Computer Security Symposium 2020
26 - 29 October 2020

ZDDのGarbled Circuit構成法の安全性評価

増井 孝之[1,a)]　森田 光[1]

概要：Kruger らは, 論理関数の表現手法の 1 つである BDD の Garbled Circuit の構成法を提案し, semi-honest 仮定での安全性を証明した [3]. 本稿では, 著者らによる BDD の改良版である ZDD に対する Garbled Circuit の構成法 [5] に対して, semi-honest 仮定の安全性を証明する.

キーワード：秘密計算, Garbled Circuit, ZDD

Safety evaluation of a method to construct Garbled Circuits for ZDD

Takayuki Masui[1,a)]　Hikaru Morita[1]

Abstract: Kruger et al. proposed a construction method of Garbled Circuits for BDD, and proved that it is safe under the semi-honest assumption. In this paper, the authors prove that a construction method of Garbled Circuits for ZDD is safe under the semi-honest assumption. The method has been proposed for ZDD which is an improved version of BDD.

Keywords: secure computation, Garbled Circuit, ZDD

1. はじめに

データの秘匿性を維持したまま, 計算過程も秘密とする秘密計算の構成法の 1 つに, Yao に基づく Garbled Circuit がある [1]. この方法は秘密分散や準同型暗号と違い, 秘密計算する関数も秘匿にできる.

Garbled Circuit では, 任意の論理関数は AND ゲート, OR ゲート, NOT ゲートの組み合わせた論理回路で実現できるため, 各ゲートごとに Garbled Computation Table(以降 GCT) と呼ぶ暗号文で構成するテーブルを与えて, ゲートごとに秘密計算をし, 論理関数の秘密計算をする. その実装では GCT のメモリ量の削減が課題となる.

論理関数を表す手法の 1 つに BDD (Binary Decision Diagram) があり, LSI の設計支援と論理合成・最適化などに広く用いられている [2].

Kruger らは BDD に対する Garbled Circuit 構成法を提案し, そのメモリ圧縮効果を示し, semi-honest 仮定での安全性を証明した [3].

BDD は論理関数の普通の真理値表に適用するが, 論理関数を集合族に適用し, 圧縮したのが, 湊が提案した ZDD(Zero–suppressed BDD) である [4].

著者らは ZDD に対する Garbld Circuit の構成法を提案し, 入力データの一致を検出する論理回路において, Kruger らの提案よりもメモリ圧縮効果があることを示した [5].

本稿では, その ZDD の Garbled Circuit 構成法に対する安全性評価を行う. 具体的には, 2 者間秘密計算の実行時に受信するメッセージ群が, 秘匿情報の推測を試みる semi-honest 仮定の攻撃者に流出しても安全であることを示す.

本稿は, 2 節では BDD と ZDD の違いを紹介し, 3 節では著者らが提案した ZDD の Garled Circuit 構成法について説明し, 4 節では安全性評価での準備を与え, 5 節で安全性評価を示し, 6 節でまとめる.

[1] 神奈川大学大学院
Graduate School of Kanagawa University
[a)] r201970122af@jindai.jp

2. BDD と ZDD [4, 6]

BDD は論理関数を表現する. $F(a,b,c) = a\bar{b}c + \bar{a}b\bar{c}$ の例を図 1 に示す. Knuth の本 [10] にならい終端は 1 を $\top$ に, 0 を $\bot$ で表す. BDD は二分木の変数順序を固定した上で, 冗長ノードを削除することで得る. 一方, 論理関数を集合族へ読み替えると $F(a,b,c) = a\bar{b}c + \bar{a}b\bar{c}$ に対応する集合族は $S = \{ac, b\}$ となる. ZDD は集合族に含まれるアイテムのみを明示的に表現することで冗長ノードを削除し, BDD を圧縮した二分木構造にする. $S = \{ac, b\}$ の ZDD は図 1(b) である. ZDD では $\top$ に至らなかった場合 $\bot$ とみなす.

一般的に, 解が疎である論理関数を扱う時, ZDD の圧縮効果が高まる [6].

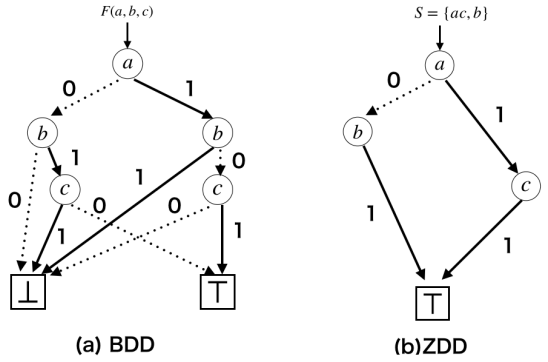

図 1 BDD と ZDD の構成例 ($F(a,b,c) = a\bar{b}c + \bar{a}b\bar{c}$)

3. ZDD に対する Garbled Circuit 構成法 [5]

本節では, 著者らが提案した ZDD の Garbled Circuit 構成法を示す. この手法では, ZDD のノードと枝ごとに鍵となる乱数を生成し, GCT を構成することで秘密計算を実現する.

