Letras e Ciências Exatas - UNESP
BIBLIOTECA
S. J. Rio Preto

$(m-2)$. Assim, no Teorema 5.2.2 temos que se $j \geq m$, então:

$$\begin{aligned} k &= \binom{j+2}{2} - \binom{j-m+2}{2} = \\ &= (1/2)(j+2)(j+1) - (1/2)(j-m+2)(j-m+1) \\ &= mj - (1/2)(m-1)(m-2) + 1 \\ &= mj - g + 1 \end{aligned} \tag{5.3}$$

Apresentaremos agora uma outra representação para os códigos algébrico-geométricos. Para isso seja X uma curva regular irredutível, $P_1, P_2, \ldots, P_n$ e Q seus pontos racionais e tome um inteiro m satisfazendo $2g-2 < m < n$, onde g é o gênero de X.

Definimos o código C sobre F_q por

$$C := \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathfrak{L}(mQ)\}$$

onde $G = mQ$ é um divisor da curva X. O espaço $\mathfrak{L}(G) = \mathfrak{L}(mQ)$ é por definição formado pelo conjunto de funções f para as quais a ordem de f em cada ponto P de X é $\geq -m$. Dessa forma temos que $\mathfrak{L}(G) = \{0\}$ se o grau de G é negativo [4, p. 196].

Com isso temos que a palavra-código nula em C corresponde a uma função $f \in \mathfrak{L}\left(G - \sum_{i=1}^{n} P_i\right)$ e portanto $f = 0$. Concluímos então que a dimensão do código C é a mesma do espaço $\mathfrak{L}(G)$ que pelo Teorema de Riemman-Roch é

$$l(G) = k = m - g + 1.$$

Da mesma forma, se uma palavra-código em C tem peso w, ou seja, $f(P_i) = 0$ em $n-w$ posições, sua função f correspondente estará no espaço $\mathfrak{L}(D)$ onde D é um divisor de grau $m-n+w$ e a distância mínima d de C satisfaz

$$d \geq n - m.$$

Se $\phi_0 = 1, \phi_1, \ldots, \phi_{k-1}$ denota uma base de $\mathfrak{L}(G)$ então

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \phi_1(P_1) & \phi_1(P_2) & \cdots & \phi_1(P_n) \\ \vdots & \vdots & & \vdots \\ \phi_{k-1}(P_1) & \phi_{k-1}(P_2) & \cdots & \phi_{k-1}(P_n) \end{pmatrix}$$

é uma matriz geradora do código C.

Seja $s = m - 2g + 1$ e considere s colunas de $\mathbf{G}$ correspondendo aos pontos $P_{i_1}, \ldots, P_{i_s}$. Pelo Teorema de Riemman-Roch temos que

$$l\left(mQ - \sum_{\nu=1}^{k-1} P_{i_\nu}\right) = m - s - g + 1 = g.$$

Com isso temos que $(\lambda_0, \ldots, \lambda_{k-1})$ são soluções das equações dadas por

$$\sum_{j=0}^{k-1} \lambda_j \phi_j(P_{i_\nu}) = 0, \ \nu = 1, \ldots, s$$

e geram um espaço de dimensão g, ou seja, a matriz $\mathbf{G}$ tem posto s o que é equivalente a dizer que qualquer conjunto de s colunas de $\mathbf{G}$ são L.I.

Podemos então concluir que o código dual $C^\perp$ de C tem distância mínima $d' \geq s + 1$ e dimensão $k' = n - k$.

Quanto ao comprimento dos códigos $C_X(j)$ e $H_X(j)$, apresentaremos a seguir um limitante superior para o número de pontos racionais da curva X.

Limitante de Serre *[6]*

Se denotarmos por n o comprimento dos códigos $C_X(j)$ e $H_X(j)$, onde X é uma curva absolutamente irredutível, temos:

$$n \leq q + 1 + g\lfloor 2\sqrt{q} \rfloor,$$

onde g é o gênus da curva X e $\lfloor x \rfloor$ denota o maior inteiro que não supera x.

Agora, quanto aos parâmetros do código $H_X(j)$, temos:

Teorema 5.2.3 *Seja X uma curva regular, com $n > mj$ e $j \geq m-2$. A distância mínima $d_{\min}$ do código $H_X(j)$ satisfaz:*

$$d_{\min} \geq mj - 2g + 2.$$

Demonstração: Considere o subespaço $I = \{f \in V_j \mid X \text{ é um fator de } f\}$. O espaço vetorial V_j/I tem dimensão $k = mj - g + 1$ de acordo com o Teorema 5.2.2 e a Equação em (5.3) e se $\{[f_0], \ldots, [f_{k-1}]\}$ é uma base desse espaço vetorial, então

$$G = \begin{bmatrix} f_0(P'_1) & \cdots & f_0(P'_n) \\ f_1(P'_1) & \cdots & f_1(P'_n) \\ \vdots & & \vdots \\ f_{k-1}(P'_1) & \cdots & f_{k-1}(P'_n) \end{bmatrix}$$

é uma matriz geradora para $C_X(j)$. Seja $s = mj - 2g + 1$. Mostremos que quaisquer s colunas de G são L.I..

Para isso, escolha f_0 de forma que a ordem de f_0 em $P_i \leq 1$, $i = 1, \ldots, n$. Temos que $X \nmid f_0$ e pelo Teorema de Bezout o número de pontos em $f_0 \cap X$ é mj. Logo a curva f_0 intercepta X num divisor G com grau mj.

Seja $\mathfrak{L}(G)$ o espaço vetorial associado a G, cuja dimensão é $l(G) = mj - g + 1$, de acordo com o Teorema de Riemann-Roch. Esta é a mesma dimensão do espaço vetorial V_R com base $\{[f_i]/[f_0] \mid i = 0, \ldots, k-1\}$ e como $V_R \subseteq \mathfrak{L}(G)$, temos $V_R = \mathfrak{L}(G)$.

Se tomarmos agora $P_{i_1}, \ldots, P_{i_s}$, s pontos racionais e considerarmos o divisor $G_1 = G - P_{i_1} - \ldots - P_{i_s}$, temos que $\deg(G_1) = mj - s = 2g - 1 > 2g - 2$ e do Teorema de Riemann-Roch, a dimensão de $\mathfrak{L}(G_1)$ é $l(G_1) = l(G) - s = g$.

