5
Conclusão

Veremos que certos códigos de Goppa podem ser pensados como códigos de avaliação definidos sobre uma $\mathbb{F}_q$-álgebra, segundo a construção do capítulo 4.

Seja $\mathcal{X}$ uma curva algébrica definida sobre o corpo finito $\mathbb{F}_q$ de gênero g. Vamos denotar por $\mathbb{F}_q(\mathcal{X})$ o corpo de funções racionais da curva $\mathcal{X}$.

Sejam $P_1, ..., P_n, P$ pontos racionais distintos de $\mathcal{X}$. Considere o divisor D dado por $D = P_1 + \cdots + P_n$ e G outro divisor tal que $supp\, G \cap supp\, D = \emptyset$.

Como foi visto no capítulo 3,

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

é um espaço vetorial sobre $\mathbb{F}_q$ cuja dimensão é igual a

$$dim\ \mathcal{L}(G) = deg\ G + 1 - g + i(G),$$

pelo teorema de Riemann-Roch 3.1.15, onde $i(G)$ é o índice de especialidade do divisor G.

Considere a seguinte função de avaliação

$$\begin{array}{rccc} av\ : & \mathcal{L}(G) & \longrightarrow & \mathbb{F}_q^n \\ & f & \longmapsto & (f(P_1), ..., f(P_n)). \end{array}$$

Esta aplicação está bem definida pela escolha dos divisores D e G. O código $C_{\mathcal{L}}(D, G)$ é por definição a imagem de $\mathcal{L}(G)$ por esta aplicação e temos que

$$k + d \geq n + 1 - g,$$

onde k é a dimensão e d é a distância mínima do código $C_{\mathcal{L}}(D, G)$.

Se $G = mP$, então $C_{\mathcal{L}}(D, G)$ é chamado código geométrico de Goppa no ponto P.

Definição 5.1

$$m \text{ é dita uma lacuna de } P \iff \mathcal{L}((m-1)P) = \mathcal{L}(mP).$$

Caso contrário, m é dito uma não-lacuna em P.

O conjunto das não-lacunas em P forma um semigrupo numérico, pois se m e n são não-lacunas em P, então existem funções f e g em $\mathbb{F}_q(\mathcal{X})$ tais que $(f)_\infty = mP$ e $(g)_\infty = nP$ e logo tomando fg vemos que $(fg)_\infty = (m+n)P$, ou seja $m+n$ é uma não-lacuna em P.

Teorema 5.0.11 (Teorema das Lacunas de Weierstrass) *Sejam $\mathcal{X}$ uma curva algébrica sobre $\mathbb{F}_q$ de gênero $g > 0$, $\mathbb{F}_q(\mathcal{X})$ seu corpo de funções e P um ponto racional de $\mathcal{X}$. Então existem g lacunas $m_1 < ... < m_g$ de P tais que*

$$m_1 = 1 \quad e \quad m_g \leq 2g-1.$$

Prova:

Observemos primeiramente que toda lacuna de P é $\leq 2g-1$, pois se $m \geq 2g$, então pelo teorema de Riemann-Roch temos:

$$dim\ \mathcal{L}(mP) = deg\ mP + 1 - g > dim\ \mathcal{L}((m-1)P) = deg\ (m-1)P + 1 - g,$$

logo m é uma não-lacuna em P.

Considere a seguinte sequência de espaços vetoriais

$$\mathcal{L}(0) \subseteq \mathcal{L}(P) \subseteq \mathcal{L}(2P) \subseteq ... \subseteq \mathcal{L}((2g-1)P), \tag{5-1}$$

Temos que $dim\,\mathcal{L}(0) = 1$ e $dim\,\mathcal{L}((2g-1)P) = g$, novamente pelo teorema de Riemann-Roch.

