ISSN 1342-2812

Research Reports on Mathematical and Computing Sciences

Constructions for Conditional Oblivious/Converge Transfer/Cast

Daisuke Inoue and Keisuke Tanaka

January 2007, C–232

Department of
Mathematical and
Computing Sciences
Tokyo Institute of Technology

SERIES C: Computer Science

Constructions for Conditional Oblivious/Converge Transfer/Cast

Daisuke Inoue and Keisuke Tanaka

Dept. of Mathematical and Computing Sciences
Tokyo Institute of Technology
W8-55, 2-12-1 Ookayama Meguro-ku, Tokyo 152-8552, Japan
{inoue1, keisuke}@is.titech.ac.jp

February 8, 2007

Abstract

In this paper, we introduce a new notion of *conditional converge cast* (CCC), such that we append the conditional property to *converge cast*. Additionally, we generalize the three primitives with conditional property, *conditional oblivious transfer* (COT), *conditional oblivious cast* (COC), and CCC.

CCC is a three-party protocol which involves two senders S_0 and S_1 and a receiver R. S_0 owns a secret x and a message m_0, and S_1 y and m_1. In a CCC protocol for the predicate Q (Q-CCC), S_0 and S_1 send their messages to R in a masked form. R obtains the message depending on the value of $Q(x, y)$, i.e. R obtains m_0 if $Q(x, y) = 0$ and m_1 otherwise. Besides, the secrets x and y cannot be revealed to R or the other sender. We propose a CCC protocol for "equality" predicate with an *additively homomorphic encryption scheme*.

Additionally, we extend 1-out-of-2 COT/COC/CCC to 1-out-of-n COT/COC/CCC. In 1-out-of-2 protocols, a sender or senders send two messages to a receiver or receivers. In 1-out-of-n protocols, a sender or senders send n messages, where $n = 2^l$ for some l. We provide the consecutive definitions and the concrete protocols for 1-out-of-n COT/COC/CCC protocols. We prove that our protocols are secure under the security of 1-out-of-2 protocols.

Keywords: conditional oblivious transfer, conditional oblivious cast, converge cast.

1 Introduction

Oblivious transfer (OT) is an important primitive proposed by Rabin [8], and it is used in many cryptographic protocols. OT involves two parties, the sender and the receiver. The sender sends a bit to the receiver and the receiver obtains it with probability 1/2. As the primitives for three parties with similar property to OT, *oblivious cast* (OC) and *converge cast* (CC) were presented by Fitzi, Garay, Maurer, and Ostrovsky [6]. OC involves one sender and two receivers, and CC two senders and one receiver. In an OC protocol, the sender sends a message and exactly one of the receivers obtains it. In a CC protocol, the senders send their own messages and the receiver obtains one of the massages. As well as in OT, unnecessary information cannot be revealed to other parties in both protocols.

OT was developed to various types, such as 1-out-of-2 OT (OT_2^1) [5], 1-out-of-n OT (OT_n^1) [2], k-out-of-n OT (OT_n^k) [7], conditional OT (COT) [4], strong COT (SCOT) [1], conditional OC (COC) [3], 1-out-of-2 COC (COC_2^1) [3], etc. In a Q-COT protocol which is COT with the conditional predicate Q, the sender owns a secret x and a message m, and the receiver owns a secret y such that the receiver obtains m from the sender if and only if the condition $Q(x, y)$ is evaluated as true. In a Q-SCOT protocol, the sender sends two messages m_0 and m_1, and the receiver obtains

$m_{Q(x,y)}$. SCOT has 1-out-of-2 property, and suffices our security notion as COT_2^1. COC and COC_2^1 are constructed similarly to COT and SCOT, but the two secrets x and y are prepared to two receivers, respectively.

In this paper, we introduce liberally two notions, *conditional converge cast* (CCC) and 1-out-of-n COT/COC/CCC (COT_n^1/COC_n^1/CCC_n^1). CCC is the protocol such that we append the conditional property to CC for generalization. CC involves two senders S_0 and S_1 and a receiver R, where S_0 and S_1 own their messages m_0 and m_1, respectively. R obtains exactly one of the messages with probability 1/2 after running the protocol without having the other message revealed. S_0 obtains no information about S_1's message, and vice versa. S_0 and S_1 also obtain no information which message is received. We append the conditional property to CC by the predicate Q. In a Q-CCC protocol, S_0 and S_1 have their secrets x and y, respectively, and R obtains $m_{Q(x,y)}$ after running the protocol. R still obtains no information about the other message, and S_0 obtains no information about S_1's message, and vice versa. S_0 and S_1 also obtain no information which message is received. Additionally, we introduce the new security that the sender's secret cannot be revealed to the other sender or the receiver. This notion implies the receiver's security, since if one of the senders obtains any idea of $Q(x, y)$ then he has some information about the other's secret. In addition, we introduce new protocols COT_n^1, COC_n^1, and CCC_n^1, which are the generalization of 1-out-of-2 protocols. COT_2^1 and COC_2^1 were presented in the previous works, and CCC_2^1 is provided in this paper, since CCC has 1-out-of-2 property consequently. We construct 1-out-of-n protocols from 1-out-of-2 ones with the technique in [7].

2 Preliminaries

In this section, we provide some necessary terminology and notation. We start with basic notations, then we provide an additively homomorphic encryption scheme.

2.1 Basic Notions and Model

We use standard notations and conventions for writing probabilistic algorithms and experiments. An algorithm is a Turing machine. An *efficient* algorithm is an algorithm running in probabilistic polynomial time. An interactive Turing machine is a probabilistic algorithm with an additional communication tape. A set of interactive Turing machines is an *interactive protocol*. If A is a probabilistic algorithm, then $y \leftarrow A(x_1, x_2, \ldots,)$ is the experiment of obtaining y by running A on inputs $(x_1, x_2, \ldots)$, where the probability space is given by the random coins of algorithm A. Similarly, the notation $t \leftarrow (A(x), B(y))(z)$ denotes the probabilistic experiment of running an interactive protocol (A, B), where x is A's input, y is B's input, z is an input common to A and B, and t is a transcript of the communication between A and B during such an execution. If S is a finite set, then $x \leftarrow S$ is the operation of picking an element uniformly from S. If α is neither an algorithm nor a set, then $x \leftarrow \alpha$ is a simple assignment statement. If A is an interactive Turing machine, then $A \leftarrow x\ (i)$ denotes a communication sending x to A, and $x \leftarrow A\ (i)$ denotes a communication receiving x from A, where (i) denotes the i-th phase of the communication. If Π is an interactive protocol and P is its participant, then $\Pi_P \leftarrow x\ (i)$ denotes running a protocol with x as P's input, and $x \leftarrow \Pi_P\ (i)$ denotes that P obtains x as a result of running a protocol, where (i) denotes the i-th phase of the communication. If $v_1, \ldots, v_n$ are variables, then $\langle v_1, \ldots, v_n \rangle$ denotes the random ordered vector.

