4. Lecture

Mittwoch, 20. September 2023 10:16

Midterm 2nd November
→ electronic (in LMU)

$GF(q) = \{0,1,2,\dots,q-1\}$; '+' ; '*'

"q" prime $\quad \alpha + \beta = p \cdot q + r \quad r \in \{0,1 \dots q-1\}$
$\Rightarrow \alpha + \beta \bmod q = r$

$\alpha \cdot \beta = p' \cdot q + r' \quad r \in \{0,1 \dots q-1\}$
$\Rightarrow \alpha \cdot \beta \bmod q = r'$

$\alpha \in GF(q) \setminus \underbrace{\{0\}}_{\text{zero null (zero element)}} \qquad \alpha^{q-1} = 1$

$\text{ord}(x) = M: \quad \alpha^M \equiv 1$; $\text{ord}(\alpha) = q-1 \rightarrow \alpha$ primitive element
Bsp.: $3 \in GF(7)$: 3,2,6,4,5,1
one zero element

Polynoms over GF(q)

$a(x) = a_0 + a_1 x + \dots + a_n x^n \qquad b(x) = b_0 + b_1 x + \dots + b_m x^m$

$\deg(a(x)) = n$,

$a_0, a_1 \dots a_n, x \in GF(q)$ root seeking easier than by traditional polynoms

$\bar{a} = (a_0, a_1, \dots, a_n)$ (vector)

$c(x) = a(x) + b(x) = a_0 + b_0 + (a_1 + b_1)x + \dots + (a_i + b_i)x^i + \dots$

$c(x) = a(x) \cdot b(x) = a_0 b_0 + (a_0 b_1 + a_1 b_0)x + (a_0 b_2 + a_1 b_1 + a_2 b_0)x^2 + \dots + \left(\sum_{j=0}^{\min\{\deg(a(x)),\, i\}} a_j b_{i-j} \right) x^i + \dots$

convolution

	polynom	vectoriale
+	$c(x) = a(x) + b(x)$	$\bar{c} = \bar{a} + \bar{b}$
•	$c(x) = a(x) \cdot b(x)$	$\bar{c} = \bar{a} \circledast \bar{b}$

Division among polynoms:

$\forall\, a(x), d(x)$ (division polynom) $\quad \deg(a(x)) = n > \deg(d(x)) = k \;;\; \exists\, q(x), r(x)$

$a(x) = q(x) d(x) + r(x) \qquad \deg(r(x)) < \deg(d(x))$

Fundamental law of Algebra: number of roots of $a(x) \leq \deg(a(x))$

$a(x)\big|_{x=x_i} = \emptyset \qquad a(x) = b(x)(x - x_i) \qquad \deg(b(x)) < \deg(a(x))$

$a(x)\big|_{x=x_i} = b(x)(x - x_i) + r = \emptyset$

C(n,k) Reed-Solomon codes

$\bar{u} = (u_0 \dots u_{k-1})$ length k
$\bar{c} = (c_0 \dots c_{n-1})$ length n

$u(x) = u_0 + u_1 x + \dots + u_{k-1} x^{k-1}$

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

$\alpha, \alpha_1, \dots, \alpha_{n-1} \in GF(q) \setminus \{0\}$

$n = q - 1$

Encoding:

$c_0 = u(x)\big|_{x=\alpha_0} = u_0 + u_1 \alpha_0 + u_2 \alpha_0^2 + \dots + u_{k-1} \alpha_0^{k-1}$
$c_1 = u(x)\big|_{x=\alpha_1} = u_0 + u_1 \alpha_1 + u_2 \alpha_1^2 + \dots + u_{k-1} \alpha_1^{k-1}$
$\vdots$
$c_{n-1} = u(x)\big|_{x=\alpha_{n-1}} = u_0 + u_1 \alpha_{n-1} + u_2 \alpha_{n-1}^2 + \dots + u_{k-1} \alpha_{n-1}^{k-1}$

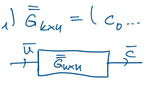
$w(\bar{c}) = n - \#0$
$n - (k-1) \leq w_{min} \leq n - k + 1$ ($w_{min} \uparrow d_{min}$ (linear code))
$\Rightarrow n - k + 1 = d_{min}$ in the case of RS code
$\Rightarrow$ are MDS codes
$\Rightarrow$ best possible codes !!

