Hindawi
Security and Communication Networks
Volume 2022, Article ID 9245516, 12 pages
https://doi.org/10.1155/2022/9245516

Research Article

Multiparty Threshold Private Set Intersection Protocol with Low Communication Complexity

Xiaopeng Yu,[1] Fagen Li,[2] Wei Zhao,[1] Zhengyi Dai,[3] and Dianhua Tang[1,2]

[1] *Science and Technology on Communication Security Laboratory, Chengdu 610041, China*
[2] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*
[3] *College of Computer, National University of Defense Technology, Changsha 410073, China*

Correspondence should be addressed to Dianhua Tang; tangdianhua86@163.com

Received 25 April 2022; Accepted 13 August 2022; Published 27 September 2022

Academic Editor: Arijit Karati

Multiparty threshold private set intersection (MP-TPSI) protocol allows n mutually untrusted parties $P_1, P_2, \ldots, P_n$ holding data sets $A_1, A_2, \ldots, A_n$ of size m respectively to jointly compute the intersection $I = A_1 \cap A_2 \cap \cdots \cap A_n$ over all their private data sets only if the size of intersection is larger than $(m - t)$, while ensuring that no other private information of the data sets other than the intersection is revealed, where t is the threshold. In the MP-TPSI protocol, multiple parties first decide whether the size of the intersection is larger than the threshold t; then, they compute the intersection if the size of the intersection is larger than the threshold t. However, the existing MP-TPSI protocols use different forms of evaluation polynomials in the cardinality testing and intersection computing phases, so that parties need to transmit and calculate a large number of evaluation values, which leads to high communication and computational complexity. In addition, the existing MP-TPSI protocols cannot guarantee the security and the correctness of the results, that is, an adversary can know the additional information beyond the intersection, and the elements that are not in the intersection are calculated as the intersection. To solve these issues, based on the threshold fully homomorphic encryption (TFHE) and sparse polynomial interpolation, we propose an MP-TPSI protocol. In the star network topology, the theoretical communication complexity of the proposed MP-TPSI protocol depends on the threshold t and the number of parties n, not on the size of set m. Moreover, the proposed MP-TPSI protocol outperforms other related MP-TPSI protocols in terms of computational and communication overheads. Furthermore, the proposed MP-TPSI protocol tolerates up to $n - 1$ corrupted parties in the semi-honest model, where no set of colluding parties can learn the input of an honest party in the strictest dishonest majority setting.

1. Introduction

The private set intersection (PSI) protocol [1] allows two mutually untrusted parties P_1 and P_2 holding data sets A_1 and A_2 to jointly compute the set intersection $I = A_1 \cap A_2$, and does not reveal anything except the intersection. PSI protocol has a large number of application scenarios, e.g., DNA matching [2], botnet detection [3], and private contact discovery [4]. Over the past few decades, in the semi-honest and malicious security model, a long line of work [5–23] has been made to effectively implement the PSI protocol. The main cryptographic primitives of the existing PSI protocols include: garbled circuits (GC) [24], oblivious transfer (OT) [25], homomorphic encryption (HE) [26] and pseudorandom functions (PRF) [27], etc. To support PSI among multiple parties, several multiparty PSI (MP-PSI) protocols [28–36] have been presented.

However, in certain application scenarios, such as vertical federated learning (VFL) [37], the MP-PSI protocol mentioned above cannot satisfy the requirements. Specifically, in vertical federated machine learning, the training data is distributed among multiple parties, and each party has different features of the same object, multiple parties want to combine different features of common samples to train a better machine learning model. It is worth noting that all parties are willing to perform multiparty entity alignment only when the number of sample intersection is large. If the number of sample intersection is too small, the sample alignment will have no effect on improving the performance of the model, and the parties will not be interested in jointly computing the intersection of training samples. To meet such demands to determine whether the size of intersection is large enough before performing sample alignment, the multiparty threshold private set intersection (MP-TPSI) protocols [38–41] have been introduced, which enables n mutually distrusted parties $P_1, P_2, \ldots, P_n$ holding data sets $A_1, A_2, \ldots, A_n$ of size m respectively to jointly compute the intersection over all their private data sets only if the size of intersection is larger than $(m - t)$, while ensuring that no other private information of the data sets other than the intersection is revealed. The MP-TPSI protocol consists of two phases: the cardinality testing phase, where multiple parties decide whether the size of intersection is larger than a certain threshold t; and the intersection computing phase, where multiple parties calculate the intersection if the size of intersection is larger than a certain threshold t. Unfortunately, the existing MP-TPSI protocols [38–41] still have the heavy communication complexity. To solve this problem, using sparse polynomial interpolation and threshold fully homomorphic encryption (TFHE) [42], this paper proposes an MP-TPSI protocol with low communication complexity.

The main contributions are as follows:

(1) Firstly, in a star network topology where the designated party P_1 can communicate with each party P_i $(i = 2, 3, \ldots, n)$, using an evaluation method that represents the set as a polynomial, we construct an MP-TPSI protocol based on the TFHE. To reduce the communication and computational cost, we use the same form of evaluation polynomial in the cardinality testing and intersection computing phases, which enables the parties to transmit and compute only a small number of evaluation values.

(2) Secondly, in the proposed MP-TPSI protocol, the theoretical communication complexity of the designated party P_1 and each party P_i $(i = 2, 3, \ldots, n)$ are $\mathcal{O}(tn)$ and $\mathcal{O}(t)$, respectively, which are smaller than the existing MP-TPSI protocols [38–40] and TAHE-based MP-TPSI protocol [41]. In contrast to conventional MP-PSI protocols [28–36], the communication complexity of the proposed MP-TPSI protocol only depends on the threshold t and the number of parties n, not on the size of set m.

(3) Finally, we evaluate the proposed MP-TPSI protocol and the related TFHE-based MP-TPSI protocol [41] under $n \in \{2, 3, \cdots, 8\}$, $m \in \{2^{10}, 2^{11}, 2^{12}\}$, and $t \in \{2^9, 2^{10}, 2^{11}\}$. The experimental results demonstrate that, compared with the TFHE-based MP-TPSI protocol [41], the computational and communication costs in the proposed MP-TPSI protocol are reduced by nearly 92.0%–97.3% and 67.2%–67.3%, respectively. The security analysis illustrates that the proposed MP-TPSI protocol can achieve semi-honest security in the dishonest majority model where up to $n - 1$ parties can be allowed to corrupt.

The remainder of the study is organized as follows. We introduce some related works in Section 2. In Section 3, we review some preliminaries. In Section 4, our protocol is described in detail. The performance evaluation of our protocol is presented in Section 5. The security analysis of our protocol is shown in Section 6. Finally, we conclude in Section 7.

2. Related Works

Some works [28–36, 38–41] closely related to this study are introduced in this section. For ease of description, we summarize the theoretical communication complexity of [28–36, 38–41] in Table 1.

By representing the set as a polynomial, based on threshold additive HE (TAHE) that can be realized from Paillier encryption [43], Kissner et al. [28] implement the PSI operations in multiparty setting. Leveraging the Bloom filters (BF) [44] and exponential additive HE (AHE) [45], Miyaji et al. [29] presented a scalable MP-PSI protocol, they set a dealer to decrease the computational complexity of the parties. In a star network topology, based on the two-party protocol of [46], Hazay et al. [30] described the MP-PSI protocols in semi-honest and malicious settings. Kolesnikov et al. [31] proposed a method called oblivious programmable PRF (OPPRF), designed MP-PSI protocols based OPPRF in the semi-honest model, and further optimized it to the augmented-semi-honest model. Inbar et al. [32] extend the PSI construction of [12] to multiparty setting, and described the MP-PSI protocols for semi-honest and augmented-semi-honest settings in a star network topology. Setting the elements of its own set to the roots of a polynomial, based on the OLE, in a star network topology, Ghosh et al. [33] presented an approach to achieving secure MP-PSI. Lu et al. [34] proposed an MP-PSI protocol for VFL in a star network topology, which is able to compute the intersection in the event that some of the parties are offline. Combining of the star and path communication patterns which in the former, one party at the center can communicate with all other parties, and in the latter, each party can communicate with neighboring parties, Kavousi et al. [35] presented an efficient protocol for MP-PSI using oblivious PRF (OPRF). Based on the TAHE schemes and BF, in a star network topology, Bay et al. [36] proposed an MP-PSI protocol, which is secure in the semi-honest model. However, the communication and computational complexity of the MPSI protocol [28–36] mentioned above depend on the size of the input data set, which directly becomes a basic obstacle to efficiency.

Based on the AHE, Ghosh et al. [38] introduced an MP-TPSI protocol, which is the first MP-TPSI protocol with communication complexity that depend on threshold t, not on the set size m. However, Abadi et al. [47] pointed out that [38]'s protocol is not secure because an adversary can learn other information about the sets of honest parties beyond the intersection. Using the OPRF and hash function, Mahdavi et al. [39] introduced two constructions for the MP-TPSI protocol, namely $t - \mathrm{PSI}_0$ and $t - \mathrm{PSI}$, but the computational complexity is exponential in the threshold t,

Table 1: Theoretical communication complexity comparison.

