

Análise dos Algoritmos de Decodificação para
Códigos de Geometria Algébrica sobre Curvas de
Hermite

Leocarlos Bezerra da Silva Lima

Dissertação de mestrado submetida à Coordenação dos Cursos de Pós-Graduação em Engenharia Elétrica da Universidade Federal da Paraíba - Campus II como parte dos requisitos necessários para obtenção do grau de Mestre.

Área de Concentração: Processamento de Informação -
Comunicações

Prof. Dr. Francisco Marcos de Assis
Orientador

Campina Grande, Paraíba, Brasil

©Leocarlos Bezerra da Silva Lima, 16 de agosto de 1999



Ficha Catalográfica

L 732 a LIMA, Leocarlos Bezerra da Silva.

Análise dos Algoritmos de Decodificação para Códigos de Geometria Algébrica sobre Curvas de Hermite. / Leocarlos Bezerra da Silva Lima. – Campina Grande, 1999.

126 p.: il.

Orientador: Francisco Marcos de Assis.

Dissertação (mestrado) – UFPB / CT / COPELE

1. Decodificação; 2. Códigos de Hermite; 3. Codificação para Controle de Erros; 4. Telecomunicações; 5. Geometria Algébrica.

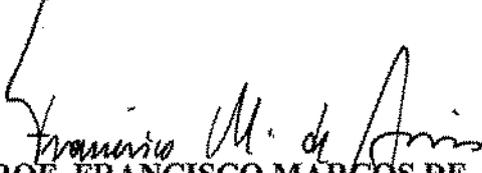
UFPB/BC

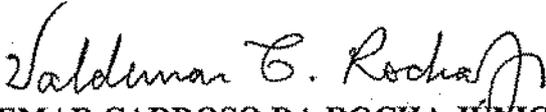
CDU 681.188:621.39 (043.3)

**ANÁLISE DOS ALGORITMOS DE DECODIFICAÇÃO PARA CÓDIGOS DE
GEOMETRIA ALGÉBRICA SOBRE CURVAS DE HERMITE**

LEOCARLOS BEZERRA DA SILVA LIMA

Dissertação Aprovada em 16.08.1999


PROF. FRANCISCO MARCOS DE ASSIS, Dr., UFPB
Orientador


PROF. VALDEMAR CARDOSO DA ROCHA JÚNIOR, Ph.D., UFPE
Componente da Banca


PROF. JOÃO MARQUES DE CARVALHO, Ph.D., UFPB
Componente da Banca

CAMPINA GRANDE - PB
Agosto - 1999

“Ainda que eu falasse línguas, as dos homens e as dos anjos, se eu não tivesse o amor, seria como um bronze que soa ou como um címbalo que tina. Ainda que eu tivesse o dom da profecia, o conhecimento de todos os mistérios e de toda a ciência, ainda que eu tivesse toda a fé, a ponto de transportar montanhas, se não tivesse o amor, eu nada seria. (...) O amor jamais passará. Quanto às profecias, desaparecerão. Quanto às línguas, cessarão. Quanto à ciência, também desaparecerá. Pois o nosso conhecimento é limitado, e limitada é a nossa profecia. (...) Agora o meu conhecimento é limitado, mas, depois, conhecerei como sou conhecido. Agora permanecem fé, esperança e amor. A maior delas, porém, é o amor.”

1Cor 13

Dedicatória

Ao meu avô, Ozório Vieira da Silva (in memoriam), homem forte, que soube enfrentar a vida e a dor com serenidade, simpatia e, principalmente, com amor.

Agradecimentos

À mulher, amiga e companheira, Ana Paula, pelos incentivos e por acreditar em mim.

Aos meus pais, Carlos e Severina, por uma vida, por um exemplo.

Aos meus irmãos, Carlos Júnior, Ana Karla e Ana Carolina, pela paciência, doação e compreensão que tiveram comigo durante o tempo em que estive empenhado na realização deste projeto.

Ao professor Francisco Marcos, por todo apoio e incentivo, sem os quais não poderia realizar este trabalho.

Aos amigos e colegas, Edmar, George, Waslon, Walter, Ronaldo, Bruno, professor Marcelo, e tantos outros, que direta ou indiretamente contribuíram com este trabalho.

Resumo

Na codificação para controle de erros, em sistemas de comunicação, o desenvolvimento mais importante nos últimos anos foi a teoria dos *códigos de geometria algébrica* (CGA's), ou *códigos de Goppa geométricos*. Esta teoria permite se obter códigos com parâmetros bem melhores que os até então conhecidos, e constitui uma abordagem matemática extremamente elegante. Em suma, um CGA de comprimento n consiste na avaliação de funções de um espaço de funções racionais gerado por um divisor G de uma curva algébrica \mathcal{X} , sendo esta avaliação feita sobre um conjunto de n pontos racionais de \mathcal{X} disjunto do suporte de G . Os CGA's baseados em curvas de Hermite apresentam excelentes parâmetros e são bastante usados e referenciados na literatura, tendo sido o presente estudo restringido a estes códigos. A decodificação destes códigos tem sido feita seguindo basicamente duas abordagens: uma, pela solução de um conjunto de equações lineares sobre um corpo de localização, em que as síndromes são definidas como um mapeamento de um subespaço linear de funções neste corpo de localização, e outra, pela solução de uma equação chave em um anel afim, em que as síndromes são definidas como elementos deste anel afim. O algoritmo básico de Skorobogatov e Vlăduț e o algoritmo de Porter são exemplos típicos da primeira e da segunda abordagens, respectivamente. A decodificação rápida, com menor complexidade ($< \mathcal{O}(n^3)$), de CGA's tem sido obtida com o uso do algoritmo BMS de Sakata, principalmente associado ao esquema de decisão por maioria de Feng e Rao. Todos estes esquemas são aqui descritos e analisados.

Abstract

On error-control coding, in communication systems, the most important development in the last years was the theory of *algebraic geometric codes* (AGC's), or *geometric Goppa codes*. This theory allows to get codes with better parameters, and forms an extremely elegant mathematical approach. Briefly, an ACG of length n consists in evaluating functions from a space of rational functions generated by a divisor G of an algebraic curve \mathcal{X} , where this evaluation is made over a set of n rational points of \mathcal{X} disjoint from the support of G . ACG's based on hermitian curves have excellent parameters and are very used and referred in the literature, being the present work restricted to these codes. Its decoding has been made basically following two approaches: solving a set of linear equations over a location field, where syndromes are defined as a map from one linear subspace of rational functions to this location field, and solving a key equation in an affine ring, where syndromes are defined as elements in this affine ring. Skorobogatov and Vlăduț's basic algorithm and Porter's algorithm are typical examples of first and second approaches, respectively. ACGs' fast decoding, with smaller complexity ($< \mathcal{O}(n^3)$), has been got using Sakata's algorithm BMS, mostly with Feng and Rao's majority voting scheme. All of these schemes are described and analysed here.

Lista de Figuras

1.1	Diagrama de blocos funcional de um sistema de comunicação digital genérico.	2
1.2	Mapeamento de k símbolos da fonte em n símbolos da palavra código na codificação de bloco.	3
2.1	Limite de Gilbert-Varshamov para o caso em que $q = 16$	13
3.1	Curvas paralelas em \mathbb{R}^2 encontram-se num único ponto no infinito. Curvas paralelas com inclinação diferente encontram-se em um ponto diferente no infinito.	34
4.1	Comparação do limite de Gilbert-Varshamov (curva GV) com o limite de Tsfasman-Vlăduț-Zink (curva TVZ) para $q = 64$	61

Lista de Tabelas

4.1	Operações “+” e “x” relacionadas ao conjunto $\{1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6, \alpha^7, 0\}$, que constituem o corpo \mathbb{F}_9 .	57
4.2	Pontos racionais da curva de Hermite $y^3 + y = x^4$ em \mathbb{F}_9 .	58
5.1	Organização do espaço decodificável em classes laterais de palavras.	65
5.2	Pontos racionais da curva de Hermite $y^3 + y = x^4$ em \mathbb{F}_9 e respectivas posições na palavra código.	74
5.3	As 11 síndromes da palavra recebida $v = (1 \ \alpha^6 \ 0 \ 0 \ 0 \ \dots \ 0 \ \alpha^5 \ 0 \ 0 \ 0)$, para o código de Hermite C_{13} .	77
5.4	Arranjo S bi-dimensional de elementos $S_{i,j} \in \mathbb{F}_9$ utilizado como entrada para o algoritmo BMS. Estes elementos de \mathbb{F}_9 são, na verdade, as síndromes da palavra recebida que está sendo utilizada nos exemplos dos algoritmos de decodificação. Os elementos * representam síndromes desconhecidas.	91
5.5	Ilustração dos cantos externos do conjunto delta $\Delta^+ = \{(1,0)\}$. Os pontos de $\text{Ext } \Delta^+$ estão representados por símbolos o. Observe que estes pontos constituem realmente cantos na tabela.	92
5.6	Saída do algoritmo BMS para o arranjo S de entrada.	94
5.7	Polinômios envolvidos no processamento do algoritmo BMS neste exemplo, com suas respectivas extensões e discrepâncias.	95
5.8	Pontos racionais da curva \mathcal{X} em \mathbb{F}_9 e respectivas posições na palavra código.	108
5.9	Arranjo S bi-dimensional das síndromes da palavra recebida v obtido no passo 1 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt. Os elementos * representam síndromes desconhecidas.	109

5.10	Arranjo S bi-dimensional de síndromes obtido no passo 2 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt após utilização do esquema de decisão por maioria. Os elementos * representam síndromes desconhecidas.	110
5.11	Saídas do algoritmo BMS durante o processamento dos passos 1 e 2 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt.	111
5.12	Polinômios envolvidos no processamento do algoritmo BMS nos passos 1 e 2 do algoritmo de Sakata. Justesen, Madelung, Jensen e Høholdt.	112
5.13	Arranjo S bi-dimensional de síndromes completo obtido após o passo 3 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt.	112

Índice

1	Introdução	1
1.1	O sistema de comunicação digital	1
1.2	A codificação para controle de erros	3
1.2.1	Os códigos de geometria algébrica	4
1.2.2	A decodificação dos CGA's	4
1.3	Estrutura do texto	6
2	Codificação para Controle de Erros	7
2.1	Conceitos básicos	7
2.1.1	Códigos de bloco	8
2.1.2	Geometria dos códigos	9
2.1.3	Códigos lineares	10
2.1.4	Matrizes dos códigos	10
2.1.5	Limites dos códigos	12
2.2	Códigos de Reed-Solomon	12
2.2.1	Definição convencional	13
2.2.2	Definição geométrica	14
2.3	Códigos de Goppa	16
2.3.1	Códigos BCH	16
2.3.2	Códigos de Goppa	17
2.4	Transição para os CGA's	19
3	Tópicos de Geometria Algébrica	22
3.1	Ideais e variedades	22
3.1.1	Variedade afim	23

3.1.2	Ideal	24
3.1.3	Bases de Gröbner	25
3.1.4	Anel de coordenadas de uma variedade afim	32
3.1.5	Corpo de funções	33
3.1.6	Variedade projetiva	33
3.1.7	Relação entre variedades afim e projetiva	37
3.2	Anel local	38
3.2.1	Anel de valorização	38
3.2.2	Relação entre anel de valorização discreto e variedade	40
3.3	Divisores	40
3.3.1	Conceito de divisor	40
3.3.2	Espaço de funções de um divisor	42
3.3.3	Diferencial	43
3.3.4	Teorema de Riemann-Roch	44
3.3.5	Lacunas e anti-lacunas	45
4	Códigos de Geometria Algébrica	47
4.1	Definição dos CGA's	47
4.1.1	Construção por funções	48
4.1.2	Construção por diferenciais	49
4.2	Códigos de Hermite	51
4.2.1	Curvas de Hermite	51
4.2.2	Código de Hermite	52
4.2.3	Exemplo de código de Hermite	56
4.3	Novo limite dos códigos	59
5	Decodificação dos Códigos de Geometria Algébrica	62
5.1	O problema da decodificação	62
5.2	Decodificação dos códigos de Hermite	66
5.2.1	Breve histórico	66
5.2.2	Abordagens na decodificação dos CGA's	68
5.3	Primeira abordagem na decodificação	69
5.3.1	Algoritmo básico de Skorobogatov e Vlăduț	69
5.3.2	Exemplo de decodificação de um código de Hermite	73

5.4	Segunda abordagem na decodificação	78
5.4.1	Algoritmo de Porter, Shen e Pellikaan	78
5.5	Decodificação rápida	83
5.5.1	Algoritmo BMS	84
5.5.2	Decisão por maioria	94
5.5.3	Algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt	100
5.5.4	Exemplo de decodificação de um código de Hermite	107
6	Conclusões	113
A	Álgebra de Corpo Finito	116
A.1	Grupos	116
A.2	Anéis	117
A.3	Corpos	118
A.4	Corpos finitos baseados em anéis de inteiros	118
A.5	Corpos finitos baseados em anéis de polinômios	119
B	Dedução do Limite de Gilbert-Varshamov	121
B.1	Primeira parte	121
B.2	Segunda parte	123
B.3	Terceira parte	124

Capítulo 1

Introdução

A partir de meados deste século, tem-se testemunhado um crescimento explosivo na utilização de sistemas de comunicação digital, que se fazem presentes hoje em inúmeras aplicações, tais como: sistemas de comunicação móvel celular, HDTV, comunicações por satélite, sistemas de armazenamento de dados, redes de computadores, dentre outros. É evidente, portanto, a importância do estudo e desenvolvimento de sistemas digitais cada vez melhores e mais eficientes.

1.1 O sistema de comunicação digital

A figura 1.1 ilustra o diagrama funcional com os elementos básicos de um sistema de comunicação digital [16].

O *codificador de fonte* realiza a compressão ou compactação eficiente dos dados fornecidos pela fonte (sinal analógico ou digital), retirando as redundâncias desnecessárias existentes na informação original. Em outras palavras, ele converte o sinal digital ou analógico da fonte em um sinal digital sem redundâncias, mais “compacto”. No caso de um sinal da fonte analógico, realiza sua quantização, implicando uma perda controlada de informação, e caracterizando uma compressão do sinal.

O *codificador de canal*, por sua vez, introduz redundâncias controladas ao sinal, o que possibilita a detecção e correção no receptor de erros que porventura ocorram durante a propagação do sinal através do canal.

O *modulador digital* mapeia os símbolos do sinal digital em um conjunto de sinais adequados à propagação através do canal utilizado. O modulador, portanto, opera

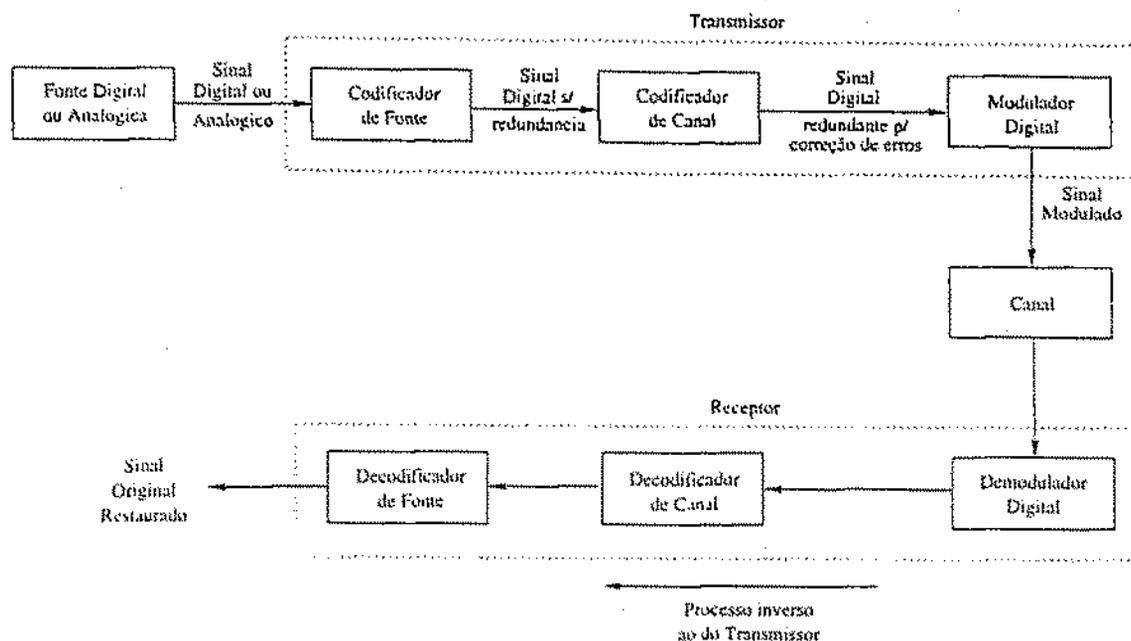


Figura 1.1: Diagrama de blocos funcional de um sistema de comunicação digital genérico.

como uma interface entre o transmissor e o canal de comunicações, que pode ser o espaço livre, um cabo coaxial, um par trançado simples, uma fibra óptica, uma mídia eletromagnética, e assim por diante.

O *canal* em um sistema de comunicação, segundo o modelo adotado, é o responsável pela introdução de ruído, interferência, e pelo desvanecimento do sinal. Todos estes fenômenos inerentes ao canal de comunicação utilizado produzem alterações ou distorções no sinal transmitido, o que se reflete na ocorrência de erros de detecção do sinal na recepção. Tornar o sistema mais robusto, diminuindo os efeitos destes fenômenos e aumentando a confiabilidade e fidelidade do sinal recebido, tem sido o grande desafio para os engenheiros no desenvolvimento desses sistemas. Com isso, observa-se um apelo forte pela obtenção de *bons*¹ esquemas de codificação de canal, também chamada de *codificação para controle de erros* ("error-control coding") [1], [27].

¹O conceito de *bons códigos* será explicado ao longo do texto e melhor entendido no capítulo 2, tão logo sejam definidos alguns conceitos relativos à codificação.

1.2 A codificação para controle de erros

A história da codificação para controle de erros teve início em 1948, quando Claude Shannon demonstrou que a todo canal de comunicação está associada uma capacidade de transmissão de informação C (medida em bits por segundo). Sempre que a taxa de transmissão de informação R_T (medida em bits por segundo) solicitada pelo sistema fosse menor que C , seria possível se obter um código que permitisse a comunicação através do canal com probabilidade de ocorrência de erros tendendo para zero. A demonstração de Shannon não foi construtiva, ou seja, ele demonstrou que este código existia, mas não como obtê-lo. Desde então, estabeleceu-se um incessante esforço na busca por códigos eficientes.

Surgiram duas linhas de abordagem na construção de códigos para controle de erros:

1. A primeira, relativa aos chamados *códigos de bloco*, possui um forte caráter algébrico. A codificação de bloco consiste no simples mapeamento de uma sequência de k símbolos de entrada em uma *palavra código* de comprimento n (veja figura 1.2). Dentre os códigos de bloco clássicos, destacam-se os códigos de Hamming (os primeiros códigos para correção de erros, em 1950), os códigos BCH (Bose-Chaudhuri-Hocquenghem, em 1959-60) e os códigos de Reed-Solomon (1960). Destacam-se também os algoritmos de decodificação PGZ (Peterson-Gorenstein-Zierler), BM (Berlekamp-Massey) e de Forney;
2. A segunda linha, relativa aos chamados *códigos em árvore*, possui um caráter mais probabilístico. Neste caso, as palavras código possuem comprimento variável e o mapeamento é norteado pelas probabilidades de ocorrência dos símbolos do sinal de entrada. Destacam-se aqui os códigos convolucionais e o algoritmo de decodificação de Viterbi (1967).



Figura 1.2: Mapeamento de k símbolos da fonte em n símbolos da palavra código na codificação de bloco.

1.2.1 Os códigos de geometria algébrica

Os códigos de Reed-Solomon, que utilizam como blocos de código os valores de todos os polinômios sobre um corpo finito até um determinado grau, são provavelmente os códigos mais amplamente utilizados atualmente. Contudo, eles apresentam certas limitações: seu comprimento de bloco não pode ser maior que a ordem do corpo finito utilizado.

Uma consequência direta da teoria de Shannon é que *bons* códigos devem ter grandes comprimentos de bloco. Na década de 70, o matemático russo V. D. Goppa sugeriu que, ao invés de avaliar polinômios em pontos simples (valores do corpo finito) como nos códigos de Reed-Solomon, poder-se-ia avaliar funções algébricas em pontos de curvas definidas sobre corpos finitos. Além disso, ele mostrou como substituir a condição sobre os graus dos polinômios por condições sobre as funções algébricas [8].

Em 1982, M. A. Tsfasman, S. G. Vlăduț e T. Zink combinaram a idéia da construção de códigos a partir de curvas algébricas sobre corpos finitos aos recentes resultados da geometria algébrica, produzindo o mais importante desenvolvimento na teoria de códigos para correção de erros dos últimos anos, os chamados *códigos de geometria algébrica* (CGA's), *códigos algébrico-geométricos* ou *códigos de Goppa geométricos* [25].

Os códigos de Reed-Solomon são um caso particular dos CGA's quando a curva é uma reta. As curvas sobre corpos finitos podem ter muito mais pontos que uma simples reta, de forma que os CGA's podem apresentar comprimento de bloco muito maior que os códigos de Reed-Solomon.

A partir do trabalho de Tsfasman, Vlăduț e Zink, passaram a surgir diversos outros trabalhos tratando dos CGA's e requerendo em muitos deles extenso conhecimento da geometria algébrica. De fato, os CGA's compõem um fascinante tópico que une a matemática profunda e abstrata das curvas algébricas aos problemas concretos da engenharia na transmissão de informação, e constituem, como pode-se observar, uma incipiente e promissora área de pesquisa no desenvolvimento de sistemas de comunicação digital mais eficientes.

1.2.2 A decodificação dos CGA's

O primeiro esquema de decodificação para os CGA's, conhecido como *algoritmo básico*, surgiu apenas em 1989, proposto por Justesen et al [10]. Este esquema é aplicável

apenas a CGA's definidos sobre curvas planas e possui baixa capacidade de correção de erros. Ele, assim como o decodificador PGZ, encontra um polinômio localizador de erros que tem as posições dos erros entre seus zeros. Desde então, diversos e variados esquemas de decodificação têm sido propostos.

O objetivo central deste trabalho é inicialmente descrever os CGA's, particularmente os definidos sobre as chamadas curvas de Hermite², e, então, descrever e analisar os principais esquemas e abordagens existentes na decodificação destes códigos. Observe que não constitui objetivo deste texto descrever por completo todo o universo de trabalhos existentes nesta área, mas, sim, discorrer sobre as principais abordagens, constituindo um importante alicerce norteador para o desenvolvimento de futuros trabalhos nesta área.

No estudo dos CGA's e de seus esquemas de decodificação, a geometria algébrica pode ser abordada de duas formas basicamente:

1. A primeira considera as curvas como objetos geométricos e descreve a teoria a partir deste ponto de vista. É a forma adotada por Fulton [7] e por van Lint [13];
2. Na segunda forma, a álgebra relacionada à geometria algébrica é apresentada sem mencionar as curvas propriamente. Esta abordagem, adotada por Stichtenoth [24], proporciona uma teoria mais elegante, contudo implica muito maior abstração.

A restrição deste trabalho aos códigos de Hermite proporciona algumas simplificações teóricas, uma vez que estes podem ser construídos, como será visto, apenas pelo uso de um espaço de funções (polinômios) e pontos de uma curva (a curva de Hermite), o que sugere a adoção da primeira forma de abordagem matemática descrita acima. Muitas publicações tratando do problema da decodificação destes códigos, entretanto, fazem uso de conceitos matemáticos bem mais complexos, relativos à segunda forma, como o de diferenciais e o de resíduos. Para o presente trabalho, em vista da restrição aos códigos de Hermite e das conseqüentes simplificações, será adotada a primeira forma, buscando-se evitar sempre que possível uma maior complexidade matemática, sem, com isso, restringir o universo para análise.

²Os, assim chamados, *códigos de Hermite* apresentam boas características de comprimento e distância mínima relativa, e têm sido largamente explorados pelos que pesquisam nesta área.

1.3 Estrutura do texto

Este capítulo introdutório tem por objetivo principal contextualizar o problema da codificação e decodificação em torno das curvas algébricas, além de descrever a importância de seu estudo.

O capítulo 2 é em grande parte uma revisão dos códigos cíclicos, de Reed-Solomon e de Goppa. Seguindo a abordagem matemática adotada (descrita na seção anterior), os CGA's serão definidos como uma generalização dos códigos de Reed-Solomon e de Goppa [13], [26], [2]. Como foi afirmado, esta abordagem algébrica é mais simples e intuitiva. Este capítulo 2, assim como os demais capítulos, subentende conhecimentos prévios de álgebra e aritmética de corpo finito. O apêndice A apresenta um resumo destes conceitos básicos.

Referente à teoria descrita no capítulo 2, há ainda um apêndice B detalhando a dedução do limite de Gilbert-Varshamov para a taxa de informação assintótica para códigos de bloco.

O capítulo 3 descreve a teoria matemática da geometria algébrica necessária ao entendimento e definição dos CGA's e de seus esquemas de decodificação. Os CGA's são definidos no capítulo 4, além dos aspectos relacionados às curvas de Hermite.

O capítulo 5 trata do problema da decodificação dos CGA's. Ele apresenta as principais abordagens utilizadas e os principais esquemas de decodificação.

Conclusões para este trabalho são apresentadas no capítulo 6.

Capítulo 2

Codificação para Controle de Erros

Os CGA's, objetos deste estudo, podem ser entendidos como uma generalização dos clássicos códigos de Reed-Solomon e de Goppa. Este capítulo tem por objetivo revisar os conceitos relacionados à codificação para controle de erros e introduzir os códigos de Reed-Solomon e de Goppa de um modo que torne natural a crítica extensão destes aos códigos construídos a partir de curvas algébricas, definidos nos capítulos seguintes.

O texto do presente capítulo, apesar de constituir uma revisão, foi escrito para não especialistas no assunto (para detalhes veja Blahut [1], Wicker [27] ou qualquer outro texto básico sobre codificação para controle de erros). Contudo, ele requer conhecimentos prévios de álgebra e aritmética de corpo finito. Esta teoria básica está resumida no apêndice A como forma de referência.

2.1 Conceitos básicos

Em sistemas de comunicação digital, a informação a ser transmitida através de um canal sofre os efeitos de sua influência (introdução de ruído, desvanecimento, etc.), ocasionando erros de interpretação na recepção. O objetivo da codificação para controle de erros é adicionar símbolos extras ao sinal original na transmissão, de modo a permitir na recepção a detecção e correção dos erros que porventura ocorram. Em outras palavras, uma seqüência de símbolos da fonte é substituída na transmissão por uma seqüência mais longa de símbolos com redundância suficiente para proteger a informação original.

2.1.1 Códigos de bloco

Os CGA's fazem parte de uma classe de códigos chamados códigos de bloco. Considere um corpo finito \mathbb{F}_q de q elementos.

Definição 1 (Código de bloco) Um código de bloco C de tamanho M , com símbolos em \mathbb{F}_q , consiste num conjunto de M seqüências $\mathbf{c} = (c_0, c_1, \dots, c_{n-1})$, com $c_i \in \mathbb{F}_q$, de comprimento n , chamadas palavras código.

Se $q = 2$, os símbolos são chamados *bits* e o código é dito *binário*. Usualmente, para algum inteiro k , $M = q^k$ palavras código, sendo o código referido como um código (n, k) . Um exemplo elementar é o código binário de repetição $(3, 1)$, no qual

informação		código
0	\leftrightarrow	000
1	\leftrightarrow	111

Na decodificação deste código, a decisão na recepção (após introdução de erros pelo canal) sobre qual informação foi transmitida é feita por maioria: se dois ou três bits forem 0, decide-se por 0; se dois ou três bits forem 1, decide-se por 1. Observa-se que este código possibilita a correção de apenas um erro por palavra código. Como será visto, este não é um *bom* código.

A taxa de informação R de um código de bloco¹ é definida por

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

A princípio, será sempre melhor utilizar um código de comprimento n grande do que um código com palavras curtas. Isto, porque os erros são de natureza aleatória e ocorrem durante intervalos de tempo variáveis. Em alguns segmentos ocorrem mais erros que a média, em alguns menos. Portanto, para a mesma taxa R , de forma a possibilitar uma transmissão confiável de informação, é preferível utilizar um código de comprimento n grande, que seja capaz de corrigir uma quantidade maior de erros de uma única vez, mesmo que isto implique codificação e decodificação mais complexas. *Bons códigos* são, portanto, aqueles de comprimento n tão grande quanto possível,

¹Esta taxa R é adimensional e não deve ser confundida com a taxa de transmissão de informação R_T , medida em bits por segundo.

sem que isto comprometa a taxa do código ($R_{n \rightarrow \infty} = \frac{k}{n} = R_0 \neq 0$) ou sua capacidade de correção. O código de repetição $(n, 1)$, com $n \rightarrow \infty$, por exemplo, terá $R = 0$ (corrige tudo, mas não transmite nada).

2.1.2 Geometria dos códigos

Define-se como *distância de Hamming* $d(x, y)$ entre duas palavras código x e y de comprimento n , com símbolos em \mathbb{F}_q , o número de posições nas quais elas diferem. Por exemplo, se $x = (\alpha^2, 0, 1, 0, \alpha)$ e $y = (\alpha, 1, 1, 0, \alpha^2)$ são palavras de comprimento $n = 5$ em \mathbb{F}_4 , então $d(x, y) = 3$ (elas diferem nas posições 1, 2 e 5).

Considere $C = \{c_i, i = 0, \dots, M - 1\}$ um código de bloco. A *distância mínima* d de C é a menor distância de Hamming entre duas palavras código de C . Ou seja,

$$d = \min_{\substack{c_i, c_j \in C \\ i \neq j}} d(c_i, c_j).$$

Um código (n, k) de distância mínima d também é referido como um código (n, k, d) . Define-se *distância mínima relativa* δ de um código (n, k, d) como sendo

$$\delta = \frac{d}{n}.$$

Este conceito também é importante na avaliação de bons códigos. Por exemplo, um código (n, n) , com $n \rightarrow \infty$, tem taxa $R = 1$, porém tem distância mínima relativa $\delta = 0$ (transmite tudo, mas não corrige nada).

Definindo uma *esfera* $S(x, r)$ de raio r com centro em x como sendo

$$S(x, r) = \{c \in \mathbb{F}_q^n \mid d(x, c) \leq r\},$$

observa-se que é possível associar a todas as palavras de um código C de distância mínima d , esferas de raio $r = \frac{d-1}{2}$ que não se interceptam. A cardinalidade $V(n, r)$ desta esfera é dada por

$$V(n, r) = \sum_{i=0}^r \binom{n}{i} (q-1)^i. \quad (2.1)$$

Portanto,

$$|C| \cdot V(n, r) \leq q^n, \quad (2.2)$$

em que $|C|$ é o número de palavras M do código C . Este resultado (equação 2.2) é conhecido como *limite de Hamming* para o código C .

2.1.3 Códigos lineares

A maioria dos bons códigos conhecidos pertence a uma classe de códigos chamados de códigos lineares. A estrutura imposta por esta classe proporciona meios para a busca de bons códigos e construção de codificadores e decodificadores práticos.

Considere o espaço vetorial \mathbb{F}_q^n . Um *código linear* é qualquer subespaço vetorial de \mathbb{F}_q^n . Em outras palavras, um código linear é um conjunto não vazio de n -úplas (vetores) em \mathbb{F}_q , chamadas palavras código, tal que a soma de duas palavras código é uma palavra código, e o produto de um escalar (elemento de \mathbb{F}_q) por uma palavra código também é uma palavra código.

O *peso de Hamming* $w(c)$ de uma palavra código c é o número de posições não nulas desta palavra. O *peso mínimo* w_{\min} de um código é o menor entre os pesos de Hamming de todas as palavras código não nulas. Por conseguinte, em um código linear,

$$d = w_{\min}.$$

Considere C um código linear em \mathbb{F}_{q^m} , em que $m > 1$. O *subcódigo de subcorpo* em \mathbb{F}_q consiste em todas as palavras de C que possuem todas as coordenadas no subcorpo \mathbb{F}_q .

2.1.4 Matrizes dos códigos

Qualquer conjunto de k vetores linearmente independentes de uma base para o código C (base para o subespaço vetorial C) pode ser usado como linhas de uma matriz $G_{k \times n}$, chamada *matriz geradora* de C . Por exemplo, para algum código binário $(7, 4)$, pode-se ter

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

Qualquer palavra código de C é uma combinação linear das linhas de G . O número de linhas k é a *dimensão* do código C .

As q^k palavras de informação (k -úplas) i são mapeadas nas q^k palavras código c da seguinte forma:

$$c = iG.$$

Por exemplo, para o código binário (7,4) cuja matriz G é dada acima, a palavra $i = [1 \ 1 \ 0 \ 1]$ é mapeada na palavra código $c = [1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 1]$.

Como C é um subespaço vetorial, ele possui um subespaço ortogonal C^\perp , que é o conjunto de todos os vetores ortogonais a C . Este código C^\perp ortogonal é dito o *código dual* de C . Este código tem dimensão $n - k$ e uma matriz geradora $H_{n-k \times n}$. Por exemplo, para o código binário (7,4) do exemplo acima, tem-se que

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

Uma n -úpla c é uma palavra código se for ortogonal às linhas de H , ou seja, se

$$cH^T = 0.$$

Por este motivo, H é dita a *matriz de paridade* (verificação de paridade) de C . Desta forma, pode-se definir o código C como sendo

$$C = \{c \in \mathbb{F}_q^n \mid cH^T = 0 \in \mathbb{F}_q^{n-k}\}.$$

Observe, então, que uma palavra código $c \in C$ de peso $w(c)$ implica uma relação de dependência entre as $w(c)$ colunas da matriz H correspondentes às coordenadas não nulas de c . Portanto, observa-se que o código C possui distância mínima d se e só se nenhum grupo de $d - 1$ colunas de H for linearmente dependente, caso em que haveria uma palavra c de peso $d - 1$, tal que $cH^T = 0$. Como as colunas de H são palavras de comprimento $n - k$, então o número máximo de colunas linearmente independentes de H é $n - k$. Portanto,

$$\begin{aligned} d - 1 &\leq n - k \Rightarrow \\ d &\leq n - k + 1. \end{aligned} \tag{2.3}$$

Esta desigualdade é conhecida como o *limite de Singleton* para códigos lineares (n, k, d) . Os códigos que apresentam igualdade nesta relação são ditos *códigos de máxima distância mínima* ("maximum-distance separable" - MDS).

Se G é a matriz geradora de um código MDS, então quaisquer k colunas de G são linearmente independentes. Como G é a matriz de paridade para o código dual, então o código dual possui distância mínima maior que k . Como o código dual possui dimensão $n - k$, sua distância mínima não pode exceder $k + 1$. Conclui-se que o dual de um código MDS também é um código MDS.

2.1.5 Limites dos códigos

As seqüências de código desejadas (os bons códigos), como já fora explicado, são aquelas com comprimento n tão grande quanto possível, mantendo finitas e não nulas a taxa de informação R e a distância mínima relativa δ (capacidade de correção de erros).

Considere a notação $A(n, d)$ para o valor máximo de $M = |C|$ para o qual um código (n, k, d) (linear ou não) existe. A avaliação de bons códigos é feita através da função taxa de informação assintótica $R(\delta)$ definida por

$$R(\delta) = \lim_{n \rightarrow \infty} \frac{\log_q A(n, \delta n)}{n}, \quad (2.4)$$

que relaciona a taxa de informação R e a distância mínima relativa δ de um código quando n tende para infinito.

Durante muitos anos, o melhor limite inferior para $R(\delta)$ foi o limite de Gilbert-Varshamov, descrito em seguida.

Teorema 2 (Limite de Gilbert-Varshamov) Considere $0 \leq \delta \leq \frac{q-1}{q}$, então

$$R(\delta) \geq 1 - H_q(\delta), \quad (2.5)$$

sendo H_q a função entropia definida por

$$H_q(x) = \begin{cases} 0 & , x = 0; \\ x \log_q(q-1) - x \log_q x - (1-x) \log_q(1-x) & , 0 < x < \frac{q-1}{q}. \end{cases}$$

Prova. A prova deste teorema está descrita com detalhes no apêndice B. ■

A figura 2.1 ilustra o limite de Gilbert-Varshamov para o caso em que $q = 16$. Os bons códigos situam-se próximos do centro da curva, enquanto os demais tendem para os extremos da curva quando n se torna grande.

No capítulo 4, após descritos os códigos de geometria algébrica, será apresentado um novo limite inferior para $R(\delta)$, que é melhor que o de Gilbert-Varshamov para todo $q \geq 49$ em parte do intervalo $\left[0, \frac{q-1}{q}\right]$.

2.2 Códigos de Reed-Solomon

Convencionalmente, os códigos de Reed-Solomon são definidos a partir de um polinômio gerador ou, equivalentemente, a partir de uma matriz geradora (ou de uma matriz de

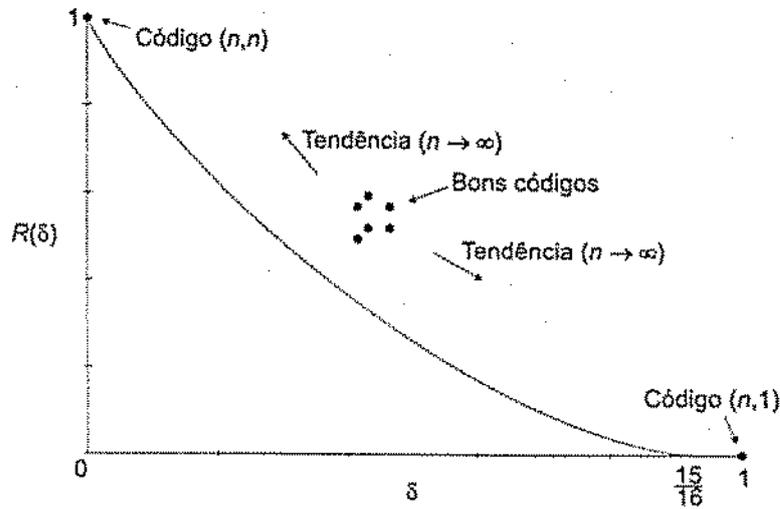


Figura 2.1: Limite de Gilbert-Varshamov para o caso em que $q = 16$.

paridade) [1], [27]. Além desta, uma construção distinta destes códigos será apresentada em seguida no intuito de facilitar a definição dos CGA's como extensão destes [13], [2], [26].

