# [I486S]
# 暗号プロトコル理論

藤崎 英一郎

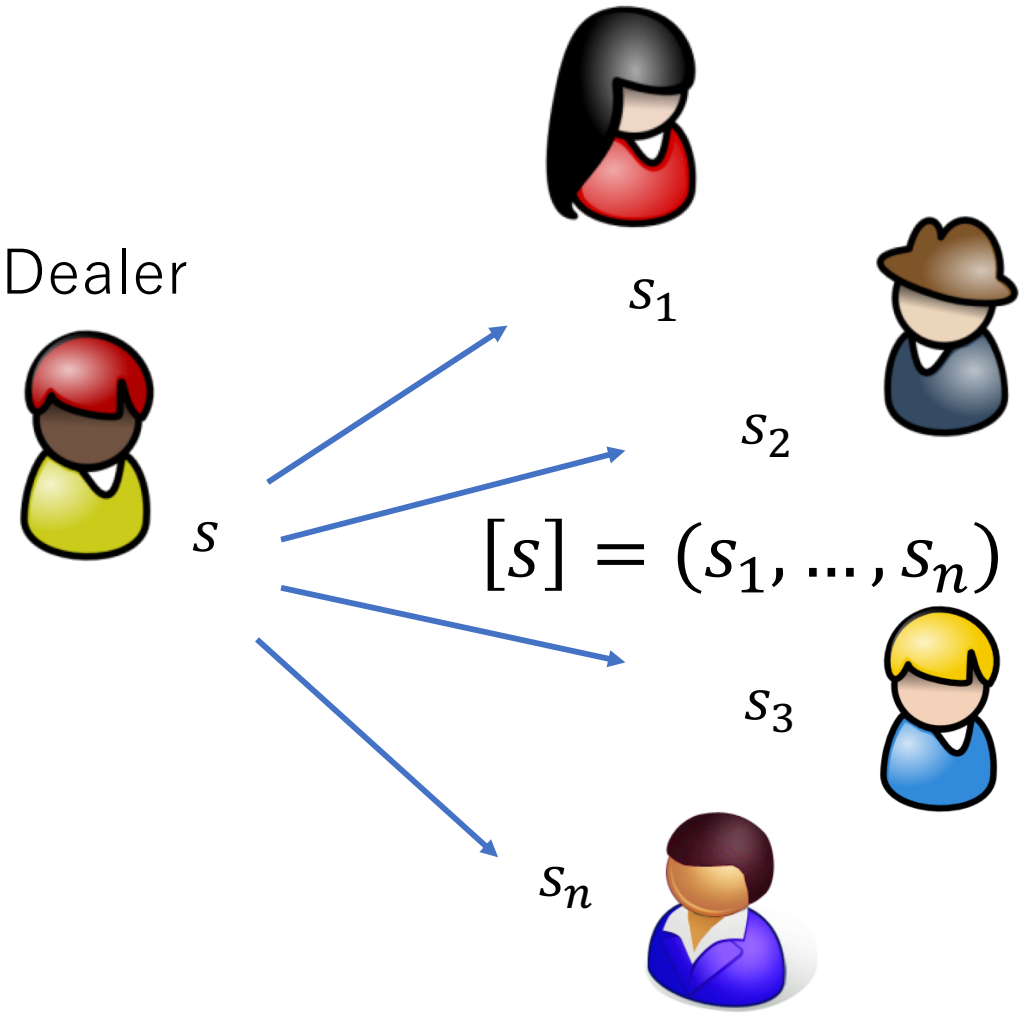
北陸先端科学技術大学院大学

2020 年 5 月 19 日

# 本日の講義の内容

# 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。




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

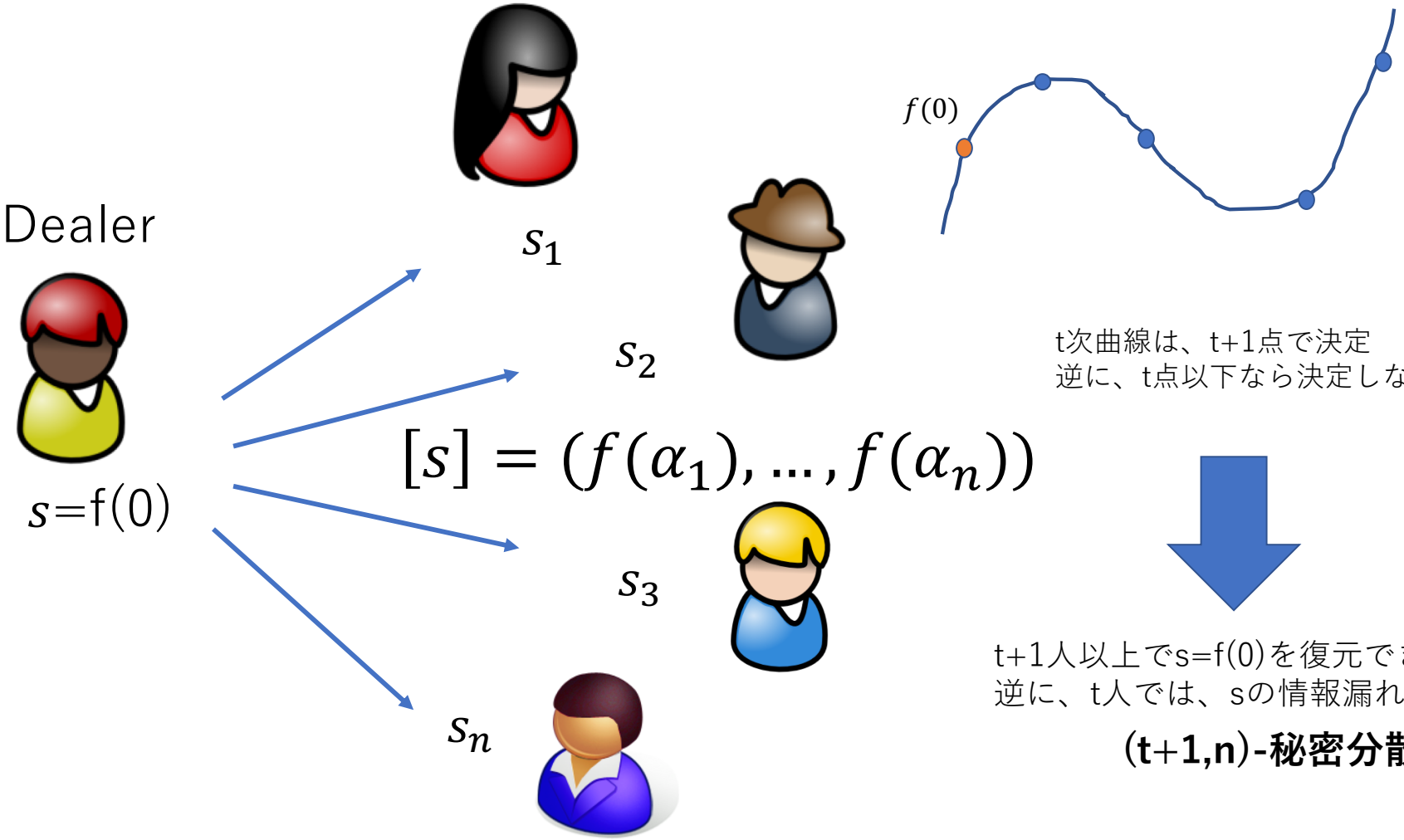
Dealer により、（n人の）参加者にシェアが分配された状態を示す

t+1人以上でsを復元できる

**(t+1,n)-秘密分散**

# Shamir 秘密分散

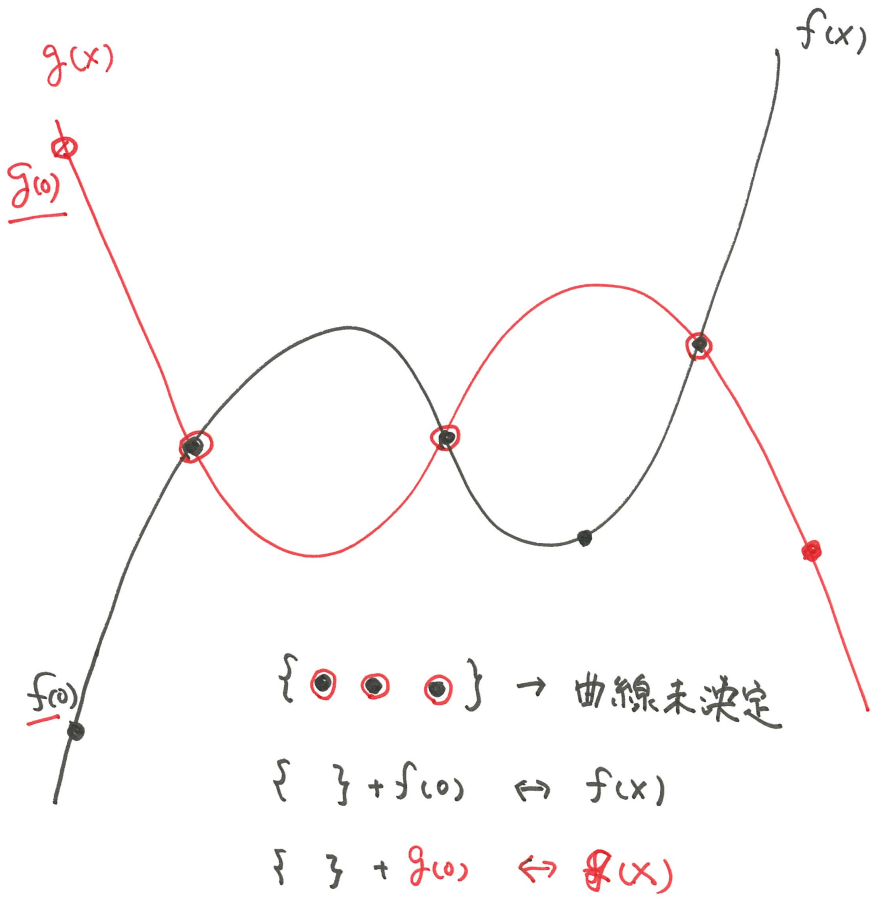
Dealer は、多項式 $f$ を $s = f(0)$ かつ $\deg(f) = t$ となる条件でランダムに選ぶ。