Um elemento $[f_i]/[f_0] \in V_R = \mathfrak{L}(G)$ pertence a $\mathfrak{L}(G_1)$ se, e somente se, $f_i(P_{i_j}) = 0$, $j = 1, \ldots, s$, pois para cada P_{i_j}, estamos considerando os casos $f_0(P_{i_j}) = 0$ e $f_0(P_{i_j}) \neq$

0 e usando que $f_0(P_{i_j}) = 0$ se e somente se $ord_{P_{i_j}}(f_0) = 1$, temos que as soluções $(\lambda_0, \ldots, \lambda_{k-1})$ das equações

$$\lambda_0 f_0(P_{i_j}) + \ldots + \lambda_{k-1} f_{k-1}(P_{i_j}) = 0,$$

$j = 1, \ldots, s$, formam um espaço vetorial de dimensão g, ou seja, o posto de G é $k - g = s$.

Com isso, temos que qualquer conjunto de s colunas de G é L.I.. Com relação à distância mínima de $H_X(j)$, temos que esta é maior ou igual a $s + 1$. □

Exemplo 5.2.1 *Seja* $f(X, Y, Z) = X^3Y + Y^3Z + Z^3X = 0$ *sobre* $GF(2^3)$.

O gênus $g = (1/2)(m-1)(m-2) = 3$. *Seja* n *o número de pontos racionais na curva, dado pelo limitante de Serre. Então* $n \leq 24$.

Devemos então encontrar os 24 *pontos em* $GF(2^3)$ *na curva, que serão os pontos* $(1,0,0)$, $(0,1,0)$, $(0,0,1)$ *e mais* 21 *pontos da forma* $(1, \beta, \gamma)$ *onde* β *é um elemento não nulo de* $GF(2^3)$ *e* γ *é qualquer uma das soluções de* $z^3 + \beta^3 z + \beta = 0$. *Fazendo* $z = \beta^5 w$, *temos* $\beta(w^3 + w + 1) = 0$, *equação esta que tem três soluções diferentes.*

Do Teorema 5.2.2, tomando $j = 2, 3, 4, 5$ *temos que* $C_f(2)$ *é o código com parâmetros* $(24, 6, 16)$, $C_f(3)$ *é dado por* $(24, 10, 12)$, $C_f(4)$ *é* $(24, 14, 8)$ *e* $C_f(5)$ *será* $(24, 18, 4)$.

Quanto aos seus códigos duais $H_f(j)$, *teremos como parâmetros:* $(24, 18, 4)$; $(24, 14, 8)$; $(24, 10, 12)$; $(24, 6, 16)$, *respectivamente, de acordo com o Teorema 5.2.3.*

5.3 Decodificação

5.3.1 Preliminares

Na seção anterior, definimos os códigos $C_X(j)$ e $H_X(j)$, onde X é uma curva regular de grau m sobre F_q com gênus g, sendo seus pontos racionais dados por $P'_1, P'_2, \ldots, P'_{n_1}$ onde a primeira coordenada não nula de cada P'_i, $1 \leq i \leq n_1$, é igual a 1. Para o código $H_X(j)$, denotaremos por $\delta = mj - 2g + 2$, a sua distância de projeto (ou seja, δ é um limitante superior para a distância mínima do código $H_X(j)$ de acordo com o Teorema

5.2.3).

Dentre os pontos racionais P_i' de X, $1 \leq i \leq n_1$, tomaremos apenas aqueles que possuam a primeira coordenada não-nula, ou seja, pontos da forma $(1, y_j, z_j)$. Estes pontos serão denotados por $Q_1, Q_2, \ldots, Q_n$ e sobre esses pontos definimos

$$C_X'(j) = \{(f(Q_1), \ldots, f(Q_n)) \mid f \in V_j\}$$

e

$$H_X'(j) = C_X'(j)^{\perp}.$$

Da mesma forma como definimos os parâmetros para os códigos $C_X(j)$ e $H_X(j)$, faremos para os códigos $C_X'(j)$ e $H_X'(j)$, onde m é o grau da curva X e g seu gênus.

De acordo com a Equação em (5.3), o número $k(j)$, $(j \geq m-2)$ é definido por:

$$k(j) = mj - g + 1.$$

Dos Teoremas 5.2.2 e 5.2.3 da seção anterior e de como foi definido o código $H_X'(j)$ segue

Teorema 5.3.1 *Suponha $n > mj$ e $j \geq m-2$. Então o código $H_X'(j)$ tem parâmetros $(n, n-k(j), d_{\min})$, onde n é o comprimento do código, $n-k(j)$ é sua dimensão e sua distância mínima satisfaz*

$$d_{\min} \geq mj - 2g + 2 = k(j) - g + 1.$$

Exemplo 5.3.2 *Seja X uma curva sobre $GF(q)$ dada pela equação*

$$x^{r+1} + y^{r+1} + z^{r+1} = 0,$$

onde $q = r^2$. Tome $q = 16$, logo $m = 5$ e $g = 6$ e para $3 \leq j \leq 11$, os códigos $H_X(j)$ terão parâmetros $n_1 = 65$, $n_1 - k = 5j-5$ e $d_{\min} = 5j-10$. Dentre os 65 pontos racionais

da curva X, cinco possuem a primeira coordenada igual a zero. Logo, se denotarmos por $Q_1, Q_2, \ldots, Q_{60}$, os pontos que têm primeira coordenada igual a 1 , podemos encontrar os parâmetros dos códigos $H'_X(j)$, $3 \leq j \leq 11$ de acordo com o Teorema 5.3.1. Calculando para os casos $j = 7$, $j = 9$ e $j = 11$ obteremos respectivamente códigos com parâmetros $(60, 30, 25)$, $(60, 20, 35)$ *e* $(60, 10, 45)$.

