March 5, 2019

REMINDERS

(1) Group composition + topic due TONIGHT: 11:59pm
- If you miss this deadline you will get a 0 on the ENTIRE miniproject
- Email me the info

(2) HW 2 due this Th at 11:59pm

(3) HW 1 has been graded → see piazza post + Autolab for details

(4) HW 3 out ~~next~~ Th (due in 2 weeks/: March 28) after that

(5) 2-page report for video due in 4 weeks (April 2)

RECAP!

(•) Singleton bound: Any $(n,k,d)_q$ code satisfies: $d \le n-k+1$.

(•) Reed-Solomon code: An $[n,k,n-k+1]_q$ code $\leftarrow (q \ge n)$

$\bar{m} \in \mathbb{F}_q^k \longmapsto \sum_{i=0}^{k-1} m_i \cdot x^i = P_{\bar{m}}(x)$ Eval points $\alpha_1, \ldots, \alpha_n$

$RS(\bar{m}) \longmapsto (P_{\bar{m}}(\alpha_1), \ldots, P_{\bar{m}}(\alpha_n))$

(•) Degree mantra: Any non-zero poly $f(x) \in \mathbb{F}_q[X]$ of deg t has $\le t$ roots (in $\mathbb{F}_q$)

TODAY: (Proof reader: Yash)

(-) Proof of degree mantra
(-) Maximum Distance separable ~~codes~~ (MDS) codes
(-) Hashes for fingerprints (by popular demand)

Pf: By induction on deg t.

Base case: $t=0$ (non-zero const polys $\Rightarrow$ 0 roots) ✓

$f(x)$ has deg $t>0$ (I.H.: $\forall$ non-zero poly of deg $d \le t-1$ have $\le d$ roots)

Case 1: f has no roots in $\mathbb{F}_q$ $\Rightarrow$ done (as $0<t$)

Case 2: $\exists\, \alpha \in \mathbb{F}_q$ s.t. $f(\alpha)=0$

Claim: If $f(\alpha)=0 \Rightarrow (x-\alpha) \mid f(x)$

Claim ⇒ $f(X) = (X-\alpha)\, g(X)$ $\quad \deg(g) = \deg(f) - 1$

If $\exists\ \beta \neq \alpha$ s.t $f(\beta) = 0 \Rightarrow g(\beta) = 0$ $\quad = t-1$

By I.H. g has $\leq t-1$ roots

$\alpha \neq \beta \Rightarrow (\alpha-\beta) \neq 0 \Rightarrow g(\beta) = 0$

$\Rightarrow f$ " $1 + \#\,\text{roots}(g) \leq 1 + t - 1 = t$ ∎

Pf of Claim: By the fundamental rule of division

$f(X) = (X-\alpha)\, g(X) + r(X)$ where $\deg(r) < \deg(X-\alpha) = 1$

$\Rightarrow \deg(r) = 0 \Rightarrow$ r is a constant, $r(X) = r$

$X \leftarrow \alpha$: $\quad f(\alpha) = (\alpha-\alpha)\cdot g(\alpha) + r(\alpha) \Rightarrow r = 0$

($f(\alpha) = 0$, $(\alpha-\alpha) = 0$, $r(\alpha) = r$)

$\Rightarrow f(X) = (X-\alpha)\cdot g(X)$ ∎

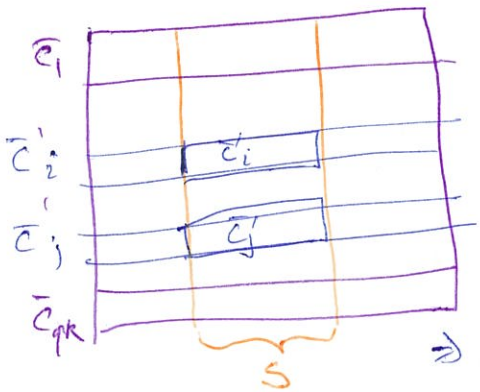
Maximum Distance Separable (MDS) codes

$(n,k,d)_q$ code is MDS if $d = n-k+1$.