3.1 記号の定義

- V: 変数ノードのシンボルの集合
- W: 入力変数の集合
- $\xleftarrow{R}$ 確率変数からその確率分布に従って 1 つのサンプルをランダムに選ぶ. 例えば $s \xleftarrow{R} X$ ならば確率変数 X から 1 つのサンプル s をその確率分布に従ってランダムに選ぶことを意味する
- $low(v_\alpha)$: 変数ノードのシンボル $v_\alpha \in V$ を入力とし, 0– 枝側の子ノードのシンボル $v_\beta \in V$ を返す関数.
- $high(v_\alpha)$: 変数ノードのシンボル $v_\alpha \in V$ を入力とし, 1– 枝側の子ノードのシンボル $v_\gamma \in V$ を返す関数.
- $E_K(m)$: 暗号文. 鍵 K で, 平文 m に復元される.
- N, λ: セキュリティパラメータ
- $x||y$: x と y を連結する.

3.2 構成法

(1) 論理関数の場合分け二分木の変数の順序を固定した上で, 等価ノードの共有を行い, 選択する要素のみを明示的に表現するグラフを生成する. 通常の ZDD では 1-枝の行先が $\bot$ であるノードを削除するが, 本構成法ではノードは削除せず, 1-枝のみを削除する.

(2) 変数ノードの個数分だけ, GCT の ID となる $L_v \in \{0,1\}^p, v \in V$ を選択する. ビット長 p に上限は指定しないが, 各 GCT に対して一意な値にする必要があるため, 下限は $\lceil log_2|V| \rceil$ ビットである.

(3) $\forall v \in V$, および $\top$ ノードに対し, 鍵となる $K_v \xleftarrow{R} \{0,1\}^N$ と $K_\top \xleftarrow{R} \{0,1\}^{N+p}$ の値を選択する.

(4) $\forall w \in W$, および各変数の入力値 $i = \{0,1\}$ に対し, 鍵となる $K_i^w \xleftarrow{R} \{0,1\}^N$ の値を選択する.

(5) 各変数ノードに対して GCT を構成する. 変数ノード $v \in V$, 入力変数 $w \in W$ のノードに対する GCT の構成法を以下の 3 つの場合に分けて表で示す. (a)v が 2 つの枝（0-枝と 1-枝）を持っている場合 (表 1). (b)v が 0-枝のみを持っている場合 (表 2). (c)v が 1-枝のみを持っている場合 (表 3).

表 1 変数ノードが 2 つの枝を持っている場合の GCT

ID	L_v
0-枝暗号文	$E_{K_v}(E_{K_0^w}(K_{low(v)}\|\|L_{low(v)}\|\|0^\lambda))$
1-枝暗号文	$E_{K_v}(E_{K_1^w}(K_{high(v)}\|\|L_{high(v)}\|\|0^\lambda))$

表 2 変数ノードが 0-枝のみを持っている場合の GCT

ID	L_v
0-枝暗号文	$E_{K_v}(E_{K_0^w}(K_{low(v)}\|\|L_{low(v)}\|\|0^\lambda))$

表 3 変数ノードが 1-枝のみを持っている場合の GCT

ID	L^v
1-枝暗号文	$E_{K_v}(E_{K_1^w}(K_{high(v)}\|\|L_{high(v)}\|\|0^\lambda))$

例題として 2 節で用いた論理関数, $F(a,b,c) = a\bar{b}c + \bar{a}b\bar{c}$ に対して構成した ZDD, および GCT を図 2 に示す.

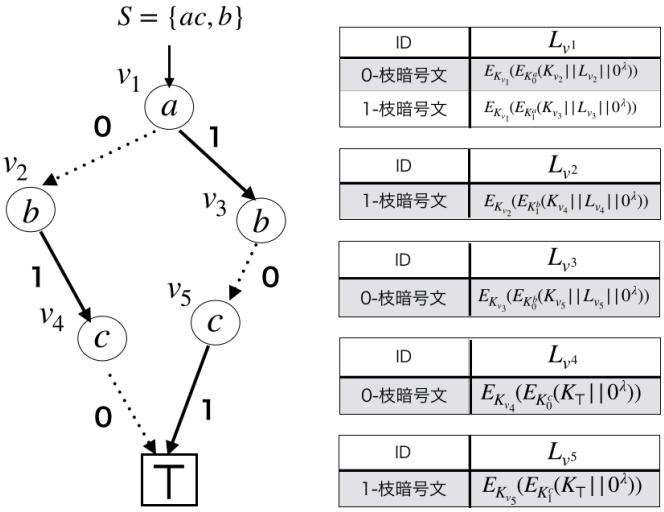

図 2 $S = \{ac, b\}$ に対し, 提案手法により構成した ZDD およびその GCT

3.3 計算法

提案手法の計算法を図 2 の例題を用いて説明する. 2 者間の秘密計算のパーティをそれぞれ P_1, P_2 とする. また, P_1 の入力を $x = \{a, c\}$, P_2 の入力を $y = \{b\}$ とする (図 3).

