準巡回シンドローム復号問題に基づく線形関数秘匿計算

祁 儀穎[1]　河内 亮周[1]　宮地 充子[1,2,3]

概要：実用的な量子計算機が実現した場合，今までの数論ベースの暗号方式のほとんどが破れることが知られているが，いくつかの符号ベース暗号方式は量子計算機の攻撃に耐えると考えられている．そのような暗号方式の一つとして，Gaborit らは準巡回シンドローム復号問題に基づいた公開鍵暗号方式を NIST のポスト量子暗号標準化コンペティションへ提案している．本研究では彼らの公開鍵暗号方式を二者間秘匿計算に応用し，準巡回シンドローム復号問題に基づいた線形関数の秘匿計算プロトコルを提案する．

キーワード：二者間秘匿計算，符号ベース暗号方式，準巡回シンドローム復号問題

Secure Computation for Linear Functions based on QCSD Problem

Yiying Qi[1]　Akinori Kawachi[1]　Atsuko Miyaji[1,2,3]

Abstract: If practical quantum computers are realized, it is known that most of cryptosystems based on number theory will be broken. However, several coding-based cryptosystems are believed to withstand attacks of quantum computers. As one such cryptosystems, Gaborit et al. propose public-key encryption schemes based on the quasi-cyclic syndrome decoding problem for NIST's Post-Quantum Cryptography Standardization. In this work, we apply their encryption scheme to secure two-party computation for linear functions based on the quasi-cyclic syndrome decoding problem.

Keywords: Secure Two-party Computation, Code-based Cryptography, Quasi-Cyclic Syndrome Decoding (QCSD) Problem

1. はじめに

情報化社会の発展がめざましい現代では，人々が持つ情報は数値化されてインターネットと繋がっている．インターネットで流されるプライバシーデータの秘匿性問題が重要な課題となっている．最近では，「パーソナルデータ」[9] を社会や産業の発展のために二次利用する機運が高まってきている．しかしパーソナルデータの扱いは，情報提供者である個人のプライバシーについて十分配慮する必要がある．情報を安全に解析するため，数値化されたデータは暗号化されてから様々な操作が行われる．例えば，患者の健康状態や持病を調べるため，患者の検査データを医療データ解析機関の健康状態の分類器で分類し，医療データ解析機関の分類器の情報を患者には明らかにせず，また患者は検査データを解析機関に明らかにせずに健康状態の分類結果のみを入手したい．ここで述べた情報の処理は，シンプルに言うと数値化したデータの加算や乗算などの線形演算を含む秘匿計算である．このような秘匿計算を達成するプロトコルが現在多く提案されている．

二者間の線形関数の秘匿計算では，通信機関 A は m を持ち，通信機関 B は線形関数 $f(m) = a \cdot m + b$ の中の a と b を持っている．A は最後に計算結果 $f(m)$ だけを手に入れるような計算プロトコルである．このような研究はすでに多く行われている．Yao の金持ちプロトコル（Millionaires' Problem）[7] が代表的な一例である．さらに彼はこの類の問題をスクランブルされた回路（Garbled Circuit）[8] という 2-入力ゲートのブール回路を利用する方法で解決した．

1 大阪大学 大学院工学研究科
Graduate School of Engineering, Osaka University
2 北陸先端科学技術大学院大学 情報科学研究科
Japan Advanced Institute of Science and Technology
3 科学技術振興機構 CREST
Japan Science and Technology Agency (JST) CREST

Barni ら [2] の線形分岐プログラムは Yao のスクランブルされた回路のアイデアを応用した分類プログラムである．中の線形計算部分は加法準同型暗号と紛失通信を利用している．その他，Bost ら [3] が提案した機械学習分類器で構成した二者間内積秘匿計算は，Paillier 暗号の加法準同型に基づいた計算方式である．Wu ら [6] の決定木とランダムフォレストの分類器で提案した二者間大小比較手法は，入力をバイナリに変換し，ビットごとに大小比較の線形計算を行うテクニックである．Wu らの提案では Paillier 暗号の加法準同型と紛失通信を使用している．以上の線形関数の秘匿計算は主に紛失通信を利用しているが，紛失通信の効率は良いとは言えないのが知られている．Bost らの提案は紛失通信を利用していないが，数論ベースの公開鍵暗号の準同型を使用している．数論ベースの暗号方式は今の段階ではまだ安全と言えるが，量子計算機による攻撃に耐えられない．一方，数論ベースの暗号を利用せず，符号ベースの二者間線形関数秘匿計算も多く提案されている．例えば，[4,5] ではリード・ソロモン符号のノイズ付き符号化ベースの二者間の内積シェアプロトコルを提案している．しかし，これらの研究も紛失通信を利用しなければならない．量子計算機の時代に向けた効率の良い二者間秘匿計算プロトコル構成のためには，数論ベースの公開鍵暗号の準同型や紛失通信を使わないことが望ましい．

また，量子計算機が日々注目を集める現代社会では，既存の数論ベースの公開鍵暗号は攻撃されると考えられる．米国国立標準技術研究所 NIST が量子計算機の攻撃にも耐えうる次世代の暗号方式の標準化のため，ポスト量子暗号標準化コンペティションを開催しており，Gaborit らは準巡回符号ベースの暗号方式 [1] を提案している．数論ベースの暗号方式と違い，Gaborit らが提案したハミング準巡回符号（Hamming Quasi-Cyclic, HQC）暗号方式は符号ベースであり，量子計算機の攻撃に耐えると主張している．また，効率的に暗号化と復号ができることもメリットとして注目されている．そのため，今後暗号分野での発展が期待できる．