2.2.1 Definição convencional

Considere um código linear C de comprimento n em que, para toda palavra código $c = (c_0, c_1, \dots, c_{n-1}) \in C$, existe outra palavra código $c' = (c_{n-1}, c_0, \dots, c_{n-2}) \in C$ resultado da rotação dos elementos da primeira palavra. Esta nova estrutura, dos chamados *códigos cíclicos*, tem permitido a obtenção de esquemas de codificação e decodificação algorítmica e computacionalmente eficientes, como é o caso dos códigos de Reed-Solomon.

Considere $\mathbb{F}_q[x]/(x^n - 1)$ o anel dos polinômios em \mathbb{F}_q com grau menor que n . Toda palavra código $(c_0, c_1, \dots, c_{n-1}) \in \mathbb{F}_q^n$ pode ser vista como um polinômio $c_0 + c_1x + \dots + c_{n-1}x^{n-1} \in \mathbb{F}_q[x]/(x^n - 1)$. Neste anel, a multiplicação de um polinômio por x equivale ao deslocamento cíclico da palavra equivalente. Observa-se, então, que existe uma correspondência entre um código cíclico e um ideal² contido em $\mathbb{F}_q[x]/(x^n - 1)$.

²Um *ideal* I é um subconjunto de um anel A que satisfaz os seguintes axiomas:

1. I é um grupo abeliano sob a operação de adição;
2. Para todo $a \in I$ e $b \in A$, tem-se que $ab \in I$.

Mostra-se [3, pp. 37–44] que este ideal (o código cíclico) pode ser gerado por um único polinômio $g(x)$, divisor de $x^n - 1$, conhecido como *polinômio gerador*. O polinômio $g(x)$ é, por definição, o máximo divisor comum – MDC dos polinômios do ideal (ele gera este ideal). Sendo $g(x)$ um polinômio de grau $n - k$, um código cíclico de comprimento n é o resultado da multiplicação de $g(x)$ pelos polinômios $i(x)$ de grau menor que k (informação). Ou seja,

$$c(x) = i(x)g(x).$$

Os códigos de Reed-Solomon são códigos cíclicos de comprimento $n = q - 1$, em \mathbb{F}_q , cujo polinômio gerador é da forma

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

em que α é um elemento primitivo de \mathbb{F}_q e d' é a distância mínima projetada do código.

Observe que $g(x)$ é sempre um polinômio de grau $d' - 1 = n - k \Rightarrow d' = n - k + 1$. Deste resultado e do limite de Singleton (equação 2.3), tem-se que a distância mínima d do código é

$$d = d' = n - k + 1.$$

Este é, então, um código MDS. Sua dimensão é $k = n - d + 1$. Este código é capaz de corrigir $d = 2t + 1 \Rightarrow t = n - k$ erros.

Pode-se também definir os *códigos de Reed-Solomon estendidos* pela adição de uma componente, fazendo $n = q$. Estes códigos, também MDS, diferentemente dos anteriores, não são cíclicos.

2.2.2 Definição geométrica

Observe agora uma formulação diferente para os códigos de Reed-Solomon apresentados acima. Primeiramente, considere o conjunto $\{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\}$ de n elementos distintos de \mathbb{F}_q . Denote por $L \subset \mathbb{F}_q[x]$ o conjunto de polinômios de grau menor que $k \leq n$. Defina um código C como sendo

$$C = \{(f(\alpha_0), f(\alpha_1), \dots, f(\alpha_{n-1})) \mid f \in L\}. \quad (2.6)$$

Uma maior discussão sobre este assunto será feita no capítulo seguinte.

Este código possui comprimento n e dimensão k .

Como um polinômio de grau menor que k possui no máximo $k - 1$ zeros, cada palavra código de C (equação 2.6) possui peso no mínimo $n - (k - 1) = n - k + 1$. Daí e do limite de Singleton (equação 2.3), conclui-se que este é um código MDS, ou seja, possui distância mínima $d = n - k + 1$. O mesmo código de Reed-Solomon cíclico da definição convencional vista na subseção anterior é obtido nesta construção para $n = q - 1$, e o código de Reed-Solomon estendido para $n = q$.

Existe ainda uma forma mais geral do código da equação 2.6. Considere os vetores $\mathbf{a} = (\alpha_0, \alpha_1, \dots, \alpha_{n-1})$ de n elementos distintos de \mathbb{F}_{q^m} e $\mathbf{v} = (v_0, v_1, \dots, v_{n-1})$ de elementos não nulos e não necessariamente distintos de \mathbb{F}_{q^m} . Considere L o conjunto de polinômios de grau menor que k em $\mathbb{F}_{q^m}[x]$. O código

$$GRS_k(\mathbf{a}, \mathbf{v}) = \{(v_0 f(\alpha_0), v_1 f(\alpha_1), \dots, v_{n-1} f(\alpha_{n-1})) \mid f \in L\} \quad (2.7)$$

possui os mesmos parâmetros do código anterior e é conhecido como *código de Reed-Solomon generalizado*.

Considere agora o conjunto dos pares de elementos (α_1, α_2) , com $\alpha_i \in \mathbb{F}_q$. Considere os pares que são múltiplos escalares entre si como sendo de uma mesma classe de equivalência, ou seja, $(\alpha_1, \alpha_2) \equiv (\beta\alpha_1, \beta\alpha_2)$, para todo $\beta \in \mathbb{F}_q^*$. Cada classe de equivalência é representada por um dos pares $(1, \alpha)$, com $\alpha \in \mathbb{F}_q$, ou $Q = (0, 1)$ (Q é chamado ponto no infinito). Este conjunto de classes de equivalência é conhecido como *linha projetiva* \mathbb{P}^1 (uma maior explanação sobre o assunto é feita no capítulo seguinte). Por exemplo, considerando o corpo \mathbb{F}_4 , tem-se que

$$\mathbb{P}^1 = \{(1, 0), (1, 1), (1, \alpha), (1, \alpha^2), Q\}.$$

Uma *função racional* em \mathbb{P}^1 é um quociente da forma $\frac{a(x,y)}{b(x,y)}$, em que $a(x, y)$ e $b(x, y)$ são polinômios homogêneos de mesmo grau em \mathbb{F}_q (sem esta restrição o quociente não seria definido em \mathbb{P}^1). Um ponto de \mathbb{P}^1 é um pólo de uma função racional $\frac{a(x,y)}{b(x,y)}$ se o polinômio $b(x, y)$ for zero neste ponto (e $a(x, y)$ não for zero). Defina, então, L como sendo o espaço vetorial de todas as funções racionais em \mathbb{P}^1 que não possuem pólos em \mathbb{P}^1 , exceto possivelmente pólos de ordem menor que k em Q . É fácil observar que funções racionais da forma $\frac{a(x,y)}{x^l}$, com $l < k$, em que $a(x, y)$ é um polinômio homogêneo de grau l , possuem as propriedades descritas. Segue, então, a definição dos códigos de Reed-Solomon.

Definição 3 (Códigos de Reed-Solomon) *O código de Reed-Solomon pode ser descrito pelo conjunto de n -úplas*

$$C = \{ (f(P_1), f(P_2), \dots, f(P_n)) \mid f \in L \}, \quad (2.8)$$

em que $P_1, P_2, \dots, P_n \in \mathbb{P}^1$ são pontos diferentes de Q .

Em outras palavras, o que se fez foi tomar uma linha projetiva, um conjunto de n pontos desta linha e um espaço vetorial de funções definido a partir de um ponto da linha projetiva diferente dos n pontos supracitados. O código C fornecido pela equação 2.8 consiste apenas num conjunto de n -úplas de valores destas funções nestes pontos. Este processo de avaliar funções racionais em pontos de uma curva (no caso uma linha) constitui uma das idéias centrais na construção de CGA's, que será chamada de *construção por funções*. Uma segunda idéia central na construção de CGA's é derivada da definição dos códigos de Goppa na seção seguinte.

2.3 Códigos de Goppa

Antes de definir os códigos de Goppa, de modo a introduzi-los, será apresentada uma conhecida classe de códigos cíclicos, os chamados códigos BCH (Bose-Chaudhuri-Hocquenghem) [1], [27].

2.3.1 Códigos BCH

Considere α um elemento de ordem n do corpo \mathbb{F}_{q^m} , em que n divide $q^m - 1$. Considere $g(x) \in \mathbb{F}_q[x]$ o polinômio de menor grau cujos zeros sejam

$$\{ \alpha^i \mid i = 1, 2, \dots, 2t \},$$

para algum inteiro $t \geq 1$. Faça o grau de $g(x)$, referido como o polinômio gerador do código, igual a $n - k$. Mostra-se que $n - k \leq 2tm$ [2]. Então,

$$C = \{ a(x)g(x) \mid \deg[a(x)] < k, a(x) \in \mathbb{F}_q[x] \} \quad (2.9)$$

é um código BCH de comprimento n , dimensão $k \geq n - 2tm$ e distância mínima $d \geq 2t + 1$.

Observe que, tomando-se o polinômio

$$h(x) = \prod_{i=1}^{2t} (x - \alpha^i) \in \mathbb{F}_{q^m}[x]$$

no lugar de $g(x)$ e substituindo-se o corpo \mathbb{F}_q por \mathbb{F}_{q^m} , o código da equação 2.9 torna-se um código de Reed-Solomon C' de comprimento n , dimensão k e distância mínima $d = 2t + 1$. Portanto, o código BCH C da equação 2.9 consiste num subcódigo de subcorpo de C' , ou seja,

$$C = C' \cap \mathbb{F}_q^n.$$

2.3.2 Códigos de Goppa

Utilizando a mesma notação da definição dos códigos BCH acima, em que as palavras código são da forma $c(x) = c_0 + c_1x + \dots + c_{n-1}x^{n-1}$, tal que $c(\alpha^i) = 0$, para $i = 1, \dots, d-1$, considere o seguinte cálculo:

$$\begin{aligned} (x^n - 1) \sum_{i=0}^{n-1} \frac{c_i}{x - \alpha^{-i}} &= \sum_{i=0}^{n-1} c_i \frac{x^n - 1}{x - \alpha^{-i}} \\ &= \sum_{i=0}^{n-1} c_i (x^{n-1} + \alpha^{-i}x^{n-2} + \dots + \alpha^{-(n-2)i}x + \alpha^{-(n-1)i}) \\ &= \sum_{i=0}^{n-1} c_i \sum_{j=0}^{n-1} x^j (\alpha^{-i})^{n-1-j} \\ &= \sum_{j=0}^{n-1} x^j \sum_{i=0}^{n-1} c_i (\alpha^{j+1})^i. \end{aligned}$$

Para valores de j no intervalo $[0, d-2]$, o somatório $\sum_{i=0}^{n-1} c_i (\alpha^{j+1})^i = c(\alpha^{j+1})$ é por definição nulo. Portanto, para algum polinômio $f(x)$, tem-se que

$$\begin{aligned} (x^n - 1) \sum_{i=0}^{n-1} \frac{c_i}{x - \alpha^{-i}} &= \sum_{j=d-1}^{n-1} x^j \sum_{i=0}^{n-1} c_i (\alpha^{j+1})^i \\ &= x^{d-1} f(x), \end{aligned}$$

ou seja, o somatório $\sum_{i=0}^{n-1} \frac{c_i}{x - \alpha^{-i}}$ é necessariamente divisível por x^{d-1} (x^{d-1} não divide $x^n - 1$). Portanto,

$$\sum_{i=0}^{n-1} \frac{c_i}{x - \alpha^{-i}} \equiv 0 \pmod{x^{2t}}. \quad (2.10)$$

Por conseguinte, um vetor $(c_0, c_1, \dots, c_{n-1})$, com $c_i \in \mathbb{F}_q$, é uma palavra código se satisfizer a equação 2.10. Dependendo do corpo utilizado, esta construção acima fornece um código de Reed-Solomon ou um código BCH. A passagem destes para os códigos de Goppa agora envolve apenas a substituição da seqüência $\{\alpha^i \mid i = 1, 2, \dots, 2t\}$ usada (α é um elemento primitivo de ordem n do corpo \mathbb{F}_{q^m}) por um conjunto arbitrário de elementos distintos, e do monômio x^{2t} por um polinômio geral $g(x)$.

Definição 4 (Códigos de Goppa) *Considere $L = \{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\}$ um conjunto arbitrário de n elementos distintos de \mathbb{F}_{q^m} e $g(x) \in \mathbb{F}_{q^m}[x]$ um polinômio mônico, tal que $g(\alpha_i) \neq 0$, para $i = 0, 1, \dots, n-1$. O código de Goppa $\Gamma(L, g)$ é o conjunto de palavras $(c_0, c_1, \dots, c_{n-1}) \in \mathbb{F}_q^n$, tal que*

$$\sum_{i=0}^{n-1} \frac{c_i}{x - \alpha_i} \equiv 0 \pmod{g(x)}. \quad (2.11)$$

O código $\Gamma(L, g)$ definido pela equação 2.11 é um subcódigo de subcorpo do dual do código de Reed-Solomon generalizado (equação 2.7). Para verificar isto, considere $g(x) = \sum_{i=0}^t g_i x^i$. Então,

$$\phi(x) = \frac{g(x) - g(a)}{x - a} = \sum_{k+j \leq t-1} g_{k+j+1} a^j x^k$$

é um polinômio de grau menor que t , para qualquer a . Como

$$(x - a)\phi(x) \equiv -g(a) \pmod{g(x)},$$

pode-se re-escrever a equação 2.11 como sendo

$$\begin{aligned} \sum_{i=0}^{n-1} \frac{c_i}{(x - \alpha_i)\phi(x)} \phi(x) &\equiv 0 \pmod{g(x)} \Rightarrow \\ \sum_{i=0}^{n-1} \frac{c_i}{-g(\alpha_i)} \sum_{k+j \leq t-1} g_{k+j+1} (\alpha_i)^j x^k &= 0 \Rightarrow \\ \sum_{i=0}^{n-1} c_i h_i \sum_{k+j \leq t-1} g_{k+j+1} (\alpha_i)^j x^k &= 0, \end{aligned} \quad (2.12)$$

em que $h_i = \frac{1}{g(\alpha_i)}$. Observe que o coeficiente de x^k na equação 2.12 é zero para $0 \leq k \leq t-1$. Isto significa que, se $c = (c_0, c_1, \dots, c_{n-1})$ é uma palavra código, então

o produto interno de c com as linhas da matriz

$$\begin{bmatrix} h_0 g_t & \cdots & h_{n-1} g_t \\ h_0 (g_{t-1} + g_t \alpha_0) & \cdots & h_{n-1} (g_{t-1} + g_t \alpha_{n-1}) \\ \vdots & \cdots & \vdots \\ h_0 \sum_{i=1}^t g_i \alpha_0^{i-1} & \cdots & h_{n-1} (\sum_{i=1}^t g_i \alpha_{n-1}^{i-1}) \end{bmatrix}$$

deve ser zero. Usando operações elementares nas linhas desta matriz, obtém-se a seguinte matriz de paridade para o código $\Gamma(L, g)$:

$$H = \begin{bmatrix} h_0 & h_1 & \cdots & h_{n-1} \\ h_0 \alpha_0 & h_1 \alpha_1 & \cdots & h_{n-1} \alpha_{n-1} \\ \vdots & \vdots & \cdots & \vdots \\ h_0 \alpha_0^{t-1} & h_1 \alpha_1^{t-1} & \cdots & h_{n-1} \alpha_{n-1}^{t-1} \end{bmatrix}$$

Compare esta matriz com os códigos da equação 2.7, em que $\mathbf{v} = (h_0, h_1, \dots, h_{n-1})$. Observa-se que H é a matriz geradora do código $GRS_k(\mathbf{a}, \mathbf{v})$ da equação 2.7. Portanto, o código de Goppa $\Gamma(L, g)$ é um subcódigo de subcorpo do dual de um código de Reed-Solomon generalizado.

Como o posto da matriz H sobre \mathbb{F}_{q^m} é exatamente t , seu posto sobre \mathbb{F}_q é no máximo mt . Portanto, a dimensão do código de Goppa $\Gamma(L, g)$ é $k \geq n - mt$ e sua distância mínima é $d \geq t + 1$, em que t é o grau de $g(x)$.

2.4 Transição para os CGA's

Agora será dada uma nova formulação aos códigos de Goppa, de forma a colocar em perspectiva sua transição para os CGA's.

Considere a função

$$f(x) = \sum_{i=0}^{n-1} \frac{c_i}{x - \alpha_i} = \frac{\omega(x)}{\lambda(x)}$$

correspondente à palavra código $(c_0, c_1, \dots, c_{n-1})$, em que

$$\lambda(x) = \prod_i (x - \alpha_i) \in \mathbb{F}_{q^m}[x],$$

$\deg \omega(x) < \deg \lambda(x) = n$ e $\{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\}$ é um conjunto arbitrário de n elementos distintos de \mathbb{F}_{q^m} . Portanto,

$$c_i = f(x)(x - \alpha_i)|_{x=\alpha_i}$$

é obtido pelo cancelamento do pólo α_i em $f(x)$ e avaliação no próprio ponto α_i . Em outras palavras, c_i é o resíduo de $f(x)$ em α_i , denotado por $\text{Res}_{\alpha_i}(f)$.

Considere

$$\mathcal{X}_j(x) = \prod_{i=1, i \neq j}^n (x - \alpha_i) = \frac{\lambda(x)}{x - \alpha_j}$$

e

$$f(x) = \frac{\omega(x)}{\lambda(x)} = \frac{g(x)q(x)}{\lambda(x)},$$

uma vez que, por definição, $g(x)|f(x)$ ($g(x)$ é o mesmo da equação 2.11). Pode-se, então, expressar o resíduo de $f(x)$ em α_i por

$$\text{Res}_{\alpha_i}(f) = \frac{\omega(x)(x - \alpha_i)}{\lambda(x)} \Big|_{x=\alpha_i} = \frac{g(\alpha_i)}{\mathcal{X}'(\alpha_i)} q(\alpha_i),$$

que é zero apenas se $q(\alpha_i) = 0$.

Agora, generalizando este conceito, defina um espaço vetorial L de funções racionais, tal que

1. $f(x) \in L$ possui pelo menos os mesmos zeros que $g(x)$, com pelo menos a mesma multiplicidade;
2. $f(x) \in L$ não possui pólos, exceto possivelmente em $\{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\}$, caso em que os pólos são de ordem 1.

Defina, então, um código C' de comprimento n em \mathbb{F}_{q^m} como sendo

$$C' = \{(\text{Res}_{\alpha_0} f, \text{Res}_{\alpha_1} f, \dots, \text{Res}_{\alpha_{n-1}} f) \mid f \in L\}. \quad (2.13)$$

Basicamente, este código C' fornecido pela equação 2.13 consiste no conjunto das n -úplas dos resíduos das funções do espaço de funções L definido pelos pontos de $\{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\}$ nestes pontos. Este processo de determinar os resíduos de funções

racionais em um conjunto de pontos constitui a segunda idéia central na construção de CGA's, que será chamada de *construção por resíduos*.

O código de Goppa definido pela equação 2.11 é um subcódigo de subcorpo deste código C' em \mathbb{F}_q . Como o código de Goppa é também um subcódigo de subcorpo do dual do código C definido pela equação 2.8, observa-se, então, que existe uma relação de dualidade entre as construções por funções e por resíduos usadas nas definições dos códigos C (equação 2.8) e C' (equação 2.13), respectivamente.

Do que foi apresentado neste capítulo, dois enfoques usados na definição de códigos são de grande importância na construção dos CGA's nos capítulos seguintes:

- A avaliação de funções racionais em um conjunto fixo de pontos distintos (equação 2.8);
- O resíduo de funções racionais em um conjunto fixo de pontos distintos (equação 2.13).

A definição dos CGA's utilizará estes dois enfoques, que fornecem códigos duais, diferindo apenas na obtenção do conjunto de pontos distintos: serão pontos de uma curva algébrica em um corpo finito. Além disso, o espaço de funções racionais será definido a partir de pontos desta mesma curva algébrica.

Apesar destas abordagens serem simples, a definição dos parâmetros do código dependerá fortemente da teoria das curvas algébricas. O próximo capítulo dedicar-se-á à apresentação desta teoria matemática, requisito necessário ao entendimento dos CGA's, definidos no capítulo seguinte.

Capítulo 3

Tópicos de Geometria Algébrica

O capítulo anterior sugere a forma como os CGA's são construídos. No entanto, não é tão direta quanto possa parecer a transição dos códigos de Reed-Solomon e de Goppa apresentados para os CGA's. Diversos conceitos relacionados à teoria da geometria algébrica são necessários para possibilitar esta transição.

Este capítulo tem por objetivo introduzir as noções da geometria algébrica, tais como variedades afim e projetiva, bases de Gröbner, corpo de funções, anel de coordenadas, anel local, divisor, etc., necessárias à discussão dos CGA's e de seus esquemas de decodificação. Não é pretensão do presente texto constituir uma referência completa ou pormenorizada sobre este denso assunto, mas apresentá-lo através de uma abordagem que permita ao leitor ter mais fácil acesso à definição dos códigos no capítulo que segue. Grande parte das provas não serão apresentadas aqui. Abordagens mais detalhadas sobre esta teoria e provas dos resultados apresentados podem ser obtidos em [7], [3], [15], [24] ou outras referências que tratem do assunto.

3.1 Ideais e variedades

Os conceitos de ideal e variedade e suas relações são a base para a teoria denominada geometria algébrica. A natureza geométrica provém das variedades, que são as curvas, superfícies e objetos de maior dimensão definidos por equações polinomiais. Os ideais, subestruturas dos anéis polinomiais $\mathbb{F}_q[x_1, \dots, x_n]$, constituem a "álgebra" relacionada às variedades. Esta seção é dedicada ao entendimento destes conceitos.

3.1.1 Variedade afim

Antes de tudo, é necessário definir polinômios em n variáveis x_1, \dots, x_n . Defina um *monômio* em x_1, \dots, x_n como sendo

$$x^\alpha = x_1^{\alpha_1} \dots x_n^{\alpha_n},$$

em que $\alpha_1, \dots, \alpha_n$ são inteiros não negativos. O grau de x^α é dado por $\alpha_1 + \dots + \alpha_n$.

Considere \mathbb{F}_p um corpo finito com p elementos e \mathbb{F}_q seu fecho algébrico (esta notação será utilizada em todo o texto). Um *polinômio* f em x_1, \dots, x_n com coeficientes em \mathbb{F}_q consiste numa combinação linear de monômios da forma

$$f = \sum_{\alpha} a_{\alpha} x^{\alpha}, \quad a_{\alpha} \in \mathbb{F}_q,$$

em que a soma é feita sobre um número finito de n -úplas $\alpha = (\alpha_1, \dots, \alpha_n)$. O anel dos polinômios em x_1, \dots, x_n com coeficientes em \mathbb{F}_q é denotado por $\mathbb{F}_q[x_1, \dots, x_n]$.

Defina agora *espaço afim* de dimensão n , sobre \mathbb{F}_q , como sendo o conjunto

$$\mathbb{A}^n = \{(a_1, \dots, a_n) \mid a_1, \dots, a_n \in \mathbb{F}_q\}.$$

Um elemento $P \in \mathbb{A}^n$ é dito ser um *ponto* do espaço \mathbb{A}^n . Se \mathbb{G} é algum subcorpo de \mathbb{F}_q que contém \mathbb{F}_p e P é um ponto de \mathbb{A}^n com coordenadas em \mathbb{G} , então P é dito um *ponto racional- \mathbb{G}* e o conjunto de pontos racionais- \mathbb{G} de \mathbb{A}^n é denotado por $\mathbb{A}^n(\mathbb{G})$.

Observe que um polinômio $f \in \mathbb{F}_q[x_1, \dots, x_n]$ pode ser considerado como um mapeamento $f: \mathbb{A}^n \rightarrow \mathbb{F}_q$ dado por

$$f(P) = f(a_1, \dots, a_n),$$

em que $P = (a_1, \dots, a_n) \in \mathbb{A}^n$. Se $f(P) = 0$, diz-se que P é um *zero* de f .

Definição 5 (Variedade afim) Considere f_1, \dots, f_s polinômios em $\mathbb{F}_q[x_1, \dots, x_n]$. A variedade afim $V(f_1, \dots, f_s)$ definida por f_1, \dots, f_s é, então, dada por

$$V(f_1, \dots, f_s) = \{P \in \mathbb{A}^n \mid f_i(P) = 0, 1 \leq i \leq s\}.$$

Em outras palavras, uma variedade afim $V(f_1, \dots, f_s) \subset \mathbb{A}^n$ é o conjunto de soluções para o sistema de equações $f_1(x_1, \dots, x_n) = \dots = f_s(x_1, \dots, x_n) = 0$.

3.1.2 Ideal

Segue a definição do objeto algébrico denominado ideal.

Definição 6 (Ideal) Um subconjunto $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ constitui um ideal se satisfizer os seguintes axiomas:

1. I é um grupo abeliano sob a adição;
2. Se $f \in I$ e $h \in \mathbb{F}_q[x_1, \dots, x_n]$, então $hf \in I$.

Considere f_1, \dots, f_s polinômios em $\mathbb{F}_q[x_1, \dots, x_n]$. O conjunto denotado por $\langle f_1, \dots, f_s \rangle$ e dado por

$$\langle f_1, \dots, f_s \rangle = \left\{ \sum_{i=1}^s h_i f_i \mid h_1, \dots, h_s \in \mathbb{F}_q[x_1, \dots, x_n] \right\} \quad (3.1)$$

é um ideal gerado por f_1, \dots, f_s (isto é facilmente verificado aplicando-se os axiomas da definição 6). Qualquer conjunto de funções que gerem um ideal é considerada uma base para este. No caso de espaços vetoriais, define-se como base para um espaço todo conjunto de vetores linearmente independentes que o gerem. Como conseqüência, toda base para um espaço vetorial possui um número de elementos igual à dimensão do espaço. Já no caso de ideais, não há um número determinado de elementos para uma base.

Considere $V \subset \mathbb{A}^n$ uma variedade afim. O conjunto denotado por $I(V)$ e dado por

$$I(V) = \{f \in \mathbb{F}_q[x_1, \dots, x_n] \mid f(P) = 0, P \in V\} \quad (3.2)$$

é um ideal gerado por V (isto é verificado também aplicando-se os axiomas da definição acima).

Todo ideal pode ser gerado por um conjunto finito de funções linearmente independentes denominado *base*. Há bases especiais, chamadas *bases de Gröbner*, que apresentam propriedades que possibilitam a solução de problemas envolvendo ideais por meios computacionais, e que serão descritas na subseção seguinte. Um ideal que pode ser gerado por uma única função é dito *principal*. Um ideal $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ é dito *primo* se $I \neq \mathbb{F}_q[x_1, \dots, x_n]$ e se, para $ab \in I$, $a \in I$ ou $b \in I$. Um ideal $I \subset A$ é dito *máximo* em um conjunto A se não existir outro ideal em A que contenha I .

Um ideal I é dito *radical* se o fato de que $f^m \in I$, para $m \geq 1$ inteiro, implicar que $f \in I$. Para uma variedade V qualquer, se $f^m \in I(V)$, então $f \in I(V)$, uma vez que $[f(x)]^m = 0 \Rightarrow f(x) = 0$. Portanto, $I(V)$ é sempre um ideal radical. Por conseguinte, um ideal $I = \langle f_1, \dots, f_s \rangle$, em que $f^k \in I$ e $f^{k-1} \notin I$, não pode ser representado na forma $I(V)$, ou seja, não pode ser gerado por qualquer variedade V .

Diversas são as questões que podem ser levantadas quanto às relações existentes entre os conceitos de ideal e variedade. Uma das principais questões diz respeito à relação existente entre os ideais $\langle f_1, \dots, f_s \rangle$ e $I(V(f_1, \dots, f_s))$ definidos acima (equações 3.1 e 3.2). Ocorre que $\langle f_1, \dots, f_s \rangle \subseteq I(V(f_1, \dots, f_s))$, não necessariamente havendo igualdade. Esta relação é fornecida pelo teorema que segue, chamado "Nullstellensatz"¹.

Teorema 7 ("Nullstellensatz" de Hilbert) *Considere as funções $f, f_1, \dots, f_s \in \mathbb{F}_q[x_1, \dots, x_n]$. A função $f \in I(V(f_1, \dots, f_s))$ se e só se $f^m \in \langle f_1, \dots, f_s \rangle$, para algum inteiro $m \geq 1$.*

Uma variedade afim V pode ser também definida como sendo um subconjunto de \mathbb{A}^n , em que $I(V)$ é um ideal primo. O conjunto de pontos racionais- \mathbb{G} de V é denotado por $V(\mathbb{G})$. Se $I(V)$ possui um conjunto de funções geradoras em $\mathbb{G}[x_1, \dots, x_n]$, diz-se que V é *definido sobre \mathbb{G}* , e denotado por $V_{\mathbb{G}}$. Daí, observa-se a seguinte relação:

$$I(V_{\mathbb{G}}) = I(V) \cap \mathbb{G}[x_1, \dots, x_n].$$

3.1.3 Bases de Gröbner

O método das *bases de Gröbner*, ou *bases padrão*, permite resolver de um modo elegante e eficiente, através de meios computacionais, diversos problemas envolvendo polinômios, tais como:

- O problema da descrição de um ideal: a determinação, se possível, de uma base para um ideal;
- O problema da pertinência a um ideal: a determinação de se uma função $f \in \mathbb{F}_q[x_1, \dots, x_n]$ pertence ou não a um ideal $\langle f_1, \dots, f_s \rangle$, sendo $f_1, \dots, f_s \in \mathbb{F}_q[x_1, \dots, x_n]$;

¹"Nullstellensatz" é uma palavra de origem alemã, composta por três palavras: "Null" (significa zero), "Stellen" (significa lugares) e "Satz" (significa teorema).

- O problema da solução de um sistema de equações polinomiais

$$f_1(x_1, \dots, x_n) = \dots = f_s(x_1, \dots, x_n) = 0,$$

o que equivale à determinação da variedade $V(f_1, \dots, f_s)$.

São diversas as aplicações para a teoria das bases de Gröbner. Contudo, no presente texto, buscar-se-á antes de tudo a solução do primeiro problema, ou seja, a determinação de uma base com características especiais para um determinado ideal, e a descrição das propriedades desta base. O segundo problema, o da pertinência a um ideal, será resolvido paralelamente ao primeiro com a descrição do algoritmo da divisão de polinômios em n variáveis, uma vez que uma função pertence a um ideal quando é divisível por uma das funções de sua base. Esta descrição do algoritmo da divisão é necessária à discussão do primeiro problema. Maiores detalhes sobre a teoria das bases de Gröbner podem ser obtidos em Cox, Little e O'Shea [3].

Ordenação de monômios

Primeiramente, observe que a todo monômio $x^\alpha = x_1^{\alpha_1} \dots x_n^{\alpha_n}$ pode-se associar uma n -úpla $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{Z}_+^n$, e vice-versa. Portanto, uma mesma regra de ordenação $>$ pode ser aplicada tanto a monômios quanto ao espaço \mathbb{Z}_+^n , ou seja, se $\alpha > \beta$ segundo a ordenação adotada, então $x^\alpha > x^\beta$, e vice-versa.

Definição 8 (Ordenação de monômios) *Uma ordenação de monômios em $\mathbb{F}_q[x_1, \dots, x_n]$ é qualquer relação $>$ sobre as n -úplas $\alpha \in \mathbb{Z}_+^n$, ou sobre os monômios x^α , que satisfaz as condições:*

1. $>$ é uma ordenação linear, ou seja, aplica-se a todos os vetores de \mathbb{Z}_+^n ;
2. Se $\alpha > \beta$, com $\alpha, \beta, \gamma \in \mathbb{Z}_+^n$, então $\alpha + \gamma > \beta + \gamma$;
3. Todo subconjunto finito não vazio de \mathbb{Z}_+^n possui um elemento menor e um elemento maior sob a ordenação $>$.

Um primeiro exemplo de ordenação de monômios é a chamada *ordenação lexicográfica*, denotada por $>_{lex}$, em que $\alpha >_{lex} \beta$, com $\alpha, \beta \in \mathbb{Z}_+^n$, se no vetor $\alpha - \beta \in \mathbb{Z}^n$ o elemento não nulo mais à esquerda for positivo. Por exemplo, para $n = 2$, tem-se

que $(0, 0) <_{lex} (0, 1) <_{lex} \dots <_{lex} (1, 0) <_{lex} (1, 1) <_{lex} \dots <_{lex} (2, 0) <_{lex} (2, 1) <_{lex} \dots <_{lex} (3, 0) <_{lex} (3, 1) <_{lex} \dots$.

Outro tipo de ordenação de monômios é a chamada *ordenação lexicográfica graduada*, denotada por $>_{grlex}$, em que $\alpha >_{grlex} \beta$, com $\alpha, \beta \in \mathbb{Z}_+^n$, se

$$|\alpha| = \sum_i \alpha_i > |\beta| = \sum_i \beta_i$$

ou se $|\alpha| = |\beta|$ e $\alpha >_{lex} \beta$. Por exemplo, para $n = 2$, tem-se que $(0, 0) <_{grlex} (0, 1) <_{grlex} (1, 0) <_{grlex} (0, 2) <_{grlex} (1, 1) <_{grlex} (2, 0) <_{grlex} (0, 3) <_{grlex} (1, 2) <_{grlex} (2, 1) <_{grlex} \dots$.

Uma ordenação que será utilizada no capítulo 5 na descrição dos algoritmos de decodificação é a chamada *ordenação lexicográfica graduada reversa*, denotada por $>_{grev}$, em que $\alpha >_{grev} \beta$, com $\alpha, \beta \in \mathbb{Z}_+^n$, se

$$|\alpha| = \sum_i \alpha_i > |\beta| = \sum_i \beta_i$$

ou $|\alpha| = |\beta|$ e o vetor $\alpha - \beta \in \mathbb{Z}^n$ tem seu elemento não nulo mais à esquerda negativo. Por exemplo, para $n = 2$, tem-se que $(0, 0) <_{grev} (1, 0) <_{grev} (0, 1) <_{grev} (2, 0) <_{grev} (1, 1) <_{grev} (0, 2) <_{grev} (3, 0) <_{grev} (2, 1) <_{grev} (1, 2) <_{grev} \dots$.

Considere $f = \sum_{\alpha} a_{\alpha} x^{\alpha}$ um polinômio não nulo em $\mathbb{F}_q[x_1, \dots, x_n]$ e $>$ uma ordenação de monômios. Defina, então, o *grau* de f , denotado por $\deg(f)$, como sendo o maior vetor α , expoente de x , segundo a ordenação $>$. Defina também o *coeficiente líder* de f , denotado por $lc(f)$, como sendo o coeficiente $a_{\deg(f)}$, e o *termo líder* de f , denotado por $lt(f)$, como sendo $a_{\deg(f)} x^{\deg(f)}$.

Divisão de polinômios

O algoritmo da divisão de polinômios em uma variável já é bem conhecido. Considere a divisão de uma função $f \in \mathbb{F}_q[x]$ pelas funções $f_1, \dots, f_s \in \mathbb{F}_q[x]$. Tem-se, então, que

$$f = q_1 f_1 + \dots + q_s f_s + r,$$

sendo q_1, \dots, q_s os quocientes e r o resto da divisão, com $\deg(r) < \deg(f_i)$, para $1 \leq i \leq s$.

Já o algoritmo da divisão de polinômios em $\mathbb{F}_q[x_1, \dots, x_n]$ (algoritmo 9) pode ser descrito como segue. Considere uma ordem de monômios $>$ e um conjunto de funções $f_1, \dots, f_s \in \mathbb{F}_q[x_1, \dots, x_n]$ ordenadas segundo $>$. Tem-se que toda função $f \in \mathbb{F}_q[x_1, \dots, x_n]$ pode ser escrita como sendo

$$f = q_1 f_1 + \dots + q_s f_s + r,$$

em que $q_1, \dots, q_s, r \in \mathbb{F}_q[x_1, \dots, x_n]$, e ou $r = 0$ ou r é uma combinação linear de monômios não divisíveis por $\text{lt}(f_1), \dots, \text{lt}(f_s)$.

Algoritmo 9 (Divisão de polinômios em várias variáveis)

Entradas:

- Os polinômios $f, f_1, \dots, f_s \in \mathbb{F}_q[x_1, \dots, x_n]$ ordenados segundo a ordem $>$.

Saídas:

- $q_1, \dots, q_s, r \in \mathbb{F}_q[x_1, \dots, x_n]$.

Inicialização:

- $q_1 = 0, \dots, q_s = 0, r = 0$.

<<< Passo único >>>

$p = f$

WHILE $p \neq 0$ **DO**

$i = 1$

$\text{ocorredivisao} = \text{false}$

WHILE $i \leq s$ **AND** $\text{ocorredivisao} == \text{false}$ **DO**

IF $\text{lt}(f_i)$ divide $\text{lt}(p)$ **THEN**

$q_i = q_i + \text{lt}(p) / \text{lt}(f_i)$

$p = p - f_i [\text{lt}(p) / \text{lt}(f_i)]$

$\text{ocorredivisao} = \text{true}$

ELSE

$i = i + 1$

IF $\text{ocorredivisao} == \text{false}$ **THEN**

$r = r + \text{lt}(p)$

$p = p - \text{lt}(p)$

Para o caso de polinômios em uma única variável, os valores de q_1, \dots, q_s e r são unicamente determinados para uma dada divisão. Já o caso geral da divisão de polinômios em n variáveis não é tão simples, uma vez que os valores dos quocientes e do resto dependem da ordem de monômios $>$ adotada. Por exemplo, a divisão do polinômio $f = xy^2 - x$ pelos polinômios $f_1 = xy + 1$ e $f_2 = y^2 - 1$ usando a ordem lexicográfica ($f_1 >_{lex} f_2$) resulta em

$$xy^2 - x = y(xy + 1) + 0(y^2 - 1) + (-x - y).$$

Contudo, a mesma divisão usando a ordem lexicográfica graduada reversa ($f_2 >_{grev} f_1$) fornece um resultado diferente, a saber

$$xy^2 - x = x(y^2 - 1) + 0(xy + 1) + 0.$$

Apesar disto, no caso de polinômios em n variáveis, o problema da pertinência de uma função f a um ideal $\langle f_1, \dots, f_s \rangle$ pode ainda ser resolvido através da divisão de polinômios. Se uma função f pode ser escrita na forma

$$f = q_1 f_1 + \dots + q_s f_s,$$

tem-se, então, que $f \in \langle f_1, \dots, f_s \rangle$. Isto implica que $r = 0$ é uma condição suficiente para a pertinência de f ao ideal $\langle f_1, \dots, f_s \rangle$. Contudo, não é uma condição necessária, pois a mesma divisão pode fornecer um resto $r \neq 0$ para uma ordenação de monômios diferente.

