階層的秘密分散法の高速化に関する研究

島 幸司† 土井 洋†

†情報セキュリティ大学院大学
221-0835 神奈川県横浜市神奈川区鶴屋町 2-14-1
mgs145506@iisec.ac.jp, doi@iisec.ac.jp

あらまし 秘密分散法は1979年にBlakleyとShamirによりそれぞれ独自に提案された．Shamirの(k,n)しきい値法は理想的秘密分散法であるが，計算コストが高いため，計算コストの軽量な手法が提案されている．一方で，参加者をレベルに分割し，そのレベルで分割された参加者のグループ間で秘密を共有する階層的秘密分散法が知られている．本稿では，藤井らの排他的論理和の手法を適用した階層的秘密分散法を提案する．また，Tassaの導関数とバーコフ補間を使ったアイデアを継承し，拡大体での高速化手法を提案する．

A study on fast hierarchical secret sharing schemes

Koji Shima† Hiroshi Doi†

†Institute of Information Security
2-14-1 Tsuruya-cho, Kanagawa-ku, Yokohama-city, Kanagawa 221-0835, JAPAN
mgs145506@iisec.ac.jp, doi@iisec.ac.jp

Abstract Blakley and Shamir independently introduced the concept of (k,n)-threshold secret sharing scheme in 1979. Shamir's (k,n)-threshold scheme is an ideal secret sharing scheme, but it requires expensive computational cost. For that reason, several lightweight methods have been proposed. On the other hand, hierarchical secret sharing scheme is known in the way that the secret is shared among a group of participants that is partitioned into levels. For hierarchical secret sharing scheme, we apply it to Fujii et al.'s method of using XOR. Also, we inherit Tassa's idea of using derivatives and Birkhoff interpolation, and we propose a faster method in an extension field.

1 はじめに

秘密情報を分散管理するための方法として秘密分散法が知られている．1979 年に Blakley と Shamir はそれぞれ独自に(k,n)しきい値法と呼ばれる秘密分散法の概念を提案した[1][2]. 秘密情報をn個のシェアに分散し，任意のk個のシェアを集めれば元の秘密情報を復元でき，$k-1$個のシェアからは元の秘密情報に関する情報が全く得られないという特徴がある．このため，シェアの一部が漏えいしても元の秘密情報は安全であり，かつ，シェアの一部が紛失しても元の秘密情報を復元できる．

一方で，参加者をレベルに分割し，そのレベ

ルで分割された参加者のグループ間で秘密を共有する階層的秘密分散法が知られている．金庫を開けるには 3 人の従業員が必要で，少なくとも一人は部長といったシナリオに見られるように，最小限の高いレベルの参加者が必要とされる秘密分散法である．Tassa は導関数の導入とバーコフ補間問題に注力している[3][4]．また，必須参加者を導入すれば，秘密消去の容易性を狙える．すなわち，秘密情報の削除は必須サーバ内のシェアの削除で保証されるからである．このとき，運用ミスや必須サーバの故障に備えて，必須サーバが 2 台にできることを視野に入れる．

1.1 秘密分散法の高速化手法

Shamir の(k,n)しきい値法は$k \leq n$を満たす任意のkとnに対して実現可能であるが，秘密情報の分散および復元において，$k-1$次多項式の処理に多倍長整数演算を用いた法p上の演算が必要であり，計算コストが大きい．藤井らは排他的倫理和演算のみを用いて秘密情報の分散および復元を行うことができる$(2,n)$しきい値法を提案した[5]．栗原らは藤井らの$(2,n)$しきい値法を最も重要な関連研究として排他的論理和演算のみを用いた$(3,n)$しきい値法を提案し，nがあまり大きくなければ，Shamir のしきい値法と比較して非常に効率が良いとの結果を得た[6]．さらに，栗原らは排他的論理和演算のみを用いた(k,n)しきい値法を提案した [7].

1.2 関連の高速化手法

一般に，秘匿関数計算の計算コストは高い．千田らは 3 主体の協調計算により，主体間の結託はないと仮定したとき，semi-honest モデルにおいて入力値を秘匿でき，malicious モデルにおいて演算結果の改ざんを従来よりも効率良く検出できる 3 パーティ秘匿関数計算プロトコルを提案した[8].

秘密情報を復元せずにシェアを更新するプロアクティブ秘密分散法は，Shamir の(k,n)しきい値法で使用する多項式の定数項を保持し，定期的に係数を更新する[9]．神宮らは古田らの方式[10]に対して，従来のプロアクティブ秘密分散法と比較を行い，プロアクティブ秘密分散法と同等の機能を持たせるように拡張し，計算量や通信量の小さい方式を提案した[11].