Como vimos anteriormente, se A, B são dois divisores de F/K tais que $A \leq B$, então $\mathcal{L}(A) \subseteq \mathcal{L}(B)$ e $dim\,(\mathcal{L}(B)/\mathcal{L}(A)) \leq deg\,B - deg\,A$. Isto implica que

$$dim\,\mathcal{L}(mP) \leq dim\,\mathcal{L}((m-1)P) + 1$$

para qualquer $m \geq 0$. Em 5-1 temos $2g-1$ inclusões sendo que $g-1$ dentre elas devem ser inclusões estritas. Então temos que existem exatamente g números m tais que $1 \leq m \leq 2g-1$ e $\mathcal{L}((m-1)P) = \mathcal{L}(mP)$, que são precisamente as lacunas em P.

Finalmente devemos mostrar que 1 é uma lacuna. Suponhamos que 1 é uma não-lacuna de P. Como o conjunto das não-lacunas forma um semigrupo numérico, todo $n \in \mathbb{N}$ é uma não-lacuna, portanto não existem lacunas e logo $g = 0$. Mas isto é absurdo, já que $g > 0$.

□

Enumeremos o conjunto infinito de não-lacunas em P em ordem crescente:

$$(\rho_l \mid l \in \mathbb{N}).$$

Tomemos

$$R = \bigcup_{m=0}^{\infty} \mathcal{L}(mP).$$

Claramente R é uma anel com unidade, mais ainda é uma $\mathbb{F}_q$-álgebra.

Consideremos agora a seguinte função

$$\begin{array}{rccc} \rho & : & R & \longrightarrow & \mathbb{N} \cup \{-\infty\} \\ & & f & \longmapsto & -v_P(f). \end{array}$$

onde v_P é a valorização associada a P.

Pelas propriedades da valorização já vistas, concluímos que ρ é uma função peso (capítulo 4). E ainda, a imagem de ρ é exatamente o conjunto das não-lacunas de P, isto é, para todo l existe uma função $f_l \in R$ tal que $\rho(f_l) = \rho_l$. Dessa forma, temos que $(f_l \mid l \in \mathbb{N})$ é uma base se R sobre $\mathbb{F}_q$.

Seja $\mathcal{L}(l)$ espaço vetorial gerado por $f_1, ..., f_l$, ou equivalentemente,

$$\mathcal{L}(l) = \{f \in R \mid \rho(f) \leq \rho_l\}.$$

Seja $l(i,j)$ o menor inteiro positivo l tal que $f_i f_j \in \mathcal{L}(l)$. Então a função $l(i,j)$ é estritamente crescente em ambos os argumentos, já que

$$\rho_i + \rho_j = \rho_{l(i,j)}$$

e ρ_i é estritamente crescente com função de i e é, de fato, uma função peso.

Com estas escolhas e tomando l uma não-lacuna em P temos que o código de Goppa $C(D, lP)$ onde $D = P_1 + \cdots + P_n$ é exatamente o código de avaliação E_l como o definido no exemplo 4.3.1.

Lembramos que dado um morfismo sobrejetivo de $\mathbb{F}_q$-álgebras

$$\varphi : R \longrightarrow \mathbb{F}_q^n$$

associando a cada elemento f_i da base de R o vetor $h_i = \varphi(f_i)$, o código de avaliação E_l estava dado por

$$E_l = \varphi(L_l) =< h_1, ..., h_l > .$$

No nosso caso, o morfismo de $\mathbb{F}_q$-álgebras considerado é

$$\varphi = av_{\mathcal{P}} \quad \text{onde} \quad \mathcal{P}\{P_1, ..., P_n\}$$

é definido por

$$\begin{array}{rccc} av_{\mathcal{P}} & : & R & \longrightarrow & \mathbb{F}_q^n \\ & & f & \longmapsto & (f(P_1), ..., f(P_n)) \end{array}$$

Vamos finalizar este trabalho descrevendo rapidamente um algoritmo para decodificar códigos de Goppa em um único ponto. Começaremos com algumas definições básicas.

Seja $C \subseteq \mathbb{F}_q^n$ um código linear com distância mínima d. Suponha que /break $c = (c_1, \ldots, c_n) \in C$ é uma palavra transmitida sendo recebida $y = c+e$. Note que c está unicamente determinada se a distância de y a c for no máximo $(d-1)/2$.

