[I486S]
暗号プロトコル理論

藤崎 英一郎

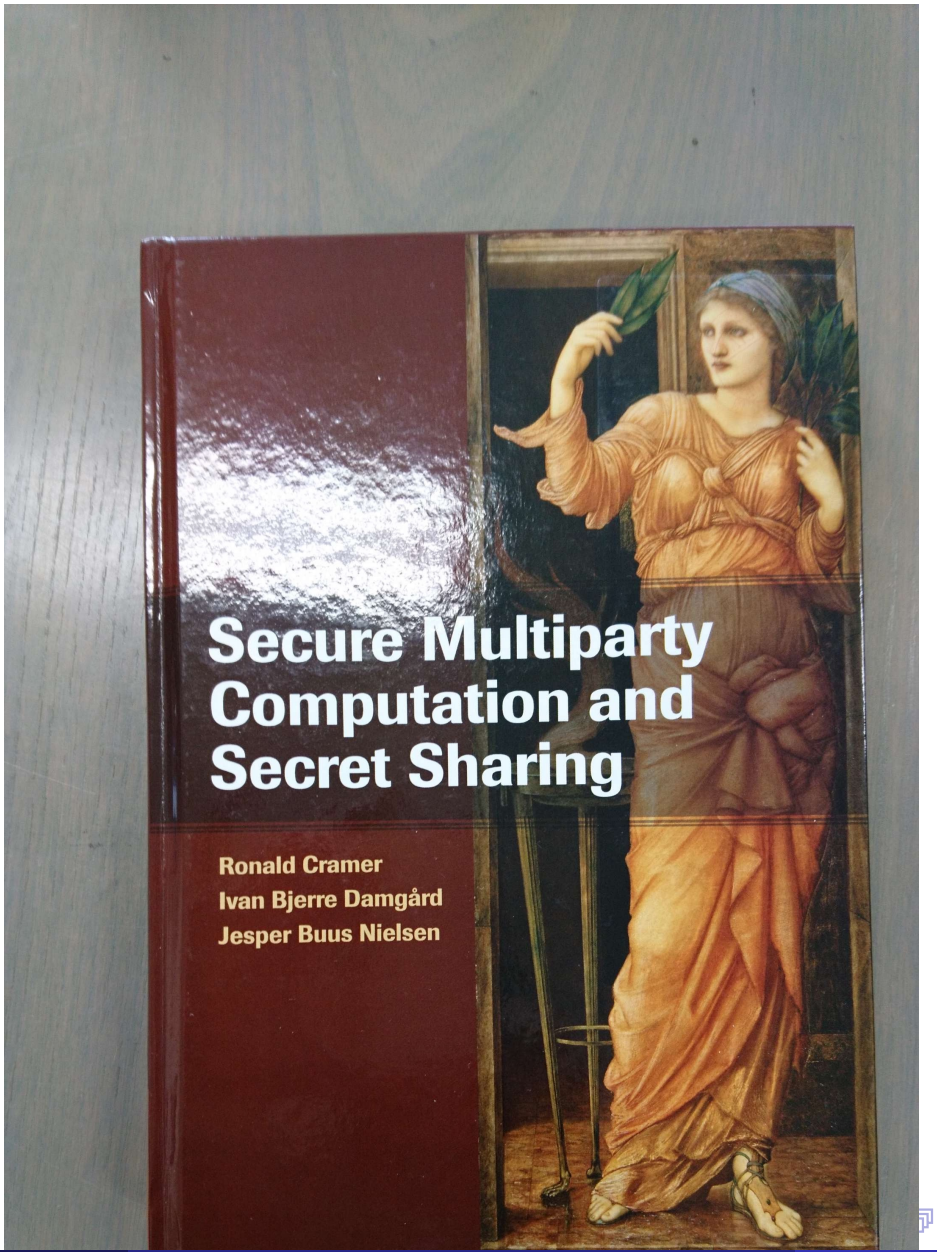
北陸先端科学技術大学院大学

2020 年 4 月 21 日

基本情報

- 開催日時
  - 火曜日 (Tuesdays) 5 時限 17:10 – 18:50
  - 予定 4/21, 4/28, 5/12, 5/19, 5/26 (3 限), 5/26, 6/2, 6/16, 6/23, 6/30, 7/7, 7/14, 7/21, 7/28 (予備日 8/4).
- 開催形態：WebEx によるリアルタイム遠隔配信
- 教科書：指定なし（ただし Cramer, Damgård, Nielsen の本 “Secure Multiparty Computation and Secret Sharing” にかなり忠実に従う）
- 資料置き場 JAIST-LMS 下の I486S.

基本とする参考書


Secure Multiparty Computation and Secret Sharing
Ronald Cramer
Ivan Bjerre Damgård
Jesper Buus Nielsen

講義内容

暗号の予備知識なしで分かる（はずの）暗号のお話

- 有限体上の四則演算、簡単な線形代数の知識は使う
- 近年、国内外の大学、企業の研究所で実装実験が盛んに行われている秘密計算（マルチパーティ計算）の理論について講義
- 秘密計算とは、複数の参加者が各自持つ秘密を漏らさず、しかしその秘密から計算できる計算結果のみを協力して計算する暗号プロトコル

本講義では、秘密計算と行ったり、マルチパーティ計算と言ったり、はたまた MPC と約したりするが同じものを指す

参考文献

- R. Cramer, I. Damgaard, and J. Nielsen: Secure Multiparty Computation and Secret Sharing, Cambridge University Press (ISBN 9781107043053).
- R. Cramer, I. Damgaard, J. Nielsen: Multiparty computation, Introduction.
- Ben-Or, S. Goldwasser and A. Wigderson: Completeness Theorems for Non-Cyptographic Fault-Tolerant Distributed Computation, in STOC88, pp. 1–10.
- R. Cramer, I. Damgaard and U. Maurer: General Secure Multi-party Computation from any Linear Secret-Sharing Scheme, in EUROCRYPT2000, pp 316–334.
- R. Cramer, I. Damgaard and Y. Ishai: Share Conversion, Pseudorandom Secret-Sharing and Applications to Secure Computation, in TCC2005, pp 342–362.

