L-040

有理関数補間を用いた (k, n) しきい値秘密分散法の不正者検知・特定

Detection and Identification of Cheaters in (k, n) Secret Sharing Scheme Using Rational Interpolation

坂口 昌隆 †
Masataka Sakaguchi

甲斐 博 †
Hiroshi Kai

1. まえがき

秘密分散法は Shamir[1] と Blakley[2] によって独立に提案された. (k, n) しきい値秘密分散法では，秘密情報から n 個の分散情報を生成し，そのうちの k 個以上が集まったときのみ秘密情報を復元できる. しかし，Shamir の多項式補間による (k, n) しきい値秘密分散法では，不正な情報が含まれるとき正しく秘密情報を復元することができないという問題点がある [5]. そのため，不正な情報が含まれるとき，不正情報を検知 (不正者検知) し，それを特定 (不正者特定) できる秘密分散法が必要となる. このような秘密分散法を検証可能秘密分散法という.

検証可能秘密分散法として Lin らによる RSA 仮定を用いた方法が提案されている [6]. しかし，この手法は計算量理論に基づくものであり，秘密分散法の安全性が問題となる. これに対して，計算量の仮定をおくことのない情報理論的な検証可能秘密分散法が存在する. このような秘密分散法として藤原らの手法 [3] がある. 藤原らの手法は秘密情報の復元時間が速いという利点があるが，不正者特定の上限 (特定可能な不正者数の上限値) が低いという問題点がある.

その他の情報理論的な手法として Harn らの手法 [4] が存在するが，この手法は藤原らの手法と比べて不正者特定の上限が高いものの，復元処理に時間がかかるという問題点がある.

そこで本論文では，藤原らの手法に Harn らの手法を適用することで，藤原らの手法における不正者特定の上限の改善を目指す. これによって，高い不正者特定能力を保ちながら，高速な復元を行う秘密分散法が可能となる.

2. 秘密分散法

2.1. 秘密分散法

(k, n) しきい値秘密分散法はディーラと n 人の参加者とからなるモデルであり，以下のような分散段階と復元段階とからなる.

［分散段階］ ディーラは秘密情報から n 個の分散情報を生成し，参加者に送る.

［復元段階］ 集まった参加者がもつ分散情報から秘密情報を得る.

このとき，n 個の分散情報は以下の性質を満たすように生成される.

1. k 個以上の分散情報から秘密情報が得られる.
2. $k-1$ 個以下の分散情報からは秘密情報が得られない.

(k, n) しきい値秘密分散法の代表例として Shamir の (k, n) しきい値秘密分散法がある. この手法は多項式補間による方法であり，2 次元平面上の点から多項式を復元することで秘密情報を得る.

［分散段階］

1. 秘密情報 s に対して素数 $p > \max(s, n)$ を選ぶ. 以降の計算は $GF(p)$ で行う.
2. 2 次平面上の $k-1$ 次の多項式を
$$f(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1} \tag{2.1}$$
とする. ここで, a_0 は秘密情報 s とし, $a_1, a_2, \cdots, a_{k-1}$ は一様分布にしたがう独立な乱数とする. ただし，$a_{k-1} \neq 0$ とする.
3. 分散情報を $D_i = (i, f(i))$ $(i = 1, 2, \cdots, n)$ とする.

［復元段階］

1. D_i を k 個以上集め，$k-1$ 次多項式 $f(x)$ を補間により求める.
2. $s = f(0)$ $(= a_0)$ により秘密情報を得る.

復元段階において，集めた分散情報に対して Lagrange 補間
$$f(x) = \sum_{i=1}^{k} \lambda_i(x) f(x_i) \tag{2.2}$$
によって多項式 $f(x)$ を求めればよい. ここで，$x_i \in \{1, 2, \cdots, n\}$ であり，
$$\lambda_i(x) = \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} \tag{2.3}$$
である.

2.2. 秘密分散法における不正

Shamir の (k, n) しきい値秘密分散法について，復元段階の参加者のうち，分散情報 D_i をもつ参加者が D_i のかわりに改ざんされた情報 D_i' を提出したとする. このとき，補間によってもとの多項式 (2.1) を得ることができないため，正しく秘密情報を復元できない. このような方法に対して，不正を検出することができる秘密分散法を検証可能秘密分散法という.