By $\Pr[R_1, \ldots, R_n : E]$ we denote the probability of event E, after the execution of probabilistic experiments $R_1, \ldots, R_n$. Let $a \oplus b$ be the string obtained as the bitwise logical xor of strings a and b. Let $a||b$ be the string obtained by concatenating strings a and b. We say a function $f : \mathbb{N} \to \mathbb{R}$ is *negligible in* n if for every positive polynomial p there exists an N, such that for all $n > N$, $f(n) < 1/p(n)$. We say a probability is *overwhelming in* n if it is negligible different from 1. Let $\{X_n\}_{n\in\mathbb{N}}$ and $\{Y_n\}_{n\in\mathbb{N}}$ be distribution ensembles. We say $\{X_n\}_{n\in\mathbb{N}}$ and $\{Y_n\}_{n\in\mathbb{N}}$

are *computationally indistinguishable* if for any polynomial-time probabilistic Turing machine D, $|\Pr_D(X_n) - \Pr_D(Y_n)| < \epsilon(n)$ is negligible in n where $\Pr_D(X_n)$ is the probability that D accepts x chosen according to the distribution X_n. We call $\{X_n\}_{n\in\mathbb{N}}$ and $\{Y_n\}_{n\in\mathbb{N}}$ are *statistically indistinguishable* if $\sum_\alpha |\Pr[X_n = \alpha] - \Pr[Y_n = \alpha]|$ is negligible.

We are working in a setting with two, or three participants, who use randomness in their computation. We denote the *view* of a party P executing a protocol Π with a party $P_1, \ldots, P_n$ on respective inputs x and $x_1, \ldots, x_n$ by $\mathrm{VIEW}^\Pi_P(x, x_1, \ldots, x_n)$. We note that $\mathrm{VIEW}^\Pi_P(x, x_1, \ldots, x_n)$ is a random variable over the random coins of P and $P_1, \ldots, P_{n-1}$. We stress that although our constructions and analysis are presented for a fixed security parameter k, we have in mind their asymptotic notions. Therefore, for example, when talking about a view of a party $\mathrm{VIEW}^\Pi_P(x, y)$, we mean an ensemble $\{\mathrm{VIEW}^\Pi_P(x, y)\}_k$ of views. We denote statistical indistinguishability of ensembles of random variables X and Y by $X \stackrel{s}{\equiv} Y$ and their computational indistinguishability by $X \stackrel{c}{\equiv} Y$.

2.2 Additively Homomorphic Encryption Scheme

Our constructions use a semantically secure *additively homomorphic encryption scheme*. An encryption scheme (G, E, D) is additively homomorphic if for any m_0 and m_1, $D(E(m_0) \otimes E(m_1)) = D(E(m_0 + m_1))$, where $\otimes$ is an operation defined on the image of E and $+$ is on the domain. The Paillier encryption scheme [9] is additively homomorphic as follows:

- $G(1^k) = (p, q, N, \alpha, g)$, where $N = pq$ is a k-bit number, p and q are two large primes, and g is an integer of order $\alpha N \bmod N^2$ for some integer α. Let $pk = (g, N)$, $sk = \mathrm{lcm}(p-1, q-1)$.
- $E(m) = g^m r^N \bmod N^2$, where $m \in \mathbb{Z}_N$, $r \in_R \mathbb{Z}_n$.
- $D(c) = \frac{L(c^{\lambda(N)} \bmod N^2)}{L(g^{\lambda(N)} \bmod N^2)}$, where $L(u) = \frac{u-1}{N}$.

For any $m_0, m_1, pk = (g, N), sk = \mathrm{lcm}(p-1, q-1)$, the operation $E(m_0) \otimes E(m_1)$ is additively homomorphic since

$$\begin{aligned} D(E(m_0) \otimes E(m_1)) &= D((g^{m_0} r_0^N)(g^{m_1} r_1^N)) \\ &= D(g^{m_0+m_1}(r_0 r_1)^N)) \\ &= D(E(m_0 + m_1)) \end{aligned}$$

We can compute $E(cm)$ from $E(m)$ via $O(\log c)$ repeated additions for a constant c, since we can compute $E(2m)$ easily. For example, we can compute $E(19m)$ by calculating $E((2^{[\log 19]} + 3)m) = E((2^4 + 3)m) = E(2m)^4 \otimes E(3m)$. For ease and clarity, we use $+$ and $-$ as operations on the image of E corresponding to operations on the domain. Note that the Paillier encryption scheme is *semantically secure* [9].

3 Definition

In this section, we provide formal definitions for a CCC protocol, and 1-out-of-n COT/COC/CCC protocols which are the natural extensions of 1-out-of-2 COT/COC/CCC protocols, respectively.

3.1 Conditional Converge Cast

Informally speaking, Q-CCC is a three party protocol with two senders S_0, S_1 who have messages m_0, m_1 and secrets x, y, respectively, and one receiver R. Q-CCC has two following properties:

- Correctness: R obtains m_1 from S_1 if $Q(x, y) = 1$, and m_0 otherwise.

– Sender's security: R obtains exactly one message from either S_0 or S_1. After running the protocol, x is kept secret from S_1 and R, and y is kept secret from S_0 and R.

The definition for Q-CCC$_2^1$ is as follows.

Definition 3.1 (Q-CCC). *Let k be the security parameter. Let S_0, S_1 and R be all polynomial-time probabilistic Turing machines (PPTMs), and Q the predicate computable in polynomial time. Let m_0 and x be the message and the secret of S_0, and m_1 and y those of S_1. Let $\langle S_0, S_1, R\rangle(\cdot)$ be the communication transcript. We say that a three-party interactive protocol $\Pi = (S_0, S_1, R)$ is a secure Q-CCC protocol if it satisfies the following requirements:*

1. *Correctness:*
   (a) *For any x, y, m_0, m_1 from appropriate domains with $Q(x, y) = 0$, the following probability is overwhelming in k:*
      – $\Pr\ [\ tr \leftarrow \langle S_0(m_0, x), S_1(m_1, x), R()\rangle(1^k) :\ R(1^k, tr) = m_0\ ]$
   (b) *For any x, y, m_0, m_1 from appropriate domains with $Q(x, y) = 1$, the following probability is overwhelming in k:*
      – $\Pr\ [\ tr \leftarrow \langle S_0(m_0, x), S_1(m_1, x), R()\rangle(1^k) :\ R(1^k, tr) = m_1\ ]$
2. *Sender's security:*
   (a) *(R obtains essentially no information other than the transferred message.) There exists a simulator Sim_R, such that for any x, y, m_0, m_1 from appropriate domains,*
      – *if $Q(x, y) = 0$ then* $\{\mathrm{Sim}_R(m_0, \bot, \bot)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_R^{\Pi}((m_0, x), (m_1, y), \bot)\}_k$
      – *if $Q(x, y) = 1$ then* $\{\mathrm{Sim}_R(\bot, m_1, \bot)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_R^{\Pi}((m_0, x), (m_1, y), \bot)\}_k$
   (b) *(S_0 and S_1 obtain no efficiently computable information about other's input.) There exists simulators $\mathrm{Sim}_{S_0}, \mathrm{Sim}_{S_1}$, such that for any x, y, m_0, m_1 from appropriate domains,*
      – $\{\mathrm{Sim}_{S_0}((m_0, x), \bot, \bot)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S_0}^{\Pi}((m_0, x), (m_1, y), \bot)\}_k$
      – $\{\mathrm{Sim}_{S_1}(\bot, (m_1, y), \bot)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S_1}^{\Pi}((m_0, x), (m_1, y), \bot)\}_k$

3.2 1-out-of-n COT/COC/CCC

We define COT$_n^1$ as the natural extension of COT$_2^1$. A sender sends n messages to a receiver and the receiver obtains the message depending on the result of the predicate with the sender's secret and the receiver's one. We consider a message index as a l-bit string or $n = 2^l$. COT$_n^1$ has l predicates, and the sender and the receiver have l secrets, respectively. It is the same in COC and CCC. We can obtain k-out-of-n COT/COC/CCC protocols from 1-out-of-n ones trivially by running the protocol k times.

