2017

Efficient k-out-of-n oblivious transfer scheme with the ideal communication cost

Jianchang Lai
University of Wollongong, jl967@uowmail.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

Fuchun Guo
University of Wollongong, fuchun@uow.edu.au

Rongmao Chen
National University of Defense Technology Changsha, rc517@uowmail.edu.au

Sha Ma
South China Agricultural University, sma@uow.edu.au

Publication Details

Lai, J., Mu, Y., Guo, F., Chen, R. & Ma, S. (2018). Efficient k-out-of-n oblivious transfer scheme with the ideal communication cost. Theoretical Computer Science, 714 15-26.

Efficient k-out-of-n oblivious transfer scheme with the ideal communication cost

Abstract
In this paper, we propose a two-round k-out-of-n oblivious transfer scheme with the minimum communication cost. In our proposed scheme, the messages sent by the receiver R to the sender S consist of only three elements, which is independent of n and k, while the messages from S to R are (n+1) elements when the sender holds n secrets. Our scheme features a nice property of universal parameter, where the system parameter can be used by all senders and receivers. The proposed k-out-of-n oblivious transfer scheme is the most efficient two-round scheme in terms of the number of messages transferred between two communicating parties in known constructions. The scheme preserves the privacy of receiver's choice and sender's security.

Disciplines
Engineering | Science and Technology Studies

Publication Details
Lai, J., Mu, Y., Guo, F., Chen, R. & Ma, S. (2018). Efficient k-out-of-n oblivious transfer scheme with the ideal communication cost. Theoretical Computer Science, 714 15-26.

Efficient k-out-of-n Oblivious Transfer Scheme with the Ideal Communication Cost

Jianchang Lai[a,*], Yi Mu[a], Fuchun Guo[a,*], Rongmao Chen[b], Sha Ma[c]

[a] *Institute of Cybersecurity and Cryptology, School of Computing and Information Technology, University of Wollongong, Wollongong, Australia*
[b] *College of Computer, National University of Defense Technology, Changsha, China*
[c] *College of Mathematics and Informatics, South China Agricultural University, Guangzhou, Guangdong, China*

Abstract

In this paper, we propose a two-round k-out-of-n oblivious transfer scheme with the minimum communication cost. In our proposed scheme, the messages sent by the receiver R to the sender S consist of only *three* elements, which is independent of n and k, while the messages from S to R are $(n+1)$ elements when the sender holds n secrets. Our scheme features a nice property of universal parameter, where the system parameter can be used by all senders and receivers. The proposed k-out-of-n oblivious transfer scheme is the most efficient two-round scheme in terms of the number of messages transferred between two communicating parties in known constructions. The scheme preserves the privacy of receiver's choice and sender's security.

Keywords: Oblivious Transfer, Privacy, Minimum Communication Cost, Encryption.

1. Introduction

Oblivious transfer (OT) is one of the most important building blocks to construct secure and privacy-preserving protocols in cryptography, such as contract

*Corresponding author
Email addresses: jl967@uowmail.edu.au (Jianchang Lai), ymu@uow.edu.au (Yi Mu), fuchun@uow.edu.au (Fuchun Guo), rc517@uowmail.edu.au (Rongmao Chen), sma@uow.edu.au (Sha Ma)

signing [11], private information retrieval [6] and secure function evaluation [12]. An oblivious transfer scheme is a two-party protocol between a *sender* S and a *receiver* R. The sender holds several secrets and the receiver wants to obtain some of them in the way that the receiver gets the secrets of his/her choice only, without revealing anything about his choice to the sender. The sender should not know which secrets are obtained by the receiver. The first OT scheme was proposed by Rabin [32], where the sender sends a secret to the receiver, and the receiver obtains the secret with probability 1/2. Even, Goldreich and Lempel [11] gave a more general OT scheme called 1-out-of-2 OT (OT_2^1) where the sender has two one-bit secrets (m_0, m_1). Brassard, Crépeau and Robert [4] extended the OT_2^1 to 1-out-of-n OT (OT_n^1) where the sender holds n secrets and the receiver wants to obtain one of them of its choice.

In OT schemes, one of the most general types is k-out-of-n oblivious transfer (OT_n^k), where the sender holds n different secrets and the receiver wants to obtain k ($k < n$) secrets simultaneously[1]. As the large bandwidth resource is not always available and usually expensive, it is desired to reduce the total communication cost as much as possible during the communication. Many OT_n^k schemes [28, 8, 7, 15, 14, 27, 34] have been studied. In these schemes, the ideal communication rounds can be two only. Supposing the indices of secrets and system parameters are known by the sender and the receiver, the receiver sends her/his choice as a request to the sender in the first round, then the sender should respond the request in the second round[2].

From the state-of-art two-round OT_n^k schemes in the literature, the most efficient one with the minimum number of messages from R to S was proposed by Guo, Mu and Susilo in [15]. The authors first presented an efficient subset membership encryption scheme (SME) and applied the SME scheme to con-

[1] The OT_n^k scheme with such features is called *non-adaptive* OT_n^k.

[2] We refer one communication (a flow of information transmission) between two parties as *one round* in this paper. Thus the ideal communication rounds OT schemes under this definition is "two-round" OT schemes.

struct an efficient OT_n^k scheme. In their scheme, the receiver sends only three messages to the sender, which is independent of k and n. While the tradeoff is that the messages from S to R are $2n$, which is larger than the other OT_n^k schemes. Chu and Tzeng [7] proposed an efficient OT_n^k scheme based on [29]. Their scheme achieves the minimum number of messages from S to R, namely $n + k$, in the known constructions. But the messages from R to S in [7] are linear in the receiver's choice k.

1.1. Ideal Communication Cost

In the OT_n^k schemes with ideal communication rounds, as the sender holds n secrets and does not know the receiver's choice, S has to send the messages including n encrypted secrets to R. To decrypt the ciphertext successfully, the messages from S to R should be $O(n)$. For example, to encrypt a secret using ElGamal encryption, the ciphertext consists of two elements. One of them is used to recover the encryption key. Thus, in a secure OT_n^k scheme, the ideal communication cost is that the messages from R to S are constant which is independent of n and k, and the messages from S to R is $(n + 1)$ in public key cryptosystems. The additional one element is used to recover the decryption keys. From the above, we note that both schemes in [15] and [7] cannot achieve ideal communication cost. The scheme in [15] achieves ideal communication cost from R to S, but cannot achieve that from S to R. While the scheme in [7] does not possess this property in any round. There are no existing known constructions of secure OT_n^k in the literature can achieve the ideal communication cost.

1.2. Our Contributions

In this paper we propose the first two-round OT_n^k scheme with ideal communication cost; precisely, the messages from R to S are constant, with only *three* elements which are independent of n and k, and the messages from S to R contain $n + 1$ elements. Compared with the two-round OT_n^k schemes in the literature, our proposed OT_n^k scheme is the most efficient one in terms of each

Table 1: Comparison of two-round OT_n^k schemes in terms of communication cost.

	System Parameter	Messages ($R \to S$)	Messages ($S \to R$)
Mu et al. [27]	$2n\|\mathbb{Z}_p\|$	$2n$	$2n$
Zhang and Wang [34]	$\|\mathbb{G}\|$	$k+3$	$2n$
Chu and Tzeng [8]	$\|\mathbb{G}\|$	k	$n+k+1$
Chu and Tzeng [7]	$\|\mathbb{G}\|$	k	$n+k$
Guo et al. [14]	$(n^2+3n+3)\|\mathbb{G}\|+\|\mathbb{G}_T\|$	4	$3n$
Guo et al. [15]	$(n+1)\|\mathbb{G}\|$	3	$2n$
Ours	$(2n+2)\|\mathbb{G}\|$	3	$n+1$

round. The system parameter in our scheme is universal, which can be used by any users. We give an overview of the comparing results in Table 1.