(1) P_1 は Garbled Circuit 化した ZDD, 根ノードに対応する鍵とその GCT の ID を連結した $K_{v_1}||L_{v_1}$, P_1 の入力 x の入力変数 $\{a, c\}$ に対応する鍵 $K_i^a, K_j^c, i \in \{0,1\}, j \in \{0,1\}$ を P_2 に送る.

(2) P_2 は自身の入力 y の入力変数 $\{b\}$ に対応する鍵 $K_l^b, l \in \{0,1\}$ を P_1 から 1-out-of-2 OT [7] により入手する.

(3) P_2 は根ノードに対応する鍵 K_{v_1}, および各入力変数に対応する鍵を用いて GCT の暗号文を復号する.

正しい各変数ノードに対応する鍵と, 入力変数に対応する鍵の組み合わせで復号した場合に限り, 復号結果の末尾に 0^λ が現れるため, 子ノードに対応する鍵と, 次に復号をする GCT の ID が手に入る. 復号に成功した結果, $K^\top$ を入手した場合, ZDD の出力を $\top$ とし, 復号結果の末尾に 0^λ が現れなかった場合, ZDD の出力を $\perp$ とし, 2 者間で共有する.

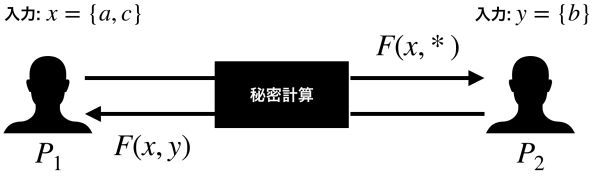

図 3 P_1,P_2 間の論理関数 $F(x, y)$ の秘密計算

4. 準備

安全性評価で用いるいくつかの記法と暗号学的要素技術を導入する.

4.1 無視できる関数

関数 $\epsilon(n) : N \to R$ が n に関して無視できるとは, 任意の多項式 $p(.)$ に対して、ある自然数 n_p が存在し, $n_p \leq n$ となる任意の自然数 n について

$$\epsilon(n) < \frac{1}{|p(n)|}$$

が成り立つ時である. 以降, 関数 $f(n)$ が n に関して無視できる時, $f(n) < \epsilon(n)$ のように書く. また, 無視できる関数は以下の 3 つの性質を満たす. ($q(n)$ は n に関する多項式)

$$f_1(n) + f_2(n) < \epsilon(n)$$
$$f_1(n) f_2(n) < \epsilon(n)$$
$$q(n) f(n) < \epsilon(n)$$

4.2 確率的多項式時間アルゴリズム

与えられた入力に対してアルゴリズム A が内部で一様ランダムな値を使用し, 出力を確率的に決める多項式時間アルゴリズムを確率的多項式時間アルゴリズムという.

4.3 計算量的識別不可能性

任意の確率的多項式時間アルゴリズムである識別器 D に対して, 以下の式が成り立つ時, 2 つの確率変数 X_n と Y_n は計算量的識別不可能であるという.

$$|Pr[D(1^n, X_n) \to 1] - Pr[D(1^n, Y_n) \to 1]| < \epsilon(n)$$

以降, 2 つの確率変数 X_n と Y_n が計算量的識別不可能である時, $X_n \equiv_c Y_n$ のように記す.

4.4 2 者間通信プロトコルの View

2 つの入力 x,y を持つ関数 $f(x, y)$ を評価する P_1 と P_2 の 2 者間で実行される通信プロトコルを $\pi(x, y)$ とする. $\pi(x, y)$ を実行する際, $P_i, i \in \{0,1\}$ が受け取るメッセージ群, および入力と出力を $View_{P_i}^{\pi}(x, y)$ とする。

4.5 Simulation-based Privacy

$View_{P_1}^{\pi}(x, y)$, $View_{P_2}^{\pi}(x, y)$ に対して, 以下の式を満たす確率的多項式時間アルゴリズムであるシミュレータ $S_{P_1}^{\pi}, S_{P_2}^{\pi}$ が存在する時, 関数 $f(x, y)$ を評価する通信プロトコル π は semi-honest 仮定で安全であるという.

$$View_{P_1}^{\pi}(x, y) \equiv_c S_{P_1}^{\pi}(x, f(x, y))$$
$$View_{P_2}^{\pi}(x, y) \equiv_c S_{P_2}^{\pi}(y, f(x, y))$$

直感的には, あるパーティの $View$ の確率分布が, そのパーティの入力と出力を引数とするシミュレータが作った $View$ の確率分布と計算量的に識別ができない事を示すことで, $View$ から秘匿情報 (この場合 x もしくは y) を推測することはできないことを意味している. 詳細は [8] 参照.