Considere agora um código $H'_X(j)$ satisfazendo o Teorema 5.3.1 e sejam $f_0, \ldots, f_{s-1}$ polinômios da forma $x^{j_1} y^{j_2} z^{j_3}$, tais que $j_1 + j_2 + j_3 = j$, onde $s = \begin{pmatrix} j+2 \\ 2 \end{pmatrix}$. Segue que

$$\mathbf{H} = \{f_i(Q_l)\}, \quad i = 0, \ldots, s-1; \quad l = 1, \ldots, n$$

é a matriz verificação de paridade para $H'_X(j)$.

Como os pontos Q_l são da forma $(1, y_l, z_l)$, consideraremos apenas os polinômios nas variáveis y e z, ou seja, vamos substituir os polinômios f_i por polinômios com a seguinte representação:

$$\varphi_i(y, z) = \varphi_{i_1, i_2}(y, z) = y^{i_1} z^{i_2}; \quad i_1 + i_2 \leq j,$$

onde os pares (i_1, i_2); $i_1 + i_2 \leq j$, serão arranjados numa determinada seqüência, obedecendo a ordenação de Cauchy para multiplicação de séries, ou seja, teremos os pares (i_1, i_2); $i_1 + i_2 \leq j$, arranjados na forma de uma seqüência do tipo $(0,0), (1,0), (0,1), (2,0), (1,1), (0,2), \ldots$, etc.

Um método para obtenção dessa seqüência é dado pela seguinte transformação:

$$\tau(i_1, i_2) = i,$$

onde $\tau(i_1, i_2) = (1/2)(i_1 + i_2 + 1)(i_1 + i_2) + i_2$.

Para encontrarmos sua inversa $\tau^{-1}(i)$, devemos resolver as seguintes inequações:

$$\begin{pmatrix} p \\ 2 \end{pmatrix} \leq i < \begin{pmatrix} p+1 \\ 2 \end{pmatrix}$$

e teremos $i_1 + i_2 = p - 1$ e $i_2 = i - \binom{p}{2}$.

Por exemplo, se tivermos $i_1 = 2$ e $i_2 = 0$, o valor de i será dado por $\tau(2,0) = (1/2)(2+1)(2) = 3$. Agora, se quisermos saber os valores de i_1 e i_2 quando $i = 3$, tome $p = 3$ e assim $\binom{p}{2} \leq 3 < \binom{p+1}{2}$, que resultará em $3 \leq 3 < 12$. Substituindo em $i_2 = i - \binom{p}{2}$, temos $i_2 = 0$. Assim, $i_1 + i_2 = p - 1$ o que implica que $i_1 = 2$. Logo, $\tau^{-1}(3) = (2,0)$.

Assim, se denotarmos $Q'_l(y_l, z_l)$, $l = 1, \ldots, n$, reescreveremos a matriz $\mathbf{H}$ da seguinte forma:

$$\overline{\mathbf{H}} = \{\varphi_i(Q'_l)\}$$

$i = 0, \ldots, s-1$, $l = 1, \ldots, n$.

5.3.2 Procedimento de Decodificação

Passemos agora ao processo de decodificação para o código $C''_X(j)$ definido anteriormente. Para isso, suponha que $v \in C''_X(j)$ seja uma palavra-código enviada, r o vetor recebido onde $r = v + e$ e o vetor erro e seja não nulo em t posições, satisfazendo $t \leq \frac{\delta - 1}{2}$ onde $\delta = mj - 2g - 2$ é a distância de projeto de $H'_X(j)$. Assim $w(e) = t$, ou seja, o peso de Hamming do vetor erro é t.

Calculando a síndrome S da maneira usual, temos $S = Hr^t = He^t$. Os elementos de S são conhecidos e serão denotados da seguinte forma

$$S_{l_1,l_2} = S_l = \sum_{i \in I} e_i y_i^{l_1} z_i^{l_2}, \quad \tau(l_1, l_2) = l, \; l = 0, \ldots, s-1 \tag{5.4}$$

onde s é o número de linhas da matriz H, e_i denota o valor do erro na posição i e $I = \{i_1, i_2, \ldots, i_t\}$ é o conjunto dos índices que indicam as t posições onde ocorreram os erros.

A idéia inicial para um procedimento de decodificação é determinar um polinômio localizador $\sigma(y,z)$ que entre suas raízes tenha os pontos Q_i', $i \in I$. Interceptando a curva dada por $\sigma(x,y) = 0$ e a curva que define o código (tomando $x = 1$) encontraremos as posições dos erros. Então ao inserirmos esses pontos na Equação em (5.4) poderemos calcular os valores dos c_i.

O exemplo a seguir tem por objetivo ilustrar essa idéia inicial antes de passarmos a uma descrição detalhada de um algoritmo de decodificação para os códigos algébrico-geométricos.

Exemplo 5.3.3 *Seja $F = X^3Y + Y^3Z + Z^3X = 0$ a curva dada no Exemplo 5.2.1. Essa curva possui 22 pontos racionais sobre $GF(8)$ com a primeira coordenada não nula e então para $j = 3$ o código $H_f'(3)$ tem parâmetros $(22, 12, 8)$ podendo corrigir até t erros, onde $t \leq \left\lfloor \frac{\delta-1}{2} \right\rfloor$ e $\delta = 8$, ou seja, poderemos corrigir até 3 erros.*

Pela Equação em (5.4) denotaremos as equações da síndrome por

$$\sum_{i=1}^{3} c_i y_i^{l_1} z_i^{l_2} = S_{l_1,l_2}, \quad l_1 + l_2 \leq 3.$$

Suponha que a partir de um vetor recebido tenhamos obtido os seguintes valores para a síndrome $S_{00} = \alpha^5$, $S_{10} = \alpha$, $S_{01} = \alpha^6$, $S_{20} = \alpha^6$, $S_{11} = \alpha$, $S_{02} = \alpha^5$, $S_{30} = \alpha^5$, $S_{21} = 0$, $S_{12} = \alpha^6$ e $S_{03} = \alpha^2$.