In the first round, the receiver sends a token which contains the receiver's choice, and a proof information which is used to prove that its choice is not larger than k to the sender. In the second round, the sender responds with encrypted secrets after checking the validity of the received token. Finally, the receiver uses its choice set and secret key to decrypt the ciphertext and only retrieves the secrets whose indexes are in its choice set. To analyze the security of our proposed scheme, we propose two new assumptions and prove that our proposed assumptions are hard in the generic group model. Based on these two assumptions, we derive the security of our proposed scheme under three non-adaptive games and show that our scheme is unconditionally secure for receiver's choice and preserves the sender's security. Any choice larger than k can be detected easily by the sender. The tradeoff of our scheme is that the system parameter consists of $2n+2$ elements in the group $\mathbb{G}$.

1.3. Other Related Work

The notion of oblivious transfer was put forth by Rabin [32] to achieve secure two-party communication, and the author proposed a bit-OT protocol based on

quadratic roots modulo a composite. Even et al. [11] extended this notion to OT_2^1, where the sender holds two one-bit secrets and the receiver would like to receive one of them of his choice. Brassard et al. [4] presented a more general OT_n^1 scheme, which is also known as "all-or-nothing disclosure of secrets (ANDOS)". After that, several oblivious transfer variants have been proposed. They mainly focused on improving the efficiency of secure computation and security.

Oblivious transfer with adaptive queries was proposed by Naor and Pinkas [28] where the receiver can query the ith value depending on the first $i-1$ values. In this setting, it requires two phases, the commitment phase and the transfer phase. S first should commit all secrets to R in the commitment phase, and then R makes queries of the secret one by one in the transfer phase. Adaptive OT scheme has many applications, such as oblivious search and oblivious database queries. Ogata and Kurosawa [31] proposed an efficient adaptive OT based on the RSA system. Chu and Tzeng [8] proposed an adaptive OT_n^k scheme based on basic cryptographic techniques, which is as efficient as the scheme in [31]. Camenisch, Neven and Shelat [5] presented simulatable adaptive oblivious transfer schemes that achieve a stronger security notion. They used the technique of blind signatures and both sender and receiver security can be fully simulated in their scheme. Green and Hohenberger [13] proposed a simulatable oblivious transfer based on blind identity-based encryption. Further research along this line appears in [22, 20, 33].

To solve the problem of that given a small number of oblivious transfers, how to implement a large number of oblivious transfers, Ishai et al. [17] proposed the first two efficient oblivious transfer extension protocols. Their first protocol is only secure against semi-honest (positive) adversaries. The improvements and optimizations to this protocol were given in [1, 18]. To achieve the security for malicious (active) adversaries, the authors used the cut-and-choose technique and proposed the second OT extension protocol. After that, constructing an efficient OT extension protocol that is secure against malicious adversaries has been extensively studied. There are some approaches to achieve security against

malicious adversaries for OT extensions, but most of them were built on the semi-honest protocol of [17] and added the consistency checks.

Nielsen et al. [30] proposed an alternative approach for achieving actively-secure OT extension in the random oracle model. They checked the consistency by hashing the strings that were transferred in the base-OTs and it was very efficient. Larraia [21] introduced a new approach to constructing OT extension schemes, which can be proved secure against malicious adversaries without the random oracle. Asharov et al. [2] presented a more efficient OT extension protocol for the setting of malicious adversaries. Their protocol requires less communication than previous works and can be proved secure with or without random oracles.

The technique of cut-and-choose has been regarded as a useful basic block to construct the two-party protocols against malicious adversaries, and formalized and proven secure by Lindell and Pinkas [24]. A number of cut-and-choose oblivious transfer protocols [25, 26, 23, 16] have focused on reducing the concrete overhead of the cut-and-choose approach. Kolesnikov and Kumaresan [19] proposed an efficient cut-and-choose OT scheme and its variants with small concrete communication overhead in an OT-hybrid model.

Organization. The rest of this paper is organized as follows. In Section 2, we show some preliminaries including notations, the definition of OT_n^k scheme and the corresponding security models. In Section 3, we first review the bilinear map and aggregation algorithm. We then present the OT_n^k construction and evaluate the scheme. We give two new complexity assumptions in Section 4. The security analysis of the proposed scheme is presented in Section 5 and we conclude the paper in Section 6.

2. Preliminaries

In this section, we give the definition of OT scheme and its security requirements. Before describing the definitions, we give some notations that will be used in the remaining sections.

Table 2: Notations

Symbol	Description
λ	Security parameter.
SP	System parameter.
M	Secret set held by the sender.
G	Selected index set by the receiver.
sk	Secret key held by the receiver.
$\mathsf{P}(\mathsf{G})$	Token.
Σ	Proof information.
CT	Ciphertext.
$\lvert\mathsf{G}\rvert$	The number of elements in G.
$[n]$	Set $\{1, 2, \cdots, n\}$.
$\mathbb{G}, \mathbb{G}_T$	Cyclic multiplicative groups.
$\lvert\mathbb{G}\rvert$	The size of group $\mathbb{G}$.
p	A large prime number.
$\mathbb{Z}_p$	Set $\{0, 1, 2, \cdots, p-1\}$.
$\mathbb{Z}_p^*$	Set $\{1, 2, \cdots, p-1\}$.
$\mathsf{L}\backslash\mathsf{G}^*$	Elements in set L but not in set G^*.

2.1. Notations

Now, we present the notations and conventions used in this paper in Table 2. Let $\mathbb{N}$ denote the set of natural numbers. We say a function $\epsilon : \mathbb{N} \to [0, 1]$ is *negligible* if for every $d \in \mathbb{N}$, there exists a $\lambda_d \in \mathbb{N}$ such that $\epsilon(\lambda) \leq \lambda^{-d}$ for all $\lambda > \lambda_d$.

2.2. Definition of OT_n^k Scheme

Let $m_1, m_2, \cdots, m_n$ be n secrets held by the sender and receiver's k chosen indices are $l_1, l_2, \cdots, l_k$, where $k < n$. We assume that the sender is honest-but-curious, that is, it will not send the secrets which are not the same as what he claimed, including the content and the order. We borrow the idea of subset membership encryption in [15] and give the definition of the non-adaptive OT_n^k

Inputs:

- The system parameter SP, generated by a Setup algorithm which takes a security parameter λ as input, denoted by $\mathsf{Setup}(1^\lambda)$.
- S holds a secret set $M = \{m_1, m_2, \cdots, m_n\}$.
- R holds his choice set $\mathsf{G} = \{l_1, l_2, \cdots, l_k\}$, where $k \in [n]$.

Protocol:

1. $R \rightarrow S$: R takes the system parameter SP and the choice set G as input, it outputs (T, sk) where sk is a secret key and T includes a token $\mathsf{P}(\mathsf{G})$ which contains the receiver's choice, a proof information Σ which is used to check the number of choice set and a number k. Then R sends T to S.
2. $S \rightarrow R$: S takes the system parameter SP, T and a secret set M as input, it first checks whether $|\mathsf{G}| \leq k$ via a verification algorithm and aborts if no. If yes, it performs an encryption algorithm $\mathsf{Encrypt}$ and outputs the ciphertext $CT = \mathsf{Encrypt}(\mathsf{SP}, \mathsf{P}(\mathsf{G}), M)$. Then S sends CT to R.