Protocols	Communication complexity		Security model
	Designated party P_1	Party P_i $(i = 2, 3, \ldots, n)$	
[28]	$\mathcal{O}(mn)$	$\mathcal{O}(mw)$	Semi-honest
[29]	$\mathcal{O}(mn)$	$\mathcal{O}(m)$	Semi-honest
[30]	$\mathcal{O}(mn)$	$\mathcal{O}(m)$	Semi-honest
[30]	$\mathcal{O}((m + m\log m + n)n)$	$\mathcal{O}(m + m\log m + n)$	Malicious
[31]	$\mathcal{O}(mn)$	$\mathcal{O}(mn)$	Semi-honest
[31]	$\mathcal{O}(mn)$	$\mathcal{O}(m)$	Augmented-semi-honest
[32]	$\mathcal{O}(mnh)$	$\mathcal{O}(mnh)$	Semi-honest
[32]	$\mathcal{O}(mnh)$	$\mathcal{O}(mh)$	Augmented-semi-honest
[33]	$\mathcal{O}(n^2 + mn)$	$\mathcal{O}(m)$	Semi-honest
[34]	$\mathcal{O}(mn)$	$\mathcal{O}(mq)$	Semi-honest
[35]	$\mathcal{O}(mn)$	$\mathcal{O}(mk)$	Semi-honest
[36]	$\mathcal{O}(mn)$	$\mathcal{O}(m)$	Semi-honest
[38]	$\mathcal{O}(t^2n)$	$\mathcal{O}(t^2)$	Semi-honest
[39]	$\mathcal{O}(mnw)$	$\mathcal{O}(mnw)$	Semi-honest $(t - \mathrm{PSI}_0)$
[39]	$\mathcal{O}(tmnc)$	$\mathcal{O}(tmnc)$	Semi-honest t – PSI
[40]	$\mathcal{O}(t^2n)$	$\mathcal{O}(t^2)$	Semi-honest
[41]	$\mathcal{O}(t^2n)$	$\mathcal{O}(t^2)$	Semi-honest (TAHE-based)
[41]	$\mathcal{O}(tn)$	$\mathcal{O}(t)$	Semi-honest (TFHE-based)

and thus have a poor performance. By employing the TAHE from Elgamal encryption [48] and Paillier encryption [43], Branco et al. [40] developed a protocol to securely compute linear algebra functions and proposed an MP-TPSI in a star network topology. Badrinarayanan et al. [41] pointed out that [38]'s protocol has a subtle issue, that is, elements that are not in the intersection may also be computed as elements in the intersection. To solve this issue, in the star network topology, they proposed the TAHE-based MP-TPSI and TFHE-based MP-TPSI protocols. However, their TFHE-based MP-TPSI protocol uses different forms of evaluation polynomials in the cardinality testing and intersection computing phases, which requires the transmission and calculation of a large number of evaluation values, and brings to heavy communication and computational cost.

3. Preliminaries

3.1. Notations. For ease of reading, the definitions of symbols in the proposed MP-TPSI protocol are described in Table 2.

Table 2: The definitions of the symbols.

Symbols	Definitions
P_i	The i-th party
A_i	The data set of party P_i
I	The intersection of $A_1, A_2, \cdots, A_n$
$\lvert I\rvert$	The size of the intersection I
n	The number of parties
m	The size of each set A_i
t	The threshold
x	The ciphertext of plaintext x
x_i	The partial decryption ciphertext of x
λ	The security parameter
$\mathrm{negl}(\lambda)$	The negligible function on λ

3.2. Security Model. We define the security of the proposed MP-TPSI protocol in universal composability (UC) framework [49]. Considering a multiparty protocol Π that realizes the ideal functionality $\mathcal{F}$, we can define the security of the protocol Π in the ideal/real world.

In an ideal world: n parties transmit all inputs to $\mathcal{F}$, and receive the computation result. Simulator $\mathcal{S}$ is regarded as an adversary in an ideal world, has complete control of the parties that are corrupted, and simulates $\mathcal{Z}$'s view of on the execution of the real protocol.

In a real world: n parties perform Π, Π is permitted to call an ideal functionality $\mathcal{G}$. Environment $\mathcal{Z}$ selects all inputs of n parties, simulates anything outside Π. $\mathcal{Z}$ can represent the adversary and corrupt any subset of the parties.

Assuming $\mathrm{Ideal}[\mathcal{Z}, \mathcal{S}, \mathcal{F}]$ and $\mathrm{Real}[\mathcal{Z}, \Pi, \mathcal{G}]$ are the output of $\mathcal{Z}$ in the ideal and real world, respectively, we define Π securely realizes $\mathcal{F}$, if there is a $\mathcal{S}$ so that for any $\mathcal{Z}$ we have

$$\lvert \Pr[\mathrm{Ideal}[\mathcal{Z}, \mathcal{S}, \mathcal{F}] = 1] - \Pr[\mathrm{Real}[\mathcal{Z}, \Pi, \mathcal{G}] = 1]\rvert \leq \mathrm{negl}(\lambda). \quad (1)$$

3.3. The definition of Threshold Fully Homomorphic Encryption. A TFHE scheme [42] consists of the distributed setup (TFHE.DisSet), encryption (TFHE.Enc), addition (TFHE.Add), multiplication (TFHE.Mul), partial decryption (TFHE.PartDec), and combination (TFHE.Comb) algorithms.TFHE.DisSet$(1^\lambda, i) \longrightarrow (\mathrm{pk}, \mathrm{sk}_i)$: On input λ and party's number i, TFHE.DisSet algorithm returns the secret key share sk_i and public key pk for the party P_i.TFHE.Enc$(\mathrm{pk}, x) \longrightarrow x$: On input pk and plaintext x, TFHE.Enc algorithm returns the ciphertext x.TFHE.Add$(x_1, x_2) \longrightarrow x_1 + x_2$: On input the ciphertexts x_1 and x_2, TFHE.Add algorithm outputs the ciphertext $x_1 + x_2$.TFHE.Mul$(x_1, x_2) \longrightarrow x_1 * x_2$: On input the ciphertexts x_1 and x_2, TFHE.Mul algorithm outputs the

Parameters: Each party P_i holds a data set A_i with m elements, and sets a threshold $T \in \mathbb{N}$, where $i \in \{1, 2, \cdots, n\}$.
Inputs: Each party P_i inputs a data set $A_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,m}\}$, where $a_{i,j} \in \mathbb{F}_p$ for each $j = 1, 2, \cdots, m$, $\mathbb{F}_p$ is a finite field.
Outputs: If $|A_i \backslash I| \leq T$, each party P_i outputs $true$, otherwise outputs $false$.

Figure 1: Ideal functionality $\mathscr{F}_{\text{MP-TPSI-CT}}$ for MP-TPSI cardinality testing.

Parameters: Each party P_i holds a data set A_i with m elements, and sets a threshold $T \in \mathbb{N}$, where $i \in \{1, 2, \cdots, n\}$.
Inputs: Each party P_i inputs a data set $A_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,m}\}$, where $a_{i,j} \in \mathbb{F}_p$ for each $j = 1, 2, \cdots, m$, $\mathbb{F}_p$ is a finite field.
Outputs: Each party P_i outputs intersection $I = A_1 \cap A_2 \cap \cdots \cap A_n$ or none $\perp$.

Figure 2: Ideal functionality $\mathscr{F}_{\text{MP-TPSI-C}}$ for MP-TPSI computing.

ciphertext $x_1 * x_2$.TFHE.PartDec$(\text{sk}_i, y) \longrightarrow y_i$: On input the secret key share sk_i and ciphertext y, TFHE.PartDec algorithm outputs the partial decryption ciphertext y_i.TFHE.Comb$(y_1, y_2, \cdots, y_n) \longrightarrow y$: On input a set of partial decryption ciphertexts $y_1, y_2, \cdots, y_n$, TFHE.Comb algorithm outputs the plaintext $y = y_1 + y_2 + \cdots + y_n$.

3.4. Functionality. Ideal functionality $\boldsymbol{\mathscr{F}}_{\textbf{MP-TPSI-CT}}$ *for MP-TPSI cardinality testing*: In a star network topology, for n parties $\text{P}_1, \text{P}_2, \cdots, \text{P}_n$ holding data sets $A_1, A_2, \cdots, A_n$ of equal size m, respectively, the goal of the $\mathscr{F}_{\text{MP-TPSI-CT}}$ is to execute a multiparty protocol Π, at the end of Π, every party P_i can know whether if its data set A_i and intersection $I = A_1 \cap A_2 \cap \cdots \cap A_n$ differ by at most t, namely $|I| \geq m - t$. The formal definition of $\mathscr{F}_{\text{MP-TPSI-CT}}$ is depicted in Figure 1.

Ideal functionality $\boldsymbol{\mathscr{F}}_{\textbf{MP-TPSI-C}}$ *for MP-TPSI computing*: In a star network topology, for n parties $\text{P}_1, \text{P}_2, \cdots, \text{P}_n$ holding data sets $A_1, A_2, \cdots, A_n$ of equal size m, respectively, the goal of the $\mathscr{F}_{\text{MP-TPSI-C}}$ is to execute an multiparty protocol Π, at the end of Π, either every party P_i outputs an intersection $I = A_1 \cap A_2 \cap \cdots \cap A_n$ or outputs none $\perp$. The formal definition of $\mathscr{F}_{\text{MP-TPSI-C}}$ is described Figure 2.