五十嵐らは消失訂正符号と関連して，GF(2^{64})上の演算の高速化を提案した[12]．拡大体の除算についても，拡張ユークリッド互除法のほか，伊東・辻井アルゴリズム[13]などの高速化手法が知られている．

1.3 本研究の貢献

本研究では，先に示したシナリオ上，有益と考えられる

・レベルの高い権限者が最低でも 1 名，かつ
・レベルの高い権限者は最大 2 名

を満たす，高速かつメモリ使用量が少ない階層的秘密分散法について検討した．具体的には，藤井らの排他的論理和の手法への適用，Tassa の導関数とバーコフ補間を使ったアイデアへの拡大体 GF(2^l) の適用のそれぞれについて検討した．本稿では，主に高速化に係る部分について述べる．

2 準備

2.1 拡大体GF(2^4)の演算

本稿の 5 章で例示するGF(2^4)の記述法について述べる．

原始多項式x^4+x+1の根 α から生成すると，$\alpha^{2^4-1}=\alpha^{15}=1$である．加算は排他的論理和であり，減算は加算に置き換えてよい．乗算は$\alpha^i\alpha^j=\alpha^{(i+j) \bmod 15}$，除算は$\alpha^i$の逆元$\alpha^{-i}=\alpha^{(15-i) \bmod 15}$の積で演算する．たとえば，$\alpha^4+\alpha+1=0$から$\alpha^4=\alpha+1$であり，指数表現$\alpha^4$はベクトル表現(0011)で表され，ベクトル表現を二進数で見ると 3 である（**図 1**）．

指数表現	ベクトル表現
0	(0000)
1	(0001)
α^1	(0010)
α^2	(0100)
α^3	(1000)
α^4	(0011)
α^5	(0110)
α^6	(1100)

指数表現	ベクトル表現
α^7	(1011)
α^8	(0101)
α^9	(1010)
α^{10}	(0111)
α^{11}	(1110)
α^{12}	(1111)
α^{13}	(1101)
α^{14}	(1001)

図 1 $\mathrm{GF}(2^4)$ の指数表現とベクトル表現

3 秘密分散法と関連研究

秘密分散法は秘密情報の分散フェーズと秘密情報の復元フェーズからなる．分散では，秘密情報を持つ 1 人のディーラがその秘密情報から多数のシェアと呼ばれる分散情報を作成し，各シェアを参加者に秘密裏に送る．復元では，あらかじめ定められた参加者の集合のシェアを集め，元の秘密情報を復元する．

n人の参加者の集合を$\boldsymbol{\mathcal{P}} = \{P_1, \dots, P_n\}$，秘密情報の集合を$\boldsymbol{S}$，$P_i \in \boldsymbol{\mathcal{P}}$が受け取りうるシェアの集合を$\boldsymbol{\mathcal{W}}_i$とする．ディーラ $\mathrm{D} \notin \boldsymbol{\mathcal{P}}$は秘密情報$s \in \boldsymbol{S}$を選択し，参加者$P_i$にシェア$w_i \in \boldsymbol{\mathcal{W}}_i$を秘密裏に送る．

秘密情報を復元する権限を持つアクセス構造$\Gamma \subset 2^{\boldsymbol{\mathcal{P}}}$とする．$(k, n)$しきい値法での$\Gamma$は次のように定義される．

$$\Gamma = \{A \in 2^{\boldsymbol{\mathcal{P}}} \mid |A| \geq k\}$$

sとw_iを含む確率変数をそれぞれSとW_i，$\boldsymbol{\mathcal{V}}_{\mathbf{A}} = \{W_i | P_i \in A, A \subset \boldsymbol{\mathcal{P}}\}$とすると，$\Gamma$が次の性質を満たすとき，完全秘密分散法という．

$$H(S|\boldsymbol{\mathcal{V}}_{\mathbf{A}}) = \begin{cases} 0 & (A \in \Gamma) \\ H(S) & (A \notin \Gamma) \end{cases}$$

$\boldsymbol{S}$と$\boldsymbol{\mathcal{W}}_i$の確率分布が一様であるとき，

$$\rho = \frac{log_2|\boldsymbol{S}|}{\max_{P_i \in \boldsymbol{\mathcal{P}}}(log_2|\boldsymbol{\mathcal{W}}_i|)}$$

の情報率を測定できることが知られており，$\rho = 1$を満たす完全秘密分散法を理想的秘密分散法という[14]．つまり，各分散情報のビット長は秘密情報のビット長よりは小さくできないが，これらのビット長が等しい場合，理想的秘密分散法となる．

Shamir の(k, n)しきい値法，藤井らの$(2, n)$しきい値法，栗原らの$(3, n)$しきい値法や(k, n)しきい値法はいずれも理想的秘密分散法である．