# 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



# Shamir 秘密分散 (formal)

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

## Shamir SS

- $\mathsf{Share}(s)$:
  - $a_0 := s$ 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

# 本日の講義の内容

# 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（$\lambda, \rho$ 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。

# 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、

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

# Shamir SS の足し算

線形性から簡単に計算できる。

## addition

$[a]$, $[b]$: $a, b \in K$ が $P_1, \ldots, P_n$ 間でシェアされた状態

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

$f, g$ をそれぞれ $a, b$ をシェアする時の $t$ 次多項式とする。$f_0 = a$, $g_0 = b$ で、

$$f(X) = f_0 + f_1 X + \ldots + f_t X^t, \text{ and } g(X) = g_0 + g_1 X + \ldots + g_t X^t.$$

$h(X) = f(X) + g(X)$ と置くと、$h(0) = a + b$ と $\deg(h) = t$ より

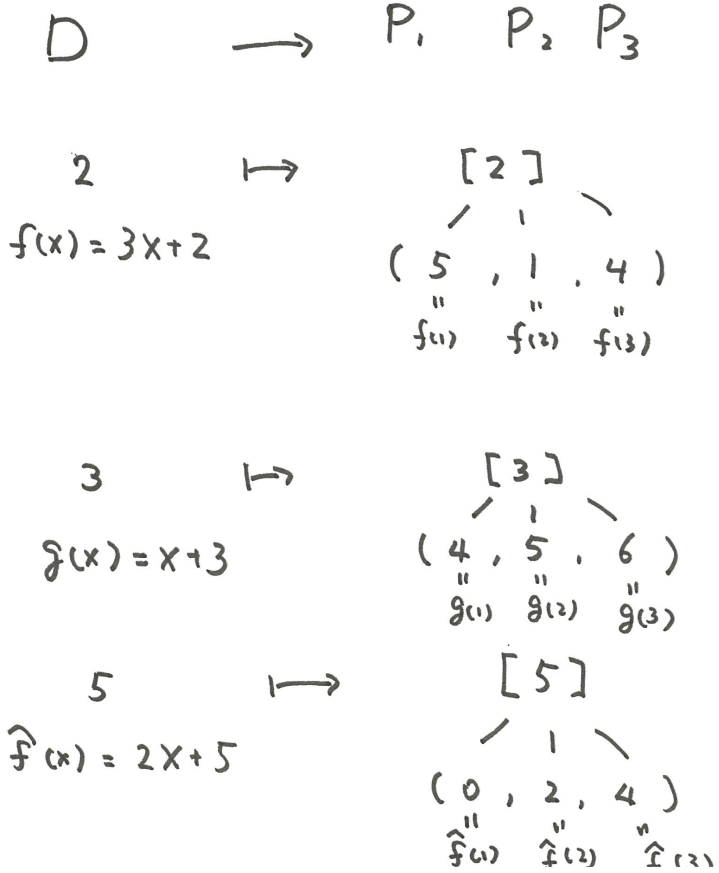
$$[a + b] = (h(\alpha_1), \ldots, h(\alpha_n))$$