Definition 3.2 (Q-COT$_n^1$). *Let k be the security parameter. Let S and R be all polynomial-time probabilistic Turing machines (PPTMs) and $Q = (Q_1, \ldots, Q_l)$ the predicates ($n = 2^l$). Let $m = (m_1, \ldots, m_n)$ and $x = (x_1, \ldots, x_l)$ be the messages and the secrets of S, and $y = (y_1, \ldots, y_l)$ the secrets of R. Let $\langle S, R\rangle(\cdot)$ be the communication transcript. We say that a two-party interactive protocol $\Pi = (S, R)$ is a secure Q-COT$_n^1$ protocol if it satisfies the following requirements:*

1. *Correctness:*

   For any m, x, y from appropriate domains with l-bit string $i = Q_1(x_1, y_1) \cdots Q_l(x_l, y_l)$, the following probability is overwhelming in k:
   – $\Pr\ [\ tr \leftarrow \langle S(m, x), R(y)\rangle(1^k) : R(y, 1^k, tr) = m_i\ ]$

2. *Sender's security:*

   (R obtains essentially no information other than the transferred message.) There exists a simulator Sim_R*, such that for any* m, x, y *with* l*-bit string* $i = Q_1(x_1, y_1) \cdots Q_l(x_l, y_l)$ *from appropriate domains,*

   - $\{\mathrm{Sim}_R(m_i, y)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_R^{\Pi}((m, x), y)\}_k$

3. *Receiver's security:*

   (S obtains no efficiently computable information about y*.) There exists a simulator* Sim_S*, such that for any* m, x, y *from appropriate domains,*

   - $\{\mathrm{Sim}_S((m, x), \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_S^{\Pi}((m, x), y)\}_k$

Definition 3.3 (Q-COC$_n^1$). *Let* k *be the security parameter. Let* S, R_0 *and* R_1 *be all polynomial-time probabilistic Turing machines (PPTMs), and* $Q = (Q_1, \ldots, Q_l)$ *the predicates* ($n = 2^l$)*. Let* $m = (m_1, \ldots, m_n)$ *be the messages. Let* $x = (x_1, \ldots, x_l)$ *and* $y = (y_1, \ldots, y_l)$ *be the secrets of* R_0 *and* R_1*, respectively. Let* $\langle S, R_0, R_1 \rangle(\cdot)$ *be the communication transcript. We say that a three-party interactive protocol* $\Pi = (S, R_0, R_1)$ *is a secure* Q-COC_n^1 *protocol if it satisfies the following requirements:*

1. *Correctness:*

   For any m, x, y *from appropriate domains with* l*-bit string* $i = Q_1(x_1, y_1) \cdots Q_l(x_l, y_l)$*, the following probability is overwhelming in* k*:*

   - $\Pr\ [\ tr \leftarrow \langle S(m), R_0(x), R_1(y) \rangle(1^k) : R_0(x, 1^k, tr) = R_1(y, 1^k, tr) = m_i\ ]$

2. *Sender's security:*

   (R_0 *and* R_1 *obtain essentially no information other than the transferred message.) There exist simulators* Sim_{R_j}*, such that for any* m, x, y *with* l*-bit string* $i = Q_1(x_1, y_1) \cdots Q_l(x_l, y_l)$ *from appropriate domains,*

   - $\{\mathrm{Sim}_{R_0}(m_i, x, \perp)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_{R_0}^{\Pi}(m, x, y)\}_k$
   - $\{\mathrm{Sim}_{R_1}(m_i, \perp, y)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_{R_1}^{\Pi}(m, x, y)\}_k$

3. *Receiver's security:*

   (a) *(S obtains no efficiently computable information about* x *and* y*.) There exists a simulator* Sim_S*, such that for any* m, x, y *from appropriate domains,*

   - $\{\mathrm{Sim}_S(m, \perp, \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_S^{\Pi}(m, x, y)\}_k$

   (b) *(*R_0 *and* R_1 *obtains no efficiently computable information about the other's secret.) There exist simulators* Sim_{R_0} *and* Sim_{R_1}*, such that for any* m, x, y *from appropriate domains,*

   - $\{\mathrm{Sim}_{R_0}(m, x, \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{R_0}^{\Pi}(m, x, y)\}_k$
   - $\{\mathrm{Sim}_{R_0}(m, x, \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{R_0}^{\Pi}(m, x, y)\}_k$

Definition 3.4 (Q-CCC$_n^1$). *Let* k *be the security parameter. Let* S_0, S_1 *and* R *be all polynomial-time probabilistic Turing machines (PPTMs), and* $Q = (Q_1, \ldots, Q_l)$ *the predicates* ($n = 2^l$)*. Let* $m = (m_1, \ldots, m_{n/2}), m' = (m_{n/2+1}, \ldots, m_n), x = (x_1, \ldots, x_l)$ *and* $y = (y_1, \ldots, y_l)$ *be the messages and the secrets of* S_0 *and* S_1*, respectively. Let* $\langle S_0, S_1, R \rangle(\cdot)$ *be the communication transcript. We say that a three-party interactive protocol* $\Pi = (S_0, S_1, R)$ *is a secure* Q-CCC_n^1 *protocol if it satisfies the following requirements:*

1. *Correctness:*

   For any $m, , m', x, y$ from appropriate domains with $i = Q_1(x_1, y_1) \cdots Q_l(x_l, y_l)$, the following probability is overwhelming in k:

   - $\Pr\ [\ tr \leftarrow \langle S_0(m, x), S_1(m', y), R()\rangle(1^k) : R(1^k, tr) = m_i\ ]$

2. *Sender's security:*

   (a) *(R obtains essentially no information other than the transferred message.) There exists a simulator Sim_R, such that for any m, x, y with l-bit string $i = Q_1(x_1, y_1) \cdots Q_l(x_l, y_l)$ from appropriate domains,*

   - $\{\mathrm{Sim}_R(m_i)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_R^{\Pi}((m, x), (m', y), \perp)\}_k$

   (b) *(S_0 and S_1 obtain no efficiently computable information about the other's secret.) There exist simulators $\mathrm{Sim}_{S_0}, \mathrm{Sim}_{S_1}$, such that for any m, x, y from appropriate domains,*

   - $\{\mathrm{Sim}_{S_0}((m, x), \perp, \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S_0}^{\Pi}((m, x), (m', y), \perp\}_k$
   - $\{\mathrm{Sim}_{S_1}(\perp, (m', y), \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S_1}^{\Pi}((m, x), (m', y), \perp\}_k$

4 Constructions

We provide a CCC protocol for "equality" and 1-out-of-n setting COT/COC/CCC protocols. In order to compute the predicate, we use the Paillier encryption scheme [9] as an additively homomorphic encryption scheme. We use 1-out-fo-2 setting COT/COC/CCC protocols for construction of 1-out-of-n COT/COC/CCC ones.