Design: Given "t" errors to be corrected

GIVEN: $k - k = 2t \qquad n = q - 1 \qquad q$ - prime

$\exists\, C(n,k)$ RS code can correct t number of errors in an optimal fashion

is a linear code: $(u_0 \dots u_{k-1})\, \bar{\bar{G}}_{k \times n} = (c_0 \dots c_{n-1})$; $\bar{\bar{G}}_{k \times n} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_0 & \alpha_1 & \dots & \alpha_{n-1} \\ \alpha_0^2 & \alpha_1^2 & \dots & \alpha_{n-1}^2 \\ \vdots & & & \\ \alpha_0^{k-1} & \alpha_1^{k-1} & \dots & \alpha_{n-1}^{k-1} \end{pmatrix}$



$\alpha \in GF(q)$ primitive element

$\alpha_0 = \alpha^0 = 1$; $\alpha_1 = \alpha^1 = \alpha$; $\alpha_2 = \alpha^2 \dots \alpha_{n-1} = \alpha^{n-1}$

RS code correcting every double error

$t = 2 \rightarrow n - k = 2 \cdot t$
$n - k = 4$

q - prime
$n = q - 1$

q	n	k
2	1	—
3	2	—
5	4	0
7	6	2
11	10	6
13	12	8

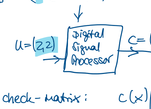
relevant

which choose?

· energy Einschränkung
→ je höher n & je skala mehr eingeteilt
& mehr noise kann detected
korrigieren

$\Rightarrow C(6,2)$ over $GF(7)$
$3 \in GF(7)$

$\bar{\bar{G}}_{2 \times 6} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 3 & 2 & 6 & 4 & 5 \end{pmatrix}$



Parity-check-matrix: $c(x)\big|_{x=\alpha^i} = \emptyset \;;\; i = 1, \dots, n-k$

$c_0 + c_1 \alpha + \dots + c_{n-1} \alpha^{n-1} = \emptyset$
$c_0 + c_1 \alpha^2 + \dots + c_{n-1} \alpha^{2(n-1)} = \emptyset$
$\vdots$
$c_0 + c_1 \alpha^{n-k} + \dots + c_{n-1} \alpha^{(n-k)(n-1)} = \emptyset$

$\bar{\bar{H}}_{(n-k) \times n} \bar{c}^T = \bar{0}^T$

$\bar{\bar{H}}_{(n-k) \times n} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(n-1)} \\ \vdots & & & & \\ 1 & \alpha^{n-k} & \alpha^{2(n-k)} & \dots & \alpha^{(n-k)(n-1)} \end{pmatrix}$

$\bar{\bar{G}}_{2 \times 6} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 3 & 2 & 6 & 4 & 5 \end{pmatrix}$

$\Rightarrow \bar{\bar{H}}_{4 \times 6} = \begin{pmatrix} 1 & 3 & 2 & 6 & 4 & 5 \\ 1 & 2 & 4 & 1 & 2 & 4 \\ 1 & 6 & 1 & 6 & 1 & 6 \\ 1 & 4 & 2 & 1 & 4 & 2 \end{pmatrix}$ jump 2x in 1 Line, jump 3x, jump 4x

$\Rightarrow \left(\bar{\bar{H}}\, \bar{c}^T \right)_e = \emptyset \qquad \sum_{i=0}^{n-1} \alpha^{e_i c_i} = \sum_{i=0}^{n-1} \alpha^{e_i} \sum_{j=0}^{k-1} \alpha^{ij} u_j = \emptyset$

$e = 1, \dots, n-k$

$\sum_{j=0}^{k-1} u_j \sum_{i=0}^{n-1} \alpha^{[e+j]i} = \sum_{j=0}^{k-1} u_j \frac{\alpha^{(e+j)n} - 1}{\alpha^{e+j} - 1}$ (finite geometric series) $\quad q = \alpha^{e+j}$

$\alpha^{(e+j)n} = \left(\alpha^n\right)^{e+j} = \left(\alpha^{q-1}\right)^{e+j} = 1^{e+j} = 1$