本論文では，復元段階における不正者は $k-1$ 人以下とし，不正者はそれぞれ自身がもつ分散情報 D_i のかわりにランダムな値 D_i' を提出するものとする. また，秘密分散法における不正としてディーラが不正な分散情報を参加者に送るモデルも考えられるが，本論文ではそのような場合は考えない.

2.3. 藤原らの手法

藤原らは，有理関数補間における到達不能点を用いることで不正者特定可能な秘密分散法を提案している.

有理関数補間とは，与えられた $M+N+1$ 個のデータ点の集合
$$D = \{D_i = (x_i, y_i) \mid i = 1, 2, \cdots, M+N+1\}$$
に対し，有理関数 $R_{M,N}(x)$ を
$$R_{M,N}(x) = \frac{P_M(x)}{Q_N(x)} = \frac{p_0 + p_1 x + \cdots + p_M x^M}{1 + q_1 x + \cdots + q_N x^N} \tag{2.4}$$

† 愛媛大学大学院 理工学研究科 電子情報工学専攻

となるように求める補間である．このとき，(2.4) の係数 p_j, q_j は，連立方程式

$$Q_N(x_i)y_i = P_M(x_i) \tag{2.5}$$

を解くことによって定める．

求まった有理関数 $R_{M,N}$ に対して，データ点集合 D のうち $R_{M,N}$ が通ることのないデータ点 D_i のことを到達不能点という．有理関数 (2.4) が $D_i = (x_i, y_i)$ を通らないとき，$P_M(x)$ と $Q_M(x)$ は $x = x_i$ に共通根をもつことが知られている．そのため，そのような点情報を提出した参加者を不正者として特定することが可能となる．

藤原らの手法の構成を以下に示す．

［分散段階］

1. 秘密情報 s に対して素数 $p > \max(s, n)$ を選ぶ．以降の計算は $GF(p)$ で行う．
2. $k-1$ 次の多項式 (2.1) の係数 $a_0 = s$ とし，$a_1, a_2, \cdots, a_{k-1}$ をランダムに選ぶ．$f(x)$ が $k-1$ 次多項式とならない場合，この手順をくり返す．
3. 分散情報 $D_i = (x_i, f(x_i))$ $(i = 1, 2, \cdots, n)$ としてシェアする．ここで，x_i はランダムに選ぶ．

［復元段階］

1. 分散情報を l $(> k+1)$ 個集める．
2. 以下を満たすように M, N を決定する．
$$l = M + N + 1 \quad (M - N \geqq k - 1) \tag{2.6}$$
3. $i = 1, 2, \cdots, l$ に対し，
$$\begin{aligned} &a_0 + a_1 x_i + \cdots + a_M {x_i}^M \\ &\qquad = f(x_i)(1 + b_1 x_i + \cdots + b_N {x_i}^N) \end{aligned} \tag{2.7}$$
の連立方程式を解き，有理関数 $R_{M,N}$ を求める．
4. 結果を各参加者に返す．秘密情報の復元は，$s = R_{M,N}(0)$ とすればよい．

藤原らの手法では，不正者の数 c が $c \leqq N$ の場合に有理関数 (2.4) が $k-1$ 次の多項式となれば不正者の特定が可能であることが示されている．これより，復元段階の条件式 (2.6) から，不正者特定の上限は $(l-k)/2$ であることがわかる．

2.4.Harn らの手法

Harn らの手法は，分散情報の一貫性と秘密情報の大多数とからなる秘密分散法である．以下にそれぞれの定義を示す．

定義 2.1 (分散情報の一貫性)

$u = {}_l\mathrm{C}_k$ とおく $(l \geqq k)$．また，l 個の分散情報 $D_1, D_2, \cdots, D_l$ から k 個を選ぶ各組み合わせの集合を $\mathcal{T} = \{T_1, T_2, \cdots, T_u\}$ とする．

各 T_i から再構成される秘密情報 s^i が $s^1 = s^2 = \cdots = s^u$ となる場合，$D_1, D_2, \cdots, D_l$ は一貫性があるという．

