7V-11

紛失多項式評価の拡張と安全な情報埋め込みサービスの一構成

Enhanced Oblivious Polynomial Evaluation and Secure Information Embedding Service

平原 耕一† 折笠 大典‡ 小瀬木 浩昭‡ 武田 正之†
Kouichi Hirahara Daisuke Orikasa Hiroaki Ozeki Masayuki Takeda

1. はじめに

紛失多項式評価（Oblivious Polynomial Evaluation，以下 OPE）[1],[2]は2者間による秘密通信プロトコルである．Alice は入力値αを持ち，Bob は1変数多項式$P(x)$を持つ．プロトコルを実行後，Alice は$P(\alpha)$を得る．OPE を用いることで，2者間においてお互いの秘密情報を保護したままデータ処理を行うことができる．本稿では，上述の$P(x)$として利用可能な関数が1変数多項式に限定されていた従来の OPE を，多変数多項式が利用可能な，多変数 OPE へと拡張する．この拡張により，一度に2変数以上の多項式関数が必要な処理を，従来の OPE と同じ安全性を保ったまま実現可能となる．また，多変数 OPE を実装し，その評価を行う．さらに，多変数 OPE が有効性を持つ具体的な例として，情報埋め込みサービスの構成を示す．

2. 基礎知識

2.1. Oblivious Transfer [1],[3],[4]

k-out-of-N Oblivious Transfer（以下，OT）では，Bob はN個の秘密$m_1,m_2,\cdots,m_N$を，Alice はk($\leq N$)個の秘密$a_1,a_2,\cdots,a_k(a_i\in \mathrm{N}, i=1,\cdots,k)$を持っており，プロトコル終了後，Alice は$m_{a_1},m_{a_2},\cdots,m_{a_k}$を取得する．その際，Alice は，$m_{a_1},m_{a_2},\cdots,m_{a_k}$以外についてまったく分からない，Bob は，$a_1,a_2,\cdots,a_k$についてまったく分からない，という要件を満たす．

2.2. Oblivious Polynomial Evaluation

OPE は2者間のプロトコルで，Alice は定数αを，Bob は1変数多項式$P(x)$を持っており，プロトコル終了後，Alice は$P(\alpha)$を取得する．その際，(1)Alice は，$P(x)$のひとつの値$P(\alpha)$だけを得ることができる，(2)Bobは，αと$P(\alpha)$についてまったく分からない，という2つの要件を満たす．

次に，OPE のプロトコルについて述べる．**(1)両者の秘密を定義する:** Bob の秘密にしたい 1 変数多項式は，$P(x)=\sum_{i=0}^{d_P} a_i x^i$で定義される．また，Alice の秘密にしたい値としてαを定義する．**(2)Bob は，2変数多項式の中にPを隠す:** Bob はd次のランダムな多項式$P'(x)=\sum_{i=1}^{d} b_i x^i (s.t\ P'(0)=0)$を生成する$(d=d_P*K)$.ここでセキュリティ定数を$K(\in \mathrm{N})$とする．セキュリティ定数とは，Bob が任意に定める自然数で，セキュリティ定数が大きいほどP'の次数が高くなり，Pの推測をより困難とするパラメータである．Bob は，2変数多項式を以下のように定義する．$Q(x,y)=P'(x)+P(y)$．2変数多項式Qは全てのyにおいて，$Q(0,y)=P(y)$となる．**(3)Alice はαを1変数多項式Sの中に隠す:** Alice はランダムにK次の多項式$S(x)(s.t\ S(0)=\alpha)$を生成する．Alice は$R(x)=Q(x,S(x))$を用いて，$P(\alpha)$を得ようとする．$R(0)=Q(0,S(0))=P(S(0))=P(\alpha)$として求める．