授業の進め方、勉強の仕方

- 授業に参加し、分からないことは積極的に聞く
- 授業中に秘密計算のゲームをやるので、ゲームをやりながら内容を掴む
- アップされた資料を復習する
- 評価方法: 授業への貢献度とレポート

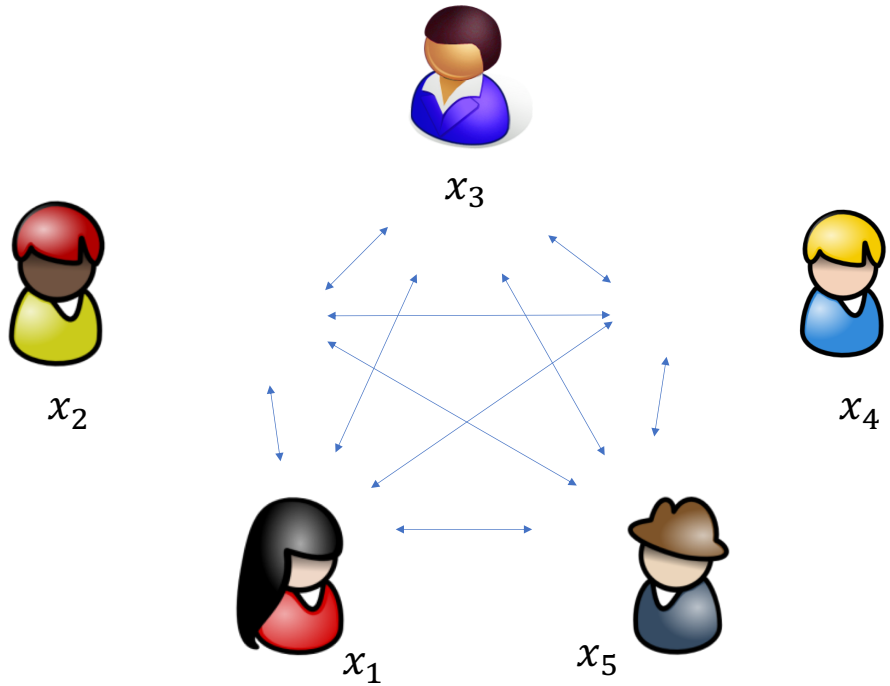
秘密計算に関するプレスリリースや記事

- NTT、秘密分散技術が国際標準化機構の国際標準において標準技術として採択
- NEC、産総研が秘密計算を実用化 データの中身を秘匿し、統計分析
- 秘密分散・秘密計算システム - 一橋大学社会科学統計情報センター
- NEC、機密情報の漏えいを強固に防止する秘密計算の高速化手法を開発
- 暗号化したままデータ分析を行う秘匿分析技術を開発 - 日立製作所
- NEC、機密情報の漏えいを強固に防止する秘密計算の高速化手法を開発
- NEC・富士通・日立に見る「マテリアルズ・インフォマティクス」最前線
- 複数の研究機関が持つゲノムデータを相互に開示せず分析する解析手法を開発

本日の講義の内容

秘密計算（マルチパーティ計算）

- 参加者: $P_1, \ldots, P_n$.
- 各 P_i への秘密の入力: $x_i \in \{0,1\}^\lambda$
- 全参加者への入力（公開情報）: 関数 $F : \{0,1\}^{n\lambda} \to \{0,1\}^*$.
- 各 P_i への出力：$F(x_1, \ldots, x_n)$. より一般的には、参加者ごとに違う出力をすることも許す.
- ネットワークモデル: synchronous network with peer-to-peer secure channel



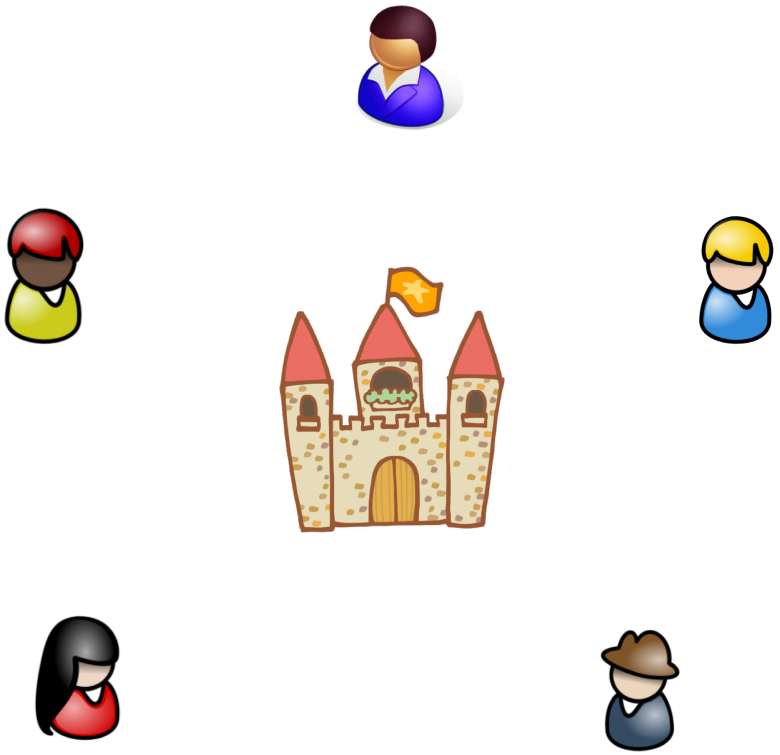
ネットワークモデル

- どの参加者同士も peer-to-peer で秘密通信可 (secure channel)
- 同期型ネットワーク (synchronous network)
  - 送られた情報は遅延なく、次のスロットで相手に必ず届く
  - cf. 非同期型ネットワーク (asynchronous network)
- 同報通信ができることを仮定する場合あり
  - 同報通信を仮定しない場合、Byzantine 合意を用いて同報通信を実現する（不正者の数が全体の参加者の 1/3 未満なら同期型ネットワークで実現できる）
  - 今回の話では同報通信を仮定しない。

Byzantine 合意問題

Byzantine 合意問題解法あり $\Longleftrightarrow$ peer-to-peer secure channel での Broadcast の問題の解法あり

定理 (Lamport): 同期型ネットワークなら、裏切り者の数が全体の 1/3 未満ならエラーなしで Byzantine 合意問題に解法あり.