定義 2.2 (秘密情報の大多数)

集めた分散情報 $D_1, D_2, \cdots, D_l$ に一貫性がない場合，再構成した秘密情報の集合 $U = \{s^1, s^2, \cdots, s^u\}$ を

$$U = U_1 \cup U_2 \cup \cdots \cup U_v \quad (U_i \cap U_j = \phi) \tag{2.8}$$

となるように分割する．ここで，$u = {}_l\mathrm{C}_k$ であり，$U_i = \{s^{i_1}, s^{i_2}, \cdots, s^{i_j}\}$ において

$$s^{w_i} := s^{i_1} = s^{i_2} = \cdots = s^{i_j} \tag{2.9}$$

である．

いま，$z = \mathrm{argmax}\{|U_i|\}$ としたとき，s^{w_z} を秘密情報の大多数という．

Harn らの手法の構成を以下に示す．ここで，$u = {}_l\mathrm{C}_k$ とおく．

［分散段階］

1. 秘密情報 s に対して素数 $p > \max(s, n)$ を選ぶ．以降の計算は $GF(p)$ で行う．
2. $k-1$ 次の多項式 (2.1) の係数 $a_0 = s$ とし，$a_1, a_2, \cdots, a_{k-1}$ をランダムに選ぶ．$f(x)$ が $k-1$ 次多項式とならない場合，この手順をくり返す．
3. 分散情報 $D_i = (x_i, f(x_i))$ $(i = 1, 2, \cdots, n)$ としてシェアする．

［復元段階］

1. 分散情報を l $(> k)$ 個集め，集合 L とする．
2. 集めた分散情報を用いて Lagrange 補間により補間多項式 $f(x)$ を計算する．このとき，$f(x)$ が $k-1$ 次の多項式となる場合，$f(x)$ の定数項から秘密情報を得る．そうでない場合，以下に進む．
3. 参加者の集合 L から k 人を選ぶすべての組み合わせの集合を $\mathcal{T} = \{T_1, T_2, \cdots, T_u\}$ とおく．$T_i \in \mathcal{T}$ に対し，秘密情報 s^i を再構成する．
4. 秘密情報の集合 $U = \{s^1, s^2, \cdots, s^u\}$ を (2.8) を満たすようにわける．このとき，各 U_i に対して，s^{w_i} を (2.9) により定める．
5. $z = \mathrm{argmax}\{|U_i|\}$ として，秘密情報 s を $s = s^{w_z}$ とする．
6. s を再構成するような $T_i \in \mathcal{T}$ を選び，$R = L \setminus T_i$ とする．
7. T_i から $k-1$ 個の分散情報を選び，これを T_i' とおく．各 $D_r \in R$ に対して，$T_i' \cup \{D_r\}$ に関する Lagrange の補間多項式を求める．これにより得られた秘密情報 s^r が $s^r = s$ となるならば，D_r は正直な参加者となり，集合 H にいれる．一方，$s^r \neq s$ となるならば，D_r は不正な参加者となり，集合 C にいれる．
8. 最終的に秘密情報 s と不正者の集合 C を得る．

Harn らの手法では，複数回の Lagrange 補間を行うことで最大 $l-(k+1)$ 人の不正者を特定することが可能である．しかし，このときに組み合わせの計算を行うため，藤原らの手法と比べて計算時間がかかるという問題点がある．

3. 提案手法

3.1. 提案手法の構成

提案手法の基本的な構成は藤原らの手法と同様である．まず，分散段階では，ディーラは秘密情報 s から n 個の分散情報を生成し，参加者に送る．復元段階では，復元者は集まった l $(> k+1)$ 個の分散情報 (藤原らの手法と同じ条件) から有理関数補間を行い，秘密情報の復元を試みる．このときに，不正者特定，秘密情報の復元が可能であれば，復元者はその結果を各参加者に返す．もし，不正検知のみが可能な場合，分散情報の集合を小さな部分集合にわけ，その部分集合に対して有理関数補間を行う．この結果，不正者特定が可能であれば，不正者を除いて秘密情報を復元し，そうでなければ不正検知を通知して終了する．

提案手法の構成を以下に示す.