3.1 Shamir の$(\boldsymbol{k}, \boldsymbol{n})$しきい値法

n人の参加者のうち，任意のk個のシェアから秘密情報sを復元するとき，素数pに対して，高々$k-1$次多項式をランダムに選ぶ．

$$f(x) = a_{k-1}x^{k-1} + \cdots + a_1 x + a_0 \bmod p$$

ここで，$a_{k-1}, \dots, a_0$はいずれも$\mathrm{GF}(p)$上の要素であり，特に定数項$a_0 = s$とする．この多項式の係数$a_{k-1}, \dots, a_0$はディーラしか知らない．pはすべての参加者で共有される．

(1) 分散アルゴリズム

ディーラはn人の参加者$P_1, \dots, P_n$に$x_i \neq 0$とする座標$(x_i, f(x_i))$に対応したシェア$f(x_i)$を秘密裏に送る．

(2) 復元アルゴリズム

復元に協力する任意のk人の参加者を$P_{i_1}, \dots, P_{i_k}$とする．連立方程式を解く代わりに，多項式補間の 1 つであるラグランジュ補間を用いて，秘密情報を求めることができる．

$$s = f(0) = \sum_{j=1}^{k} f(x_{i_j}) \prod_{t=1, t\neq j}^{k} \frac{-x_{i_t}}{x_{i_j} - x_{i_t}}$$

3.2 藤井らの$(\boldsymbol{2}, \boldsymbol{n})$しきい値法

n人の参加者を$P_0, \dots, P_{n-1}$とする．n_pを$n_p \geq n$を満たす素数とする．nが合成数であれば，$(2, n_p)$しきい値法を用いてn個のシェアを生成す

ると読み替える．**図 2** のように，秘密情報$s \in \{0,1\}^{d(n_p-1)}$をn_p-1個のdビットに等分割し，$s_0=\{0\}^d$とする．

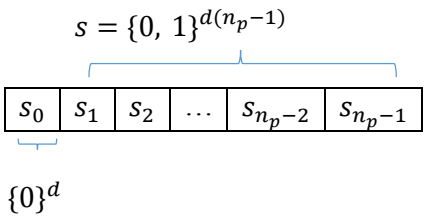

図 2 秘密情報sと添え字の関係

(1) 分散アルゴリズム

参加者P_iに(1)の処理でシェアw_iを秘密裏に送る．n_p個のr_jは互いに独立，かつランダムに選ばれる．r_{n_p-1}は使用せず，添え字は法n_pの演算を行う．**図 3** に(2, 5)しきい値法を用いた例を示す．

$$w_i = w_{(i,0)} \parallel w_{(i,1)} \parallel \cdots \parallel w_{(i,n_p-2)}$$
$$w_{(i,j)} = s_{i-j} \oplus r_j \qquad (1)$$
$$(\,0 \le i < n, 0 \le j < n_p - 1\,)$$

シェア	$j=0$	$j=1$	$j=2$	$j=3$
w_0	r_0	$s_4 \oplus r_1$	$s_3 \oplus r_2$	$s_2 \oplus r_3$
w_1	$s_1 \oplus r_0$	r_1	$s_4 \oplus r_2$	$s_3 \oplus r_3$
w_2	$s_2 \oplus r_0$	$s_1 \oplus r_1$	r_2	$s_4 \oplus r_3$
w_3	$s_3 \oplus r_0$	$s_2 \oplus r_1$	$s_1 \oplus r_2$	r_3
w_4	$s_4 \oplus r_0$	$s_3 \oplus r_1$	$s_2 \oplus r_2$	$s_1 \oplus r_3$

図 3 (2, 5)しきい値法における分散

(2) 復元アルゴリズム

図 3 においてP_1とP_3が復元に協力するとする．r_1を起点に復元すると，秘密情報の断片s_2を得る．r_3を起点に復元すると，s_3, r_0, s_1, r_2, s_4を順番に得る．結果，必要な秘密情報の断片から，秘密情報$s = s_1 \parallel s_2 \parallel s_3 \parallel s_4$を得る．

シェア	$j=0$	$j=1$	$j=2$	$j=3$
w_1	$s_1 \oplus r_0$	r_1	$s_4 \oplus r_2$	$s_3 \oplus r_3$
w_3	$s_3 \oplus r_0$	$s_2 \oplus r_1$	$s_1 \oplus r_2$	r_3

図 4 (2, 5)しきい値法における復元

4 階層的秘密分散法と関連研究

$\boldsymbol{U}$を次で構成されるn人の参加者集合とする．

$$\boldsymbol{U} = \bigcup_{i=0}^{m} \boldsymbol{U}_i,\ \boldsymbol{U}_i \cap \boldsymbol{U}_j = \emptyset\ (0 \le i < j \le m)$$