$h(\alpha_i) = f(\alpha_i) + g(\alpha_i)$ であるから、各 $P_i$ がローカルに自分のシェアを足し算すれば、$[a + b]$ と言う状態になる。

# 線型性

例) Shamir SS での線型性の確認

$K = \mathbb{Z}/7\mathbb{Z}$




[2]+[3]

def (5+4, 1+5, 4+6)

(f(1)+g(1), f(2)+g(2), f(3)+g(3))

= [5] ← 秘密を復元した時に同じという意味

= (0, 2, 4)

# 本日の講義の内容

# Shamir SS の掛け算（試み）

掛け算は少し工夫がいる。

## multiplication

$[a]_{(t)}$, $[b]_{(t)}$: $a, b \in K$ が $P_1, \ldots, P_n$ 間で $(t+1, n)$-SS でシェアされた状態.

$$[a]_{(t)} \cdot [b]_{(t)} = [ab]_{(2t)}$$

が成り立つ。ただし、$[a] \cdot [b] := (a_1 b_1, \ldots, a_n b_n)$ と定義（各 $P_i$ がローカルに自分のシェアを掛け合わせた状態）。

$f, g$ をそれぞれ $a, b$ をシェアする時の $t$ 次多項式とする。$f_0 = a$, $g_0 = b$ で、

$$f(X) = f_0 + f_1 X + \ldots + f_t X^t, \text{ and } g(X) = g_0 + g_1 X + \ldots + g_t X^t.$$

$h(X) = f(X)g(X)$ と置くと、$h(0) = ab$ と $\deg(h) = 2t$ より

$$[ab]_{(2t)} = (f(\alpha_1)g(\alpha_1), \ldots, f(\alpha_n)g(\alpha_n))$$

よって、$P_1, \ldots, P_n$ 間で $ab$ をシェアした状態になるが、$2t+1$ 個のシェアが集まらないと $ab$ が復元できない。

# Shamir SS の掛け算（アイデア）

## multiplication

Lagrange と線型性により

$$[ab] = [h(0)] = [\sum_{i=1}^{n} \lambda_{i,n} h(\alpha_i)] = \sum_{i=1}^{n} \lambda_{i,n} [h(\alpha_i)]$$

$(\lambda_{1,n}, \ldots, \lambda_{n,n})$ は $\{\alpha_1, \ldots, \alpha_n\}$ のみから決まる。

$h(X) = f(X)g(X)$ であるから、$h(\alpha_i) = f(\alpha_i)g(\alpha_i)$. よって、$[a]$, $[b]$ の状態から、$[ab]$ を作るには各 $P_i$ がローカルに $a_i b_i = f(\alpha_i)g(\alpha_i)$ を計算した後、その値のシェアを他の参加者に $(t+1, n)$-線型秘密分散で配り $[f(\alpha_i)g(\alpha_i)]$ という状態を作り出せば良い。後は線型性からローカルな計算で $[ab]$ の状態に持っていく。