将軍A~Eが、Byzantine 帝国の首都コンスタンチノープルを囲んでいる。

将軍達は、全員で、「攻撃」(attack) または「撤退」(retreat) ができるように各将軍に伝令を送る。

多数決で決まった方に全員一致で行動したい。

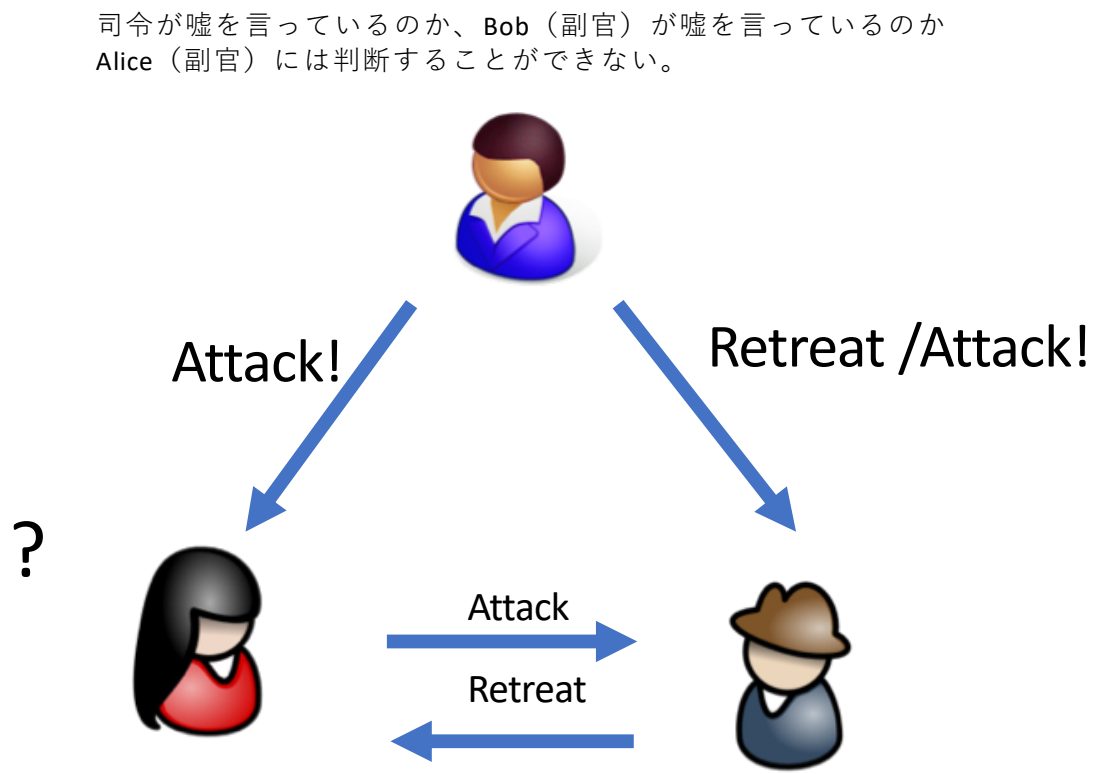
ただし、裏切り者がこの中にいる。

自分の情報を全員に正しく伝えるにどうしたら良いか？

司令官と副官の問題

Byzantine 合意問題は、司令官と副官の問題を解くことと同じになる。

司令と副官の中に 1/3 以上の裏切り者がいた場合、（エラーのない）解法は存在しない。
以下、$n = 3$ の例。

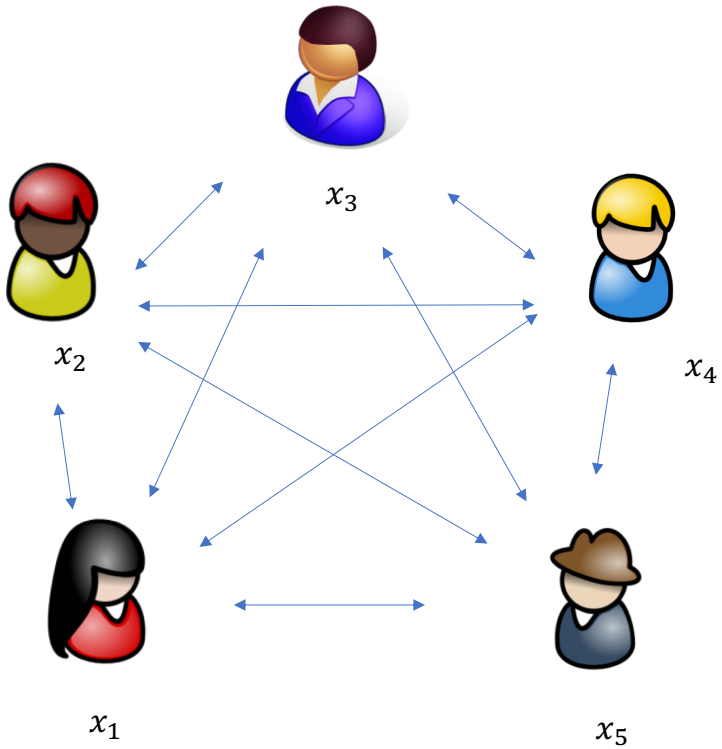


どのようにMPCの安全性を考えるか

各参加者が信頼できる第三者に自分の秘密を渡し、計算結果だけをもらえるのを理想のプロトコル。現実のプロトコルが理想と（ほぼ）同じになれば安全という。

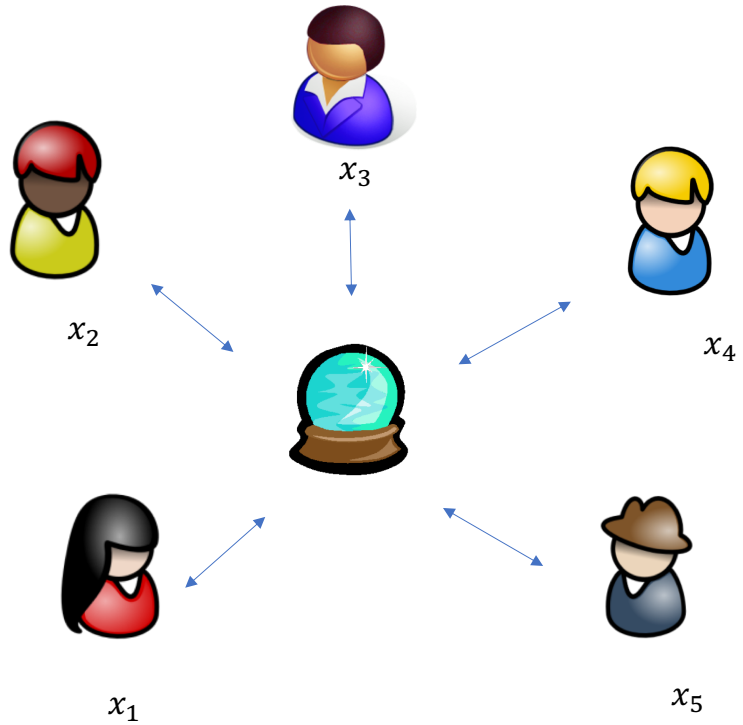
現実のプロトコル

互いに通信し計算結果を得る



理想状態

各自の秘密を  に預けると計算結果を返してくれる



不正者（攻撃者）に関する決め事

（多くの場合は）不正者の数を制限（この話でも）。不正者同士は常に結託して情報を共有できる。さらに、不正者のタイプを次のように分類する。

- Passive Adversary (aka semi-honest or honest-but-curious).
  - 受動的攻撃者
  - 正規のプロトコルからは逸脱しないが、プロトコルから得た情報から正直な参加者の秘密情報を得ようとする
- Active Adversary
  - 能動的攻撃者
  - プロトコルから逸脱しても良い。正直な参加者の秘密情報を得ようとすることと、プロトコルを失敗に終わらせようとする。

より細かい話はのちの安全性モデルのところで

知られている結果

不正者数に応じた結果

	Passive	Active w/ broadcast	Active w/o broadcast
情報理論的安全	$< \frac{n}{2}$	$< \frac{n}{3}$	$< \frac{n}{3}$
統計的安全	$< \frac{n}{2}$	$< \frac{n}{2}$	$< \frac{n}{3}$
計算量的安全	n	$< \frac{n}{2}$	$< \frac{n}{2}$

ただし、$n > 2$. 結果は全て optimal

- 情報理論的安全 = 現実と理想の差が全くない
- 統計的安全: 現実と理想の差がわずかな統計的差しかない。
- 計算量的安全: 多項式時間制限のアルゴリズムでは、識別できないぐらいしか、現実と理想の分布に違いはない。

ここで話す MPC

BGW [BGW88] / CCD [CCD88] 型の秘密計算方式.

- 情報理論的安全な三者以上の秘密計算プロトコル (MPC).
  - 不正者数 $< n/2$ としたときの Passive 安全な MPC
    [BGW88] に Michael Rabin の改良提案を組み合わせたもの
  - 不正者数 $< n/3$ としたときの Active 安全な MPC
    [BGW88] に [CDM00] のアイデアなどを混ぜ合わせ [CDN15] で解説されたもの
- 主要テクニック
  - 線型秘密分散 (Linear Secret Sharing (LSS))