(4)Alice は Bob に値を送信: $d_R=d=d_P*K$と定義する．Alice は(d_R+1)個のデータ$(x_i,S(x_i))$を作成し，ダミーデータと混ぜて，Bob に送信する．**(5)Bob は受け取ったデータを処理する:** Bobは，(4)でAliceから送られたデータを計算し，$Q(x_i,S(x_i))$を生成する．**(6)Alice は Bob からデータを受け取り$R(x)$を再構築する:** Bob が(5)で処理したデータの中から，Alice は，(d_R+1)-out-of-N OT（Nは，(d_R+1)+ダミーデータの数)を用い，(4)で Bob に送った，x_iを回収する．そこから$R(x)$（次数はd_R）を再構築し，$R(0)=P(\alpha)$を得る．以上が OPE のプロトコルである．

3. 多変数 OPE への拡張

従来の OPE では，2者間において，Bob の秘匿できる関数が，1変数多項式という制約があった．情報を埋め込むアルゴリズムでの OPE の利用を考える場合，少なくとも，情報の埋め込み対象データと，埋め込み情報の2変数が必要である．そこで，OPE を多変数でも使えるよう，n変数 k 次多項式（$k=n*m$，mは1つの変数の最大次数）への拡張について述べる．

3.1. 定義

拡張した OPE（多変数 OPE）プロトコルは次の機能を実現する．Alice は定数列$\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\cdots,\alpha_n)$を持っており，$\boldsymbol{\alpha}$を Bob に知られずに，$P(\boldsymbol{\alpha})$を得たいとする．Bob は，$n$変数 k 次多項式 $P(\boldsymbol{x})$ $(\boldsymbol{x}=(x_1,x_2,\cdots,x_n))$を持っており，多項式 $P(\boldsymbol{x})$ については，Alice に知られたくないとする．プロトコル終了後，Alice は$P(\boldsymbol{\alpha})$を取得する．その際，(1)Alice は，$P(\boldsymbol{x})$の1つの値$P(\boldsymbol{\alpha})$だけを得ることができる，(2)Bob は，$\boldsymbol{\alpha}$と$P(\boldsymbol{\alpha})$についてまったく分からない，という2つの要件を満たす．

3.2. プロトコル

(1)Bob が持っている多項式$P(d_P=k)$を定義する: この多項式$P(x_1,x_2,\cdots,x_n)=\sum_{k_1=0}^{m}\sum_{k_2=0}^{m}\cdots\sum_{k_n=0}^{m} a_{k_1k_2\cdots k_n}x_1^{k_1}x_2^{k_2}\cdots x_n^{k_n}$は Alice に知られたくないとする．**(2)Bob はPを隠す多項式P' $(P'(0,0,\cdots,0)=0)$を準備する:** $k'=k\cdot K$と定義すると，$P'(x_1,x_2,\cdots,x_n)=\sum_{k_1=0}^{k'}\sum_{k_2=0}^{k'}\cdots\sum_{k_n=0}^{k'} b_{k_1k_2\cdots k_n}x_1^{k_1}x_2^{k_2}\cdots x_n^{k_n}(b_{00\cdots 0}=0)$．また，次の多項式$Q$を定義する．$Q((x_1,x_2,\cdots,x_n),(y_1,y_2,\cdots,y_n))=P'(x_1,x_2,\cdots,x_n)+P(y_1,y_2,\cdots,y_n)$．**(3)Alice は定数列$\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\cdots,\alpha_n)$を隠す多項式関数列$\boldsymbol{S}=(S_1,S_2,\cdots,S_n)$を準備する:** 各多項式は，$S_i(0)=\alpha_i$ $(\deg S_i=K, i=1,\cdots,n)$とする．**(4)Alice は Bob にデータを送信:** Alice の用意する，真の値の数sは，$s=\sum_{i=0}^{k*n}{}_{(n+i-1)}C_i$となる．$\{\gamma_i^j \mid i=1,\cdots,n, j=1,\cdots,s\}(\gamma_i^j\neq 0)$を任意に生成し，$(\gamma_1^j,\gamma_2^j,\cdots,\gamma_n^j)$と，それらを$(S_1,S_2,\cdots,S_n)$に代入した値$(S_1(\gamma_1^j),S_2(\gamma_2^j),\cdots,S_n(\gamma_n^j))$からなる組を$s$個作る．また，$t$組のダミーデータを同様に準備する．このデータをダミーデータと混ぜて Bob に送信する．**(5)Bob は Alice から送られてきたデータを処理する:** Bobは，(4)で送られてきたデータをQに