Outputs: S receives no output. R takes the system parameter SP, the ciphertext CT, his secret key sk and the choice set G as input.

- For every $i \in \mathsf{G}$, it outputs m_i.
- For every $i \notin \mathsf{G}$, it outputs $\perp$.

Figure 1: The definition of OT_n^k scheme

scheme in Fig.1.

Correctness. An OT_n^k scheme is correct if the receiver obtains the secrets of his choice only when the sender and the receiver follow the steps of the scheme.

2.3. Security Models

An OT_n^k scheme should meet the following security requirements:

- Receiver's privacy

  Given any two distinct sets of choice $\mathsf{G_0} = \{l_1, l_2, \cdots, l_{k_0}\}$ and $\mathsf{G_1} = \{l'_1, l'_2, \cdots, l'_{k_1}\}$, where $k_0, k_1 < k$, the $\mathsf{P(G)}$ received by the sender can not be distinguished whether it generated under $\mathsf{G_0}$ or $\mathsf{G_1}$. That is, it is hard to distinguish whether $\mathsf{G} = \mathsf{G_0}$ or $\mathsf{G} = \mathsf{G_1}$.

- Sender's security

  - Indistinguishability: Given any choice set $\mathsf{G} = \{l_1, l_2, \cdots, l_k\}$, the unchosen secrets are indistinguishable from the random ones. More precisely, given $(\mathsf{G}, \mathsf{P(G)})$, a target index $I^* \in [n]$ where $I^* \notin \mathsf{G}$, two secrets m_0, m_1 (one of them is regarded as I^*-th secret) from the same space and a ciphertext CT^* generated under $\left(I^*, m_{I^*}, \mathsf{P(G)}\right)$ where $m_{I^*} = m_c$, c is randomly chosen from $\{0, 1\}$, it is hard to distinguish whether $c = 0$ or $c = 1$.
  - Accountability: Given $(\mathsf{P(G)}, \mathsf{G}, sk)$ satisfying $|\mathsf{G}| > k$, it is hard to generate T that passes the verification algorithm.

We define three non-adaptive security models via game playing to meet the above security requirements. The games are defined between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$, which are stated as follows.

Game 1: Receiver's privacy.

This model guarantees that the adversary (sender) learns nothing about which indices are chosen by the receiver without a valid secret key.

- **Setup:** $\mathcal{C}$ takes as input a security parameter λ and runs the Setup algorithm to generate the system parameter SP. Then it sends SP to $\mathcal{A}$.

- **Challenge:** $\mathcal{A}$ outputs two distinct choice sets $\mathsf{G_0} = \{l_1, l_2, \cdots, l_{k_0}\}$, $\mathsf{G_1} = \{l'_1, l'_2, \cdots, l'_{k_1}\}$, where $k_0, k_1 \leq k$. $\mathcal{C}$ picks a random bit $c \in \{0, 1\}$, sets $\mathsf{G} = \mathsf{G_c}$ and generates $(\mathsf{P}(\mathsf{G}), \mathsf{G}, sk)$. Then it computes $T = (\mathsf{P}(\mathsf{G}), \Sigma, k)$ and sends it to $\mathcal{A}$.

- **Output:** $\mathcal{A}$ outputs its guess c' and wins the game if $c' = c$.

We define the advantage of $\mathcal{A}$ in winning the game as

$$\mathsf{Adv}^{\mathsf{RP}}(\lambda) = |\Pr[c' = c] - 1/2|.$$

Definition 1. *We say that an OT_n^k scheme preserves the privacy of receiver's choice with (t, ϵ) if for all t-probability polynomial time (PPT) adversaries, $\epsilon = \mathsf{Adv}^{\mathsf{RP}}(\lambda)$ is negligible. The choices of the receiver are unconditionally secure if $\epsilon = 0$ for all PTT adversaries.*

Game 2: Sender's security (Indistinguishability).

The model of indistinguishability captures the attack that even the adversary (receiver) can access the secret key, it cannot retrieve the secrets whose indices are not in the choice set.

- **Init:** $\mathcal{A}$ outputs a set $\mathsf{G}^* = \{l_1^*, l_2^*, \cdots, l_k^*\}$ and a target index $I^* \in [n] \backslash \mathsf{G}^*$ which it wants to challenge.

- **Setup:** $\mathcal{C}$ takes as input a security parameter λ and runs the Setup algorithm to generate the system parameter SP. Then it sends SP to $\mathcal{A}$.

- **Challenge:** $\mathcal{A}$ outputs $\left(\mathsf{G}^*, \mathsf{P}(\mathsf{G}^*), sk^*\right)$ and two secrets m_0, m_1 from the same secret space for challenge. $\mathcal{C}$ first checks whether $\mathsf{P}(\mathsf{G}^*)$ is generated from G^* by using the secret key sk^*. If yes, it picks a random bit $c \in \{0, 1\}$ and generates the challenge ciphertext $CT^* = \mathsf{Encrypt}(\mathsf{SP}, \mathsf{P}(\mathsf{G}^*), m_c)$ for the secret with index I^*. It then sends CT^* to $\mathcal{A}$. Otherwise, it aborts.

- **Guess:** $\mathcal{A}$ outputs a guess c' and wins the game if $c' = c$.

We define the advantage of $\mathcal{A}$ in winning the game as

$$\mathsf{Adv}^{\mathsf{SS}}(\lambda) = |\Pr[c' = c] - 1/2|.$$

Definition 2. *An OT_n^k scheme preserves the indistinguishability security and it is (t, ϵ)-secure if for all t-polynomial time adversaries, $\epsilon = \mathsf{Adv}^{\mathsf{SS}}(\lambda)$ is negligible.*

Game 3: Sender's security (Accountability).

The accountability security guarantees that any forge proofs generated by the choice set with number larger than k will be detected even the adversary (receiver) knows the secret key of computing $\mathsf{P}(\mathsf{G})$.

- **Setup:** $\mathcal{C}$ takes as input a security parameter λ and runs the Setup algorithm to generate the system parameter SP. Then it sends SP to $\mathcal{A}$.
- **Challenge:** $\mathcal{A}$ outputs $(\mathsf{P}(\mathsf{G}^*), \mathsf{G}^*, sk^*)$ and k for challenge. $\mathsf{P}(\mathsf{G}^*)$ was generated from (G^*, sk^*) and $|\mathsf{G}^*| > k$.
- **Output:** $\mathcal{A}$ outputs $T^* = (\mathsf{P}(\mathsf{G}^*), \Sigma^*, k)$ and wins the game if T^* passes the verification algorithm.

We define the advantage of $\mathcal{A}$ in winning the game as $\mathsf{Adv}^{\mathsf{A}}(\lambda)$ in computing such T^*.

Definition 3. *An OT_n^k scheme captures the accountability security and it is (t, ϵ)-secure if for all t-polynomial time adversaries, $\epsilon = \mathsf{Adv}^{\mathsf{A}}(\lambda)$ is negligible.*

3. Our OT_n^k Scheme

In this section, we present the concrete construction of our scheme. First, we briefly review bilinear groups and the aggregation algorithm used in our proposed scheme.

Let $\mathbb{G}$, $\mathbb{G}_T$ be two cyclic multiplicative groups of the same prime order p, g be the generator of $\mathbb{G}$. A map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is said to be a bilinear map if (1) for all $u, v \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$, we have $e\left(u^a, v^b\right) = e(u, v)^{ab}$ and $e\left(g, g\right) \neq 1$, and there exists an efficient algorithm to compute $e\left(g, h\right)$ for all $g, h \in \mathbb{G}$. A bilinear group $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ is composed of objects as described above.