［分散段階］

1. 秘密情報 s に対して素数 $p > \max(s, n)$ を選ぶ. 以降の計算は $GF(p)$ で行う.
2. $k-1$ 次の多項式 (2.1) の係数 $a_0 = s$ とし，$a_1, a_2, \cdots, a_{k-1}$ をランダムに選ぶ. $f(x)$ が $k-1$ 次多項式とならない場合，この手順をくり返す.
3. 分散情報 $D_i = (x_i, f(x_i))$ $(i = 1, 2, \cdots, n)$ としてシェアする.

［復元段階］

1. 分散情報を l $(> k+1)$ 個集める. これを集合 L とする.
2. (2.6) を満たすように M, N を定める.
3. $i = 1, 2, \cdots, l$ に対し，(2.7) の連立方程式を解き，有理関数 $R_{M,N}$ を求める.
4. 有理関数 $R_{M,N}$ の次数と k, l の値に応じ，以下のいずれかに分岐する.
   I. $R_{M,N}$ が $k-1$ 次の多項式となった場合，不正者特定および秘密情報の復元が可能となり，手順 7 へ進む.
   II. $R_{M,N}$ が $k-1$ 次の多項式とならなかった場合
      i. $l = k+2$ の場合，不正検知のみ可能となり，手順 7 へ進む.
      ii. そうでない場合，手順 5 へ進む.
5. 有理関数補間のループ
   (a) 集めた l 個の分散情報から $k+2$ 個を選ぶすべての組み合わせの集合 $\mathcal{T} = \{T_1, T_2, \cdots, T_{{}_lC_{k+2}}\}$ を生成する.
   (b) 各 $T_i \in \mathcal{T}$ に対して有理関数補間を行い，有理関数 $R_{k,1}$ を求める. このとき，$R_{k,1}$ が $k-1$ 次の多項式となった場合，秘密情報の復元と不正者の特定が完了する. 特定された不正者を集合 C にいれ，手順 6 へ進む.
   (c) すべての T_i に対して $k-1$ 次多項式が得られなかった場合，不正検知のみ可能となり，手順 7 へ進む.
6. 全不正者特定
   (a) 秘密情報を復元した分散情報を集合 H にいれる. このうち，$k-1$ 個の分散情報を選び，H' とする. この時点で正しい分散情報か不正な分散情報か決定されていない分散情報の集合を $R := L \setminus H \setminus C$ とおく.
   (b) すべての $D_r \in R$ に対して，$D' = \{D_r\} \cup H'$ として Lagrange 補間を行う. このとき，得られた秘密情報 $s^r = s$ となるならば，D_r は正しい分散情報となり，集合 H にいれる. $s^r \neq s$ ならば，D_r は不正な分散情報となり，集合 C にいれる.
7. 復元者は結果を各参加者に返す.

復元段階 4, 5 (b) において，補間多項式 $R(x) := P(x)/Q(x)$ が $k-1$ 次多項式となる場合を考える. $R(x)$ が $P(x)$ と $Q(x)$ の共通因子 $\prod_i (x - x_i)$ をキャンセルすることで $k-1$ 次となる場合，各 i において $x = x_i$ なる分散情報が不正な情報として検出される. $R(x)$ がキャンセリングなしで $k-1$ 次多項式となる場合，使用した分散情報はすべて正しいと判断される. 不正者検出の後，全不正者特定の手順によって，他の不正情報の特定を行う.

3.2. 提案手法の解析

ここでは，提案手法における不正者特定の評価を行う.

定理 3.1

素数 p を十分大きくとる. このとき，提案手法において，正直な参加者が k 人より多いとき，不正者特定が可能である.

[証明]

不正者がいない場合，$R_{M,N}$ が $k-1$ 次多項式となることは明らかであるため，不正者が c 人 $(1 \leqq c \leqq k-1)$ いるとする.

素数 p が十分大きいとき，不正者が提出するランダムな値がもとの $k-1$ 次多項式上の点である確率は低い. そのため，以下では，不正情報はもとの多項式上の点ではないものとする.