$\boldsymbol{k} = \{k_i\}_{i=0}^{m}$ $(0 < k_0 < \cdots < k_m)$ を単調増加の数列とする．$(\boldsymbol{k}, n)$階層的しきい値法では，各参加者$u \in \boldsymbol{U}$に次のアクセス構造を満たすようにシェアを割り当てる．

$$\Gamma = \{\boldsymbol{V} \subset \boldsymbol{U} : \left|\boldsymbol{V} \cap \left(\bigcup_{j=0}^{i} \boldsymbol{U}_j\right)\right| \ge k_i,$$
$$\forall i \in \{0,1,\ldots,m\}\}$$

4.1 関連研究

Tassa は文献[3][4]で，Shamir の(k, n)しきい値法のように，大きな有限体上の$k-1$次多項式$p(x)$の定数を秘密情報とし，最大しきい値を$k = k_m$とする．各参加者$u \in \boldsymbol{U}$は自身の階層の位置に依存する何らかのj階導関数値$p^{(j)}(u)$をシェアとして受け取る．より重要な参加者はより小さいi番目の参加者$\boldsymbol{U}_i$に所属し，より低いj階導関数を用いたシェアを得る．導関数を適切に選ぶことで，階層的秘密分散法の要求するアクセス構造を満たし，権限を持つ部分集合が協力して秘密の復元を試みる．

Shamir の文献[2]では，階層的秘密分散の実現方法として，より重要な参加者にはより多くのシェアを与えることで達成することを提案した．しかし，Tassa が文献[3][4]で指摘するように，Shamir の手法は，参加者の部分集合の中で表現されるそれぞれのレベルで関係づけられたしきい値の加重平均で決まるため，低いレベルの参加者の部分集合が十分に大きいときは，低いレベルの参加者のみで秘密の復元ができてしまう課題がある．

4.2 エルミート補間

秘密情報の復元で多項式補間を利用すると，計算量削減により高速化に貢献する．しかし，シェアに導関数値が含まれると，ラグランジュ補間では秘密情報を復元できない．そこで，導関数値を含めた補間手法としてエルミート補間が知られている．差分商で表現されるニュートンの補間公式において，差分商に極限を適用したと考えることができる．しかし，$f'(x_1)$と同時に$f(x_1)$も与えられる必要があるため，シェア配布の観点で制限が入る．バーコフ補間はこの制限を解消しうる．

4.3 バーコフ補間[15][16]

$\mathcal{G} = \{g_0, g_1, \dots, g_N\}$を線形独立な[a, b]で$n$回連続微分可能なℝ上の関数系とし，線形結合$P = \sum_{k=0}^{N} a_k g_k \ \ (a_k \in \mathbb{R})$を$\mathcal{G}$の多項式と呼ぶ．

$m \times (n+1)$補間行列

$$E = \left[e_{i,j}\right]_{i=1,j=0}^{m \quad n} \quad (m \geq 1,\ n \geq 0)$$

は要素$e_{i,j}$が 0 または 1 であり，かつ，$\sum e_{i,j} = N+1$である．ただし，Eは空行を含まない．すなわち，$e_{i,j} = 0 \ \ (\forall j = 0, \dots n)$となる行$i$は含まないとする．

今，[a, b]のm個の異なる点の集合$X = \{x_1, \dots, x_m\} \ \ (x_1 < \cdots < x_m)$が与えられているとする．バーコフ補間問題とは，$<E, X, \mathcal{G}>$の組と与えられたデータ$c_{i,j}$により，次の条件を満たす多項式を見つけることである．

$$p^{(j)}(x_i) = c_{i,j}, \quad e_{i,j} = 1 \tag{2}$$

式(2)は$N+1$個の等式からなる．次の行列式が 0 以外のときに限り，$<E, X, \mathcal{G}>$の組が$c_{i,j}$の各集合に対して唯一の解をもつ．

$$D(E, X, \mathcal{G}) = \det[g_0^{(j)}(x_i), \dots, g_N^{(j)}(x_i); e_{i,j} = 1] \tag{3}$$

式(3)は$e_{i,j} = 1$の(i, j)の組に対応した 1 つの行だけを示した行列式である．また，行の並びは$i < i'$または$i = i', j < j'$のときに(i, j)が(i', j')より先に並ぶ辞書的な順番とする．

$(N+1) \times (N+1)$行列を$A(E, X, \mathcal{G})$と表現すると，$D(E, X, \mathcal{G})$は次式で表せる．

$$D(E, X, \mathcal{G}) = \det\big(A(E, X, \mathcal{G})\big) = |A(E, X, \mathcal{G})|$$