Agora se um polinômio localizador linear $\sigma(y,z)$ satisfaz as equações da síndrome. Então

$$(\sigma_{00} + \sigma_{10}y + \sigma_{01}z)\, S_{l_1,l_2} = 0$$

e

$$\sigma_{00}S_{l_1,l_2} + \sigma_{10}S_{l_1,l_2}y + \sigma_{01}S_{l_1,l_2}z = 0.$$

Como por (5.4) $S_{l_1,l_2} = \sum_{i=1}^{3} c_i y_i^{l_1} z_i^{l_2}$, então

$$yS_{l_1,l_2} = \sum_{i=1}^{3} c_i y_i^{l_1+1} z_i^{l_2} = S_{l_1+1,l_2}$$

e da mesma forma $zS_{l_1,l_2} = S_{l_1,l_2+1}$. *Assim*

$$\sigma_{00}S_{l_1,l_2} + \sigma_{10}S_{l_1,l_2}y + \sigma_{01}S_{l_1,l_2}z = \sigma_{00}S_{l_1,l_2} + \sigma_{10}S_{l_1+1,l_2} + \sigma_{01}S_{l_1,l_2+1} = 0.$$

Neste caso as síndromes não satisfazem essas relações, ou seja, o polinômio localizador não poderá ser linear e devemos então adicionar um termo de segunda ordem ao polinômio localizador obtendo:

$$\sigma(y,z) = y^2 + \sigma_{10}y + \sigma_{01}z + \sigma_{00}.$$

Da mesma forma que fizemos para o caso linear faremos agora, por exemplo, para S_{00} *e* $\sigma(y,z)$ *de grau 2. Assim*

$$\left(y^2 + \sigma_{10}y + \sigma_{01}z + \sigma_{00}\right) S_{00} = 0$$

e

$$S_{00}y^2 + \sigma_{10}S_{00}y + \sigma_{01}S_{00}z + \sigma_{00}S_{00} = 0$$

de onde vem que

$$S_{20} + \sigma_{10}S_{10} + \sigma_{01}S_{01} + \sigma_{00}S_{00} = 0.$$

Essas relações pode ser obtidas utilizando um sistema da forma

$$\begin{bmatrix} S_{00} & S_{10} & S_{01} \\ S_{10} & S_{20} & S_{11} \\ S_{01} & S_{11} & S_{02} \end{bmatrix} \begin{bmatrix} \sigma_{00} \\ \sigma_{10} \\ \sigma_{01} \end{bmatrix} = \begin{bmatrix} S_{20} \\ S_{30} \\ S_{21} \end{bmatrix}$$

onde os valores das síndromes são conhecidos. Uma solução para esse sistema é da forma $\sigma(y,z) = y^2 + \alpha^2 y + \alpha^5 z$ *onde* α *é uma raiz de* $x^3 + x + 1$.

Os pontos pertencentes a interseção de f *e* σ *são* $(1,0,0), (1,1,\alpha), (1,\alpha,1), (1,\alpha,^3\alpha^3)$ *e* $(1,\alpha^4,1)$ *e expandindo as equações da síndrome de forma a incluir esses pontos obte-*

remos $c_{i_1} = 1$, $c_{i_2} = \alpha^2$, $c_{i_3} = 0$, $c_{i_4} = \alpha$ *e* $c_{i_5} = 0$.

Voltaremos agora a descrição do processo de decodificação.

O problema da decodificação é então determinar a posição $i \in I$ do erro e seu valor c_i. Para solucionar esse problema, construiremos uma matriz $\overline{S}$ e para isso, assumimos a existência de um número h, tal que,

$$t + 1 \leq k(h) \leq \delta \quad q - t,$$

onde $m - 2 \leq h \leq j - m + 2$. Tomemos $h' = j - h$ e façamos $h_1 = (1/2)(h+1)(h+2)$ e $h'_1 = (1/2)(h'+1)(h'+2)$. Com isso definimos

$$\overline{S} = [a_{l,r}], \ \ l = 0, \ldots, h'_1 - 1; \ \ r = 0, \ldots, h_1 - 1$$

onde

$$a_{l,r} = S_{l_1+r_1, l_2+r_2}, \tag{5.5}$$

com $\tau(l_1, l_2) = l$ e $\tau(r_1, r_2) = r$.

Lema 5.3.1 *Para quaisquer* l *e* r *dados pela Equação em (5.5), temos* $l_1 + r_1 + l_2 + r_2 \leq j$.

Demonstração: Usando (5.5) e a definição de τ temos que $l_1 + l_2 \leq h'$ e $r_1 + r_2 \leq h$.

Como $h' = j - h$, então $h' + h = j$, o que implica que $l_1 + r_1 + l_2 + r_2 \leq h' + h = j$.□

Consideremos agora três matrizes:

$$F = [b_{l,u}]; \ \ l = 0, \ldots, h'_1 - 1, \ u = 1, \ldots, t$$

onde $b_{l,u} = y_{i_u}^{l_1} z_{i_u}^{l_2}$ com $\tau(l_1, l_2) = l$,

$$E = [c_{u,v}]; \ \ u = 1, \ldots, t, \ v = 1, \ldots, t,$$

onde $c_{u,u} = c_{i_u}$ e $c_{u,v} = 0$, se $u \neq v$ e

$$P = [d_{v,r}]; \quad v = 1, \ldots, t, \ r = 0, \ldots, h_1 - 1,$$

onde $d_{v,r} = y_{i_v}^{r_1} z_{i_v}^{r_2}$, com $\tau(r_1, r_2) = r$.

Temos então que $\overline{S} = FEP$, pois uma entrada da matriz $[FEP]_{h_1' \times h_1}$ na posição (l, r) será da forma

$$\sum_{j=1}^{t} \left(d_{j,r} \sum_{i=1}^{t} b_{l,i} c_{i,j} \right).$$

Lembrando que $c_{i,j} = 0$ se $i \neq j$ temos

$$d_{1,r}(b_{l,1}c_{1,1}) + d_{2,r}(b_{l,2}c_{2,2}) + \ldots + d_{t,r}(b_{l,t}c_{t,t})$$

Substituindo os valores de acordo com as definições das matrizes P, E e F respectivamente, temos:

$$\begin{aligned} y_{i_1}^{r_1} z_{i_1}^{r_2} \left(y_{i_1}^{l_1} z_{i_1}^{l_2} c_{i_1} \right) + y_{i_2}^{r_1} z_{i_2}^{r_2} \left(y_{i_2}^{l_1} z_{i_2}^{l_2} c_{i_2} \right) + \ldots + y_{i_t}^{r_1} z_{i_t}^{r_2} \left(y_{i_t}^{l_1} z_{i_t}^{l_2} c_{i_t} \right) &= \sum_{v=1}^{t} c_{i_v} y_{i_v}^{l_1 + r_1} z_{i_v}^{l_2 + r_2} \\ &= S_{l_1 + r_1, l_2 + r_2} \end{aligned}$$

Veja também que ao considerarmos o código $H_X(h')$, temos que F é formada por t colunas da matriz verificação de paridade desse código, onde essas t colunas correspondem aos pontos $(1, y_{i_v}, z_{i_v})$, $v = 1, \ldots, t$. Como a distância mínima desse código é $d_{\min} \geq mh' - \partial g + 2$ (de acordo com o Teorema 5.2.2) e pelo fato de $k(h') \geq t + g$, temos:

$$t < k(h') - g + 1 = mh' - \partial g + 2$$

e então o posto da matriz F é t.