4.6 1-out-of-2 OT プロトコルの安全性

提案手法による秘密計算を実行する際に使用する 1-out-of-2 OT プロトコルは, semi-honest 仮定において安全であるとする. つまり, $View_{P_1}^{OT}((m_0,m_1),b)$, $View_{P_2}^{OT}((m_0,m_1),b)$ に対して以下の式を満たすシミュレータ $S_{P_1}^{OT}, S_{P_2}^{OT}$ が存在する.

$$View_{P_1}^{OT}((m_0,m_1),b) \equiv_c S_{P_1}^{OT}(m_0,m_1)$$
$$View_{P_2}^{OT}((m_0,m_1),b) \equiv_c S_{P_2}^{OT}(b,m_b)$$

ここでは, P_1 を OT プロトコルの Sender とし入力を 2 つのメッセージ $m_0, m_1 \in \{0,1\}^n$, P_2 を Receiver とし入力を $b \in \{0,1\}$ としている. 詳細は [8, Section 7.3.2] 参照.

4.7 二重暗号文の安全性

GCT を構成する際に用いる, 2 つの共通鍵で暗号化されている暗号文の安全性は, 確率的多項式時間アルゴリズムである識別器 D と暗号化オラクル $\mathcal{O}$ との間で行われる実験 $Exp_{D(n)}^{double-\sigma}(K_v, K_b, (m_0,m_1))$ で定義される.

(1) 識別器 D は 2 つの鍵 $K_v, K_b \xleftarrow{R} \{0,1\}^n$, および n ビット長の 2 つのメッセージ m_0, m_1 を受け取り, 暗号化オラクル $\mathcal{O}$ に送る.

(2) 暗号化オラクルは, 鍵である $\bar{K_b} \xleftarrow{R} \{0,1\}^n$ を選択する. その後, 選択ビット $\sigma \in \{0,1\}$ を受け取り, 以下のような 2 つの暗号文を識別器 D に送る.

$$(c_0 = E_{K_v}(E_{K_b}(m_0)), c_1 = E_{K_v}(E_{\bar{K_b}}(m_\sigma)))$$

(3) 識別器 D は σ の予測値 σ' を出力する.

この時, 使用している共通鍵暗号方式が選択平文攻撃下で強秘匿性を満たすなら, 以下が成り立つ.

$|Pr[Exp_{D(n)}^{double-0}(K_v, K_b, (m_0,m_1)) \rightarrow 1] - Pr[Exp_{D(n)}^{double-1}(K_v, K_b, (m_0,m_1)) \rightarrow 1]| < \epsilon(n)$

詳細は [9,Section 3.2.1] 参照.

4.8 $Active-GCTs$,$Inactive-GCTs$

著者らが提案した手法を用いて秘密計算をする際, 計算に使用する GCT を $Active-GCTs$ とし, 使用しない GCT を $Inactive-GCTs$ とする. また, ID が L_{v_i} の GCT を $GCT_{L_{v_i}}$ のように表記する. 例えば, 3.2 節で例題として用いた Garbled Circuit 化した ZDD において, P_1 の入力を $x = \{a \leftarrow 1, c \leftarrow 1\}$, P_2 の入力を $y = \{b \leftarrow 0\}$ とした時, $Active-GCTs$, $Inactive-GCTs$ は以下のようになる.

$$Active-GCTs = (GCT_{L_{v_1}}, GCT_{L_{v_3}}, GCT_{L_{v_5}})$$
$$Inactive-GCTs = (GCT_{L_{v_2}}, GCT_{L_{v_4}})$$

4.9 $Active-keys$, $Inactive-keys$

著者らが提案した手法を用いて秘密計算をする際, 計算に使用する変数ノードに対応する鍵を K_v^{act}, 入力変数に対応する鍵を K_e^{act} とする. また, 使用しなかった鍵をそれぞれ K_v^{inact}, K_e^{inact} とする. 4.8 節で挙げた例では以下のようになる.

$$K_v^{act} = (K_{v_1}, K_{v_3}, K_{v_5})$$
$$K_v^{inact} = (K_{v_2}, K_{v_4})$$
$$K_e^{act} = (K_1^a, K_0^b, K_1^c)$$
$$K_e^{inact} = (K_0^a, K_1^b, K_0^c)$$

5. 安全性評価

著者らが提案した手法による秘密計算は semi-honest 仮定にて安全である. つまり, $View_{P_1}^{GC}(x,y)$, $View_{P_2}^{GC}(x,y)$ に対して以下の式を満たすシミュレータ $S_{P_1}^{GC}, S_{P_2}^{GC}$ が存在する.

$$View_{P_1}^{GC}(x,y) \equiv_c S_{P_1}^{GC}(x, f(x,y)) \quad (1)$$
$$View_{P_2}^{GC}(x,y) \equiv_c S_{P_2}^{GC}(y, f(x,y)) \quad (2)$$

ここでは, 片方のパーティの $View$ が, 秘匿情報を推測する semi-honest 仮定における攻撃者に流出したとしても, 攻撃者は秘匿情報を推測できないことをシミュレーションパラダイムを用いて証明する. また, 秘密計算をする論理関数への P_1 の入力を $x = (x_1 \leftarrow b_1, x_2 \leftarrow b_2, ..., x_n \leftarrow b_n)$, P_2 の入力を $y = (y_1 \leftarrow b_{n+1}, y_2 \leftarrow b_{n+2}, ..., y_n \leftarrow b_{2n}), b_i \in \{0,1\}$ とする.