4. Multiparty Threshold Private Set Intersection

In a star network topology where party P_1 to be the designated party that can communicate with other parties $\text{P}_2, \text{P}_3, \cdots, \text{P}_n$, suppose n, parties $\text{P}_1, \text{P}_2, \cdots, \text{P}_n$ with input sets $A_1, A_2, \cdots, A_n$ of equal size m, respectively, based on TFHE with distributed setup, we propose an MP-TPSI protocol, in which each party P_i can compute the intersection $I = A_1 \cap A_2 \cap \cdots \cap A_n$ only if $|I| \geq m - t$. The proposed MP-TPSI protocol is formally described in Figure 3.

4.1. Correctness. MP-TPSI cardinality testing: First we consider the situation where the MP-TPSI cardinality testing outputs true. Based on the correctness of the TFHE, we only need to illustrate $b = 0$ only if $|A_i/I| \leq t$ for any $i = 1, 2, \cdots, n$. Observe the rational interpolation polynomial

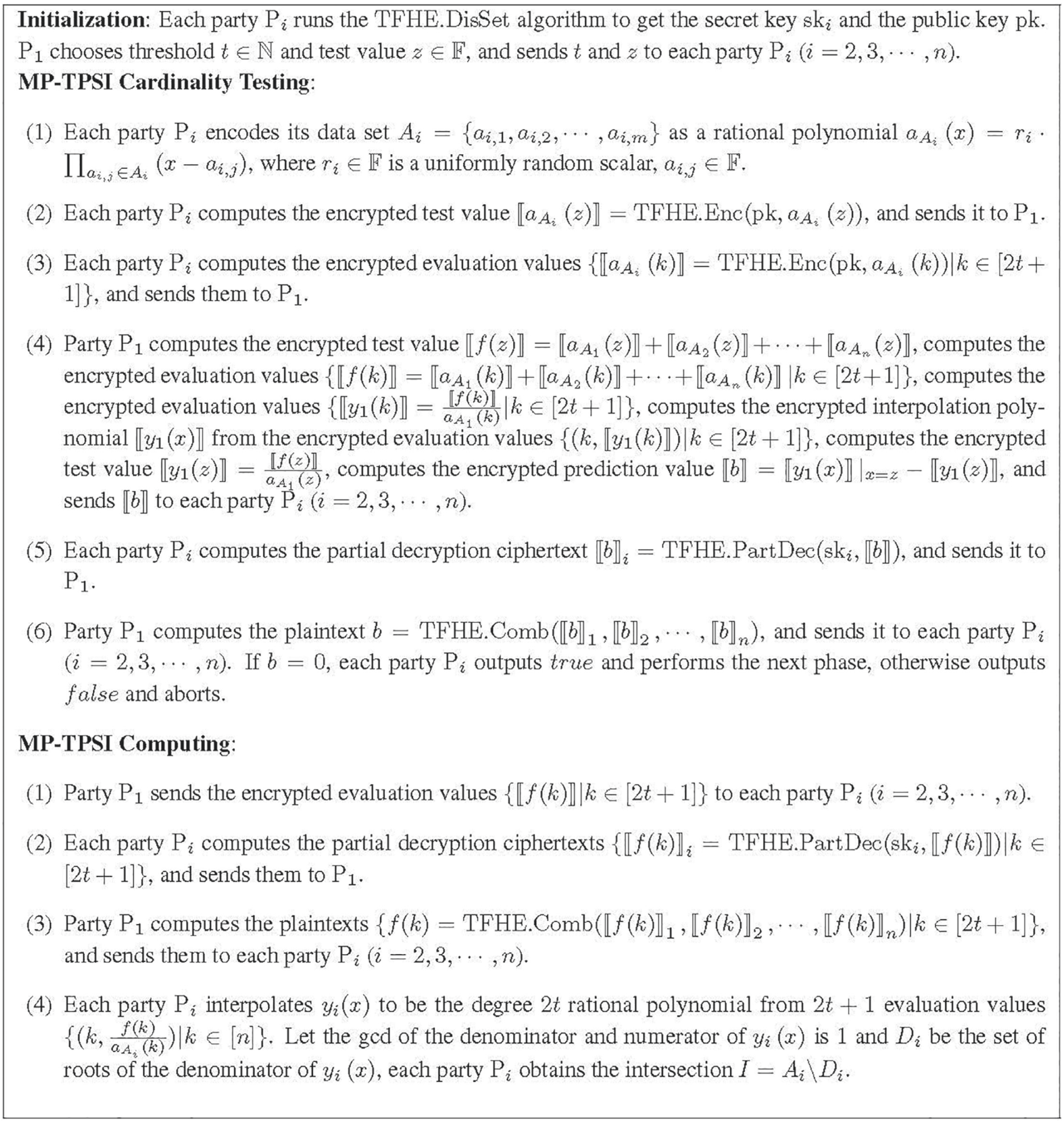
$$\begin{aligned} y_1(x) &= \frac{a_{A_1}(x) + a_{A_2}(x) + \cdots + a_{A_n}(x)}{a_{A_1}(x)} \\ &= \frac{a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x)}{a_{A_1 \backslash I}(x)} \\ &= \frac{\sum_{i=1}^{n} \left(r_i \cdot \prod_{a_{i,j} \in A_i \backslash I} \left(x - a_{i,j} \right) \right)}{r_1 \cdot \prod_{a_{1,j} \in A_1 \backslash I} \left(x - a_{1,j} \right)}, \end{aligned} \tag{2}$$

we can see that the degree of numerator $a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x)$ and denominator $a_{A_1 \backslash I}(x)$ is at most t, and the degree of rational polynomial $y_1(x)$ is at most $2t$. Therefore, $y_1(x)$ can be computed from a total of $2t + 1$ evaluation values, and the equation $y_1(x)|_{x=z} = f(z)/a_{A_1}(z) = a_{A_1}(z) + a_{A_2}(z) + \cdots + a_{A_n}(z)/ \ a_{A_1}(z)$ holds. Next, we consider the situation where the MP-TPSI cardinality testing outputs false. From the above equation, we can observe that $\gcd(a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x), a_{A_1 \backslash I}(x)) = 1$. Since $|A_i \backslash I| \geq (t + 1)$, the degree of $a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x)$ and $a_{A_1 \backslash I}(x)$ are at least $t + 1$, the degree of rational polynomial $y_1(x)$ is at least $2t + 3$, and hence calculating $y_1(x)$ requires at least $2t + 3$ evaluation values. However, there are only $2t + 1$ evaluation values in the MP-TPSI cardinality testing. Therefore, the equation $y_1(x)|_{x=z} = f(z)/a_{A_1}(z) = a_{A_1}(z) + a_{A_2}(z) + \cdots + a_{A_n}(z)/a_{A_1}(z)$ does not hold. From the above analysis, we are able to obtain that the MP-TPSI cardinality testing is correct.

MP-TPSI computing: If $|A_i \backslash I| > t$ for any $i = 1, 2, \cdots, n$, the MP-TPSI computing quits after the MP-TPSI cardinality testing. If $|A_i \backslash I| \leq t$, observe the rational interpolation polynomial

$$\begin{aligned} y_i(x) &= \frac{a_{A_1}(x) + a_{A_2}(x) + \cdots + a_{A_n}(x)}{a_{A_i}(x)} \\ &= \frac{a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x)}{a_{A_i \backslash I}(x)} \\ &= \frac{\sum_{i=1}^{n} \left(r_i \cdot \prod_{a_{i,j} \in A_i \backslash I} \left(x - a_{i,j} \right) \right)}{r_i \cdot \prod_{a_{1,j} \in A_1 \backslash I} \left(x - a_{1,j} \right)}, \end{aligned} \tag{3}$$

we can see that the degree of numerator $a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x)$ and denominator $a_{A_i \backslash I}(x)$ are at most t, and hence $y_i(x)$ is a random polynomial with degree at most $2t + 1$. Since $\gcd(a_{A_1 \backslash I}(x) + a_{A_2 \backslash I}(x) + \cdots + a_{A_n \backslash I}(x), a_{A_i \backslash I}(x)) = 1$, no



Initialization: Each party P_i runs the TFHE.DisSet algorithm to get the secret key sk_i and the public key pk. P_1 chooses threshold $t \in \mathbb{N}$ and test value $z \in \mathbb{F}$, and sends t and z to each party P_i $(i = 2, 3, \cdots, n)$.

MP-TPSI Cardinality Testing:

(1) Each party P_i encodes its data set $A_i = \{a_{i,1}, a_{i,2}, \cdots, a_{i,m}\}$ as a rational polynomial $a_{A_i}(x) = r_i \cdot \prod_{a_{i,j} \in A_i} (x - a_{i,j})$, where $r_i \in \mathbb{F}$ is a uniformly random scalar, $a_{i,j} \in \mathbb{F}$.

(2) Each party P_i computes the encrypted test value $[\![a_{A_i}(z)]\!] = \mathrm{TFHE.Enc}(\mathrm{pk}, a_{A_i}(z))$, and sends it to P_1.

(3) Each party P_i computes the encrypted evaluation values $\{[\![a_{A_i}(k)]\!] = \mathrm{TFHE.Enc}(\mathrm{pk}, a_{A_i}(k)) | k \in [2t + 1]\}$, and sends them to P_1.

