Sec 4

$C/\mathbb{F}_p \qquad y^2 = f(x)$

$g \geq 1 \qquad f \in \mathbb{F}_p[x]$

h

Prop 4.1.1 let $h = f^{\frac{p-1}{2}}$

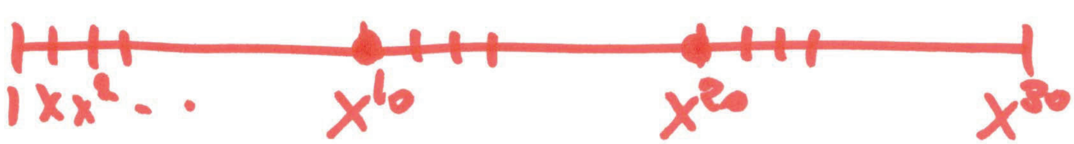
$$\# C(\mathbb{F}_p) \equiv 1 - \sum_{j=1}^{g} h_{j(p-1)} \pmod p$$

Example : $p = 11 \qquad g = 2$

$\deg f = 2g + 2 = 6.$

$\deg h = 6 \cdot \frac{p-1}{2} = 30.$

$$\# C(\mathbb{F}_{11}) \equiv 1 - h_{10} - h_{20} \pmod{11}$$



Idea of proof:

evaluate $\sum_{\alpha \in \mathbb{F}_p^*} \left(f(\alpha)^{\frac{p-1}{2}} + 1\right) \mod p$ in 2 ways.

Legendre symbol $\left(\frac{f(\alpha)}{p}\right)$

$$= \sum_{\alpha \in \mathbb{F}_p^*} (h(\alpha) + 1)$$

$$\rightsquigarrow \sum_{\alpha \in \mathbb{F}_p^*} \alpha^i .$$

Cartier trace formula (Thm 4.2.7)

Define $A_f \in \mathrm{Mat}_g(\mathbb{F}_p)$
by $(A_f)_{v,u} = h_{vp-u}$ $\quad 1 \le u, v \le g.$
(still $h = f^{\frac{p-1}{2}}$)

Hasse-Witt / Cartier-Manin matrix.

$$\#C(\mathbb{F}_{p^r}) \equiv 1 - \operatorname{tr}(A_f^r) \mod p$$

$$r = 1, 2, \ldots$$

<u>Proof:</u> idea: use $f(\alpha)^{\frac{p^r-1}{2}}$.

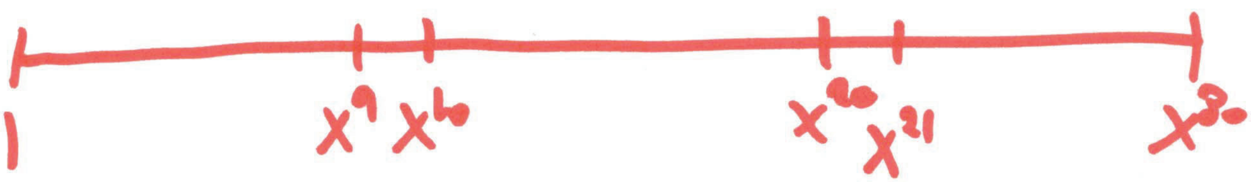
<u>Example:</u> $p = 11$ $g = 2$

$$A_f = \begin{bmatrix} h_{10} & h_9 \\ h_{21} & h_{20} \end{bmatrix}$$

$$\#C(\mathbb{F}_p) \equiv 1 - \operatorname{tr} A_f$$

$$\#C(\mathbb{F}_{p^2}) \equiv 1 - \operatorname{tr}(A_f^2)$$

$(\mathrm{mod}\ p)$



---

recall:

$$Z_C(T) = \exp\left(N_1 T + \frac{N_2}{2} T^2 + \cdots\right)$$

$$\exp(z) = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \cdots$$

Consequences for L-polynomial

$N_r = \# C(\mathbb{F}_{p^r})$

$$A_f \xrightarrow[\text{formula}]{\text{trace}} N_1, \dots N_g \pmod p$$

$\Big\downarrow$ if $p > g$ (Prob 4.3.3)

CAN FAIL if $p \leq g$ (Prob 4.3.1)

$$Z_C(T) \pmod p$$

up to T^g.

$\Big\downarrow$ easy.

$$L_C(T) \pmod p$$

AMAZINGLY:

Thm 4.3.2

$$L_C(T) = \det(I - TA_f) \mod p$$

(even if $p \leq g$!!)

"Expansion strategy" to compute $L_C(T)$ mod p.

1) Expand out $h = f^{\frac{p-1}{2}}$ to get A_f.

2) Compute $\det(I - TA_f)$.

<u>Complexity analysis.</u> (Multitape Turing Machine – bit complexity) Sec 1.6

Compute $h = f^{\frac{p-1}{2}}$ by "repeated squaring". (sec 2.3)

eg. $p = 83$.

want $f^{41} = f \cdot (f^{20})^2$

$f^{20} = (f^{10})^2$

$f^{10} = (f^{5})^2$

$f^{5} = f(f^{2})^2$

$f^{2} = f^{2}$.

Complexity dominated by last multiplication

Let $M_p(n)$ = cost of multiplying polys of deg $\leq n$ in $\mathbb{F}_p[x]$.

$M_{int}(n)$ = cost of " integers of $\leq n$ bits.

<u>Know:</u> $M_{int}(n) = O(n \log n)$ (2019).

<u>Think:</u> $M_p(n) = O(b \cdot \log b)$???

$b = n \log p$. (total bitsize)

Can use "Kronecker substitution" to get

$$M_p(n) = O(b_1 \log b_1)$$

$$b_1 = n \log(np)$$

Cost of step ①? $\deg h = O(gp)$.

$$b_1 = O(gp \cdot \log(gp^2))$$
$$= O(gp \cdot \log(gp))$$

$$b_1 \cdot \log b_1 = O(gp \cdot \log(gp) \log(gp \cdot \log(gp)))$$
$$\boxed{= O(gp \cdot \log^2(gp)).}$$

If $p >> g$ then this is

$O(gp \cdot \log^2 p)$. "linear" in p.
poly. in g.

Step (2) Let ω = exponent of matrix multiplication.

i.e. can multiply $d \times d$ matrices over K using $O(d^\omega)$ field ops in K

Classical: $\omega = 3$

Strassen: $\omega = \frac{\log 7}{\log 2} \doteq 2.81$

record: $\omega = 2.372\ldots$

Thm: Can compute $\det(I - TA_f)$ in $O(d^\omega)$ ops in K.

Each op in $\mathbb{F}_p$ needs $O((\log p)^{1+\epsilon})$ bit operations (see 2.1)

Cost of step ② is:

$$O\left(g^{\omega} (\log p)^{1+\varepsilon} \right) \qquad \text{(bit complexity)}$$

Total: $O\left(g p \log^2(gp) + g^{\omega} (\log p)^{1+\varepsilon} \right)$

Notice: polynomial in g.

Unlike enumeration (exp. in g).

But only have $L_C(T)$ mod p.

$p \equiv a \pmod d$ $\qquad (a,d)=1.$

Linnik: $\quad p = O(d^L)$ $\quad$ absolute L.

record: $L=5$

$p \equiv 1 \mod 1000.$

1001

2001

3001 noam

4001

5001

⋮

1000000000000001

RH: $L=2+\varepsilon$

1,000,001

Conjectured $L=1+\varepsilon$

$p = O(d \lg^2 d)$

We need $L < 1+2^{-100}$