  - 検証可能線型秘密分散 (Verifiable Linear Secret Sharing (VLSS))
  - (V)LSS での掛け算
    - 加減算は（線形性から）容易なので、乗算をするテクニックが重要。
    - 加減算、乗算ができると論理回路を計算できるので、任意の（多項式時間計算可能な）関数の MPC ができるようになる。

MPC の構成

一般には次のような構成の仕方になる

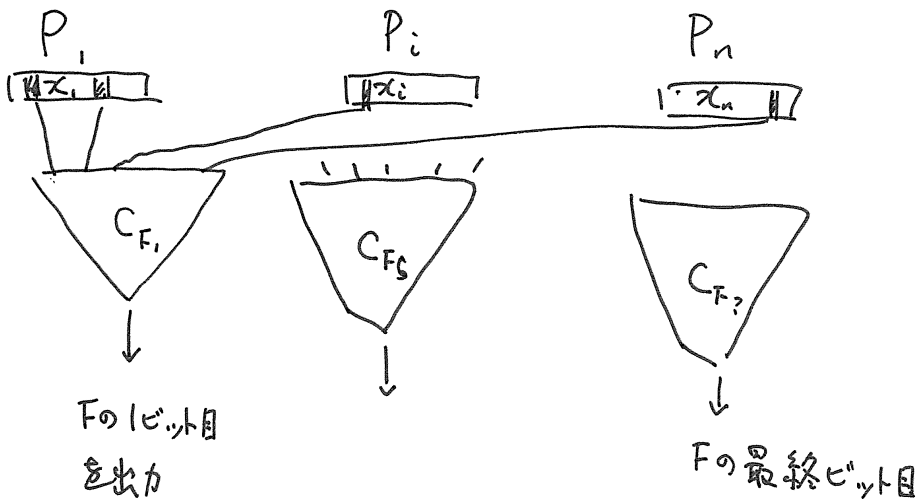
- $F : \{0,1\}^{n\lambda} \to \{0,1\}^*$: 多項式時間計算可能関数
- C_F: F を計算する多項式サイズの AND, OR, NOT で構成された論理回路
- K: $n < \#K$ な有限体
- C_F の論理演算子への入力 $b \in \{0,1\}$ を $b \in \{0,1\} \subset K$ と K の元とみなす。
- AND, OR, NOT の論理ゲートを K 上の演算に置き換える。
  - $b \wedge b' \iff b \cdot b' \in K$.
  - $b \vee b' \iff b + b - b \cdot b' \in K$.
  - $\neg b \iff 1 - b \in K$.

関数 F の回路

$F : \{0,1\}^{\lambda} \times \cdots \times \{0,1\}^{\lambda} \to \{0,1\}^{*}$ ← 任意の有限ビット列の集合

(n 個)

回路に分解



参考文献 I

[BGW88] M. Ben-Or, S. Goldwasser, and A. Wigderson.

Completeness theorems for non-cyptographic fault-tolerant distributed computation.

In Simon [Sim88], pages 1–10.

[CCD88] D. Chaum, C. Crépeau, and I. Damgård.

Multiparty unconditionally secure protocols.

In Simon [Sim88], pages 11–19.

[CDM00] Ronald Cramer, Ivan Damgård, and Ueli M. Maurer.

General secure multi-party computation from any linear secret-sharing scheme.

In Bart Preneel, editor, EUROCRYPT 2000, volume 1807 of Lecture Notes in Computer Science, pages 316–334. Springer, Heidelberg, 2000.

参考文献 II

[CDN15] Ronald Cramer, Ivan Damgård, and Jesper Nielsen.
Secure Multiparty Computation and Secret Sharing.
Cambridge University Press, 2015.

[Sim88] Janos Simon, editor.
Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC '88). ACM, 1988.