5.1 P_1 の $View$ が流出した場合

$S_{P_1}^{GC}(x, f(x,y))$ が構成できることで, $View_{P_1}^{GC}(x,y)$ が攻撃者に流出したとしても, P_2 の秘匿情報である y を攻撃者は推測できないことを示す. $View_{P_1}^{GC}(x,y)$ は P_1 の論理関数への入力 x, 論理関数の出力値 $f(x,y)$, P_2 と実行する n 回の $1-out-of-2OT$ プロトコルの $View_{P_1}^{OT}$ で構成される.

$$View_{P_1}^{GC}(x,y)$$
$$:= (x, f(x,y), View_{P_1}^{OT}((K_0^{y_1}, K_1^{y_1}), b_{n+1})$$
$$, ..., View_{P_1}^{OT}((K_0^{y_n}, K_1^{y_n}), b_{2n}))$$

4.6 節にて $S_{P_1}^{OT}$ の存在を保証したので, $S_{P_1}^{GC}(x, f(x,y))$ を以下のように構成する.

$$S_{P_1}^{GC}(x, f(x,y))$$
$$:= (x, f(x,y), S_{P_1}^{OT}(K_0^{y_1}, K_1^{y_1}), ..., S_{P_1}^{OT}(K_0^{y_n}, K_1^{y_n}))$$

$View_{P_1}^{GC}(x,y)$ と $S_{P_1}^{GC}(x,f(x,y))$ が計算量的に識別が困難であることを示す. 任意の多項式時間アルゴリズム D を所持する攻撃者のアドバンテージ $Adv_{D(n)}^{P_1}$ を

$Adv_{D(n)}^{P_1} := |Pr[D(1^n, View_{P_1}^{GC}(x,y)) \rightarrow 1] - Pr[D(1^n, S_{P_1}^{GC}(x,f(x,y))) \rightarrow 1]|$

と定義するこの時

$Adv_{D(n)}^{P_1} < \epsilon(n)$

が成立することをハイブリット論法に基づく背理法で証明する. ハイブリット分布 H_i を

$H_i := (1^n, x, f(x,y), View_{P_1}^{OT}((K_0^{y_1}, K_1^{y_1}), b_{n+1}), ..., View_{P_1}^{OT}((K_0^{y_i}, K_1^{y_i}), b_{n+i})), S_{P_1}^{OT}(K_0^{y_{i+1}}, K_1^{y_{i+1}}), ..., S_{P_1}^{OT}(K_0^{y_n}, K_1^{y_n}))$

と定義する. この時, $Adv_{D(n)}^{P_1}$ は

$Adv_{D(n)}^{P_1} = |Pr[D(H_n) \rightarrow 1] - Pr[D(H_0) \rightarrow 1]|$

となる. 背理法の仮定として, ある D に対して $Adv_{D(n)}^{P_1}$ が無視できないとする. すなわち

$Adv_{D(n)}^{P_1} = |Pr[D(H_n) \rightarrow 1] - Pr[D(H_0) \rightarrow 1]| > \epsilon(n)$

が成り立つと仮定する. 三角不等式により

$\sum_{i=1}^{n} |Pr[D(H_i) \rightarrow 1] - Pr[D(H_{i-1}) \rightarrow 1]| > \epsilon(n)$

よってある $p(1 \leq p \leq n)$ に対して以下の式が成り立つ

$$|Pr[D(H_p) \rightarrow 1] - Pr[D(H_{p-1}) \rightarrow 1]| > \frac{\epsilon(n)}{n}$$

このとき D を使って, 計算量的識別不可能であると保証している $View_{P_1}^{OT}$ と $S_{P_1}^{OT}$ を識別する D' を構成できることを示す.

D' の構成法:

D' は入力 α を受け取り
$(\alpha \xleftarrow{R} View_{P_1}^{OT}((K_0^{y_p}, K_1^{y_p}), b_{n+p})$ または $\alpha \xleftarrow{R} S_{P_1}^{OT}(K_0^{y_p}, K_1^{y_p}))$
ハイブリット分布 $\bar{h}$ を生成する.
$\bar{h} = (1^n, x, f(x,y), View_{P_1}^{OT}((K_0^{y_1}, K_1^{y_1}), b_{n+1}), ..., View_{P_1}^{OT}((K_0^{y_{p-1}}, K_1^{y_{p-1}}), b_{n+p-1}), \alpha, S_{P_1}^{OT}(K_0^{y_{p+1}}, K_1^{y_{p+1}})), ..., S_{P_1}^{OT}(K_0^{y_n}, K_1^{y_n}))$
その後 $\bar{h}$ を引数として D を呼び出す. D' の出力は D の出力とする.