3.1. Aggregation Algorithm

Given group elements $\left(g^{\frac{1}{a+1}}, g^{\frac{1}{a+2}}, \cdots, g^{\frac{1}{a+n}}\right)$ for unknown $a \in \mathbb{Z}_p^*$, the aim of aggregation algorithm is to compute

$$g^{\frac{1}{(a+1)(a+2)\cdots(a+n)}}.$$

From [10], we have that there exists integers $s_1, s_2, \cdots, s_n$ satisfying

$$s_1 \prod_{i\in[n], i\neq 1}(a+i) + s_2 \prod_{i\in[n], i\neq 2}(a+i) + \cdots + s_n \prod_{i\in[n], i\neq n}(a+i) = 1$$

for all a such that we have

$$\begin{aligned} g^{\frac{s_1}{a+1}} \cdot g^{\frac{s_2}{a+2}} \cdots g^{\frac{s_n}{a+n}} &= g^{\frac{s_1}{a+1} + \frac{s_2}{a+2} + \cdots \frac{s_n}{a+n}} \\ &= g^{\frac{s_1 \prod_{i\in[n], i\neq 1}(a+i) + s_2 \prod_{i\in[n], i\neq 2}(a+i) + \cdots + s_n \prod_{i\in[n], i\neq n}(a+i)}{(a+1)(a+2)\cdots(a+n)}} \\ &= g^{\frac{1}{(a+1)(a+2)\cdots(a+n)}}. \end{aligned}$$

The aggregation is independent of the group element g, which means it is available for all group elements as long as they are identical.

3.2. Overview of technique

In this subsection, we give a highlight of our scheme. Let $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ be a bilinear group with two generators $g, h \in \mathbb{G}$. α is known by the system generator only. In our setting, the token $\mathsf{P}(\mathsf{G}) = g^{\frac{s}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}$ and each secret m_i is encrypted using a key $e\left(g^{\frac{1}{\alpha+i}}, h\right)^r$, that is $C_i = e\left(g^{\frac{1}{\alpha+i}}, h\right)^r \cdot m_i$. If $i \in \mathsf{G}$, $h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}$ is computable from the system parameters. Thus, the decryptor can retrieve the encryption (decryption) key $e\left(g^{\frac{1}{\alpha+i}}, h\right)^r$ together with $\mathsf{P}(\mathsf{G})^r$ and its private key s. If $i \notin \mathsf{G}$, the value of $h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}$ cannot be computed. Therefore, the decryptor is unable to retrieve the encryption key and get the corresponding secret.

3.3. Construction

The concrete OT_n^k scheme is illustrated in Fig.2. A trusted third party called PKG establishes the system by choosing a security parameter λ and a

Inputs:

- System parameter SP. The PKG runs the **Setup** algorithm as follow. Given a security parameter λ, this algorithm generates a bilinear group $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ with two generators $g, h \in \mathbb{G}$. Then it randomly chooses $\alpha \in \mathbb{Z}_p^*$ as the system secret key and computes $g_i = g^{\frac{1}{\alpha+i}}, h_i = h^{\alpha^i}$ for all $i = 1, 2, \cdots, n$. The system parameter SP consists of $(\mathbb{BG}, g, h, g_1, g_2, \cdots, g_n, h_1, h_2, \cdots, h_n)$.
- S holds a set of secrets $M = \{m_1, m_2, \cdots, m_n\} \in \mathbb{G}_T$.
- R holds his choice set $\mathsf{G} = \{l_1, l_2, \cdots, l_k\} \subset [n]$.

Protocol:

1. $R \to S$: Given a choice set $\mathsf{G} = \{l_1, l_2, \cdots, l_k\}$ and the system parameters SP, R picks a random $s \in \mathbb{Z}_p^*$ as his secret key sk and uses the $\mathsf{Aggregation}$ algorithm to compute
$$\mathsf{P}(\mathsf{G}) = g^{\frac{s}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}, \quad \Sigma = h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)\cdot\alpha^{n-k}}{s}},$$
and sets $T = (\mathsf{P}(\mathsf{G}), \Sigma, k)$.
2. $S \to R$: S runs the **Encrypt** algorithm as follow. Given a set $T = (\mathsf{P}(\mathsf{G}), \Sigma, k)$, a set of secrets $M = \{m_1, m_2, \cdots, m_n\}$ and the system parameter SP, it first performs the verification algorithm as: $e\big(\mathsf{P}(\mathsf{G}), \Sigma\big) = e\left(g, h^{\alpha^{n-k}}\right)$. If the equation does not hold, it aborts. Otherwise, it accepts $|\mathsf{G}| \leq k$. Then it picks a random $r \in \mathbb{Z}_p^*$ and computes the ciphertext CT for the secrets as
$$C_0 = \mathsf{P}(\mathsf{G})^r = g^{\frac{rs}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}$$
together with, for each $i = 1, 2, \cdots, n$:
$$C_i = e\left(g^{\frac{1}{\alpha+i}}, h\right)^r \cdot m_i.$$

Outputs: R runs the decryption algorithm as follow. Given a ciphertext $CT = (C_0, C_1, \cdots, C_n)$, a choice set $\mathsf{G} = \{l_1, l_2, \cdots, l_k\}$, a secret key sk and the system parameter SP, for each $i \in \mathsf{G}$, R computes
$$m_i = C_i \cdot e\left(C_0, h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}\right)^{-\frac{1}{sk}}.$$

Figure 2: Our OT_n^k scheme

random α as the system secret key, and generates the system parameter $SP = (\mathbb{BG}, g, h, g_1, g_2, \cdots, g_n, h_1, h_2, \cdots, h_n)$, which is public known.

In the first round, the receiver chooses a random $s \in \mathbb{Z}_p^*$ as its secret key and a set $\mathsf{G} = \{l_1, l_2, \cdots, l_k\}$. It then uses the Aggregation algorithm to compute a token $\mathsf{P}(\mathsf{G}) = g^{\frac{s}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}$, and computes a proof information $\Sigma = h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)\cdot\alpha^{n-k}}{s}}$ for its choice set G.

In the second round, upon receiving a request from a receiver, the sender first tests whether $e\big(\mathsf{P}(\mathsf{G}), \Sigma\big) = e\left(g, h^{\alpha^{n-k}}\right)$ and aborts if no. Otherwise, S picks a random $r \in \mathbb{Z}_p^*$ and computes a ciphertext CT for the secrets as $C_0 = \mathsf{P}(\mathsf{G})^r = g^{\frac{rs}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}$ together with, for each $i = 1, 2, \cdots, n$: $C_i = e\left(g^{\frac{1}{\alpha+i}}, h\right)^r \cdot m_i$.

Once receiving the encrypted secrets CT from the sender, the receiver, for each $i \in \mathsf{G}$, computes $m_i = C_i \cdot e\left(C_0, h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}\right)^{-\frac{1}{sk}}$. After the decryption, R gets only k secrets with indexes in G from S.