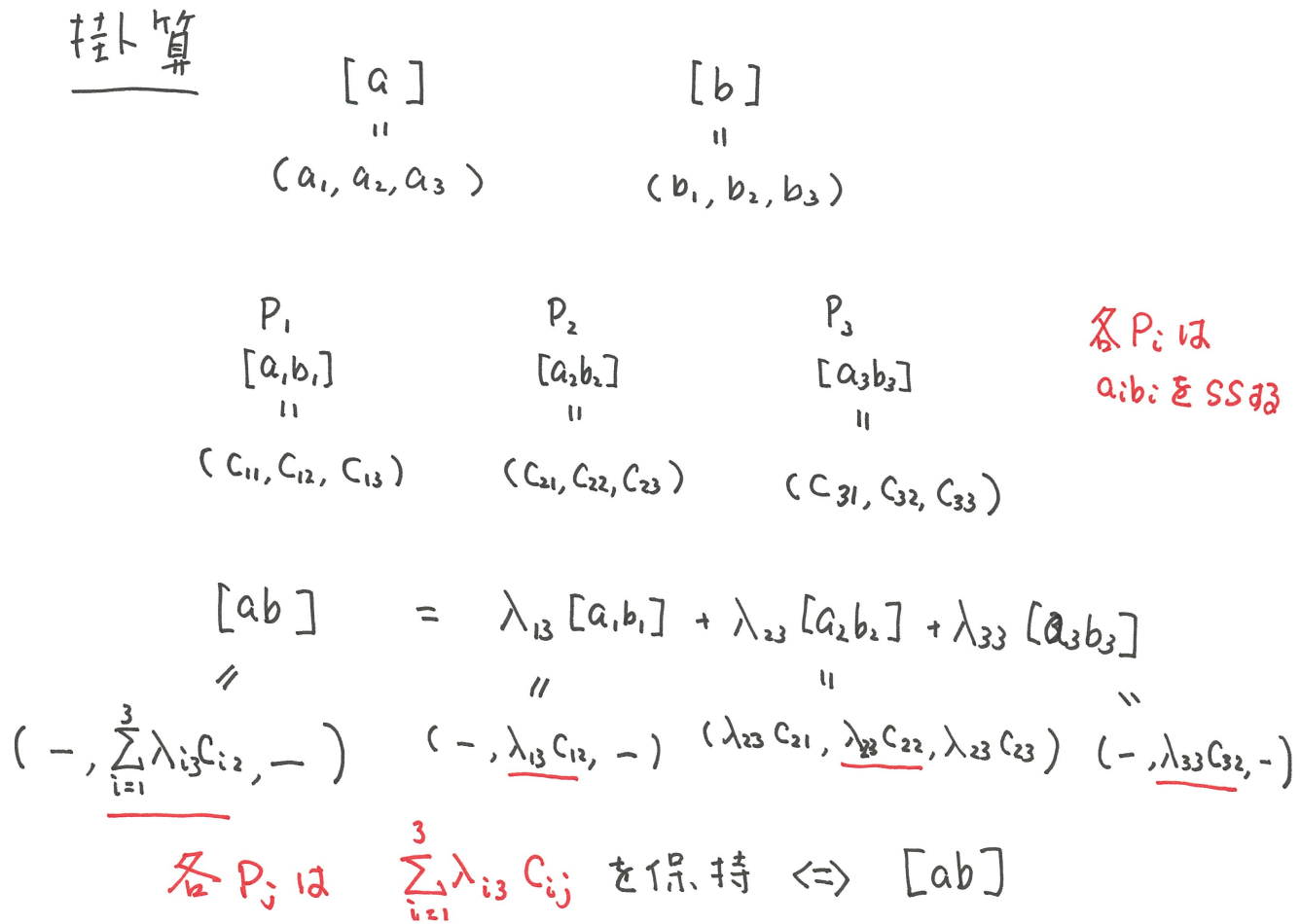
# Shamir SS の掛け算（まとめ）

- Input: $[a], [b]$: $a, b$ がそれぞれ $P_1, \ldots, , P_n$ 間でシェアされている
- Output: $[ab]$: $ab$ が $P_1, \ldots, , P_n$ 間でシェアされる

1. 各 $P_i$ が $c_i = a_i b_i$ をローカルに計算
2. 各 $P_i$ が $c_i$ を $P_1, \ldots, P_n$ 間で線型秘密分散。$[c_1], \ldots, [c_n]$ の状態になる。$P_i$ は、$c_{1,i}, \ldots, c_{i,i}, \ldots, c_{n,i}$ を、$[c_1], \ldots, [c_n]$ の自分のシェアとして持つ。
3. $i = 1, \ldots, n$ に対して、$\lambda_{i,n} = \prod_{j \neq i} \frac{\alpha_i}{\alpha_j - \alpha_i}$ を計算する。
4. 各 $P_i$ が $d_i = \sum_{j=1} \lambda_{j,n} c_{j,i}$ を計算する。
5. $[ab] = (d_1, \ldots, d_n)$ なので、$[ab]$ の状態になる。

# 補足

$[a]$, $[b]$ の状態から


掛算
[a]
=
(a1, a2, a3)
[b]
=
(b1, b2, b3)
P1
[a1b1]
=
(c11, c12, c13)
P2
[a2b2]
=
(c21, c22, c23)
P3
[a3b3]
=
(c31, c32, c33)
各Piは
aibiをSSする
[ab] = λ13[a1b1] + λ23[a2b2] + λ33[a3b3]
=
(-, Σ_{i=1}^{3} λi3 ci2, -)
(-, λ13 c12, -)
(λ23 c21, λ23 c22, λ23 c23)
(-, λ33 c32, -)
各Pjは Σ_{i=1}^{3} λi3 cij を保持 <=> [ab]

# Shamir SS による MPC の線型和と積

初期状態 $[a] = (a_1, \ldots, a_n)$, $[b] = (b_1, \ldots, b_n)$: すなわち $a, b \in K$ が $P_1, \ldots, P_n$ 間でシェアされた状態.