Há, entretanto, a possibilidade de, a partir de um conjunto de funções geradoras de um ideal I , obter-se um conjunto diferente de funções geradoras de I que apresentem boas propriedades, tais como a de serem unicamente determinadas e a de tornarem a condição $r = 0$ suficiente e necessária para a pertinência de uma função f ao ideal I . Será visto que as chamadas bases de Gröbner apresentam estas propriedades.

Ideais de monômios e bases de Gröbner

Considere a seguinte definição.

Definição 10 (Ideal de monômios) *Um ideal $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ é dito ser um ideal de monômios se for constituído de todos os polinômios na forma da soma finita*

$$\sum_{\alpha \in A} h_\alpha x^\alpha,$$

em que $A \subset \mathbb{Z}_+^n$ e $h_\alpha \in \mathbb{F}_q[x_1, \dots, x_n]$.

Um fato importante acerca desta definição é que todo ideal de monômios I pode ser escrito na forma $\langle x^{\alpha_1}, \dots, x^{\alpha_s} \rangle$, em que $\alpha_1, \dots, \alpha_s \in A$. Em outras palavras, todo ideal de monômios I possui uma base finita de monômios. Observe, então, que um monômio x^β pertence a um ideal de monômios $I = \langle x^\alpha \mid \alpha \in A \rangle$ se e só se for divisível por algum x^α , para $\alpha \in A$. Como consequência, todo polinômio $f \in I$ é necessariamente uma combinação linear de monômios de I , ou seja, todo termo de f pertence também a I .

Considere agora $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ um ideal qualquer diferente de $\{0\}$ e denote por $\text{lt}(I)$ o conjunto dos termos líderes dos elementos de I . Tem-se que o ideal gerado pelos elementos de $\text{lt}(I)$, denotado por $\langle \text{lt}(I) \rangle$, constitui um ideal de monômios. Além disso, existem sempre funções $g_1, \dots, g_t \in I$, tais que $\langle \text{lt}(I) \rangle = \langle \text{lt}(g_1), \dots, \text{lt}(g_t) \rangle$. Associado a este fato, o chamado *teorema das bases de Hilbert* afirma que todo e qualquer ideal $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ possui sempre um conjunto gerador finito, ou seja, todo ideal I pode ser escrito na forma $\langle g_1, \dots, g_t \rangle$, para algum $g_1, \dots, g_t \in I$. Segue, então, a definição de bases de Gröbner.

Definição 11 (Bases de Gröbner) *Dada uma ordem de monômios, um subconjunto finito $\mathcal{G} = \{g_1, \dots, g_t\}$ de um ideal I é dito ser uma base de Gröbner (também chamada base padrão) se*

$$\langle \text{lt}(g_1), \dots, \text{lt}(g_t) \rangle = \langle \text{lt}(I) \rangle.$$

Ainda, um conjunto $\mathcal{G} = \{g_1, \dots, g_t\} \subset I$ é dito ser uma base de Gröbner se e só se o termo líder de qualquer elemento de I for divisível por algum $\text{lt}(g_i)$, com $g_i \in \mathcal{G}$.

Uma base de Gröbner \mathcal{G} de um ideal I é dita *mínima* se $\text{lc}(g) = 1$ e $\text{lt}(g) \notin \langle \text{lt}(\mathcal{G} - \{g\}) \rangle$, para todo $g \in \mathcal{G}$. Uma base de Gröbner \mathcal{G} de um ideal I é dita *reduzida* se for mínima e nenhum monômio de g pertencer a $\langle \text{lt}(\mathcal{G} - \{g\}) \rangle$, para todo $g \in \mathcal{G}$.

Seguem algumas propriedades das bases de Gröbner:

- Todo ideal $I \neq \{0\}$ possui uma base de Gröbner;
- Dada uma ordem de monômios, todo ideal $I \neq \{0\}$ possui uma única base de Gröbner reduzida;
- O resto r da divisão de uma função $f \in \mathbb{F}_q[x_1, \dots, x_n]$ por uma base de Gröbner \mathcal{G} de um ideal $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ é unicamente determinado, independentemente da ordem de monômios adotada.

Observe que o problema da pertinência de uma função f a um ideal I reduz-se à determinação de uma base de Gröbner \mathcal{G} para I . Portanto, tem-se que $f \in I$ se e só se a divisão de f por \mathcal{G} for exata (resto $r = 0$).

Algoritmos, como o de Buchberger [3], podem ser utilizados na determinação de uma base de Gröbner para um dado ideal $I = \langle f_1, \dots, f_s \rangle$. Para a descrição de uma versão simples do algoritmo de Buchberger (algoritmo 12), observe a seguinte notação.

Denote por $\bar{f}^{\mathcal{F}}$ o resto da divisão de uma função f pelo conjunto ordenado $\mathcal{F} = \{f_1, \dots, f_s\}$ (se \mathcal{F} for uma base de Gröbner, a ordem de seus elementos torna-se irrelevante).

Considere agora os polinômios não nulos $p, q \in \mathbb{F}_q[x_1, \dots, x_n]$. Considere os vetores $\alpha = \deg(p)$, $\beta = \deg(q)$ e $\gamma = (\gamma_1, \dots, \gamma_n)$, em que $\gamma_i = \max\{\alpha_i, \beta_i\}$. Defina, então, o *mínimo múltiplo comum* de $\text{lt}(p)$ e $\text{lt}(q)$ como sendo

$$\text{MMC}(\text{lt}(p), \text{lt}(q)) = x^\gamma.$$

Defina, por fim, o *polinômio-S* de p e q como sendo

$$S(p, q) = \frac{x^\gamma}{\text{lt}(p)}p - \frac{x^\gamma}{\text{lt}(q)}q.$$

Algoritmo 12 (Buchberger)

Entradas:

- Um conjunto gerador $\mathcal{F} = \{f_1, \dots, f_s\}$ para um ideal $I \subset \mathbb{F}_q[x_1, \dots, x_n]$.

Saídas:

- Uma base de Gröbner $\mathcal{G} = \{g_1, \dots, g_t\}$ para I .

<<< Passo único >>>

$\mathcal{G} = \mathcal{F}$

DO

$\mathcal{G}' = \mathcal{G}$

FOR cada $\{p, q\}$, com $p, q \in \mathcal{G}'$ e $p \neq q$ DO

$S = \overline{S(p, q)}^{\mathcal{G}'}$

IF $S \neq 0$ THEN $\mathcal{G} = \mathcal{G} \cup \{S\}$

WHILE $\mathcal{G} \neq \mathcal{G}'$

As aplicações são inúmeras para esta teoria das bases de Gröbner, que é também útil, como será visto no capítulo 5, na construção de algoritmos eficientes de decodificação para códigos para controle de erros.

3.1.4 Anel de coordenadas de uma variedade afim

Considere o ideal $I \subset \mathbb{F}_q[x_1, \dots, x_n]$ e faça $f, g \in \mathbb{F}_q[x_1, \dots, x_n]$. Diz-se que f e g são *congruentes módulo I* , denotando-se por

$$f \equiv g \pmod{I},$$

se $f - g \in I$. Esta operação de congruência módulo I estabelece uma relação de equivalência, que subdivide $\mathbb{F}_q[x_1, \dots, x_n]$ nas chamadas *classes de equivalência*. Para qualquer $f \in \mathbb{F}_q[x_1, \dots, x_n]$, define-se a classe de f como sendo o conjunto

$$[f] = \{g \in \mathbb{F}_q[x_1, \dots, x_n] \mid g \equiv f \pmod{I}\}.$$

Defina agora o quociente de $\mathbb{F}_q[x_1, \dots, x_n]$ módulo I , denotado por $\mathbb{F}_q[x_1, \dots, x_n]/I$, como sendo o conjunto de classes de equivalência por congruência módulo I dado por

$$\mathbb{F}_q[x_1, \dots, x_n]/I = \{[f] \mid f \in \mathbb{F}_q[x_1, \dots, x_n]\}.$$

Um quociente $\mathbb{F}_q[x_1, \dots, x_n]/I$ constitui, na verdade, um anel comutativo sobre as operações $[f] + [g] = [f + g]$ de adição e $[f] \cdot [g] = [f \cdot g]$ de multiplicação realizadas entre as classes de equivalência, sendo, por isso, denominado *anel quociente*.

Definição 13 (Anel de coordenadas) *Considere uma variedade afim V . No caso em que $I = I(V)$ (um ideal radical), o anel quociente dado por*

$$\Gamma(V) = \mathbb{F}_q[x_1, \dots, x_n]/I(V)$$

é denominado o anel de coordenadas de V .

Se V é definido sobre \mathbb{G} , o anel quociente $\Gamma(V_{\mathbb{G}}) = \mathbb{G}[x_1, \dots, x_n]/I(V_{\mathbb{G}})$ é chamado anel de coordenadas de $V_{\mathbb{G}}$.

3.1.5 Corpo de funções

Utilizando-se o anel de inteiros \mathbb{Z} , é possível se contruir diversos corpos. Um caso simples é o corpo de número racionais \mathbb{Q} , que é constituído por elementos da forma $\frac{m}{n}$, em que $m, n \in \mathbb{Z}$. Analogamente, a partir do anel de polinômios $\mathbb{F}_q[x_1, \dots, x_n]$, pode-se contruir um corpo com elementos da forma $\frac{f}{g}$, em que $f, g \in \mathbb{F}_q[x_1, \dots, x_n]$, denominados *funções racionais*.

Definição 14 (Corpo de funções) O corpo denotado por $\mathbb{F}_q(x_1, \dots, x_n)$ e dado por

$$\mathbb{F}_q(x_1, \dots, x_n) = \left\{ \frac{f(x_1, \dots, x_n)}{g(x_1, \dots, x_n)} \mid f, g \in \mathbb{F}_q[x_1, \dots, x_n], g \neq 0 \right\}$$

é denominado corpo de funções.

Se V é uma variedade afim e $\Gamma(V)$ seu anel de coordenadas, então o corpo das frações $\frac{f}{g}$, com $f, g \in \Gamma(V)$, é dito o corpo de funções de V e denotado por $\mathbb{F}_q(V)$. Se V é definido sobre \mathbb{G} , o corpo de funções de $V_{\mathbb{G}}$, denotado por $\mathbb{G}(V)$, é o corpo das frações $\frac{f}{g}$, com $f, g \in \Gamma(V_{\mathbb{G}})$.

A *dimensão* de uma variedade afim V é o grau de transcendência do corpo $\mathbb{F}_q(V)$ sobre o corpo \mathbb{F}_q . Desta forma, define-se uma *curva algébrica* $\mathcal{X} \subseteq \mathbb{A}^n$ como uma variedade de dimensão 1.

Uma curva \mathcal{X} é dita ser *não singular*, ou *regular*, se todos os seus pontos forem não singulares, ou seja, possuírem os mesmos zeros em duas derivadas parciais. Caso contrário, a curva é dita *singular*.

3.1.6 Variedade projetiva

As variedades tratadas até então têm sido subconjuntos do espaço afim \mathbb{A}^n . A adição a \mathbb{A}^n de "pontos no infinito" permite criar o chamado espaço projetivo \mathbb{P}^n de dimensão n , de onde derivam as variedades projetivas.

Espaço projetivo

Antes de definir propriamente o espaço projetivo \mathbb{P}^n sobre \mathbb{F}_q , de modo a auxiliar em sua compreensão, considere o caso do corpo \mathbb{R} dos números reais. Observe que duas linhas em \mathbb{R}^2 sempre se interceptam em um ponto, exceto linhas paralelas. Esta exceção

pode ser contornada pela consideração de pontos no infinito, um para cada coleção de retas paralelas com determinada inclinação (todas as retas com mesma inclinação), ou classes de equivalência de linhas paralelas (figura 3.1). Defina o plano projetivo $\mathbb{P}_{\mathbb{R}^2}$ como sendo a união de \mathbb{R}^2 com o conjunto dos pontos no infinito de cada classe de equivalência de linhas paralelas. Uma linha projetiva em $\mathbb{P}_{\mathbb{R}^2}$ é a união de uma reta em \mathbb{R}^2 com o ponto no infinito correspondente à classe de equivalência à qual a reta pertence. Observe, portanto, que duas linhas projetivas sempre determinam um ponto e que dois pontos quaisquer determinam uma linha projetiva (o conjunto dos pontos no infinito forma uma linha no infinito).

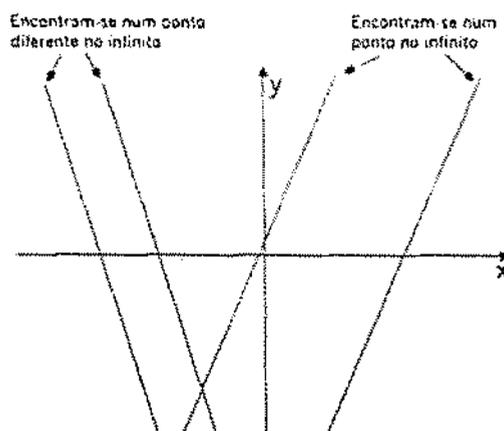


Figura 3.1: Curvas paralelas em \mathbb{R}^2 encontram-se num único ponto no infinito. Curvas paralelas com inclinação diferente encontram-se em um ponto diferente no infinito.

O problema da representação dos pontos em $\mathbb{P}_{\mathbb{R}^2}$ é resolvido da seguinte forma:

- Acrescenta-se uma terceira coordenada, mapeando-se um ponto $(x, y) \in \mathbb{R}^2$ em um ponto $(x, y, z) \in \mathbb{R}^3 \setminus \{(0, 0, 0)\}$;
- Define-se uma relação de equivalência em que $(x, y, z) \equiv \lambda(x, y, z)$, para qualquer $\lambda \in \mathbb{R} \setminus \{0\}$.

Diz-se que $(x, y, z) \in \mathbb{R}^3 \setminus \{(0, 0, 0)\}$, segundo esta regra de equivalência, são as *coordenadas homogêneas* dos pontos de $\mathbb{P}_{\mathbb{R}^2}$. Desta forma, vê-se que o plano projetivo é constituído pelos pontos $(1, y, z)$, equivalentes aos pontos $(y, z) \in \mathbb{R}^2$, e pelos pontos $(0, y, z)$, que constituem a linha projetiva no infinito.

Esta mesma idéia apresentada de classes de equivalência de linhas paralelas, pontos no infinito e coordenadas homogêneas pode ser estendida a espaços de maior dimensão e a corpos finitos.

De modo similar ao que foi feito acima, considere a relação de equivalência no conjunto $\mathbb{A}^{n+1} \setminus \{(0, 0, \dots, 0)\}$, com termos em \mathbb{F}_q , dada por

$$(a_0, \dots, a_n) \equiv (b_0, \dots, b_n) \Leftrightarrow \exists \lambda \in \mathbb{F}_q \setminus \{0\} \mid b_i = \lambda a_i, i = 0, 1, \dots, n.$$

A classe de equivalência de (a_0, \dots, a_n) é denotada por $(a_0 : \dots : a_n)$.

Definição 15 (Espaço projetivo) O espaço projetivo \mathbb{P}^n de dimensão n é o conjunto de todas as classes de equivalência $\{(a_0 : \dots : a_n) \mid a_i \in \mathbb{F}_q\}$.

Um elemento $P = (a_0 : \dots : a_n) \in \mathbb{P}^n$ é chamado *ponto*, e a_0, \dots, a_n são ditas as coordenadas homogêneas de P . Se \mathbb{G} é um subcorpo de \mathbb{F}_q que contém \mathbb{F}_p e $P = (a_0 : \dots : a_n)$ é um ponto com coordenadas homogêneas $a_0, \dots, a_n \in \mathbb{G}$, então P é dito um *ponto racional- \mathbb{G}* e o conjunto dos pontos racionais- \mathbb{G} de \mathbb{P}^n é denotado por $\mathbb{P}^n(\mathbb{G})$.

O conjunto $H = \{(0 : a_1 : \dots : a_n) \in \mathbb{P}^n\}$ é chamado de *hiperplano no infinito* e os pontos de H , denotados por Q , são chamados *pontos no infinito*. O mapeamento $\varphi : \mathbb{A}^n \rightarrow \mathbb{P}^n \setminus H$ é definido por $\varphi(a_1, \dots, a_n) = (1, a_1, \dots, a_n)$.

O espaço projetivo de dimensão 1, também chamado de linha projetiva, consiste nos pontos $(1 : a)$, com $a \in \mathbb{F}_q$, e do ponto no infinito $Q = (0 : 1)$. Este conjunto foi utilizado no capítulo anterior na construção dos códigos de Reed-Solomon.

Variedade projetiva

Um polinômio constituído pela soma de monômios de mesmo grau é chamado *polinômio homogêneo*. Um polinômio homogêneo $f \in \mathbb{F}_q[x_0, \dots, x_n]$ pode ser considerado como um mapeamento $f : \mathbb{P}^n \rightarrow \mathbb{F}_q$ dado por

$$f(P) = f(a_0, \dots, a_n),$$

em que $P = (a_0 : \dots : a_n) \in \mathbb{P}^n$, com $a_0, \dots, a_n \in \mathbb{F}_q$. Se $f(P) = 0$, f é dito ter um zero no ponto $P = (a_0 : \dots : a_n) \in \mathbb{P}^n$. Isto faz sentido, uma vez que, se f é homogêneo de grau d ,

$$f(\lambda a_0, \dots, \lambda a_n) = \lambda^d f(a_0, \dots, a_n).$$

Definição 16 (Varietade projetiva) Considere $f_1, \dots, f_s \in \mathbb{F}_q[x_0, \dots, x_n]$ polinômios homogêneos. A variedade projetiva $V(f_1, \dots, f_s)$ definida por f_1, \dots, f_s é, então, dada por

$$V(f_1, \dots, f_s) = \{P \in \mathbb{P}^n \mid f_i(P) = 0, 1 \leq i \leq s\}.$$

Observe que a soma de dois polinômios homogêneos de diferentes graus não preserva a homogeneidade. Portanto, um ideal $I = \langle f_1, \dots, f_s \rangle \subset \mathbb{F}_q[x_0, \dots, x_n]$ gerado por polinômios homogêneos sempre contém polinômios não homogêneos, que não podem ser usados na definição de uma variedade projetiva. Observe também que, apesar disto, todos os polinômios deste ideal I se anulam nos pontos da variedade projetiva $V(f_1, \dots, f_s)$. Além disto, mostra-se que, para todo $f \in I$, se f_i é uma componente (parte) homogênea de f , então $f_i \in I$. Estes ideais I gerados por polinômios homogêneos são chamados *ideais homogêneos*.

De forma similar ao caso afim, uma variedade projetiva V pode ser também definida como um subconjunto de \mathbb{P}^n , tal que $I(V)$ é um ideal primo homogêneo. O conjunto de pontos racionais- \mathbb{G} da variedade projetiva V é denotado por $V(\mathbb{G})$. Se $I(V)$ possui um conjunto de polinômios homogêneos geradores em $\mathbb{G}[x_0, \dots, x_n]$, diz-se que V é *definida sobre \mathbb{G}* e denotado por $V_{\mathbb{G}}$. Com isso, observa-se a seguinte relação:

$$I(V_{\mathbb{G}}) = I(V) \cap \mathbb{G}[x_0, \dots, x_n].$$

Define-se, assim como no caso afim, o *anel de coordenadas homogêneo* de uma variedade projetiva $V \subset \mathbb{P}^n$ como sendo o anel quociente

$$\Gamma_h(V) = \mathbb{F}_q[x_0, \dots, x_n] / I(V).$$

Se V é definido sobre \mathbb{G} , então $\Gamma_h(V_{\mathbb{G}}) = \mathbb{G}[x_0, \dots, x_n] / I(V_{\mathbb{G}})$.

Um elemento $f \in \Gamma_h(V)$ é dito ser uma *forma* de grau d se $f = F + I(V)$, em que $F \in \mathbb{F}_q[x_0, \dots, x_n]$ é um polinômio homogêneo de grau d .

O *corpo de funções* de uma variedade projetiva V é definido por

$$\mathbb{F}_q(V) = \left\{ \frac{f}{g} \mid f, g \in \Gamma_h(V) \text{ são formas de mesmo grau e } g \neq 0 \right\}$$

e

$$\mathbb{G}(V) = \left\{ \frac{f}{g} \mid f, g \in \Gamma_h(V_{\mathbb{G}}) \text{ são formas de mesmo grau e } g \neq 0 \right\}.$$

A *dimensão* da variedade projetiva V é o grau de transcendência de $\mathbb{F}_q(V)$ sobre \mathbb{F}_q . Define-se uma *curva projetiva* $\mathcal{X} \subseteq \mathbb{P}^n$ como sendo uma variedade projetiva de dimensão 1.

3.1.7 Relação entre variedades afim e projetiva

É possível sempre converter qualquer polinômio não homogêneo em um polinômio homogêneo. Para qualquer polinômio $f \in \mathbb{F}_q[x_1, \dots, x_n]$ de grau d , o polinômio $f' \in \mathbb{F}_q[x_0, \dots, x_n]$ dado por

$$f'(x_0 : \dots : x_n) = x_0^d f\left(\frac{x_1}{x_0}, \dots, \frac{x_n}{x_0}\right)$$

é homogêneo de grau d .

Considere agora uma variedade afim $V \in \mathbb{A}^n$ e o ideal correspondente $I(V) \subset \mathbb{F}_q[x_1, \dots, x_n]$. Define-se uma variedade projetiva $\tilde{V} \subset \mathbb{P}^n$, chamada *fechamento projetivo* de V , como sendo

$$\tilde{V} = \{P \in \mathbb{P}^n \mid f'(P) = 0, f' \in I(V)\}.$$

Em contrapartida, considere uma variedade projetiva $\tilde{V} \subset \mathbb{P}^n$ e suponha que

$$W = \tilde{V} \cap \{(a_0 : \dots : a_n) \in \mathbb{P}^n \mid a_0 \neq 0\} \neq \emptyset.$$

Defina o mapeamento $\varphi : \mathbb{A}^n \rightarrow \mathbb{P}^n$ como sendo $\varphi(a_1, \dots, a_n) = (1 : a_1 : \dots : a_n)$. Então,

$$V = \varphi^{-1}(W)$$

é uma variedade afim,

$$I(V) = \left\{ f(1 : x_1 : \dots : x_n) \mid f \in I(\tilde{V}) \right\}$$

é seu ideal e \tilde{V} é o fechamento projetivo de V .

Se V é uma variedade afim e \tilde{V} seu fechamento projetivo, os corpos de funções $\mathbb{F}_q(V)$ e $\mathbb{F}_q(\tilde{V})$ são isomórficos e V e \tilde{V} possuem a mesma dimensão.

3.2 Anel local

Considere a variedade V e um ponto $P \in V$. Considere $f \in \mathbb{F}_q(V)$ uma função racional, então $f = \frac{g}{h}$, sendo $g, h \in \Gamma(V)$ para uma variedade V afim, ou $g, h \in \Gamma_h(V)$ para uma variedade V projetiva. Se $h(P) \neq 0$, então f é dito ser *definido* em P .

Definição 17 (Anel local) O anel denotado por $O_P(V)$ e dado por

$$O_P(V) = \{f \in \mathbb{F}_q(V) \mid f \text{ é definido em } P\}$$

é chamado o anel local em P .

A avaliação de uma função $f \in O_P(V)$ em um ponto P é feita da seguinte forma:

- No caso em que V é uma variedade afim $\Rightarrow f(P) = \frac{g(P)}{h(P)}$.
- No caso em que V é uma variedade projetiva \Rightarrow Considere $g, h \in \Gamma_h(V)$, com $g = G + I(V)$ e $h = H + I(V)$, em que G e H são polinômios homogêneos de grau d . Tem-se, então, que $f(P) = \frac{G(P)}{H(P)}$ se $H(P) \neq 0$.

Defina o *ideal máximo* $M_P(V)$ de um anel local $O_P(V)$ como sendo

$$M_P(V) = \{f \in O_P(V) \mid f(P) = 0\}.$$

3.2.1 Anel de valorização

Definição 18 (Anel de valorização) Chama-se anel de valorização de um corpo de funções $\mathbb{F}_q(V)$ um anel O , tal que $\mathbb{F}_q \subset O \subset \mathbb{F}_q(V)$ e, para todo $z \in \mathbb{F}_q(V)$, $z \in O$ ou $z^{-1} \in O$.

Este anel de valorização O de $\mathbb{F}_q(V)$ é também um anel local, tendo como único ideal máximo $\mathcal{P} = O \setminus O^*$, em que $O^* = \{z \in O \mid \exists w \in O, zw = 1\}$. Este único ideal máximo \mathcal{P} de O é dito ser um *lugar*² de $\mathbb{F}_q(V)$. O conjunto dos lugares de um corpo de funções $\mathbb{F}_q(V)$ é denotado por

$$\mathbb{P} = \{\mathcal{P} \mid \mathcal{P} \text{ é um lugar de } \mathbb{F}_q(V)\}. \quad (3.3)$$

²O conceito de lugar é freqüentemente associado a ou confundido com o conceito de ponto. Observe uma analogia com números reais e complexos. Um número $a \in \mathbb{R}$ constitui um ponto no eixo horizontal do plano \mathbb{R}^2 . Pode-se dizer que o conjunto dos números complexos $a + bi$, com $b \in \mathbb{R}$, constitui um lugar. Um lugar costuma ser vulgarmente referido como um "ponto gordo".

Observa-se que, se \mathbb{F}_q é um corpo fechado algebricamente, que é o caso aqui considerado, então \mathcal{P} possui grau 1 (veja definição de grau de um ponto fechado, ou lugar, na subseção seguinte). Se um lugar \mathcal{P} possuir grau 1, então ele é equivalente a um ponto racional P .

Um anel de valorização O apresenta ainda as seguintes propriedades:

- Sendo $0 \neq z \in \mathbb{F}_q(V)$, então $z \in \mathcal{P} \Leftrightarrow z^{-1} \notin O$;
- \mathcal{P} é um ideal principal;
- Se $\mathcal{P} = tO$, então qualquer $0 \neq z \in \mathbb{F}_q(V)$ possui uma representação única da forma $z = t^n u$, com $u \in O^*$ e $n \in \mathbb{Z}$;
- Se $\mathcal{P} = tO$ e $\{0\} \neq I \subseteq O$ é um ideal, então $I = t^n O$ para algum $n \in \mathbb{N}$.

A função t é chamada *parâmetro local*, *parâmetro de uniformização* ou *elemento primo* de \mathcal{P} . Um anel de valorização O , cujo respectivo lugar $\mathcal{P} = tO$, é chamado *anel de valorização discreto*.

Considere O um anel de valorização discreto de $\mathbb{F}_q(V)$ e \mathcal{P} seu único ideal máximo. Tem-se que, para $z \in \mathbb{F}_q(V)$, $z = t^n u$, com $u \in O^*$ e $n \in \mathbb{Z}$. Defina, então, a função $v : \mathbb{F}_q(V) \rightarrow \mathbb{Z}$, chamada *função de valorização discreta* de $\mathbb{F}_q(V)$, como sendo

$$v_{\mathcal{P}}(z) = \begin{cases} \infty, & z = 0; \\ n, & z \neq 0. \end{cases} \quad (3.4)$$

Esta função apresenta as seguintes propriedades:

- $v_{\mathcal{P}}(x) = \infty \Leftrightarrow x = 0$;
- $v_{\mathcal{P}}(xy) = v_{\mathcal{P}}(x) + v_{\mathcal{P}}(y)$, para qualquer $x, y \in \mathbb{F}_q(V)$;
- $v_{\mathcal{P}}(x + y) \geq \min[v_{\mathcal{P}}(x), v_{\mathcal{P}}(y)]$, com igualdade se $v_{\mathcal{P}}(x) \neq v_{\mathcal{P}}(y)$;
- Existe um elemento $z \in \mathbb{F}_q(V)$, tal que $v_{\mathcal{P}}(z) = 1$;
- $v_{\mathcal{P}}(a) = 0$, para todo $0 \neq a \in \mathbb{F}_q$.

A subseção seguinte estabelece uma conexão entre uma variedade V e os anéis de valorização discretos do seu corpo de funções $\mathbb{F}_q(V)$.

3.2.2 Relação entre anel de valorização discreto e variedade

Considere uma variedade V e seu ideal $I(V) = \langle G_1, \dots, G_s \rangle$. Considere P um ponto de V e a matriz

$$J_{V,P} = \{a_{ij}\},$$

em que $a_{ij} = \frac{\partial G_i(P)}{\partial x_j}$, para $i = 1, \dots, s$ e $j = 1, \dots, n$ (caso afim) ou $j = 0, \dots, n$ (caso projetivo). O ponto P é dito *não singular* se o posto de $J_{V,P}$ for igual a $n - \dim V$, e *singular* em caso contrário.

Se \mathcal{X} uma curva (afim ou projetiva) e P um ponto de \mathcal{X} , o ponto P é não singular se e só se $O_P(\mathcal{X})$ for um anel de valorização discreto.

Se uma variedade é definida sobre \mathbb{G} , pode-se também considerar o corpo de funções $\mathbb{G}(V)$. As definições e resultados apresentados sobre o corpo \mathbb{F}_q também são válidos para \mathbb{G} . Se v é uma função de valorização discreta de $\mathbb{G}(V)$, com um anel de valorização O e um lugar \mathcal{P} , então o par (O, \mathcal{P}) é dito um *ponto fechado* de V e $[O/\mathcal{P} : \mathbb{G}]$ é dito o *grau* deste ponto. Se $\mathbb{G} = \mathbb{F}_q$, então os pontos fechados correspondem aos pontos não singulares e todos possuem grau 1.

Denota-se por $\mathcal{P}_{\mathcal{X}}$ o conjunto dos pontos fechados de uma curva \mathcal{X} .

3.3 Divisores

O conceito de divisor constitui uma forma extremamente elegante de manipular um conjunto de pontos relacionados a uma curva. A partir dele, derivam-se espaços vetoriais de funções, estando toda a construção dos CGA's alicerçada neste conceito.

Esta seção apresenta a definição de divisores e sua utilização na formação de espaços vetoriais de funções. Apresenta também o importante teorema de Riemann-Roch, que determina a dimensão do espaço de funções gerado por um divisor, além dos conceitos de lacuna e anti-lacuna.

3.3.1 Conceito de divisor

Considere \mathcal{X} uma curva projetiva definida sobre \mathbb{F}_q .

Definição 19 (Divisor) Um divisor D de \mathcal{X} é uma soma formal (um conjunto de pontos dispostos em forma de uma soma ponderada) dada por

$$D = \sum_{P \in \mathcal{X}} n_P P,$$

em que $n_P \in \mathbb{Z}$, podendo ser $n_P = 0$ para alguns, mas não todos, dos pontos $P \in \mathcal{X}$.

O suporte de D é definido como sendo

$$\text{sup } D = \{P \in \mathcal{X} \mid n_P \neq 0\}.$$

Um divisor D é dito ser *efetivo*, denotando-se por $D \succ 0$, se todo n_P for não negativo.

Os divisores de uma curva \mathcal{X} formam um grupo abeliano, denotado por $\text{Div}(\mathcal{X})$, chamado *grupo dos divisores* de \mathcal{X} . O grau de um divisor D é dado por

$$\text{deg } D = \sum_{P \in \mathcal{X}} n_P,$$

o que constitui um mapeamento $\text{deg} : \text{Div}(\mathcal{X}) \rightarrow \mathbb{Z}$.

Considere uma função racional $f \in \mathbb{F}_q(\mathcal{X})$. A *ordem* de uma função f em um ponto racional $P \in \mathcal{X}$ (um ponto racional P equivale a um lugar de grau 1) é definida como sendo $v_P(f)$, em que v_P é a função de valorização discreta (equação 3.4) correspondente ao anel de valorização discreto \mathcal{O} do corpo de funções $\mathbb{F}_q(\mathcal{X})$. Se $v_P(f) > 0$, diz-se que f possui um *zero* no ponto P , e se $v_P(f) < 0$, diz-se que f possui um *pólo* em P .

Defina agora o *divisor principal* (f) da função racional $f \in \mathbb{F}_q(\mathcal{X})$ como sendo

$$(f) = \sum_{P \in \mathcal{X}} v_P(f) P, \tag{3.5}$$

o *divisor de zeros* $(f)_0$ de f como sendo

$$(f)_0 = \sum_{v_P(f) > 0} v_P(f) P$$

e o *divisor de pólos* $(f)_\infty$ de f como sendo

$$(f)_\infty = - \sum_{v_P(f) < 0} v_P(f) P.$$

O grau de um divisor principal é, por definição, nulo, o que implica que

$$\sum_{v_P(f) > 0} v_P(f) = - \sum_{v_P(f) < 0} v_P(f).$$

Dado um grupo de divisores $Div(\mathcal{X})$, define-se uma ordem entre os elementos deste grupo da forma

$$D_1 = \sum_{P \in \mathcal{P}_X} n_P P \leq D_2 = \sum_{P \in \mathcal{P}_X} n'_P P$$

se e só se

$$n_P \leq n'_P,$$

para todo $P \in \mathcal{X}$.

3.3.2 Espaço de funções de um divisor

A definição seguinte descreve a formação de um espaço vetorial de funções a partir de um determinado divisor.

Definição 20 (Espaço de funções de um divisor) *Considere $G \in Div(\mathcal{X})$ um divisor de uma curva \mathcal{X} , então*

$$L(G) = \{f \in \mathbb{F}_q(\mathcal{X}) \mid (f) + G \succ 0\} \cup \{0\}$$

é o espaço vetorial das funções racionais com pólos determinados pelos pontos do divisor G , com multiplicidades determinadas pelas ordens destes pontos.

Observe que o divisor do produto de duas funções $f, h \in \mathbb{F}_q(\mathcal{X})$ é a soma dos respectivos divisores, ou seja, $(fh) = (f) + (h)$. Ocorre também que o divisor da soma destas funções satisfaz a desigualdade $(f+h) \geq \min\{(f), (h)\}$, em que $\min\{(f), (h)\}$ é o divisor formado com os menores coeficientes v_P (equação 3.5) ponto a ponto.

O espaço vetorial $L(G)$ definido sobre \mathbb{F}_q possui dimensão finita, denotada por $l(G)$. Esta dimensão é determinada pelo teorema de Riemann-Roch, que é tratado numa subseção seguinte.

3.3.3 Diferencial

De modo a determinar a dimensão $l(G)$ do espaço vetorial $L(G)$, torna-se necessário o uso e entendimento do conceito de *diferencial*.

Na matemática clássica, uma integral

$$\int_C f dx$$

consiste num operador que fornece um número a partir de três entradas:

1. Um caminho C , que costuma ser fechado e simples;
2. Uma função f sem pólos em C ;
3. Um diferencial dx .

O diferencial é o mais obscuro dos três conceitos, uma vez que suas propriedades permanecem normalmente implícitas. Observe que se pode ter um diferencial

$$dy = \frac{dy}{dx} dx = f dx.$$

Sendo $f = \frac{dy}{dx}$ uma função, tem-se que diferenciais na matemática clássica compõem um espaço vetorial unidimensional sobre um espaço de funções.

No caso do presente estudo, pode-se imaginar diferenciais como sendo objetos da forma $f dh$, em que $f, h \in \mathbb{F}_q(\mathcal{X})$ são funções racionais associadas à curva \mathcal{X} e o mapeamento linear que leva de h para dh é denominado *derivação*³. Para esta derivação vale a conhecida regra

$$d(h_1 h_2) = h_1 dh_2 + h_2 dh_1.$$

Para cada ponto fechado (O, \mathcal{P}) de \mathcal{X} , existe um parâmetro local t , em que $v_{\mathcal{P}}(t) = 1$ (equação 3.4). Também, para cada diferencial denotado por ω , existe uma função f , tal que $\omega = f dt$. Tem-se, então, que a função de valorização discreta $v_{\mathcal{P}}(\omega)$ é, por definição, igual a $v_{\mathcal{P}}(f)$.

Assim como no caso das funções racionais, pode-se falar em pólos e zeros de diferenciais. Diz-se que ω possui um zero de ordem ρ , se $\rho = v_{\mathcal{P}}(f) > 0$, e que ω possui um pólo de ordem ρ , se $\rho = -v_{\mathcal{P}}(f) > 0$.

³Será omitida aqui uma definição formal ou mais adequada deste mapeamento, o que demandaria uma longa discussão envolvendo outros conceitos de menor relevância para o presente trabalho.

Define-se o *divisor* de um diferencial ω como sendo

$$(\omega) = \sum v_{\mathcal{P}}(f) \mathcal{P}.$$

O divisor de um diferencial é dito ser *canônico* e possui sempre grau $2g - 2$, em que g é o gênero da curva \mathcal{X} definido em seguida no teorema de Riemann.

Denote por $\Omega_{\mathcal{X}}$ o conjunto dos diferenciais associados à curva \mathcal{X} . Assim como foi definido o espaço de funções $L(G)$ de um divisor G , define-se também o espaço vetorial de diferenciais denotado por $\Omega(G)$ e dado por

$$\Omega(G) = \{\omega \in \Omega_{\mathcal{X}} \mid (\omega) - G \succ 0\} \cup \{0\}.$$

A dimensão do espaço $\Omega(G)$ é chamada de *índice de especialidade* de G e denotada por $i(G)$.