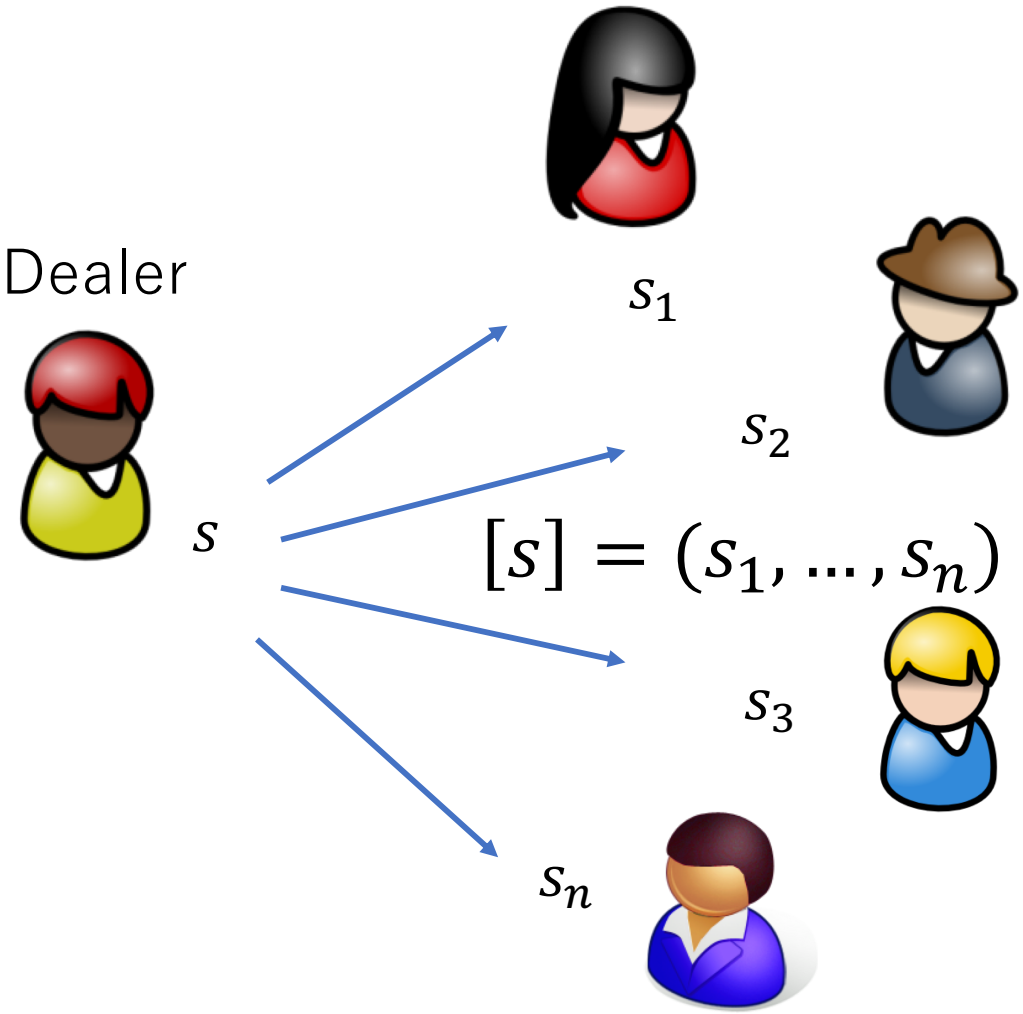
本日の講義の内容

1. 導入

2. Shamir 秘密分散

秘密分散

情報理論的安全な MPC での基本技術。秘密を分割して参加者間で保持。
t 人では秘密が全く漏れない。$t+1$ 人以上で秘密を復元できる。



$$[s] = (s_1, \ldots, s_n)$$

Dealer により、（n人の）参加者にシェアが分配された状態を示す

t+1人以上でsを復元できる

(t+1,n)-秘密分散

秘密分散 (Secret Sharing)

$(t+1, n)$-秘密分散 $\mathsf{SS} = (\mathsf{Share}, \mathsf{Recon}, \Gamma_{t+1,n})$ とは次のスペックを満たすもの。

Specification

- Share は秘密 $s \in K$（K は体）を入力に取り、n 個の share からなるベクトル $[s] := (s_1, \ldots, s_n)$ を出力する確率的アルゴリズム

$$\mathsf{Share} : s \in K \mapsto [s] := (s_1, \ldots, s_n) \in K^n$$

- $\Gamma_{t+1,n} \triangleq \{Q \subset \{1, \ldots, n\} \mid \#Q \geq t+1\}$（閾値型アクセス構造）
- Recon は、$[s]$ の t 個以上の要素を含む部分集合 $S_Q = \{s_i | i \in Q\}$（$Q \in \Gamma_{t+1,n}$）を入力に取り、秘密 s を復元する確定的アルゴリズム

上記 $\mathsf{SS} = (\mathsf{Share}, \mathsf{Recon}, \Gamma_{t+1,n})$ が秘密分散とは、次の条件を満たすものである。

Security

- (Perfect Reconstructability) 全ての $s \in K$, $[s] \leftarrow \mathsf{Share}(s)$, $Q \in \Gamma_{t+1,n}$ となる S_Q に対して、$\mathsf{Recon}(S_Q) = s$.
- (Perfect Privacy) 全ての $s \in K$, $[s] \leftarrow \mathsf{Share}(s)$, $T \notin \Gamma_{t+1,n}$ となる S_T に対して、$H(s) = H(s|S_T)$。すなわち、確率変数 s と S_T は独立。

線型秘密分散 (Linear Secret Sharing)

$\mathsf{SS} = (\mathsf{Share}, \mathsf{Recon}, \Gamma_{t+1,n})$ が線型秘密分散とは、次の条件を満たすものである。

Linearlity

- SS が秘密分散かつ、
- (Linearlity) 全ての $a, b, \lambda, \rho \in K$, $[a] \leftarrow \mathsf{Share}(a)$, $[b] \leftarrow \mathsf{Share}(b)$, $[\lambda a + \rho b] \leftarrow \mathsf{Share}(\lambda a + \rho b)$ に対して、

$$[\lambda a + \rho b] = \lambda[a] + \rho[b]$$

が成り立つ。ここで、$\lambda[a] := (\lambda a_1, \ldots, \lambda a_n)$, $[a] + [b] := (a_1 + b_1, \ldots, a_n + b_n)$ と定義。