†東京理科大学 理工学部 情報科学科,
Dept. of Information Sciences, Tokyo University of Science
‡東京理科大学大学院 理工学研究科 情報科学専攻,
Graduate School of Science and Technology,
Tokyo University of Science

代入し計算する. **(6)**Alice は$P(\alpha_1,\alpha_2,\cdots,\alpha_n)$**を得る:** $s+t=u$とすると, Alice は Bob が(5)で計算した結果をs-out-of-u OT を行い取得し, OPE プロトコルにより $R(\boldsymbol{x})(=Q(x_1,x_2,\cdots,x_n,(S_1(x_1),S_2(x_2),\cdots,S_n(x_n))))$を再構築し, $\boldsymbol{x}=(0,0,\cdots,0)$を代入する.

$$R(0,0,\cdots,0)=Q((0,0,\cdots,0),(S_1(0),S_2(0),\cdots,S_n(0)))$$
$$=P(S_1(0),S_2(0),\cdots,S_n(0))$$
$$=P(\alpha_1,\alpha_2,\cdots,\alpha_n)$$

よって, Alice は$P(\alpha_1,\alpha_2,\cdots,\alpha_n)$を得る.

4. データとアルゴリズムの双方を保護する安全な情報埋め込みサービスの構成

4.1. 構成

この章では, 埋め込みサービスを, クライアントから埋め込み対象データと埋め込み情報をサーバが受け取り, サーバの埋め込みプログラムを用いて埋め込み処理を行った後, 埋め込み済データをクライアントに返す一連のサービスと定義する. また, 提案する構成はプログラムの提供者, サーバ(受託業者), クライアントの3つの主体からなる.

多変数多項式 OPE を用いた情報埋め込み処理プロトコルでは, Alice は埋め込み対象データ列$\boldsymbol{\alpha}=(\alpha_1,\alpha_2,\cdots,\alpha_m)$と, 埋め込み情報列$\boldsymbol{\beta}=(\beta_1,\beta_2,\cdots,\beta_n)$を持っており, $\boldsymbol{\alpha}$と$\boldsymbol{\beta}$を Bob に知られずに, 埋め込み済データ$P(\boldsymbol{\alpha},\boldsymbol{\beta})$を得たいとする. Bob は, 埋め込み処理を行う$(m+n)$変数多項式$P(\boldsymbol{x})$ $(\boldsymbol{x}=(x_1,x_2,\cdots,x_{(m+n)}))$を持っており, 埋め込み処理をする際, 埋め込み処理関数$P(\boldsymbol{x})$については, Alice に知られたくないとする. 多変数 OPE により, (1)Alice は, $P(\boldsymbol{x})$の1つの値である$P(\boldsymbol{\alpha},\boldsymbol{\beta})$しか得ることはできない, (2)Bob は, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$と$P(\boldsymbol{\alpha},\boldsymbol{\beta})$についてまったく分からない, という2つの要件が満たされる.

4.2. 手順

提案モデルの手順は, 以下のようになる. **(1)プログラムの委託:**プログラムの提供者は, 多変数 OPE を利用可能な, 埋め込みプログラムを作成する. プログラムを難読化した後, サーバに情報埋め込みサービス業務を委託する. **(2)クライアントがデータを送信:**クライアントは, 埋め込み対象データと埋め込み情報を, 多変数 OPE を用いて暗号化してサーバに送信する. **(3)サーバによる埋め込み処理:**サーバは, (2)でクライアントから送られてきた埋め込み対象データと埋め込み情報を, (1)で委託された埋め込みプログラムを用いて, 埋め込み処理を行う. **(4)クライアントが埋め込み済データを受信:**クライアントは, 暗号化された埋め込み済データをサーバから受信し, 多変数 OPE を用いて復号する.