Como afirmado acima, se (O, \mathcal{P}) é um ponto fechado de \mathcal{X} de grau d e t é um parâmetro local neste ponto, então existe uma função racional f , tal que $\omega = f dt$. Se uma função f apresenta uma expansão em série de Laurent $\sum_{i=\rho}^{\infty} a_i t^i$, em que $a_i \in \mathbb{F}_q$, $\rho = v_{\mathcal{P}}(\omega)$ e $a_{\rho} \neq 0$, define-se o *resíduo de f com relação a \mathcal{P} e t* como sendo

$$\text{Res}_{\mathcal{P},t}(f) = a_{-1}.$$

Observe que, se $v_{\mathcal{P}}(f) \geq 0$, então $\text{Res}_{\mathcal{P},t}(f) = 0$. O *resíduo de ω em \mathcal{P}* , denotado por $\text{Res}_{\mathcal{P}}(\omega)$, é definido, então, como sendo

$$\text{Res}_{\mathcal{P}}(\omega) = \text{Res}_{\mathcal{P},t}(f).$$

Este resultado é independente da escolha do parâmetro local t .

Dado um diferencial $\omega \in \Omega_{\mathcal{X}}$, o *teorema do resíduo* estabelece que

$$\sum_{\mathcal{P} \in \mathcal{P}_{\mathcal{X}}} \text{Res}_{\mathcal{P}}(\omega) = 0. \tag{3.6}$$

3.3.4 Teorema de Riemann-Roch

Definido o espaço vetorial de diferenciais, pode-se, então, retornar à análise da dimensão $l(G)$ do espaço de funções $L(G)$.

O chamado *teorema de Riemann* afirma que existe um inteiro não negativo r , tal que, para todo divisor G de uma curva \mathcal{X} ,

$$l(G) \geq \deg(G) + 1 - r,$$

sendo o valor mínimo de r denominado *gênero* de \mathcal{X} e denotado por g . Se \mathcal{X} é uma curva não singular (todos os seus pontos são não singulares) e m é seu grau, então seu gênero é dado por

$$g = \frac{(m-1)(m-2)}{2}. \quad (3.7)$$

O problema da determinação da dimensão $l(G)$ é resolvido pelo teorema que segue.

Teorema 21 (Riemann-Roch) *Para um divisor G de uma curva de gênero g , tem-se que*

$$l(G) = \deg(G) + 1 - g + i(G).$$

Além disso, o índice de especialidade $i(G)$ (dimensão do espaço $\Omega(G)$) é dado por

$$i(G) = l(K - G)$$

para todos os divisores G e divisores canônicos K .

Uma consequência do teorema de Riemann-Roch é que, para qualquer divisor G , com $\deg(G) > 2g - 2$, tem-se que

$$l(G) = \deg(G) + 1 - g. \quad (3.8)$$

3.3.5 Lacunas e anti-lacunas

Considere um ponto racional Q de uma curva algébrica \mathcal{X} de gênero g . Sendo m um inteiro não negativo, considere divisores do tipo mQ (divisores de um único ponto).

Definição 22 (Lacuna) *Um inteiro não negativo m é dito ser uma lacuna ("gap") de um ponto Q de uma curva \mathcal{X} , se $l(mQ) = l((m-1)Q)$.*

Segundo o teorema de Riemann-Roch, para $m > 2g - 2$, tem-se que $l(mQ) = m + 1 - g$. Portanto, valores de $m > 2g - 1$ não constituem lacunas. Para valores de m até $2g - 1$, observe que

$$1 = l(0) \leq l(Q) \leq \dots \leq l((2g-1)Q) = g.$$

Ou seja, existem g valores diferentes de $l(iQ)$, para $0 \leq i \leq 2g - 1$. Conclui-se que o número total de lacunas de um ponto Q de uma curva \mathcal{X} é igual ao gênero g de \mathcal{X} .

Definição 23 (Anti-lacuna) Um inteiro positivo m é dito ser uma anti-lacuna ("non-gap") de um ponto Q de uma curva \mathcal{X} , se $l(mQ) \neq l((m-1)Q)$, ou seja, se e só se existir uma função racional $f \in L(mQ)$, tal que $v_Q(f) = -m$ (possui pólos de ordem m em Q).

Se m_i é uma anti-lacuna e $m_{i-1} < m_i$, então

$$0 = m_0 < m_1 < \cdots < m_{g-1} < m_g = 2g$$

e $m_i = i + g$, para $i \geq g$. Também, se m_1 e m_2 são anti-lacunas de Q , então $m_1 + m_2$ também é uma anti-lacuna de Q . Portanto, as anti-lacunas de um ponto Q formam um semi-grupo na adição.

O próximo capítulo utiliza, por fim, toda a teoria apresentada até então para definir os CGA's, os chamados códigos de geometria algébrica, ou algébrico-geométricos.

Capítulo 4

Códigos de Geometria Algébrica

Os capítulos precedentes trataram dos conceitos relativos à codificação para controle de erros e à geometria algébrica, subsídios necessários para a definição e análise dos CGA's e de seus esquemas de decodificação.

O presente capítulo tem o objetivo de definir os CGA's segundo a abordagem traçada no capítulo 2. Em seguida, ele descreve a classe particular dos CGA's construídos sobre as curvas de Hermite. Serão os, assim chamados, códigos de Hermite que servirão de objeto para a análise dos algoritmos de decodificação no capítulo seguinte. Por fim, este capítulo apresenta um novo limite inferior para bons códigos obtido pelos CGA's, melhor que o limite de Gilbert-Varshamov descrito no capítulo 2 (equação 2.5).

São diversas as referências que apresentam as definições descritas aqui. Algumas referências importantes são van Lint [13], [26], Blake et al [2], Justesen et al [10], Stichtenoth [23], [24] e Pretzel [15].

4.1 Definição dos CGA's

No capítulo 2, foram apresentados dois enfoques diferentes na construção de códigos:

1. No primeiro enfoque, relativo aos códigos de Reed-Solomon (equação 2.8), chamado aqui de *construção por funções*, o código consiste na avaliação de funções racionais em um conjunto de pontos distintos;
2. No segundo enfoque, relativo aos códigos de Goppa (equação 2.13), chamado aqui de *construção por diferenciais*, o código consiste no resíduo de diferenciais em um

conjunto de pontos distintos.

Na presente seção, estes enfoques serão utilizados na construção de CGA's, sendo o conjunto dos pontos referidos acima obtido de uma curva algébrica.

4.1.1 Construção por funções

Considere \mathcal{X} uma curva projetiva não singular de gênero g , P_1, \dots, P_n pontos racionais de \mathcal{X} e $D = P_1 + \dots + P_n$ um divisor desta curva. Considere G um divisor de \mathcal{X} cujo suporte seja disjunto de D , e suponha que

$$2g - 2 < \deg(G) < n.$$

Definição 24 (CGA pela avaliação de funções racionais) *Define-se o código de geometria algébrica denotado por $C(D, G)$, sobre o corpo finito \mathbb{F}_q , como sendo o mapeamento linear $\alpha : L(G) \rightarrow \mathbb{F}_q^n$ dado por*

$$\alpha(f) = (f(P_1), \dots, f(P_n)), \quad (4.1)$$

em que

$$L(G) = \{f \in \mathbb{F}_q(\mathcal{X}) \mid (f) + G \succ 0\} \cup \{0\}$$

é o espaço de funções gerado pelo divisor G , $\mathbb{F}_q(\mathcal{X})$ é o corpo de funções da curva \mathcal{X} ,

$$(f) = \sum_{P_i} v_{P_i}(f) P_i$$

é o divisor principal da função f e $v_{P_i}(f)$ é a ordem da função f no ponto $P_i \in \mathcal{X}$.

Observe que o núcleo do mapeamento da equação 4.1 é o conjunto de funções

$$\{f \in L(G) \mid v_{P_i}(f) > 0, i = 1, \dots, n\},$$

que constitui, na verdade, o espaço de funções $L(G - D)$. Então, a dimensão k do código $C(D, G)$ é dada por

$$k = \dim[L(G)] - \dim[L(G - D)].$$

Observe que, sendo $\deg(G) < n$ por hipótese, então $\deg[(G - D)] < 0$. Com isso, para qualquer função $f \in \mathbb{F}_q(\mathcal{X})$, tem-se que $\deg[(f) + G - D] < 0$. Isto implica que

$\dim [L(G - D)] = 0$. Considerando também o teorema de Riemann-Roch apresentado no capítulo 3 (equação 3.8), tem-se que

$$k = \deg(G) - g + 1. \quad (4.2)$$

A distância mínima d do código $C(D, G)$ da equação 4.1 satisfaz a desigualdade

$$d \geq n - \deg(G),$$

uma vez que qualquer função $f \in L(G)$ possui no máximo $\deg(G)$ zeros, ou seja, o peso de qualquer palavra código $\alpha(f)$ não é menor que $n - \deg(G)$. Do limite de Singleton apresentado no capítulo 2 (equação 2.3), tem-se que

$$\begin{aligned} d &\leq n - k + 1 \\ &= n - \deg(G) + g \Rightarrow \\ n - \deg(G) &\leq d \leq n - \deg(G) + g. \end{aligned} \quad (4.3)$$

4.1.2 Construção por diferenciais

Considere $\mathcal{X}, P_1, \dots, P_n, D$ e G da mesma forma da seção anterior (os pontos racionais P_1, \dots, P_n equivalem a lugares de grau 1 da curva \mathcal{X}).

Definição 25 (CGA pelos resíduos de diferenciais) *Define-se o código de geometria algébrica denotado por $C^*(D, G)$, sobre o corpo finito \mathbb{F}_q (corpo algebricamente fechado), como sendo o mapeamento linear $\alpha^* : \Omega(G - D) \rightarrow \mathbb{F}_q^n$ dado por*

$$\alpha^*(\omega) = (\text{Res}_{P_1}(\omega), \dots, \text{Res}_{P_n}(\omega)), \quad (4.4)$$

em que

$$\Omega(G - D) = \{\omega \in \Omega_{\mathcal{X}} \mid (\omega) \succ G - D\} \cup \{0\}$$

é o espaço de diferenciais gerado pelo divisor $G - D$, $\Omega_{\mathcal{X}}$ é o conjunto de diferenciais associados à curva \mathcal{X} , $\text{Res}_{P_i}(\omega)$ é o resíduo de ω no ponto P_i ,

$$(\omega) = \sum_{P_i} v_{P_i}(\omega) P_i$$

é o divisor do diferencial $\omega = f dt$ e $v_{P_i}(\omega) = v_{P_i}(\omega)$ é sua ordem no ponto $P_i \in \mathcal{X}$.

O núcleo do mapeamento α^* da equação 4.4 é o espaço de diferenciais $\Omega(G)$. Com isso, a *dimensão* k^* do código $C^*(D, G)$ é dada por

$$k^* = \dim[\Omega(G - D)] - \dim[\Omega(G)] = i(G - D) - i(G).$$

Considerando a hipótese inicial $\deg(G) > 2g - 2$, tem-se que $i(G) = 0$. Considerando o teorema de Riemann-Roch (equação 3.8), tem-se que a dimensão do código $C^*(D, G)$ é dada por¹

$$k^* = n - \deg(G) + g - 1. \quad (4.5)$$

A *distância mínima* d^* do código $C^*(D, G)$ da equação 4.4 satisfaz a desigualdade²

$$d^* \geq \deg(G) - 2g + 2.$$

Do limite de Singleton, tem-se que

$$\begin{aligned} d^* &\leq n - k^* + 1 \\ &= \deg(G) - g + 2 \Rightarrow \\ \deg(G) - 2g + 2 &\leq d^* \leq \deg(G) - g + 2. \end{aligned} \quad (4.6)$$

Das equações 4.2 e 4.5, observe que $k + k^* = n$. Considere agora $f \in L(G)$ e $\omega \in \Omega(G - D)$. Das definições dos códigos $C(D, G)$ e $C^*(D, G)$, observa-se que o diferencial $f\omega$ não possui pólos, exceto possivelmente nos pontos P_1, \dots, P_n . O resíduo de $f\omega$ no ponto P_i é igual a $f(P_i) \text{Res}_{P_i}(\omega)$. Do teorema de resíduos (equação 3.6), tem-se que

$$\sum_{i=1}^n f(P_i) \text{Res}_{P_i}(\omega) = 0,$$

que é o produto interno de duas palavras código $\alpha(f)$ e $\alpha^*(\omega)$. Conclui-se, portanto, que $C(D, G)$ e $C^*(D, G)$ são códigos duais.

Além disso, mostra-se [24, p. 48] que existe um diferencial ω com pólos simples e resíduo 1 nos pontos P_1, \dots, P_n (a determinação de diferenciais requer subsídios teóricos que divergem dos objetivos deste trabalho), tal que

$$C^*(D, G) = C(D, (\omega) + D - G), \quad (4.7)$$

¹Uma prova detalhada deste resultado pode ser obtida em Stichtenoth [24, pp. 45-46].

²A prova deste resultado pode ser vista em Stichtenoth [24, pp. 45-46].

em que (ω) é o divisor de ω . Isto implica que a construção de resíduos fornece a mesma classe de códigos da construção de funções. Apesar dos códigos de Hermite descritos na seção seguinte utilizarem apenas a construção por funções, as duas abordagens são necessárias no estudo dos algoritmos de decodificação.

4.2 Códigos de Hermite

Os códigos de Hermite constituem hoje uma das mais exploradas classes de CGA's, em vista dos excelentes parâmetros que apresenta. Esta seção descreve esta classe de códigos, suas propriedades e analisa o caso em que o comprimento do código é r^3 (comprimento máximo).

4.2.1 Curvas de Hermite

Considere o corpo finito \mathbb{F}_q de $q = r^2$ elementos, em que q é um potência de algum inteiro primo. Considere o corpo de funções $\mathbb{F}_q(\mathcal{X})$ da curva de Hermite \mathcal{X} dada pela equação

$$\mathcal{X} : u^{r+1} + v^{r+1} + 1 = 0. \quad (4.8)$$

Considere $x, y \in \mathbb{F}_q(\mathcal{X})$, em que

$$x = \frac{b}{v - bu}, \quad y = ux - a \text{ e } a^r + a = b^{r+1} = -1.$$

É fácil verificar que a curva de Hermite \mathcal{X} (equação 4.8) pode ser escrita numa forma mais conveniente como sendo

$$\mathcal{X} : y^r + y = x^{r+1}. \quad (4.9)$$

A versão projetiva em \mathbb{P}^2 , sobre \mathbb{F}_q , da curva de Hermite é dada pela equação (forma homogênea da equação 4.8)

$$\mathcal{X} : u^{r+1} + v^{r+1} + w^{r+1} = 0. \quad (4.10)$$

Com relação ao número de pontos racionais desta curva, considere inicialmente o caso em que uma das coordenadas na equação 4.10 é nula, por exemplo w . Sem perda de generalidade, pode-se fazer $v = 1$, obtendo-se $u^{r+1} + 1 = 0$, que apresenta $r + 1$ soluções

em \mathbb{F}_q . Conclui-se, então, que a curva \mathcal{X} apresenta $3(r+1)$ pontos racionais quando $uvw = 0$. Para o caso em que $uvw \neq 0$, considere $w = 1$ e v igual a qualquer elemento não nulo de \mathbb{F}_q , tal que $v^{r+1} \neq 1$. Para cada valor diferente de v , existem $r+1$ soluções para u na equação 4.10. Neste caso há, portanto, $(r-2)(r+1)^2$ pontos racionais em \mathcal{X} . Totalizando, tem-se que a curva de Hermite \mathcal{X} apresenta sempre

$$3(r+1) + (r-2)(r+1)^2 = 1 + r^3$$

pontos racionais, que são:

1. O ponto no infinito $Q = (a, b, 0)$, com $a, b \in \mathbb{F}_q$, em que $a^{r+1} + b^{r+1} = 0$ (equação 4.10);
2. Os r^3 pontos (a, b) , com $a, b \in \mathbb{F}_q$, em que $b^r + b = a^{r+1}$ (equação 4.9).

A curva de Hermite \mathcal{X} é não singular, portanto irredutível. Sendo o grau da equação 4.9 igual a $r+1$, tem-se que o gênero g (equação 3.7) de \mathcal{X} é dado por

$$g = \frac{q-r}{2}. \quad (4.11)$$

Considere um divisor $G = mQ$, para algum inteiro $m \geq 0$. Mostra-se [23] que o conjunto

$$B_m = \{x^i y^j \mid 0 \leq i, 0 \leq j \leq r-1, ir + j(r+1) \leq m\} \quad (4.12)$$

constitui uma base para o espaço vetorial de funções $L(G)$. A partir dos monômios desta base, pode-se construir a matriz geradora de um código de Hermite, dado um parâmetro m .

4.2.2 Código de Hermite

Considere os dois divisores $G = mQ$ e $D = P_1 + \dots + P_n$, em que P_1, \dots, P_n são os pontos racionais da curva \mathcal{X} (equação 4.9) e $n \leq r^3$.

Definição 26 (Código de Hermite) O código de Hermite $C(D, G)$, denotado por C_m , é dado por

$$C_m = \{(f(P_1), \dots, f(P_n)) \mid f \in L(G)\}. \quad (4.13)$$

A construção do código da equação 4.13 é óbvia, considerando a base B_m fornecida pela equação 4.12 para o espaço de funções $L(G)$.

Considere a função $z \in \mathbb{F}_q[x, y]$ dada por

$$z = x^q - x,$$

cujo divisor principal (z) é dado por

$$(z) = D - r^3Q.$$

Considere o diferencial $\omega = \frac{dz}{z}$, cujo resíduo é igual a 1 em todos os pontos de $D = P_1 + \dots + P_n$ (como P_i é um zero simples de z , então $\frac{dz}{z}$ possui um pólo simples em P_i com resíduo 1). Tem-se que [23]

$$\begin{aligned} (\omega) &= (dz) - (z) \\ &= (-dx) - (z) \\ &= (r(r-1) - 2)Q - D + r^3Q \\ &= (r^3 + r^2 - r - 2)Q - D. \end{aligned}$$

Utilizando-se a equação 4.7 e considerando um divisor $G^* = (\omega) + D - G$, em que $G = mQ$, obtém-se que

$$\begin{aligned} G^* &= (r^3 + r^2 - r - 2)Q - D + D - mQ \\ &= (r^3 + r^2 - r - 2 - m)Q \Rightarrow \\ C^*(D, mQ) &= C(D, (r^3 + r^2 - r - 2 - m)Q). \end{aligned}$$

Portanto, conclui-se que os códigos de Hermite C_m e $C_{r^3+r^2-r-2-m}$ são duais entre si, uma vez que equivalem a $C(D, mQ)$ e $C^*(D, mQ)$, respectivamente. Particularmente, se r é um número par e $m = \frac{1}{2}(r^3 + r^2 - r - 2)$, então o código C_m é auto-dual.

O fato do dual de um código de Hermite C_m poder ser construído também pela avaliação de funções em pontos da curva, sem envolver os conceitos de diferenciais e resíduos de diferenciais, implica simplificações importantes, que justificam também a restrição do escopo de análise desta dissertação a estes códigos.

Caso em que $n = r^3$

Considerando o comprimento máximo do código de Hermite C_m , em que $n = r^3$, serão descritas agora a dimensão deste código, sua matriz geradora, além de outras propriedades [23].

É fácil verificar que a dimensão k do código C_m é $k = 0$, para $m < 0$, e $k = n = r^3$, para $m > r^3 + r^2 - r - 2$. Considerando o parâmetro m no intervalo

$$0 \leq m \leq r^3 + r^2 - r - 2,$$

tem-se que a dimensão k do código C_m é dada por [23]

$$k = \begin{cases} |\mathcal{B}_m|, & m \leq r^2 - r - 2, \\ m + 1 - \frac{(r^2 - r)}{2}, & r^2 - r - 2 < m < r^3, \\ r^3 - |\mathcal{B}_{r^3 + r^2 - r - 2 - m}|, & m \geq r^3, \end{cases} \quad (4.14)$$

sendo $|\mathcal{B}_m|$ a cardinalidade da base \mathcal{B}_m (equação 4.12) para o espaço $L(mQ)$.

A matriz geradora M_m de um código de Hermite C_m , para $0 \leq m < r^3$, é uma matriz $|\mathcal{B}_m| \times r^3$ dada por

$$M_m = [a^i b^j]_{|\mathcal{B}_m| \times r^3}, \quad (4.15)$$

em que:

- $a, b \in \mathbb{F}_q, b^r + b = a^{r+1}$;
- $i \geq 0, 0 \leq j \leq r - 1$;
- e $ir + j(r + 1) \leq m$.

Tomando o parâmetro m no intervalo

$$r^2 - r - 2 < m < r^3 + r^2 - r - 2,$$

tem-se que a matriz $M_{r^3 + r^2 - r - 2 - m}$ consiste também na matriz de paridade do código C_m .

Considere agora $m = ir + j(r + 1) < r^3$, com $i \geq 0$ e $0 \leq j \leq r - 1$. Se $j = 0$ (ou seja, $m \equiv 0 \pmod{r}$) ou $m \leq r^3 - r^2$, então a distância mínima d do código C_m é dada por

$$d = r^3 - m. \quad (4.16)$$

Em suma, as informações contidas nesta seção permitem construir computacionalmente o código de Hermite C_m definido pela equação 4.13 (veja algoritmo 27), de parâmetro m , comprimento $n = r^3$ e dimensão e distância mínima dados pelas equações 4.14 e 4.16, respectivamente.

Algoritmo 27 (Construção do código de Hermite C_m)*Entradas:*

- Corpo \mathbb{F}_q , em que $q = r^2$ é uma potência de algum inteiro primo;
- Curva $\mathcal{X} : y^r + y = x^{r+1}$;
- Parâmetro $m \leq r^3 + r^2 - r - 2$.

Saídas:

- Divisor D dos pontos da curva \mathcal{X} ;
- Gênero g de \mathcal{X} ;
- Base B_m para o espaço $L(mQ)$;
- Parâmetros n, k, d e matriz M_m do código C_m .

Inicialização:

- $a, b \in \mathbb{F}_q$;
- $g = \frac{q-1}{2}$;
- $n = r^3$;
- $d = r^3 - m$.

<<< Passo 1 : Determinação do divisor D >>>FOR $a = 0$ TO $q - 1$ FOR $b = 0$ TO $q - 1$ IF $b^r + b == a^{r+1}$ THEN $D \leftarrow D + (a, b)$ <<< Passo 2 : Determinação da base B_m e da dimensão k >>> $i = 0, j = 0$ e $k = 0$ WHILE $ir + j(r+1) \leq m$ AND $j \leq r - 1$ WHILE $ir + j(r+1) \leq m$ AND $i < q - 1$ $x^i y^j \rightarrow B_m$

```

    k ← k + 1 e i ← i + 1
    i = 0 e j ← j + 1
<<< Passo 3 : Determinação da matriz geradora M_m >>>
FOR i = 0 TO k
    FOR j = 0 TO n
        (a, b) = D[j] e f(x, y) = B_m[i]
        M_m[i][j] = f(a, b)

```

4.2.3 Exemplo de código de Hermite

Observe um exemplo simples de um código de Hermite C_{18} , de comprimento $n = 27$, em \mathbb{F}_9 .

O corpo \mathbb{F}_9 consiste no conjunto $\{1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6, \alpha^7, 0\}$ associado às operações “+” e “×” descritas na tabela 4.1.

Sendo $q = 9 \Rightarrow r = 3$, tem-se que a curva de Hermite correspondente (veja equação 4.9) é dada por

$$\mathcal{X} : y^3 + y = x^4.$$

O gênero desta curva é $g = 3$ (equação 4.11). Os pontos racionais de \mathcal{X} são o ponto Q no infinito e os $r^3 = 27$ pontos descritos na tabela 4.2.

O divisor D , cujo suporte são os 27 pontos racionais da curva \mathcal{X} (tabela 4.2), é

$$\begin{aligned}
 D = & (1, \alpha^4) + (1, \alpha^5) + (1, \alpha^7) + (\alpha, 1) + (\alpha, \alpha) + (\alpha, \alpha^3) + (\alpha^2, \alpha^4) + \\
 & (\alpha^2, \alpha^5) + (\alpha^2, \alpha^7) + (\alpha^3, 1) + (\alpha^3, \alpha) + (\alpha^3, \alpha^3) + (\alpha^4, \alpha^4) + (\alpha^4, \alpha^5) + \\
 & (\alpha^4, \alpha^7) + (\alpha^5, 1) + (\alpha^5, \alpha) + (\alpha^5, \alpha^3) + (\alpha^6, \alpha^4) + (\alpha^6, \alpha^5) + (\alpha^6, \alpha^7) + \\
 & (\alpha^7, 1) + (\alpha^7, \alpha) + (\alpha^7, \alpha^3) + (0, \alpha^2) + (0, \alpha^6) + (0, 0).
 \end{aligned}$$

A base para o espaço $L(18Q)$ (veja equação 4.12) é o conjunto dos monômios $x^i y^j$, em que

$$3i + 4j \leq 18 \text{ e } j \leq 2.$$

Tem-se, então, que

$$\mathcal{B}_{18} = \left\{ \underset{1}{1}, \underset{2}{x}, \underset{3}{x^2}, \underset{4}{x^3}, \underset{5}{x^4}, \underset{6}{x^5}, \underset{7}{x^6}, \underset{8}{y}, \underset{9}{xy}, \underset{10}{x^2y}, \underset{11}{x^3y}, \underset{12}{x^4y}, \underset{13}{y^2}, \underset{14}{xy^2}, \underset{15}{x^2y^2}, \underset{16}{x^3y^2} \right\}.$$

+	0	1	α	α^2	α^3	α^4	α^5	α^6	α^7
0	0	1	α	α^2	α^3	α^4	α^5	α^6	α^7
1	1	α^4	α^7	α^3	α^5	0	α^2	α	α^6
α	α	α^7	α^5	1	α^4	α^6	0	α^3	α^2
α^2	α^2	α^3	1	α^6	α	α^5	α^7	0	α^4
α^3	α^3	α^5	α^4	α	α^7	α^2	α^6	1	0
α^4	α^4	0	α^6	α^5	α^2	1	α^3	α^7	α
α^5	α^5	α^2	0	α^7	α^6	α^3	α	α^4	1
α^6	α^6	α	α^3	0	1	α^7	α^4	α^2	α^5
α^7	α^7	α^6	α^2	α^4	0	α	1	α^5	α^3

×	0	1	α	α^2	α^3	α^4	α^5	α^6	α^7
0	0	0	0	0	0	0	0	0	0
1	0	1	α	α^2	α^3	α^4	α^5	α^6	α^7
α	0	α	α^2	α^3	α^4	α^5	α^6	α^7	1
α^2	0	α^2	α^3	α^4	α^5	α^6	α^7	1	α
α^3	0	α^3	α^4	α^5	α^6	α^7	1	α	α^2
α^4	0	α^4	α^5	α^6	α^7	1	α	α^2	α^3
α^5	0	α^5	α^6	α^7	1	α	α^2	α^3	α^4
α^6	0	α^6	α^7	1	α	α^2	α^3	α^4	α^5
α^7	0	α^7	1	α	α^2	α^3	α^4	α^5	α^6

Tabela 4.1: Operações “+” e “×” relacionadas ao conjunto $\{1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6, \alpha^7, 0\}$, que constituem o corpo \mathbb{F}_9 .

Os parâmetros deste código C_{18} são $n = r^3 = 27$, $k = 16$ (equação 4.14) e $d = 9$

Nº	Ponto	Nº	Ponto	Nº	Ponto
1	$(1, \alpha^4)$	10	$(\alpha^3, 1)$	19	(α^6, α^4)
2	$(1, \alpha^5)$	11	(α^3, α)	20	(α^6, α^5)
3	$(1, \alpha^7)$	12	(α^3, α^3)	21	(α^6, α^7)
4	$(\alpha, 1)$	13	(α^4, α^4)	22	$(\alpha^7, 1)$
5	(α, α)	14	(α^4, α^5)	23	(α^7, α)
6	(α, α^3)	15	(α^4, α^7)	24	(α^7, α^3)
7	(α^2, α^4)	16	$(\alpha^5, 1)$	25	$(0, \alpha^2)$
8	(α^2, α^5)	17	(α^5, α)	26	$(0, \alpha^6)$
9	(α^2, α^7)	18	(α^5, α^3)	27	$(0, 0)$

Tabela 4.2: Pontos racionais da curva de Hermite $y^3 + y = x^4$ em \mathbb{F}_9 .

(equação 4.16). Sua matriz geradora (equação 4.15) é

$$M_{13} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & \dots & 23 & 24 & 25 & 26 & 27 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ \vdots \\ 13 \\ 14 \\ 15 \\ 16 \end{matrix} & \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \alpha & \alpha & \dots & \alpha^7 & \alpha^7 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^2 & \alpha^2 & \dots & \alpha^6 & \alpha^6 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^3 & \alpha^3 & \dots & \alpha^5 & \alpha^5 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^2 & \alpha^6 & 1 & \alpha^2 & \dots & \alpha^2 & \alpha^6 & \alpha^4 & \alpha^4 & 0 \\ 1 & \alpha^2 & \alpha^6 & \alpha & \alpha^2 & \dots & \alpha & \alpha^5 & 0 & 0 & 0 \\ 1 & \alpha^2 & \alpha^6 & \alpha^2 & \alpha^4 & \dots & 1 & \alpha^4 & 0 & 0 & 0 \\ 1 & \alpha^2 & \alpha^6 & \alpha^3 & \alpha^5 & \dots & \alpha^7 & \alpha^3 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

O espaço de funções ortogonal a este código C_{13} é $L(13Q)$. A base para este espaço é (equação 4.12)

$$\mathcal{B}_{13} = \left\{ 1, x, x^2, x^3, x^4, y, xy, x^2y, x^3y, y^2, xy^2 \right\}.$$

Portanto, a matriz de paridade do código C_{18} é (equação 4.15)

$$H_{18} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & \dots & 23 & 24 & 25 & 26 & 27 \end{matrix} \\ \begin{matrix} 1 \\ 1 \\ 1 \\ 1 \\ \vdots \\ \alpha^4 \\ \alpha^4 \\ 1 \\ 1 \end{matrix} & \left[\begin{array}{cccccccccccc} 1 & 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \alpha & \alpha & \dots & \alpha^7 & \alpha^7 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^2 & \alpha^2 & \dots & \alpha^6 & \alpha^6 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^3 & \alpha^3 & \dots & \alpha^5 & \alpha^5 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^2 & \alpha^3 & \dots & \alpha^7 & \alpha & 0 & 0 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^3 & \alpha^4 & \dots & \alpha^6 & 1 & 0 & 0 & 0 \\ 1 & \alpha^2 & \alpha^6 & 1 & \alpha^2 & \dots & \alpha^2 & \alpha^6 & \alpha^4 & \alpha^4 & 0 \\ 1 & \alpha^2 & \alpha^6 & \alpha & \alpha^2 & \dots & \alpha & \alpha^5 & 0 & 0 & 0 \end{array} \right] & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \end{matrix} \end{matrix}$$

No capítulo seguinte, quando da descrição dos algoritmos de decodificação para os CGA's, serão apresentados exemplos da decodificação deste código de Hermite C_{18} .

4.3 Novo limite dos códigos

Um dos principais motivos que despertaram grande interesse nos CGA's foi o fato destes constituírem bons códigos com parâmetros que ultrapassam o limite de Gilbert-Varshamov (equação 2.5) em uma determinada faixa, para um corpo de dimensão mínima.

Reconsidere a notação utilizada na subseção 2.1.5:

- $A(n, d)$ é o valor máximo de $M = q^k = |C|$ para o qual o código $C = (n, k, d)$ existe, sendo n seu comprimento, k sua dimensão e d sua distância mínima;
- $R = \frac{k}{n}$ é a taxa de informação do código e $\delta = \frac{d}{n}$ é sua distância mínima relativa;
- $R(\delta) = \lim_{n \rightarrow \infty} \frac{\log_q A(n, \delta n)}{n}$ é a taxa de informação assintótica para códigos com distância mínima relativa δ .

O limite de Gilbert-Varshamov descrito naquela seção é dado por

$$R(\delta) \geq 1 - H_q(\delta), \tag{4.17}$$

sendo a entropia $H_q(x)$ dada por

$$H_q(x) = \begin{cases} 0 & , x = 0; \\ x \log_q(q-1) - x \log_q x - (1-x) \log_q(1-x) & , 0 < x < \frac{q-1}{q}. \end{cases}$$

Considere agora o código $C(D, G)$ definido pela equação 4.1, em que a curva \mathcal{X} de gênero g possui os $n+1$ pontos racionais P_1, \dots, P_n e Q e o divisor $G = mQ$, com $2g - 2 < \deg(G) = m < n$. Defina $\gamma(\mathcal{X}) = \frac{g}{n}$. Foi mostrado por Tsfasman, Vlăduț e Zink [25] que existe uma seqüência de curvas \mathcal{X}_i 's, cujos CGA's correspondentes apresentam parâmetros que ultrapassam o limite da equação 4.17. De fato, eles provaram o teorema que segue.

Teorema 28 *Considere q uma potência de um inteiro primo e o quadrado de algum inteiro. Existe uma seqüência de curvas \mathcal{X}_i 's sobre \mathbb{F}_q , tal que \mathcal{X}_i possui $n_i + 1$ pontos racionais e gênero g_i , em que $n \rightarrow \infty$ e $\gamma(\mathcal{X}_i) \rightarrow \frac{1}{\sqrt{q}-1}$ quando $i \rightarrow \infty$.*

Das equações 4.2 e 4.3, tem-se que um código $C_i = C(D, m_i Q)$ definido sobre uma curva \mathcal{X}_i possui taxa de informação

$$R_i = \frac{m_i - g_i + 1}{n_i}$$

e distância mínima

$$d_i \geq n_i - m_i \Rightarrow \delta_i \geq 1 - \frac{m_i}{n_i}.$$

Daí, obtém-se que

$$\begin{aligned} R_i + \delta_i &\geq \frac{m_i - g_i + 1}{n_i} + 1 - \frac{m_i}{n_i} \\ &= 1 - \gamma(\mathcal{X}_i) + \frac{1}{n_i}. \end{aligned}$$

Fazendo n tender para infinito e eliminando o termo que tende a zero, obtém-se o chamado *limite de Tsfasman-Vlăduț-Zink* dado por

$$R(\delta) \geq 1 - \frac{1}{\sqrt{q}-1} - \delta. \tag{4.18}$$

Deduz-se facilmente que este limitante inferior é melhor que o limitante da equação 4.17 em uma determinada faixa de valores, para $q \geq 43$. Como q deve ser o quadrado de algum inteiro, então o limite de Tsfasman-Vlăduț-Zink é melhor que o limite de

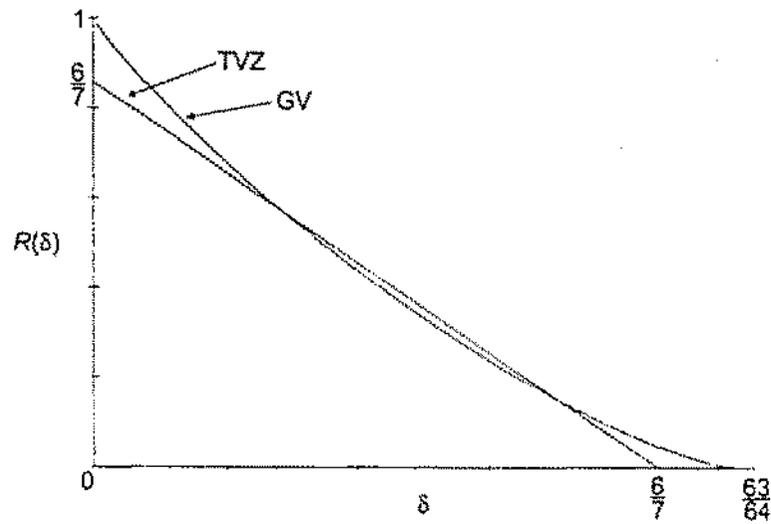


Figura 4.1: Comparação do limite de Gilbert-Varshamov (curva GV) com o limite de Tsfasman-Vlăduț-Zink (curva TVZ) para $q = 64$.

Gilbert-Varshamov para $q \geq 49$. A figura 4.1 compara os limites das equações 4.17 e 4.18 para $q = 64$.

O capítulo seguinte descreverá as principais abordagens utilizadas na decodificação dos CGA's, em particular dos códigos de Hermite, apresentando os principais algoritmos e suas características.

Capítulo 5

Decodificação dos Códigos de Geometria Algébrica

Tendo sido apresentados os conceitos relativos à codificação para controle de erros e à geometria algébrica e tendo sido definidos os códigos de Hermite nos capítulos anteriores, pode-se, enfim, tratar dos esquemas de decodificação para estes códigos.

O objetivo deste capítulo é abordar a questão da decodificação dos códigos de Hermite, analisando os esquemas mais importantes desenvolvidos neste curto intervalo de 10 anos, desde a publicação do primeiro algoritmo de decodificação para códigos sobre curvas planas por Justesen et al [10]. Neste capítulo, inicialmente, é descrito o problema da decodificação de códigos para correção de erros. As seções seguintes analisam o desenvolvimento dos esquemas de decodificação para os códigos de Hermite e apresentam suas principais abordagens.

5.1 O problema da decodificação

Considere aqui os conceitos e a notação apresentados no capítulo 2.

Considere C um código em \mathbb{F}_q^n de distância mínima¹ d . Em um sistema de co-

¹A distância mínima considerada neste capítulo é a chamada distância mínima projetada do código, que, no caso dos códigos de Hermite, é dada pela equação 4.16, a saber

$$d = r^3 - m,$$

sendo $r^2 = q$.

municação digital, se $c \in C$ for uma palavra código transmitida através do canal de comunicação e se $v = c + e$ for a palavra recebida correspondente, então o vetor $e \in \mathbb{F}_q^n$ será dito o *vetor erro*, $\{i \mid e_i \neq 0\}$ será o conjunto das *posições dos erros*, os e_i 's serão os *valores dos erros* naquelas posições e $w(e)$ (peso do vetor e) será o *número de erros* ocorridos na palavra recebida. Se $w(e) \leq \frac{d-1}{2}$, então a palavra recebida v pode ser unicamente associada pelo decodificador à palavra código mais próxima c .