すなわち、$P_1, \ldots, P_n$ 間で秘密 a, b をシェア状態 $[a]$,$[b]$ であるとき、各 P_i がローカルに手元で $\lambda a_i + \rho b_i$ を計算して（λ, ρ は既知）新たにシェアされる $(\lambda a_1 + \rho b_1, \ldots, \lambda a_n + \rho b_n)$ が秘密 $\lambda a + \rho b$ をシェアした状態 $[\lambda a + \rho b]$ になることを意味する。

（蛇足）線形秘密分散

Reconstruction で $\sim$ を導入してみるとわかりやすいかも

$$\mathrm{Share} : K \longrightarrow K^n/\sim$$

$$\cup$$

$$a \longmapsto [a] = (a_1, \cdots, a_n)$$

$$(a_1, \cdots a_n) \sim (a_1', \cdots a_n')$$

$$\Leftrightarrow [a] = [a']$$

Reconstruction で同じ値になる

linear

$$[a+b] = [a] + [b]$$

$$[\lambda a] = \lambda [a]$$

自明な (n, n)-線型秘密分散

K を有限体。秘密 $s \in K$.

自明な (n, n)-線型秘密分散

- Share(s):
  - 独立で一様ランダムに $s_1, \ldots, s_{n-1}$ を K から選ぶ。すなわち、$s_1, \ldots, s_{n-1} \ldots_R K$.
  - $s_n = s - \sum_{i=1}^{n-1} s_i \in K$ を計算。
  - $[s] = (s_1, \ldots, s_n)$ を出力する。
- Recon$(s_1, \ldots, s_n)$: $s = \sum_{i=1}^{n} s_i \in K$ を出力。

- (Perfect Reconstruction) 自明。
- (Perfect Privacy) どの $(n-1)$ 個のシェア $(s_{i_1}, \ldots, s_{i_{n-1}})$ をとっても、s に無関係で独立一様ランダムであるため、s と $(s_{i_1}, \ldots, s_{i_{n-1}})$ は独立。よって、Perfect Privacy を満たす。
- (Linearlity) 明らかに $[\lambda s + \rho s'] = \lambda[s] + \rho[s']$ を満たす。

Shamir 秘密分散

Shamir 秘密分散 SS $= (\mathsf{Share}, \mathsf{Recon}, \Gamma_{t+1,n})$. $\alpha_1, \ldots, \alpha_n \in K$ はシステムで予め定められた異なる値。

Shamir SS

- Share(s):
  - $a_0 := s$ の条件のもと K 係数のランダムな t 次多項式 $f(X) = \sum_{i=0}^{t} a_i X^i$ を選ぶ。

$$f(X) \leftarrow_R K[X] \quad \text{such that} \quad \deg(f) = t \text{ and } a_0 = s.$$

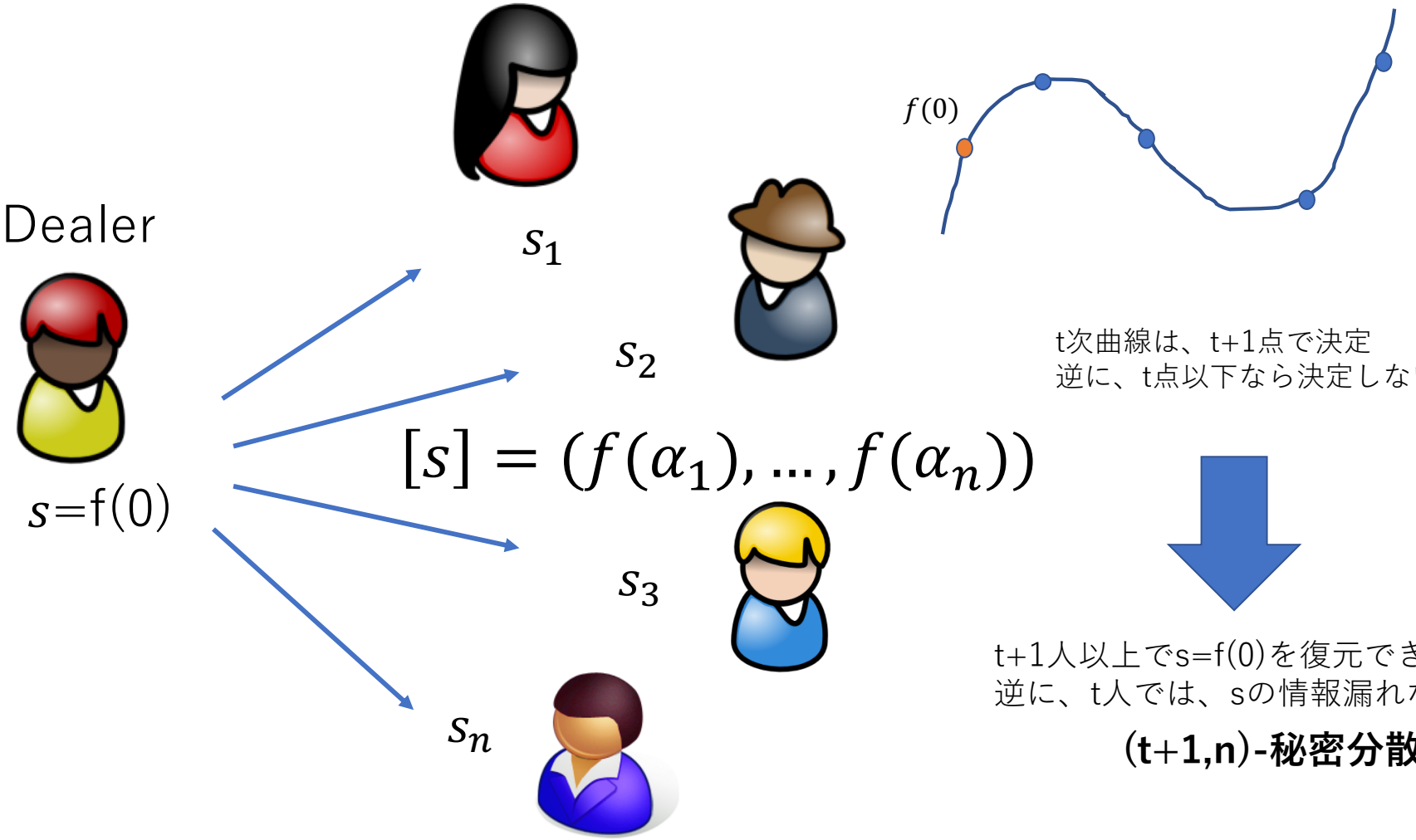
  - $[s] = (f(\alpha_1), \ldots, f(\alpha_n))$ を出力する。
- $\mathsf{Recon}(S_Q)$ $(S_Q = \{f(\alpha_i) | i \in Q \text{ s.t. } Q \in \Gamma_{t+1,n}\})$: s を出力。

$$s = \sum_{i \in Q} \lambda_{i,Q} f(\alpha_i) \text{ where } \lambda_{i,Q} = \prod_{j \in Q \setminus \{i\}} \left(\frac{\alpha_j}{\alpha_j - \alpha_i} \right).$$