D' が $View_{P_1}^{OT}$ と $S_{P_1}^{OT}$ を識別できていることを確認する. ここで, D' の構成法より

$Pr[D'(\alpha) \rightarrow 1 | \alpha \xleftarrow{R} View_{P_1}^{OT}((K_0^{y_p}, K_1^{y_p}), b_{n+p})]$
$= Pr[D(H_p) \rightarrow 1]$
$Pr[D'(\alpha) \rightarrow 1 | \alpha \xleftarrow{R} S_{P_1}^{OT}(K_0^{y_p}, K_1^{y_p})]$
$= Pr[D(H_{p-1}) \rightarrow 1]$

となる. よって

$|Pr[D'(\alpha) \rightarrow 1 | \alpha \xleftarrow{R} View_{P_1}^{OT}((K_0^{y_p}, K_1^{y_p}), b_{n+p})] - Pr[D'(\alpha) \rightarrow 1 | \alpha \xleftarrow{R} S_{P_1}^{OT}(K_0^{y_p}, K_1^{y_p})]|$
$= |Pr[D(H_p) \rightarrow 1] - Pr[D(H_{p-1}) \rightarrow 1]| > \frac{\epsilon(n)}{n}$

となり, D' は $View_{P_1}^{OT}$ と $S_{P_1}^{OT}$ を識別しており矛盾であるため, (1) が成り立つことを証明した.

5.2 P_2 の $View$ が流出した場合

$S_{P_2}^{GC}(y, f(x,y))$ が構成できることで, $View_{P_2}^{GC}(x,y)$ が攻撃者に流出したとしても, P_1 の秘匿情報である x を攻撃者は推測できないことを示す. $View_{P_2}^{GC}(x,y)$ は P_2 の論理関数への入力 y, 論理関数の出力値 $f(x,y)$, P_1 の入力 x に対応する n 個の鍵 $K_{b_1}^{x_1}, K_{b_2}^{x_2}, ..., K_{b_n}^{x_n}$, 根ノードに対応する GCT の鍵と ID である $K_{v_1}||L_{v_1}$, P_1 と実行する n 回の $1-out-of-2OT$ プロトコルの $View_{P_2}^{OT}$, $Active-GCTs, Inactive-GCTs$ で構成される. $Inctive-GCTs$ は攻撃者から見れば一様ランダムであり, 秘匿情報の推測が困難であることは明らかなので, $View_{P_2}^{GC}(x,y)$ を以下のように定義する.

$View_{P_2}^{GC}(x,y) :=$
$(y, f(x,y), K_{b_1}^{x_1}, ..., K_{b_n}^{x_n}, K_{v_1}||L_{v_1}, Active-GCTs,$
$View_{P_2}^{OT}((K_0^{y_1}, K_1^{y_1}), b_{n+1}), ..., View_{P_2}^{OT}((K_0^{y_n}, K_1^{y_n}), b_{2n}))$

最初に, $GCTs$ 以外のメッセージ群から秘匿情報 x を推測することが困難であることを示す. $K_{b_1}^{x_1}, ..., K_{b_n}^{x_n}$ および $K_{v_1}||L_{v_1}$ は, 攻撃者から見て一様ランダムであるため x を推測するのは困難である. また, 4.6 節にて $S_{P_2}^{OT}$ を保証したので, 以下の式が成り立つ

$View_{P_2}^{GC}(x,y) \equiv_c$
$(y, f(x,y), K_0^{x_1}, ..., K_0^{x_n}, K'_{v_1}||L'_{v_1}, Active-GCTs,$
$S_{P_2}^{OT}(b_{n+1}, K_{b_{n+1}}^{y_1}), ..., S_{P_2}^{OT}(b_{2n}, K_{b_{2n}}^{y_n}))$

ここでは, $S_{P_2}^{GC}(y, f(x,y))$ は $x = (0,0,...,0)$ と仮定して $View$ を生成するとしている. また, $K'_{v_1}||L'_{v_1}$ はシミュレータが生成する根ノードに対応する鍵と GCT の ID である

次に, $Active-GCTs$ から秘匿情報を推測することが困難であることを示す. $Active-GCTs, K_v^{act}, K_e^{act}$ を以下のように定義する.