Considere ainda que C é um código linear de dimensão k e taxa de informação $R = \frac{k}{n}$. Sua matriz geradora é uma matriz $G_{k \times n}$, tal que $C = \{xG \mid x \in \mathbb{F}_q^k\}$. Portanto, um determinado mapeamento

$$\psi: \mathbb{F}_q^k \rightarrow \mathbb{F}_q^n$$

definido por $\psi(x) = xG$ é dito ser um *codificador* linear, que codifica palavras de comprimento k em palavras código de C de comprimento n .

Um mapeamento

$$\delta: \mathbb{F}_q^n \rightarrow C^*$$

em que $C^* = C \cup C^?$ e $C \cap C^? = \emptyset$, é chamado *decodificador* para o código C , se $\delta(v) = c$, em que $c \in C$, ou se $\delta(v) \in C^?$, para os casos em que o decodificador não encontra uma palavra código correspondente à palavra recebida v .

Definição 29 (Decodificador de menor distância) Um decodificador de menor distância para um código C é um decodificador δ , em que $\delta(v) = c$ é a palavra código mais próxima de v , ou $\delta(v) \in C^?$.

Diz-se que ocorreu *erro de decodificação* quando a palavra decodificada é diferente da palavra código transmitida.

Observe que o código linear C pode também ser definido em função de sua matriz de paridade $H_{(n-k) \times n}$ como sendo $C = \{c \in \mathbb{F}_q^n \mid Hc^T = 0\}$. As linhas h_i , com $i = 1, \dots, n-k$, da matriz H formam, por definição, uma base para o código C^\perp dual de C (ou ortogonal ao espaço vetorial C).

Considere agora $v = c + e$ uma palavra recebida, em que $v, e \in \mathbb{F}_q^n$ e $c \in C$. As

$n - k$ síndromes de \mathbf{v} podem ser definidas como sendo²

$$S_i(\mathbf{v}) = \mathbf{h}_i \mathbf{v}^T, \quad i = 1, \dots, n - k. \quad (5.1)$$

Observe que, como, por definição, $S_i(\mathbf{c}) = \mathbf{h}_i \mathbf{c}^T = 0$, tem-se que $S_i(\mathbf{v}) = \mathbf{h}_i (\mathbf{c} + \mathbf{e})^T = \mathbf{h}_i \mathbf{e}^T$. Conclui-se que se pode obter informações importantes acerca dos erros ocorridos na comunicação (vetor \mathbf{e}) por meio das síndromes da palavra recebida \mathbf{v} . Na prática, a tarefa de decodificação consiste basicamente na determinação do vetor \mathbf{e} ocorrido através do uso destas síndromes, partindo-se da premissa de que o peso de \mathbf{e} não ultrapassa um limite t (*decodificador de distância limitada* definido a seguir), caso em que ocorreria erro de decodificação (decodificador de t erros).

Um esquema importante na decodificação de CGA's consiste em considerar uma extensão da matriz H denotada por $\hat{H}_{n \times n}$, cujas n linhas $\hat{\mathbf{h}}_i$, com $i = 1, \dots, n$, constituem uma base para o espaço \mathbb{F}_q^n e as primeiras $n - k$ linhas coincidem com a matriz de paridade H . As n síndromes obtidas das linhas da matriz \hat{H} determinam unicamente o vetor erro, mas apenas as $n - k$ primeiras são conhecidas. As k síndromes restantes são chamadas *síndromes desconhecidas*. Numa subseção seguinte, será apresentado um procedimento baseado em um esquema denominado *decisão por maioria* ("majority voting"), que determina as síndromes desconhecidas para a classe dos códigos de Hermite (além de outras), permitindo determinar o vetor erro \mathbf{e} , conseqüentemente, a palavra código original.

O conjunto de todas as palavras de \mathbb{F}_q^n que apresentam a mesma síndrome de uma palavra \mathbf{v} é dito uma *classe lateral*. A função síndrome, então, estabelece uma relação de equivalência entre as palavras, sendo cada classe lateral uma classe de equivalência (veja tabela 5.1). O elemento de uma classe lateral de menor peso é dito o *líder da classe lateral*.

Um esquema simples de decodificação de menor distância consiste na procura exaustiva pelo líder da classe lateral correspondente à palavra recebida. Outra possibilidade seria listar e armazenar todos os líderes de classes laterais, de modo a evitar o esforço de procura. No primeiro caso, é necessária a busca entre os q^k elementos da classe

²Há uma abordagem diferente na decodificação, em que as síndromes são definidas como elementos de um anel afim (um anel de funções racionais), ao invés de na forma de um mapeamento de um subespaço linear de funções em um corpo de localização (corpo finito), como descrito acima. Ambas as abordagens serão analisadas nas seções seguintes.

					Esfera de decodificação
	0	c_2	c_3	\dots	c_{q^k}
	v_2	$c_2 + v_2$	$c_3 + v_2$	\dots	$c_{q^k} + v_2$
Classe Lateral	v_3	$c_2 + v_3$	$c_3 + v_3$	\dots	$c_{q^k} + v_3$
	\vdots	\vdots	\vdots	\ddots	\vdots
	$v_{q^{n-k}}$	$c_2 + v_{q^{n-k}}$	$c_3 + v_{q^{n-k}}$	\dots	$c_{q^k} + v_{q^{n-k}}$
	Líderes				

Tabela 5.1: Organização do espaço decodificável em classes laterais de palavras.

lateral da palavra recebida pelo elemento de peso mínimo. No segundo caso, é necessário armazenar q^{n-k} líderes de classes laterais. Observe que ambos os esquemas implicam uma complexidade³ computacional que é função exponencial do comprimento do código, o que inviabiliza sua utilização em aplicações práticas. Os decodificadores de menor distância conhecidos, todos eles possuem complexidade exponencial, não sendo mais abordados neste trabalho.

Os decodificadores conhecidos que apresentam complexidade polinomial (não exponencial) são, em contrapartida, limitados em sua capacidade de correção.

Definição 30 (Decodificador de distância limitada) Um decodificador δ para um código C é dito ser um decodificador de distância limitada corretor de t erros, se $\delta(v) = c$ for a palavra código mais próxima de v e $d(v, c) \leq t$, para todo $v \in \mathbb{F}_q^n$, ou se $\delta(v) \in C^t$.

Sendo C um código de distância mínima d , no caso em que $t = \frac{d-1}{2}$, o decodificador δ é dito decodificar até metade da distância mínima.

Considere agora C um código linear em \mathbb{F}_q^n com uma matriz de paridade H . Suponha uma palavra recebida v associada a um vetor erro e de peso $w(e) \leq \frac{d-1}{2}$, em que o conjunto $\{i \mid e_i \neq 0\}$ das até $\frac{d-1}{2}$ posições dos erros ocorridos é conhecido. Observe que o vetor e é a única solução da equação

$$Hx^T = Hv^T, \tag{5.2}$$

³Neste trabalho, é utilizada a notação $\mathcal{O}(g(n))$ para a complexidade dos algoritmos descritos, em que $g(n)$ é normalmente um polinômio e expressa a ordem de grandeza do número de operações básicas (ou iterações algorítmicas) necessárias para a obtenção do resultado.

em que $x_j = 0$, para todo $j \notin \{i \mid e_i \neq 0\}$.

É óbvio que o vetor erro e é uma solução desta equação. Supondo que x seja uma solução diferente de e , tem-se que $Hx^T = He^T \Rightarrow H(x - e)^T = 0$. Portanto, $x - e$ é um elemento de C e, além disso, tem seu suporte em $\{i \mid e_i \neq 0\}$. Como seu peso é menor ou igual a $\frac{d-1}{2}$, conclui-se que $x - e = 0$, ou seja, $x = e$ é a única solução possível para a equação 5.2.

Desta forma, vê-se que o problema da decodificação fica reduzido à procura pelas posições dos erros. Uma vez conhecido o conjunto $\{i \mid e_i \neq 0\}$ das posições dos erros ocorridos, pode-se determinar o vetor e através da equação 5.2.

A grande parte dos algoritmos de decodificação existentes fornece um polinômio, um vetor, uma função ou um ideal de funções que localiza os erros. O conjunto $\{i \mid e_i \neq 0\}$ das posições dos erros consiste, na prática, no conjunto dos zeros da função ou das funções fornecidas pelos algoritmos.

5.2 Decodificação dos códigos de Hermite

Como já foi afirmado no início deste trabalho, apesar do curto tempo de vida dos CGA's, são diversos e variados os trabalhos já desenvolvidos no que se refere à sua decodificação. A presente dissertação tem por objetivo descrever apenas as principais abordagens, tecendo exemplos de forma a facilitar a compreensão do leitor.

A título de menção, alguns dos algoritmos descritos aqui tiveram sua implementação realizada durante a construção desta dissertação. Estas implementações tiveram por fim auxiliar na compreensão dos algoritmos analisados, além de servirem, ao lado desta dissertação, como referência para futuros trabalhos nesta área.

5.2.1 Breve histórico

O primeiro esquema de decodificação para os códigos de Hermite (na verdade, para CGA's baseados em curvas planas) foi proposto no final da década de 80 por Justesen, Larsen, Jensen, Havemose e Høholdt [10] e consiste numa generalização do algoritmo PGZ (Peterson-Gorenstein-Zierler) [1] para decodificação de códigos BCH. Ele, de modo semelhante ao PGZ, fornece um polinômio localizador de erros em duas variáveis, que possui entre seus zeros as posições dos erros ocorridos na palavra recebida.

Em 1990, Skorobogatov e Vlăduț [22] propuseram uma generalização do algoritmo de Justesen et al para curvas arbitrárias (ao invés de planas apenas). O algoritmo proposto ficou conhecido como *algoritmo básico* e é capaz de corrigir até $\frac{d-g-1}{2}$ erros ocorridos na palavra recebida, em que g é o gênero da curva usada na geração do código e d sua distância mínima projetada. Sua complexidade algorítmica é $\mathcal{O}(d^2n + g^2n) \leq \mathcal{O}(n^3)$, sendo n o comprimento do código.

Ainda neste mesmo artigo, Skorobogatov e Vlăduț apresentaram uma modificação do algoritmo básico, conhecida como *algoritmo modificado*, que corrige até $\frac{d-1}{2} - \sigma$ erros, em que σ é o chamado *defeito de Clifford* [22], que é aproximadamente igual a $\frac{g}{2}$ no caso de curvas planas. O algoritmo modificado tem complexidade $\mathcal{O}(n^4)$.

Uma abordagem diferente na decodificação dos CGA's, que consiste numa generalização do algoritmo de Euclides para solução da *equação chave* [12], [6], foi proposta por Porter [14] em 1992. O algoritmo proposto corrige até $\frac{d-1}{2} - \sigma$ erros. Neste caso, o conjunto dos zeros correspondentes às posições dos erros é obtido de um ideal, ao invés de um polinômio. Mostrou-se que ambos, o algoritmo modificado e o algoritmo de Porter, são, na verdade, equivalentes.

Um dos primeiros trabalhos a proporcionar uma decodificação até metade da distância mínima foi proposto por Feng e Rao [4]. A elegante solução apresentada utiliza um esquema de decisão por maioria para determinar as síndromes desconhecidas da palavra recebida. Como foi afirmado na seção anterior, conhecidas todas as n síndromes, pode-se determinar o vetor erro ocorrido. Este esquema de decisão por maioria de Feng e Rao foi aplicado posteriormente ao algoritmo de Porter por Shen e Tzeng [21].

A complexidade dos algoritmos acima descritos são proibitivas para aplicações práticas, em que necessita-se utilizar códigos com comprimento elevado. No entanto, diversos esquemas rápidos de decodificação, ou seja, com menor complexidade, têm sido propostos.

Em 1990, Sakata [18], [17], apresentou uma generalização em várias variáveis do clássico algoritmo de Berlekamp-Massey [1], que passou a ser conhecida como *algoritmo BMS*. Com base neste algoritmo, diversas implementações rápidas têm sido apresentadas, tais como uma versão do algoritmo modificado por Justesen, Larsen, Jensen e Høholdt [11] e uma versão usando decisão por maioria por Sakata, Justesen, Madelung, Jensen e Høholdt [19], além de outras.

Com o uso de matrizes de blocos de Hankel de códigos sobre curvas planas, Feng,

Wei, Rao e Tzeng [5] obtiveram uma redução da complexidade do esquema de decisão por maioria. A implementação rápida proposta tem, no pior caso, complexidade $\mathcal{O}((r+1)n^2)$, em que $r+1$ é o grau da curva algébrica usada na definição dos códigos.

Com relação à correção de erros e apagamentos, o próprio algoritmo básico de Skorobogatov e Vlăduț [22] a faz. Existem diversos outros trabalhos propostos no sentido da correção de erros e apagamentos, a exemplo do esquema de Sakata, Leonard, Jensen e Høholdt [20], que associa o algoritmo BMS à decisão por maioria e à correção de apagamentos.

Para maiores detalhes sobre a história da decodificação dos CGA's, veja o artigo de Høholdt e Pellikaan [9].

5.2.2 Abordagens na decodificação dos CGA's

Pode-se observar duas abordagens distintas no desenvolvimento dos algoritmos de decodificação para CGA's:

1. Uma primeira abordagem, que é a utilizada no algoritmo básico de Skorobogatov e Vlăduț [22];
2. Uma segunda abordagem, que é a utilizada no algoritmo de Porter [14].

A diferença entre as duas abordagens reside na forma como as síndromes são definidas. Na primeira abordagem, a síndrome é definida como um mapeamento de um subespaço linear de funções em um corpo de localização⁴ (caso da equação 5.1). Na segunda abordagem, a síndrome é definida como um elemento em um anel afim. Esta diferença na definição das síndromes implica duas formulações diferentes na decodificação:

1. A primeira, pela solução de um conjunto de equações lineares sobre um corpo de localização;
2. A segunda, através da solução de uma equação chave em um anel afim.

De modo a ilustrar estas duas abordagens, serão descritos nas seções seguintes o algoritmo básico e o algoritmo de Porter.

⁴Este corpo de localização é, de fato, um corpo finito.

São também esquemas importantes que devem ser observados o de decisão por maioria e o algoritmo BMS, que possibilitaram implementações mais eficientes na decodificação dos CGA's. Estes dois esquemas serão descritos numa subsecção seguinte e ilustrados com a descrição do algoritmo apresentado por Sakata, Justesen, Madelung, Jensen e Høholdt [19], que incorpora tanto o algoritmo BMS quanto o esquema de decisão por maioria de Feng e Rao.

5.3 Primeira abordagem na decodificação

A primeira abordagem utilizada no desenvolvimento de esquemas de decodificação para CGA's é aquela em que as síndromes são definidas como um mapeamento de um subespaço linear de funções em um corpo de localização. O algoritmo básico apresentado por Skorobogatov e Vlăduț [22] é um exemplo típico desta primeira abordagem descrita na subsecção 5.2.2.

5.3.1 Algoritmo básico de Skorobogatov e Vlăduț

Considere \mathcal{X} uma curva algébrica de gênero g . Considere $\mathcal{P}_{\mathcal{X}} = \{P_1, \dots, P_n\}$ um conjunto de pontos racionais (equivalentes a lugares de grau 1) da curva \mathcal{X} sob o corpo algebricamente fechado \mathbb{F}_q e o divisor $D = P_1 + \dots + P_n$, cujo suporte é $\mathcal{P}_{\mathcal{X}}$. Considere $G = aQ$ um divisor de grau $a \geq 0$, em que $Q \notin \mathcal{P}_{\mathcal{X}}$ é um ponto de \mathcal{X} (suporte de G disjunto de $\mathcal{P}_{\mathcal{X}}$). Suponha também que

$$2g - 2 < a \leq n + g - 1.$$

Considere aqui o CGA $C^*(D, G)$ (dual do código $C(D, G)$), que será denotado simplesmente por C , cuja matriz de paridade é

$$H_G = \begin{bmatrix} f_1(P_1) & f_1(P_2) & \cdots & f_1(P_n) \\ f_2(P_1) & f_2(P_2) & \cdots & f_2(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ f_m(P_1) & f_m(P_2) & \cdots & f_m(P_n) \end{bmatrix}, \quad (5.3)$$

em que $\{f_1, \dots, f_m\}$ é uma base para o espaço $L(G)$ ⁵.

⁵Normalmente, os algoritmos de decodificação trabalham sobre CGA's construídos por resíduos de

Considere uma palavra recebida $\mathbf{v} \in \mathbb{F}_q^n$ e as funções f_1, \dots, f_m da base de $L(G)$. As síndromes de \mathbf{v} são definidas (equação 5.1) como sendo

$$S(\mathbf{v}, f_i) = \sum_{j=1}^n v_j f_i(P_j), \quad i = 1, \dots, m, \quad (5.4)$$

caracterizando a primeira abordagem supracitada.

O algoritmo básico proposto por Skorobogatov e Vlăduț [22] permite a correção simultânea de erros e apagamentos⁶. Considere $\mathbf{e} \in \mathbb{F}_q^n$ o vetor de erros ocorridos, em que $w(\mathbf{e}) = t$, e $\mathbf{r} \in \mathbb{F}_q^n$ o vetor de apagamentos, sendo $w(\mathbf{r}) = \tau$. Denote por $\{E_1, \dots, E_t\} \subset \mathcal{P}_X$ e por $\{R_1, \dots, R_\tau\} \subset \mathcal{P}_X$ os conjuntos disjuntos de pontos de \mathcal{P}_X correspondentes às posições dos erros e dos apagamentos, respectivamente. Deve-se observar que, de fato, o algoritmo trata estas posições (pontos de \mathcal{P}_X) como elementos de \mathbb{F}_q .

Além do divisor $G = aQ$, o algoritmo básico (algoritmo 31 a seguir) depende ainda de um divisor auxiliar $F = bQ$, em que $b \leq a$. A capacidade do algoritmo básico de corrigir t erros e τ apagamentos está condicionada a este divisor F , que pode ser determinado pelas inequações [22]

$$l(F) > t + \tau \quad (5.5)$$

e

$$a - b > t + 2g - 2. \quad (5.6)$$

Relativas a este divisor auxiliar F , são consideradas ainda as matrizes

$$H_F = \begin{bmatrix} k_1(P_1) & k_1(P_2) & \cdots & k_1(P_n) \\ k_2(P_1) & k_2(P_2) & \cdots & k_2(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ k_s(P_1) & k_s(P_2) & \cdots & k_s(P_n) \end{bmatrix}, \quad (5.7)$$

diferenciais ($C^*(D, G)$), ao invés de CGA's construídos pela avaliação de funções racionais ($C(D, G)$). Isto, porque, no primeiro caso, a descrição da matriz de paridade é mais simples, feita pela avaliação de funções racionais em pontos da curva.

⁶A única diferença entre erros e apagamentos é que as posições dos apagamentos são previamente conhecidas pelo decodificador, restando ao algoritmo determinar apenas os valores destes apagamentos.

em que $\{k_1, \dots, k_s\}$ é uma base para o espaço de funções $L(F)$, e

$$H_{G-F} = \begin{bmatrix} h_1(P_1) & h_1(P_2) & \cdots & h_1(P_n) \\ h_2(P_1) & h_2(P_2) & \cdots & h_2(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ h_k(P_1) & h_k(P_2) & \cdots & h_k(P_n) \end{bmatrix}, \quad (5.8)$$

sendo $\{h_1, \dots, h_k\}$ uma base para o espaço $L(G - F)$.

Algoritmo 31 (Algoritmo Básico de Decodificação)

Entradas:

- Uma palavra recebida $\mathbf{v} = (v_1, \dots, v_n)$;
- O conjunto $\{R_1, \dots, R_r\}$ das posições de apagamentos;
- As matrizes H_G (equação 5.3), H_F (equação 5.7) e H_{G-F} (equação 5.8).

Saídas:

- O vetor $e + r$ de erros e apagamentos.

<<< Passo 1 >>>

Determine a matriz

$$H_{F-R} = \begin{bmatrix} g_1(P_1) & g_1(P_2) & \cdots & g_1(P_n) \\ g_2(P_1) & g_2(P_2) & \cdots & g_2(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ g_l(P_1) & g_l(P_2) & \cdots & g_l(P_n) \end{bmatrix},$$

em que $\{g_1, \dots, g_l\}$ é uma base para o espaço de funções $L(F - \sum R_i)$. Observe que este espaço consiste nas funções de $L(F)$ que possuem zeros nas posições R_1, \dots, R_r . Como $L(F - \sum R_i) \subseteq L(F)$, então as funções da base $\{g_1, \dots, g_l\}$ podem ser escritas na forma $\sum_j x_j k_j$, sendo $x_j \in \mathbb{F}_q$. Portanto,

$$\sum_j k_j(R_i) x_j = 0.$$

Tem-se, pois, o sistema

$$\begin{bmatrix} k_1(R_1) & k_2(R_1) & \cdots & k_s(R_1) \\ k_1(R_2) & k_2(R_2) & \cdots & k_s(R_2) \\ \vdots & \vdots & \ddots & \vdots \\ k_1(R_r) & k_2(R_r) & \cdots & k_s(R_r) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Do espaço de soluções deste sistema, $l(F - \sum R_i) = \deg(F - \sum R_i) + 1 - g$ soluções linearmente independentes podem ser obtidas. As combinações lineares das funções da base de $L(F)$ correspondentes às soluções obtidas neste sistema determinam as funções da base de $L(F - \sum R_i)$, das quais deriva a matriz H_{F-R} .

<<< Passo 2 >>>

Observe que $g_i h_j \in L(G)$. Determine as lk síndromes $S(v, g_i h_j)$ (equação 5.4), com $i = 1, \dots, l$ e $j = 1, \dots, k$.

<<< Passo 3 >>>

Determine uma solução não trivial (y_1, \dots, y_l) para o sistema

$$\begin{bmatrix} S(v, g_1 h_1) & S(v, g_2 h_1) & \cdots & S(v, g_l h_1) \\ S(v, g_1 h_2) & S(v, g_2 h_2) & \cdots & S(v, g_l h_2) \\ \vdots & \vdots & \ddots & \vdots \\ S(v, g_1 h_k) & S(v, g_2 h_k) & \cdots & S(v, g_l h_k) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (5.9)$$

A condição da equação 5.5 garante que este sistema possui pelo menos uma solução não trivial.

<<< Passo 4 >>>

Dada a solução (y_1, \dots, y_l) obtida no passo 3, determine o conjunto de zeros $\{Q_1, \dots, Q_u\} \subset \mathcal{P}_X$ da equação

$$g_y = y_1 g_1 + \cdots + y_l g_l. \quad (5.10)$$

Isto é feito examinando-se os pontos de \mathcal{P}_X um a um nesta equação 5.10. A condição da equação 5.6 garante que g_y possui entre seus zeros as posições dos erros e apagamentos ocorridos.

<<< Passo 5 >>>

Determine as síndromes $S(v, f_i)$ (equação 5.4), com $i = 1, \dots, m$.

<<< Passo 6 >>>

Determine uma solução para o sistema linear (equação 5.2)

$$\begin{bmatrix} f_1(Q_1) & f_1(Q_2) & \cdots & f_1(Q_u) \\ f_2(Q_1) & f_2(Q_2) & \cdots & f_2(Q_u) \\ \vdots & \vdots & \ddots & \vdots \\ f_m(Q_1) & f_m(Q_2) & \cdots & f_m(Q_u) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_u \end{bmatrix} = \begin{bmatrix} S(\mathbf{v}, f_1) \\ S(\mathbf{v}, f_2) \\ \vdots \\ S(\mathbf{v}, f_m) \end{bmatrix},$$

que determina os valores dos erros e apagamentos indicados por g_y (equação 5.10).

O algoritmo básico possui uma capacidade de correção de t erros e τ apagamentos, em que

$$2t + \tau \leq d - g - 1 = a - 3g + 1. \quad (5.11)$$

Com relação à sua complexidade algorítmica, obedecido o limite acima para t e τ (equação 5.11), tem-se que ela não é maior que $\mathcal{O}(d^2n + g^2n) \leq \mathcal{O}(n^3)$ [22].

5.3.2 Exemplo de decodificação de um código de Hermite

Considere o mesmo código de Hermite C_{13} ($m = 18$) apresentado no exemplo da subseção 4.2.3 do capítulo anterior. Será este código de Hermite C_{13} o código C decodificado aqui.

A curva de Hermite \mathcal{X} usada na construção deste código é dada pela equação

$$\mathcal{X} : y^3 + y = x^4.$$

O gênero de \mathcal{X} é $g = 3$. O corpo utilizado é o \mathbb{F}_9 (veja tabela com aritmética deste corpo no exemplo da subseção 4.2.3, tabela 4.1). Os pontos racionais de \mathcal{X} com as correspondentes posições na palavra código estão ilustrados na tabela 5.2 (da definição do código, tem-se que as posições numa palavra código estão associadas aos pontos racionais da curva).

A matriz H_G , matriz de paridade do código C_{13} , consiste na matriz geradora do código de Hermite C_{13} , dual de C_{13} . Tem-se, então, o divisor $G = 13Q$ (parâmetro $a = 13$). A base para este espaço $L(13Q)$ é (equação 4.12)

$$B_{13} = \left\{ 1, x, x^2, x^3, x^4, y, xy, x^2y, x^3y, y^2, xy^2 \right\}.$$

Posição	Ponto	Posição	Ponto	Posição	Ponto
1	$(1, \alpha^4)$	10	$(\alpha^3, 1)$	19	(α^6, α^4)
2	$(1, \alpha^5)$	11	(α^3, α)	20	(α^6, α^5)
3	$(1, \alpha^7)$	12	(α^3, α^3)	21	(α^6, α^7)
4	$(\alpha, 1)$	13	(α^4, α^4)	22	$(\alpha^7, 1)$
5	(α, α)	14	(α^4, α^5)	23	(α^7, α)
6	(α, α^3)	15	(α^4, α^7)	24	(α^7, α^3)
7	(α^2, α^4)	16	$(\alpha^5, 1)$	25	$(0, \alpha^2)$
8	(α^2, α^5)	17	(α^5, α)	26	$(0, \alpha^6)$
9	(α^2, α^7)	18	(α^5, α^3)	27	$(0, 0)$

Tabela 5.2: Pontos racionais da curva de Hermite $y^3 + y = x^4$ em \mathbb{F}_9 e respectivas posições na palavra código.

A matriz H_G é, portanto, (equação 4.15)

$$H_G = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 & \dots & 23 & 24 & 25 & 26 & 27 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \end{matrix} & \left[\begin{array}{cccccccccccc} 1 & 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \alpha & \alpha & \dots & \alpha^7 & \alpha^7 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^2 & \alpha^2 & \dots & \alpha^6 & \alpha^6 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^3 & \alpha^3 & \dots & \alpha^5 & \alpha^5 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^4 & \alpha^4 & \dots & \alpha^4 & \alpha^4 & 0 & 0 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & 1 & \alpha & \dots & \alpha & \alpha^3 & \alpha^2 & \alpha^6 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha & \alpha^2 & \dots & 1 & \alpha^2 & 0 & 0 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^2 & \alpha^3 & \dots & \alpha^7 & \alpha & 0 & 0 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^3 & \alpha^4 & \dots & \alpha^6 & 1 & 0 & 0 & 0 \\ 1 & \alpha^2 & \alpha^6 & 1 & \alpha^2 & \dots & \alpha^2 & \alpha^6 & \alpha^4 & \alpha^4 & 0 \\ 1 & \alpha^2 & \alpha^6 & \alpha & \alpha^2 & \dots & \alpha & \alpha^5 & 0 & 0 & 0 \end{array} \right. \end{matrix}$$

Considere que a palavra código

$$c = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \dots \ 0 \ 0 \ 0 \ 0 \ 0)$$

foi transmitida e que os vetores erro e e apagamento r ocorridos foram

$$e = (1 \ 0 \ 0 \ 0 \ 0 \ 0 \ \dots \ 0 \ \alpha^5 \ 0 \ 0 \ 0),$$

sendo $t = 2$ erros nas posições 1 e 24 (pontos $E_1 = (1, \alpha^4)$ e $E_2 = (\alpha^7, \alpha^3)$), e

$$r = \left(0 \quad \alpha^6 \quad 0 \quad 0 \quad 0 \quad \dots \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \right),$$

sendo $\tau = 1$ apagamento na posição 2 (ponto $R_1 = (1, \alpha^5)$). A palavra recebida v foi, portanto,

$$v = \left(1 \quad \alpha^6 \quad 0 \quad 0 \quad 0 \quad \dots \quad 0 \quad \alpha^5 \quad 0 \quad 0 \quad 0 \right).$$

Com relação ao divisor auxiliar $F = bQ$, observe que (equação 5.6)

$$\begin{aligned} a - b &> t + 2g - 2 \Rightarrow \\ b &< 13 - 2 - 6 + 2 \Rightarrow \\ b &< 7. \end{aligned}$$

Faça $b = 6$. Observe que, do teorema de Riemann-Roch (subseção 3.3.4 do capítulo 3), como $\deg(F) = 6 > 2g - 2 = 4$, tem-se que $l(F) = \deg(F) + 1 - g = 4$ (equação 3.8). Como $l(F) = 4 > t + \tau = 3$, então também a condição da equação 5.5 está satisfeita. Sendo $2t + \tau \leq d - g - 1 = 5$ (equação 5.11), veja que o decodificador básico é capaz de corrigir para este código até 2 erros e 1 apagamento, ou 1 erro e 3 apagamentos, ou 5 apagamentos.

A base para o espaço $L(F = 6Q)$ é

$$B_6 = \left\{ 1, x, x^2, y \right\}.$$

A matriz H_F é, portanto,

$$H_F = \begin{array}{cccccccccccc} & 1 & 2 & 3 & 4 & 5 & \dots & 23 & 24 & 25 & 26 & 27 \\ \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array} & \left[\begin{array}{cccccccccccc} 1 & 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \alpha & \alpha & \dots & \alpha^7 & \alpha^7 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^2 & \alpha^2 & \dots & \alpha^6 & \alpha^6 & 0 & 0 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & 1 & \alpha & \dots & \alpha & \alpha^3 & \alpha^2 & \alpha^6 & 0 \end{array} \right. \end{array}$$

A base para o espaço $L(G - F = 7Q)$ é

$$B_7 = \left\{ 1, x, x^2, y, xy \right\}.$$

A matriz H_{G-F} é, então,

$$H_{G-F} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & \dots & 23 & 24 & 25 & 26 & 27 \\ 1 & 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \alpha & \alpha & \dots & \alpha^7 & \alpha^7 & 0 & 0 & 0 \\ 1 & 1 & 1 & \alpha^2 & \alpha^2 & \dots & \alpha^6 & \alpha^6 & 0 & 0 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & 1 & \alpha & \dots & \alpha & \alpha^3 & \alpha^2 & \alpha^6 & 0 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha & \alpha^2 & \dots & 1 & \alpha^2 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix}$$

Do passo 1 do algoritmo 31, tem-se o sistema

$$\begin{bmatrix} 1 & 1 & 1 & \alpha^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = [0].$$

São necessárias $l(F - \sum R_i) = 3$ soluções linearmente independentes deste sistema. Considere, então, $(1, \alpha^4, 0, 0)$, $(1, 0, \alpha^4, 0)$ e $(1, 0, 0, \alpha^7)$. A base para $L(F - \sum R_i)$ será $\left\{ 1 + \alpha^4 x, 1 + \alpha^4 x^2, 1 + \alpha^7 y \right\}$. Tem-se, então,

$$H_{F-R} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & \dots & 23 & 24 & 25 & 26 & 27 \\ 0 & 0 & 0 & \alpha^2 & \alpha^2 & \dots & \alpha^5 & \alpha^5 & 1 & 1 & 1 \\ 0 & 0 & 0 & \alpha & \alpha & \dots & \alpha^3 & \alpha^3 & 1 & 1 & 1 \\ \alpha^5 & 0 & \alpha & \alpha^6 & \alpha^4 & \dots & \alpha^4 & \alpha^3 & \alpha^7 & \alpha^2 & 1 \end{bmatrix} \begin{matrix} 1 \\ 2 \\ 3 \end{matrix}$$

No passo 2, tem-se $S(v, g_1 h_1) = \alpha^2$, $S(v, g_1 h_2) = \alpha$, $S(v, g_1 h_3) = 1$, $S(v, g_1 h_4) = \alpha^5$, $S(v, g_1 h_5) = \alpha^4$, $S(v, g_2 h_1) = 1$, $S(v, g_2 h_2) = \alpha^7$, $S(v, g_2 h_3) = \alpha^6$, $S(v, g_2 h_4) = \alpha^3$, $S(v, g_2 h_5) = \alpha^2$, $S(v, g_3 h_1) = \alpha^2$, $S(v, g_3 h_2) = 1$, $S(v, g_3 h_3) = \alpha^4$, $S(v, g_3 h_4) = \alpha^4$ e $S(v, g_3 h_5) = 1$.

No passo 3, o sistema

$$\begin{bmatrix} \alpha^2 & 1 & \alpha^2 \\ \alpha & \alpha^7 & 1 \\ 1 & \alpha^6 & \alpha^4 \\ \alpha^5 & \alpha^3 & \alpha^4 \\ \alpha^4 & \alpha^2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

permite a solução não trivial $(y_1, y_2, y_3) = (\alpha^2, 1, 0)$.

Do passo 4, tem-se que

$$g_y = \alpha^3 + \alpha^6 x + \alpha^4 x^2.$$

Examinando-se os pontos da tabela 5.2 um a um, obtém-se os zeros $Q_1 = (1, \alpha^4)$ (posição 1), $Q_2 = (1, \alpha^5)$ (posição 2), $Q_3 = (1, \alpha^7)$ (posição 3), $Q_4 = (\alpha^7, 1)$ (posição 22), $Q_5 = (\alpha^7, \alpha)$ (posição 23) e $Q_6 = (\alpha^7, \alpha^3)$ (posição 24). Observe que entre estes zeros de g_y estão as posições E_1, E_2 e R_1 dos erros e do apagamento.

Do passo 5, obtém-se as síndromes da palavra recebida v , que estão descritas na tabela 5.3. Nos exemplos dos decodificadores descritos ao longo do texto, será decodificada a mesma palavra recebida v , sendo, portanto utilizado este mesmo conjunto de síndromes.

$$\begin{array}{l} S(v, f_1) = 0 \\ S(v, f_2) = \alpha^6 \\ S(v, f_3) = \alpha^4 \\ S(v, f_4) = 1^2 \end{array} \left| \begin{array}{l} S(v, f_5) = \alpha^5 \\ S(v, f_6) = \alpha^3 \\ S(v, f_7) = \alpha^4 \\ S(v, f_8) = 0 \end{array} \right. \begin{array}{l} S(v, f_9) = \alpha^7 \\ S(v, f_{10}) = \alpha \\ S(v, f_{11}) = \alpha^5 \end{array}$$

Tabela 5.3: As 11 síndromes da palavra recebida $v = (1 \alpha^6 0 0 0 \dots 0 \alpha^5 0 0 0)$, para o código de Hermite C_{18} .

No passo 6, os valores dos erros e apagamentos são obtidos como solução do sistema

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & \alpha^7 & \alpha^7 & \alpha^7 \\ 1 & 1 & 1 & \alpha^6 & \alpha^6 & \alpha^6 \\ 1 & 1 & 1 & \alpha^5 & \alpha^5 & \alpha^5 \\ 1 & 1 & 1 & \alpha^4 & \alpha^4 & \alpha^4 \\ \alpha^4 & \alpha^5 & \alpha^7 & 1 & \alpha & \alpha^3 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^7 & 1 & \alpha^2 \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^6 & \alpha^7 & \alpha \\ \alpha^4 & \alpha^5 & \alpha^7 & \alpha^5 & \alpha^6 & 1 \\ 1 & \alpha^2 & \alpha^6 & 1 & \alpha^2 & \alpha^6 \\ 1 & \alpha^2 & \alpha^6 & \alpha^7 & \alpha & \alpha^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} 0 \\ \alpha^6 \\ \alpha^4 \\ 1 \\ \alpha^5 \\ \alpha^3 \\ \alpha^4 \\ 0 \\ \alpha^7 \\ \alpha^2 \\ \alpha^5 \end{bmatrix}$$

A solução única obtida para este sistema é $(x_1, x_2, x_3, x_4, x_5, x_6) = (1, \alpha^6, 0, 0, 0, \alpha^5)$.

Obtém-se, então, os valores $(1, \alpha^6 \text{ e } \alpha^5)$ e as posições $(Q_1 = E_1, Q_2 = R_1 \text{ e } Q_6 = E_2)$ dos erros e apagamento ocorridos.

5.4 Segunda abordagem na decodificação

A segunda abordagem utilizada no desenvolvimento de esquemas de decodificação para CGA's é aquela em que as síndromes são definidas como elementos em um anel afim. O algoritmo apresentado por Porter, Shen e Pellikaan [14] é um exemplo típico desta segunda abordagem descrita na subseção 5.2.2.

Esta abordagem, como será constatado na descrição do algoritmo a seguir, baseia-se fortemente nos conceitos de diferenciais e resíduos de diferenciais. Por este motivo, por envolver fortemente conceitos matemáticos que divergem do escopo do presente trabalho, a descrição do algoritmo de Porter, Shen e Pellikaan a seguir será feita de forma simplificada, objetivando apenas transmitir a idéia do processo de decodificação de CGA's através desta segunda abordagem.

5.4.1 Algoritmo de Porter, Shen e Pellikaan

O algoritmo de decodificação de Porter pode ser visto como uma generalização da solução da equação chave para códigos de Goppa clássicos pelo algoritmo de Euclides em um anel de polinômios em uma única variável. No caso dos CGA's, ao invés de um anel de polinômios em uma variável, tem-se o anel de funções racionais em uma curva, denotado por $K_\infty(P)$, definido a seguir.

Anel afim $K_\infty(P)$

Considere uma curva projetiva, não singular e irredutível \mathcal{X} de gênero g , definida sobre o corpo \mathbb{F}_q . Considere $\mathbb{F}_q(\mathcal{X})$ o corpo das funções racionais em \mathcal{X} . Considere P, P_1, \dots, P_n pontos racionais (equivalentes a lugares de grau 1) de \mathcal{X} , e D o divisor $P_1 + \dots + P_n$. Considere também o divisor G , cujo suporte é disjuncto de $\{P_1, \dots, P_n\}$.