(i) $R_{M,N}$ が $k-1$ 次多項式となるとき
有理関数 $R_{M,N}$ の分母の多項式の次数は

$$N = \left\lfloor \frac{l-k}{2} \right\rfloor \tag{3.1}$$

とすればよい. このとき，$l > k+1$ であるから，提案手法では c 人 $(1 \leqq c \leqq N)$ までの不正者を特定できる.
$c > N$ の場合は (ii) となる.

(ii) $R_{M,N}$ が $k-1$ 次多項式とならないとき
$l = k+2$ とする. 仮定より，正直な参加者は k 人より多いため，不正者数 $c = 1$ となる. このとき，(3.1) より，$N = 1$ となるから，1 人の不正者を特定することが可能であり，$k-1$ 次の既約多項式 $R_{M,N}$ が得られる. これは (i) に該当するため，$R_{M,N}$ が $k-1$ 次多項式とならないときは $l > k+2$ に限る. このとき，有理関数補間のループによって不正者特定を行う.
ここで，各 T_i における不正者数を c_i とおく.
(1) $c_i = 0, 1$ のとき
(i) の議論と同様にして，$k-1$ 次の補間多項式を得ることができる. また，手順 6 によって，正しい分散情報を用いて全不正者を特定することが可能である.
(2) $1 < c_i \leqq k-1$ のとき
各 T_i において特定することが可能な不正者は 1 人であるため，この場合に得られる補間多項式は $k-1$ 次多項式となる確率は低い. そのため，他の $T_i \in \mathcal{T}$ に対して有理関数補間を行う.
いま，正直な参加者は k 人より多いと仮定しているので，上の (1) を満たすような $T_i \in \mathcal{T}$ が存在する. これより，$R_{M,N}$ が $k-1$ 次多項式とならない場合でも不正者を特定することが可能である. ■

4. 既存手法と提案手法の比較

これまでに，既存手法として藤原らの手法および Harn らの手法について述べた. 以下で，既存手法と提案手法における不正者特定の上限および復元段階の計算時間の比較を行う.

使用した計算機環境を表 4.1 に示す. なお，$GF(p)$ 上の計算には，Java.math.BigInteger クラスで実装されたメソッドを用いた. また，素数 p は，$\max(s, n)$ に対して，同クラスの nextProbablePrime() メソッドをを 5000 回実行したものを利用した.

以下，分散段階における参加者の数 n を 20，復元に必要な参

表 4.1　計算機環境

OS	Ubuntu 14.04 (64 bit)
CPU	Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz
メモリ	8 GB
コンパイラ	javac 1.7.0_75

加者の数 k を 7 として設定する．

まず，不正者特定の上限を比べる．このとき，藤原らの手法では $(l-k)/2$，Harn らの手法および提案手法では $l-(k+1)$ までの不正者を特定することが可能である．そのため，集めた分散情報数と不正者特定可能な上限は図 4.1 のようになる．

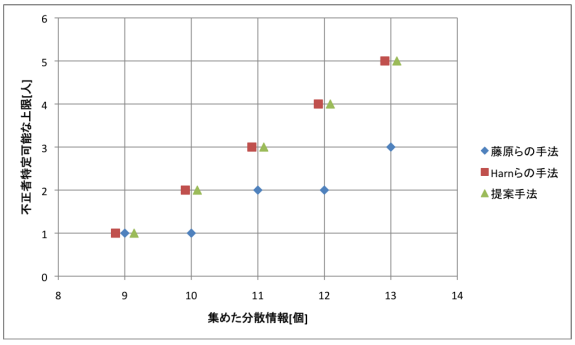

図 4.1　不正者特定可能な上限

次に，各手法における復元段階の計算時間を比較する．復元段階の参加者数 $l=11$ とした場合の復元時間を図 4.2 に示す．ここで，図 4.2 においては，不正者のもつ分散情報をランダムにとり，5 回行った際の平均時間を示している．なお，計算時間の測定には，Java の System クラスの nanoTime() メソッドを用いた．