Reconstruction vector $(\lambda_{1,Q}, \ldots, \lambda_{\#Q,Q})$ は、S_Q のみで決定することに注意

Shamir 秘密分散 (2)

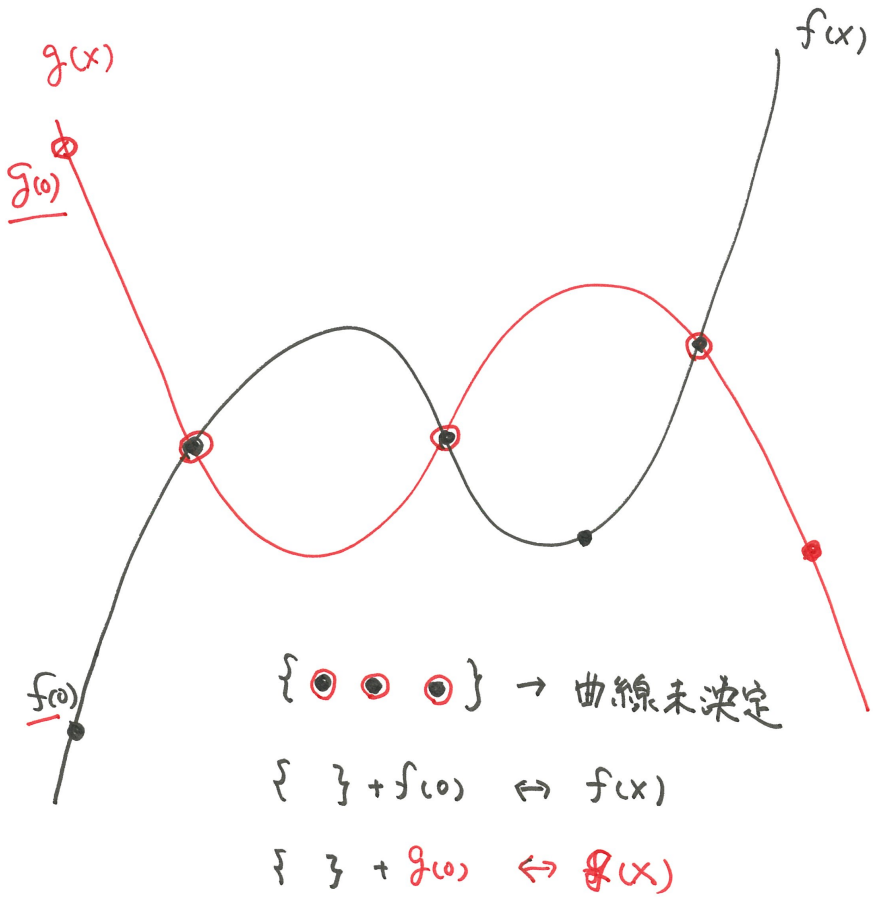
Dealer は、多項式 f を $s = f(0)$ かつ $\deg(f) = t$ となる条件でランダムに選ぶ。



多項式の決定から

$f(X)$ の決定 $\Longleftrightarrow$ 曲線の $\deg(f)+1$ 点の決定

- （$f(0)$ 以外の）$\deg(f)$ 点が漏れる　$\Longrightarrow$ 秘密 $f(0)$ の情報全くなし
- $\deg(f)+1$ 点を公開　$\Longrightarrow$ $f(X)$ が決定するので $f(0)$ が決定



Perfect Privacy

以下の $(f(\alpha_1), \ldots, f(\alpha_n))$ は完全同分布.

- Original: D は $s := a_0$ とし、$a_1, \ldots, a_t$ をランダムに決定（$f(X)$ が決定）。D は、$f(\alpha_1), \ldots, f(\alpha_n)$ を $P_1, \ldots, P_n$ にそれぞれ配る。
- $\mathrm{Dist}_{(\alpha_{i_1}, \ldots, \alpha_{i_t})}$: D は $s := f(0)$ とし、$f(\alpha_{i_1}), \ldots, f(\alpha_{i_t})$ をランダムに決定（$f(X)$ が決定）. D は、$f(\alpha_1), \ldots, f(\alpha_n)$ を $P_1, \ldots, P_n$ にそれぞれ配る。

$\{\alpha_{i_1}, \ldots, \alpha_{i_t}\}$ はどの組み合わせであっても Original の share の配り方と同分布。

$(a_0, a_1, \ldots, a_t) \Leftrightarrow (f(0), f(\alpha_{i_1}), \ldots, f(\alpha_{i_t}))$

$f(X) = a_0 + a_1 X + \cdots + a_t X^t$. Vandermonde 行列より正則なので一対一.

$$\begin{pmatrix} f(0) \\ f(\alpha_{i_1}) \\ \vdots \\ f(\alpha_{i_t}) \end{pmatrix} = \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 1 & \alpha_{i_1}^1 & \ldots & \alpha_{i_1}^t \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \alpha_{i_t}^1 & \ldots & \alpha_{i_t}^t \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_t \end{pmatrix}$$

Perfect Privacy: $S_T = \{f(\alpha_{i_1}), \ldots, f(\alpha_{i_t})\}$ は、$s = f(0)$ と無関係かつ独立に選ばれているので、これらの値が漏れても s の情報を含まない。

Vandermonde 行列

$$V = (\vec{v_1}, \ldots, \vec{v_t}) = \begin{pmatrix} 1 & \alpha_{i_1}^1 & \ldots & \alpha_{i_1}^{t-1} \\ 1 & \alpha_{i_2}^1 & \ldots & \alpha_{i_2}^{t-1} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \alpha_{i_\ell}^1 & \ldots & \alpha_{i_\ell}^{t-1} \end{pmatrix}.$$