(4) Party P_1 computes the encrypted test value $[\![f(z)]\!] = [\![a_{A_1}(z)]\!] + [\![a_{A_2}(z)]\!] + \cdots + [\![a_{A_n}(z)]\!]$, computes the encrypted evaluation values $\{[\![f(k)]\!] = [\![a_{A_1}(k)]\!] + [\![a_{A_2}(k)]\!] + \cdots + [\![a_{A_n}(k)]\!] \, | k \in [2t+1]\}$, computes the encrypted evaluation values $\{[\![y_1(k)]\!] = \frac{[\![f(k)]\!]}{a_{A_1}(k)} | k \in [2t + 1]\}$, computes the encrypted interpolation polynomial $[\![y_1(x)]\!]$ from the encrypted evaluation values $\{(k, [\![y_1(k)]\!]) | k \in [2t + 1]\}$, computes the encrypted test value $[\![y_1(z)]\!] = \frac{[\![f(z)]\!]}{a_{A_1}(z)}$, computes the encrypted prediction value $[\![b]\!] = [\![y_1(x)]\!] \, |_{x=z} - [\![y_1(z)]\!]$, and sends $[\![b]\!]$ to each party P_i $(i = 2, 3, \cdots, n)$.

(5) Each party P_i computes the partial decryption ciphertext $[\![b]\!]_i = \mathrm{TFHE.PartDec}(\mathrm{sk}_i, [\![b]\!])$, and sends it to P_1.

(6) Party P_1 computes the plaintext $b = \mathrm{TFHE.Comb}([\![b]\!]_1, [\![b]\!]_2, \cdots, [\![b]\!]_n)$, and sends it to each party P_i $(i = 2, 3, \cdots, n)$. If $b = 0$, each party P_i outputs $true$ and performs the next phase, otherwise outputs $false$ and aborts.

MP-TPSI Computing:

(1) Party P_1 sends the encrypted evaluation values $\{[\![f(k)]\!] | k \in [2t + 1]\}$ to each party P_i $(i = 2, 3, \cdots, n)$.

(2) Each party P_i computes the partial decryption ciphertexts $\{[\![f(k)]\!]_i = \mathrm{TFHE.PartDec}(\mathrm{sk}_i, [\![f(k)]\!]) | k \in [2t + 1]\}$, and sends them to P_1.

(3) Party P_1 computes the plaintexts $\{f(k) = \mathrm{TFHE.Comb}([\![f(k)]\!]_1, [\![f(k)]\!]_2, \cdots, [\![f(k)]\!]_n) | k \in [2t + 1]\}$, and sends them to each party P_i $(i = 2, 3, \cdots, n)$.

(4) Each party P_i interpolates $y_i(x)$ to be the degree $2t$ rational polynomial from $2t + 1$ evaluation values $\{(k, \frac{f(k)}{a_{A_i}(k)}) | k \in [n]\}$. Let the gcd of the denominator and numerator of $y_i(x)$ is 1 and D_i be the set of roots of the denominator of $y_i(x)$, each party P_i obtains the intersection $I = A_i \backslash D_i$.

Figure 3: Multiparty threshold private set intersection.

other terms will be canceled out in the numerator and denominator. Therefore, based on the correctness of the TFHE, each party P_i is able to interpolate the rational random polynomial $y_i(x)$ by utilizing $2t + 1$ evaluation values. Finally, each P_i can easily compute intersection I from the set $A_i \backslash I$ of the roots of the denominator of polynomial $y_i(x)$.

5. Performance Evaluation

The proposed MP-TPSI protocol is an improvement of the TFHE-based MP-TPSI protocol [41], so we evaluate the proposed MP-TPSI protocol and the TFHE-based MP-TPSI protocol [41]. In the star network topology, we implement the proposed MP-TPSI protocol on top of the lattice-based multiparty HE library Lattigo [50] that implements the full-RNS BFV scheme [51] and its multiparty versions in Go. We run all experiments on a 32-core Intel Xeon CPU with 256 GB of RAM. For the multiparty BFV scheme in Go, to ensure 128 bits security, we choose that polynomial-degree is 4096, ciphertext-modulus is 109 bits, and plaintext-modulus is 17 bits. For ease of comparison, we perform all experiments on the same machine with 16 threads, emulate the networks latency by utilizing the Linux *tc* command, and consider a LAN with a 10 Gbps throughput and 0.2 ms round-trip time. It is worth noting that the authors of [41]

Table 3: Comparison of computational cost.

Set size	Threshold	Protocols	Computation cost (seconds)						
			$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$
$m = 2^{10}$	$t = 2^{9}$	[41]	395.48	410.51	437.93	489.15	555.43	641.92	792.88
		Ours	29.97	32.12	35.18	38.60	42.17	45.67	49.43
$m = 2^{11}$	$t = 2^{10}$	[41]	919.64	964.69	1011.61	1095.70	1277.49	1478.64	1821.63
		Ours	49.54	52.09	55.35	60.91	75.03	80.12	86.56
$m = 2^{12}$	$t = 2^{11}$	[41]	2815.10	3077.36	3307.96	3593.90	4177.39	4864.73	6011.38
		Ours	92.23	100.14	108.83	121.60	134.56	147.71	161.87

did not implement their TFHE-based MP-TPSI protocol, for a fair comparison, we implement the TFHE-based MP-TPSI protocol [41] in the same experimental environment.

5.1. Analysis of Computational Cost. The computational cost of the proposed MP-TPSI protocol and the TFHE-based MP-TPSI protocol [41] under $n \in \{2, 3, \cdots, 8\}$, $m \in \{2^{10}, 2^{11}, 2^{12}\}$, and $t \in \{2^{9}, 2^{10}, 2^{11}\}$ are shown in Table 3. All running times are shown as an average of 10 experiments.

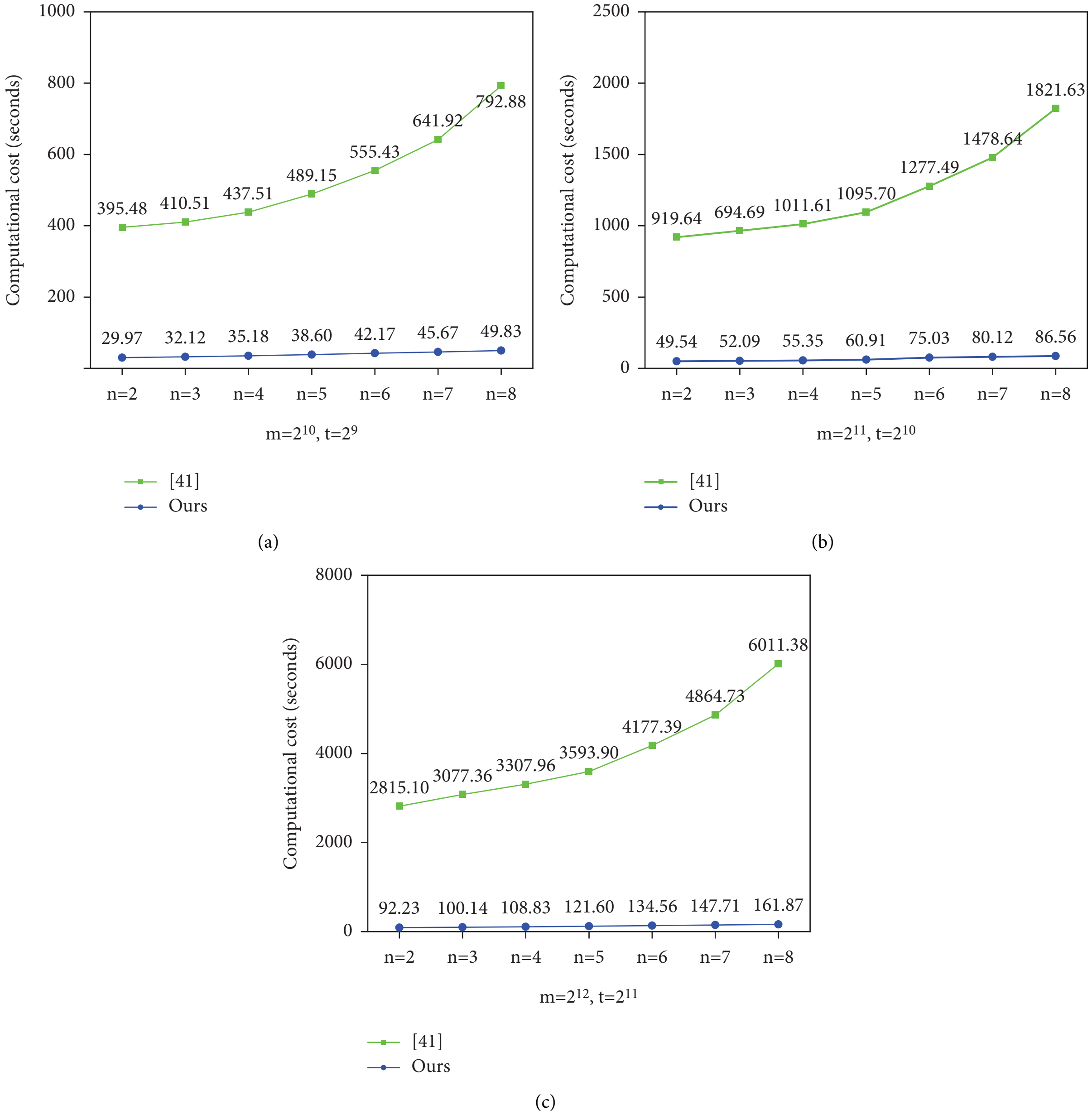
As shown in Figure 4, compared with the TFHE-based MP-TPSI protocol [41], the proposed MP-TPSI protocol has a better performance in terms of computational cost. Specifically, under $m = 2^{10}$ and $t = 2^{9}$, for $n \in \{2, 3, \cdots, 8\}$, the computational cost in the proposed MP-TPSI protocol is almost reduced by 92.4%, 92.2%, 92.0%, 92.1%, 92.4%, 92.9%, and 93.8%, respectively, compared with the TFHE-based MP-TPSI protocol [41]. Under $m = 2^{11}$ and $t = 2^{10}$, with regard to $n \in \{2, 3, \cdots, 8\}$, the proposed MP-TPSI protocol decreases by almost 94.6%, 94.6%, 94.5%, 94.4%, 94.1%, 94.6% and 95.2% respectively in computational cost in comparison with the TFHE-based MP-TPSI protocol [41]. Under $m = 2^{12}$ and $t = 2^{11}$, regarding $n \in \{2, 3, \cdots, 8\}$, the proposed MP-TPSI protocol reduces the computational cost by almost 96.7%, 96.7%, 96.7%, 96.6%, 96.8%, 97.0%, and 97.3%, respectively, than the TFHE-based MP-TPSI protocol [41].