3.4. Correctness

Below we show that our construction meets the correctness requirements. When everything is computed as above, if $|\mathsf{G}| \leq k$, we have

$$\begin{aligned} e\big(\mathsf{P}(\mathsf{G}), \Sigma\big) &= e\left(g^{\frac{s}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}, h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)\cdot\alpha^{n-k}}{s}}\right) \\ &= e\left(g, h^{\alpha^{n-k}}\right). \end{aligned}$$

If $|\mathsf{G}| > k$, it is impossible for receiver to compute the Σ which passes the checking process. In decryption phase, if $i \in \mathsf{G}$, we can compute $h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}$ from the system parameter, thus we have

$$\begin{aligned} & C_i \cdot e\left(C_0, h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}\right)^{-\frac{1}{sk}} \\ &= m_i \cdot e\left(g^{\frac{1}{\alpha+i}}, h\right)^r \cdot e\left(g^{\frac{rs}{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}}, h^{\frac{(\alpha+l_1)(\alpha+l_2)\cdots(\alpha+l_k)}{(\alpha+i)}}\right)^{-\frac{1}{s}} \\ &= m_i \cdot e\left(g^{\frac{1}{\alpha+i}}, h\right)^r \cdot e\left(g^{\frac{1}{\alpha+i}}, h\right)^{-r} \\ &= m_i \cdot e(g, h)^{r\left(\frac{1}{\alpha+i} - \frac{1}{\alpha+i}\right)} \\ &= m_i. \end{aligned}$$

3.5. Evaluation

Our proposed two-round OT_n^k scheme is very efficient in terms of communication cost. In our construction, the receiver only sends two group elements and one integer to the sender. The overhead from R to S is the same as the most efficient one [15] in the literature, which is independent of n and k. While, in the second round, the transfer messages from S to R are $n+1$, more precisely, one element in $\mathbb{G}$ and n elements in $\mathbb{G}_T$. Our scheme achieves the minimum communication overhead in both two rounds comparing with the existing known OT_n^k schemes and achieves the ideal communication cost in public key setting. The total communication cost in our scheme is $n+4$ elements, namely $3|\mathbb{G}|$, $n|\mathbb{G}_T|$ and $|\mathbb{Z}_p|$. The size of the system parameter in our scheme is linear in the number of the secrets held by the sender.

Note that in our construction, to avoid sending linear messages to the sender and achieving constant overhead, the receiver requires performing the Aggregation algorithm once which is expensive and k times exponentiation operations in $\mathbb{G}$. We stress that it is a tradeoff. To reduce the communication overhead, the price to pay is to execute more computation. For the sender, as we described above, without knowing which secrets are required, it has to encrypt and send all secrets, which requires performing one group exponentiation operation in $\mathbb{G}$ and n exponentiation operations in $\mathbb{G}_T$.

In the decryption phase, as the receiver holds its choice set G and knows the secret key sk, for each index, it can pre-compute $k-1$ group exponentiation operations associated with h without the knowledge of response from the sender. When receiving a ciphertext from the sender, the receiver needs to perform k pairing computations to retrieve k message. Therefore, the decryption cost is k pairing computations in all, which is the same as that in [15].

4. Complex Assumptions

In this section, we define two specific Diffie-Hellman exponent problems introduced by Boneh, Boyen and Goh [3] and denote them as $(f, \hat{g}, F)$-GDDHE

and $(\hat{f}, n)$-DHE. The security of the proposed scheme relies on the corresponding two defined assumptions. The former one is by slightly modifying the (f, g, F)-GDDHE problem in [9]. We first give the definition of $(f, \hat{g}, F)$-GDDHE as follows.

$(f, \hat{g}, F)$-GDDHE **Problem.**

Input: Let $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ be a bilinear group and g_0 be a generator of $\mathbb{G}$. Let $f(x) = (x+1)(x+2)\cdots(x+n) \in \mathbb{Z}_p[x]$ and $(x+t)^2 \nmid f(x)$, $g(x) = \frac{f(x)}{(x+t)\cdot\prod_{i=1}^{k}(x+l_i)} \in \mathbb{Z}_p[x]$, where $t, l_i \in [n]$ for $i = 1, 2, \cdots, k < n$ and all are distinct. The following group elements are given

$$g_0, \quad g_0^a, \quad g_0^{a^2}, \quad \cdots, \quad g_0^{a^n}, \quad g_0^{f(a)},$$
$$g_0^{g(a)\cdot(a+t)\cdot b},$$
$$g_0^{w\cdot(a+t)}, \quad g_0^{w\cdot(a+t)\cdot a}, \quad g_0^{w\cdot(a+t)\cdot a^2}, \quad \cdots, \quad g_0^{w\cdot(a+t)\cdot a^n},$$

and $T \in \mathbb{G}_T$, where a, b, t and w are unknown random exponents in $\mathbb{Z}_p$.

Output: a bit μ.

We define that the problem has been solved correctly if the output is

$$\mu = \begin{cases} 1 & T = e(g_0, g_0)^{w\cdot b\cdot f(a)}, \\ 0 & T \neq e(g_0, g_0)^{w\cdot b\cdot f(a)}. \end{cases}$$

Let us denote by I the input of the instance, true the event that T is indeed equal to $e(g_0, g_0)^{w\cdot b\cdot f(a)}$ and by false the event that T is a random element of $\mathbb{G}_T$. The goal of $(f, \hat{g}, F)$-GDDHE is to decide whether T is equal to $e(g_0, g_0)^{w\cdot b\cdot f(a)}$ or to some random element of $\mathbb{G}_T$. We then define the advantage of an algorithm $\mathcal{B}$ in solving the $(f, \hat{g}, F)$-GDDHE problem in $\mathbb{BG}$ as

$$\mathsf{Adv}^{\mathsf{gddhe}}(f, \hat{g}, F, \mathcal{B}) = \left| \Pr\left[\mathcal{B}(I) = 1 \middle| \mathsf{true}\right] - \Pr\left[\mathcal{B}(I) = 1 \middle| \mathsf{false}\right] \right|,$$

where the probability is over all random choices and over the random coins of $\mathcal{B}$. We say that the $(f, \hat{g}, F)$-GDDHE assumption holds if for any probability polynomial time algorithm $\mathcal{B}$, we have $\mathsf{Adv}^{\mathsf{gddhe}}(f, \hat{g}, F, \mathcal{B})$ is negligible.

Theorem 1. *The defined* $(f, \hat{g}, F)$-GDDHE *problem is one of the GDDHE problems fulfilling the hardness conditions defined in [3].*

Proof. As the analysis in [9], the $(f, \hat{g}, F)$-GDDHE problem can be reformulated as

$$\begin{aligned} P &= \left(\begin{array}{l} 1,\ a,\ a^2,\ \cdots,\ a^n,\ f(a), \\ g(a)\cdot(a+t)\cdot b, \\ w\cdot(a+t), w\cdot(a+t)\cdot a, w\cdot(a+t)\cdot a^2, \cdots, w\cdot(a+t)\cdot a^n, \end{array} \right), \\ Q &= 1, \\ F &= w\cdot b\cdot f(a). \end{aligned}$$

Now we proof that F is independent of $\langle P, Q\rangle$. We show this statement by giving a contradiction as follows.