Consideremos agora o seguinte sistema linear de equações

$$\overline{S}\sigma = 0. \tag{5.6}$$

Se $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_{h_1-1})$ é uma solução para as Equações em (5.6), denotemos por $\sigma(y,z)$ o polinômio em duas variáveis cujo grau é menor ou igual a h e $\sigma_{ab} = \sigma_{\tau(a,b)}$, como sendo

$$\sigma(y,z) = \sum \sigma_{ab} y^a z^b, \tag{5.7}$$

onde $\tau(a,b) \in \{0, \ldots, h_1 - 1\}$.

Lema 5.3.2 *Seja σ uma solução para as Equações em (5.6) e seja (y_{i_v}, z_{i_v}) um ponto, onde $i_v \in I$ e $e_{i_v} \neq 0$. Então $\sigma(y_{i_v}, z_{i_v}) = 0$.*

Demonstração: Como $S = FEP$, temos $S\sigma = F(EP\sigma)$ e como o posto da matriz F é t, então uma solução para as Equações em (5.6) satisfaz

$$EP\sigma = 0 \tag{5.8}$$

A componente na posição v na Equação em (5.8) será então dada por

$$\begin{aligned} \sum_{s=0}^{q-1} e_{i_v} d_{v,s} \sigma_s &= e_{i_v} \sum_s y_{i_v}^{s_1} z_{i_v}^{s_2} \sigma_s, \ \tau(s_1, s_2) = s \\ &= e_{i_v} \sigma(y_{i_v}, z_{i_v}) = 0. \end{aligned}$$

Logo, $\sigma(y_{i_v}, z_{i_v}) = 0$. □

Lema 5.3.3 *O Sistema Linear em (5.6) sempre tem uma solução σ tal que o polinômio correspondente $\sigma(y,z)$ não tem a curva C como divisor.*

Demonstração: Para um polinômio $\sigma(y,z)$ em (5.6), teremos:

$$\sigma(x,y,z) = \sum \sigma_{ab} x^{h-a-b} y^a z^b,$$

onde $\tau(a,b) \in \{0, \ldots, h_1 - 1\}$. Dessa forma, dizer que a curva $X(x,y,z)$ não é divisor de $\sigma(y,z)$, significa dizer que $X(x,y,z)$ não é divisor de $\sigma(x,y,z)$.

O sistema $\overline{S}\sigma = 0$ tem h_1 incógnitas, onde $\overline{S}$ tem posto no máximo t. Então em $h_1 - (t+1)$ posições, haverá uma solução própria para o sistema o qual é zero nessas

posições.

Considere agora o termo $x^{m-a-b}y^a z^b$ na equação da curva X, tal que $a+b$ seja o maior possível e suponha que $X(x,y,z)$ seja um divisor de $\sigma(x,y,z)$. Então, em $\sigma(x,y,z)$, deverão haver termos da forma $P(x,y,z)\,x^{m-a-b}y^a z^b$, com grau de $P(x,y,z)$ igual a $h-m$.

Veja que haverão no máximo $\begin{pmatrix} h-m+2 \\ 2 \end{pmatrix}$ termos desse tipo em $\sigma(y,z)$. Assim, se os coeficientes desses termos forem nulos, então $\sigma(y,z)$ não terá X como divisor. Logo o lema segue se

$$\begin{pmatrix} h-m+2 \\ 2 \end{pmatrix} \leq h_1 - (t+1).$$

Então

$$\begin{pmatrix} n+2 \\ 2 \end{pmatrix} - \begin{pmatrix} h-m+2 \\ 2 \end{pmatrix} \geq t+1,$$

ou seja, $k(h) \geq t+1$ o que ocorre pela definição de h. □

Com o exposto até agora, chegamos a um processo de decodificação o qual está resumido no seguinte.

Algoritmo de Decodificação

1º Passo: Para um vetor recebido v, calcule a síndrome $S = Hv^t$.

2º Passo: Considere a matriz $\overline{S}$ definida em (5.5). Encontre a solução σ para o Sistema em (5.6) cujo polinômio correspondente $\sigma(y,z)$ não tenha a curva X como divisor.

3º Passo: Entre os pontos $Q'_1, \ldots, Q'_n$, encontre aqueles para os quais $\sigma(Q'_i) = 0$ e denote-os por $Q''_1, Q''_2, \ldots, Q''_{t'}$.

4º Passo: Insira os valores de $Q''_1, \ldots, Q''_{t'}$ no Sistema Linear em (5.4) e calcule os valores dos e_i.

Teorema 5.3.2 *Este algoritmo só será capaz de corrigir t erros se existir um número h tal que*

$$t+1 \leq k(h) \leq \delta - g - t$$

onde $m-2 \leq h \leq j-m+2$.

Demonstração: Pelo Lema 5.3.2, $\sigma(y,z)$ tem como raízes todos os erros e_i determinados pelo $2^{\underline{0}}$ Passo no algoritmo de decodificação.

Como $\sigma(y,z)$ não tem a curva X como divisor, segue do Teorema de Bezout que para o número t' (dado no $3^{\underline{0}}$ Passo), $t' \leq mh$. Como $k(j-h) \geq t+g$, temos $mj-2g+1 \geq mh+t$ e então $mh \leq mj-2g+2 = \delta$. Assim o sistema considerado no $4^{\underline{0}}$ Passo tem solução única.