データ$c_{i,j}$が$c_{i,j} = p^{(j)}(x_i)$で与えられたとき，補間多項式は次式で与えられる．

$$p(x) = \sum_{j=0}^{N} \frac{D\big(E, X, \mathcal{G}_j\big)}{D(E, X, \mathcal{G})} \cdot g_j(x)$$

$\mathcal{G}_j$は$\mathcal{G}$のg_jをpに置き換えた関数の集合で，たとえば，$\mathcal{G}_1 = \{g_0, p, g_2, \dots, g_N\}$である．

4.4 バーコフ補間の具体例

$g_0(x) = 1, g_1(x) = x, g_2(x) = x^2$において，すなわち，$\mathcal{G} = \{1, x, x^2\}$において，次のように$X$と$E$が与えられたとする．

$$X = \{1, 2, 3\}, \qquad E = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

すなわち，次の値を満たす多項式$p(x) = \sum_{j=0}^{2} a_j x^j$を探すことである．

$$p(1) = c_{1,0}, p(2) = c_{2,0}, p'(3) = c_{3,1}$$

具体的に$p(1) = 15, p(2) = 29, p'(3) = 23$が与えられたとすると，多項式(4)を得る．

$$D(E, X, \mathcal{G}) = \begin{vmatrix} g_0(x_1) & g_1(x_1) & g_2(x_1) \\ g_0(x_2) & g_1(x_2) & g_2(x_2) \\ g_0'(x_3) & g_1'(x_3) & g_2'(x_3) \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 0 & 1 & 6 \end{vmatrix} = 3,$$

$$D(E,X,\mathcal{G}_0)=\begin{vmatrix} p(x_1) & g_1(x_1) & g_2(x_1) \\ p(x_2) & g_1(x_2) & g_2(x_2) \\ p'(x_3) & g_1'(x_3) & g_2'(x_3) \end{vmatrix}=\begin{vmatrix} 15 & 1 & 1 \\ 29 & 2 & 4 \\ 23 & 1 & 6 \end{vmatrix} = 21,$$

$$D(E,X,\mathcal{G}_1)=\begin{vmatrix} 1 & 15 & 1 \\ 1 & 29 & 4 \\ 0 & 23 & 6 \end{vmatrix}=15,$$

$$D(E,X,\mathcal{G}_2)=\begin{vmatrix} 1 & 1 & 15 \\ 1 & 2 & 29 \\ 0 & 1 & 23 \end{vmatrix}=9,$$

$$p(x)=\sum_{j=0}^{2}\frac{D\left(E,X,\mathcal{G}_j\right)}{D(E,X,\mathcal{G})}\cdot g_j(x)=7+5x+3x^2 \tag{4}$$

ここで, 階層的秘密分散を考える. 十分大きい素数pにおいて, $f(x)=3x^2+5x+7 \bmod p$でシェアが分散されたとき, $f(1),f(2),f'(3)$のシェアが集まると, 秘密情報が得られる.

$$s=f(0)=\frac{D(E,X,\mathcal{G}_0)}{D(E,X,\mathcal{G})}=\frac{21}{3}=7$$

5 提案方式

5.1 モデル

権限レベルの高い参加者 2 人を実現するために, n人の参加者集合$\boldsymbol{U}$を必須参加者のレベルとそれ以外のレベルに分け, $\boldsymbol{k}=\{k_i\}_{i=0}^{1}$の$(\boldsymbol{k},n)$階層的しきい値法をモデルにする.

$$\boldsymbol{U}=\bigcup_{i=0}^{1}\boldsymbol{U}_i,\boldsymbol{U}_0\cap\boldsymbol{U}_1=\emptyset,$$

$$|\boldsymbol{U}_0|=2,|\boldsymbol{U}_1|=n-2,(n\geq 3)$$

$$\Gamma=\{\boldsymbol{V}\subset\boldsymbol{U}:\left|\boldsymbol{V}\cap\left(\bigcup_{j=0}^{1}\boldsymbol{U}_j\right)\right|\geq k_i,\ \forall i\in\{0,1\}\} \tag{5}$$

$|\boldsymbol{U}_0|=1,|\boldsymbol{U}_1|=n-1,(n\geq 3)$も(5)のモデルに含まれる. 秘密情報は$s$とする.