4.1 1-out-of-2 EQ-CCC

In order to compare x and y, we use an additively homomorphic encryption scheme. We mask m_1 with $r(x - y)$ via an additively homomorphic encryption, where r is a random number. We prepare a flag per bit whose value depends on the bitwise comparison of x with y. The value is a random number if the result of the comparison is "equal", and 0 otherwise. We compute the messages up to the number of bit-length of x or y, where each message generated by masking of m_0 with such a flag.

4.1.1 Construction

Let M be the message space of the Paillier encryption scheme (G, E, D), i.e. $M = \mathbb{Z}_{N=pq}$ where $p < q$ for ease and clarity. Let M' be the message space which suffices the following. For any $m \in M'$, $m||0^k$ is the element of M. Let (m_0, x) be the message and the secret of S_0, and (m_1, y) those of S_1, where $m_0, m_1 \in M'$, $|x| = |y| = n$ and n is smaller than the bit length of p. x_i and y_i denote the i-th bit of x and y. We construct a EQ-CCC$_2^1$ protocol $\Pi = (S_0, S_1, R)$ as follows:

$\mathtt{Algorithm}\ S_0(m_0, x, 1^k)$
  $pk, M \leftarrow R \qquad (0)$
  $(M_1, Y_1, \ldots, Y_n) \leftarrow S_1 \qquad (1)$
  $C_{eq} \leftarrow M_1$
  $\mathtt{for}(i = 1, i \leq n, i++)\{$
    $D_i \leftarrow E_{pk}(x_i) - Y_i, D_i' \leftarrow E_{pk}(x_i) + Y_i - E_{pk}(1)$
    $E_0 \leftarrow E_{pk}(0), E_i \leftarrow 2E_{i-1} + D_i$

$r_i \leftarrow M, r'_i \leftarrow M$
$C_{eq} \leftarrow C_{eq} + r_i D_i$
$C_i \leftarrow E_{pk}(m_0||0^k) + r'_i(E_i - D_i + D'_i)$
}
$R \leftarrow (C_{eq}, \langle C_1, \ldots, C_n \rangle)$ (2)
return $\perp$

Algorithm $S_1(m_1, y, 1^k)$
$pk, M \leftarrow R$ (0)
$S_0 \leftarrow (E_{pk}(m_1||0^k), E_{pk}(y_1), \ldots, E_{pk}(y_n))$ (1)
return $\perp$

Algorithm $R(1^k)$
$pk, sk \leftarrow G(1^{2k})$
$S_0, S_1 \leftarrow pk, M$ (0)
$(C_0, C_1, \ldots, C_n) \leftarrow S_0$ (2)
for$(i = 0, i \leq n, i++)\{$
$a_i||b_i \leftarrow D_{sk}(C_i)$ (b_i is k bit)
if $b_i = 0^k$ then return a_i
}
return $\perp$

In the algorithm S_0 we calculate following variables via additively homomorphic encryption.

$$
\begin{aligned}
&D_{sk}(D_i) := d_i = x_i - y_i \\
&D_{sk}(D'_i) := d'_i = x_i + y_i - 1 \\
&D_{sk}(E_i) := e_i = 2e_{i-1} + d_i \text{ where } e_0 = 0 \\
&D_{sk}(C_{eq}) := c_{eq} = m_1||0^k + \sum_{i=1}^{n} r_i d_i \\
&D_{sk}(C_i) := c_i = m_0||0^k + r'_i(e_i - d_i + d'_i)
\end{aligned}
$$

If $x_i = y_i$, $d_i = 0$ and $d'_i = \pm 1$; otherwise, $d_i = \pm 1$ and $d'_i = 0$. Let l be the rightmost different bit between x and y. We have $e_i = 0$ if $i < l$, $0 < |e_i| < p$ if $i > l$, and $e_i = d_i$ if $i = l$.

4.1.2 Security proof

The interactive protocol $\Pi = (S_0, S_1, R)$ is a secure CCC protocol against the semi-honest (honest-but-curious) senders and the malicious receiver, assuming semantic security of the employed encryption scheme.

Correctness

(a) Assume that $Q(x, y) = 0$: Let l be the index of the first different bit of x and y. In the algorithm S_0, we see that $d_l = e_l$ and $d'_l = 0$, hence $c_l = m_0||0^k$. R verifies ciphertexts from a

younger index to an elder one, hence R returns m_0 when "if $a_i \neq m_0$ then $b_i \neq 0$" holds for any i which is smaller than the index of the correct ciphertext $E_{pk}(c_l)$.

First, we consider the value of C_{eq}. In the algorithm S_0, r_i is uniformly picked from M and $d_i = x_i - y_i = \pm 1$, and thus c_{eq} is also uniformly distributed on M. Therefore, in the algorithm R, the probability that "$b_0 = 0$ and $a_0 \neq m_0$" is $2^{-k}(1 - 2^{-k})$. Next, we consider the values of C_i $(1 \leq i \leq n)$. In the algorithm S_0, r'_i is uniformly picked from M. Because $|x|$ and $|y|$ is smaller than the bit length of p, $GCD(e_i - d_i + d'_i, N) = 1$, hence $r'_i(e_i - d_i + d'_i)$ is uniformly distributed on M. Therefore, the probability that "$b_i = 0$ and $a_i \neq m_0$" is $2^{-k}(1 - 2^{-k})$. The worst case is that the last element of $\langle C_1, \ldots, C_n \rangle$ is $E_{pk}(c_l)$. From the above discussion, we have

$$\begin{aligned} \Pr\ [\ tr \leftarrow \langle S_0(m_0, x), S_1(m_1, x), R() \rangle (1^k) : R(1^k, tr) = m_0\ ] \\ > (1 - 2^{-k}(1 - 2^{-k}))^n > 1 - \epsilon(k) \end{aligned}$$

(b) Assume that $Q(x, y) = 1$: In the algorithm S_0, since $d_i = 0$ for any i $(1 \leq i \leq n)$, we have $c_{eq} = m_1 || 0^k$. Therefore, R returns m_1 with probability 1.

Sender's security

(a) Security against the receiver: The view of R is $\mathrm{VIEW}_R^{\Pi}((m_0, x), (m_1, y), \perp) = (C_0, C_1, \ldots, C_n)$.

Assume that $Q(x, y) = 0$. As we showed above, one element of $(C_1, \ldots, C_n)$ is $E_{pk}(m_0 || 0^k)$, and others are all uniformly distributed on M. Therefore, we can construct the simulator as follows:

$$\begin{array}{l} \texttt{Algorithm}\ \mathrm{Sim}_R(m_0) \\ \quad \texttt{for}(i = 1, i \leq n, i++)\{ \\ \quad\quad r_i \leftarrow M \\ \quad \} \\ \quad \texttt{return}\ (E_{pk}(r_1), \langle E_{pk}(m_0 || 0^k), E_{pk}(r_2), \ldots, E_{pk}(r_n) \rangle) \end{array}$$

The output of $\mathrm{Sim}_R(m_0)$ is statistically indistinguishable from the view of R.