$Active-GCTs := (GCT_{L_{v_1}}, GCT_{L_{v_2}}, ..., GCT_{L_{v_p}})$
$K_v^{act} := (K_{v_1}, K_{v_2}, ..., K_{v_p})$
$K_e^{act} := (K_1^e, K_2^e, ..., K_p^e)$

攻撃者が $Active-GCTs$, K_v^{act} および K_e^{act} を得て, K_v^{inact} を推測するような攻撃を想定する. そのため, $S_{P_2}^{GC}$ が K_v^{act} および K_e^{act} を用いて $Active-GCTs$ と識別ができない $Fake-Active-GCTs$ を構成できるかを考察する. $Active-GCTs$ のうち, 暗号文 1 つで構成されている GCT に関しては $Active-keys$ のみから構成されてい

るため, $S_{P_2}^{GC}$ は $Fake-GCT$ を構成可能である. しかし, 暗号文 2 つで構成されている GCT に関しては, K_e^{inact} も用いて構成されているため, $S_{P_2}^{GC}$ は暗号化オラクル $\mathcal{O}$ を用いて, $Fake-GCT$ を構成する. 1 つの暗号文を持つ $GCT_{L_{v_i}}(1 \leq i \leq p)$ に対する $Fake-GCT_{L_{v_i}}$ を表 4 に, 2 つの暗号文を持つ $GCT_{L_{v_i}}$ に対する $Fake-GCT_{L_{v_i}}$ を表 5 に示す.

表 4 暗号文が 1 つの $Fake-GCT$

L_{v_i}
$E_{K_{v_i}}(E_{K_i^e}(K_{v_{i+1}}\|\|L_{v_{i+1}}\|\|0^\lambda))$

表 5 暗号文が 2 つの $Fake-GCT$

L_{v_i}
$E_{K_{v_i}}(E_{K_i^e}(K_{v_{i+1}}\|\|L_{v_{i+1}}\|\|0^\lambda))$
$E_{K_{v_i}}(E_{\bar{K_i^e}}(K_{v_{i+1}}\|\|L_{v_{i+1}}\|\|0^\lambda))$

直感的には暗号文が 2 つある場合, 入力変数に対応する鍵が実際の入力の 0 と 1 どちらに対応しているとしても, GCT を計算する者が $K_{v_{i+1}}||L_{v_{i+1}}$ を入手できるよう, $Fake-GCT$ を構成する.

$S_{P_2}^{GC}$ が構成する $Fake-active-GCTs$ を以下のように定義する.

$Fake-active-GCTs :=$
$(Fake-GCT_{L_{v_1}}, Fake-GCT_{L_{v_2}}, ..., Fake-GCT_{L_{v_p}})$

$Active-GCTs$ と $Fake-active-GCTs$ が計算量的に識別が困難であることを示す. 任意の多項式時間アルゴリズム D を所持する攻撃者のアドバンテージ $Adv_{D(n)}^{P_2}$ を

$Adv_{D(n)}^{P_2} := |Pr[D(1^n, Active-GCTs) \rightarrow$
$1] - Pr[D(1^n, Fake-active-GCTs) \rightarrow 1]|$

と定義する. この時

$Adv_{D(n)}^{P_2} < \epsilon(n)$

が成立することをハイブリット論法に基づく背理法で証明する.

ハイブリット分布 H_i を

$H_i := (1^n, GCT_{L_{v_1}}, ..., GCT_{L_{v_i}}, Fake-$
$GCT_{L_{v_{i+1}}}, ..., Fake-GCT_{L_{v_p}})$

と定義する. この時, $Adv_{D(n)}^{P_2}$ は

$Adv_{D(n)}^{P_2} = |Pr[D(H_p) \rightarrow 1] - Pr[D(H_0) \rightarrow 1]|$

となる. 背理法の仮定として, ある D に対して $Adv_{D(n)}^{P_2}$ が無視できないとする. すなわち

$Adv_{D(n)}^{P_2} = |Pr[D(H_p) \rightarrow 1] - Pr[D(H_0) \rightarrow 1]| > \epsilon(n)$

が成り立つと仮定する. 三角不等式により

$\sum_{i=1}^{p} |Pr[D(H_i) \rightarrow 1] - Pr[D(H_{i-1}) \rightarrow 1]| > \epsilon(n)$

よってある $r(1 \leq r \leq p)$ に対して以下の式が成り立つ

$|Pr[D(H_r) \rightarrow 1] - Pr[D(H_{r-1}) \rightarrow 1]| > \frac{\epsilon(n)}{p}$

$GCT_{L_{v_r}}$ が 1 つの暗号文のみから構成されている場合, K_v^{inact} に関する情報の推測が困難であることは明らかであるため, ここでは $GCT_{L_{v_r}}$ が 2 つの暗号文で構成されているとする. $GCT_{L_{v_r}}$, および $Fake-GCT_{L_{v_r}}$ は以下のように構成されているとする. また, $K_{v_j} \in K_v^{inact}$ とする.

表 6 $GCT_{L_{v_r}}$

L_{v_r}
$E_{K_{v_r}}(E_{K_r^e}(K_{v_{r+1}}\|\|L_{v_{r+1}}\|\|0^\lambda))$
$E_{K_{v_r}}(E_{\bar{K_r^e}}(K_{v_j}\|\|L_{v_j}\|\|0^\lambda))$

表 7 $Fake-GCT_{L_{v_r}}$

L_{v_r}
$E_{K_{v_r}}(E_{K_r^e}(K_{v_{r+1}}\|\|L_{v_{r+1}}\|\|0^\lambda))$
$E_{K_{v_r}}(E_{\bar{K_r^e}}(K_{v_{r+1}}\|\|L_{v_{r+1}}\|\|0^\lambda))$

この時, D を使って二重暗号文の安全性を破る識別器 D' を構成できることを示す.