5.2 排他的論理和を用いた$(\{\boldsymbol{1},\boldsymbol{3}\},\boldsymbol{n})$階層的しきい値法

藤井らの$(2,n)$しきい値法を拡張する. たとえば, $n=5$において, **図 3** の秘密情報の断片を乱数$R_0,\dots,R_4$に置き換え, **図 5** のように$w_0,\dots,w_4$を生成し, シェア$W_0,\dots,W_4$を式(6)で配布する.

w_0	r_0	$R_4\oplus r_1$	$R_3\oplus r_2$	$R_2\oplus r_3$
w_1	$R_1\oplus r_0$	r_1	$R_4\oplus r_2$	$R_3\oplus r_3$
w_2	$R_2\oplus r_0$	$R_1\oplus r_1$	r_2	$R_4\oplus r_3$
w_3	$R_3\oplus r_0$	$R_2\oplus r_1$	$R_1\oplus r_2$	r_3
w_4	$R_4\oplus r_0$	$R_3\oplus r_1$	$R_2\oplus r_2$	$R_1\oplus r_3$

図 5 $w_0,\dots,w_4$の生成

$$W_0=w_0\oplus s,W_1=w_1\oplus s$$
$$W_2=w_2,W_3=w_3,W_4=w_4 \tag{6}$$

W_0,W_2,W_3のシェアが集まれば, W_2,W_3から$r_0,\dots,r_3,R_1,\dots,R_4$を得るので, w_0を計算し, $w_0\oplus W_0$で秘密情報sを得る.

W_0,W_1,W_3のシェアが集まれば, W_0,W_1から$w_0\oplus w_1$を得て, $R_1,\dots,R_4$を得る. その後, $W_3(=w_3)$から$r_0,\dots,r_3$を得るので, w_0を計算し, $w_0\oplus W_0$で秘密情報sを得る（**図 6**）.

w_3	$R_3\oplus r_0$	$R_2\oplus r_1$	$R_1\oplus r_2$	r_3
$w_0\oplus w_1$	R_1	R_4	$R_3\oplus R_4$	$R_2\oplus R_3$

図 6 W_0,W_1,W_3のシェアで復元

5.3 導関数を用いた$(\{\boldsymbol{1},\boldsymbol{3}\},\boldsymbol{n})$階層的しきい値法

$l\geq|s|$とする拡大体$\mathrm{GF}(2^l)$上の$k-1$次多項式をランダムに選ぶ. 多項式の係数はいずれも$\mathrm{GF}(2^l)$上の要素で, 定数項を秘密情報とする.

$k=k_1=3$のため, 2 次多項式を選ぶのは自然だが, 拡大体での演算はその導関数値を固定にしてしまう. たとえば, $\mathrm{GF}(2^4)$上の多項式$f(x)=3x^2+5x+7$を選べば, ディーラの選択する値 5 そのものが常に配布されてしまう.

$$f'(x)=\lim_{h\to 0}\frac{3(x+h)^2+5(x+h)+7-\left(3x^2+5x+7\right)}{h}$$
$$=\lim_{h\to 0}\frac{3\left(x^2+h^2\right)+5(x+h)+7-(3x^2+5x+7)}{h}$$
$$=\lim_{h\to 0}\frac{3h^2+5h}{h}=\lim_{h\to 0}(3h+5)=5$$

このように，標数 2 の拡大体上で微分すると次数が偶数の項の結果は消えてしまうので，単に拡大体を適用するだけではなく工夫が必要である．

そこで，$k-1$が奇数ならば，$k-1$次多項式をランダムに選ぶこともできるが，$k-1$が偶数のときは，k次多項式をランダムに選択し，n人の参加者に各シェアが配布されると同時に，$u \in \boldsymbol{U}_0$の参加者に相当する 1 つのシェアはグローバルに共有されるようにした．すなわち，(5)のモデルを(7)のモデルに拡張し，$(\{1,3\},n)$階層的しきい値法においては，3 次多項式をランダムに選ぶ．

$$\begin{gathered}\boldsymbol{U} = \boldsymbol{U}_G \times \bigcup_{i=0}^{1} \boldsymbol{U}_i, \boldsymbol{U}_G \cap \boldsymbol{U}_0 \cap \boldsymbol{U}_1 = \emptyset, \\ |\boldsymbol{U}_G| = 1, |\boldsymbol{U}_0| = 2, |\boldsymbol{U}_1| = n-2, (n \geq 3), \\ \Gamma = \{\boldsymbol{V} \subset \boldsymbol{U} : \left| \boldsymbol{V} \cap \left(\boldsymbol{U}_G \times \bigcup_{j=0}^{1} \boldsymbol{U}_j \right) \right| \geq k_i + 1, \\ \forall i \in \{0,1\}\}\end{gathered} \tag{7}$$

(1) 分散アルゴリズム

n 人の参加者 $P_{i,j} \in \boldsymbol{U}_j$ $(1 \leq i \leq |\boldsymbol{U}_j|, j \in \{0,1\})$に$x_{i,j} \neq 0$とするシェア$f^{(j)}(x_{i,j})$を秘密裏に送る．