Gaborit らの提案では，準巡回符号と効率的な誤り訂正符号の二つの符号を使用している．上で述べたように，準巡回符号は符号ベースであり，多くの攻撃に耐えられる．そのため，他の安全性要件を使用せず，準巡回符号の準巡回シンドローム復号問題に基づくことにより暗号化方式の安全性が保障される．それだけでなく，効率的な誤り訂正符号を使用することにより，暗号化と復号が効率的にでき，復号失敗率が低いメリットがある．また，暗号化方式の復号はこの誤り訂正符号の誤り訂正機能によって成立する．この研究では，HQC 暗号方式のこれらの特性を活かし，加算と乗算を含める線形演算へ拡張する．そして二者間の線形関数の秘匿計算プロトコルを構成する．

2. 準備

2.1 表記

以下，本稿で使用する記号の説明を行う．

- $\mathbb{F}_2$：二元体
- $\mathbb{F}_2^n$：二元体 $\mathbb{F}_2$ 上の n 次元のベクトル空間
- $\mathcal{R}$：係数が $\mathbb{F}$ の X^n-1 を法とする多項式環，つまり $\mathcal{R}=\mathbb{F}[X]/(X^n-1)$
- $\mathcal{C}$：符号語空間
- $\boldsymbol{x}^\top$：ベクトル $\boldsymbol{x}$ の転置
- $\omega(\cdot)$：ベクトルのハミング重み
- δ：誤り訂正符号の最大訂正可能なエラー数

以下，$\mathcal{R}$ の元と n 次元ベクトルを同一視する．つまり，次数が $n-1$ の多項式 $f(x)=a_0+a_1x+\cdots+a_{n-1}x^{n-1}\in\mathcal{R}$ を係数からなるベクトル $\boldsymbol{a}=(a_0,\ldots,a_{n-1})$ と同一視する．

2.2 符号

定義 1 (線形符号)．任意の符号語 $c_1,c_2\in\mathcal{C}$ に対して，$c_1+c_2\in\mathcal{C}$ が常に成り立つような符号 $\mathcal{C}$ を線形符号と呼ぶ．符号長 n および情報ビット数 k の符号 $\mathcal{C}$ を $[n,k]$ 符号と表記する．

定義 2 (生成行列)．行列 $\mathbf{G}\in\mathbb{F}^{k\times n}$ に対して，

$$\mathcal{C}=\{\boldsymbol{m}\cdot\mathbf{G}|\boldsymbol{m}\in\mathbb{F}^k\} \tag{1}$$

を満たすような $\mathbf{G}$ を生成行列と呼ぶ．生成行列は線型符号の基底であり，全ての符号語を生成する．

定義 3 (パリティ検査行列)．行列 $\mathbf{H}\in\mathbb{F}^{(n-k)\times n}$ に対して，

$$\mathcal{C}=\{\boldsymbol{x}\in\mathbb{F}^n|\mathbf{H}\cdot\boldsymbol{x}^\top=\mathbf{0}\} \tag{2}$$

を満たすような $\mathbf{H}$ をパリティ検査行列と呼ぶ．

定義 4 (循環行列)．$\boldsymbol{x}=(x_1,\ldots,x_n)\in\mathbb{F}^n$ とする時，$\boldsymbol{x}$ に関する循環行列は以下のように定義される：

$$\mathbf{rot}(\boldsymbol{x})=\begin{pmatrix}x_1 & x_n & \cdots & x_2\\ x_2 & x_1 & \cdots & x_3\\ \vdots & \vdots & \ddots & \vdots\\ x_n & x_{n-1} & \cdots & x_1\end{pmatrix}\in\mathbb{F}^{n\times n} \tag{3}$$

また，二つの多項式 $\boldsymbol{x},\boldsymbol{y}$ の乗法は以下の性質を持つ：

$$\begin{aligned}\boldsymbol{x}\cdot\boldsymbol{y}&=\boldsymbol{x}\times\mathbf{rot}(\boldsymbol{y})^\top\\&=(\mathbf{rot}(\boldsymbol{x})\times\boldsymbol{y}^\top)^\top\\&=\boldsymbol{y}\times\mathbf{rot}(\boldsymbol{x})^\top\\&=\boldsymbol{y}\cdot\boldsymbol{x}.\end{aligned} \tag{4}$$

定義 5 (巡回シフト)．n 次元ベクトル $(c_0,\ldots,c_{n-1})$ に対し

て，c_i $(i=0,\ldots,n-2)$ を一つ右の位置へ移動（シフト）し，さらに c_{n-1} をベクトルの先頭へ移動する操作を巡回シフトと呼ぶ．つまり，任意の n 次元ベクトル $(c_0,\ldots,c_{n-1})$ に対して，写像 $\sigma:(c_0,c_1,\ldots,c_{n-1})\mapsto(c_{n-1},c_0,\ldots,c_{n-2})$ である．