Assume that $Q(x, y) = 1$. As we showed above, c_i $(1 \leq i \leq n)$ are all uniformly distributed on M. Therefore, we can construct the simulator as follows,

$$\begin{array}{l} \texttt{Algorithm}\ \mathrm{Sim}_R(m_1) \\ \quad \texttt{for}(i = 1, i \leq n, i++)\{ \\ \quad\quad r_i \leftarrow M \\ \quad \} \\ \quad \texttt{return}\ (E_{pk}(m_1), \langle E_{pk}(r_1), \ldots, E_{pk}(r_n) \rangle) \end{array}$$

The output of $\mathrm{Sim}_R(m_0)$ is statistically indistinguishable from the view of R.

(b) Security against the sender: The view of S_0 is $(m_0, x, M_1, Y_1, \ldots, Y_n, r_1, \ldots, r_n, r'_1, \ldots, r'_n)$. The simulator $\mathrm{Sim}_{S_0}((x, m_0), \perp, \perp)$ has m_0 and x as the input, and r_i and r'_i is uniformly distributed on M. For $M_1, Y_1, \ldots, Y_n$ we construct the simulator as follows,

$$\begin{array}{l} \texttt{Algorithm}\ \mathrm{Sim}_{S_0}((x, m_0), \perp, \perp) \\ \quad r \leftarrow M \\ \quad M'_1 \leftarrow E_{pk}(r) \\ \quad \texttt{for}(i = 1, i \leq n, i++)\{ \end{array}$$

$a_i \leftarrow M, b_i \leftarrow M, c_i \leftarrow M$
$Y'_i \leftarrow E_{pk}(a_i)$
}
return $(m_0, x, M'_1, Y'_1, \ldots, Y'_n, b_1, \ldots, b_n, c_1, \ldots, c_n)$

b_i, c_i and r_i, r'_i are all uniformly distributed on M. It follows directly that there is no efficient distinguisher between $M_1, Y_1, \ldots, Y_n$ and $M'_1, Y'_1, \ldots, Y'_n$ from the semantic security of the employed encryption scheme.

4.2 1-out-of-n Q-COT

Our construction of a COT_n^1 protocol uses the secure COT_2^1 one as a special case of COT_n^1 one. For example, a SCOT protocol [1] suffices our security notions as COT_2^1.

4.2.1 Construction

Let $Q_1, \ldots, Q_l$ be the predicates and $Q = (Q_1, \ldots, Q_l)$. Let $Q_i\text{-}\mathrm{COT}_2^1 = (S^i, R^i)$ be a secure COT_2^1 protocol with the security parameter k. We construct a $Q\text{-}\mathrm{COT}_{2^l}^1$ protocol with $Q_1\text{-}\mathrm{COT}_2^1$,...,$Q_l\text{-}\mathrm{COT}_2^1$. Let M be the message space of COT_2^1, and M' the message space which suffices the following. For any $K \in M'$, $0||K$ and $1||K$ is the element of M. Let $m = (m_1, \ldots, m_n)$ be the messages from M, and $x = (x_1, \ldots, x_l)$ and $y = (y_1, \ldots, y_l)$ the secrets of S and R, respectively, from the domain of the secrets of $Q_i\text{-}\mathrm{COT}_2^1$. We construct a $Q\text{-}\mathrm{COT}_n^1$ protocol $\Pi = (S, R)$ as follows:

Algorithm $S(m, x, 1^k)$
for$(i = 1, i \leq l, i++)\{$
$K_i^0 \leftarrow M', K_i^1 \leftarrow M'$
$Q_i\text{-}\mathrm{COT}_{2S^i}^1 \leftarrow (0||K_i^0, 1||K_i^1, x_i)$ (1)
}
for$(i = 1, i \leq n, i++)\{$
$c_i \leftarrow m_i \oplus \bigoplus_{j=1}^{l} K_j^{i_j}$ where i_j denotes $j-$th bit of i
}
$R \leftarrow (c_1, \ldots, c_n)$ (2)
return $\perp$

Algorithm $R(y, 1^k)$
for$(i = 1, i \leq l, i++)\{$
$Q_i\text{-}\mathrm{COT}_{2R^i}^1 \leftarrow y_i$ (1)
$k_i \leftarrow Q_i\text{-}\mathrm{COT}_{2R^i}^1$ (1)
$I_i||K_i^{I_i} \leftarrow k_i$ (I_i is 1 bit)
}
$(c_1, \ldots, c_n) \leftarrow S$ (2)
return $c_I \oplus \bigoplus_{j=1}^{l} K_j^{I_j}$ where I_j denotes $j-$th bit of I

The complexity of the whole protocol is $\log n$ invocations of the COT_2^1 protocol.

4.2.2 Security proof

The interactive protocol Π is a secure COT_n^1 protocol against the semi-honest (honest-but-curious) sender and the malicious receiver.

Correctness Let ν_i be the success probability of Q_i-COT_2^1. R obtains the correct message if and only if all Q_i-COT_2^1 is successful.

$$\Pr\left[\ tr \leftarrow \langle S(x,m), R(y)\rangle(1^k) : R(y, 1^k, tr) = m_i\ \right] = \prod_{i=1}^{n} \nu_i > 1 - \epsilon(k)$$

Sender's security We denote the view of R by (y, c, a) where $y = (y_1, \ldots, y_n)$, $c = (c_1, \ldots, c_n)$, and $a = (a_1, \ldots, a_l)$ are the views of R^i. Because of the sender's security of Q_i-COT_2^1, for all i $(1 \le i \le l)$ there exists a simulator which simulates the view of R^i, i.e.

$$\{\mathrm{Sim}_{R^i}(K_i^{Q_i(x_i,y_i)}, y_i)\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_{R^i}^{Q_i\text{-}\mathrm{COT}_2^1}((K_i^0, K_i^1, x_i), y_i)\}_k$$

We construct the simulator $\mathrm{Sim}_R(m_{Q(x,y)}, y)$ as follows:

$$\begin{array}{l}
\texttt{Algorithm}\ \mathrm{Sim}_R(m_{Q(x,y)}, y) \\
\quad \texttt{for}(i = 1, i \le l, i{+}{+})\{ \\
\qquad r_i \leftarrow M, c_i' \leftarrow M \\
\qquad a_i' \leftarrow \mathrm{Sim}_{R^i}(r_i, y_i) \\
\quad \} \\
\quad c' \leftarrow (c_1', \ldots, c_n') \\
\quad a' \leftarrow (a_1', \ldots, a_n') \\
\quad \texttt{return}\ (y, c', a')
\end{array}$$

We show that there is no efficient distinguisher between (y, c, a) and (y, c', a'). The elements of c and c' are all uniformly distributed on M, hence there are statistically indistinguishable. a and a' are statistically indistinguishable because of the sender's security of Q_i-COT_2^1.