⁷Em todos os algoritmos de decodificação para CGA's que utilizam esta segunda abordagem, é necessário reservar um dos pontos racionais da curva para a definição de um divisor extra E . Portanto, um código de Hermite em \mathbb{F}_9 , por exemplo, que teria um comprimento máximo $n = 27$, terá, neste esquema, um comprimento máximo $n = 26$.

Defina, então, o anel afim $K_\infty(P)$ com relação ao ponto (lugar) P como sendo

$$K_\infty(P) = \{f \in \mathbb{F}_q(\mathcal{X}) \mid \text{sup}((f)_\infty) \subseteq \{P\}\},$$

ou seja, o conjunto das funções racionais em \mathcal{X} que possuem pólos exclusivamente em P . Como as curvas de Hermite só possuem pólos no ponto no infinito, P deverá ser este ponto. Para as curvas de Hermite, tem-se, então, que

$$K_\infty(P) = \mathbb{F}_q[x, y],$$

em que $x^{r+1} = y^r + y$, sendo $r^2 = q$.

Defina o grau de uma função $f \in K_\infty(P)$ como sendo

$$\text{deg}(f) = -v_P(f),$$

em que $v_P(f)$ é a função de avaliação discreta de f em P . Observe que, se $f, g \in K_\infty(P)$, então

$$\text{deg}(fg) = \text{deg}(f) + \text{deg}(g),$$

e

$$\text{deg}(f + g) \leq \max[\text{deg}(f), \text{deg}(g)].$$

Também, se $\text{deg}(f) = \text{deg}(g)$, então existe um $\lambda \in \mathbb{F}_q^*$, tal que $\text{deg}(f - \lambda g) < \text{deg}(f)$.

Isometria de códigos

Considere um código linear C . Se $c = (x_1, \dots, x_n)$ é uma palavra código de C , defina $\sigma c = (x_{\sigma(1)}, \dots, x_{\sigma(n)})$ como sendo uma permutação das posições da palavra c , e $\sigma C = \{\sigma c \mid c \in C\}$. Dois códigos lineares C_1 e C_2 em \mathbb{F}_q^n são ditos equivalentes se $C_1 = \sigma C_2$, para alguma permutação σ . Considere $\lambda = (\lambda_1, \dots, \lambda_n) \in \mathbb{F}_q^n$ uma n -úpla não nula. Defina, então, $\lambda c = (\lambda_1 x_1, \dots, \lambda_n x_n)$ e $\lambda C = \{\lambda c \mid c \in C\}$. Dois códigos lineares C_1 e C_2 em \mathbb{F}_q^n são ditos *isométricos* se existir uma n -úpla λ de elementos não nulos de \mathbb{F}_q e uma permutação σ , tais que $C_1 = \lambda \sigma C_2$. Pode-se ver que este mapeamento $\lambda \sigma$ mantém a métrica de Hamming do código invariante.

Considere C_1 e C_2 dois códigos isométricos em \mathbb{F}_q^n , com $C_1 = \lambda \sigma C_2$. Suponha que $A(C_2)$ é um algoritmo de decodificação de C_2 que corrige até t erros. O seguinte procedimento, chamado *algoritmo de decodificação induzido* $\lambda \sigma A(C_2)$, corrige o código C_1 também até t erros:

1. Entrada: v ;
2. $u = \sigma^{-1} \left(\frac{z_1}{\lambda_1}, \dots, \frac{z_n}{\lambda_n} \right)$;
3. Execute $A(C_2)$ com o vetor de entrada u para obter $c' \in C_2$;
4. Saída: $c = \lambda \sigma c'$.

Portanto, uma vez que um decodificador para um dos códigos de uma classe de isometria é dado, então todos os decodificadores dos códigos desta classe são obtidos através de algoritmos induzidos.

Se m é um inteiro, então existe uma função $h \in K_\infty(P)$ e um inteiro positivo μ , tais que os CGA's $C^*(D, mP)$ e $C^*(D, (h)_0 - \mu P)$ são isométricos.

Neste processo de decodificação, sem perda de generalidade, portanto, será considerado o código $C^*(D, G)$, em que $G = E - \mu P$ e E é um divisor efetivo.

Resíduos de diferenciais

Considere P um ponto racional de \mathcal{X} (o ponto no infinito, no caso das curvas de Hermite) disjunto de $\{P_1, \dots, P_n\}$. Considere E um divisor efetivo e μ um inteiro positivo, tais que E e $D = P_1 + \dots + P_n$ possuem suportes disjuntos e $\deg(E - \mu P) \geq 2g - 1$ (teorema de Riemman-Roch).

Mostra-se [14] que existem sempre n diferenciais $\varepsilon_1, \dots, \varepsilon_n \in \Omega(-D - \mu P)$ (espaço de diferenciais gerado pelo divisor $-D - \mu P$) independentes módulo $\Omega(-\mu P)$, tais que $\text{Res}_{P_i}(\varepsilon_j) = 1$, se $i = j$, e $\text{Res}_{P_i}(\varepsilon_j) = 0$, se $i \neq j$. Se, além disso, $\mu = 1$, então $(\varepsilon_i)_\infty = P_i + P$, para $1 \leq i \leq n$.

Mostra-se também [14] que, para todo diferencial $\omega \in \Omega(E - \mu P - D)$,

$$\omega = \sum_{j=1}^n \text{Res}_{P_j}(\omega) \varepsilon_j.$$

Defina

$$\varepsilon(c) = \sum_i c_i \varepsilon_i.$$

Este mapeamento $\varepsilon : \mathbb{F}_q^n \rightarrow \Omega_{\mathcal{X}}$ ($\Omega_{\mathcal{X}}$ é o espaço de diferenciais associados à curva \mathcal{X}) é, na verdade, um mapeamento inverso de Res_D , uma vez que $\text{Res}_D(\varepsilon(c)) = c$. Além disso, $\varepsilon(c) \in \Omega(E - \mu P - D)$ se e só se $c \in C^*(D, E - \mu P)$.

As síndromes

Segundo a primeira abordagem na decodificação de CGA's (veja subsecção 5.2.2), as síndromes de uma palavra recebida $\mathbf{v} \in \mathbb{F}_q^n$ são definidas por um mapeamento S do espaço de funções $L(G)$ para o corpo \mathbb{F}_q . Este mapeamento é dado por

$$S(\mathbf{v}, f) = \sum_{i=1}^n v_i f(P_i),$$

em que $f \in L(G)$.

Nesta segunda abordagem, a síndrome de uma palavra recebida $\mathbf{v} \in \mathbb{F}_q^n$ é definida como um elemento do anel afim $K_\infty(P)$, que consiste numa generalização das síndromes dos códigos de Goppa clássicos. A definição das síndromes que será apresentada é válida para códigos da forma $C^*(D, E - \mu P)$, o que não constitui uma restrição, uma vez que qualquer CGA é isométrico a algum código deste tipo.

Suponha que $E = (h)_0$ (divisor dos zeros de h), com $h \in K_\infty(P)$, em que h não possui zeros em qualquer dos pontos P_1, \dots, P_n de \mathcal{X} .

Mostra-se [14] que existe sempre um diferencial η , tal que

$$\begin{aligned} \text{sup}((\eta)_0) \subseteq \{P\} \Rightarrow \\ (\eta)_0 = lP \end{aligned} \tag{5.12}$$

e

$$\text{sup}((\eta)) \cap (\{P_1, \dots, P_n\} \cup \text{sup}(E)) = \emptyset.$$

Se \mathcal{X} é uma curva de gênero $g > 1$, então $l > 0$. No caso das curvas de Hermite, considerando $\eta = dx$, tem-se que

$$(\eta) = (2g - 2)P.$$

Definição 32 (Síndrome) A síndrome de uma palavra recebida $\mathbf{v} \in \mathbb{F}_q^n$ para um código $C^*(D, E - \mu P)$ é definida como sendo o mapeamento linear $S : \mathbb{F}_q^n \rightarrow \mathbb{F}_q(\mathcal{X})$ dado por

$$S(\mathbf{v})\eta = \sum_{i=1}^n v_i \frac{h(P_i) - h}{h(P_i)} \epsilon_i.$$

O nome síndrome para este mapeamento S é justificado pelo fato de que, se $E = (h)_0$, então

$$v \in C^*(D, E - \mu P) \Leftrightarrow S(v) \equiv 0 \pmod{h}.$$

Como já foi afirmado, esta síndrome $S(v)$ constitui um elemento de $K_\infty(P)$.

Decodificação pela solução da equação chave

Por simplicidade, assumamos que o diferencial η é tal que $(\eta) = (2g - 2)P$, que é o caso das curvas de Hermite.

Considere W um divisor em \mathcal{X} , tal que o ponto P não pertence ao suporte de W . Defina o ideal $K_\infty(P, W)$ como sendo

$$K_\infty(P, W) = \{f \in K_\infty(P) \mid f = 0 \text{ ou } v_Q(f) \geq n_Q(W)\},$$

em que $n_Q(W)$ é o coeficiente do ponto Q no divisor W .

Considere $E = (h)_0$ (suporte disjunto de $\{P_1, \dots, P_n\}$). Dada a síndrome $S(v)$ da palavra recebida $v \in \mathbb{F}_q^n$, a decodificação de v é feita pela solução da equação chave

$$\begin{aligned} fS(v) &\equiv r \pmod{h} \Rightarrow \\ fS(v) &= r + qh, \end{aligned} \tag{5.13}$$

em que $f \in K_\infty(P)$, $r, q \in K_\infty(P, (\eta)_\infty)$ e $\deg(r) \leq \deg(f) + 2g - 2 + \mu$. O par (f, r) é dito ser uma *solução válida* para a equação chave. Uma solução válida (f, r) é dita ser *mínima* se $\deg(f)$ for o menor entre os graus de todas as funções f' , tais que (f', r') também é uma solução válida.

Defina agora o *defeito de Clifford* σ do par (E, P) como sendo

$$\sigma = \max \left\{ \frac{\deg(E - kP)}{2} - (l(E - kP) - 1) \mid k \in \mathbb{N} \right\}.$$

Mostra-se [22] que $\sigma \leq \frac{g}{2}$. No caso de curvas de Hermite, tem-se que $\sigma \approx \frac{g}{2}$.

Segue, então, o chamado teorema da decodificação.

Teorema 33 (Decodificação) *Considere a palavra recebida $v = c + e$, em que $c \in C^*(D, E - \mu P)$ é uma palavra código e $e \in \mathbb{F}_q^n$ é um vetor de erros ocorridos. Tem-se que:*

1. (Existência) Existe uma solução válida (f, r) para a equação chave de v (equação 5.13), tal que

$$\frac{r}{f}\eta \in \Omega(-D - \mu P) \text{ e } e = \text{Res}_D \left(\frac{r}{f}\eta \right);$$

2. (Unicidade) Considere a ocorrência de até $t = \frac{d-1}{2} - \sigma$ erros, em que d é a distância mínima projetada do código e σ o defeito de Clifford. Se (f, r) é uma solução válida mínima da equação chave de v , então

$$\frac{r}{f}\eta \in \Omega(-D - \mu P) \text{ e } e = \text{Res}_D \left(\frac{r}{f}\eta \right).$$

Este teorema estabelece que o problema da decodificação resume-se à obtenção de uma solução válida para a equação chave (equação 5.13). Esta solução para a equação chave pode ser obtida através do algoritmo proposto por Ba-Zhong Shen (não descrito aqui), que utiliza uma *seqüência de subresultantes* e constitui uma generalização do algoritmo de Euclides. A utilização deste algoritmo de Shen associado ao algoritmo de Porter permite a decodificação dos $\frac{d-1}{2} - \sigma$ erros com uma complexidade $\mathcal{O}(n^3)$. Algoritmos mais eficientes podem ser obtidos com o uso do algoritmo BMS de Sakata descrito na seção seguinte.

Ressalta-se novamente que a presente descrição do algoritmo de Porter não teve como objetivo proporcionar uma completa compreensão deste algoritmo, mas apenas ilustrar a idéia existente em torno desta segunda abordagem na decodificação dos CGA's. A determinação dos diferenciais $\varepsilon_1, \dots, \varepsilon_n, \omega$ e η , por exemplo, envolveria conceitos que desviam-se dos objetivos desta dissertação. Maiores detalhes podem ser obtidos em Porter, Shen e Pellikaan [14].

5.5 Decodificação rápida

A grande parte dos primeiros algoritmos de decodificação para CGA's propostos está baseada no processo de eliminação de Gauss, o que implica uma complexidade algorítmica $\mathcal{O}(n^3)$, em que n é o comprimento do código. Como foi afirmado antes, complexidades altas são proibitivas para aplicações práticas, em que são necessários códigos com comprimento elevado.

Em vista desta necessidade de esquemas de decodificação para CGA's mais rápidos, ou seja, de menor complexidade, diversos trabalhos vêm sendo apresentados neste sentido. O protagonista ou elemento mais importante, que tem tornado possível o desenvolvimento destes esquemas mais eficientes de decodificação, é o algoritmo BMS proposto por Sakata [18], [17]. O algoritmo BMS tem possibilitado implementações rápidas do algoritmo de Porter e do algoritmo modificado de Skorobogatov e Vlăduț, mas, principalmente, tem dado origem a algoritmos rápidos de decodificação associado ao esquema de decisão por maioria proposto por Feng e Rao [4].

Devido à importância destes dois esquemas, de Sakata e de Feng e Rao, na decodificação rápida de CGA's, ambos serão descritos neste capítulo. Em seguida, eles serão ilustrados com a descrição do algoritmo rápido de decodificação apresentado Sakata, Justesen, Madelung, Jensen e Høholdt [19], que incorpora ambos os esquemas.

5.5.1 Algoritmo BMS

Dado um inteiro $\mu > 0$, considere \mathbb{Z}_+^μ o conjunto das μ -úplas de inteiros não negativos. Sendo $\alpha = (\alpha_1, \dots, \alpha_\mu) \in \mathbb{Z}_+^\mu$, considere a notação $x^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \dots x_\mu^{\alpha_\mu}$ para um monômio nas μ variáveis x_1, \dots, x_μ . Defina, então,

$$f(x_1, \dots, x_\mu) = \sum_{\alpha} f_{\alpha} x^{\alpha}$$

como sendo um polinômio em $\mathbb{F}_q[x_1, \dots, x_\mu]$.

Denote por S um arranjo de dimensão μ de elementos $S_{\alpha} \in \mathbb{F}_q$, em que $\alpha \in \mathbb{Z}_+^\mu$. Diz-se que este arranjo S satisfaz a *relação de recursão linear μ -dimensional* com polinômio característico f se

$$\sum_{\alpha} f_{\alpha} S_{\alpha+\gamma} = 0, \tag{5.14}$$

para todo $\gamma \in \mathbb{Z}_+^\mu$, em que $S_{\alpha+\gamma}$ existe. A relação de recursão linear μ -dimensional representada pelo polinômio f é dita ser *válida para o arranjo S* , se a equação 5.14 for satisfeita. O conjunto dos polinômios característicos de todas as relações de recursão lineares μ -dimensionais válidas para o arranjo S constitui um ideal de funções e será denotado aqui por $I(S)$.

O algoritmo BMS (Berlekamp-Massey-Sakata) constitui uma generalização em μ variáveis do clássico algoritmo de Berlekamp-Massey [1] para decodificação de códigos

BCH. O algoritmo de Berlekamp-Massey determina através de relações de recursão lineares (registradores de deslocamento) em uma variável, relações estas válidas para as síndromes da palavra recebida, um polinômio localizador dos erros ocorridos. Já o algoritmo BMS, que tem como entrada um arranjo μ -dimensional S de elementos de \mathbb{F}_q , fornece um conjunto de polinômios mínimos característicos (uma base de Gröbner mínima para o ideal de funções $I(S)$) correspondente às relações de recursão lineares μ -dimensionais válidas para o arranjo S dado.

Num processo de decodificação, em que $v = c + e$ é o vetor recebido, sendo e o vetor de erros ocorridos, se S é um arranjo μ -dimensional de síndromes de v , então tem-se que S satisfaz as relações de recursão lineares μ -dimensionais com polinômio característico f se e só se $f(P) = 0$, para todo $P \in \text{sup}(e)$. Portanto, o algoritmo BMS é utilizado para construir, a partir de um conjunto completo de síndromes de um vetor recebido, uma base para um ideal de funções (conjunto de polinômios mínimos característicos) em cujos zeros estão as posições dos erros ocorridos. No entanto, deve-se observar que o algoritmo BMS consiste apenas num dos componentes de um esquema de decodificação, uma vez que apenas uma parte do conjunto de síndromes é conhecido. Um algoritmo de decodificação necessita ainda de um componente adicional que determine as síndromes desconhecidas, tal como o esquema de decisão por maioria descrito na subseção seguinte.

Apesar do presente trabalho estar restrito às curvas de Hermite, que são curvas planas, ou seja, com $\mu = 2$, a descrição do algoritmo BMS será feita aqui de modo mais geral considerando arranjos μ -dimensionais de entrada, com $\mu > 0$.

Ordenação de monômios

Considere agora a ordenação de monômios lexicográfica graduada reversa definida na subseção 3.1.3. Nesta ordenação, que é denotada por $<_{grev}$, sendo $\alpha = (\alpha_1, \dots, \alpha_\mu)$ e $\beta = (\beta_1, \dots, \beta_\mu)$, diz-se que $x^\alpha <_{grev} x^\beta$ ou $\alpha <_{grev} \beta$ se e só se

$$|\alpha| = \sum_i \alpha_i < |\beta| = \sum_i \beta_i$$

ou

$$|\alpha| = |\beta|, \alpha_1 = \beta_1, \dots, \alpha_{j-1} = \beta_{j-1} \text{ e } \alpha_j > \beta_j.$$

Citando o mesmo exemplo da subseção 3.1.3, para $\mu = 2$, tem-se que $(0,0) <_{grev} (1,0) <_{grev} (0,1) <_{grev} (2,0) <_{grev} (1,1) <_{grev} (0,2) <_{grev} (3,0) <_{grev} (2,1) <_{grev}$

$(1, 2) <_{\text{grev}} \dots$, ou $1 <_{\text{grev}} x <_{\text{grev}} y <_{\text{grev}} x^2 <_{\text{grev}} xy <_{\text{grev}} y^2 <_{\text{grev}} x^3 <_{\text{grev}} x^2y <_{\text{grev}} xy^2 <_{\text{grev}} y^3 <_{\text{grev}} \dots$.

Recordando ainda da subseção 3.1.3, dado um polinômio $f(x_1, \dots, x_\mu) = \sum_{\alpha} f_{\alpha} x^{\alpha}$, segundo esta ordenação $<_{\text{grev}}$, denota-se por $\deg(f)$ e $\text{lc}(f)$ o expoente e o coeficiente do monômio líder (monômio de maior ordem), respectivamente.

Será considerada também uma *ordenação simples* de μ -úplas denotada por $<$, em que $\alpha < \beta$ se e só se $\alpha_i < \beta_i$, para todo $1 \leq i \leq \mu$. Esta ordenação simples não constitui propriamente uma ordenação de monômios, segundo a definição da subseção 3.1.3, mas servirá para expressar a divisibilidade de um monômio por outro.

Conjunto de polinômios mínimos

Uma base de Gröbner mínima \mathcal{F} para o ideal $I(S)$ descrito a pouco consiste em polinômios característicos de relações de recursão lineares μ -dimensionais que sejam válidas para o arranjo μ -dimensional S , e que possuam apenas monômios líderes mínimos segundo a ordenação de monômios $<_{\text{grev}}$ adotada. Diz-se, daí, que \mathcal{F} é um *conjunto de polinômios mínimos* para o arranjo S de entrada.

Considere agora o caso em que a equação 5.14 é satisfeita apenas em parte do arranjo S . Considere os elementos válidos do arranjo S ordenados segundo a ordem $<_{\text{grev}}$ de seus índices (μ -úplas). Se S_{α} é o elemento válido de maior ordem, em que $\alpha = \deg(f) + \gamma$, para todo $\gamma \in \mathbb{Z}_+^{\mu}$, então a equação 5.14 pode ser escrita expressando S_{α} em função dos demais elementos S_{β} , em que $\beta <_{\text{grev}} \alpha$, ou seja,

$$S_{\alpha} = \frac{-1}{\text{lc}(f)} \sum_{\beta <_{\text{grev}} \alpha} f_{\deg(f) - \alpha + \beta} S_{\beta}. \quad (5.15)$$

A relação de recursão linear μ -dimensional representada pelo polinômio $f(x_1, \dots, x_{\mu})$ é dita agora *válida para o arranjo S até a entrada S_{α}* , se $\alpha \geq \deg(f)$ (ordenação simples) e a equação 5.15 for satisfeita, ou se $\alpha \not\geq \deg(f)$. Caso contrário, se $\alpha \geq \deg(f)$ e a equação 5.15 não for satisfeita, ela é dita *inválida*.

O conjunto dos polinômios característicos de todas as relações de recursão lineares μ -dimensionais válidas para o arranjo S até a entrada S_{α} é denotado por $I_{\alpha}(S)$. Observe que este conjunto $I_{\alpha}(S)$ não constitui um ideal, uma vez que não é fechado sob a adição. Apesar disso, é fechado sob a multiplicação de monômios, ou seja, se $f(x_1, \dots, x_{\mu}) \in I_{\alpha}(S)$, então $x^{\gamma} f(x_1, \dots, x_{\mu}) \in I_{\alpha}(S)$. Além disso, a definição de

um conjunto de polinômios mínimos para este conjunto $I_\alpha(S)$ coincide ainda com a definição de uma base de Gröbner.

Reformulando a equação 5.15, tem-se que um polinômio f pertence ao conjunto $I_\alpha(S)$ se e só se

$$\sum_{\alpha} f_{\alpha} S_{\alpha+\gamma} = 0, \quad (5.16)$$

para todo $\gamma \in \mathbb{Z}_+^{\mu}$, tal que $\deg(f) + \gamma \leq_{\text{grev}} \alpha$.

Conjunto de sentinelas e conjunto delta

Defina o *conjunto delta*, denotado por $\Delta(I_\alpha(S))$, como sendo o conjunto dos monômios (na verdade, índices) que não constituem termos líderes em qualquer polinômio de $I_\alpha(S)$. Por esta razão, Sakata denominou $\Delta(I_\alpha(S))$ de *conjunto de pontos excluídos*. Um conjunto $\mathcal{F} \subset I_\alpha(S)$ é um conjunto de polinômios mínimos para $I_\alpha(S)$, se $\Delta(\mathcal{F}) = \Delta(I_\alpha(S))$.

A validade dos polinômios característicos de um conjunto \mathcal{F} , que constitui a saída do algoritmo BMS, pode ser verificada pela equação 5.16, contudo é necessário ainda verificar se os polinômios de \mathcal{F} são mínimos (se seus monômios líderes são mínimos). Para realizar esta verificação, será necessário utilizar um segundo conjunto \mathcal{G} de polinômios, chamado *conjunto de sentinelas*, definido em seguida.

Considere f um polinômio característico de uma relação de recursão linear μ -dimensional válida para o arranjo S até todas as entradas S_β , com $\beta <_{\text{grev}} \alpha$, mas não necessariamente até S_α . Defina o *valor predito* P_α para a entrada S_α associado ao polinômio f como sendo

$$P_\alpha(f) = \frac{-1}{\text{lc}(f)} \sum_{\beta <_{\text{grev}} \alpha} f_{\deg(f)-\alpha+\beta} S_\beta. \quad (5.17)$$

Observe que esta expressão consiste simplesmente no lado direito da equação 5.15. Diz-se, então, que a relação de recursão linear μ -dimensional representada pelo polinômio f é válida até a entrada S_α do arranjo S se e só se o valor real de S_α for igual ao valor predito P_α .

Considere agora um polinômio característico $g(x_1, \dots, x_\mu)$ de uma relação de recursão linear μ -dimensional que é válida para o arranjo S μ -dimensional até todas as

entradas S_β , com $\beta <_{\text{grev}} \alpha$, mas que é inválida até a entrada S_α . Defina, então, a *extensão* de g como sendo o vetor (índice)

$$\text{Span}(g) = \alpha - \deg(g) \quad (5.18)$$

e sua *discrepância* como sendo

$$\delta_g = \text{lc}(g) \{S_\alpha - P_\alpha(g)\} = \sum_{\alpha} g_\alpha S_{\alpha + \text{Span}(g)} \neq 0. \quad (5.19)$$

Caso $g \notin I_\alpha(S)$, ou seja, caso $P_\alpha(g) \neq S_\alpha$, tem-se que $\text{Span}(g) \in \Delta(I_\alpha(S))$. Neste caso, o polinômio $g(x_1, \dots, x_\mu)$ é dito ser um *sentinela* para o ponto $\text{Span}(g)$.

Defina o conjunto Δ como tendo um canto interior (veja exemplo na seqüência do texto, tabela 5.5) associado a cada sentinela do conjunto de sentinelas $\mathcal{G} \subset \mathbb{F}_q[x_1, \dots, x_\mu] \setminus I(S)$. Estes cantos interiores de Δ são também membros do conjunto $\Delta(I(S))$.

Por fim, a verificação de se um conjunto \mathcal{F} é um conjunto de polinômios mínimos, dado um conjunto de sentinelas \mathcal{G} , é feita com base no fato que segue. Se $\mathcal{F} \subset I_\alpha(S)$ e $\mathcal{G} \subset \mathbb{F}_q[x_1, \dots, x_\mu] \setminus I_\alpha(S)$ é um conjunto de sentinelas para o conjunto delta $\Delta(\mathcal{F})$, então $\Delta(\mathcal{F}) = \Delta(I_\alpha(S))$, o que implica que \mathcal{F} é um conjunto de polinômios mínimos para $I_\alpha(S)$. Se $\mathcal{F} \subset I(S)$ e $\mathcal{G} \subset \mathbb{F}_q[x_1, \dots, x_\mu] \setminus I(S)$ é um conjunto de sentinelas para o conjunto delta $\Delta(\mathcal{F})$, então $\Delta(\mathcal{F}) = \Delta(I(S))$, o que implica que \mathcal{F} é uma base de Gröbner mínima para o ideal $I(S)$.

Descrição do algoritmo

O algoritmo BMS (algoritmo 34) basicamente opera sobre dois conjuntos: um conjunto de polinômios mínimos \mathcal{F} e um conjunto de sentinelas \mathcal{G} . Cada iteração do algoritmo toma como entrada um conjunto de polinômios mínimos \mathcal{F} para um conjunto $I_\alpha(S)$ e um conjunto de sentinelas \mathcal{G} para um conjunto delta $\Delta = \Delta(I_\alpha(S))$, e produz um conjunto de polinômios mínimos \mathcal{F}^+ para um conjunto $I_{\alpha^+}(S)$ e um conjunto de sentinelas \mathcal{G}^+ para um conjunto delta $\Delta^+ = \Delta(I_{\alpha^+}(S))$, sendo α^+ o índice de ordem imediatamente superior a α (ordenamento $<_{\text{grev}}$). Observe que a saída do algoritmo consiste apenas numa atualização da entrada.

Algoritmo 34 (Algoritmo BMS)

Entradas:

- Um arranjo μ -dimensional S de elementos de \mathbb{F}_q ;
- Um índice $\alpha \in \mathbb{Z}_+^\mu$;
- Um conjunto de polinômios mínimos \mathcal{F} para $I_\alpha(S)$;
- Um conjunto de sentinelas \mathcal{G} para $\Delta(I_\alpha(S))$, com suas extensões e discrepâncias.

Saídas:

- Um conjunto de polinômios mínimos \mathcal{F}^+ para $I_{\alpha^+}(S)$;
- Um conjunto de sentinelas \mathcal{G}^+ para $\Delta(I_{\alpha^+}(S))$, com suas extensões e discrepâncias.

<<< Passo 1 >>>

Considere o conjunto $\mathcal{F}' = \{f \in \mathcal{F} \mid \deg(f) \leq \alpha^+\}$.

Para cada $f \in \mathcal{F}'$, calcule o valor predito (equação 5.17)

$$P_{\alpha^+}(f) = \frac{-1}{\text{lc}(f)} \sum_{\beta <_{\text{rev}} \alpha^+} f_{\deg(f) - \alpha^+ + \beta} S_\beta.$$

Considere o conjunto $\mathcal{N} = \{f \in \mathcal{F}' \mid P_{\alpha^+}(f) \neq S_{\alpha^+}\}$.

<<< Passo 2 >>>

Faça $\mathcal{G}^+ = \mathcal{G} \cup \mathcal{N}$.

Para cada $f \in \mathcal{N}$, calcule e armazene a extensão (equação 5.18)

$$\text{Span}(f) = \alpha^+ - \deg(f).$$

Faça $\Delta^+ = \Delta \cup \{\text{Span}(f) \mid f \in \mathcal{N}\}$.

Para cada $f \in \mathcal{N}$, calcule e armazene a discrepância (equação 5.19)

$$\delta_f = \text{lc}(f) [S_{\alpha^+} - P_{\alpha^+}(f)].$$

<<< Passo 3 >>>

Para cada $\beta \in \text{Ext } \Delta^+$ (cantos externos de Δ^+), proceda o seguinte:

- Primeiro, se existir um $f \in \mathcal{F} \setminus \mathcal{N}$, tal que $\deg(f) = \beta$, então faça

$$h^{(\beta)}(x_1, \dots, x_\mu) = f(x_1, \dots, x_\mu);$$

– Em caso contrário, se $\beta \not\leq \alpha^+$, tome um $f \in \mathcal{N}$, tal que $\deg(f) \leq \beta$, e faça

$$h^{(\beta)}(x_1, \dots, x_\mu) = x^{\beta - \deg(f)} f(x_1, \dots, x_\mu);$$

– Em último caso, tome um $f \in \mathcal{N}$, tal que $\deg(f) \leq \beta$, e um $g \in \mathcal{G}$, tal que $\text{Span}(g) \geq \alpha^+ - \beta$. Considere os índices $q = \beta - \deg(f)$ e $p = \text{Span}(g) - \alpha^+ + \beta$. Faça, então,

$$h^{(\beta)}(x_1, \dots, x_\mu) = x^q f(x_1, \dots, x_\mu) - \frac{\delta_f}{\delta_g} x^p g(x_1, \dots, x_\mu);$$

Por fim, tem-se que

$$\mathcal{F}^+ = \{h^{(\beta)}(x_1, \dots, x_\mu) \mid \beta \in \text{Ext } \Delta^+\}.$$

No passo 1 do algoritmo BMS (algoritmo 34), a validade dos polinômios de \mathcal{F} , que, por hipótese, correspondem a relações de recursão lineares μ -dimensionais válidas para todas as entradas do arranjo S até a entrada S_{α^+} , é testada para a próxima entrada S_{α^+} . Os polinômios inválidos para a entrada S_{α^+} podem ser usados como sentinelas, sendo armazenados no conjunto \mathcal{N} .

No passo 2, o conjunto de pontos excluídos $\Delta = \Delta(I_{\alpha}(S))$ é atualizado usando os novos sentinelas contidos no conjunto \mathcal{N} . Observe que pode ocorrer de um ou mais $f \in \mathcal{N}$ serem sentinelas de pontos excluídos $\text{Span}(f)$ que já pertençam ao conjunto Δ . Além disso, deve-se levar em consideração que ao acrescentar um novo ponto excluído γ ao conjunto Δ , deve-se certificar que todos os pontos $\beta \leq \gamma$ também pertençam ao conjunto atualizado Δ^+ (acrescentá-los se necessário), de modo que este conjunto Δ^+ também seja um conjunto delta, segundo a definição. Os valores de $\text{Span}(f)$ e δ_f são armazenados para um possível uso posteriormente no passo 3.

O passo 3 consiste na determinação do conjunto atualizado \mathcal{F}^+ de polinômios válidos para o arranjo S até a entrada S_{α^+} . Este conjunto \mathcal{F}^+ é um conjunto de polinômios mínimos para $I_{\alpha^+}(S)$ e \mathcal{G}^+ é um conjunto de sentinelas para $\Delta(I_{\alpha^+}(S))$.

Exemplo

Considere o arranjo S de dimensão $\mu = 2$ com elementos de \mathbb{F}_9 (veja a aritmética deste corpo na tabela 4.1, subseção 4.2.3) descrito na tabela 5.4. Observe que os elementos

deste arranjo são os mesmos valores das síndromes da palavra recebida v que foi utilizada no exemplo do algoritmo básico (subseção 5.3.2). Isto não é feito despropositadamente. Esta mesma entrada será utilizada também no exemplo do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt descrito numa subseção seguinte. Intenciona-se, com isso, facilitar a compreensão do processo de decodificação dos CGA's. Os elementos $*$ da tabela 5.4 são síndromes desconhecidas que, num processo de decodificação, devem ser determinadas. O elemento $S_{0,3}$ foi obtido através da relação de recursão linear

$$S_{a+4,b} = S_{a,b+3} + S_{a,b+1}$$

fornecida pela própria equação da curva geradora do código ($x^4 = y^3 + y$).

		j						
		0	1	2	3	4	5	6
i	0	0	α^3	α^2	1	*	*	...
	1	α^6	α^4	α^5	*	*	...	
	2	α^4	0	*	*	...		
	3	1	α^7	*	...			
	4	α^5	*	\vdots				
	5	*	\vdots					
	6	\vdots						

Tabela 5.4: Arranjo S bi-dimensional de elementos $S_{i,j} \in \mathbb{F}_9$ utilizado como entrada para o algoritmo BMS. Estes elementos de \mathbb{F}_9 são, na verdade, as síndromes da palavra recebida que está sendo utilizada nos exemplos dos algoritmos de decodificação. Os elementos $*$ representam síndromes desconhecidas.

O arranjo S dado apresenta 12 elementos de entrada. Como o algoritmo BMS consiste numa única iteração, será necessário processá-lo 12 vezes para se obter o conjunto de polinômios mínimos das relações de recursão lineares válidas para o arranjo S até a entrada $S_{3,1}$, que é a entrada de maior ordem segundo a ordenação lexicográfica graduada reversa $<_{grrev}$ adotada.

Iteração 1 Considere inicialmente $\mathcal{F} = \{f_0\}$, em que $f_0 = 1$, $\mathcal{G} = \emptyset$ e $\Delta = \emptyset$. Faça $\alpha = (0,0)$ e $\alpha^+ = (1,0)$.

Do passo 1, tem-se

- $\mathcal{F}' = \{f_0\}$;
- $P_{1,0}(f_0) = -1(0) = 0$;
- $\mathcal{N} = \{f_0\}$.

Do passo 2, tem-se

- $\mathcal{G}^+ = \{f_0\}$;
- $\text{Span}(f_0) = (1, 0) - (0, 0) = (1, 0)$ (este ponto é o sentinela da função f_0);
- $\Delta^+ = \{(1, 0)\}$;
- $\delta_{f_0} = 1[\alpha^6 - 0] = \alpha^6$.

No passo 3, tem-se que $\text{Ext } \Delta^+ = \{(0, 1), (2, 0)\}$ (veja tabela 5.5). Então,

	0	1	2	3
0	•	◦	•	•
1	•	•	•	•
2	◦	•	•	•
3	•	•	•	•

Tabela 5.5: Ilustração dos cantos externos do conjunto delta $\Delta^+ = \{(1, 0)\}$. Os pontos de $\text{Ext } \Delta^+$ estão representados por símbolos ◦. Observe que estes pontos constituem realmente cantos na tabela.

- Para $\beta = (0, 1)$, tem-se que

– Primeiro, não há $f \in \mathcal{F} \setminus \mathcal{N}$, tal que $\text{deg}(f) = (0, 1)$;

* Segundo, $\beta \notin \alpha^+$. Sendo $f_0 \in \mathcal{N}$ e $\text{deg}(f_0) \leq (0, 1)$, faça

$$h^{(0,1)}(x, y) = x^{(0,1)-(0,0)} f_0 = y = f_1.$$

- Para $\beta = (2, 0)$, tem-se que

– Primeiro, não há $f \in \mathcal{F} \setminus \mathcal{N}$, tal que $\deg(f) = (2, 0)$;

* Segundo, $\beta \not\leq \alpha^+$. Sendo $f_0 \in \mathcal{N}$ e $\deg(f_0) \leq (2, 0)$, faça

$$h^{(2,0)}(x, y) = x^{(2,0)-(0,0)} f_0 = x^2 = f_2.$$

• Portanto, $\mathcal{F}^+ = \{f_1, f_2\}$ é um conjunto de polinômios mínimos para $I_{1,0}(S)$.

Iteração 2 Considere $\mathcal{F} = \{f_1, f_2\}$, $\mathcal{G} = \{f_0\}$ e $\Delta = \{(1, 0)\}$. Faça $\alpha = (1, 0)$ e $\alpha^+ = (0, 1)$.

Do passo 1, tem-se

- $\mathcal{F}' = \{f_1\}$;
- $P_{0,1}(f_1) = -1(0 + 0) = 0$;
- $\mathcal{N} = \{f_1\}$.

Do passo 2, tem-se

- $\mathcal{G}^+ = \{f_0, f_1\}$;
- $\text{Span}(f_1) = (0, 1) - (0, 1) = (0, 0)$;
- Como $\text{Span}(f_1) = (0, 0)$ já é um ponto interno de Δ ($(0, 0) < (1, 0)$), então pode ser descartado. A função f_1 , que tem como sentinela este ponto $(0, 0)$, também pode ser descartada de \mathcal{G}^+ . Então, tem-se que $\mathcal{G}^+ = \{f_0\}$ e $\Delta^+ = \{(1, 0)\}$;
- $\delta_{f_1} = 1[\alpha^3 - 0] = \alpha^3$.