定義 6 (準巡回符号)．$\boldsymbol{c}=(\boldsymbol{c}_0,\ldots,\boldsymbol{c}_{s-1})\in(\mathbb{F}_2^n)^s$ を符号 $\mathcal{C}$ の任意の符号語とし，σ を巡回シフト演算とする．$(\sigma(\boldsymbol{c}_0),\ldots,\sigma(\boldsymbol{c}_{s-1})\in\mathcal{C}$ ならば，$\mathcal{C}$ を s-準巡回符号と呼ぶ．特に $s=1$ の時，$\mathcal{C}$ を巡回符号と呼ぶ．

定義 7 (系統的準巡回符号)．s-準巡回 $[sn,n]$ 符号が，以下の形のパリティ検査行列を持つなら，系統的準巡回符号と呼ぶ．

$$\boldsymbol{H}=\begin{bmatrix}\boldsymbol{I}_n & 0 & \cdots & 0 & \boldsymbol{A}_1\\ 0 & \boldsymbol{I}_n & & & \boldsymbol{A}_2\\ & & \ddots & & \vdots\\ 0 & & \cdots & \boldsymbol{I}_n & \boldsymbol{A}_{s-1}\end{bmatrix} \tag{5}$$

ここで，$\boldsymbol{A}_1,\cdots,\boldsymbol{A}_{s-1}$ は $n\times n$ の循環行列である．

2.3 安全性仮定

前述のように，公開鍵暗号方式 HQC の安全性は準巡回シンドローム復号問題の計算困難性に基づいている．より具体的には以下の準巡回シンドローム復号決定仮定の下で安全性が証明される．

定義 8 (s-準巡回シンドローム復号決定仮定)．n,w が整数，ブロック数が s の準巡回符号の準巡回シンドローム復号決定問題は，ランダムな系統的準巡回符号のパリティ検査行列 $\mathbf{H}\xleftarrow{\$}\mathbb{F}^{(sn-n)\times sn}$ と行列 $\boldsymbol{y}\xleftarrow{\$}\mathbb{F}^{sn-n}$ が与えられた時に，$(\mathbf{H},\boldsymbol{y}^\top)$ が準巡回シンドローム復号分布で計算した結果か，あるいは分布 $\mathbb{F}^{(sn-n)\times sn}\times\mathbb{F}^{(sn-n)}$ から選んだ一様乱数かを無視できない確率で区別することができない．

後述するが，本稿で提案される秘匿計算プロトコルの安全性は HQC の安全性に帰着されるので，HQC と同様に本稿の秘匿計算プロトコルもこの仮定の下で安全であると証明される．

2.4 2PC の安全性要件

安全な二者間計算（Secure Two-Party Computation, 2PC）は，多者間計算 (Multi-Party Computation, MPC) の部分問題である．多くの暗号プロトコルと深く関わるため，多くの研究者によって研究が進められている．2PC の目的は，二者の入力の値を相手方と共有せず，任意の機能を共同で計算できるように汎用プロトコルを作成することである．2PC の最もよく知られている例の一つは Yao らの金持ち問題 [7]，アリスとボブは自分の所持金を明らかにせずに誰がより金持ちを決定する問題である．具体的に，アリスは a 円を持ち，ボブは b 円を持っている．双方の値 a または b を互いに秘密にしたまま $a\geq b$ かどうかを計算する問題である．一般的に言うと，2PC の安全性要件は二者の入力を相手方に漏らさず，任意の機能の計算をプロトコルで行い，計算結果のみが知られるというものである．

二者間線形関数秘匿計算は 2PC の一種で，2PC の安全性要件を満たす．つまり計算の参加者は自分の入力を相手に知らせず計算を行う．また，プロトコルの機能は線形関数の計算である．具体的に言うと，線形関数秘匿計算は $f(m)=a\cdot m+b$ を計算する．プロトコルの参加者をアリスとボブと呼ぶ．アリスの入力は m，ボブの入力は線形関数のパラメータ a,b．プロトコルを通してアリスは $f(m)=a\cdot m+b$ の結果だけが得られて，ボブは何も手に入れない．

以下で二者間線形関数秘匿計算の安全性要件を定義する．

定義 9 (半誠実敵対者に対する安全性)．

アリス（A）の入力 x とボブ（B）の入力 y を $f_A(x,y)$，$f_B(x,y)$ に写像する機能を $f=(f_A,f_B)$ と表す．A は $f_A(x,y)$ を入手し，B は $f_B(x,y)$ を入手することが目的である．

$f=(f_A,f_B)$ は確率多項式時間の機能，π は機能 f を計算する二者間プロトコルとする．(x,y) の実行 $\pi(x,y)$ とセキュリティパラメータ n での A のビューを $\mathrm{view}_A^\pi(x,y,n)$ として，B のビューを $\mathrm{view}_B^\pi(x,y,n)$ とする．A の出力は $\mathrm{output}_A^\pi(x,y,n)$ として，B の出力は $\mathrm{output}_B^\pi(x,y,n)$ とする．また，二者の共同の出力は $\mathrm{output}^\pi(x,y,n)=\mathrm{output}_A^\pi(x,y,n),\mathrm{output}_B^\pi(x,y,n))$ と表す．

半誠実の敵対者に対して，以下の式を満たす確率多項式時間アルゴリズム S_A と S_B が存在するならば，プロトコル $\pi(x,y)$ は機能 f を安全に計算できると言う．$|x|=|y|=n$，$n\in\mathbb{N}$ を満たす任意の x,y について，以下が成立する：

$$\begin{aligned}&\{(S_A(1^n,x,f_A(x,y)),f(x,y))\}_{x,y,n}\\ \overset{c}{\equiv}&\{(\mathrm{view}_A^\pi(x,y,n),\mathrm{output}^\pi(x,y,n))\}_{x,y,n},\\ &\{(S_B(1^n,x,f_B(x,y)),f(x,y))\}_{x,y,n}\\ \overset{c}{\equiv}&\{(\mathrm{view}_B^\pi(x,y,n),\mathrm{output}^\pi(x,y,n))\}_{x,y,n}.\end{aligned}$$