Definição 5.2 *O vetor $e = (e_1, \ldots, e_n) \in \mathbb{F}_q^n$ é chamado de vetor erro de y e o peso $w(e)$ é chamado o número de erros de y.*

O conjunto $\{i \in \{1, \ldots, n\} : e_i \neq 0\}$ é o conjunto de "coordenadas erradas"de y.

Lema 5.0.12 ((Pel3, Prop. 6.1)) *Seja H a matriz de paridade de C. Suponha que $y = c + e$, $c \in C$, e que J é um conjunto de no máximo $d-1$ elementos tal que o conjunto de posições erradas está contido em J. Então e é a unica solução do seguinte sistema de equações em $x = (x_1, \ldots, x_n)$:*

$$Hx^t = Hy^t \qquad e \qquad x_j = 0 \quad \textit{para todo } j \notin J\,.$$

Prova:
Claramente e satisfaz as equações. Seja x outra solução, então

$H(x-e)^t = (0, \cdots, 0)^t$ e logo $x-e \in C$, mais ainda como $w(x-c) \leq \#J \leq d-1$, temos que $x = c$. □

Agora consideremos $C = C_l$ como anteriormente. Para $y \in \mathbb{F}_q^n$, $i, j \in \mathbb{N}$ tais que $\rho_i + \rho_j \leq \rho_l$ e $J \subseteq \{1, \ldots, n\}$, definamos:

$$\begin{aligned} K_{ij}(y) &:= \{f \in \mathcal{L}(\rho_j P) : y \cdot e(fg) = 0, \text{ para todo } g \in \mathcal{L}(\rho_i P)\} \\ L_j(J) &:= \{f \in \mathcal{L}(\rho_j P) : e(f)_k = 0 \text{ para todo } k \in J\}, \end{aligned}$$

onde $e(f)_k$ é k-ésima coordenada do vetor $e(f) = (f(P_1), \ldots, f(P_n))$.
Note que $K_{ij}(y)$ é o núcleo da aplicacão linear $\mathcal{L}(\rho_j P) \to \mathcal{L}(\rho_i P)$ definida pela matriz

$$(s_{i',j'}(y))_{1 \leq i' \leq i, 1 \leq j' \leq j}.$$

Então se $y = c + e$, com $c \in C_l$, $K_{ij}(y) = K_{ij}(e)$ (pois $\rho_i + \rho_j \leq \rho_l$).

Lema 5.0.13 *Seja $y = c + e$, $c \in C_l$ e seja I o conjunto de posições erradas de y. Então*

(1) $L_j(I) \subseteq K_{ij}(y)$;

(2) $L_j(I) = K_{ij}(y)$ *se* $d(C_i) > w(e)$.

Prova:
(1) Seja $f \in L_j(I)$. Então $e(f)_k = 0$ para $k \in I$, já que para $g \in \mathcal{L}(\rho_i P)$,

$$e \cdot e(fg) = \sum_{k \in I} e_k e(fg)_k.$$

(2) Seja $f \in K_{ij}(y) = K_{ij}(e)$. Então para $g \in \mathcal{L}(\rho_i P)$,

$$0 = e \cdot e(fg) = e * e(f) \cdot e(g)$$

e também $e * e(f) \in C_i$. Como $w(e * e(f)) \leq w(e) < d(C_i)$ e $e * e(f) = 0$ temos que $e(f)_k = 0$ para $k \in I$ e portanto $f \in L_j(I)$.

□

Temos o assim chamado *algoritmo básico* para o código C_l, i.e. dado $y = c + e$ com $c \in C_l$, podemos calcular e se certas condições são satisfeitas.