## 線形和と積

$$[a] + [b] = [a + b], \qquad -[a] = [-a]$$

線型性から、各 $P_i$ がローカルに $a_i + b_i$ を計算すれば、$[a + b]$ という状態になる。また各 $P_i$ が $-a_i$ を計算すれば $[-a]$ になる。

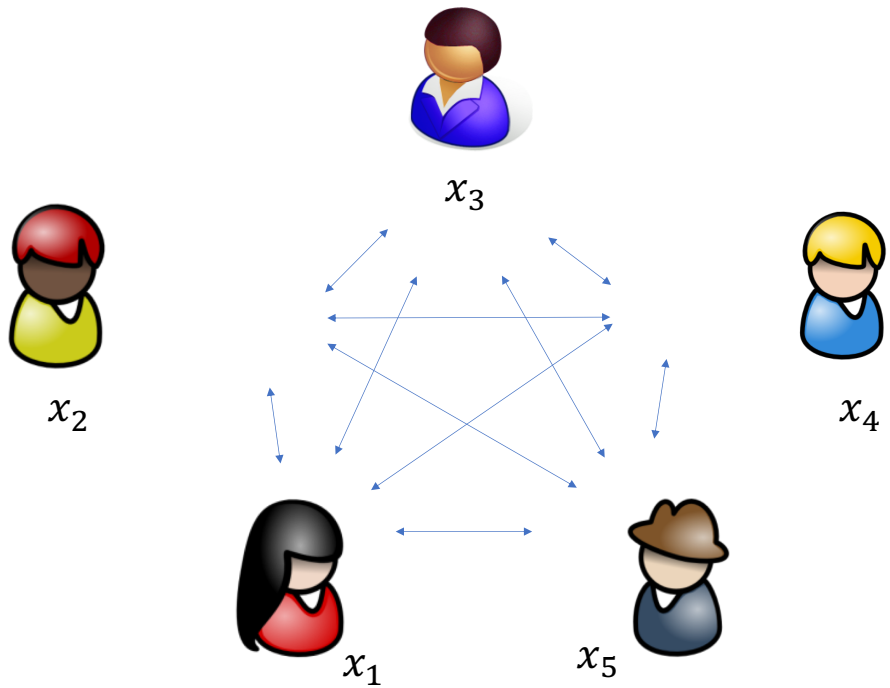
$$[ab] = [\sum_{i=1}^{n} \lambda_{i,n} a_i b_i] = \sum_{i=1}^{n} \lambda_{i,n} [a_i b_i].$$

$(\lambda_{1,n}, \ldots, \lambda_{n,n})$ は $\{\alpha_1, \ldots, \alpha_n\}$ のみから決まる。各 $P_i$ がローカルに $a_i b_i$ を計算後、その値を $(t+1, n)$-線型秘密分散で $[a_i b_i]$ を行う。後は線型性からローカルな計算で $[ab]$ という状態にできる.

# 本日の講義の内容

# 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: Pair-wise private & synchronized.



# Secure MPC against Passive Adversaries

- 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: 関数 $F : \{0,1\}^{n\lambda} \to \{0,1\}^*$.
- 各 $P_i$ への出力：$F(x_1, \ldots, x_n)$.
- パラメータ：$t, n$ $(t < n/2)$
- 不正者：passive, $\mathcal{A}_{t,n}(\triangleq \{A \subset \{1, \ldots, n\} \mid \#A \leq t\})$.