Se houvessem duas soluções no $4^{\underline{0}}$ Passo, poderíamos construir uma palavra-código não nula com peso menor que δ somando essas duas soluções, uma vez que as componentes não-nulas dessas palavras se encontram entre as posições $Q''_1, \ldots, Q''_{t'}$. As posições onde $e_i \neq 0$ dão as posições dos erros e e_i os seus valores correspondentes. □

Exemplo 5.3.4 *Tome X como sendo a curva do Exemplo 5.3.2, ou seja,*

$$x^5 + y^5 + z^5 = 0.$$

Para $j=7$ o código $H'_X(7)$ sobre $GF(16)$ terá parâmetros $(60,30,25)$.

Pelo Teorema 5.3.2 temos que $3 \leq h \leq 5$ e tomando $h=3$ o algoritmo de decodificação deverá ser capaz de corrigir até 9 erros.

Suponha que v seja a palavra-código enviada e $r = v+e$ o vetor recebido onde e é o padrão de erro ocorrido.

Calculando a síndrome de maneira usual temos que

$$S_l = Hr^t = \sum_{i \in I} e_i y_i^{l_1} z_i^{l_2} = S_{l_1,l_2},$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

$\tau(l_1, l_2) = l$, $l = 0, \ldots, 29$ *e* $i \in I = \{i_1, i_2, \ldots, i_9\}$, *ou seja, um total de 30 equações independentes.*

Construiremos agora a matriz $\overline{S}$ *cujas entradas são dadas pelos elementos da síndrome (de acordo com (5.5)). Para isso calculamos os seguintes valores obtidos a partir de* $h = 3$

$$\begin{aligned} h' &= j - h = 4 \\ h_1 &= 1/2\,(h' + 1)\,(h' + 2) = 10 \\ h'_1 &= 1/2\,(h + 1)\,(h + 2) = 15. \end{aligned}$$

A matriz $\overline{S}$ *é por definição uma matriz com entradas* $a_{l,r}$, $l = 0, \ldots, h'_1 - 1$; $r = 0, \ldots, h_1 - 1$ *e portanto nesse exemplo* $\overline{S}$ *será uma matriz* (15×10).

De acordo com as Equações em (5.6) e (5.7), se $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_9)$ *é uma solução para o sistema linear* $\overline{S}\sigma = 0$ *então o polinômio* $\sigma(y, z)$ *cujo grau é menor ou igual a 3 é da forma*

$$\sigma(y, z) = \sum \sigma_{ab} y^a z^b,\ \tau(a, b) \in \{0, \ldots, 9\}.$$

Seja α um elemento primitivo de $GF(16)$ e suponha o sistema $\overline{S}\sigma = 0$ dado por

$$\begin{bmatrix}
\alpha^{10} & 0 & \alpha & \alpha^8 & 1 & 0 & \alpha^9 & \alpha^2 & 1 & \alpha^4 \\
0 & \alpha^8 & 1 & \alpha^9 & \alpha^2 & 1 & \alpha & \alpha^{14} & \alpha^9 & \alpha^4 \\
\alpha & 1 & 0 & \alpha^2 & 1 & \alpha^1 & \alpha^{11} & \alpha^9 & \alpha^4 & \alpha^{12} \\
\alpha^8 & \alpha^9 & \alpha^2 & \alpha & \alpha^{14} & \alpha^9 & \alpha^{10} & \alpha^{14} & \alpha^8 & \alpha \\
1 & \alpha^2 & 1 & \alpha^{14} & \alpha^9 & \alpha^4 & \alpha^{14} & \alpha^8 & \alpha & \alpha^5 \\
0 & 1 & \alpha^4 & \alpha^9 & \alpha^4 & \alpha^{12} & \alpha^8 & \alpha & \alpha^5 & 0 \\
\alpha^9 & \alpha & \alpha^{14} & \alpha^{10} & \alpha^{14} & \alpha^8 & \alpha^2 & \alpha^3 & \alpha^2 & \alpha^{14} \\
\alpha^2 & \alpha^{14} & \alpha^9 & \alpha^{14} & \alpha^8 & \alpha & \alpha^3 & \alpha^2 & \alpha^{14} & \alpha^7 \\
1 & \alpha^9 & \alpha^4 & \alpha^8 & \alpha & \alpha^5 & \alpha^2 & \alpha^{14} & \alpha^7 & \alpha^2 \\
\alpha^4 & \alpha^4 & \alpha^{12} & \alpha & \alpha^5 & 0 & \alpha^{14} & \alpha^7 & \alpha^2 & \alpha^9 \\
\alpha & \alpha^{10} & \alpha^{14} & \alpha^2 & \alpha^3 & \alpha^2 & \alpha^3 & \alpha^{11} & \alpha^2 & \alpha^9 \\
\alpha^{14} & \alpha^{14} & \alpha^8 & \alpha^3 & \alpha^2 & \alpha^{14} & \alpha^{11} & \alpha^2 & \alpha^9 & 0 \\
\alpha^9 & \alpha^8 & \alpha & \alpha^2 & \alpha^{14} & \alpha^7 & \alpha^{20} & \alpha^9 & 0 & \alpha^{13} \\
\alpha^4 & \alpha & \alpha^5 & \alpha^{14} & \alpha^7 & \alpha^2 & \alpha^9 & 0 & \alpha^{13} & \alpha^{12} \\
\alpha^{12} & \alpha^5 & 0 & \alpha^7 & \alpha^7 & \alpha^9 & 0 & \alpha^{13} & \alpha^{12} & \alpha^2
\end{bmatrix}
\begin{bmatrix}
\sigma_{00} \\ \sigma_{10} \\ \sigma_{01} \\ \sigma_{20} \\ \sigma_{11} \\ \sigma_{02} \\ \sigma_{30} \\ \sigma_{21} \\ \sigma_{12} \\ \sigma_{03}
\end{bmatrix} = 0$$

Temos uma única solução para esse sistema dado por

$$\sigma = 1 + \alpha^4 y + \alpha^5 z + \alpha^{13} y^2 + \alpha^7 zy + \alpha^5 z^2 + \alpha^{12} y^3 + \alpha^3 y^2 z + \alpha^8 yz^2 + z^3.$$

O grau de σ é 3 e pelo Lema 5.3.3 σ não divide X. Segue então do Teorema de Bezout que haverão 15 pontos em $\sigma \cap X$ (lembrando que agora σ representa o polinômio homogêneo $\sigma(x,y,z) = x^3 + \alpha^4 x^2 y + \alpha^5 x^2 z + \alpha^7 xzy + \alpha^5 xz^2 + \alpha^{12} y^3 + \alpha^3 y^2 z + \alpha^8 yz^2 + z^3$).