5.2. Analysis of Communication Cost. In a star network topology, according to the selected parameters in Section 5.1, we can obtain the size of ciphertext, partial decryption ciphertext and plaintext are $|x| = 2 \times 4096 \times 109$ bits, $|x_i| = 4096 \times 109$ bits, and $|x| = 4096 \times 17$ bits, respectively. The comparison of communication cost between the proposed MP-TPSI protocol and the TFHE-based MP-TPSI protocol [41] are shown in Table 4.

For the TFHE-based MP-TPSI protocol [41], n parties first run the MPSI cardinality testing. Each P_i $(i = 2, 3, \cdots, n)$ sends $(2t + 3)$ ciphertexts $\{e_{i,j} | j \in [2t + 3]\}$ and one ciphertext e_i' to P_1. P_1 returns one ciphertext b to each P_i $(i = 2, 3, \cdots, n)$. Each P_i $(i = 2, 3, \cdots, n)$ sends one partial decryption ciphertext b: sk_i to P_1. P_1 returns one plaintext b to each P_i $(i = 2, 3, \cdots, n)$. If the MP-TPSI cardinality testing passes, n parties then run the MP-TPSI computing. Each P_i $(i = 2, 3, \cdots, n)$ sends $(3t + 4)$ ciphertexts $\{R_i(j) | j \in [3t + 4]\}$ to P_1. P_1 returns $(3t + 4)$ ciphertexts $\{e_{i,j} | j \in [3t + 3]\}$ to each P_i $(i = 2, 3, \cdots, n)$. Each P_i $(i = 2, 3, \cdots, n)$ sends $(3t + 4)$ ciphertexts $\{v_{i,j} | j \in [3t + 3]\}$ to P_1. P_1 returns $(3t + 4)$ ciphertexts $\{v_j | j \in [3t + 3]\}$ to each P_i $(i = 2, 3, \cdots, n)$. Each P_i $(i = 2, 3, \cdots, n)$ sends $(3t + 4)$ partial decryption ciphertexts $\{v_j\text{: } sk_i | j \in [3t + 4]\}$ to P_1. P_1 returns $(3t + 4)$ plaintexts $\{v_j | j \in [3t + 3]\}$ to each P_i $(i = 2, 3, \cdots, n)$. Therefore, the communication cost of the designated party P_1 is $(n - 1) \times (6t + 9) \times |x| + (n - 1) \times (3t + 5) \times |x|$ (namely, $\mathcal{O}(tn)$), the communication cost of each P_i $(i = 2, 3, \cdots, n)$ is $(8t + 12) \times |x| + (3t + 5) \times |x_i|$ (namely, $\mathcal{O}(t)$), and the total communication cost is $(n - 1) \times ((14t + 21) \times |x| + (3t + 5) \times |x_i| + (3t + 5) \times |x|)$ bits.

For our MP-TPSI protocol, n parties first run the MP-TPSI cardinality testing. Each P_i $(i = 2, 3, \cdots, n)$ sends $(2t + 1)$ ciphertexts $\{a_{A_i}(k) | k \in [2t + 1]\}$ and one ciphertext $a_{A_i}(z)$ to P_1. P_1 returns one ciphertext b to each P_i $(i = 2, 3, \cdots, n)$. Each P_i $(i = 2, 3, \cdots, n)$ sends one partial decryption ciphertext b_i to P_1. P_1 returns one plaintext b to each P_i $(i = 2, 3, \cdots, n)$. If the MP-TPSI cardinality testing passes, n parties then run the MP-TPSI computing. P_1 returns $(2t + 1)$ ciphertexts $\{f(k) | k \in [2t + 1]\}$ to each P_i $(i = 2, 3, \cdots, n)$. Each P_i $(i = 2, 3, \cdots, n)$ sends $(2t + 1)$ partial decryption ciphertexts $\{f(k)_i | k \in [2t + 1]\}$ to P_1. P_1 returns $(2t + 1)$ plaintexts $\{f(k) | k \in [2t + 1]\}$ to each P_i $(i = 2, 3, \cdots, n)$. Therefore, the communication cost of the designated party P_1 is $(n - 1) \times (2t + 9) \times |x| + (n - 1) \times (2t + 2) \times |x|$ (namely, $\mathcal{O}(tn)$), the communication cost of each P_i $(i = 2, 3, \cdots, n)$ is $(2t + 2) \times |x| + (2t + 2) \times |x_i|$ (namely, $\mathcal{O}(t)$), the total communication cost is $(n - 1) \times ((4t + 4) \times |x| + (2t + 2) \times |x_i| + (2t + 2) \times |x|)$ bits.

As shown in Figure 5, compared with the TFHE-based MP-TPSI protocol [41], the proposed MP-TPSI protocol has a better performance in terms of communication cost. Specifically, when comparing with $m = 2^{10}$ and $t = 2^{9}$, for $n \in \{2, 3, \cdots, 8\}$ parties, the communication cost in the proposed MP-TPSI protocol is almost reduced by 67.3%, 67.3%, 67.3%, 67.3%, 67.3%, 67.3%, and 67.3%, respectively, compared with the TFHE-based MP-TPSI protocol [41]. When comparing with $m = 2^{11}$ and $t = 2^{10}$, with regard to $n \in \{2, 3, \cdots, 8\}$ parties, the proposed MP-TPSI protocol decreases by almost 67.2%, 67.2%, 67.2%, 67.2%, 67.2%, 67.2%, and 67.2%, respectively, in communication cost in comparison with the TFHE-based MP-TPSI protocol [41]. When comparing with $m = 2^{12}$ and $t = 2^{11}$, regarding $n \in \{2, 3, \cdots, 8\}$ parties, the proposed MP-TPSI protocol reduces the communication cost by almost 67.2%, 67.2%, 67.2%, 67.2%, 67.2%, 67.2% and 67.2% respectively than the TFHE-based MP-TPSI protocol [41].



(a)

(b)

(c)

Figure 4: Comparison of computational cost.

Table 4: Comparison of communication costs.

Set size	Threshold	Protocols	Communication cost (GB)						
			$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$
$m = 2^{10}$	$t = 2^9$	[41] Ours	0.84 0.27	1.68 0.55	2.52 0.82	3.36 1.10	4.20 1.37	5.04 1.65	5.88 1.92
$m = 2^{11}$	$t = 2^{10}$	[41] Ours	1.68 0.55	3.35 1.10	5.03 1.65	6.71 2.20	8.39 2.75	10.06 3.30	11.74 3.85
$m = 2^{12}$	$t = 2^{11}$	[41] Ours	3.35 1.10	6.70 2.20	10.06 3.29	13.41 4.39	16.76 5.49	20.11 6.59	23.46 7.69

6. Security Analysis

In security model, we assume an environment $\mathscr{Z}$ who is able to corrupt the set $\mathscr{A}^*$ of $n^* < n$ parties, a simulator $\mathscr{S}$ knows the output value $w \in \{\text{true}, \text{false}\}$ of the ideal functionality $\mathscr{F}_{\text{MP-TPSI-CT}}$. If $w = \text{true}$, $\mathscr{S}$ sets $b = 0$, otherwise sets $b = 1$. $\mathscr{S}$ also has the output value I or $\perp$ of the ideal functionality $\mathscr{F}_{\text{MP-TPSI-C}}$. In addition, for each corrupt party $\mathscr{A}_i \in \mathscr{A}^*$, $\mathscr{S}$