Def: $S \subseteq [n]$ $\quad C_S$ be C projected to co-ordinates in S

Prop: $C \subseteq \Sigma^n$ be MDS & $\dim(C) = k$. Then $\forall\ S \subseteq [n]$ $|S| = k$, $\quad C_S = \Sigma^k$ ← true for all $\binom{n}{k}$ choices of S.

Pf:



Claim 2: $\forall\ i \neq j$ $\quad \bar{c}'_i \neq \bar{c}'_j$

(Pf: If not ⇒ $\Delta(\bar{c}_i, \bar{c}_j) \leq n - |S| \leq n-k$ ⇒ contradicts C having $d = n-k+1$)

⇒ $C_S = \{\bar{c}'_1, \bar{c}'_2, \ldots, \bar{c}'_{q^k}\}$

By claim 2, $|C_S| \geq q^k \Rightarrow |C_S| = q^k$

Since $C_S \subseteq \Sigma^k \Rightarrow C_S = \Sigma^k$ ∎

Hashing for fingerprints

→ String based hashing

→ secret s store $h(s)$ $\quad h: \Sigma^* \to \Sigma^*$ (ideally $s_1 \neq s_2 \Rightarrow h(s_1) \neq h(s_2)$)

↳ Completeness: $s_1 = s_2$ then $h(s_1) = h(s_2)$ (easy to compute)

→ Soundness: given $h(s)$ "hard" to compute s.

Idea: Use biometrics instead of string passwords

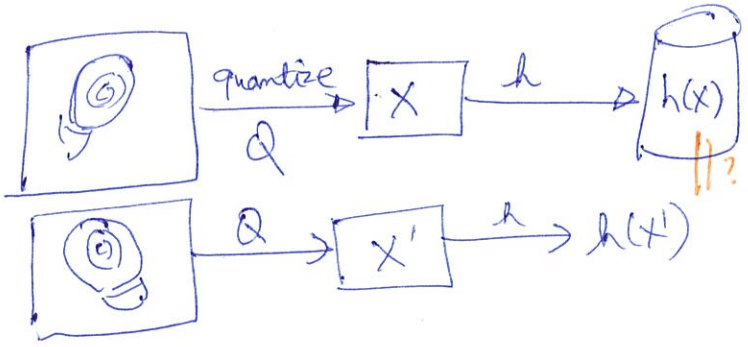
Today: Technical challenges in designing hashes for fingerprints

---

Outline of process:

Enrollment step:

Verification step:



Main Q: Which h should we use?

Option 1: Use any off-the-shelf hash function h for strings

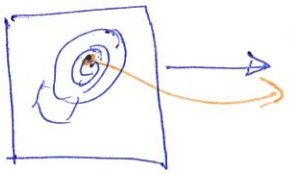
Problems:

(P1) Translation + rotation

(P2) Varying pressure

(P3) Different parts of fingerprints get captures

(P4) (x_i, y_i, θ_i) are not ordered.



ridge points. minutia → where ridges bifurcate/ends.

(x_i, y_i, θ_i)

Fuzzy vault

Use fingerprint f to "lock" a secret s

→ $f' \approx f \Rightarrow$ unlock the vault to get s

→ f' is "far" from f → cannot get s.

hash for fingerprints

Lock (f, s), $h(s)$

---

formalize fuzzy vault : Assum $(x_i, y_i, \theta_i) \in \mathbb{F}_q$

Notation: $\binom{S}{t}$ = set of all subsets of S of size $= t$.

- Integers: $k \geq 1$, $n \geq t \geq 1$
- Secret : $\bar{s} \in \mathbb{F}_q^k$
- fingerprint : $f \in \binom{\mathbb{F}_q}{t}$
- LOCK : $\mathbb{F}_q^k \times \binom{\mathbb{F}_q}{t} \longrightarrow \binom{\mathbb{F}_q}{n}$
- UNLOCK : $\binom{\mathbb{F}_q}{t} \times \binom{\mathbb{F}_q}{n} \longrightarrow \mathbb{F}_q^k$

GOAL: Design LOCK, ONLOCK s.t for some $c < t$

① c-completeness: $f, f' \in \binom{\mathbb{F}_q}{t}$ s.t $|f \setminus f'| \leq c$

$\Rightarrow$ UNLOCK $(f', \text{LOCK}(s, f)) = s$

② Soundness: "Hard" to "get" s fm LOCK (s, f).