$\alpha_{i_1}, \ldots, \alpha_{i_\ell}$ は相異なる値で、$\ell, t < \#K$ なら、列ベクトル $\{\vec{v_i}\}_i$ は線型独立。正方行列（$t = \ell$）なら V は正則。

$$\begin{pmatrix} f(\alpha_{i_1}) \\ f(\alpha_{i_2}) \\ \vdots \\ f(\alpha_{i_\ell}) \end{pmatrix} = \begin{pmatrix} 1 & \alpha_{i_1}^1 & \ldots & \alpha_{i_1}^{t-1} \\ 1 & \alpha_{i_2}^1 & \ldots & \alpha_{i_2}^{t-1} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \alpha_{i_\ell}^1 & \ldots & \alpha_{i_\ell}^{t-1} \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_{t-1} \end{pmatrix}$$

$a_0 = f(0)$ より、$f(0) = \sum_{i=1}^{\ell} \lambda_i f(\alpha_i)$ のような線形和になる。

Perfect Reconstruction

Lagrange 補間公式から Reconstruction algorithm を構成。
$f(X)$ を次数 $\deg(f) = t$ の多項式とする。任意の $n(\geq t+1)$ の異なる点 $\alpha_1, \ldots, \alpha_n$ の関数値を $f(\alpha_1), \ldots, f(\alpha_n)$ とすると、$f(X)$ は

$$f(X) = \sum_{i=1}^{n} \lambda_{i,n}(X) \cdot f(\alpha_i) \text{ where } \lambda_{i,n}(X) = \prod_{j=1, j\neq i}^{n} \left(\frac{X - \alpha_j}{\alpha_i - \alpha_j} \right)$$

と、$n(\geq t+1)$ の個数と $\alpha_1, \ldots, \alpha_n$ の選び方によらず一意に復元される。

$$\lambda_i(\alpha_j) = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j. \end{cases}$$

かつ、

$$s = f(0) = \sum_{i=1}^{n} \lambda_{i,n}(0) f(\alpha_i)$$

に注意。$(\lambda_{1,n}, \ldots, \lambda_{n,n}) := (\lambda_{1,n}(0), \ldots, \lambda_{n,n}(0))$ を、Shamir SS の $\alpha_1, \ldots, \alpha_n$ での reconstruction vector と呼ぶ.

多項式補間

Theorem 1

K を体とする。任意の $\{(\alpha_i, y_i)\}_{i=1}^{n}$ $(\alpha_i, y_i \in K)$ に対して（ただし $\alpha_i \neq \alpha_j$）、

- $y_i = F(\alpha_i)$ *for* $i = 1, \ldots, n$
- $\deg(F(X)) \leq n - 1$

となる多項式関数 $Y = F(X)$ が一意に定まる。

この定理（と証明）から

- n 点から補間される Lagrange 多項式 $F(X)$ の次数は $n-1$ 以下（見せかけの次数は $n-1$）。
- $\deg(f(X)) = t < n - 1$ とする。$\{(\alpha_i, f(\alpha_i)\}_{i=1}^{n}$ から補間された Lagrange 多項式を $F(X)$ とすると、$F(X) \equiv f(X)$（Reconstruction の一意性）。

Theorem 1 の証明

$Q = \{1, \ldots, n\}$ とする。全ての x_i $(i \in Q)$ に対して、$F(\alpha_i) = f(\alpha_i)$ となるような多項式を一つ考える。$F(X) = \sum_i \lambda_{i,Q}(X) \cdot f(\alpha_i)$ とおくと

$$\lambda_{i,Q}(\alpha_j) = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j. \end{cases}$$

なら条件を満たす。そのような $\lambda_{i,Q}(X)$ は、

$$\begin{aligned} \lambda_{i,Q}(X) &= \frac{\prod_{j \in Q \setminus \{i\}} (X - \alpha_j)}{\prod_{j \in Q \setminus \{i\}} (\alpha_i - \alpha_j)} \\ &= \frac{(X - \alpha_1) \ldots (X - \alpha_{i-1}) \cdot \cancel{(X - \alpha_i)} \cdot (X - \alpha_{i+1}) \ldots (X - \alpha_n)}{(\alpha_i - \alpha_1) \ldots (\alpha_i - \alpha_{i-1}) \cdot \cancel{(\alpha_i - \alpha_i)} \cdot (\alpha_i - \alpha_{i+1}) \ldots (\alpha_i - \alpha_n)} \end{aligned}$$

が条件を満たす。さらに、$\deg(\lambda_{i,Q}) \leq n - 1$ より、$\deg(F) \leq n - 1$.

Theorem 1 の証明（続き）

$F(X)$ 以外にも条件を満たす t 次（以下の）多項式 $G(X)$ が存在すると仮定。すなわち

- $y_i = F(\alpha_i) = G(\alpha_i)$ for $i = 1, \ldots, n$
- $\deg(F(X)), \deg(G(X)) \leq n - 1$

今、$H(X) \triangleq F(X) - G(X)$ を考えると、全て $i \in Q$ に対して、$H(\alpha_i) = 0$. H の次数は高々 $n - 1$ であるから、n 個の α_i を根に持つためには、$H(X) \equiv 0$ が必要。よって、$F(X) \equiv G(X)$ であり、このような多項式は、$n - 1$ 次以下では一意に決定する。

Shamir 秘密分散は線型

$f, g \in K[X]$ $(\deg(f), \deg(g) \leq t)$ に対して次のことは明らか。

線型性

$$[\alpha f(0) + \beta g(0)] = \alpha[f(0)] + \beta[g(0)] \qquad \text{where } \alpha, \beta \in K.$$

$$f(X) = a_0 + a_1 X + \ldots a_t X^t \text{ および } g(X) = b_0 + b_1 X + \ldots b_t X^t$$

とする。$h(X) \triangleq \alpha f(X) + \beta g(X)$ と定義すると、$\deg(h) \leq t$ で、

$$\begin{aligned}
[h(0)] &= (h(\alpha_1), \ldots, h(\alpha_n)) \\
&= (\alpha f(\alpha_1) + \beta g(\alpha_1), \ldots, \alpha f(\alpha_n) + \beta g(\alpha_n)) \\
&= \alpha[f(0)] + \beta[g(0)]
\end{aligned}$$