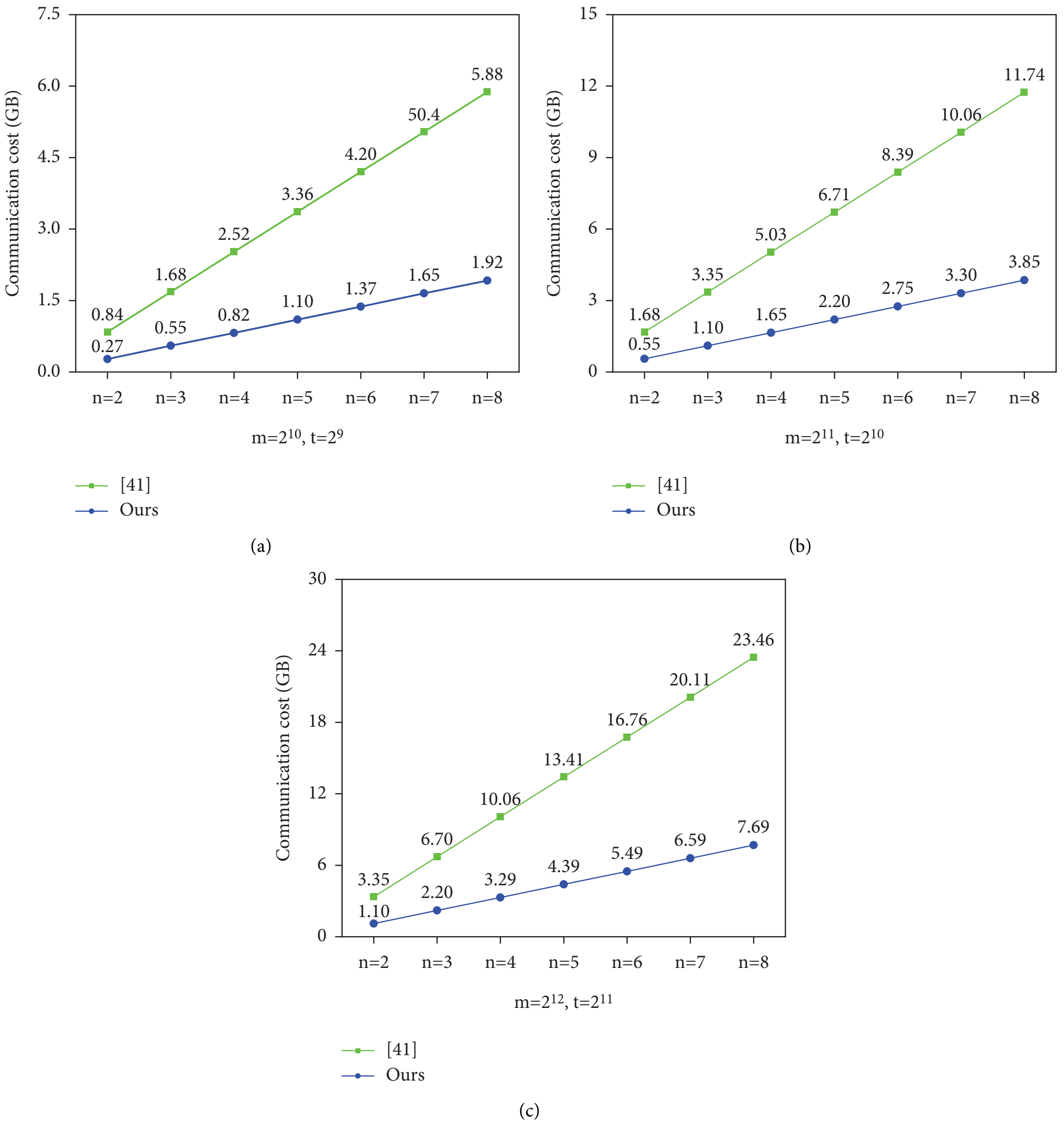


Figure 5: Comparison of communication cost.

has the input data set A_i and random value r_i of $\mathscr{A}_i$. The simulation strategy of $\mathcal{S}$ is described as follows.

Initialization. $\mathcal{S}$ represents each honest party P_i running the distributed setup TFHE.DisSet algorithm just like in the real world. $\mathcal{S}$ also knows the secret key share $\{sk_i\}_{\mathscr{A}_i \in \mathscr{A}^*}$ of all corrupt parties $\mathscr{A}^*$.

MP-TPSI Cardinality Testing. $\mathcal{S}$ does the following:

In Step 1, $\mathcal{S}$ encodes the intersection set $I = \{a_1, a_2, \cdots, a_I\}$ as a rational polynomial $a_I(x) = \prod_{a_i \in I}(x - a_i)$, chooses randomly a rational polynomial $u(x)$ of degree t, and computes a rational polynomial $f(x) = a_I(x) \cdot u(x)$.

In Steps 2–4, whenever each honest party P_i sends any encrypted value, $\mathcal{S}$ computes the ciphertext $0 = \mathrm{TFHE.Enc}(0)$ employing fresh random value on behalf of P_i just like in the real world.

In Steps 5–6, instead of computing the value b_i by executing the partial decryption algorithm $\mathrm{TFHE.PartDec}(sk_i, b)$ on behalf of every honest party P_i just like in the real world, $\mathcal{S}$ calculates the value b_i by executing the simulator algorithm $\{b_i\}_{\mathrm{P}_i \in \mathscr{P}} = \mathrm{TFHE}.\mathcal{S}(\mathscr{C}, b, b, \{sk_i\}_{\mathscr{A}_i \in \mathscr{A}^*})$, where $\mathscr{C}$ represents the computation circuit performed by P_1 to calculate the value b just like in the real world, this corresponds to the ideal world, $\mathscr{P}$ denotes the set of the honest parties. If P_1 is honest, $\mathcal{S}$ sends the evaluation value b just like in the real world.

MPSI Computing. $\mathcal{S}$ does the following:

In steps 1, instead of computing the value $f(k)_i$ by executing the partial decryption algorithm TFHE.PartDec $(sk_i, f(k))$ on behalf of every honest party P_i just like in the real world, $\mathcal{S}$ calculates the value $f(k)_i$ by executing the simulator algorithm $\{f(k)_i\}_{\mathrm{P}_i\in\mathcal{P}} = \mathrm{TFHE}.\mathcal{S}(\mathcal{C}, f(k), f(k)_i, \{sk_i\}_{\mathcal{A}_i\in\mathcal{A}^*})$, where $\mathcal{C}$ represents the computation circuit performed by P_1 to calculate the value $f(k)$ just like in the real world, this corresponds to the ideal world. If P_1 is honest, $\mathcal{S}$ sends the evaluation value $f(k)$ just like in the real world.

In steps 2, $\mathcal{S}$ outputs the interpolation polynomial $y_i(x)$ and set intersection I on behalf of every honest party P_i just like in the real world.

Next, suppose a simulator $\mathcal{S}_h$, we show that the proposed MP-TPSI protocol is secure against the environment $\mathcal{Z}$ in the semi-honest setting through a set of computationally indistinguishable consecutive hybrids.

Hybrid$_0$: $\mathcal{S}_h$ simulates all operations of honest parties just like in the real world.

Hybrid$_1$: $\mathcal{S}_h$ simulates a ideal functionality $\mathcal{F}_{\mathrm{MP-TPSI-CT}}$. If $|A_i \backslash I| \leq t$, $\mathcal{S}_h$ returns true, otherwise returns false.

Hybrid$_2$: $\mathcal{S}_h$ simulates the partial decryption performed by the honest parties just like in the ideal world. For each $k \in [2t+1]$, $\mathcal{S}_h$ computes the partial decryption as $\{f(k)_i\}_{\mathrm{P}_i\in\mathcal{P}} = \mathrm{TFHE}.\mathcal{S}(\mathcal{C}, f(k), f(k)_i, \{sk_i\}_{\mathcal{A}_i\in\mathcal{A}^*})$. The rational polynomial $f(k)$ is still calculated as in the real world.

Hybrid$_3$: Instead of calculating the rational polynomial $f(k)$ just like in the real world, $\mathcal{S}_h$ selects randomly a rational polynomial $u(x)$ of degree t, and computes a rational polynomial $f(x) = a_I(x) \cdot u(x)$.

Hybrid$_4$: $\mathcal{S}_h$ simulates the ciphertexts computed by any honest parties as encryption of 0, just like $\mathcal{S}$ does in the ideal world.

Theorem 1. *Assuming that the TFHE scheme is secure, the proposed MP-TPSI protocol $\Pi_{MP-TPSI}$ securely realizes $\mathcal{F}_{MP-TPSI-CT}$ and $\mathcal{F}_{MP-TPSI-C}$ in a star network topology, and resists a semi-honest adversary who has the ability to corrupt up to $(n-1)$ parties. It can be proved by Lemma 1–4 in Appendix.*

7. Conclusion

In this study, using sparse polynomial interpolation and TFHE, we introduce a MP-TPSI protocol with low communication complexity, in which the communication complexity only depends on the threshold t and the number of parties n, not on the size of data set m. Compared with the existing MP-TPSI protocols, the proposed MP-TPSI protocol utilizes the same form of evaluation polynomial in the cardinality testing and intersection computing phases, which enables the parties to transmit and compute only a small number of evaluation values, and hence reduces the communication and computational cost. Performance evaluation demonstrates that our MP-TPSI protocol requires 92.0% and 67.2% less computational and communication costs respectively than the competitive MP-TPSI protocol. Moreover, the proposed MP-TPSI protocol can achieve the correctness of the intersection result, and ensure the security of the data of the parties, that is, the semi-honest adversary cannot learn additional information beyond the intersection. In the future, we will explore the MP-TPSI protocol in the broadcast communication setting, optimize the rounds of MP-TPSI, and design a more efficient MP-TPSI protocol with malicious security.