## Theorem 1

*There is an efficient MPC protocol to evaluate any efficiently computable function $F$ such that the following conditions hold:*

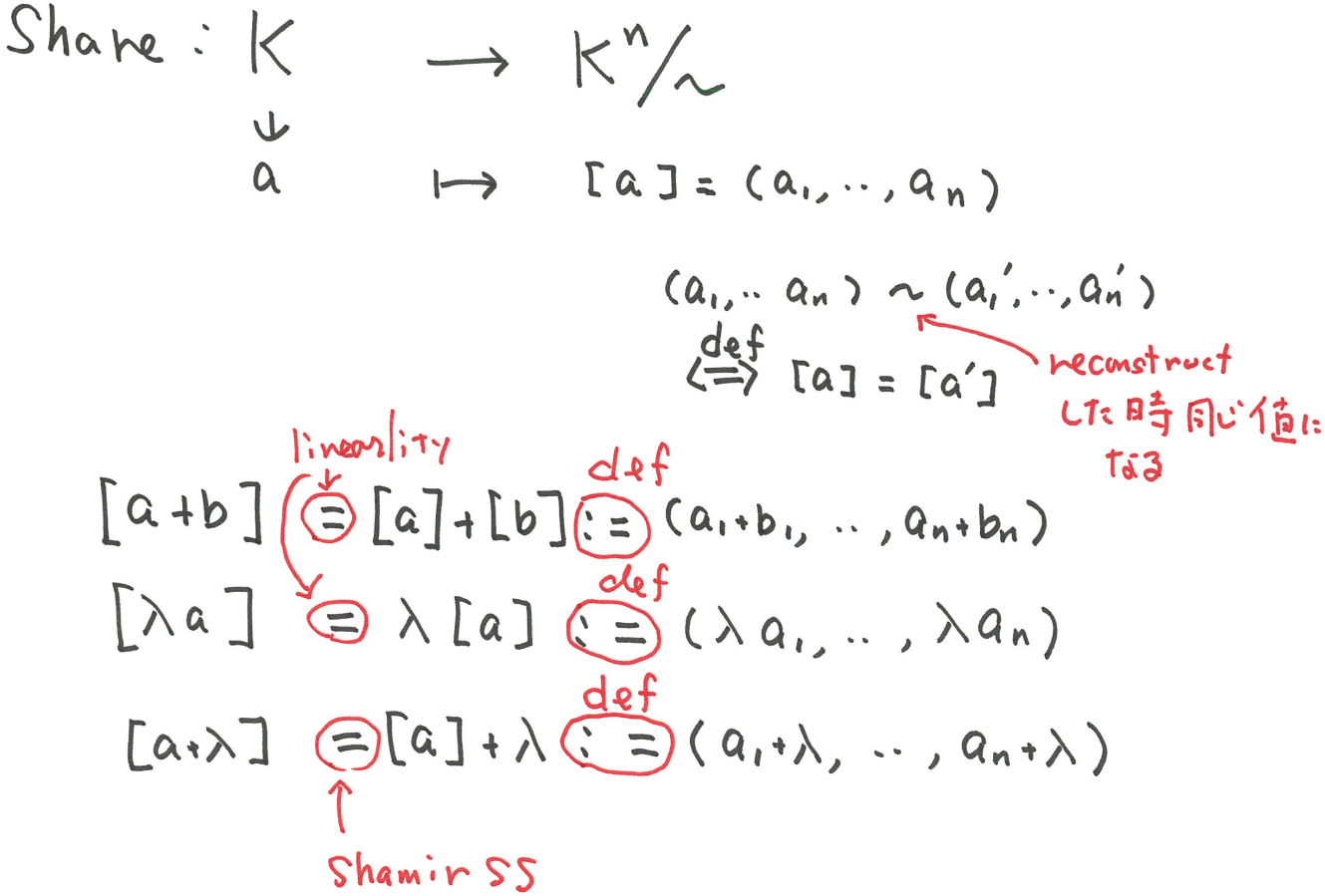
- *(Perfect Correctness) All players receive $F(x_1, \ldots, x_n)$ with prob. 1.*
- *(Perfect Privacy) Any passive $\mathcal{A}_{t,n}$-adversary with $t < n/2$ learns no information beyond $\{x_i\}_{i \in \mathcal{A}_{t,n}}$ and $F(x_1, \ldots, x_n)$ from executing the protocol regardless of their computing power and memory.*

# MPC の構成（準備）

準備

- $F : \{0,1\}^{n\lambda} \to \{0,1\}^*$: 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。
  - $b \wedge b' \iff b \cdot b' \in K$.
  - $b \vee b' \iff b + b - b \cdot b' \in K$.
  - $\neg b \iff 1 - b \in K$.