3. 公開鍵暗号方式 HQC

本稿で提案する二者間線形関数秘匿計算は Gaborit らのハミング準巡回符号（Hamming Quasi-Cyclic, HQC）暗号方式 [1] に基づくプロトコルである．

まず Gaborit らが提案した暗号方式を紹介する．Gaborit らの HQC 暗号方式は準巡回シンドローム復号問題に基づいた公開鍵暗号方式であり，この暗号方式では準巡回符号

と誤り訂正符号 $\mathcal{C}$ の二種類の符号を用いる．誤り訂正符号 $\mathcal{C}$ はメッセージの符号化と復号のため使用し，十分な誤り訂正能力を持った任意の線形符号（例えば BCH 符号）である．準巡回符号はこの公開鍵暗号方式の安全性要件であり，敵対者が復号できないノイズを生成するために使用する．

HQC 暗号方式の参加者はをアリス（A）とボブ（B）とし，B は入力メッセージ $\boldsymbol{m}$ を安全に A に送ることを目標とする．暗号方式は以下の順序で行う：

(1) グローバルパラメータの設定：
パラメータ $\mathrm{param} = (n, k, \delta, w_x, w_r, w_e)$ および符号 $\mathcal{C}$ の生成行列 $\mathbf{G} \in \mathbb{F}^{k \times n}$.

(2) 鍵生成：
A はランダムな $\boldsymbol{h} \xleftarrow{\$} \mathcal{R}$ を生成する．
さらに $(\boldsymbol{x}, \boldsymbol{y}) \xleftarrow{\$} \mathcal{R}^2$ を生成し，$\boldsymbol{x}, \boldsymbol{y}$ のハミング重みは w_x とする．
秘密情報 $\mathrm{sk} = (\boldsymbol{x}, \boldsymbol{y})$,
公開情報 $\mathrm{pk} = (\boldsymbol{h}, \boldsymbol{s} = \boldsymbol{x} + \boldsymbol{h} \cdot \boldsymbol{y})$．A は公開情報 pk を B に送る．

(3) 暗号化：
B はランダムな $\boldsymbol{e} \xleftarrow{\$} \mathcal{R}, (\boldsymbol{r_1}, \boldsymbol{r_2}) \xleftarrow{\$} \mathcal{R}^2$ を生成する．$\boldsymbol{e}$ のハミング重みは w_e とし，$\boldsymbol{r_1}, \boldsymbol{r_2}$ のハミング重みは w_r とする．
そして自分の入力 $\boldsymbol{m}$ を式 $\boldsymbol{u} = \boldsymbol{r_1} + \boldsymbol{h} \cdot \boldsymbol{r_2}$ と $\boldsymbol{v} = \boldsymbol{m} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_2} + \boldsymbol{e}$ で計算する．
B は暗号文 $\boldsymbol{u}, \boldsymbol{v}$ を A に送り返す．

(4) 復号：
A は誤り訂正符号 $\mathcal{C}$ の復号機能 $\mathcal{C}.\mathrm{Decode}(\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{y})$ を使用し，B のメッセージ $\boldsymbol{m}$ を復元する．

HQC 暗号方式では暗号化する時に，公開情報 $\boldsymbol{s}$ を誤り訂正符号で符号化したメッセージ $\boldsymbol{m}$ に加えた．$\boldsymbol{s}$ は準巡回符号で生成したハミング重みが大きいノイズなので，**2.3** 節で紹介した準巡回シンドローム復号決定仮定により安全性が保証される．また，A は復号段階では暗号化したエラー付きの暗号文に秘密鍵を使用して，$\boldsymbol{s}$ の部分の大量のノイズが取り除ける．しかし，$\boldsymbol{x} \cdot \boldsymbol{r_2} - \boldsymbol{r_1} \cdot \boldsymbol{y} + \boldsymbol{e}$ のノイズが残ってしまう．このノイズの重みが誤り訂正符号の最大訂正可能のエラー数 δ より小さければ，正しい復号は可能となる．ハミング重み $w, w_r, w_e = \mathcal{O}(\sqrt{n})$ で想定して解析している．また，符号長 n が大きくなるほど，$\omega(\boldsymbol{x} \cdot \boldsymbol{r_2} + \boldsymbol{e} - \boldsymbol{y} \cdot \boldsymbol{r_1}) \leq \delta$ となる確率は高くなるという結論が Gaborit らの論文で示され，パラメータの実験的評価も行っている．また，2-準巡回シンドローム復号決定仮定および 3-準巡回シンドローム復号決定仮定の下で HQC 暗号方式は IND-CPA 安全である．

4. 提案プロトコル

本稿で提案する二者間線形関数の秘匿計算プロトコルを紹介する．

Gaborit らの HQC 暗号方式に基づき，準巡回符号と誤り訂正符号 $\mathcal{C}$ の二つの符号を使用する．プロトコルの参加者はアリス（A）とボブ（B），A の入力は $m \in \mathbb{F}_2$，B の入力は $a, b \in \mathbb{F}_2$，そして B の出力は無しで，A の出力は $a \cdot m + b$ ことを目標とする．プロトコルは **Protocol 1** で表す．