(2) 復元アルゴリズム

復元に協力する 3 人は，A) $\boldsymbol{U}_0$から 1 人，$\boldsymbol{U}_1$から 2 人，または B) $\boldsymbol{U}_0$から 2 人，$\boldsymbol{U}_1$から1人，のときアクセス構造Γを満たす．

$\mathcal{G} = \{1, x, x^2, x^3\}$において，A) B)のそれぞれに対応する補間行列が次式で与えられる．

$$E_A = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}, \qquad E_B = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

例として，4 ビットで表現される秘密情報 $s = 7$を考える．GF(2^4)上の多項式 $f(x) = 8x^3 + 3x^2 + 5x + 7$ を選ぶ．GF(2^4)上の演算では，$\mathrm{y} = x^3$の導関数は$\mathrm{y}' = x^2$である．よって，次式が得られる．

$$f'(x) = 8x^2 + 5$$

参加者$P_{i,j}$の補間点$x_{i,j}$を**図 7** のように割り当てる．$* \in \boldsymbol{U}_G$はグローバルに共有される$\boldsymbol{U}_0$の参加者に相当するシェアである．

$\boldsymbol{U}_0$	$x_{i,j}$	シェア
$P_{1,0}$	1	$f(1) = 9$
$P_{2,0}$	2	$f(2) = 13$
*	3	$f(3) = 6$

$\boldsymbol{U}_1$	$x_{i,j}$	シェア
$P_{1,1}$	4	$f'(4) = 14$
$P_{2,1}$	5	$f'(5) = 6$
…	…	…
$P_{n-2,1}$	$n+1$	$f'(n+1)$

図 7 シェア配布

たとえば，$P_{2,0}$のシェアは次の計算式で求まる．

$$\begin{aligned} f(2) &= 8 \cdot 2^3 + 3 \cdot 2^2 + 5 \cdot 2 + 7 \\ &= \alpha^3(\alpha^1)^3 + \alpha^4(\alpha^1)^2 + \alpha^8\alpha^1 + 7 \\ &= \alpha^6 + \alpha^6 + \alpha^9 + 7 = \alpha^9 + 7 \\ &= (1010) + (0111) = 13 \end{aligned}$$

$X = \{1,3,4,5\}$と補間行列E_Aが与えられ，$f(1), f'(4), f'(5)$のシェアが集まると，グローバルのシェア$f(3)$が加わるので，次の計算式により秘密情報を復元できる．

$$\begin{aligned} D(E_A, X, \mathcal{G}) &= \begin{vmatrix} g_0(x_1) & g_1(x_1) & g_2(x_1) & g_3(x_1) \\ g_0(x_2) & g_1(x_2) & g_2(x_2) & g_3(x_2) \\ g_0'(x_3) & g_1'(x_3) & g_2'(x_3) & g_3'(x_3) \\ g_0'(x_4) & g_1'(x_4) & g_2'(x_4) & g_3'(x_4) \end{vmatrix} \\ &= \begin{vmatrix} 1 & 1 & 1 & 1 \\ 1 & 3 & 5 & 15 \\ 0 & 1 & 0 & 3 \\ 0 & 1 & 0 & 2 \end{vmatrix} = 4, \end{aligned}$$

$$\begin{aligned} D(E_A, X, \mathcal{G}_0) &= \begin{vmatrix} f(x_1) & g_1(x_1) & g_2(x_1) & g_3(x_1) \\ f(x_2) & g_1(x_2) & g_2(x_2) & g_3(x_2) \\ f'(x_3) & g_1'(x_3) & g_2'(x_3) & g_3'(x_3) \\ f'(x_4) & g_1'(x_4) & g_2'(x_4) & g_3'(x_4) \end{vmatrix} \\ &= \begin{vmatrix} 9 & 1 & 1 & 1 \\ 6 & 3 & 5 & 15 \\ 14 & 1 & 0 & 3 \\ 6 & 1 & 0 & 2 \end{vmatrix} = 15, \end{aligned}$$

$$f(0) = \frac{D(E_A, X, \mathcal{G}_0)}{D(E_A, X, \mathcal{G})} = \frac{15}{4} = \frac{\alpha^{12}}{\alpha^2} = \alpha^{12}\alpha^{13} = \alpha^{10} = 7$$

同様に，$X = \{1,2,3,4\}$と補間行列E_Bが与えられ，$f(1), f(2), f'(4)$のシェアが集まると，グローバルのシェア$f(3)$が加わるので，秘密情

報を復元できる．なお，$y = x^2$の導関数が$y' = 0$だから，$g_2'(x_3)$は 0 である．

5.4 導関数を用いた$(\{1,3\},n)$階層的しきい値法の計算量

秘密情報の復元に必要な計算量について，$D(E,X,\mathcal{G}_0)$は4×4行列式であるから，**図 8** から乗算 40 回，加算 23 回である．