Assume that F is dependent of $\langle P, Q\rangle$ (i.e. $F \in \langle P, Q\rangle$), then there exists not-all-zero coefficients $\{a_{i,j}\}$ and b_1 such that

$$F = \sum a_{i,j} p_i p_j + b_1 q_1,$$

where $p_i, p_j \in P$ and $q_1 = 1 \in Q$. Noting that $F = w\cdot b\cdot f(\alpha)$, we have that, all $a_{i,j}p_ip_j$ should be multiples of $w\cdot b$. Below we list all the possible products of two polynomials from P which are multiples of $w\cdot b$.

$$R = \left(\begin{array}{ll} w\cdot b\cdot(a+t)^2\cdot g(a), & w\cdot b\cdot a\cdot(a+t)^2\cdot g(a), \\ w\cdot b\cdot a^2\cdot(a+t)^2\cdot g(a), & \cdots,\ \ w\cdot b\cdot a^n\cdot(a+t)^2\cdot g(a) \end{array} \right).$$

If $F \in \langle P, Q\rangle$, we have

$$F = w\cdot b\cdot f(a) = w\cdot b\cdot(a+t)^2\cdot A(a)\cdot g(a)$$

where $A(a)$ is a polynomial with $deg(A(a)) \leq n$. We simplify the above equation as follows,

$$f(a) = (a+t)^2\cdot A(a)\cdot g(a).$$

Therefore, we have $(a+t)^2 | f(a)$, which is contradiction with $(a+t)^2 \nmid f(a)$. The assumption that F is dependent of $\langle P, Q\rangle$ cannot hold and hence $F \notin \langle P, Q\rangle$. This completes the proof.

Definition 4 ($(\hat{f},n)$-DHE-Problem). *Let $\mathbb{G}$ be a cyclic group of prime order p and g,h be two generators of $\mathbb{G}$. The $(\hat{f},n)$-DHE problem is, given $\left(g, g^{\frac{1}{a+1}}, \cdots, g^{\frac{1}{a+n}}, h, h^a, h^{a^2}, \cdots, h^{a^n}\right)$ for unknown $a \in \mathbb{Z}_p$, output $\left(f(x), h^{f(a)}\right)$, where $f(x) \in \mathbb{Z}_p[x]$ is a polynomial function with degree $n' > n$.*

Suppose that $q(x) = (a+1)(a+2)\cdots(a+n) \in \mathbb{Z}_p$ and $g = h^{q(x)}$, we can compute $\left(g, g^{\frac{1}{a+1}}, g^{\frac{1}{a+2}}, \cdots, g^{\frac{1}{a+n}}\right)$ from $\left(h, h^a, h^{a^2}, \cdots, h^{a^n}\right)$. That is, the $(\hat{f},n)$-DHE problem equals to the hard problem, given $\left(h, h^a, h^{a^2}, \cdots, h^{a^n}\right)$ for unknown $a \in \mathbb{Z}_p$, output $\left(f(x), h^{f(a)}\right)$, where $f(x) \in \mathbb{Z}_p[x]$ is a polynomial function with degree $n' > n$, which is defined in [14]. We say that the $(\hat{f},n)$-DHE assumption holds if for any probability polynomial time algorithm $\mathcal{B}$, the probability to solve the $(\hat{f},n)$-DHE problem is negligible.

5. Security Analysis

We derive the security of our proposed scheme under $(f,\hat{g},F)$-GDDHE assumption and $(\hat{f},n)$-DHE assumption.

5.1. Receiver Privacy

Theorem 2. *For our scheme, receiver's choices are unconditionally secure.*

Proof. The simulator $\mathcal{B}$ establishes the system by choosing a security parameter λ and a random $\alpha \in \mathbb{Z}_p^*$. Then it generates the system parameter SP as

$$\left(\mathbb{BG}, g, g^{\frac{1}{\alpha+1}}, g^{\frac{1}{\alpha+2}}, \cdots g^{\frac{1}{\alpha+n}}, h, h^{\alpha}, h^{\alpha^2}, \cdots, h^{\alpha^n}\right).$$

Challenge: $\mathcal{A}$ outputs two distinct choice sets $\mathsf{G_0} = \{l_1, l_2, \cdots, l_{k_0}\}$ and $\mathsf{G_1} = \{l'_1, l'_2, \cdots, l'_{k_1}\}$ where $k_0, k_1 \leq k$. $\mathcal{B}$ picks a random $sk \in \mathbb{Z}_p^*$ as the secret key. Let $T = \left(\mathsf{P}(\mathsf{G}), \Sigma, k\right)$ be generated from $\mathsf{G} = \mathsf{G_0}$. We have

$$\mathsf{P}(\mathsf{G}) = g^{\frac{sk}{\prod_{i=1}^{k_0}(\alpha+l_i)}}, \quad \Sigma = h^{\frac{\prod_{i=1}^{k_0}(\alpha+l_i)\cdot\alpha^{n-k}}{sk}}.$$

For any distinct set $\mathsf{G_1} = \{l'_1, l'_2, \cdots, l'_{k_1}\}$, we define $sk' \in \mathbb{Z}_p$ as

$$sk' = sk \cdot \frac{\prod_{i=1}^{k_1}(\alpha + l'_i)}{\prod_{i=1}^{k_0}(\alpha + l_i)}.$$

We have

$$\mathsf{P}(\mathsf{G}) = g^{\frac{sk}{\prod_{i=1}^{k_0}(\alpha+l_i)}} = g^{\frac{sk'}{\prod_{i=1}^{k_1}(\alpha+l'_i)}} = \mathsf{P}(\mathsf{G}_1),$$
$$\Sigma = h^{\frac{\prod_{i=1}^{k_0}(\alpha+l_i)\cdot\alpha^{n-k}}{sk}} = h^{\frac{\prod_{i=1}^{k_1}(\alpha+l'_i)\cdot\alpha^{n-k}}{sk'}} = \Sigma',$$

such that $(\mathsf{P}(\mathsf{G}_0), \Sigma, k) = (\mathsf{P}(\mathsf{G}_1), \Sigma', k)$. As sk is randomly chosen from $\mathbb{Z}_p^*$, we have sk' is also universally random in $\mathbb{Z}_p^*$. Thus, the distributions of $(\mathsf{P}(\mathsf{G}), \Sigma, k)$ for both G_0 and G_1 are identical. Adversary has no advantage in guessing the receiver's choice set in $\mathsf{P}(\mathsf{G})$. Thus, our scheme achieves unconditionally secure for the choice of receiver. This completes the proof of **Theorem** 2.

5.2. Sender Security

Theorem 3 (Indistinguishability). *Our proposed scheme captures the indistinguishability of sender's security. That is, if the* $(f, \hat{g}, F)$-GDDHE *assumption holds, the adversary (receiver) gets no information about secrets* m_i*, where* $i \notin \mathsf{G}$.

Proof. Suppose there exists an adversary $\mathcal{A}$ who can distinguish the secret with advantage ϵ. We can construct an algorithm $\mathcal{B}$ that solves the $(f, \hat{g}, F)$-GDDHE problem with advantage ϵ. Let $\mathbb{BG} = (\mathbb{G}, \mathbb{G}_T, e, p)$ be a bilinear group, g_0 be a generator of $\mathbb{G}$, $\mathsf{L} = \{1, 2, \cdots, n\}$ be all secrets' indices. All these objects are known by $\mathcal{A}$ and $\mathcal{B}$. $\mathcal{B}$ runs $\mathcal{A}$ and programs the reduction as follows.

Init: $\mathcal{A}$ outputs a set $\mathsf{G}^* = \{l_1^*, l_2^*, \cdots, l_k^*\} \subset \mathsf{L}$ and a target secret index $I^* \in \mathsf{L}\backslash\mathsf{G}^*$ for challenge.