Protocol 1 2PC プロトコル

入力	A：	$m \in \mathbb{F}_2$
	B：	$a, b \in \mathbb{F}_2$
出力	A：	$a \cdot m + b$
	B：	$\perp$

(1) グローバルパラメータ $\mathrm{param} = (n, k, \delta, w_x, w_r, w_e)$ 及び符号 $\mathcal{C}$ の生成行列 $\mathbf{G} \in \mathbb{F}^{k \times n}$ を設定する．

(2) A はランダムな $\boldsymbol{h} \xleftarrow{\$} \mathcal{R}$ を生成する．さらに $(\boldsymbol{x}, \boldsymbol{y}) \xleftarrow{\$} \mathcal{R}^2$ を生成し，$\boldsymbol{x}, \boldsymbol{y}$ のハミング重みは w とする．秘密情報 $\mathrm{sk} = (\boldsymbol{x}, \boldsymbol{y})$，公開情報 $\mathrm{pk} = (\boldsymbol{h}, \boldsymbol{s} = \boldsymbol{x} + \boldsymbol{h} \cdot \boldsymbol{y})$ とする．

(3) A は入力 m に 0 をパディングし，次元が k の $\boldsymbol{m} = (m, 0, \ldots, 0)$ にする．A はランダムな $\boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v} \xleftarrow{\$} \mathcal{R}$ を生成する．ここで，$\boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}$ のハミング重みは w_r とする．そして $(\boldsymbol{u} = \boldsymbol{h} \cdot \boldsymbol{r_A} + \boldsymbol{r_u}, \boldsymbol{v} = \boldsymbol{m} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_A} + \boldsymbol{r_v})$ を計算する．公開情報 $\boldsymbol{h}, \boldsymbol{s}$ と暗号文ペア $\boldsymbol{u}, \boldsymbol{v}$ を B に送る．

(4) B は $\boldsymbol{b} = (b, 0, \ldots, 0)$ とする．そして $\boldsymbol{r_B} \xleftarrow{\$} \mathcal{R}$ と $(\boldsymbol{e_u}, \boldsymbol{e_v}) \xleftarrow{\$} \mathcal{R}^2$ を生成する．ここで，$\boldsymbol{r_B}$ のハミング重みは w_r とし，$\boldsymbol{e_u}, \boldsymbol{e_v}$ のハミング重みは w_e とする．B は $\boldsymbol{u'} = a \cdot \boldsymbol{u} + \boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u}$, $\boldsymbol{v'} = a \cdot \boldsymbol{v} + \boldsymbol{b} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v}$ を計算する．$\boldsymbol{u'}, \boldsymbol{v'}$ を A に送り返す．

(5) A は誤り訂正符号 $\mathcal{C}$ の復号機能 $\mathcal{C}.\mathrm{Decode}(\boldsymbol{v'} - \boldsymbol{u'} \cdot \boldsymbol{y})$ を使用し，その結果の最初の一ビットを取ることで $a \cdot m + b$ を復元する．

まずはグローバルパラメータを設定する．n は符号 $\mathcal{C}$ の符号長で k は情報ビット数，δ は誤り訂正符号の最大訂正可能なエラー数，w_x, w_r, w_e は各事前に設定するハミング重みであり，例えば Gaborit らが想定している $\mathcal{O}(\sqrt{n})$ の半分の重みとする．公開パラメータの $\mathbf{G}$ は誤り訂正符号 $\mathcal{C}$ の生成行列であり，メッセージと符号語を $\mathbb{F}_2^k \to \mathbb{F}_2^n$ のように写像する．

A はランダムな $\boldsymbol{h} \xleftarrow{\$} \mathcal{R}$ と $(\boldsymbol{x}, \boldsymbol{y}) \xleftarrow{\$} \mathcal{R}^2$ を生成し，

$\boldsymbol{s} = \boldsymbol{x} + \boldsymbol{h} \cdot \boldsymbol{y}$ を計算する．ここで

$$\begin{aligned} \boldsymbol{s} &= \boldsymbol{x} + \boldsymbol{h} \cdot \boldsymbol{y} \\ &= \boldsymbol{x} + \boldsymbol{y} \cdot \mathbf{rot}(\boldsymbol{h})^\top \\ &= (\boldsymbol{x} \quad \boldsymbol{y})(\boldsymbol{I}_n \quad \mathbf{rot}(\boldsymbol{h}))^\top \end{aligned} \tag{6}$$

に変換でき，2-準巡回シンドローム復号仮定に帰着できる．そして A は秘密情報 sk を $(\boldsymbol{x}, \boldsymbol{y})$ とし，公開情報 pk を $(\boldsymbol{h}, \boldsymbol{s})$ とする．

A は入力 m に 0 をパディングし，次元が k の $\boldsymbol{m} = (m, 0, \ldots, 0)$ にする．A は $\boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v} \xleftarrow{\$} \mathcal{R}$ を生成し，$\boldsymbol{m}$ の値を誤り訂正符号により符号化し，再ランダム化する．$(\boldsymbol{u} = \boldsymbol{h} \cdot \boldsymbol{r_A} + \boldsymbol{r_u}, \boldsymbol{v} = \boldsymbol{m} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_A} + \boldsymbol{r_v})$ の暗号文ペアを生成し，B に送る．B に対して，$\boldsymbol{v}$ には復号できないノイズ $\boldsymbol{s}$ が存在し，それを取り除けるような秘密情報を持っていないため，メッセージ $\boldsymbol{m}$ を知ることができない．