Receiver's security We denote the view of S by (m, x, K, a) where $m = (m_1, \ldots, m_n)$, $x = (x_1, \ldots, x_l)$, $K = ((K_1^0, K_1^1), \ldots, (K_l^0, K_l^1))$, and $a = (a_1, \ldots, a_l)$ are the views of S^i. Because of the receiver's security of Q_i-COT_2^1, for all i $(1 \le i \le l)$ there exists a simulator which simulates the view of S^i, i.e.

$$\{\mathrm{Sim}_{S^i}((K_i^0, K_i^1, x_i), \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S^i}^{Q_i\text{-}\mathrm{COT}_2^1}((K_i^0, K_i^1, x_i), y_i)\}_k$$

We construct the simulator $\mathrm{Sim}_S((m, x), \perp)$ as follows,

$$\begin{array}{l}
\texttt{Algorithm}\ \mathrm{Sim}_S((m, x), \perp) \\
\quad \texttt{for}(i = 1, i \le l, i{+}{+})\{ \\
\qquad r_i^0 \leftarrow M, r_i^1 \leftarrow M \\
\qquad a_i' \leftarrow \mathrm{Sim}_{S^i}((r_i^0, r_i^1, x_i), \perp) \\
\quad \} \\
\quad K' \leftarrow (r_1^0, r_1^1), \ldots, (r_l^0, r_l^1) \\
\quad a' \leftarrow a_1', \ldots, a_l' \\
\quad \texttt{return}\ (m, x, K', a')
\end{array}$$

The elements of K and K' are all uniformly distributed on M, hence they are computationally indistinguishable. Suppose that there exists a efficient distinguisher between (m, x, K, a) and (m, x, K', a'), we can construct the adversary who breaks the receiver's security of Q_i-COT_2^1. Let A, B be PPTMs, where A attacks the receiver's security of Q_i-COT_2^1, and B distinguishes a from a' in probability ν and in efficient time t. The construction is as follows,

$$
\begin{aligned}
&\texttt{Algorithm } \mathrm{A}(x)\\
&\quad r_1, \ldots, r_n - 1 \leftarrow M.\\
&\quad b \leftarrow B(\langle x, r_1, \ldots, r_{n-1}\rangle)\\
&\quad \texttt{return } b
\end{aligned}
$$

B distinguishes the tuple of n distributions with probability ν, thus A distinguishes x with probability at least ν/n in time t. If ν is not negligible then it contradicts the receiver's security of Q_i-COT_2^1.

4.3 1-out-of-n Q-COC

Our construction of a COC_n^1 protocol uses the secure COC_2^1 one as a special case of COC_n^1 one. For example, a COC_2^1 protocol [3] suffices our security notions.

4.3.1 Construction

Let $Q_1, \ldots, Q_l$ be the predicates, and $Q = (Q_1, \ldots, Q_l)$. Let Q_i-$\mathrm{COC}_2^1 = (S^i, R_0^i, R_1^i)$ be a secure COC_2^1 protocol with the security parameter k. We construct a Q-$\mathrm{COC}_{2^l}^1$ protocol, with Q_1-COC_2^1,...,Q_l-COC_2^1. Let M be the message space of COC_2^1, and M' the message space which suffices the following. For any $K \in M'$, $0||K$ and $1||K$ is the element of M. Let $m = (m_1, \ldots, m_n)$ be the messages from M, and $x = (x_1, \ldots, x_l)$ and $y = (y_1, \ldots, y_l)$ the secrets of R_0 and R_1, respectively, from the domain of the secrets of Q_i-COC_2^1. We construct a Q-COC_n^1 protocol $\Pi = (S, R_0, R_1)$ as follows:

$$
\begin{aligned}
&\texttt{Algorithm } S(m, 1^k)\\
&\quad \texttt{for}(i = 1, i \leq l, i++)\{\\
&\qquad K_i^0 \leftarrow M', K_i^1 \leftarrow M'\\
&\qquad Q_i\text{-}\mathrm{COC}_{2S^i}^1 \leftarrow (0||K_i^0, 1||K_i^1, x_i) \qquad (1)\\
&\quad \}\\
&\quad \texttt{for}(i = 1, i \leq n, i++)\{\\
&\qquad c_i \leftarrow m_i \oplus \bigoplus_{j=1}^{l} K_j^{i_j} \text{ where } i_j \text{ denotes } j\text{−th bit of } i\\
&\quad \}\\
&\quad R_0, R_1 \leftarrow (c_1, \ldots, c_n) \qquad (2)\\
&\quad \texttt{return } \perp
\end{aligned}
$$

$$
\begin{aligned}
&\texttt{Algorithm } R_0(x, 1^k)\\
&\quad \texttt{for}(i = 1, i \leq l, i++)\{\\
&\qquad Q_i\text{-}\mathrm{COC}_{2R_0^i}^1 \leftarrow x_i \qquad (1)\\
&\qquad k_i \leftarrow Q_i\text{-}\mathrm{COC}_{2R_0^i}^1 \qquad (1)\\
&\qquad I_i||K_i^{I_i} \leftarrow k_i \ (I_i \text{ is 1 bit})
\end{aligned}
$$

}
$(c_1, \ldots, c_n) \leftarrow S \qquad (2)$
return $c_I \oplus \bigoplus_{j=1}^{l} K_j^{I_j}$ where I_j denotes $j-$th bit of I

Algorithm $R_1(y, 1^k)$
for$(i = 1, i \leq l, i++)\{$
$Q_i\text{-COC}^1_{2R_1^i} \leftarrow y_i \qquad (1)$
$k_i \leftarrow Q_i\text{-COC}^1_{2R_1^i} \qquad (1)$
$I_i || K_i^{I_i} \leftarrow k_i$ (I_i is 1 bit)
}
$(c_1, \ldots, c_n) \leftarrow S \qquad (2)$
return $c_I \oplus \bigoplus_{j=1}^{l} K_j^{I_j}$ where I_j denotes $j-$th bit of I

The complexity of the whole protocol is $\log n$ invocations of the COC protocol$_2^1$.

4.3.2 Security proof

The interactive protocol Π is a secure COT_n^1 protocol against the semi-honest (honest-but-curious) sender and the malicious receivers.

Correctness Let ν_i be the success probability of Q_i-COC_2^1. R_0 and R_1 obtain the correct message if and only if all Q_i-COC_2^1 is successful.

$$\Pr\ [\ tr \leftarrow \langle S(m), R_0(x), R_1(y)\rangle(1^k) : R_0(x, 1^k, tr) = R_1(y, 1^k, tr) = m_i\ ] = \prod_{i=1}^{n} \nu_i > 1 - \epsilon(k)$$

Sender's security We denote the views of R_0 and R_1 by (x, c, a) and (y, c, b), respectively, where $x = (x_1, \ldots, x_n)$, $y = (y_1, \ldots, y_n)$, $c = (c_1, \ldots, c_n)$ and $a = (a_1, \ldots, a_l), b = (b_1, \ldots, b_l)$ are the views of R_0^i, R_1^i, respectively. Because of the sender's security of Q_i-COC_2^1, for all i $(1 \leq i \leq l)$ there exists simulators which simulate the views of R_0^i, R_1^i, respectively, i.e.