Setup: To setup the system parameter, $\mathcal{B}$ requires an instant of the $(f, \hat{g}, F)$-GDDHE problem where

$$t = I^*, \quad f(x) = \prod_{i=1}^{n}(x+i), \quad g(x) = \frac{f(x)}{(x+I^*)\cdot\prod_{i=1}^{k}(x+l_i^*)},$$

and $f(x)$ and $g(x)$ have n degrees and $(n-k-1)$ degrees respectively. $\mathcal{B}$ is then given a $(f, \hat{g}, F)$-GDDHE instance in $\mathbb{BG}$ as:

$$g_0, \quad g_0^a, \quad g_0^{a^2}, \quad \cdots, \quad g_0^{a^n}, \quad g_0^{f(a)},$$
$$g_0^{g(a)\cdot(a+I^*)\cdot b},$$
$$g_0^{w\cdot(a+I^*)}, \quad g_0^{w\cdot(a+I^*)\cdot a}, \quad g_0^{w\cdot(a+I^*)\cdot a^2}, \quad \cdots, \quad g_0^{w\cdot(a+I^*)\cdot a^n},$$

and $T \in \mathbb{G}_T$, which is either equal to $e(g_0, g_0)^{w \cdot b \cdot f(a)}$ or to some random element of $\mathbb{G}_T$. It implicitly sets $\alpha = a$ and

$$g = g_0^{f(a)}, \quad h = g_0^{w(a+I^*)},$$

which are available from the first row and third row of the instance. Then for g_i, h_i in the system parameter, we have

$$g_i = g^{\frac{1}{\alpha+i}} = g_0^{\frac{f(a)}{a+i}}, \quad h_i = h^{\alpha^i} = g_0^{w(a+I^*)\cdot a^i},$$

which can be computed from the group elements in the first row and third row of the instance respectively. Then the system parameter SP is

$$SP = (\mathbb{BG}, g, h, g_i, h_i : i = 1, 2, \cdots, n).$$

Challenge: $\mathcal{A}$ outputs $\left(\mathsf{P}(\mathsf{G}^*), sk^*\right)$ and two secrets m_0, m_1 from the same secret space for challenge. $\mathcal{B}$ first verifies that $\mathsf{P}(\mathsf{G}^*)$ is indeed generated from G^* by using the secret key sk^*. If the verification fails, it abouts. Otherwise, it picks a random bit $c \in \{0, 1\}$ and computes the challenge ciphertext $CT^* = (C_0^*, C_{I^*})$ as

$$C_0^* = g_0^{sk^* \cdot g(a) \cdot (a+I^*) \cdot b}, \quad C_{I^*} = T \cdot m_c.$$

C_0^* is computable from in the second row of the given instance. Let $r^* = b$, if $T = e(g_0, g_0)^{w \cdot b \cdot f(a)}$, we have

$$\begin{aligned}
\mathsf{P}(\mathsf{G}^*) &= g^{\frac{sk^*}{(\alpha+l_1^*)(\alpha+l_2^*)\cdots(\alpha+l_k^*)}} = g_0^{\frac{sk^* \cdot f(a)}{(a+l_1^*)(a+l_2^*)\cdots(a+l_k^*)}}, \\
C_0^* &= g_0^{sk^* \cdot g(a) \cdot (a+I^*) \cdot b} = g_0^{\frac{sk^* \cdot f(a) \cdot b}{(a+l_1^*)(a+l_2^*)\cdots(a+l_k^*)}} = \mathsf{P}(\mathsf{G}^*)^b = \mathsf{P}(\mathsf{G}^*)^{r^*}, \\
C_{I^*} &= T \cdot m_c \\
&= e\left(g_0, g_0\right)^{w \cdot b \cdot f(a)} \cdot m_c \\
&= e\left(g_0^{\frac{f(a)}{a+I^*}}, g_0\right)^{w \cdot (a+I^*) \cdot b} \cdot m_c \\
&= e\left(g^{\frac{1}{a+I^*}}, h\right)^b \cdot m_c \\
&= e\left(g^{\frac{1}{\alpha+I^*}}, h\right)^{r^*} \cdot m_c.
\end{aligned}$$

Therefore, CT^* is a valid challenge ciphertext.

Guess: $\mathcal{A}$ outputs its guess c' of c.

This completes the description of the reduction. At the end, $\mathcal{B}$ outputs 1 as the solution to the $(f, \hat{g}, F)$-GDDHE problem if $c' = c$. Otherwise, it outputs 0.

In the above simulation, there is no abortion caused by that $\mathcal{B}$ cannot simulate. If $T = e(g_0, g_0)^{w \cdot b \cdot f(a)}$, we have that the adversary will guess c correctly with probability $\frac{1}{2} + \epsilon$ according to the assumption. Otherwise, we have C_{I^*} is uniformly random and independent of C_0^*. The challenge ciphertext is a one-time pad encryption, such that the adversary can only guess a correct c with probability $\frac{1}{2}$. Therefore, the advantage of solving the $(f, \hat{g}, F)$-GDDHE problem is also ϵ, e.g. $\mathsf{Adv}^{\mathsf{gddhe}}(f, \hat{g}, F) = \epsilon$. This yields the **Theorem** 3.

We note that the security of accountability is due to the fact that if $|\mathsf{G}| > k$, for example $|\mathsf{G}| = k + 1$, the corresponding $\Sigma = h^{sk^{-1} \cdot \alpha^{n-k} \sum_{i \in G}(\alpha + i)}$ contains the group element $h^{\alpha^{n+1}}$ which is not computable from the system parameter SP. The proof is completed in the following theorem.

Theorem 4 (Accountability). *Our OT_n^k scheme captures the accountability of the sender security if the $(\hat{f}, n)$-DHE assumption holds.*

Proof. Suppose there exists an adversary $\mathcal{A}$ who can break accountability. We can construct an algorithm $\mathcal{B}$ that solves the $(\hat{f}, n)$-DHE problem. Let $\left(g, g^{\frac{1}{a+1}}, g^{\frac{1}{a+2}}, \cdots g^{\frac{1}{a+n}}, h, h^a, h^{a^2}, \cdots, h^{a^n}\right)$ be a random instance of $(\hat{f}, n)$-DHE problem taken as input by $\mathcal{B}$ and its goal is to output $\left(f(x), h^{f(a)}\right)$, where $f(x) \in \mathbb{Z}_p[x]$ is a polynomial function with degree $n' > n$. $\mathcal{B}$ interacts with $\mathcal{A}$ as the follows.

Setup: $\mathcal{B}$ implicitly sets $\alpha = a$ for unknown a and generates the system parameter as $SP = \left(\mathbb{BG}, g, g^{\frac{1}{a+1}}, g^{\frac{1}{a+2}}, \cdots g^{\frac{1}{a+n}}, h, h^a, h^{a^2}, \cdots, h^{a^n}\right)$.

Challenge: $\mathcal{A}$ outputs $(\mathsf{P}(\mathsf{G}^*), \mathsf{G}^*, sk^*)$ and k for challenge, where $|\mathsf{G}^*| > k$ and

$$\mathsf{P}(\mathsf{G}^*) = g^{\frac{sk^*}{\prod_{l_i \in \mathsf{G}^*}(a + l_i)}}.$$

Output: $\mathcal{A}$ outputs $T^* = (\mathsf{P}(\mathsf{G}^*), \Sigma^*, k)$ and wins the game if the tuple $(\mathsf{P}(\mathsf{G}^*), \Sigma^*, k)$ passes the verification algorithm. In this case, we have

$$\Sigma^* = h^{sk^{*-1} \cdot a^{n-k} \cdot \prod_{l_i \in \mathsf{G}^*}(a + l_i)}.$$

Let $f(x) = sk^{*-1} \cdot x^{n-k} \cdot \prod_{l_i \in \mathsf{G}^*} (x + l_i)$. We have $f(x)$ is a polynomial function in $\mathbb{Z}_p$ with degree larger than n. $\mathcal{B}$ outputs $(f(x), \Sigma^*)$ as the solution to the $(\hat{f}, n)$-DHE problem. This completes the description of the simulation and we obtain the **Theorem** 4.