Appendix

Lemma 1. *Hybrid*$_0$ *and Hybrid*$_1$ *is computationally indistinguishable due to the correctness of the MP-TPSI protocol $\Pi_{MP-TPSI}$.*

Proof. The difference between Hybrid$_0$ and Hybrid$_1$ is that in Hybrid$_0$, $\mathcal{S}_h$ calls $\mathcal{F}_{\mathrm{MP-TPSI-CT}}$ honestly, while in Hybrid$_1$, $\mathcal{S}_h$ simulates the ideal functionality $\mathcal{F}_{\mathrm{MP-TPSI-CT}}$ that returns *true* if $|A_i \backslash I| \leq t$ and false otherwise. In Hybrid$_0$, the output result of $\mathcal{F}_{\mathrm{MP-TPSI-CT}}$ is correct due to the correctness of our protocol $\Pi_{\mathrm{MP-TPSI}}$. In Hybrid$_1$, the output result of $\mathcal{F}_{\mathrm{MP-TPSI-CT}}$ is always correct. Therefore, Hybrid$_0$ and Hybrid$_1$ are computationally indistinguishable.

Lemma 2. *Hybrid*$_1$ *and Hybrid*$_2$ *is computationally indistinguishable due to the simulation-based security of TFHE [42].*

Proof. The difference between Hybrid$_1$ and Hybrid$_2$ is that in Hybrid$_1$, $\mathcal{S}_h$ computes the partial decryption of TFHE of all honest parties just like in the real world, while in Hybrid$_2$, $\mathcal{S}_h$ simulates the partial decryption by running TFHE.$\mathcal{S}$. If there is an $\mathcal{Z}$ that is able to distinguish Hybrid$_1$ and Hybrid$_2$ with a non-negligible probability ϵ, we are able to build a reduction algorithm $\mathcal{B}$ that has the ability to break TFHE's simulation-based security with a non-negligible probability ϵ'.$\mathcal{B}$ interacts with a challenger $\mathcal{C}$ in TFHE's simulation-based security game, and interacts with $\mathcal{Z}$ in the game of Hybrid$_1$ and Hybrid$_2$. The corrupt parties in the game of $\mathcal{B}$ and $\mathcal{Z}$ are the same as the corrupt parties in the game of $\mathcal{B}$ and $\mathcal{C}$. $\mathcal{B}$ sends the public key share pk_i and secret key share sk_i of the corrupt party that it receives from $\mathcal{Z}$ to $\mathcal{C}$, and sends the public key share pk_i of the honest party that it receives from $\mathcal{C}$ to $\mathcal{Z}$. $\mathcal{B}$ sends the corrupt party's input data set A_i and random value R_i that it receives from $\mathcal{Z}$ to $\mathcal{C}$. $\mathcal{B}$ sends the honest party's ciphertext that it receives from $\mathcal{C}$ to $\mathcal{Z}$. $\mathcal{B}$ sends the evaluation circuit of rational polynomial $f(x)$ to $\mathcal{C}$. $\mathcal{C}$ returns the honest party's partial decryption to $\mathcal{B}$. $\mathcal{B}$ continues to interact with $\mathcal{Z}$ for the rest progress just like in Hybrid$_1$. In the interaction process, if $\mathcal{C}$ sends honestly computed partial decryption, then the interaction process between $\mathcal{B}$ and $\mathcal{Z}$ is associated with Hybrid$_1$, if the partial decryption is simulated by TFHE.$\mathcal{S}$, the interaction process between $\mathcal{B}$ and $\mathcal{Z}$ is associated with Hybrid$_2$.

From above, if there is an $\mathscr{Z}$ that is able to distinguish Hybrid_1 and Hybrid_2 with a non-negligible probability ϵ, $\mathscr{B}$ has the ability to break TFHE's simulation-based security with a non-negligible probability ϵ', this contradicts with TFHE's simulation-based security [42]. Therefore, Hybrid_1 is computationally indistinguishable from Hybrid_2.

Lemma 3. *Hybrid_2 is statistically close to Hybrid_3.*

Proof. The difference between Hybrid_2 and Hybrid_3 is how the rational polynomial $f(x)$ is calculated. In Hybrid_2, $\mathcal{S}_h$ computes

$$\begin{aligned} f(x) &= \sum_{\mathrm{P}_i \in \mathscr{P}} \left(r_i \cdot \prod_{a_{i,j} \in A_i} \left(x - a_{i,j} \right) \right) = a_I(x) \cdot \sum_{i \in [n]} \left(r_i \cdot \prod_{a_{i,j} \in A_i} \left(x - a_{i,j} \right) \right) \\ &= a_I(x) \cdot \sum_{\mathrm{P}_i \in \mathscr{A}} \left(r_i \cdot \prod_{a_{i,j} \in A_i} \left(x - a_{i,j} \right) \right) + a_I(x) \cdot \sum_{\mathrm{P}_i \in \mathscr{P}} \left(r_i \cdot \prod_{a_{i,j} \in A_i} \left(x - a_{i,j} \right) \right) \\ &= a_I(x) \cdot v_1(x) + a_I(x) \cdot v_2(x) \\ &= a_I(x) \cdot \left(v_1(x) + v_2(x) \right). \end{aligned} \tag{A.1}$$

For each $i \in [n]$, $\mathrm{Deg}(r_i \cdot a_{A_i \setminus I}(x)) = t$. Thus, $\mathrm{Deg}(v_1(x)) = \mathrm{Deg}(v_2(x)) = t$. Since $v_2(x)$ is statistically close to a uniform random polynomial of degree t, we can obtain $f(x) = a_I(x) \cdot (v_1(x) + v_2(x)) = a_I(x) \cdot u(x)$, where $u(x)$ is uniform random polynomials of degree t. In Hybrid_3, $\mathcal{S}_h$ computes $f(x) = a_I(x) \cdot u(x)$. Therefore, the distribution of $f(x)$ in Hybrid_2 is statistically close to the distribution of $f(x)$ in Hybrid_3.

Lemma 4. *Hybrid_3 and Hybrid_4 is computationally indistinguishable due to the semantic security of TFHE [42].*

Proof. The difference between Hybrid_3 and Hybrid_4 is that in Hybrid_3, $\mathcal{S}_h$ computes the encryption of TFHE of all honest parties just like in the real world, while in Hybrid_4, $\mathcal{S}_h$ computes the encryption of 0.

If there is an $\mathscr{Z}$ that is able to distinguish Hybrid_3 and Hybrid_4 with a non-negligible probability ϵ, we are able to build a reduction algorithm $\mathscr{B}$ that has the ability to break TFHE's semantic security with a non-negligible probability ϵ'. $\mathscr{B}$ interacts with a challenger $\mathscr{C}$ in TFHE's semantic security game, and interacts with $\mathscr{Z}$ in the game of Hybrid_3 and Hybrid_4. The corrupt parties in the game of $\mathscr{B}$ and $\mathscr{Z}$ are the same as the corrupt parties in the game of $\mathscr{B}$ and $\mathscr{C}$. $\mathscr{B}$ sends the public key share pk_i and secret key share sk_i of the corrupt party that it receives from $\mathscr{Z}$ to $\mathscr{C}$, and sends the public key share pk_i of the honest party that it receives from $\mathscr{C}$ to $\mathscr{Z}$. $\mathscr{B}$ sends the honestly generated plaintext and 0 to $\mathscr{C}$. $\mathscr{C}$ returns their ciphertexts to $\mathscr{B}$. $\mathscr{B}$ uses the ciphertext it receives from $\mathscr{C}$ to interact with $\mathscr{Z}$. $\mathscr{B}$ continues to interact with $\mathscr{Z}$ for the rest progress just like in Hybrid_3. In the interaction process, if $\mathscr{C}$ sends honestly computed ciphertext, then the interaction process between $\mathscr{B}$ and $\mathscr{Z}$ is associated with Hybrid_3, if the ciphertext is computed as 0's encryption, the interaction between $\mathscr{B}$ and $\mathscr{Z}$ is associated with Hybrid_4.

From above, if there is an $\mathscr{Z}$ that is able to distinguish Hybrid_3 and Hybrid_4 with a non-negligible probability ϵ, $\mathscr{B}$ has the ability to break TFHE's simulation-based security with a non-negligible probability ϵ', this contradicts with TFHE's semantic security [42]. Therefore, Hybrid_3 is computationally indistinguishable from Hybrid_4.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. U19B2021) and the National Natural Science Foundation of China (Grant no. 62172427).

References

[1] C. Meadows, "A more efficient cryptographic matchmaking protocol for use in the absence of a continuously available third party," in *Proceedings of the IEEE Symposium on Security & Privacy*, pp. 134–137, Oakland, CA, USA, April 1986.

[2] J. R. Troncoso-Pastoriza, S. Katzenbeisser, and M. Celik, "Privacy preserving error resilient DNA searching through oblivious automata," in *Proceedings of the ACM Conference on Computer & Communications Security*, pp. 519–528, Alexandria, VA, USA, October 2007.

[3] S. Nagaraja, P. Mittal, C. Y. Hong, M. Caesar, and N. Borisov, "BotGrep: finding P2P bots with structured graph analysis," in *Proceedings of the 19th USENIX Security Symposium*, pp. 95–110, Washington, DC, USA, August 11-13, 2010.

[4] D. Demmler, P. Rindal, M. Rosulek, and N. Trieu, "PIR-PSI: scaling private contact discovery," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 4, pp. 159–178, 2018.