Dentre os 15 pontos obtidos tomaremos apenas os pontos da forma (x,y,z) com $x \neq 0$ que são 14 e serão listados na forma (y,z). São eles $(0,1)$, $(0,\alpha^3)$, $(0,\alpha^{12})$, $(1,0)$, (α,α^{14}), (α^2,α^{10}), (α^2,α^7), (α^2,α^{13}), $(\alpha^5,\alpha^{10})(\alpha^{10},\alpha^{14})$, (α^7,α^5), $(\alpha^6,0)$, (α^{13},α^2), $(\alpha^{12},0)$.

Utilizando esses pontos para resolver as 30 componentes independentes das equações da síndrome obteremos os valores dos erros que são não nulos e sua localização. No nosso exemplo α^5, α^8, 1, 0, α^3, α^{10}, 0, α^2, α^8, 0, α^5.

O algoritmo de decodificação para os códigos algébrico-geométricos apresentado, pode ser visto como uma generalização do algoritmo de Peterson para os códigos BCH e sua complexidade está no fato deste envolver a solução de um sistema com grande número de equações lineares.

A capacidade de correção de erros desse algoritmo é limitada pela existência do número h (Teorema 5.3.2) e pelo fato de estarmos considerando apenas os pontos da curva (sobre a qual está definida o código) com primeira coordenada não nula.

5.4 Comportamento Assintótico dos Códigos Algébrico-Geométricos

No Capítulo 1, ao estudarmos entropia e quantidade de informação vimos que se C é um código com parâmetros (n, M, d) sobre F_q e

$$A_q(n,d) = \max\{M \mid \text{existe um código } (n,M,d) \text{ sobre } F_q\},$$

então a quantidade

$$\alpha(\delta) = \lim_{n\to\infty} \sup \frac{\log_q A_q(n,\delta n)}{n},\ 0 \le \delta \le 1$$

é zero ($\alpha(\delta) = 0$), para $1 - \frac{1}{q} \le \delta \le 1$ e o limitante de Gilbert-Varshamov é dado por

$$\alpha(\delta) \ge 1 - H_q(\delta),\ 0 \le \delta \le 1 - \frac{1}{q}$$

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

onde $H_q(x)$ é a função entropia dada por

$$\begin{aligned} H_q(0) &= 0 \\ H_q(x) &= x\log_q(q-1) - x\log_q x - (1-x)\log_q(1-x), \quad 0 < x < 1 - \frac{1}{q}. \end{aligned}$$

A razão de estarmos interessados nos códigos algébrico-geométricos advém do fato destes serem utilizados para obtermos sequências de códigos assintoticamente bons, com parâmetros melhores que os dados pelo limitante de Gilbert-Varshamov. O estudo do comportamento assintótico dos códigos geométricos, apresentado por Tsfasman, Vladut e Zink [5] utiliza a teoria de curvas modulares e mostra que

$$\alpha(\delta) + \delta \geq 1 - \frac{1}{\sqrt{q}-1} \tag{5.9}$$

se q é um quadrado ≥ 49.

Veja que se denotarmos por $C^*(D,G)$ um código geométrico sobre uma curva X com gênus g, dimensão k e distância d, onde D e G representam divisores da curva com suportes disjuntos. Seus principais parâmetros são dados pela taxa $R = \frac{k}{n}$ e pela distância mínima relativa $\delta = \frac{d}{n}$. Nos códigos geométricos, esses parâmetros satisfazem

$$\begin{aligned} R &= \frac{n - \deg(G) + g - 1}{n} \\ \delta &= \frac{\deg(G) - 2g + 2}{n} \end{aligned}$$

Se tomarmos $P_1, \ldots, P_n$, Q, como sendo os pontos racionais da curva X sobre F_q, ou seja, pontos em X de grau 1 e se θ denota um inteiro satisfazendo

$$2g - 2 < \theta < n,$$

então os parâmetros para o código $C^*(D,G)$, $(D = P_1 + \ldots + P_n,\ G = \theta Q)$ satisfazem

$$R + \delta \geq 1 - \frac{g-1}{n}.$$

Defina

$$A(q) = \lim_{g \to \infty} \sup \frac{N(g)}{g}$$

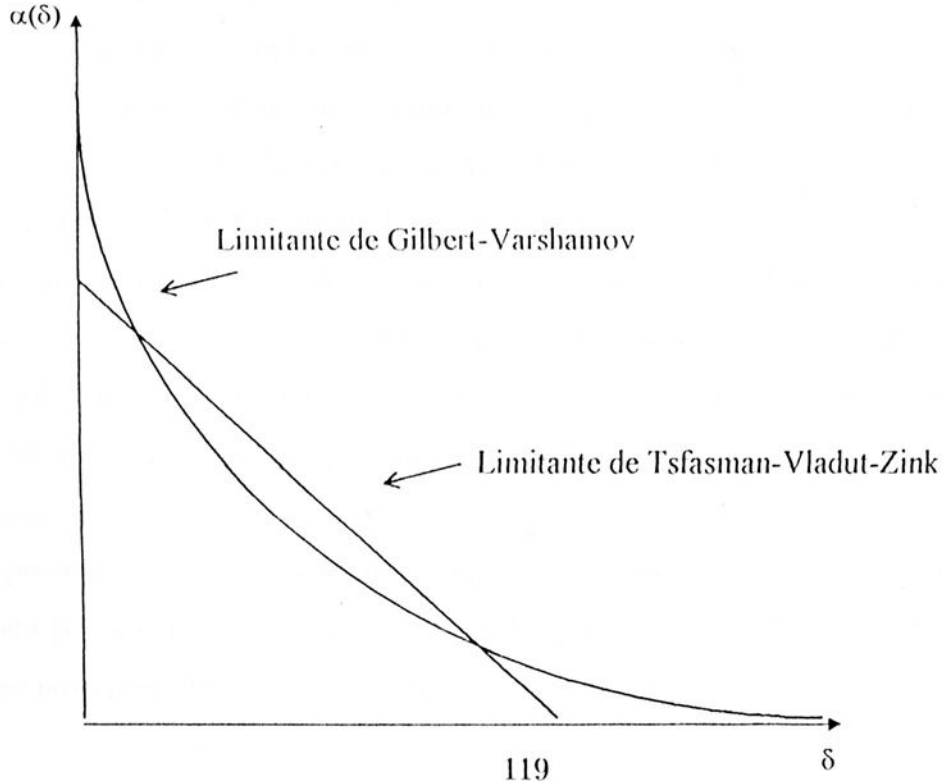
onde $N(g)$ é o número máximo de pontos racionais de uma curva com gênus g sobre F_q e $N \leq q + 1 + 2g\sqrt{q}$ (pelo Limitante de Serre, Seção 5.2). Segue então que $A(q) \leq 2\sqrt{q}$.