6. Conclusion

We proposed an efficient two-round OT_n^k scheme with ideal communication cost. Precisely, the receiver sends only two group elements and one integer to the sender which are independent of the number of secrets held by the sender and the number of receiver's choice, while the number of messages from S to R is $n+1$. Compared with the existing OT_n^k schemes in the literature, our scheme achieves the minimum communication cost in both two rounds. We show that our OT_n^k scheme is unconditionally secure for receiver's choice and preserves the sender security under the specific $(f, \hat{g}, F)$-GDDHE assumption and $(\hat{f}, n)$-DHE assumption.

References

[1] G. Asharov, Y. Lindell, T. Schneider, and M. Zohner. More efficient oblivious transfer and extensions for faster secure computation. In A. Sadeghi, V. D. Gligor, and M. Yung, editors, *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13*, pages 535–548. ACM, 2013.

[2] G. Asharov, Y. Lindell, T. Schneider, and M. Zohner. More efficient oblivious transfer extensions with security for malicious adversaries. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015*, volume 9056 of *LNCS*, pages 673–701. Springer, 2015.

[3] D. Boneh, X. Boyen, and E. Goh. Hierarchical identity based encryption with constant size ciphertext. In R. Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 440–456. Springer, 2005.

[4] G. Brassard, C. Crépeau, and J. Robert. All-or-nothing disclosure of secrets. In A. M. Odlyzko, editor, *CRYPTO 1986*, volume 263 of *LNCS*, pages 234–238. Springer, 1987.

[5] J. Camenisch, G. Neven, and A. Shelat. Simulatable adaptive oblivious transfer. In M. Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 573–590. Springer, 2007.

[6] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan. Private information retrieval. *Journal of the ACM*, 45(6):965–981, 1998.

[7] C. Chu and W. Tzeng. Efficient k-out-of-n oblivious transfer schemes. *J. UCS*, 14(3):397–415, 2008.

[8] C.-K. Chu and W.-G. Tzeng. Efficient k-out-of-n oblivious transfer schemes with adaptive and non-adaptive queries. In S. Vaudenay, editor, *PKC 2005*, volume 3386 of *LNCS*, pages 172–183. Springer, 2005.

[9] C. Delerablée. Identity-based broadcast encryption with constant size ciphertexts and private keys. In K. Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 200–215. Springer, 2007.

[10] C. Delerablée, P. Paillier, and D. Pointcheval. Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys. In T. Takagi, T. Okamoto, E. Okamoto, and T. Okamoto, editors, *Pairing 2007*, volume 4575 of *LNCS*, pages 39–59. Springer, 2007.

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

[12] O. Goldreich and R. Vainish. How to solve any protocol problem - an efficiency improvement. In C. Pomerance, editor, *CRYPTO 1987*, volume 293 of *LNCS*, pages 73–86. Springer, 1988.

[13] M. Green and S. Hohenberger. Blind identity-based encryption and simulatable oblivious transfer. In K. Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 265–282. Springer, 2007.

[14] F. Guo, Y. Mu, W. Susilo, and V. Varadharajan. Membership encryption and its applications. In C. Boyd and L. Simpson, editors, *Information Security and Privacy - 18th Australasian Conference, ACISP 2013*, volume 7959 of *LNCS*, pages 219–234. Springer, 2013.

[15] F. Guo, Y. Mu, and W. Susilo. Subset membership encryption and its applications to oblivious transfer. *IEEE Transactions on Information Forensics and Security*, 9(7):1098–1107, 2014.

[16] Y. Huang, J. Katz, V. Kolesnikov, R. Kumaresan, and A. J. Malozemoff. Amortizing garbled circuits. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014*, volume 8617 of *LNCS*, pages 458–475. Springer, 2014.

[17] Y. Ishai, J. Kilian, K. Nissim, and E. Petrank. Extending oblivious transfers efficiently. In D. Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 145–161. Springer, 2003.

[18] V. Kolesnikov and R. Kumaresan. Improved OT extension for transferring short secrets. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013*, volume 8043 of *LNCS*, pages 54–70. Springer, 2013.

[19] V. Kolesnikov and R. Kumaresan. On cut-and-choose oblivious transfer and its variants. In T. Iwata and J. H. Cheon, editors, *ASIACRYPT 2015*, volume 9452 of *LNCS*, pages 386–412. Springer, 2015.

[20] K. Kurosawa, R. Nojima, and L. T. Phong. Generic fully simulatable adaptive oblivious transfer. In J. Lopez and G. Tsudik, editors, *ACNS 2011*, volume 6715 of *LNCS*, pages 274–291, 2011.

[21] E. Larraia. Extending oblivious transfer efficiently - or - how to get active security with constant cryptographic overhead. In D. F. Aranha and A. Menezes, editors, *LATINCRYPT 2014*, volume 8895 of *LNCS*, pages 368–386. Springer, 2015.

[22] A. Y. Lindell. Efficient fully-simulatable oblivious transfer. In T. Malkin, editor, *CT-RSA 2008*, volume 4964 of *LNCS*, pages 52–70. Springer, 2008.

[23] Y. Lindell. Fast cut-and-choose based protocols for malicious and covert adversaries. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013*, volume 8043 of *LNCS*, pages 1–17. Springer, 2013.

[24] Y. Lindell and B. Pinkas. An efficient protocol for secure two-party computation in the presence of malicious adversaries. In M. Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 52–78. Springer, 2007.

[25] Y. Lindell and B. Pinkas. Secure two-party computation via cut-and-choose oblivious transfer. In Y. Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 329–346. Springer, 2011.

[26] Y. Lindell and B. Riva. Cut-and-choose yao-based secure computation in the online/offline and batch settings. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014*, volume 8617 of *LNCS*, pages 476–494. Springer, 2014.

[27] Y. Mu, J. Zhang, and V. Varadharajan. m out of n oblivious transfer. In L. M. Batten and J. Seberry, editors, *Information Security and Privacy, 7th Australian Conference, ACISP 2002*, volume 2384 of *LNCS*, pages 395–405. Springer, 2002.

[28] M. Naor and B. Pinkas. Oblivious transfer with adaptive queries. In M. J. Wiener, editor, *CRYPTO 1999*, volume 1666 of *LNCS*, pages 573–590. Springer, 1999.

[29] M. Naor and B. Pinkas. Efficient oblivious transfer protocols. In S. R. Kosaraju, editor, *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms*, pages 448–457. ACM/SIAM, 2001.

[30] J. B. Nielsen, P. S. Nordholt, C. Orlandi, and S. S. Burra. A new approach to practical active-secure two-party computation. In R. Safavi-Naini and R. Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 681–700. Springer, 2012.

[31] W. Ogata and K. Kurosawa. Oblivious keyword search. *Journal of Complexity*, 20(2-3):356–371, 2004.

[32] M. O. Rabin. How to exchange secrets by oblivious transfer. *Technical Report TR-81, Aiken Computation Laboratory, Harvard University*, 1981.

[33] B. Zhang, H. Lipmaa, C. Wang, and K. Ren. Practical fully simulatable oblivious transfer with sublinear communication. In A. Sadeghi, editor, *Financial Cryptography and Data Security - 17th International Conference, FC 2013*, volume 7859 of *LNCS*, pages 78–95. Springer, 2013.

[34] J. Zhang and Y. Wang. Two provably secure k-out-of-n oblivious transfer schemes. *Applied Mathematics and Computation*, 169(2):1211–1220, 2005.