5. 実装と評価

3 章で述べた多変数 OPE について, Java 2 Standard Edition 5.0 を利用して実装し, 以下の環境で評価を行った. CPU: Pentium M 1100MHz, RAM: 256MB, OS: Windows XP Pro SP2, VM: JRE 1.5.0_01. また, この章では真の値の数をs, ダミーデータ数をtとおく.

5.1. 通信コストの予測

多変数OPEにおけるダミーデータ数と, 計算量との関係を予測する. ここで, セキュリティ係数$K=1$で一定, また, 多項式の1つの変数の最大次数mは一定であると仮定する.

ダミーデータ数による処理時間の増加に最も関係しているのは, 3.2 節の(4)でのダミーデータの生成である. Bob が送信する$\{\gamma_i^j\}$は$\{\gamma_1^k,\gamma_2^k,\cdots,\gamma_n^k\}\neq\{\gamma_1^l,\gamma_2^l,\cdots,\gamma_n^l\}(k\neq l,k,l=1,\cdots,s)$を満たす必要がある. この条件を満たすため, 値の組の生成時にすでに生成されている各組との比較が必要になる. $(s+t)$組のデータの総比較回数は, $\sum_{i=1}^{(s+t)-1} i=(s+t)(s+t-1)/2$回となり, ダミーデータ数による処理時間の増加分は, 2 次関数に従うと推測される.

5.2. 通信コストの評価(図 1)

5.1 節で行った予測との比較のために, 多変数 OPE のプログラムの通信コストの評価を実際に行った. ここで, 扱う多項式は$P(x_1)=x_1+1$, $P(x_1,x_2)=x_1x_2+x_1+x_2+1$とした. また, この場合の$s$は順に, 2 個, 15 個(3.2 節(4)の式より)である. **図 1** より ダミーデータが 10^4 個までは双方とも処理時間の増加はほとんど見られない. よって, ダミーデータが 10^4 個以下ならば, 実装プログラムの処理時間に影響を与えないといえる. 一方 10^5 個以降は双方ともに処理時間の増加が大きく, ダミーデータ数の影響が現れている.

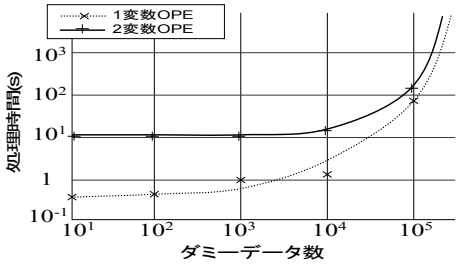

図 1 ダミーデータ数と処理時間との関係

6. まとめ

本稿では, 利用可能な関数が1変数多項式に限定されていた従来のOPEを, 多変数多項式が利用可能な, 多変数OPEへと拡張した. この拡張により, 一度に2変数以上の多項式関数が必要な処理を, 従来のOPEと同じ安全性を保ったまま実現可能となる. また, 拡張した多変数 OPE が有効性を持つ具体的な例として, クライアント側の埋め込み対象データと埋め込み情報, サーバ側の埋め込みプログラムの双方を保護可能な, 安全な情報埋め込みサービスの構成を示した. そして, 実装した多変数 OPE プログラムの処理時間について評価を行った.

今後の課題として, OPE で扱える関数の多項式以外への拡張, 安全な情報埋め込みサービスの実装などがある.

参考文献

[1] Moni Naor, Benny Pinkas: Oblivious transfer and polynomial evaluation, Proc. of the 31st Symp. on Theory of Computer Science (STOC'99), pp.245-254 (1999).

[2] 駒木 寛隆, 渡邊 裕治, 花岡 悟一郎, 今井 秀樹: 検証可能な紛失多項式評価, SCIS2001, pp.471-476 (2001).

[3] Shimon Even, Oded Goldreich and Abraham Lempel: A Randomized Protocol for Signing Contracts, Comm. of the ACM, Vol.28, No.6, pp.637-647 (1985).

[4] Sheng Zhong and Yang Richard Yang: Verifiable Distributed Oblivious Transfer and Mobile Agent Security, DIALM-POMC'03, pp.12-21 (2003).