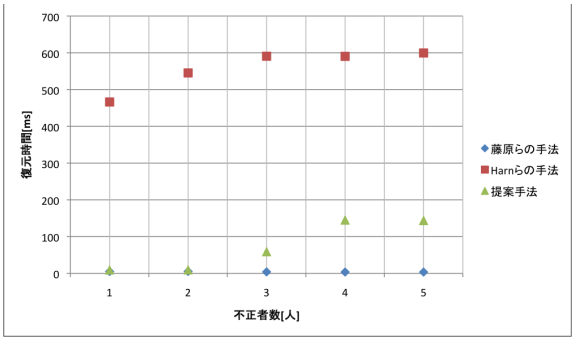

図 4.2　復元時間

図 4.2 において，提案手法では不正者数が 2 以下の場合は 1 回の有理関数補間によって不正者特定を行うため，藤原らの手法と同程度の計算時間で復元が終了する．これに対し，Harn らの手法では ${}_lC_k$ 回の Lagrange 補間が必要となるため，他の手法よりも計算時間がかかる．なお，提案手法において，不正者数が 3 以上の場合は，1 回の有理関数補間では不正者特定ができないため，手順 5 のとおり，分散情報を $k+2$ 個の分散情報の集合にわけ，それに対して有理関数補間を実行する．

一般に，$k-1$ 次の多項式の補間は $O(k^2)$ の計算量が必要である．Harn らの手法は多項式補間の計算を ${}_lC_k$ 回必ず実行する．

一方，提案手法では，$R_{k,1}$ の次数の有理関数補間を計算する．有理関数補間の計算は連立 1 次方程式で行う場合，$O(k^3)$ の計算量が必要である．提案手法では，有理関数補間の計算を最悪で ${}_lC_{k+2}$ 回実行することになるが，計算の途中で有理関数が $k-1$ 次の多項式になると，計算を打ち切ることができる．次数が小さい場合は有理関数補間の計算量があまり影響せず，Harn らの手法より速く計算が終了した．

次数が大きくなると，有理関数補間の計算量が勝り，提案手法が不利になる場合があるかもしれない．計算量が $O(k^2)$ の有理関数近似アルゴリズム [7] もあるが，そのようなアルゴリズムを用いた実験は今後の課題である．

5. 結論

本論文では，既存手法である藤原らの手法と Harn らの手法を組み合わせた手法を提案した．提案手法では，Harn らの手法と同様の不正者特定可能な上限をもちながら，不正者数が $(l-k)/2$ 以下の場合に Harn らの手法と比べて計算時間の改善を行った．不正者数が $(l-k)/2$ より多い場合についても Harn らの手法より効率的な復元が可能である．しかし，l の値が大きいときについては本論文では示していないため，そのような場合についても効率的な復元手順を示すことが必要である．また，上で述べたように，効率的な有理関数近似アルゴリズムを用いた場合における検討も重要である．

参考文献

[1] Adi Shamir : How to Share a Secret, Communication of the ACM, Volume 22, Issue 11, pp.612–613 (1979)

[2] George Robert Blakley : Safeguarding cryptographic keys, Proceedings of the National Computer Conference, Volume 48, pp.313–317 (1979)

[3] 藤原名穂，甲斐博 : 有理関数補間を用いた秘密分散法の一考察，情報処理学会 コンピュータセキュリティシンポジウム 2008 論文集，pp.959–962 (2008)

[4] Lien Harn, Changlu Lin : Detection and identification of cheaters in (t, n) secret sharing scheme, Designs, Codes and Cryptography, Volume 52, pp.15–24 (2009)

[5] Martin Tompa, Heather Woll : How to share a secret with cheaters, Journal of Cryptography, Volume 1, Issue 2, pp.133–138 (1998)

[6] Hung-Yu Lin, Lein Harn : A generalized secret sharing scheme with cheater detection, Proceedings of ASIACRYPT'91, Lecture Notes in Computer Science Volume 739, pp.149–158, Springer-Verlag (1993)

[7] Ömer Eğecioğlu, Çetin K. Koç : A Fast Algorithm for Rational Interpolation Via Orthogonal Polynomials, Mathematics of Computation, Vol. 53, No. 187, pp. 249–264 (1989)