D' と暗号化オラクル $\mathcal{O}$ との間で行われる実験 $Exp_{D'(n)}^{double-\sigma}$ を以下のように定義する.

$Exp_{D'(n)}^{double-\sigma}$ の構成法:

$Exp_{D'(n)}^{double-\sigma}$ は入力
$\alpha = (K_{v_r}, K_r^e, L_{v_r}, (K_{v_{r+1}}||L_{v_{r+1}}||0^\lambda, K_{v_j}||L_{v_j}||0^\lambda))$
を受け取り,
$m_0 = K_{v_{r+1}}||L_{v_{r+1}}||0^\lambda$
$m_1 = K_{v_j}||L_{v_j}||0^\lambda$
とし, 暗号化オラクル $\mathcal{O}$ を用いて 2 つの暗号文 (c_0, c_1) を取得する. そして, ハイブリット分布 $\bar{h}$ を生成する.

$\bar{h} = (1^n, GCT_{L_{v_1}}, ..., GCT_{L_{v_{r-1}}}, (L_{v_r}, c_0, c_1), Fake - GCT_{L_{v_{r+1}}}, ..., Fake-GCT_{L_{v_p}})$

この時, $\sigma = 0$ ならば $(L_{v_r}, c_0, c_1) = Fake-GCT_{L_{v_r}}$ が, $\sigma = 1$ ならば $(L_{v_r}, c_0, c_1) = GCT_{L_{v_r}}$ が成り立つことに注意する.
その後, $\bar{h}$ を引数として D を呼び出す. D' は D の出力を σ の推測値 σ' として出力とする.

D' が二重暗号文の安全性を破っていることを確認する. ここで $Exp_{D'(n)}^{double-\sigma}$ の構成法より

$Pr[Exp^{double-0}_{D'(n)}(\alpha) \to 1] = Pr[D(H_{r-1}) \to 1]$
$Pr[Exp^{double-1}_{D'(n)}(\alpha) \to 1] = Pr[D(H_r) \to 1]$

となる. よって

$|Pr[Exp^{double-0}_{D'(n)}(\alpha) \to 1] - Pr[Exp^{double-1}_{D'(n)}(\alpha) \to 1]|$
$= |Pr[D(H_r) \to 1] - Pr[D(H_{r-1}) \to 1]| > \frac{\epsilon(n)}{p}$

となり, D' は σ の値を無視できない確率で推測している. これは二重暗号文の安全性を破っているため, 矛盾が示せた.

以上の議論より, $S^{GC}_{P_2}$ の構成を
$S^{GC}_{P_2}(y, f(x,y)) :=$
$(y, f(x,y), K^{x_1}_0, ..., K^{x_n}_0, K'_{v_1}||L'_{v_1}, S^{OT}_{P_2}(b_{n+1}, K^{y_1}_{b_{n+1}})$
$, ..., S^{OT}_{P_2}(b_{2n}, K^{y_n}_{b_{2n}}), Fake - Active - GCTs)$
とした時, (2) が成り立つことを証明した.

6. まとめ

著者らによる提案法が, semi-honest 仮定で安全であることを証明した. ここでの証明は Lindell らと同様のシミュレーションパラダイム [9] の技法による.

参考文献

[1] A. Yao, "How to generate and exchange secrets," in 27th FOCS, pp.162-167, Oct 1986.
[2] 笹尾 勤, "論理設計 スイッチング回路理論," 近代科学社, 2005.
[3] L. Kruger, S. Jha, E. -J. Goh, and D. Boneh, "Secure function evaluation with ordered binary decision diagrams," in Proceedings of the 13th ACM conference on Computer and communications security CCS' 06, pp.410–420, Alexandria, Nov 2006.
[4] Shin-ichi Minato, "Zero Suppressed BDDs for Set Manipulation in Combinatorial Problems," in DAC, pp.272-277, ACM, July 1993.
[5] 増井 孝之, 森田 光, "ZDD の Garbled Circuit 法," ISEC, 信学技報, vol. 119, no. 475, March 2020.
[6] 湊 真一, "超高速グラフ列挙アルゴリズム<フカシギの数え方>が拓く, 組合わせ問題への新アプローチ," 森北出版, 2015.
[7] S. Even, O. Goldreich and A. Lemple, "A randomized protocol for signing contracts," Comm of the ACM, pp.637-647, June 1985.
[8] O. Goldreich, "Foundation of Cryptography: Volume2 - Basic Application, " Cambridge University Press, 2004.
[9] C. Hazay and Y. Lindell. "Efficient Secure Two-Party Protocols: Techniques and Constructions," Springer, 2010.
[10] D. E. Knuth, "The Art of Computer Programming Vol. 4," fascicle 1, Addison-Wesley, 2009.