No passo 3, tem-se novamente $\text{Ext } \Delta^+ = \{(0, 1), (2, 0)\}$. Daí,

• Para $\beta = (0, 1)$, tem-se que

– Primeiro, não há $f \in \mathcal{F} \setminus \mathcal{N}$, tal que $\deg(f) = (0, 1)$;

* Segundo, $\beta \leq \alpha^+$;

* Por último, tem-se que $f_1 \in \mathcal{N}$ e $\deg(f_1) \leq \beta$, e $f_0 \in \mathcal{G}$ e $\text{Span}(f_0) \geq \alpha^+ - \beta = (0, 0)$. Faça $q = (0, 1) - (0, 1) = (0, 0)$ e $p = (1, 0) - (0, 1) + (0, 1) = (1, 0)$. Então,

$$h^{(0,1)}(x, y) = x^{(0,0)} f_1 - \frac{\alpha^3}{\alpha^6} x^{(1,0)} f_0 = y + \alpha x = f_3.$$

- Para $\beta = (2, 0)$, tem-se que

– Primeiro, $f_2 \in \mathcal{F} \setminus \mathcal{N}$ e $\deg(f_2) = (2, 0)$. Então,

$$h^{(2,0)}(x, y) = f_2.$$

- Portanto, $\mathcal{F}^+ = \{f_3, f_2\}$ é um conjunto de polinômios mínimos para $I_{0,1}(S)$.

Demais iterações A tabela 5.6 descreve os conjuntos \mathcal{F} , \mathcal{G} e Δ resultados de cada iteração do algoritmo BMS para o arranjo S dado na tabela 5.4. A tabela 5.7 denota os polinômios envolvidos neste procedimento, suas extensões e discrepâncias.

$\alpha = (i, j)$	\mathcal{F}	\mathcal{G}	Δ
(0, 0)	$\{f_0\}$	\emptyset	\emptyset
(1, 0)	$\{f_1, f_2\}$	$\{f_0\}$	$\{(1, 0)\}$
(0, 1)	$\{f_3, f_2\}$		
(2, 0)	$\{f_3, f_4\}$		
(1, 1)	$\{f_5, f_4\}$		
(0, 2)	$\{f_4, f_6, f_7\}$	$\{f_0, f_5\}$	$\{(1, 0), (0, 1)\}$
(3, 0)	$\{f_8, f_6, f_7\}$		
(2, 1)	$\{f_8, f_6, f_7\}$		
(1, 2)	$\{f_8, f_9, f_{10}\}$		
(0, 3)	$\{f_8, f_9, f_{11}\}$		
(4, 0)	$\{f_8, f_9, f_{11}\}$		
(3, 1)	$\{f_8, f_9, f_{11}\}$		

Tabela 5.6: Saída do algoritmo BMS para o arranjo S de entrada.

5.5.2 Decisão por maioria

No corpo \mathbb{C} dos números complexos, a transformada de Fourier discreta de um vetor $\mathbf{u} = (u_0, \dots, u_{n-1})$ de números complexos é um vetor $\mathbf{U} = (U_0, \dots, U_{n-1})$, em que

$$U_k = \sum_{i=0}^{n-1} e^{-j\frac{2\pi}{n}ki} u_i, \quad k = 0, \dots, n-1.$$

Polinômios f_i	Span (f_i)	δ_{f_i}
$f_0(x, y) = 1$	(1, 0)	α^6
$f_1(x, y) = y$	(0, 0)	α^3
$f_2(x, y) = x^2$	(0, 0)	α^4
$f_3(x, y) = y + \alpha x$	(1, 0)	α^3
$f_4(x, y) = x^2 + \alpha^2 x$	(1, 0)	α
$f_5(x, y) = y + \alpha x + \alpha$	(0, 1)	α
$f_6(x, y) = xy + \alpha x^2 + \alpha x$	(0, 1)	α
$f_7(x, y) = y^2 + \alpha xy + \alpha y + \alpha^7 x$	(1, 0)	α^4
$f_8(x, y) = x^2 + \alpha^2 x + \alpha^7$	-	-
$f_9(x, y) = xy + \alpha x^2 + \alpha^4 y + \alpha^5$	-	-
$f_{10}(x, y) = y^2 + \alpha xy + \alpha y + \alpha^7 x + \alpha^2$	(0, 1)	α^7
$f_{11}(x, y) = y^2 + \alpha xy + y + \alpha$	-	-

Tabela 5.7: Polinômios envolvidos no processamento do algoritmo BMS neste exemplo, com suas respectivas extensões e discrepâncias.

Observe que o termo $e^{-j\frac{2\pi}{n}}$ constitui uma raiz n -ésima da unidade em \mathbb{C} , ou seja, $(e^{-j\frac{2\pi}{n}})^n = 1$. Num corpo finito \mathbb{F}_q , um elemento α de ordem n também constitui uma raiz n -ésima da unidade. Por analogia, define-se a transformada de Fourier de um vetor $\mathbf{v} = (v_1, \dots, v_n)$ em um corpo finito \mathbb{F}_q como sendo o vetor $\mathbf{V} = (V_1, \dots, V_n)$, em que

$$V_j = \sum_{i=1}^n \alpha^{ij} v_i, \quad j = 1, \dots, n, \quad (5.20)$$

sendo $\alpha \in \mathbb{F}_q$ um elemento de ordem n . O vetor \mathbf{v} pode ser obtido pela transformada inversa dada por

$$v_i = \frac{1}{n} \sum_{j=1}^n \alpha^{-ij} V_j, \quad i = 1, \dots, n,$$

em que $n \equiv 0 \pmod{p}$, sendo p um inteiro primo e $q = p^m$, para algum inteiro m .

Observe que a equação 5.20 equivale à expressão das síndromes de uma palavra recebida \mathbf{v} para um código em uma variável (códigos BCH, de Reed-Solomon, etc.). De um modo geral, a transformada de Fourier de um vetor recebido $\mathbf{v} = \mathbf{c} + \mathbf{e}$, em

que e é um vetor erro, para um código (n, k) qualquer, pode ser vista como sendo o conjunto das síndromes (equação 5.1)

$$S_i(v) = S_i(e) = h_i e^T, \quad i = 1, \dots, n,$$

em que h_i são as linhas da matriz de paridade do código. O importante neste fato é que, uma vez conhecidas todas as n síndromes de v (apenas $n - k$ síndromes são conhecidas a princípio), é possível determinar o vetor e ocorrido através da transformada inversa de Fourier.

O esquema de decisão por maioria proposto por Feng e Rao [4] permite se obter recursivamente as k síndromes desconhecidas, possibilitando, assim, a decodificação de v .

Na subseção seguinte, será descrito um algoritmo rápido de decodificação para CGA's definidos sobre curvas planas, que usa o algoritmo BMS e o esquema de decisão por maioria adaptado ao algoritmo BMS. Na presente subseção, o esquema de decisão por maioria será descrito numa forma semelhante à apresentada por Feng e Rao, o que envolve o conceito de anti-lacunas apresentado na subseção 3.3.5 do capítulo 3.

Anti-lacunas

Considere uma curva algébrica plana \mathcal{X} de gênero g sobre um corpo \mathbb{F}_q . Considere o divisor $D = P_1 + \dots + P_n$ de pontos racionais de \mathcal{X} e o ponto racional Q também de \mathcal{X} , mas disjunto do suporte de D .

Denote por C_{m_k} o CGA $C^*(D, m_k Q)$ de comprimento n e dimensão $n - m_k + g - 1$. Assuma $m_k > 2g$ e considere $k = m_k - g + 1$ a dimensão do código dual.

Denote por $(m_i \mid i \in \mathbb{N})$ a seqüência das anti-lacunas de Q , em que

$$0 < m_1 < \dots < m_{g-1} < 2g$$

e $m_i = i + g$, para $i = g, g + 1, \dots, m_k - g$.

Denote por g_i uma função racional que possui um pólo de ordem m_i no ponto Q , e nenhum outro pólo mais. Tem-se que g_1, \dots, g_k constitui uma base para o espaço $L(m_k Q)$ ortogonal ao código C_{m_k} .

Matriz de síndromes bi-dimensionais

Se e é um vetor erro ocorrido em uma palavra recebida, defina S como sendo a *matriz de síndromes bi-dimensionais* correspondentes a e e dada por

$$S = \begin{bmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,k} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ S_{k,1} & S_{k,2} & \cdots & S_{k,k} \end{bmatrix},$$

cujos elementos são dados por

$$S_{i,j}(e) = \sum_{l=1}^t e_{u_l} g_i(P_{u_l}) g_j(P_{u_l}),$$

em que os u_l 's são as posições dos t erros ocorridos. Se $v = c + e$ é uma palavra recebida, para $c \in C_{m_k}$, e $m_i + m_j = m_p \leq m_k$, então $g_i g_j \in L(m_p Q) \subseteq L(m_k Q)$ e $S_{i,j}(e) = S_{i,j}(v)$. Portanto, $S_{i,j}$ é uma entrada conhecida da matriz S , se $m_i + m_j \leq m_k$.

Defina o conjunto de pares N_k como sendo

$$N_k = \{(i, j) \in \mathbb{N}^2 \mid m_i + m_j = m_{k+1}\}$$

e denote por n_k seu número de elementos. As entradas da matriz S com índices $(i, j) \in N_k$ são as primeiras síndromes desconhecidas com relação ao código C_{m_k} a serem determinadas. Uma vez determinada uma entrada $S_{i,j}$, com $(i, j) \in N_k$, então todas as outras entradas $S_{i',j'}$, com $(i', j') \in N_k$, que não são necessariamente iguais, podem ser obtidas. Isto, porque cada uma das funções $g_i g_j$, $g_{i'} g_{j'}$ e g_{k+1} gera o espaço $L(m_{k+1} Q) \setminus L(m_k Q)$. Ou seja, existem elementos $\lambda_{i,j}, \lambda_{i',j',r} \in \mathbb{F}_q$, com $\lambda_{i,j} \neq 0$, tais que

$$g_i g_j = \lambda_{i,j} g_{k+1} + \sum_{r \leq k} \lambda_{i,j,r} g_r \Rightarrow$$

$$S_{i,j} = \lambda_{i,j} S_{k+1} + \sum_{r \leq k} \lambda_{i,j,r} S_r,$$

para todo $(i, j) \in N_k$. Além disso, esta relação é a mesma para todos os vetores erro.

Considere agora a matriz $S(i, j)$, dada por

$$S(i, j) = \begin{bmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,j-1} & S_{1,j} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,j-1} & S_{2,j} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ S_{i-1,1} & S_{i-1,2} & \cdots & S_{i-1,j-1} & S_{i-1,j} \\ S_{i,1} & S_{i,2} & \cdots & S_{i,j-1} & S_{i,j} \end{bmatrix} \quad (5.21)$$

Se $m_i + m_j = m_{k+1}$, então todos os elementos de $S(i, j)$ são conhecidos, exceto $S_{i,j}$. Se $m_i + m_j = m_k$, então $S(i, j)$ equivale à matriz de síndromes determinada no passo 2 do algoritmo básico (algoritmo 31) para o código C_{m_k} (m_k é o parâmetro a naquele algoritmo).

Candidatos e discrepâncias

Considere $(i, j) \in N_k$, ou seja, $m_i + m_j = m_{k+1}$.

Definição 35 (Candidato) *Se as matrizes $S(i-1, j-1)$, $S(i-1, j)$ e $S(i, j-1)$ possuem o mesmo posto, então (i, j) é dito ser um candidato com relação ao código C_{m_k} .*

Se (i, j) é um candidato, então existe um único valor $S'_{i,j}$ a ser atribuído à entrada desconhecida $S_{i,j}$, de modo que as matrizes $S(i-1, j-1)$ e $S(i, j)$ possuam o mesmo posto. O elemento $S'_{i,j}$ é dito ser um *valor candidato* ou *predito* da síndrome desconhecida $S_{i,j}$.

Denote o número de candidatos *verdadeiros* ou *corretos*, em que $S'_{i,j} = S_{i,j}$, por T e o número de candidatos *falsos* ou *incorretos*, em que $S'_{i,j} \neq S_{i,j}$, por F .

Definição 36 (Discrepância) *Um índice (i, j) é dito ser uma discrepância (não confundir com o conceito de discrepância usado na descrição do algoritmo BMS na subseção 5.5.1), se as matrizes $S(i-1, j-1)$, $S(i-1, j)$ e $S(i, j-1)$ possuem um mesmo posto, que difere do posto de $S(i, j)$.*

Se for aplicado o algoritmo de eliminação de Gauss sem troca de linhas ou colunas à matriz de síndromes bi-dimensionais S , as discrepâncias serão os pivôs da matriz

resultante. Portanto, o número total de discrepâncias, denotado por DT , é igual ao posto de S . Além disso, a matriz S pode ser escrita na forma

$$S = XYX^T,$$

em que

$$X = \begin{bmatrix} g_1(P_{u_1}) & g_1(P_{u_2}) & \cdots & g_1(P_{u_t}) \\ g_2(P_{u_1}) & g_2(P_{u_2}) & \cdots & g_2(P_{u_t}) \\ \vdots & \vdots & \ddots & \vdots \\ g_k(P_{u_1}) & g_k(P_{u_2}) & \cdots & g_k(P_{u_t}) \end{bmatrix}$$

e

$$Y = \begin{bmatrix} e_{u_1} & 0 & \cdots & 0 \\ 0 & e_{u_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e_{u_t} \end{bmatrix}$$

Portanto, o número total de discrepâncias é, no máximo, igual ao número de erros ocorridos t .

Tomada de decisão

Considere $v = c + e$ uma palavra recebida com $t \leq \frac{n_r-1}{2}$ erros com relação ao código C_{m_k} . Então, todas as síndromes bi-dimensionais $S_{i,j}$, com $m_i + m_j \leq m_k$, são conhecidas e as demais síndromes são desconhecidas.

Denote o número de discrepâncias conhecidas por K . Um candidato é incorreto se e só se for uma discrepância. Então,

$$K + F \leq DT \leq t. \quad (5.22)$$

Se (i, j) é uma discrepância conhecida, então todas as entradas (i, j') e (i', j) , com $j' > j$ e $i' < i$, não são candidatos. Se $(i, j) \in N_k$ não é um candidato, então existe pelo menos uma discrepância conhecida na mesma linha i ou coluna j . Portanto, o número de pares $(i, j) \in N_k$ que não são candidatos é, no máximo, $2K$.

O número de pares $(i, j) \in N_k$ que são candidatos é igual a $T + F$. Portanto,

$$n_r = n^\circ \text{ de candidatos} + n^\circ \text{ de não candidatos} \leq (T + F) + 2K. \quad (5.23)$$

Como, por hipótese, $w(e) = t \leq \frac{n_r - 1}{2}$, tem-se das inequações 5.22 e 5.23 que

$$\begin{aligned} K + F &\leq \frac{n_r - 1}{2} \Rightarrow \\ 2K + 2F + 1 &\leq n_r \leq T + F + 2K \Rightarrow \\ F &< T. \end{aligned}$$

Não há uma forma direta de determinar se um candidato é ou não verdadeiro. Contudo, se for atribuído um valor predito a cada um dos candidatos, então a maioria, T candidatos, terá o mesmo valor, que é, por definição, o valor correto de S_{r+1} .

Esta presente descrição teve por objetivo introduzir a idéia do esquema de decisão por maioria de Feng e Rao. Um algoritmo de decodificação com decisão por maioria na forma apresentada aqui tem uma complexidade alta $\mathcal{O}(n^3)$. Na subseção seguinte, será apresentado um algoritmo rápido de decodificação, que adapta o esquema de decisão por maioria ao algoritmo BMS, e possui complexidade $\mathcal{O}(n^{\frac{3}{2}})$. Em seguida, será dado um exemplo ilustrativo deste processo.

5.5.3 Algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt

Nas subseções anteriores, foram descritos o algoritmo BMS e o esquema de decisão por maioria, que têm propiciado o desenvolvimento de diversos algoritmos rápidos de decodificação para CGA's. Um bom exemplo da utilização destes esquemas é o algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt [19], que incorpora tanto o algoritmo BMS quanto o esquema de decisão por maioria de Feng e Rao.

Algoritmo BMS

Considere \mathcal{X} uma curva algébrica plana de gênero g , cuja equação possui grau $\mu = r + 1$ e é definida em $\mathbb{F}_q[x, y]$. Considere $\{P_1, \dots, P_n\}$ um conjunto de pontos racionais da curva \mathcal{X} sob o corpo \mathbb{F}_q e o divisor $D = P_1 + \dots + P_n$. Considere o divisor $G = mQ$, em que $Q \notin \{P_1, \dots, P_n\}$ é um ponto racional de \mathcal{X} . Suponha também que $m > 2g - 2$.

Considere o CGA $C^*(D, G)$, denotado por C , de distância mínima projetada d ,

comprimento n , dimensão $n - k$ (k é a dimensão do código dual) e matriz de paridade

$$H = \begin{bmatrix} g_{a_1, b_1}(P_1) & g_{a_1, b_1}(P_2) & \cdots & g_{a_1, b_1}(P_n) \\ g_{a_2, b_2}(P_1) & g_{a_2, b_2}(P_2) & \cdots & g_{a_2, b_2}(P_n) \\ \vdots & \vdots & \ddots & \vdots \\ g_{a_k, b_k}(P_1) & g_{a_k, b_k}(P_2) & \cdots & g_{a_k, b_k}(P_n) \end{bmatrix},$$

em que $g_{a_i, b_i} = x^{a_i} y^{b_i}$ e $\{g_{a_1, b_1}, \dots, g_{a_k, b_k}\}$ é uma base para o espaço $L(G)$.

Considere a descrição do algoritmo BMS feita na subseção 5.5.1.

Considere uma palavra recebida $v \in \mathbb{F}_q^n$, acrescida de um vetor erro $e \in \mathbb{F}_q^n$, cujas síndromes conhecidas são

$$S_{a_i, b_i} = \sum_{j=1}^t e_{u_j} g_{a_i, b_i}(P_{u_j}), \quad i = 1, \dots, k, \quad (5.24)$$

em que os u_j 's são as posições dos t erros ocorridos. Considere estas síndromes ordenadas segundo a ordenação lexicográfica graduada reversa $<_{grev}$ utilizada na descrição do algoritmo BMS na subseção 5.5.1, e dispostas em um arranjo $S_{q-1 \times q-1}$, em que $S_{i,j}$ é o elemento da linha i e coluna j .

Mostrou-se que, se todas as síndromes $S_{i,j}$, com $i, j < q - 1$, são conhecidas, então o vetor erro ocorrido pode ser determinado pela transformada inversa de Fourier dada por

$$e_u = \sum_{i,j < q-1} S_{i,j} p_{xu}^{-i} p_{yu}^{-j}, \quad (5.25)$$

em que e_u é o componente do vetor erro e correspondente ao ponto $P_u = (p_{xu}, p_{yu})$ da curva \mathcal{X} .

Relembrando da subseção 5.5.1, um polinômio

$$f(x, y) = \sum_{i,j} f_{i,j} x^i y^j$$

é dito fornecer uma relação de recursão linear válida para o arranjo S até a entrada $S_{a,b}$, se

$$\sum_{(i,j) \leq_{grev} (a,b)} f_{i,j} S_{i+u, j+v} = 0,$$

sendo u e v inteiros, em que $S_{i+u, j+v}$ existe. Por exemplo, a equação da curva \mathcal{X} fornece uma relação de recursão linear válida para todo o arranjo S . Para uma curva de Hermite dada por (equação 4.9)

$$x^\mu = y^{\mu-1} + y,$$

tem-se que

$$S_{i+\mu, j} = S_{i, j+\mu-1} + S_{i, j+1} \quad (5.26)$$

ou

$$S_{i+\mu, j-\mu+1} = S_{i, j} + S_{i, j-\mu+2}, \text{ para } j \geq \mu - 1. \quad (5.27)$$

Considere $S_{a,b}$ a síndrome conhecida de maior ordem, segundo a ordenação \leq_{grev} . As síndromes $S_{i,j}$, em que $(i, j) <_{\text{grev}} (a, b)$, que não foram obtidas pela equação 5.24, podem ser determinadas pela relação de recursão linear fornecida pela equação da curva \mathcal{X} , de modo que se pode ter um arranjo S constituído por todas as síndromes $S_{i,j}$, com $(i, j) <_{\text{grev}} (a, b)$.

O algoritmo BMS recebe como entrada o arranjo S e fornece um conjunto de polinômios mínimos característicos, denotado por

$$\mathcal{F}_{a,b} = \{f^{(1)}, \dots, f^{(\nu)}\},$$

correspondente às relações de recursão lineares válidas para o arranjo S até a entrada de maior ordem $S_{a,b}$. Se $x^{s_1^{(p)}} y^{s_2^{(p)}}$ é o termo líder de $f^{(p)}$ (assume-se, sem perda de generalidade, que os coeficientes dos termos líderes são iguais a 1), então tem-se que

$$s_1^{(1)} > s_1^{(2)} > \dots > s_1^{(\nu)} = 0$$

e

$$0 = s_2^{(1)} < s_2^{(2)} < \dots < s_2^{(\nu)}.$$

Mostra-se que [11], se $l = \max\{a + b - 1, b\}$ (máximo inteiro l , tal que $(0, l) \leq_{\text{grev}} (a, b)$), então os polinômios mínimos gerados pelo algoritmo BMS para um arranjo S de síndromes de entrada $S_{i,j}$, com $i + j \leq l + \mu$, fornecem relações de recursão lineares válidas para todas as demais síndromes. Ou seja, obtidos os polinômios mínimos

válidos para o arranjo S até a entrada $S_{0,l+\mu}$, pode-se determinar pelas relações de recursão lineares correspondentes a estes polinômios as demais síndromes do arranjo S . Portanto, faz-se necessário apenas encontrar por meio do esquema de decisão por maioria as síndromes $S_{i,j}$, com $i + j \leq l + \mu$.

Outra observação importante com relação ao algoritmo BMS, já descrito na subseção 5.5.1, é que a cardinalidade do conjunto delta Δ obtido no fim do processamento é sempre igual ao número de erros ocorridos.

Decisão por maioria adaptada ao algoritmo BMS

Suponha que são conhecidas todas as síndromes $S_{i,j}$, em que $(i, j) <_{\text{grev}} (a, b) \leq_{\text{grev}} (0, l + \mu)$. Deseja-se, então, determinar a partir das síndromes conhecidas a síndrome $S_{a,b}$.

Considere $\mathcal{F}_{a,b} = \{f^{(1)}, \dots, f^{(p)}\}$ e suponha que, para algum $f^{(p)} \in \mathcal{F}_{a,b}$, existem $\alpha \geq 0$ e $\beta \geq 0$, tais que $\alpha + s_1^{(p)} = a$ e $\beta + s_2^{(p)} = b$. Pode-se, então, escrever a relação de recursão linear correspondente a $f^{(p)}$

$$\sum_{(i,j) <_{\text{grev}} (s_1^{(p)}, s_2^{(p)})} f_{i,j}^{(p)} S_{i+\alpha, j+\beta} = -S_{a,b}^{(p)}. \quad (5.28)$$

Observe que, se $f^{(p)}$ for consistente com a síndrome $S_{a,b}$, ou seja, se a relação de recursão linear correspondente a $f^{(p)}$ for válida até $S_{a,b}$, então conclui-se que $S_{a,b}^{(p)} = S_{a,b}$.

Considere também o caso em que existem $\alpha \geq 0$ e $\beta \geq 0$, tais que $\alpha + s_1^{(p)} = a + \mu$ e $\beta + s_2^{(p)} = b - \mu + 1$. Pode-se, daí, escrever a relação

$$\sum_{(i,j) <_{\text{grev}} (s_1^{(p)}, s_2^{(p)})} f_{i,j}^{(p)} S_{i+\alpha, j+\beta} - S_{a,b-\mu+2} = -S_{a+\mu, b-\mu+1}^{(p)}. \quad (5.29)$$

Da mesma forma, se $f^{(p)}$ for consistente com $S_{a+\mu, b-\mu+1}$, então segue da equação 5.27 que $S_{a+\mu, b-\mu+1}^{(p)} = S_{a+\mu, b-\mu+1}$.

O restante desta discussão é dedicada a observar os casos em que as equações 5.28 e 5.29 podem ser calculadas e compará-los com o número de casos em que os polinômios são consistentes. Nesta comparação do número de casos é que reside a mesma idéia do esquema de decisão por maioria apresentado na subseção anterior.

Considere (a, b) , em que $a < \mu$ (se $a \geq \mu$, então $S_{a,b}$ pode ser calculada pela equação 5.26) e $(a, b) \leq (0, l + \mu)$. Segue do algoritmo BMS que existe um inteiro p ,

com $1 \leq p \leq \nu$, tal que a equação 5.28 pode ser calculada. Por outro lado, mostra-se [19] que, se $l \geq 2\mu - 3$, então existe um inteiro p , com $1 \leq p \leq \nu$, tal que a equação 5.29 pode ser calculada.

Agora, considere

$$K_1 = \{(x, y) \mid 0 \leq x \leq a \text{ e } 0 \leq y \leq b\}$$

e

$$K_2 = \{(x, y) \mid 0 \leq x < \mu \text{ e } 0 \leq y \leq b - \mu + 1\},$$

e faça

$$K = K_1 \cup K_2.$$

Pode ocorrer de K_2 estar vazio, mas apenas no caso em que $l < 2\mu - 3$.

Considere

$$A_p = \{(x, y) \in K \mid x + s_1^{(p)} \leq a \text{ e } y + s_2^{(p)} \leq b\}$$

e

$$B_p = \{(x, y) \in K \mid x + s_1^{(p)} \leq a + \mu \text{ e } y + s_2^{(p)} \leq b - \mu + 1\},$$

e faça

$$K' = \left(\bigcup_{p=1}^{\nu} A_p \cup B_p \right) \setminus \Delta_{a,b}$$

($\Delta_{a,b}$ é o conjunto delta obtido do algoritmo BMS associado ao conjunto de polinômios mínimos $\mathcal{F}_{a,b}$).

Mostra-se que [19], se $a + b \leq l + \mu$ e $a < \mu$, então

$$|K| > 2 \left\lfloor \frac{d-1}{2} \right\rfloor \tag{5.30}$$

e

$$|K'| \geq |K| - 2|\Delta_{a,b}|, \tag{5.31}$$

em que $|\ast|$ representa a cardinalidade do conjunto \ast . Combinando estas duas inequações e considerando que ocorreram $t \leq \lfloor \frac{d-1}{2} \rfloor$ erros na palavra recebida e que $|\Delta_{a,b}| \leq t$, obtém-se que

$$|K'| \geq 1.$$

Agora, denote por $\gamma_1, \gamma_2, \dots, \gamma_\nu$ os diferentes valores obtidos usando a equação 5.28 ou a equação 5.29, e considere

$$K_q = \left(\bigcup_{S_{a,b}^{(p)} = \gamma_q} A_p \cup \bigcup_{S_{a+\mu, b-\mu+1}^{(p)} = \gamma_q} B_p \right) \setminus \Delta_{a,b}, \quad q = 1, \dots, \nu.$$

Tem-se, então, que

$$K' = \bigcup_{q=1}^{\nu} K_q.$$

Supondo-se que $\gamma_q \neq S_{a,b}$, para todo q , com $1 \leq q \leq \nu$, então o próximo conjunto delta fornecido pelo algoritmo BMS, $\Delta_{a,b}^+$, seria acrescido de K' [19]. Desta forma, pela equação 5.31,

$$\begin{aligned} |\Delta_{a,b}^+| &\geq |\Delta_{a,b}| + |K'| \geq |\Delta_{a,b}| + |K| - 2|\Delta_{a,b}| \Rightarrow \\ |\Delta_{a,b}^+| &\geq |K| - |\Delta_{a,b}| \geq d - t > t, \end{aligned}$$

o que contradiz o fato de que $|\Delta_{a,b}^+| \leq t$.

Isto significa que pelo menos um dos γ_q 's é igual a $S_{a,b}$. Considerando, por exemplo, que apenas $\gamma_1 = S_{a,b}$, então o conjunto delta seria acrescido de

$$K'_1 = \bigcup_{q=2}^{\nu} K_q$$

e, portanto, $|\Delta_{a,b}| + |K'_1| \leq t$. Então, das equações 5.30 e 5.31, tem-se que

$$|K'_1| \leq t - |\Delta_{a,b}| < \frac{d}{2} - |\Delta_{a,b}| \leq \frac{1}{2} |K'|.$$

Portanto,

$$|K_1| > \frac{1}{2} |K'|.$$

Isto, em geral, implica que, se

$$K'_\eta = \bigcup_{q \neq \eta} K_q,$$

então, para os valores corretos de $S_{a,b}$, tem-se $|K'_\eta| < \frac{1}{2}|K'|$, e para todos os demais, tem-se $|K'_\eta| > \frac{1}{2}|K'|$. Como $K'_\eta \supseteq K_\eta$, conclui-se que o valor correto de $S_{a,b}$ apresenta um $|K'_\eta|$ mínimo ou, equivalentemente, um $|K_\eta|$ máximo.

Estes argumentos mostram que a síndrome $S_{a,b}$ pode ser determinada pelo seguinte procedimento:

1. Use a equação 5.28, ou a equação 5.29, para se obter os valores $\gamma_1, \gamma_2, \dots, \gamma_\nu$;
2. Determine

$$K_\eta = \left(\bigcup_{S_{a,b}^{(p)} = \gamma_\eta} A_p \cup \bigcup_{S_{a+\mu, b-\mu+1}^{(p)} = \gamma_\eta} B_p \right) \setminus \Delta_{a,b}$$

para cada γ_η obtido;

3. Faça η igual ao valor para o qual $|K_\eta|$ é máximo. Tem-se, então, que $S_{a,b} = \gamma_\eta$.

Este procedimento pode ainda ser continuado, de modo a se obter todas as síndromes $S_{i,j}$, com $i, j < q - 1$. Sabe-se que, após o processamento, o conjunto \mathcal{F} obtido constitui uma base Gröbner mínima para o ideal dos polinômios localizadores de erros, cujos zeros coincidem com as posições dos erros ocorridos.

Algoritmo de Sakata et al

O algoritmo 37 apresenta o esquema de decodificação discutido nesta subseção.

Algoritmo 37 (De Sakata, Justesen, Madelung, Jensen e Høholdt)

Entradas:

- Uma palavra recebida v ;
- Uma base $\{g_{a_1, b_1}, \dots, g_{a_k, b_k}\}$ para o espaço ortogonal ao código C .

Saídas:

- O vetor e de erros ocorridos.

<<< Passo 1 >>>

Calcular as primeiras síndromes a partir da base $\{g_{a_1, b_1}, \dots, g_{a_k, b_k}\}$ ortogonal ao código e da equação 5.26, e gerar o arranjo S .

Processar o algoritmo BMS para o arranjo S de entrada.

<<< Passo 2 >>>

Determinar as síndromes $S_{i,j}$, com $i + j \leq l + \mu$, utilizando o esquema de decisão por maioria descrito e o conjunto de polinômios mínimos $\mathcal{F}_{i,j}$ fornecido pelo algoritmo BMS.

Processar o algoritmo BMS após o cálculo de cada síndrome.

<<< Passo 3 >>>

A partir da equação 5.26 e das recursões fornecidas pelo conjunto $\mathcal{F}_{0, l+\mu} = \{f^{(1)}, \dots, f^{(v)}\}$ correspondente ao arranjo S até a entrada $S_{0, l+\mu}$, determinar todas as demais síndromes desconhecidas do arranjo $S_{q-1 \times q-1}$.

<<< Passo 4 >>>

Usar a transformada inversa de Fourier (equação 5.25) para se obter o vetor erro e ocorrido.

Este algoritmo é capaz de corrigir até a metade da distância mínima do código. Sua complexidade é determinada pelo passo 2. No caso em que o comprimento do código é $n = r^3$, tem-se uma complexidade $\mathcal{O}(n^{\frac{7}{3}})$. A versão original do decodificador com decisão por maioria de Feng e Rao, que não utiliza o algoritmo BMS, apresenta complexidade maior, $\mathcal{O}(n^3)$.

5.5.4 Exemplo de decodificação de um código de Hermite

Como nos exemplo anteriores, considere o código de Hermite C_{18} ($m = 18$) e denote-o por C . A curva de Hermite \mathcal{X} usada na construção deste código é dada pela equação

$$\mathcal{X} : y^3 + y = x^4.$$

O gênero de \mathcal{X} é $g = 3$. O corpo utilizado é o \mathbb{F}_9 (veja tabela 4.1 na subseção 4.2.3). Os pontos racionais de \mathcal{X} com as correspondentes posições na palavra código estão ilustrados na tabela 5.8.

Posição	Ponto	Posição	Ponto	Posição	Ponto
1	$(1, \alpha^4)$	10	$(\alpha^3, 1)$	19	(α^6, α^4)
2	$(1, \alpha^5)$	11	(α^3, α)	20	(α^6, α^5)
3	$(1, \alpha^7)$	12	(α^3, α^3)	21	(α^6, α^7)
4	$(\alpha, 1)$	13	(α^4, α^4)	22	$(\alpha^7, 1)$
5	(α, α)	14	(α^4, α^5)	23	(α^7, α)
6	(α, α^3)	15	(α^4, α^7)	24	(α^7, α^3)
7	(α^2, α^4)	16	$(\alpha^5, 1)$	25	$(0, \alpha^2)$
8	(α^2, α^5)	17	(α^5, α)	26	$(0, \alpha^6)$
9	(α^2, α^7)	18	(α^5, α^3)	27	$(0, 0)$

Tabela 5.8: Pontos racionais da curva \mathcal{X} em \mathbb{F}_9 e respectivas posições na palavra código.

A base para o espaço $L(13Q)$ ortogonal ao código C é (equação 4.12)

$$B_{13} = \left\{ \underset{1}{1}, \underset{2}{x}, \underset{3}{x^2}, \underset{4}{x^3}, \underset{5}{x^4}, \underset{6}{y}, \underset{7}{xy}, \underset{8}{x^2y}, \underset{9}{x^3y}, \underset{10}{y^2}, \underset{11}{xy^2} \right\}.$$

Considere a mesma palavra recebida dos exemplos anteriores

$$v = \left(1 \quad \alpha^6 \quad 0 \quad 0 \quad 0 \quad \dots \quad 0 \quad \alpha^5 \quad 0 \quad 0 \quad 0 \right),$$

em que se observam $t = 3$ erros nas posições 1, 2 e 24 (pontos $E_1 = (1, \alpha^4)$, $E_2 = (1, \alpha^5)$ e $E_3 = (\alpha^7, \alpha^3)$).

Procedendo com o passo 1 do algoritmo 37, obtém-se o conjunto de síndromes S de entrada descrito na tabela 5.9, que foi utilizado no exemplo do algoritmo BMS da subseção 5.5.1. Todas as síndromes foram obtidas das funções da base B_{13} , com exceção da síndrome $S_{0,3}$, que foi obtida usando-se a equação 5.26.

As saídas do algoritmo BMS aplicado a este arranjo S de entrada estão descritas nas tabelas 5.6 e 5.7, no exemplo da subseção 5.5.1. O conjunto de polinômios mínimos obtido após o processamento do passo 1 é

$$\mathcal{F}_{3,1} = \{x^2 + \alpha^2x + \alpha^7, xy + \alpha x^2 + \alpha^4y + \alpha^5, y^2 + \alpha xy + y + \alpha\}.$$

Observe agora, por exemplo, a determinação da síndrome $S_{2,2}$, a primeira síndrome desconhecida. Como $a = 2 < \mu = 4$, a entrada $S_{2,2}$ pode ser obtida pelo procedimento de decisão por maioria descrito. Da equação 5.28 e usando-se os polinômios de $\mathcal{F}_{3,1}$,

		j						
		0	1	2	3	4	5	6
i	0	0	α^3	α^2	1	*	*	...
	1	α^6	α^4	α^5	*	*	...	
	2	α^4	0	*	*	...		
	3	1	α^7	*	...			
	4	α^5	*	⋮				
	5	*	⋮					
	6	⋮						

Tabela 5.9: Arranjo S bi-dimensional das síndromes da palavra recebida v obtido no passo 1 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt. Os elementos * representam síndromes desconhecidas.

obtem-se

$$\begin{aligned}
 -\gamma_1 &= \alpha^2 S_{1,2} + \alpha^7 S_{0,2} = \alpha^2 \alpha^5 + \alpha^7 \alpha^2 = \alpha^2 \Rightarrow \\
 \gamma_1 &= \alpha^6, \\
 -\gamma_2 &= \alpha S_{3,1} + \alpha^4 S_{1,2} + \alpha^5 S_{1,1} = \alpha \alpha^7 + \alpha^4 \alpha^5 + \alpha^5 \alpha^4 = \alpha^2 \Rightarrow \\
 \gamma_2 &= \alpha^6, \\
 -\gamma_3 &= \alpha S_{3,1} + S_{2,1} + \alpha S_{2,0} = \alpha \alpha^7 + 0 + \alpha \alpha^4 = \alpha^2 \Rightarrow \\
 \gamma_3 &= \alpha^6.
 \end{aligned}$$

Daí, tem-se que

$$\begin{aligned}
 A_1 &= \{(0, 0), (0, 1), (0, 2)\} \text{ e } B_1 = \emptyset, \\
 A_2 &= \{(0, 0), (1, 0), (0, 1), (1, 1)\} \text{ e } B_2 = \{(0, 0), (1, 0), (2, 0)\}, \\
 A_3 &= \{(0, 0), (1, 0), (2, 0)\} \text{ e } B_3 = \{(0, 0), (1, 0), (0, 1), (2, 0), (1, 1), (2, 1)\}.
 \end{aligned}$$

Como $\Delta_{3,1} = \{(1, 0), (0, 1)\}$, tem-se, então, que

$$K_1 = K_2 = K_3 = \{(2, 0), (1, 1), (0, 2), (2, 1)\}.$$

Portanto, $S_{2,2} = \alpha^6$. Observe que estes últimos passos não seriam necessários, uma vez que $\gamma_1 = \gamma_2 = \gamma_3$.

		j							
		0	1	2	3	4	5	6	7
i	0	0	α^3	α^2	1	α^4	α^6	α^7	0
	1	α^6	α^4	α^5	α^6	α^5	α^4	α^6	*
	2	α^4	0	α^6	α^2	0	1	*	*
	3	1	α^7	0	α^7	1	*	*	⋮
	4	α^5	α^5	α	α	*	*	⋮	
	5	α^7	α	α^7	*	*	⋮		
	6	α^2	α^6	*	*	⋮			
	7	α^3	*	*	⋮				

Tabela 5.10: Arranjo S bi-dimensional de síndromes obtido no passo 2 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt após utilização do esquema de decisão por maioria. Os elementos * representam síndromes desconhecidas.