行列	行列式	乗算	加算
4×4	3×3: 4 回	4 回	3 回
3×3	2×2: 3 回	3 回	2 回
2×2		2 回	1 回

図 8 $D(E,X,\mathcal{G}_0)$の計算量

次に，$D(E,X,\mathcal{G})$はシェアが集まる前にあらかじめ計算することもできるが，$D(E,X,\mathcal{G}_0)$と同時に計算したとしても，$D(E,X,\mathcal{G}_0)$と第 1 列が異なるだけであり，$D(E,X,\mathcal{G}_0)$で既に計算された3×3行列式の結果を流用できる．また，$D(E,X,\mathcal{G})$は第 1 列が 0 または 1 であり，補間行列E_Bから導かれる高々2 回の加算で求まる．

最後に，$f(0)$を求めるための除算を含め，秘密情報の復元に必要に必要な計算量は，高々乗算 40 回，加算 25 回，除算 1 回である．

6 おわりに

階層的秘密分散法に着目した．先行研究の考察から，排他的論理和の手法への適用，導関数の拡大体への適用のそれぞれについて提案した．今後は，導関数を用いた階層的秘密分散法の$\mathrm{GF}(2^{64})$への適用を中心に，拡大体の演算で使用するメモリ使用量の視点も含め，具体的に性能評価を行う．

謝辞

本研究の一部は JSPS 科研費 25330161 の助成を受けたものである．

参考文献

[1] Blakley, G. R: Safeguarding cryptographic keys, Proceedings of the National Computer Conference 48, pp.313–317, 1979.

[2] Shamir, Adi: How to share a secret, Communications of the ACM 22 (11), pp.612–613, 1979.

[3] Tamir Tassa: Hierarchical Threshold Secret Sharing, TCC 2004, LNCS 2951, pp. 473–490, 2004.

[4] Tamir Tassa: Hierarchical Threshold Secret Sharing, Journal of Cryptology 20 (2), pp. 237–264, 2007.

[5] 藤井吉弘, 多田美奈子, 保坂範和, 栃窪孝也, 加藤岳久: 高速な(2, n) 閾値法の構成法とシステムへの応用, CSS, 8C-2, 2005.

[6] Jun KURIHARA, Shinsaku KIYOMOTO, Kazuhide FUKUSHIMA, and Toshiaki TANAKA, Members: A Fast (3,n)-Threshold Secret Sharing Scheme Using Exclusive-OR Operations, IEICE TRANS. FUNDAMENTALS, VOL.E91-A, NO.1, 2008.

[7] Jun KURIHARA, Shinsaku KIYOMOTO, Kazuhide FUKUSHIMA, and Toshiaki TANAKA, Members: On a Fast (k,n)-Threshold Secret Sharing Scheme, IEICE TRANS. FUNDAMENTALS, VOL.E91-A, NO.9, 2008.

[8] 千田浩司, 五十嵐大, 濱田浩気, 高橋克巳: エラー検出可能な軽量 3 パーティ秘匿関数計算の提案と実装評価, 情報処理学会論文誌, Vol.52, No.9, pp.2674-2685, 2011.

[9] Amir Herzberg, Stanislaw Jarecki, Hugo Krawczyk, Moti Yung: Proactive Secret Sharing Or:How to Cope With Perpetual Leakage, CRYPTO' 95, pp. 339-352, 1995.

[10] 古田英之, 須賀祐治, 岩村惠市: 秘密分散システムにおける分散データの更新手法, 情報処理学会, 第 65 回 CSEC 研究会, 2014.

[11] 神宮武志, 古田英之, 岩村惠市: 秘密分散法における検証可能な分散情報の更新手法, 情報処理学会, 第 67 回 CSEC 研究会, 2014.

[12] 五十嵐大, 露崎浩太, 川原祐人: SHSS: オブジェクトストレージ向けの超高速秘密分散ライブラリ, 情報処理学会, 第 70 回 CSEC 研究会, 2015.

[13] Toshiya Itoh, Shigeo Tsujii: A fast algorithm for computing multiplicative inverses in GF(2^m) using normal bases, Information and Computation, Vol.78 Issue 3, pp.171-177, 1988.

[14] D. R. Stinson, Cryptography: Theory and Practice, CRC Press, 1995.

[15] G. G. Lorentz, K. Jetter, S. D. Riemenschneider: Birkhoff Interpolation, Encyclopedia of Mathematics and its Applications 19, 1983, 1984.

[16] G. G. Lorentz and K. L. Zeller: Birkhoff Interpolation, SIAM Journal on Numerical Analysis Vol. 8, No. 1, pp. 43-48, Mar., 1971.