Utilizando curvas modulares sobre F_q, onde q é um quadrado maior ou igual a 49 Tsfasman, Vladut e Zink [5] demonstraram que $A(q) = \sqrt{q} - 1$ e com isso mostraram que existe uma sequência de códigos satisfazendo

$$R + \delta \geq 1 - \frac{1}{\sqrt{q}-1}$$

quando q é um quadrado de onde segue a Equação em (5.9).

Podemos representar esse resultado através do seguinte gráfico



CONCLUSÃO

Neste trabalho apresentamos os processos de codificação e decodificação de algumas das principais classes de códigos lineares tendo em vista o seu comportamento assintótico, dado pelo limitante de Gilbert-Varshamov.

Na busca de bons códigos, os melhores resultados são obtidos quando a partir de uma generalização dos códigos Reed-Solomon (os códigos RS são uma introdução natural para a construção dos códigos algébrico-geométricos) e utilizando resultados de geometria algébrica construímos os códigos algébrico-geométricos.

Iniciamos este trabalho estudando códigos lineares, BCH e RS. É a partir dos códigos RS que construímos classes de códigos que atingem o limitante de Gilbert-Varshamov como é o caso dos códigos alternantes e Goppa. Os códigos RS são também o ponto de partida para a construção dos códigos algébricos-geométricos.

Finalizamos esse trabalho com o que seria um primeiro passo na construção de códigos sobre curvas algébricas. Essa construção é obtida de maneira elementar utilizando apenas polinômios e pontos de uma curva plana. O processo de decodificação apresentado é dado por um algoritmo simples que de certa forma resume-se na solução de um sistema de equações lineares e corresponde a uma introdução ao que seria um método eficaz na decodificação dos códigos algébrico-geométricos. Este procedimento pode ser visto como uma generalização do algoritmo de Peterson para os códigos BCH.

Uma construção mais complexa dos códigos algébrico-geométricos utilizando resultados mais profundos de geometria algébrica pode ser obtida tendo como base códigos RS e Goppa, utilizando o fato destes serem códigos residuais redundantes. Esse novo processo de decodificação produz códigos algébrico-geométricos com parâmetros ainda melhores que os apresentados nesse trabalho.

Um processo de decodificação mais elaborado para os códigos algébrico-geométricos é dado em [11] e é obtido a partir da generalização do algoritmo de Berlekamp-Massey [10]. Esse novo procedimento de decodificação é mais eficiente e tem maior capacidade de

cm 1 2 3 4 5 6 7 unesp 11 12 13 14 15 16 17 18

correção de erros que o apresentado nesse trabalho pois explora melhor o fato dos códigos algébrico-geométricos possuirem altas distâncias mínimas entre suas palavras-códigos.

unesp

BIBLIOGRAFIA

[1] R. G. Gallager, *"Information Theory and Reliable Communication"*, John Wiley and Sons, Inc., 1968.

[2] C. E. Shannon, *"A Mathematical Theory of Comunication"*, Bill. Syst. Tech., Journal 27, pp. 379 - 423, 623 - 656, 1948.

[3] F. J. MacWillians and N. J. A. Sloane, *"The Theory of Error-Correcting Codes"*. Amsterdam, the Netherlands: North - Holland, 1977.

[4] W. Fulton, *"Algebraic Curves"*. Reading, MA: Benjamin Cummings, 1969.

[5] M. A. Tsfasman, S. G. Vladut, and T. Zink, *"Modular curves, Shimura curvas and Goppa codes, better than Varshamov-Gilbert bound"*, Math. Nachr, vol. 104, pp. 13 - 28, 1982.

[6] Serre, J. P., *"Sur le nombre des points rationels d'une courbe algébrique sur un corps fini"*, C. R. Acad. Sci. Paris, 296 (1983).

[7] Bombieri, E., *"Counting Points on Curves over Finite Fields"*, Séminaire Bourbaki, 25e anneé, 430 (1972/1973).

[8] J. H. van Lint and T. A. Springer, *"Generalized Reed-Solomon codes from algebraic geometry"*, IEEE Trans. Inform. Theory, vol. II - 33, pp. 305 - 310, May 1987.

[9] J. Justesen, K. J. Larsen, H. Jensen, *"Construction and Decoding of a Class of Algebraic Geometry Codes"*, IEEE Trans. Inform. Theory, vol. 35, pp. 841 - 821, July 1989.

[10] G. L. Feng and K. K. Tzeng, *"A generalization of the Berlekamp-Massey algorithm for multisequence shift-register synthesis with applications to decoding ciclic codes"*, IEEE Trans. Inform. Theory, vol. 37, pp. 1274 - 1287, Sept. 1991.

unesp

[11] G. L. Feng and T. R. N. Rao, *"Decoding Algebraic-Geometric Codes up to the Designed Minimum Distance"*, IEEE Trans. Inform. Theory, vol. 39, pp. 37 - 45, Jan. 1993.


cm 1 2 3 4 5 6 7
unesp
11 12 13 14 15 16 17 18