A tabela 5.10 descreve o arranjo S de síndromes obtido após o processamento de todo o passo 2 do algoritmo 37.

A tabela 5.11 descreve os conjuntos \mathcal{F} , \mathcal{G} e Δ , saídas do algoritmo BMS para cada iteração processada nos passos 1 e 2 do algoritmo 37. Observe que, neste caso, o conjunto de polinômios mínimos não se alterou durante o passo 2.

A tabela 5.12 descreve os polinômios envolvidos no processamento do algoritmo BMS, com as extensões e discrepâncias correspondentes. Como o conjunto \mathcal{F} não foi alterado durante o passo 2 do algoritmo 37, esta tabela é igual à tabela 5.7 do exemplo da subseção 5.5.1.

No passo 3, usando-se a equação 5.26 e as relações de recursão lineares fornecidas pelos polinômios de $\mathcal{F}_{0,7} = \{f_8, f_9, f_{11}\}$, a saber

$$\begin{aligned} f_8 : -S_{a,b} &= \alpha^2 S_{a-1,b} + \alpha^7 S_{a-2,b}, \\ f_9 : -S_{a,b} &= \alpha S_{a+1,b-1} + \alpha^4 S_{a-1,b} + \alpha^5 S_{a-1,b-1} \text{ e} \\ f_{11} : -S_{a,b} &= \alpha S_{a+1,b-1} + S_{a,b-1} + \alpha S_{a,b-2}, \end{aligned}$$

é possível se determinar as demais síndromes do arranjo S . A tabela 5.13 ilustra o conjunto completo das síndromes.

Por fim, aplicando-se a transformada inversa de Fourier (equação 5.25) ao arranjo S da tabela 5.13 (passo 4), pode-se determinar o vetor erro e ocorrido. Por exemplo,

$\alpha = (i, j)$	\mathcal{F}	\mathcal{G}	Δ
(0, 0)	$\{f_0\}$	\emptyset	\emptyset
(1, 0)	$\{f_1, f_2\}$	$\{f_0\}$	$\{(1, 0)\}$
(0, 1)	$\{f_3, f_2\}$		
(2, 0)	$\{f_3, f_4\}$		
(1, 1)	$\{f_5, f_4\}$		
(0, 2)	$\{f_4, f_6, f_7\}$	$\{f_0, f_5\}$	$\{(1, 0), (0, 1)\}$
(3, 0)	$\{f_8, f_6, f_7\}$		
(2, 1)	$\{f_8, f_6, f_7\}$		
(1, 2)	$\{f_8, f_9, f_{10}\}$		
(0, 3)	$\{f_8, f_9, f_{11}\}$		
\vdots	\vdots		
(0, 7)	$\{f_8, f_9, f_{11}\}$		

Tabela 5.11: Saídas do algoritmo BMS durante o processamento dos passos 1 e 2 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt.

para a posição 2 (ponto $(1, \alpha^5)$), obtém-se que

$$\begin{aligned}
 e_2 &= \sum_{i,j < 8} S_{i,j} (1)^{-i} (\alpha^5)^{-j} \\
 &= \sum_{i=0}^7 1^{-i} \sum_{j=0}^7 \alpha^{-5j} S_{i,j} \\
 &= 1 \cdot \alpha^2 + 1 \cdot \alpha^2 + 1 \cdot \alpha^2 + 1 \cdot \alpha^2 \\
 &\quad + 1 \cdot \alpha^2 + 1 \cdot \alpha^2 + 1 \cdot \alpha^2 + 1 \cdot \alpha^2 \\
 &= \alpha^6.
 \end{aligned}$$

Polinômios f_i	Span (f_i)	δ_{f_i}
$f_0(x, y) = 1$	(1, 0)	α^6
$f_1(x, y) = y$	(0, 0)	α^3
$f_2(x, y) = x^2$	(0, 0)	α^4
$f_3(x, y) = y + \alpha x$	(1, 0)	α^3
$f_4(x, y) = x^2 + \alpha^2 x$	(1, 0)	α
$f_5(x, y) = y + \alpha x + \alpha$	(0, 1)	α
$f_6(x, y) = xy + \alpha x^2 + \alpha x$	(0, 1)	α
$f_7(x, y) = y^2 + \alpha xy + \alpha y + \alpha^7 x$	(1, 0)	α^4
$f_8(x, y) = x^2 + \alpha^2 x + \alpha^7$	-	-
$f_9(x, y) = xy + \alpha x^2 + \alpha^4 y + \alpha^5$	-	-
$f_{10}(x, y) = y^2 + \alpha xy + \alpha y + \alpha^7 x + \alpha^2$	(0, 1)	α^7
$f_{11}(x, y) = y^2 + \alpha xy + y + \alpha$	-	-

Tabela 5.12: Polinômios envolvidos no processamento do algoritmo BMS nos passos 1 e 2 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt.

		j							
		0	1	2	3	4	5	6	7
i	0	0	α^3	α^2	1	α^4	α^6	α^7	0
	1	α^6	α^4	α^5	α^6	α^5	α^4	α^6	α^3
	2	α^4	0	α^6	α^2	0	1	α^5	α
	3	1	α^7	0	α^7	1	α^5	α^4	α^5
	4	α^5	α^5	α	α	α^6	α^7	α^3	α^2
	5	α^7	α	α^7	α^4	α^2	α^2	α^2	α^4
	6	α^2	α^6	α^3	α^5	α^7	α^3	α	α^7
	7	α^3	1	1	0	α	0	1	1

Tabela 5.13: Arranjo S bi-dimensional de síndromes completo obtido após o passo 3 do algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt.

Capítulo 6

Conclusões

Apesar de incipiente, o estudo da teoria dos códigos de geometria algébrica e o desenvolvimento de algoritmos de decodificação para estes códigos em muito já evoluiu. Foram dezenas de trabalhos publicados envolvendo este assunto em todo o mundo. Trabalhos importantes, como os de Skorobogatov, Vlăduț, Porter, Shen, Tzeng, Sakata, van Lint, Jensen, Høholdt, Feng, Rao, Stichtenoth, além de outros, que apresentam em sua grande maioria variações dos esquemas discutidos na presente dissertação. Em contraste com esta realidade, observa-se que pouco tem sido feito nesta área a nível de Brasil.

Em vista disto, determinou-se como objetivo central desta dissertação o estudo da teoria dos CGA's e a descrição, implementação e análise de seus esquemas de decodificação, no intuito de criar um suporte para posteriores trabalhos nesta área. A restrição do escopo desta análise aos códigos de Hermite se justifica não apenas nos excelentes parâmetros obtidos com tais códigos, mas principalmente nas simplificações que esta abordagem proporciona à descrição dos códigos e à compreensão de seus algoritmos de decodificação. Isto, porque é possível se obter o dual de um código de Hermite sem se recorrer à complicada construção através de resíduos de diferenciais.

No capítulo 1, foi feita uma contextualização e uma introdução geral ao tema do presente trabalho, seguida de uma rápida descrição do conteúdo dos capítulos subsequentes. No capítulo 2, abordou-se genericamente a questão da codificação para controle de erros, e definiu-se os códigos de Goppa e os códigos de Reed-Solomon de uma maneira que tornasse menos traumática, pode-se assim dizer, a conceituação dos CGA's, que posteriormente seriam descritos como uma generalização dos primeiros.

Esta abordagem aqui adotada, utilizada por van Lint [13] e por Blake et al [2], torna bem mais acessível o entendimento dos CGA's e de seus esquemas de decodificação, e constitui uma boa alternativa à abordagem utilizada por Stichtenoth [24], que solicita uma compreensão muito mais aprofundada da geometria algébrica.

No capítulo 3, foi feita uma rápida descrição dos conceitos relativos à geometria algébrica, necessários à discussão dos CGA's e seus algoritmos de decodificação. Vale ressaltar o espaço dedicado à descrição da teoria das bases de Gröbner, que tem proporcionado a obtenção de esquemas de decodificação bastante eficientes e elegantes, não só no caso dos CGA's (algoritmo BMS), mas também de outros códigos, como os BCH, os de Reed-Solomon [12] etc.

No capítulo 4, foi feita a descrição dos CGA's, tanto dos relativos à construção pela avaliação de funções racionais em pontos da curva algébrica, quanto dos relativos à construção através de resíduos de diferenciais, códigos estes duais dos primeiros. Esta segunda definição, apesar de envolver elementos mais abstratos da geometria algébrica, é bastante usada na descrição de algoritmos de decodificação, uma vez que é a matriz de paridade do código, definida como na primeira construção, que é necessária no processo de decodificação. Eis o porque das duas definições e da facilidade obtida com o uso de códigos autoduais, como os códigos de Hermite, que também são descritos neste capítulo 4.

No capítulo 5, foi discutido o problema da decodificação e, em seguida, feita uma classificação das abordagens utilizadas na construção de esquemas de decodificação. As duas abordagens apresentadas foram, então, exemplificadas com a descrição de dois algoritmos típicos (pode-se dizer até mesmo clássicos), o algoritmo básico de Skorobogatov e Vlăduț e o algoritmo de Porter. Por fim, a questão da decodificação rápida de CGA's foi abordada com a descrição do algoritmo BMS de Sakata, que envolve a teoria das bases de Gröbner, e do esquema de decisão por maioria de Feng e Rao. Ambos os esquemas foram utilizados no algoritmo de Sakata, Justesen, Madelung, Jensen e Høholdt descrito em seguida.

Além do presente texto, diversas implementações foram realizadas em linguagem de programação C++ de algoritmos de codificação e decodificação estudados. Para estas implementações, utilizou-se o compilador Borland C++ Builder 3.0 Client/Server, tendo, contudo, havido a preocupação de se escrever o código de modo a ser portátil, possibilitando sua utilização em qualquer ambiente ou plataforma. A intenção é a de

que, não só o presente texto, mas também estas implementações, sirvam como suporte a futuros desenvolvimentos. Neste sentido, todo o código C++ escrito foi devidamente comentado. Além disso, foi elaborada uma documentação eletrônica (arquivos texto), que deve auxiliar na utilização das rotinas em C++.

Para concluir, deve-se observar que, apesar de serem bastante simples os objetivos almejados com este trabalho, o grau de dificuldade imposto pela complexidade da teoria matemática envolvida e o seu pioneirismo no âmbito brasileiro o justificam e o validam. São estas referidas dificuldades, enfrentadas na construção da presente dissertação, que se buscou atenuar para as futuras investidas nesta área. Considera-se que estes objetivos simples, porém muito relevantes, tenham sido alcançados, e espera-se que novos trabalhos no âmbito do Brasil imprimam um passo além na utilização desta promissora teoria dos CGA's na construção de sistemas de comunicação digital mais eficientes.

Apêndice A

Álgebra de Corpo Finito

Os códigos de bloco, em geral, baseiam-se em estruturas de anéis de polinômios e nos sistemas aritméticos dos corpos de Galois. Estes e outros conceitos básicos da álgebra necessários ao entendimento dos esquemas de codificação e decodificação são descritos neste apêndice.

A álgebra apresenta três estruturas básicas, chamadas grupo, anel e corpo, a partir das quais toda a teoria é desenvolvida. Estas estruturas consistem em conjuntos de objetos matemáticos (como o dos números reais, dos números inteiros, etc.) associados a regras de relação entre seus elementos. As três primeiras seções deste apêndice descrevem formalmente estes conceitos. As demais seções tratam da aritmética de corpos de Galois. Maiores detalhes e provas dos resultados apresentados aqui podem ser obtidos em Blahut [1, pp. 65-92] ou qualquer texto básico sobre álgebra de corpo finito.

A.1 Grupos

Um *grupo* G é um conjunto associado a uma operação (denotada por "+") sobre pares de elementos deste conjunto que satisfaz as quatro propriedades seguintes:

1. *Fechamento* → Para todo $a, b \in G$, $c = a + b \in G$;
2. *Associatividade* → Para todo $a, b, c \in G$, $a + (b + c) = (a + b) + c$;
3. *Identidade* → Existe um elemento e chamado identidade, em que $a + e = e + a = a$;

4. *Inversas* \rightarrow Se $a \in G$, então existe $b \in G$ chamado inversa de a , em que $a + b = b + a = e$;

Os grupos com número finito de elementos são ditos *grupos finitos*, e seu número de elementos é chamado de *ordem do grupo*. A *ordem do elemento* $a \in G$ é o menor valor n , tal que

$$\underbrace{a + a + \cdots + a}_{n \text{ vezes}} = e \Rightarrow \underbrace{a + a + \cdots + a}_{n+1 \text{ vezes}} = a.$$

A ordem de um elemento sempre divide a ordem do grupo.

O grupo G que apresenta a propriedade de comutatividade, na qual, para $a, b \in G$, $a + b = b + a$, é dito *grupo comutativo* ou *grupo abeliano*.

Não se deve confundir a operação “+” com a adição convencional. Esta operação sobre os elementos do grupo pode ser definida de diferentes formas, desde que satisfaça as propriedades descritas.

Em todo grupo, o elemento identidade é único. Além disso, a inversa a^{-1} de cada elemento a do grupo é única, e $(a^{-1})^{-1} = a$.

A.2 Anéis

Um *anel* R é um conjunto com duas operações, *adição* (“+”) e *multiplicação* (justaposição), que satisfaz os seguintes axiomas:

1. R é um grupo abeliano sob a adição;
2. *Fechamento* \rightarrow Para todo $a, b \in R$, o produto $ab \in R$;
3. *Associatividade* \rightarrow Para todo $a, b, c \in R$, $a(bc) = (ab)c$;
4. *Distributividade* $\rightarrow a(b+c) = ab+ac$ e $(b+c)a = ba+ca$, para todo $a, b, c \in R$;

A adição em um anel é sempre comutativa. Um *anel comutativo* é aquele em que a multiplicação é comutativa, ou seja, em que $ab = ba$ para todo $a, b \in R$.

Um anel não necessariamente possui identidade na multiplicação, assim como inversas de seus elementos. Caso um anel tenha identidade, esta identidade é única. Além disso, se $ab = 1$ e $ca = 1$, então $b = c$, e a é dito ter uma única inversa denotada por a^{-1} .

Um elemento que possua inversa em um anel é dito uma *unidade*.

A.3 Corpos

Um *corpo* F é um conjunto que possui duas operações definidas sobre seus elementos, adição e multiplicação, e que satisfaz os seguintes axiomas:

1. É um grupo abeliano sob a adição;
2. É fechado sob a multiplicação, e $F - \{0\}$ é um grupo abeliano sob a multiplicação;
3. Para todo $a, b, c \in F$, $(a + b)c = ac + bc$;

Convenciona-se denotar por 0 a identidade sob a adição, por $-a$ a inversa aditiva de a , por 1 a identidade sob a multiplicação, e por a^{-1} a inversa multiplicativa de a . Entende-se por subtração $(a - b) = a + (-b)$ e por divisão $(\frac{a}{b}) = b^{-1}a$.

São exemplos de corpos os conjuntos \mathbb{R} (números reais), \mathbb{C} (números complexos) e \mathbb{Q} (números racionais). Um corpo com número finito de elementos é chamado *corpo finito*, ou *corpo de Galois*, e denotado por \mathbb{F}_q (corpo com q elementos).

Um subconjunto de um corpo F é dito *subcorpo* de F se constituir um corpo sob as operações inerentes a F . O corpo F é dito uma *extensão* deste subcorpo.

Um corpo comporta-se como um anel, que permite a divisão ou cancelamento. Em um corpo, se $ab = ac$ e $a \neq 0$, então $a^{-1}ab = a^{-1}ac \Rightarrow b = c$. Existem alguns anéis, como o anel dos inteiros, que permitem o cancelamento, mesmo não sendo corpos, ou seja, mesmo não existindo a inversa na multiplicação para todos os seus elementos. Um anel comutativo em que $b = c$ sempre que $ab = ac$, com $a \neq 0$, é chamado de *domínio integral*.

A.4 Corpos finitos baseados em anéis de inteiros

O conjunto \mathbb{Z} dos inteiros forma um domínio integral sob as operações de adição e multiplicação usuais, sendo denotado por \mathbb{Z} .

Um inteiro s é dito *divisível* pelo inteiro r , ou de modo igual r *divide* s , se $ra = s$ para algum inteiro a . Um inteiro p é dito *primo* se for divisível apenas por $\pm p$ ou ± 1 . O *máximo divisor comum* $MDC(r, s)$ de dois inteiros r e s é o maior inteiro positivo que divide ambos r e s . O *mínimo múltiplo comum* $MMC(r, s)$ de dois inteiros r e s é o menor inteiro positivo que é divisível por ambos r e s .

Em geral, a divisão não é possível em um anel. Pode-se, entretanto, definir uma divisão com resto e um cancelamento (resto igual a zero), razão pela qual o anel de inteiros constitui um domínio integral. O chamado *algoritmo da divisão* estabelece que, para todo par de inteiros c e d , com $d \neq 0$, existe um único par de inteiros Q (quociente) e s (resto), tal que $c = dQ + s$, em que $0 \leq s < |d|$. O resto s também pode ser escrito como $s = [c]_d$. Outra expressão comum é a de *congruência* $s \equiv c \pmod{d}$. Dizer que s é congruente a c módulo d significa que s e c possuem o mesmo resto na divisão por d , mas s não necessariamente é menor que d .

Considere q um inteiro positivo. O anel dos inteiros módulo q , denotado por \mathbb{Z}/q , é o conjunto $\{0, \dots, q-1\}$ associado à adição e à multiplicação definidos por $a + b = [a + b]_q$ e $ab = [ab]_q$.

Quando q for um inteiro primo, o anel \mathbb{Z}/q constituirá um corpo, sendo denotado por \mathbb{F}_q . Portanto, tomando-se q inteiro primo, pode-se obter um corpo \mathbb{F}_q a partir do anel de inteiros \mathbb{Z} associado à operação de módulo q .

A.5 Corpos finitos baseados em anéis de polinômios

Um polinômio em \mathbb{F}_q consiste numa expressão da forma

$$f(x) = f_{n-1}x^{n-1} + f_{n-2}x^{n-2} + \dots + f_1x + f_0,$$

em que x é uma variável e $f_{n-1}, \dots, f_0 \in \mathbb{F}_q$. O polinômio nulo é $f(x) = 0$. Um *polinômio mônico* é um polinômio no qual o coeficiente f_{n-1} é igual a 1. O grau $\deg f(x)$ de um polinômio não nulo $f(x)$ é o índice do coeficiente f_{n-1} .

O conjunto de todos os polinômios em \mathbb{F}_q associado à adição e à multiplicação de polinômios (com adição e multiplicação dos coeficientes em \mathbb{F}_q) constitui um anel, denotado por $\mathbb{F}_q[x]$.

Um polinômio $s(x)$ é *divisível* por $r(x)$, ou $r(x)$ *divide* $s(x)$, se existir um polinômio $a(x)$, tal que $s(x) = r(x)a(x)$. Um polinômio $p(x)$ é dito *irredutível* se for divisível apenas por $\alpha p(x)$ ou α , em que $\alpha \in \mathbb{F}_q$. Um polinômio mônico irredutível não nulo é dito um *polinômio primo*. O *máximo divisor comum* $MDC[r(x), s(x)]$ é o polinômio mônico de maior grau que divide ambos $r(x)$ e $s(x)$. O *mínimo múltiplo comum* $MMC[r(x), s(x)]$ é o polinômio mônico de menor grau que é divisível por ambos $r(x)$ e $s(x)$.

O chamado *algoritmo da divisão de polinômios* determina que, para todo par de polinômios $c(x)$ e $d(x)$, com $d(x) \neq 0$, existe um único par de polinômios $Q(x)$ (quociente) e $s(x)$ (resto), tal que $c(x) = d(x)Q(x) + s(x)$, em que $\deg s(x) < \deg d(x)$. O resto $s(x)$ também pode ser escrito como $s(x) = [c(x)]_{d(x)}$. Assim como no caso dos anéis de inteiros, a congruência $s(x) \equiv c(x) \pmod{d(x)}$ indica que $s(x)$ e $c(x)$ possuem o mesmo resto na divisão por $d(x)$.

Assim como em certos casos é útil expressar inteiros como produto de inteiros primos (fatoração), também o é no caso de polinômios. Um polinômio não nulo $p(x)$ em um corpo \mathbb{F}_q possui uma fatoração única (exceto pela ordem dos fatores) em um produto de um escalar (elemento de \mathbb{F}_q) e de polinômios primos em \mathbb{F}_q .

Um polinômio em \mathbb{F}_q pode ser avaliado em qualquer elemento β de \mathbb{F}_q substituindo-se a variável x por β . Um elemento β é um zero de ordem m de um polinômio $p(x)$ se e só se $(x - \beta)^m$ dividir $p(x)$ e $(x - \beta)^{m+1}$ não dividi-lo. Além disso, um polinômio $p(x)$ de grau n possui no máximo n zeros.

Para qualquer polinômio mônico $p(x)$ em \mathbb{F}_q de grau não nulo, o *anel de polinômios módulo $p(x)$* é o conjunto de todos os polinômios com grau menor que $\deg p(x)$, associado à adição e multiplicação de polinômios módulo $p(x)$. Este anel é denotado por $\mathbb{F}_q[x]/p(x)$. Quando $p(x)$ for um polinômio primo de grau n , o anel $\mathbb{F}_q[x]/p(x)$ será um corpo com q^n elementos, chamado *corpo de Galois* e denotado por \mathbb{F}_{q^n} .

Um *elemento primitivo* α do corpo \mathbb{F}_q é tal que todo elemento não nulo de \mathbb{F}_q pode ser expresso como potência de α . Os elementos primitivos são úteis na construção de corpos, já que, conhecido um elemento primitivo, é possível construir-se todo o corpo e sua tabela de multiplicação através das potências deste elemento (pode-se mostrar que todo corpo de Galois possui um elemento primitivo).

Um *polinômio primitivo* $p(x)$ em \mathbb{F}_q consiste num polinômio primo em \mathbb{F}_q , tal que o elemento representado por x é primitivo no corpo extensão construído módulo $p(x)$. Existem polinômios primitivos de todos os graus em todos os corpos de Galois. Além disso, mostra-se que um elemento primitivo de um corpo é um zero de qualquer polinômio primitivo neste corpo.

Um corpo \mathbb{F}_q é dito *fechado algebricamente* se todo polinômio $f(x) \in \mathbb{F}_q[x]$ de grau maior ou igual a 1 tiver raízes em \mathbb{F}_q . Qualquer corpo \mathbb{F}_q possui uma extensão $\overline{\mathbb{F}_q}$ fechada algebricamente. Esta extensão $\overline{\mathbb{F}_q}$ é dita o *fecho algébrico* de \mathbb{F}_q .

Apêndice B

Dedução do Limite de Gilbert-Varshamov

Este apêndice descreve a dedução do limite de Gilbert-Varshamov apresentado no capítulo 2 (equação 2.5), repetido a seguir por conveniência.

Teorema 38 (Limite de Gilbert-Varshamov) *Considere que $0 \leq \delta \leq \frac{q-1}{q}$, então*

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

em que q é o tamanho do corpo finito \mathbb{F}_q usado, $\delta = \frac{d}{n}$ é a distância mínima relativa do código C de comprimento n e distância mínima d , e H_q é a função entropia definida por

$$H_q(x) = \begin{cases} 0 & , x = 0; \\ x \log_q(q-1) - x \log_q x - (1-x) \log_q(1-x) & , 0 < x < \frac{q-1}{q}. \end{cases} \quad (\text{B.1})$$

Este teorema estabelece um limite inferior para a taxa de informação assintótica $R(\delta)$ para códigos com distância mínima relativa δ . A prova descrita a seguir está dividida em três partes.

B.1 Primeira parte

Considere inicialmente a desigualdade de Stirling [15, p. 178], dada por

$$\ln(n!) - \frac{1}{12n} \leq \left(n + \frac{1}{2}\right) \ln(n) - n + \frac{\ln(2\pi)}{2} \leq \ln(n!),$$

em que, mudando os logaritmos para base q , tem-se

$$\log_q(n!) - \frac{\log_q(e)}{12n} \leq \left(n + \frac{1}{2}\right) \log_q(n) - n \log_q(e) + \frac{\log_q(2\pi)}{2} \leq \log_q(n!). \quad (\text{B.2})$$

Considere esferas de raio d de cardinalidade $V(n, d)$. Da equação 2.1 de $V(n, d)$ (repetida abaixo), observe a seguinte manipulação matemática:

$$\begin{aligned} V(n, d) &= \sum_{i=0}^d \binom{n}{i} (q-1)^i \Rightarrow \\ V(n, d) &\geq \binom{n}{d} (q-1)^d \\ &= \frac{n!}{d!(n-d)!} (q-1)^d. \end{aligned} \quad (\text{B.3})$$

Aplicando o logaritmo de base q à equação acima, obtém-se

$$\log_q[V(n, d)] \geq \log_q(n!) - \log_q(d!) - \log_q[(n-d)!] + d \log_q(q-1).$$

Usando, então, a equação B.2, obtém-se

$$\begin{aligned} \log_q[V(n, d)] &\geq \left(n + \frac{1}{2}\right) \log_q(n) - n \log_q(e) + \frac{\log_q(2\pi)}{2} \\ &\quad - \left(d + \frac{1}{2}\right) \log_q(d) + d \log_q(e) - \frac{\log_q(2\pi)}{2} \\ &\quad - \left(n - d + \frac{1}{2}\right) \log_q(n - d) + (n - d) \log_q(e) - \frac{\log_q(2\pi)}{2} \\ &\quad + d \log_q(q-1) - \frac{\log_q(e)}{12d} - \frac{\log_q(e)}{12(n-d)} \\ &= \left(n + \frac{1}{2}\right) \log_q(n) - \left(d + \frac{1}{2}\right) \log_q(d) \\ &\quad - \left(n - d + \frac{1}{2}\right) \log_q(n - d) + d \log_q(q-1) \\ &\quad - \frac{\log_q(2\pi)}{2} - \frac{\log_q(e)}{12d} - \frac{\log_q(e)}{12(n-d)} \end{aligned}$$

Dividindo agora tudo por n , fazendo n tender para infinito e eliminando os termos que tendem a zero, obtém-se que

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{\log_q V(n, d)}{n} &\geq \lim_{n \rightarrow \infty} \left[\log_q(n) - \frac{d}{n} \log_q(d) \right. \\ &\quad \left. - \frac{(n-d)}{n} \log_q(n-d) + \frac{d}{n} \log_q(q-1) \right]. \end{aligned}$$

Agora, considerando d o maior valor possível que satisfaça $d \leq \delta n$, observa-se que quando n tende para infinito, tanto $\frac{\delta n}{d}$ quanto $\frac{(1-\delta)n}{n-d}$ tendem para 1. Então,

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{\log_q V(n, \delta n)}{n} &\geq \lim_{n \rightarrow \infty} [\log_q(n) - \delta \log_q(\delta n) \\ &\quad - (1-\delta) \log_q[n(1-\delta)] + \delta \log_q(q-1)] \\ &= \lim_{n \rightarrow \infty} [\log_q(n) - \delta \log_q(\delta) - \delta \log_q(n) \\ &\quad - (1-\delta) \log_q(1-\delta) - (1-\delta) \log_q(n) + \delta \log_q(q-1)] \\ &= \lim_{n \rightarrow \infty} [-\delta \log_q(\delta) - (1-\delta) \log_q(1-\delta) + \delta \log_q(q-1)]. \end{aligned}$$

Da equação B.1, tem-se então que

$$\lim_{n \rightarrow \infty} \frac{\log_q V(n, \delta n)}{n} \geq H_q(\delta). \quad (\text{B.4})$$

B.2 Segunda parte

Como por hipótese $0 \leq \delta \leq \frac{q-1}{q}$, observe que

$$\begin{aligned} 0 \leq \frac{1}{q} = 1 - \frac{(q-1)}{q} &\leq 1 - \delta \Rightarrow \\ \frac{1}{q} &\leq 1 - \delta \Rightarrow \\ \frac{q-1}{q} &\leq (q-1)(1-\delta). \end{aligned}$$

Portanto, para qualquer $k \geq 0$, tem-se

$$\delta^k \leq \frac{(q-1)^k}{q^k} \leq (q-1)^k (1-\delta)^k.$$

Fazendo $k = \delta n - i$, para $0 \leq i \leq \delta n$, tem-se

$$\begin{aligned} \delta^{\delta n - i} &\leq (q-1)^{\delta n - i} (1-\delta)^{\delta n - i} \Rightarrow \\ \frac{\delta^{\delta n}}{\delta^i} &\leq \frac{(q-1)^{\delta n} (1-\delta)^{\delta n}}{(q-1)^i (1-\delta)^i}. \end{aligned}$$

Multiplicando a equação por $(1 - \delta)^n$, obtém-se

$$\begin{aligned} \frac{\delta^{\delta n} (1 - \delta)^n}{\delta^i} &\leq \frac{(q - 1)^{\delta n} (1 - \delta)^{\delta n} (1 - \delta)^n}{(q - 1)^i (1 - \delta)^i} \Rightarrow \\ \frac{\delta^{\delta n} (1 - \delta)^n}{(q - 1)^{\delta n} (1 - \delta)^{\delta n}} &\leq \frac{\delta^i (1 - \delta)^n}{(q - 1)^i (1 - \delta)^i} \Rightarrow \\ \frac{\delta^{\delta n} (1 - \delta)^{n - \delta n}}{(q - 1)^{\delta n}} &\leq \frac{\delta^i (1 - \delta)^{n - i}}{(q - 1)^i} \end{aligned} \quad (\text{B.5})$$

Agora, tomando a entropia (equação B.1) $H_q(\delta)$ multiplicada por n e aplicando o anti-logaritmo na base q , obtém-se

$$\begin{aligned} q^{nH_q(\delta)} &= \frac{(q - 1)^{\delta n}}{\delta^{\delta n} (1 - \delta)^{(1 - \delta)n}} \Rightarrow \\ q^{-nH_q(\delta)} &= \frac{\delta^{\delta n} (1 - \delta)^{n - \delta n}}{(q - 1)^{\delta n}} \end{aligned} \quad (\text{B.6})$$

Escrevendo $1 = 1^n = [\delta + (1 - \delta)]^n$ e fazendo a expansão binomial, obtém-se

$$\begin{aligned} 1 = 1^n &= [\delta + (1 - \delta)]^n \\ &= \sum_{i=0}^n \binom{n}{i} \delta^i (1 - \delta)^{n-i} \\ &\geq \sum_{i=0}^d \binom{n}{i} (q - 1)^i \frac{\delta^i (1 - \delta)^{n-i}}{(q - 1)^i} \quad (\text{usando equação B.5}) \\ &\geq \sum_{i=0}^d \binom{n}{i} (q - 1)^i \frac{\delta^{\delta n} (1 - \delta)^{n - \delta n}}{(q - 1)^{\delta n}} \quad (\text{usando equações B.3 e B.6}) \\ &= V(n, d) q^{-nH(\delta)}. \end{aligned}$$

Aplicando, então, o logaritmo na base q a este resultado, tem-se que

$$\log_q V(n, d) \leq nH_q(\delta). \quad (\text{B.7})$$

B.3 Terceira parte

Considerando as equações B.4 e B.7, obtém-se que

$$\lim_{n \rightarrow \infty} \frac{\log_q V(n, \delta n)}{n} = H_q(\delta). \quad (\text{B.8})$$

Se um código C possui uma distância mínima d e uma cardinalidade $V(n, d)$ mínima para esta distância, então todas as palavras do espaço \mathbb{F}_q^n possuem distância menor que d para alguma palavra código. Então,

$$\begin{aligned} |C| \cdot V_q(n, d-1) &\geq q^n \Rightarrow \\ |C| \cdot V_q(n, d) &\geq q^n. \end{aligned}$$

Considere a notação $A(n, d)$ para o maior valor de $q^k = |C|$ para o qual o código (n, k, d) existe. Tem-se, então, que

$$\begin{aligned} A(n, d) V(n, d) &\geq q^n \Rightarrow \\ \log_q [A(n, d)] + \log_q [V(n, d)] &\geq n \Rightarrow \\ \frac{\log_q [A(n, d)]}{n} &\geq 1 - \frac{\log_q [V(n, d)]}{n} \end{aligned}$$

Agora, considerando d o maior valor possível que satisfaça $d \leq \delta n$ e aplicando o limite quando n tende para infinito, obtém-se por fim, da equação B.8, que¹

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{\log_q [A(n, \delta n)]}{n} &\geq 1 - \lim_{n \rightarrow \infty} \frac{\log_q [V(n, \delta n)]}{n} \Rightarrow \\ R(\delta) &\geq 1 - H_q(\delta). \end{aligned}$$

¹A taxa de informação assintótica é definida como sendo

$$R(\delta) = \lim_{n \rightarrow \infty} \frac{\log_q [A(n, \delta n)]}{n}.$$

Bibliografia

- [1] Richard E. Blahut. *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley, 1983.
- [2] Ian Blake, Chris Heegard, Tom Høholdt, e Victor Wei. Algebraic-Geometry Codes. *IEEE Transactions on Information Theory*, 44(6):2596–2618, October 1998.
- [3] David Cox, John Little, e Donal O’Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. New York: Springer-Verlag, 1992.
- [4] Gui-Liang Feng e T. R. N. Rao. Decoding of Algebraic Geometric Codes up to the Designed Minimum Distance. *IEEE Transactions on Information Theory*, 39(1):37–45, January 1993.
- [5] Gui-Liang Feng, Victor K. Wei, T. R. N. Rao, e Kenneth K. Tzeng. Simplified Understanding and Efficient Decoding of a Class of Algebraic-Geometric Codes. *IEEE Transactions on Information Theory*, 40(4):981–1002, July 1994.
- [6] Patrick Fitzpatrick. On the Key Equation. *IEEE Transactions on Information Theory*, 41(5):1290–1302, September 1995.
- [7] William Fulton. *Algebraic Curves: An Introduction to Algebraic Geometry*. Reading, MA: W. A. Benjamin, 1969.
- [8] V. D. Goppa. A New Class of Linear Error-Correcting Codes. *Problems of Information Theory*, 6:207–212, 1970.
- [9] Tom Høholdt e Ruud Pellikaan. On the Decoding of Algebraic-Geometric Codes. *IEEE Transactions on Information Theory*, 41(6):1589–1614, November 1995.

- [10] Jørn Justesen, Knud J. Larsen, H. Elbrønd Jensen, Allan Havemose, e Tom Høholdt. Construction and Decoding of a Class of Algebraic Geometric Codes. *IEEE Transactions on Information Theory*, 35(4):811–821, July 1989.
- [11] Jørn Justesen, Knud J. Larsen, H. Elbrønd Jensen, e Tom Høholdt. Fast Decoding of Codes from Algebraic Plane Curves. *IEEE Transactions on Information Theory*, 38(1):111–119, January 1992.
- [12] Leocarlos B. S. Lima e Francisco M. Assis. Decoding Algorithm for Reed-Solomon Codes Using the Method of Gröbner Basis. In *Proc. 2nd Conference on Telecommunications*, 1999.
- [13] Jacobus H. Van Lint. Algebraic Geometric Codes. In *Coding Theory and Design Theory (Part I)*, volume 21 of *IMA Volumes Math. Appl.*, pp. 137–162. Berlin: Springer-Verlag, 1990.
- [14] S. C. Porter, Ba-Zhong Shen, e Ruud Pellikaan. Decoding Geometric Goppa Codes Using an Extra Place. *IEEE Transactions on Information Theory*, 38(6):1663–1676, November 1992.
- [15] Oliver Pretzel. *Codes and Algebraic Curves*. New York: Oxford University Press, 1998.
- [16] John G. Proakis. *Digital Communications*. New York: McGraw-Hill, 1995.
- [17] Keith Saints e Chris Heegard. Algebraic-Geometric Codes and Multidimensional Cyclic Codes: A Unified Theory and Algorithms for Decoding Using Gröbner Bases. *IEEE Transactions on Information Theory*, 41(6):1733–1751, November 1995.
- [18] Shajiro Sakata. Extension of the Berlekamp-Massey Algorithm to N Dimensions. *Informat. Comput.*, 84:207–239, February 1990.
- [19] Shajiro Sakata, Jørn Justesen, Y. Madelung, H. Elbrønd Jensen, e Tom Høholdt. Fast Decoding of Algebraic-Geometric Codes up to the Designed Minimum Distance. *IEEE Transactions on Information Theory*, 41(5):1672–1677, September 1995.

- [20] Shojiro Sakata, Douglas A. Leonard, Helge Elbrond Jensen, e Tom Høholdt. Fast Erasure-and-Error Decoding of Algebraic Geometry Codes Up to the Feng-Rao Bound. *IEEE Transactions on Information Theory*, 44:1558–1565, July 1998.
- [21] Ba-Zhong Shen e Kenneth K. Tzeng. Decoding Geometric Goppa Codes up to Designed Minimum Distance by Solving a Key Equation in a Ring. *IEEE Transactions on Information Theory*, 41(6):1694–1702, November 1994.
- [22] Alexei N. Skorobogatov e Sergei G. Vlăduț. On the Decoding of Algebraic-Geometric Codes. *IEEE Transactions on Information Theory*, 36(5):1051–1060, September 1990.
- [23] Henning Stichtenoth. A Note on Hermitian Codes over $GF(q^2)$. *IEEE Transactions on Information Theory*, 34(5):1345–1348, September 1988.
- [24] Henning Stichtenoth. *Algebraic Function Fields and Codes*. Berlin: Springer-Verlag, 1993.
- [25] M. A. Tsfasman, S. G. Vlăduț, e T. Zink. Modular Curves, Shimura Curves and Goppa Codes, Better Than Varshamov-Gilbert Bound. *Math. Nachr.*, 104:13–28, 1982.
- [26] Jacobus H. van Lint e T. A. Springer. Generalized Reed-Solomon Codes from Algebraic Geometry. *IEEE Transactions on Information Theory*, 33(3):305–309, May 1987.
- [27] Stephen B. Wicker. *Error Control Systems for Digital Communication and Storage*. Englewood Cliffs, NJ: Prentice Hall, 1995.