# 線形性と ShamirSS の性質があると

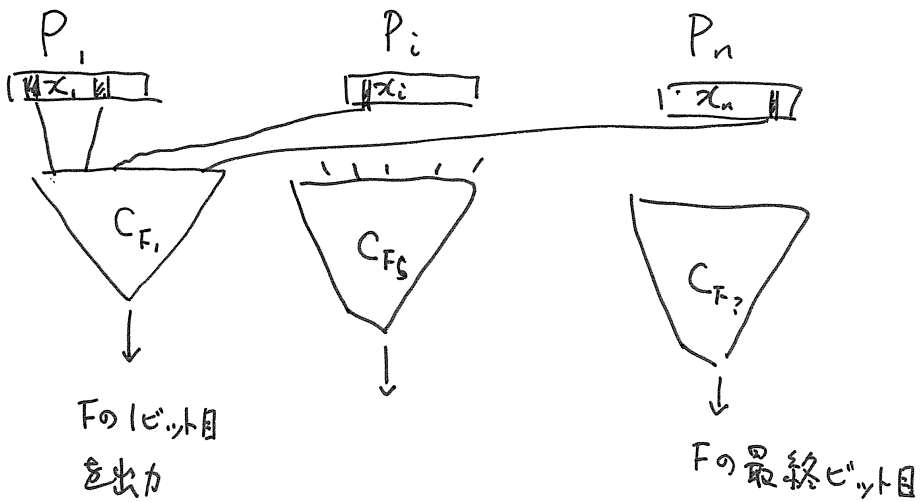


# 関数 $F$ の回路

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

($n$ 個)

回路に分解



# MPC の構成

- Input sharing phase: 各参加者 $P_i$ は、$x_i$ の各ビット $b$ を $(t+1, n)$-Shamir SS を使い $[b]$ の状態にする。この結果、全ての参加者の秘密の全てのビットが（線型）秘密分散される。
- Computation phase: 各論理ゲートを秘密分散した形で実行
  - $[a] + [b] = [a + b]$
  - $1 - [a] = [1 - a]$
  - $[ab] = \sum_{i=1}^{n} \lambda_{i,n} [a_i b_i]$
- Output reconstruction phase: 出力ゲート結果が秘密分散された状態になっているので、各 $P_i$ は自分の出力ゲート結果に関するシェアを他の参加者に公開（全員に送る）。各 $P_i$ は、出力結果を復元する。

# 本日の講義の内容

# Perfect Privacy I

- 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。
- 任意の $\{\alpha_{i_1}, \ldots, \alpha_{i_t}\} (\subset \{\alpha_1, \ldots, \alpha_n\})$ に対して
  $S_T = \{f(\alpha_{i_1}), \ldots, f(\alpha_{i_t})\}$ は、$s = f(0)$ と無関係かつ独立になり、よって、これらの値が漏れても $s$ の情報を含まない。

# Perfect Privacy II

(証明) 秘密 $s$ が、$S_T$（ここで、$T \notin \Gamma_{t+1,n}$）と独立なら良い。$a_1, \ldots, a_t$ は独立にランダムに選ばれているので、$i = 1, \ldots, t$ $(\ell \leq t)$ に対して、$f(\alpha_i)$ は、$f(0)$ を

$$K_i = \sum_{j=1}^{t} \alpha_i^j a_j$$

というランダムな値で one-time pad した

$$f(\alpha_i) = f(0) + K_i$$

形になる。さらに $K_1, \ldots, K_\ell$ は互いに独立になる

$$\begin{pmatrix} f(\alpha_1) \\ \vdots \\ f(\alpha_t) \end{pmatrix} = \begin{pmatrix} f(0) \\ \vdots \\ f(0) \end{pmatrix} + \begin{pmatrix} \alpha_1 & \alpha_1^2 & \ldots & \alpha_1^t \\ \ldots\ldots\ldots\ldots & & & \\ \alpha_\ell & \alpha_\ell^2 & \ldots & \alpha_\ell^t \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_t \end{pmatrix}$$

よって、$f(0)$ と $S_T$ は互いに独立。

# 多項式補間

## Theorem 2

$K$ 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 2 の証明

$Q = \{1, \ldots, n\}$ とする。全ての $x_i$ $(i \in Q)$ に対して、$F(\alpha_i) = y_i$ となるような多項式を一つ考える。$F(X) = \sum_i \lambda_{i,Q}(X) \cdot y_i$ 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 2 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 $\alpha_i$ 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。

# 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.