$$\{\text{Sim}_{R_0^i}(K_i^{Q_i(x_i,y_i)}, x_i)\}_k \stackrel{s}{\equiv} \{\text{VIEW}_{R_0^i}^{Q_i\text{-COT}_2^1}((K_i^0, K_i^1), x_i, y_i)\}_k$$
$$\{\text{Sim}_{R_1^i}(K_i^{Q_i(x_i,y_i)}, y_i)\}_k \stackrel{s}{\equiv} \{\text{VIEW}_{R_1^i}^{Q_i\text{-COT}_2^1}((K_i^0, K_i^1), x_i, y_i)\}_k$$

We construct the simulator $\text{Sim}_{R_0}(m_{Q(x,y)}, y)$ and $\text{Sim}_{R_0}(m_{Q(x,y)}, y)$ as follows:

Algorithm $\text{Sim}_{R_0}(m_{Q(x,y)}, x)$
for$(i = 1, i \leq l, i++)\{$
$r_i \leftarrow M, c_i' \leftarrow M$
$a_i' \leftarrow \text{Sim}_{R_0^i}(r_i, x_i)$
}
$c' \leftarrow (c_1', \ldots, c_n')$
$a' \leftarrow (a_1', \ldots, a_l')$

$$\texttt{return } (x, c', a')$$

$$\texttt{Algorithm } \mathrm{Sim}_{R_1}(m_{Q(x,y)}, y)$$
$$\texttt{for}(i = 1, i \leq l, i++)\{$$
$$r_i \leftarrow M, c'_i \leftarrow M$$
$$b'_i \leftarrow \mathrm{Sim}_{R_1^i}(r_i, y_i)$$
$$\}$$
$$c' \leftarrow (c'_1, \ldots, c'_n)$$
$$b' \leftarrow (b'_1, \ldots, b'_l)$$
$$\texttt{return } (y, c', b')$$

We just show that there is no efficient distinguisher between (x, c, a) and (x, c', a'), since the same can be said for (y, c, b) and (y, c', b'). The elements of c and c' are all uniformly distributed on M, hence there are statistically indistinguishable. Because of the receiver's security of Q_i-COC_2^1, a and a' are statistically indistinguishable.

Receiver's security

(a) Security against the sender: We denote the view of S by (m, K, a) where $m = (m_1, \ldots, m_n)$, $K = ((K_1^0, K_1^1), \ldots, (K_l^0, K_l^1))$, and $a = (a_1, \ldots, a_l)$ are the views of S^i. Because of the receiver's security of Q_i-COC_2^1 against the sender, for all i $(1 \leq i \leq l)$ there exists a simulator which simulates the view of S^i, i.e.

$$\{\mathrm{Sim}_{S^i}((K_i^0, K_i^1), \perp, \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S^i}^{Q_i\text{-}\mathrm{COT}_2^1}((K_i^0, K_i^1), x_i, y_i)\}_k$$

We construct the simulator $\mathrm{Sim}_S((m, x), \perp)$ as follows:

$$\texttt{Algorithm } \mathrm{Sim}_S(m, \perp, \perp)$$
$$\texttt{for}(i = 1, i \leq l, i++)\{$$
$$r_i^0 \leftarrow M, r_i^1 \leftarrow M$$
$$a'_i \leftarrow \mathrm{Sim}_{S_i}((r_i^0, r_i^1), \perp, \perp)$$
$$\}$$
$$K' \leftarrow (r_1^0, r_1^1), \ldots, (r_l^0, r_l^1)$$
$$a' \leftarrow a'_1, \ldots, a'_l$$
$$\texttt{return } (m, x, K', a')$$

The elements of K and K' are all uniformly distributed on M, hence they are computationally indistinguishable. a and a' are computationally indistinguishable because of the receiver's security of Q_i-COC_2^1. We can show this by exactly the same technique as we showed in the receiver's security of COT_n^1.

(b) Security against the receiver: We can show this by exactly the same process as the receiver's security against the sender.

4.4 1-out-of-n Q-CCC

Our construction of a CCC_n^1 protocol uses the secure CCC one as a special case of CCC_n^1 one, since CCC provides 1-out-of-2 property.

4.4.1 Construction

Let $Q_1, \ldots, Q_l$ be the predicates, and $Q = (Q_1, \ldots, Q_l)$. Let Q_i-CCC$= (S_0^i, S_1^i, R^i)$ be a secure CCC protocol with the security parameter k. We construct a Q-CCC$_{2^l}^1$ protocol, with Q_1-CCC,...,Q_l-CCC. Let M be the message space of CCC, and M' the message space which suffices the following. For any $K \in M'$, $0||K$ and $1||K$ is the element of M. Let $m = (m_1, \ldots, m_{n/2}), m' = (m_{n/2+1}, \ldots, m_n)$,$x = (x_1, \ldots, x_l)$, and $y = (y_1, \ldots, y_l)$ be the messages and the secrets of S_0, S_1, respectively. The messages are from M and the secrets from the domain of the secrets of Q_i-CCC. We construct a Q-CCC$_n^1$ protocol $\Pi = (S_0, S_1, R)$ as follows:

Algorithm $S_0(m, x, 1^k)$
- **for**$(i = 1, i \leq l, i++)\{$
  - $K_i^0 \leftarrow M'$
  - Q_i-CCC$_{S_0^i} \leftarrow (0||K_i^0, x_i)$ (1)
- $\}$
- $S_1 \leftarrow (K_1^0, \ldots, K_l^0)$ (2)
- $(K_1^1, \ldots, K_l^1) \leftarrow S_1$ (2)
- **for**$(i = 1, i \leq n/2, i++)\{$
  - $c_i \leftarrow m_i \oplus \bigoplus_{j=1}^{l} K_j^{i_j}$ where i_j denotes $j-$th bit of i
- $\}$
- $R \leftarrow (c_1, \ldots, c_{n/2})$ (3)
- **return** $\perp$

Algorithm $S_1(m', y, 1^k)$
- **for**$(i = 1, i \leq l, i++)\{$
  - $K_i^1 \leftarrow M'$
  - Q_i-CCC$_{S_1^i} \leftarrow (1||K_i^1, y_i)$ (1)
- $\}$
- $S_0 \leftarrow (K_1^1, \ldots, K_l^1)$ (2)
- $(K_1^0, \ldots, K_l^0) \leftarrow S_0$ (2)
- **for**$(i = n/2 + 1, i \leq n, i++)\{$
  - $c_i \leftarrow m_i \oplus \bigoplus_{j=1}^{l} K_j^{i_j}$ where i_j denotes $j-$th bit of i
- $\}$
- $R \leftarrow (c_n/2 + 1, \ldots, c_n)$ (4)
- **return** $\perp$

Algorithm $R(y, 1^k)$
- **for**$(i = 1, i \leq l, i++)\{$
  - Q_i-CCC$_{R^i} \leftarrow y_i$ (1)
  - $k_i \leftarrow Q_i$-CCC$_{R^i}$ (1)
  - $I_i||K_i^{I_i} \leftarrow k_i$ (I_i is 1 bit)

}
$(c_1, \ldots, c_{n/2}) \leftarrow S_0$ (3)
$(c_{n/2+1}, \ldots, c_n) \leftarrow S_1$ (4)
return $c_I \oplus \bigoplus_{j=1}^{l} K_j^{I_j}$ where I_j denotes $j-$th bit of I

The complexity of the whole protocol is $\log n$ invocations of the 1-out-of-2 CCC protocol.