B は $\boldsymbol{b} = (b, 0, \ldots, 0)$ とし，さらに $\boldsymbol{r_B} \xleftarrow{\$} \mathcal{R}$ と $(\boldsymbol{e_u}, \boldsymbol{e_v}) \xleftarrow{\$} \mathcal{R}^2$ を生成する．$\boldsymbol{u}' = a \cdot \boldsymbol{u} + \boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u}$, $\boldsymbol{v}' = a \cdot \boldsymbol{v} + \boldsymbol{b} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v}$ を計算し，更新後の $\boldsymbol{u}$ と $\boldsymbol{v}$ を再ランダム化する．誤り訂正符号は線形符号であるため，更新後の $\boldsymbol{u}', \boldsymbol{v}'$ は

$$\boldsymbol{u}' = \begin{cases} \boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u} & (a = 0 \text{ の場合}) \\ \boldsymbol{u} + \boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u} & (a = 1 \text{ の場合}) \end{cases} \tag{7}$$

$$\boldsymbol{v}' = \begin{cases} \boldsymbol{b} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v} & (a = 0 \text{ の場合}) \\ \boldsymbol{v} + \boldsymbol{b} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v} & (a = 1 \text{ の場合}) \end{cases} \tag{8}$$

になる．B は $\boldsymbol{u}', \boldsymbol{v}'$ を A に送る．

最後に A は自分が持つ秘密情報を利用し，$\boldsymbol{v}' - \boldsymbol{u}' \cdot \boldsymbol{y}$ を復号する．計算した結果は

$$\begin{aligned} &\boldsymbol{v}' - \boldsymbol{u}' \cdot \boldsymbol{y} \\ =&(a\boldsymbol{m} + \boldsymbol{b})\mathbf{G} + \boldsymbol{x}(a\boldsymbol{r_A} + \boldsymbol{r_B}) - \boldsymbol{y}(a\boldsymbol{r_u} + \boldsymbol{e_u}) + (a\boldsymbol{r_v} + \boldsymbol{e_v}) \\ =&\begin{cases} \boldsymbol{b}\mathbf{G} + \boldsymbol{x}\boldsymbol{r_B} - \boldsymbol{y}\boldsymbol{e_u} + \boldsymbol{e_v} & (a = 0 \text{ の場合}) \\ (\boldsymbol{m} + \boldsymbol{b})\mathbf{G} + \boldsymbol{x}(\boldsymbol{r_A} + \boldsymbol{r_B}) - \boldsymbol{y}(\boldsymbol{r_u} + \boldsymbol{e_u}) + (\boldsymbol{r_v} + \boldsymbol{e_v}) & (a = 1 \text{ の場合}) \end{cases} \end{aligned} \tag{9}$$

式 (9) で示されたように，$\boldsymbol{v}' - \boldsymbol{u}' \cdot \boldsymbol{y}$ の計算結果はすでに $\boldsymbol{h}$ と $\boldsymbol{s}$ を取り除けた結果になる．最初の一ビットを取ることで，$a \cdot m + b$ が A に入手できる．

5. 提案プロトコルの正当性と安全性

5.1 正当性

この論文で提案した二者間線形関数秘匿計算プロトコルの正当性は明らかに符号 $\mathcal{C}$ の復号能力に依存する．具体的には，$\mathcal{C}$.Decode が $\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{y}$ を正しく復号すると仮定すると，以下の式が満たされる：

$$\mathrm{Decrypt}(sk, \mathrm{Encrypt}(pk, a \cdot \boldsymbol{m} + \boldsymbol{b})) = a \cdot \boldsymbol{m} + \boldsymbol{b}. \tag{10}$$

また，$\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{y}$ のエラーを ϵ と設定する．符号 $\mathcal{C}$ の誤り訂正能力について，エラーが

$$\epsilon = \begin{cases} \boldsymbol{x}\boldsymbol{r_B} - \boldsymbol{y}\boldsymbol{e_u} + \boldsymbol{e_v} & (a = 0 \text{ の場合}) \\ \boldsymbol{x}(\boldsymbol{r_A} + \boldsymbol{r_B}) - \boldsymbol{y}(\boldsymbol{r_u} + \boldsymbol{e_u}) + (\boldsymbol{r_v} + \boldsymbol{e_v}) & (a = 1 \text{ の場合}) \end{cases} \tag{11}$$

になる．Gaborit らの論文では，$\omega(\boldsymbol{x} \cdot \boldsymbol{r_2} + \boldsymbol{e} - \boldsymbol{y} \cdot \boldsymbol{r_1}) \leq \delta$ を満たすときに $\mathcal{C}$.Decode が正しく復号でき，実際に評価する時に w_r と w_e は同じ値に設定している．本稿で提案するプロトコルの $\boldsymbol{r_A}, \boldsymbol{r_B}, \boldsymbol{r_u}, \boldsymbol{r_v}$ のハミング重みを Gaborit らの w_r の 1/2 に設定し，$\boldsymbol{e_u}, \boldsymbol{e_v}$ のハミング重みを Gaborit らの w_e の 1/2 に設定すれば，式 (11) のエラーのハミング重みは Gaborit らエラーのハミング重みと一致するか，あるいはそれ以下になる．よって，Gaborit らの論文の結論が本稿でも成立する．符号長 n が大きくなるほど，誤り訂正符号の復号失敗率が小さくなる．適切な符号空間サイズ n とノイズのハミング重み w_r, w_e を設定していれば，復号失敗率は限りなく 0 に近づく．