Algoritmo básico.
(1) Encontrar $i, j \in \mathbb{N}$ tais que

(1.1) $\rho_i + \rho_j \leq \rho_l$;

(1.2) $d(C_i) > w(e)$;

(1.3) $L_j(I) \neq \{0\}$;

(2) Tome $f \in L_j(I) = K_{ij}(y)$ (podemos fazer isto graças ao lema 5.0.13; já que a escolha de f não vai depender de e) tal que

$$\#\{k : e(f)_k = 0\} \leq d(C_l) - 1\,.$$

(3) Aplique o lema 5.0.12 com $J = \{k : e(f)_k = 0\}$ para calcular e.

Proposição 5.0.14 *Seja $d^* := l + 1 - g$. Então o algoritmo básico corrige até $t := \lfloor (d^* - 1 - g)/2 \rfloor$ erros.*

Prova:
Podemos supor que $t \geq 1$, ou seja, $l \geq 2g+2$. Seja $w(e) \leq t$ e $I := \{k : e_k \neq 0\}$.
Suponha que l é par (o caso ímpar é análogo). Então $t = l/2 - g$.
Sabemos que $d(C_i) \geq i + 1 - g$ (veja teorema 4.4.5); então a condição (1.2) acima é satisfeita para $i + 1 - g \geq t + 1$, ou seja, $i \geq t + g = l/2$.
Escolhendo $i = l/2$ e como $\rho_j \leq j + g - 1$ a condição (1.1) é satisfeita para $j \leq l/2 - g + 1 = t + 1$.
Então se $j = t + 1$ a condição (1.3) também é satisfeita já que I impõe no máximo t condições sobre $\mathcal{L}(\rho_j P)$ que tem dimensão $t + 1$.
Seja $f \in L_j(I)$, então

$$\#\{k : e(f)_k = 0\} \leq \rho_j \leq l/2 < l + 1 - g \leq d(C_l)$$

pois $l > 2g - 2$. Finalmente, aplicando o lema 5.0.12, terminamos a demonstração da proposição. □

Na verdade o algoritmo básico descrito acima também funciona para os códigos C_l definidos de maneira mais geral como em 4.7 (dos quais os códigos de Goppa em um único ponto são um caso particular).
No caso mais geral, o algoritmo básico também corrige até $t := \lfloor (d^* - 1 - g)/2 \rfloor$ erros, onde $d^* = l + 1 - g$ e g é o número de lacunas da função peso ρ.

Para códigos de Goppa em geral, também existe um algoritmo de decodificação (veja (Sti), Capítulo VII).

Se $C = C_{\mathcal{L}}(D, G)$ é o código de Goppa associado a D e G (veja definição 3.18), então o algoritmo acima mencionado corrige até $s := \lfloor (d^* - 1 - g)/2 \rfloor$ erros, onde neste caso $d^* = degG - (2g - 2)$.

Suponhamos que $D = P_1 + \cdots + P_n$ e $G = lP$. Neste caso podemos usar tanto o algoritmo básico como o algortimo para os códigos de Goppa, uma pergunta natural seria: Qual dos dois corrige mais erros?

Fazendo as contas temos que o algoritmo básico corrige

$$t = \lfloor ((d^* - 1 - g)/2 \rfloor = \lfloor ((l + 1 - g) - 1 - g)/2 \rfloor = \lfloor (l - 2g)/2 \rfloor \text{ erros.}$$

Já o outro algoritmo corrige

$$s = \lfloor ((l - 2g + 2) - 1 - g)/2 \rfloor = \lfloor (l + 1 - 3g)/2 \rfloor \text{ erros.}$$

Se $g > 1$, o algoritmo básico corrige mais erros e por isso é interessante considerar os códigos de Goppa em um único ponto como um caso particular dos códigos de avaliação.

Na prática estaremos interessados em códigos longos e sabemos que o comprimento do código depende do número de pontos racionais da curva (no exemplo anterior utilizamos $n + 1$ de tais pontos). O gênero g e o número de pontos racionais N de uma curva definida sobre $\mathbb{F}_q$ satisfazem a cota de Hasse-Weil:

$$N \leq q + 1 + 2g\sqrt{g}.$$

Então para aumentar o número de pontos racionais devemos aumentar também o gênero da curva considerada e, neste caso, o algoritmo básico torna-se mais eficiente.