4.4.2 Security proof

The interactive protocol Π is a secure CCC_n^1 protocol against the semi-honest (honest-but-curious) senders and the malicious receiver.

Correctness Let ν_i be the probability that Q_i-CCC successfully conclude. R obtains the correct message if and only if all the Q_i-CCC protocol is successful.

$$\Pr\ [\ tr \leftarrow \langle S_0(x,m), S_1(y,m'), R()\rangle(1^k) : R(1^k, tr) = m_i\ ] = \prod_{i=1}^{n} \nu_i > 1 - \epsilon(k)$$

Sender's security

(a) Security against the receiver: We denote the view of R by (c, a) where $c = (c_1, \ldots, c_n)$, and $a = (a_1, \ldots, a_l)$ are the views of R^i. Because of the sender's security of Q_i-CCC, for all i $(1 \le i \le l)$ there exists a simulator which simulates the view of R^i, i.e.

$$\{\mathrm{Sim}_{R^i}(K_i^{Q_i(x_i,y_i)})\}_k \stackrel{s}{\equiv} \{\mathrm{VIEW}_{R^i}^{Q_i\text{-}\mathrm{COT}_2^1}((K_i^0, x_i), (K_i^1, y_i), \perp)\}_k$$

We construct the simulator $\mathrm{Sim}_R(m_{Q(x,y)}, y)$ as follows:

Algorithm $\mathrm{Sim}_R(m_{Q(x,y)})$
  for$(i = 1, i \le l, i++)\{$
    $r_i \leftarrow M, c_i' \leftarrow M$
    $a_i' \leftarrow \mathrm{Sim}_{R_i}(r_i)$
  $\}$
  $c' \leftarrow (c_1', \ldots, c_n')$
  $a' \leftarrow (a_1', \ldots, a_n')$
  return (c', a')

We show that there is no efficient distinguisher between (c, a) and (c', a'). The elements of c and c' are all uniformly distributed on M, hence there are statistically indistinguishable. Because of the sender's security of Q_i-CCC_n^1 a and a' are statistically indistinguishable.

(b) Security against the sender: We denote the views of S_0 and S_1 by (m, x, K^0, a) and (m', y, K^1, b), respectively where $m = (m_1, \ldots, m_{n/2})$, $m' = (m_{n/2+1}, \ldots, m_n)$, $x = (x_1, \ldots, x_l)$, $y = (y_1, \ldots, y_l)$, $K^0 = (K_1^0, \ldots, K_l^0)$, $K^1 = (K_1^1, \ldots, K_l^1)$, and $a = (a_1, \ldots, a_l), b = (b_1, \ldots, b_l)$ are the views of S_0^i, S_1^i, respectively. Because of the sender's security of Q_i-CCC, for all i $(1 \le i \le l)$ there exists simulators which simulate the views of S_0^i, S_1^i, respectively, i.e.

$$\{\mathrm{Sim}_{S_0^i}((K_i^0, x_i), \perp, \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S_0^i}^{Q_i\text{-}\mathrm{COT}_2^1}((K_i^0, x_i), (K_i^1, y_i), \perp)\}_k$$
$$\{\mathrm{Sim}_{S_1^i}(\perp, (K_i^1, y_i), \perp)\}_k \stackrel{c}{\equiv} \{\mathrm{VIEW}_{S_1^i}^{Q_i\text{-}\mathrm{COT}_2^1}((K_i^0, x_i), (K_i^1, y_i), \perp)\}_k$$

We construct the simulator $\mathrm{Sim}_{S_0}((m,x),\perp,\perp)$ and $\mathrm{Sim}_{S_1}(\perp,(m',y),\perp)$ as follows,

$$
\begin{array}{l}
\texttt{Algorithm } \mathrm{Sim}_{S_0}((m,x),\perp,\perp) \\
\quad \texttt{for}(i=1, i \leq l, i++)\{ \\
\quad\quad r_i \leftarrow M \\
\quad\quad a'_i \leftarrow \mathrm{Sim}_{S_0^i}((r_i,x_i),\perp,\perp) \\
\quad \} \\
\quad K' \leftarrow r_1,\ldots,r_l \\
\quad a' \leftarrow a'_1,\ldots,a'_l \\
\quad \texttt{return } (m,x,K',a') \\
\\
\texttt{Algorithm } \mathrm{Sim}_{S_1}(\perp,(m',y),\perp) \\
\quad \texttt{for}(i=1, i \leq l, i++)\{ \\
\quad\quad r_i \leftarrow M \\
\quad\quad b'_i \leftarrow \mathrm{Sim}_{S_1^i}(\perp,(r_i,y_i),\perp) \\
\quad \} \\
\quad K'' \leftarrow r_1,\ldots,r_l \\
\quad b' \leftarrow b'_1,\ldots,b'_l \\
\quad \texttt{return } (m,x,K',b')
\end{array}
$$

We just show that there is no efficient distinguisher between (m,x,K^0,a) and (m,x,K',a'), since the same can be said for (m',y,K^1,b) and (m',y,K'',b'). The elements of K^0 and K' are all uniformly distributed on M, hence they are computationally indistinguishable. a and a' are computationally distinguishable because of the receiver's security of Q_i-CCC. We can show this by exactly the same technique as we showed in receiver's security of COT_n^1.

5 Conclusion

We introduce a new notion of *conditional converge cast*, such that we append the conditional property to converge cast, and new notions of 1-out-of-n *conditional oblivious/converge transfer/cast*, which are the generalization of 1-out-of-2 protocols. The definitions of these notions are given. We also provide an implementation for these notions.

References

[1] Blake, I. F., and Kolesnikov, V. Strong conditional oblivious transfer and computing on intervals. In *ASIACRYPT* (2004), pp. 515–529.

[2] Brassard, G., Crépeau, C., and Robert, J.-M. All-or-nothing disclosure of secrets. In *CRYPTO* (1986), pp. 234–238.

[3] Chu, C.-K., and Tzeng, W.-G. Conditional oblivious cast. In *Public Key Cryptography* (2006), pp. 443–457.

[4] Di Crescenzo, G., Ostrovsky, R., and Rajagopalan, S. Conditional oblivious transfer and timed-release encryption. In *EUROCRYPT* (1999), pp. 74–89.

[5] Even, S., Goldreich, O., and Lempel, A. A randomized protocol for signing contracts. *Commun. ACM 28*, 6 (1985), 637–647.

[6] Fitzi, M., Garay, J. A., Maurer, U. M., and Ostrovsky, R. Minimal complete primitives for secure multi-party computation. In *CRYPTO* (2001), pp. 80–100.

[7] Naor, M., and Pinkas, B. Oblivious transfer and polynomial evaluation. In *STOC* (1999), pp. 245–254.

[8] O.Rabin, M. How to exchange secrets by oblivious transfer. Tech. Rep. TR-81, Aiken computation laboratory, Harvard University, 1981.

[9] Paillier, P. Public-key cryptosystems based on composite degree residuosity classes. In *EUROCRYPT* (1999), pp. 223–238.