5.2 安全性

本稿で提案するプロトコルの安全性要件は **2.4** 節で述べていた．本節では半誠実敵対者に対する安全性を証明する．

定理 1. 2-準巡回シンドローム復号仮定と 3-準巡回シンドローム復号仮定の下で 2PC プロトコルは半誠実な敵対者に対して安全に線形関数を計算する．

証明. まず，半誠実の敵対者 A について考える．グローバルパラメータを省略して，A のビューは $\mathrm{view}_A = (\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \boldsymbol{h}, \boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}; \boldsymbol{u}', \boldsymbol{v}')$ である．シミュレータ $S_A(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y})$ を以下のように構成する：

(1) ランダムに $\widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{r_A}}, \widetilde{\boldsymbol{r_u}}, \widetilde{\boldsymbol{r_v}}, \widetilde{\boldsymbol{u}'}, \widetilde{\boldsymbol{v}'} \xleftarrow{\$} \mathcal{R}$ を生成する．
ここで $\widetilde{r_A}, \widetilde{r_u}, \widetilde{r_v}$ のハミング重みは w_r とする．

(2) $(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{r_A}}, \widetilde{\boldsymbol{r_u}}, \widetilde{\boldsymbol{r_v}}, \widetilde{\boldsymbol{u}'}, \widetilde{\boldsymbol{v}'})$ を出力する．

まず，$\boldsymbol{h}, \boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}$ と $\widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{r_A}}, \widetilde{\boldsymbol{r_u}}, \widetilde{\boldsymbol{r_v}}$ は同じ分布に従うため，以下の式が成立する：

$$\begin{aligned} &(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{r_A}}, \widetilde{\boldsymbol{r_u}}, \widetilde{\boldsymbol{r_v}}, \widetilde{\boldsymbol{u}'}, \widetilde{\boldsymbol{v}'}) \\ \equiv_s\ &(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \boldsymbol{h}, \boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}, \widetilde{\boldsymbol{u}'}, \widetilde{\boldsymbol{v}'}). \end{aligned} \tag{12}$$

view_A では $\boldsymbol{u}' = a \cdot \boldsymbol{u} + \boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u}$, $\boldsymbol{v}' = a \cdot \boldsymbol{v} + \boldsymbol{b} \cdot \mathbf{G} + \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v}$ であり，

$$\begin{bmatrix} \boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u} \\ \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_n & 0 & \mathrm{rot}(\boldsymbol{h}) \\ 0 & \boldsymbol{I}_n & \mathrm{rot}(\boldsymbol{s}) \end{bmatrix} \begin{bmatrix} \boldsymbol{e_u} \\ \boldsymbol{e_v} \\ \boldsymbol{r_B} \end{bmatrix} \tag{13}$$

が成立する．よって，準巡回符号の 3-準巡回シンドローム復号決定仮定より，確率的多項式時間の敵対者は $(\boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u}, \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v})$ と一様乱数の区別がつかない．

$\boldsymbol{u}, \boldsymbol{v}$ も同様に 3-準巡回シンドローム復号決定仮定の下なので，$\boldsymbol{u}, \boldsymbol{v}$ と一様乱数の区別もつかない．よって，$\boldsymbol{u}', \boldsymbol{v}'$ の分布も一様乱数に近づき，以下の式が満たす：

$$\begin{aligned}&(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \boldsymbol{h}, \boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}, \widetilde{\boldsymbol{u'}}, \widetilde{\boldsymbol{v'}})\\ \equiv_c\ &(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \boldsymbol{h}, \boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}, \boldsymbol{u'}, \boldsymbol{v'}).\end{aligned} \tag{14}$$

よって，A のビュー view_A とシミュレータ S_A の分布が多項式時間の敵対者に対して区別がつかない：

$$\begin{aligned}&S_A(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y})\\ \equiv_c\ &\mathrm{view}_A(\boldsymbol{m}, \boldsymbol{x}, \boldsymbol{y}; \boldsymbol{h}, \boldsymbol{r_A}, \boldsymbol{r_u}, \boldsymbol{r_v}; \boldsymbol{u'}, \boldsymbol{v'}).\end{aligned} \tag{15}$$

次に半誠実の敵対者 B について考える．グローバルパラメータを省略して，B のビューは $\mathrm{view}_B = (a, b; \boldsymbol{h}, \boldsymbol{s}, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v})$ である．シミュレータ $S_B(a, b)$ を以下のように構成する：

(1) ランダムに $\widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{s}}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \widetilde{\boldsymbol{r_B}}, \widetilde{\boldsymbol{e_u}}, \widetilde{\boldsymbol{e_v}} \xleftarrow{\$} \mathcal{R}$ を生成する，ここで $\widetilde{\boldsymbol{r_B}}$ のハミング重みは w_r とし，$\widetilde{\boldsymbol{e_u}}, \widetilde{\boldsymbol{e_v}}$ のハミング重みは w_e とする．

(2) $(a, b; \widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{s}}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \widetilde{\boldsymbol{r_B}}, \widetilde{\boldsymbol{e_u}}, \widetilde{\boldsymbol{e_v}})$ を出力する．

$\boldsymbol{h}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v}$ と $\widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{r_B}}, \widetilde{\boldsymbol{e_u}}, \widetilde{\boldsymbol{e_v}}$ は同じ分布に従うため，以下の式が成立する：

$$\begin{aligned}&(a, b; \widetilde{\boldsymbol{h}}, \widetilde{\boldsymbol{s}}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \widetilde{\boldsymbol{r_B}}, \widetilde{\boldsymbol{e_u}}, \widetilde{\boldsymbol{e_v}})\\ \equiv_s\ &(a, b; \boldsymbol{h}, \widetilde{\boldsymbol{s}}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v}).\end{aligned} \tag{16}$$

4 節の式 (6) により，$\boldsymbol{s}$ は 2-準巡回シンドローム復号仮定に帰着でき，多項式時間の敵対者に対して分布が一様乱数と区別がつかない．よって，以下の式が満たす：

$$\begin{aligned}&(a, b; \boldsymbol{h}, \widetilde{\boldsymbol{s}}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v})\\ \equiv_c\ &(a, b; \boldsymbol{h}, \boldsymbol{s}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v}).\end{aligned} \tag{17}$$

また，$\boldsymbol{u}, \boldsymbol{v}$ は 3-準巡回シンドローム復号決定仮定より，確率的多項式時間の敵対者は $(\boldsymbol{h} \cdot \boldsymbol{r_B} + \boldsymbol{e_u}, \boldsymbol{s} \cdot \boldsymbol{r_B} + \boldsymbol{e_v})$ と一様乱数の区別がつかないため．以下が成立する：

$$\begin{aligned}&(a, b; \boldsymbol{h}, \boldsymbol{s}, \widetilde{\boldsymbol{u}}, \widetilde{\boldsymbol{v}}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v})\\ \equiv_c\ &(a, b; \boldsymbol{h}, \boldsymbol{s}, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v}).\end{aligned} \tag{18}$$

よって，B のビュー view_B とシミュレータ S_B の分布が多項式時間の敵対者にたいして区別がつかない：

$$\begin{aligned}&S_B(a, b)\\ \equiv_c\ &\mathrm{view}_B(a, b; \boldsymbol{h}, \boldsymbol{s}, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{r_B}, \boldsymbol{e_u}, \boldsymbol{e_v}).\end{aligned} \tag{19}$$

□

6. まとめ

本研究では，準巡回符号を利用した公開鍵暗号方式 HQC を応用し，準巡回シンドローム復号問題に基づく二者間線形関数の秘匿計算プロトコルを提案した．このプロトコルは符号ベースの仮定により耐量子安全性を持ち，また紛失通信を使わないという特徴を持つ．

なお，本研究で提案されたプロトコルは $\mathbb{F}_2$ に属する入力のみ対応できる，今後は二者間の入力を $\mathbb{F}_2$ から $\mathbb{F}_q$ へ拡張することが課題である．

謝辞　本研究の一部は科学研究費補助金基盤研究 (A) No. 16H01705，基盤研究 (B) No. 17H01695，若手研究 (B) No. 17K12640，及び JSPS 科研費基盤 C (JP15K00183)，Microsoft Research Asia の共同研究費，科学技術振興機構（JST）の CREST(JPMJCR1404) と国際科学技術協力基盤整備事業（日本一台湾研究交流），及び文部科学省の情報技術人材育成のための実践教育ネットワーク形成事業 分野・地域を越えた実践的情報教育協働ネットワークの助成を受けています．

参考文献

[1] C. Aguilar-Melchor, O. Blazy, J. Deneuville, P. Gaborit, and G. Zémor. Efficient encryption from random quasi-cyclic codes. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 64(5), May 2018.

[2] M. Barni, P. Failla, V. Kolesnikov, R. Lazzeretti, A.-R. Sadeghi, and T. Schneider. Secure evaluation of private linear branching programs with medical applications. *Computer Security (ESORICS)*, pages 424–439, 2009.

[3] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser. Machine learning classification over encrypted data. *Symposium on Network and Distributed System Securtiy (NDSS)*, 2015.

[4] S. Ghosh, J. B. Nielsen, and T. Nilges. Maliciously secure oblivious linear function evaluation with constant overhead. *23rd International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, 2017.

[5] Y. Ishai, M. Prabhakaran, and A. Sahai. Secure arithmetic computation with no honest majority. *6th Theory of Cryptography Conference on Theory of Cryptography (TCC)*, pages 294–314, 2009.

[6] D. J. Wu, T. Feng, M. Naehrig, and K. Lauter. Privately evaluating decision trees and random forests. In *Proceeding on Privacy Enhancing Technologies*, 4, pages 1–21, 2016.

[7] A. C. Yao. Protocols for secure computations. *23rd Annual Symposium on Foundations of Computer Science (FOCS 1982)*, pages 160–164, 1982.

[8] A. C. Yao. How to generate and exchange secrets. *27th Annual Symposium on Foundations of Computer Science (FOCS 1986)*, pages 162–167, 1986.

[9] 千田浩司. 安全な情報処理を目指す秘密計算技術の研究動向と実用化に向けた取り組み. **情報処理**, 54(11), Nov 2013.