[5] D. Mutchler, "Matching secrets in the absence of a continuously available trusted authority," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 289–292, 1987.

[6] D. Dachman-Soled, T. Malkin, M. Raykova, and M. Yung, "Efficient robust private set intersection," in *Proceedings of the*

International Conference on Applied Cryptography and Network Security, pp. 125–142, Paris-Rocquencourt, France, June 2009.

[7] E. D. Cristofaro, J. Kim, and G. Tsudik, "Linear-complexity private set intersection protocols secure in malicious model," in *Proceedings of the Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 213–231, Singapore, December 2010.

[8] E. D. Cristofaro and G. Tsudik, "Practical private set intersection protocols with linear complexity," in *Proceedings of the 14th International Conference on Financial Cryptography & Data Security*, pp. 143–159, Tenerife, Canary Islands, January 2010.

[9] Y. Huang, D. Evans, and J. Katz, "Private set intersection: are garbled circuits better than custom protocols?" in *Proceedings of the 19th Network and Distributed Security Symposium*, pp. 1–15, San Diego, CA, USA, February 2012.

[10] C. Hazay and K. Nissim, "Efficient set operations in the presence of malicious adversaries," *Journal of Cryptology*, vol. 25, no. 3, pp. 383–433, 2012.

[11] C. Hazay, "Oblivious polynomial evaluation and secure set-intersection from algebraic PRFs," in *Proceedings of the 20th Annual TCC Worldwide Online Conference*, pp. 90–120, 2012.

[12] C. Dong, L. Chen, and Z. Wen, "When private set intersection meets big data: an efficient and scalable protocol," in *Proceedings of the 20th ACM SIGSAC Conference on Computer & communications security*, pp. 789–800, Berlin, Germany, November 2013.

[13] B. Pinkas, T. Schneider, and M. Zohner, "Faster private set intersection based on OT extension," *ACM Transactions on Privacy & Security*, vol. 21, no. 2, pp. 797–812, 2014.

[14] B. Pinkas, T. Schneider, and G. Segev, "Phasing: private set intersection using permutation-based hashing," in *Proceedings of the 24th USENIX Security Symposium*, pp. 515–530, Washington, DC, USA, August 2015.

[15] V. Kolesnikov, R. Kumaresan, M. Rosulek, and N. Trieu, "Efficient batched oblivious PRF with applications to private set intersection," in *Proceedings of the 23rd ACM SIGSAC Conference on Computer & Communications Security*, pp. 818–829, Vienna, Austria, October 2016.

[16] M. J. Freedman, C. Hazay, K. Nissim, and B. Pinkas, "Efficient set intersection with simulation-based security," *Journal of Cryptology*, vol. 29, no. 1, pp. 115–155, 2016.

[17] M. Orru, E. Orsini, and P. Scholl, "Actively secure 1-out-of-N OT extension with application to private set intersection," in *Proceedings of the Cryptographers' Track at the RSA Conference - CTRSA 2017*, pp. 381–396, San Francisco, CA, USA, February 2018.

[18] P. Rindal and M. Rosulek, "Improved private set intersection against malicious adversaries," in *Proceedings of the Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 235–259, Paris, France, May 2017.

[19] P. Rindal and M. Rosulek, "Malicious-secure private set intersection via dual execution," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1229–1242, Dallas, TX, USA, October 2017.

[20] B. Pinkas, T. Schneider, and M. Zohner, "Scalable private set intersection based on OT extension," *ACM Transactions on Privacy and Security*, vol. 21, no. 2, pp. 1–35, 2018.

[21] B. Pinkas, T. Schneider, C. Weinert, and U. Wieder, "Efficient circuit-based PSI via cuckoo hashing," in *Proceedings of the Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 125–157, Tel Aviv, Israel, May 2018.

[22] B. Pinkas, M. Rosulek, N. Trieu, and A. Yanai, "SpOT-Light: lightweight private set intersection from sparse OT extension," in *Proceedings of the Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference*, pp. 401–431, Santa Barbara, CA, USA, August 2019.

[23] B. Pinkas, M. Rosulek, N. Trieu, and A. Yanai, "PSI from paxos: fast, malicious private set intersection," in *Proceedings of the Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 739–767, Zagreb, Croatia, May 2020.

[24] A. C. Yao, "Protocols for secure computations," in *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, pp. 1–5, Chicago, IL, USA, November 1982.

[25] S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," *Communications of the ACM*, vol. 28, no. 6, pp. 637–647, 1985.

[26] R. L. Rivest, L. M. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," *Foundations of Secure Computation*, vol. 76, no. 4, pp. 169–179, 1978.

[27] M. J. Freedman, Y. Ishai, B. Pinkas, and O. Reingold, "Keyword search and oblivious pseudorandom functions," in *Proceedings of the Theory of Cryptography Conference - TCC 2005*, pp. 303–324, Cambridge, MA, USA, February 2005.

[28] L. Kissner and D. Song, "Privacy-preserving set operations," in *Proceedings of the Advances in Cryptology - CRYPTO 2005 - 25th Annual International Cryptology Conference*, pp. 241–257, Santa Barbara, CA, USA, August 2005.

[29] A. Miyaji and S. Nishida, "A scalable multiparty private set intersection," in *Proceedings of the International Conference on Network and System Security*, pp. 376–385, New York, NY, USA, November 2015.

[30] C. Hazay and M. Venkitasubramaniam, "Scalable multi-party private set-intersection," in *Proceedings of the Public-Key Cryptography - PKC 2017 - 20th IACR International Conference on Practice and Theory in Public-Key Cryptography*, pp. 175–203, Amsterdam, Netherlands, May 2017.

[31] V. Kolesnikov, N. Matania, B. Pinkas, M. Rosulek, and N. Trieu, "Practical multi-party private set intersection from symmetric-key techniques," in *Proceedings of the ACM SIGSAC Conference on Computer & Communications Security*, pp. 1257–1272, Dallas, TX, USA, November 2017.

[32] R. Inbar, E. Omri, and B. Pinkas, "Efficient scalable multiparty private set-intersection via garbled bloom filters," in *Proceedings of the 11th International Conference on Security and Cryptography for Networks, Lecture Notes in Computer Science*, vol. 7, pp. 235–252, Amalfi, Italy, September 2018.

[33] S. Ghosh and T. Nilges, "An algebraic approach to maliciously secure private set intersection," in *Proceedings of the Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 154–185, Darmstadt, Germany, May 2019.

[34] L. Lu and N. Ding, "Multi-party private set intersection in vertical federated learning," in *Proceedings of the 19th International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 707–714, Guangzhou, China, January 2020.

[35] A. Kavousi, J. Mohajeri, and M. Salmasizadeh, "Efficient scalable multi-party private set intersection using oblivious PRF," in *Proceedings of the 17th International Workshop on*

Security and Trust Management, pp. 81–99, Darmstadt, Germany, October 2021.

[36] A. Bay, Z. Erkin, J. H. Hoepman, S. Samardjiska, and J. Vos, “Practical multi-party private set intersection protocols,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1–15, 2022.

[37] Q. Yang, Y. Liu, T. Chen, and Y. Tong, “Federated machine learning,” *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1–19, 2019.

[38] S. Ghosh and M. Simkin, “The communication complexity of threshold private set intersection,” in *Proceedings of the Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference*, pp. 3–29, Santa Barbara, CA, USA, August 2019.

[39] R. A. Mahdavi, T. Humphries, B. Kacsmar et al., “Practical over-threshold multi-party private set intersection,” in *Proceedings of theAnnual Computer Security Applications Conference*, pp. 772–783, Austin, USA, December 2020.

[40] P. Branco, N. Dttling, and S. Pu, “Multiparty cardinality testing for threshold private intersection,” in *Proceedings of the Public-Key Cryptography - PKC 2021 - 24th IACR International Conference on Practice and Theory of Public Key Cryptography*, pp. 32–60, Virtual Event, May 2021.

[41] S. Badrinarayanan, P. Miao, S. Raghuraman, and P. Rindal, “Multi-party threshold private set intersection with sublinear communication,” in *Proceedings of the Public-Key Cryptography - PKC 2021 - 24th IACR International Conference on Practice and Theory of Public Key Cryptography*, pp. 349–379, Virtual Event, May 2021.

[42] D. Boneh, R. Gennaro, S. Goldfeder, A. Jain, and A. Sahai, “Threshold cryptosystems from threshold fully homomorphic encryption,” in *Proceedings of the Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference*, pp. 565–596, Santa Barbara, CA, USA, August 2018.

[43] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Proceedings of the Advances in Cryptology - EUROCRYPT 1999 - International Conference on the Theory and Application of Cryptographic Techniques*, pp. 223–238, Prague, Czech Republic, May 1999.

[44] B. H. Bloom, “Space/time trade-offs in hash coding with allowable errors,” *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[45] R. Cramer, R. Gennaro, and B. Schoenmakers, “A secure and optimally efficient multi-authority election scheme,” in *Proceedings of the Advances in Cryptology - EUROCRYPT 1997 - International Conference on the Theory and Application of Cryptographic Techniques*, pp. 103–118, Konstanz, Germany, May 1997.

[46] M. J. Freedman, K. Nissim, and B. Pinkas, “Efficient private matching and set intersection,” in *Proceedings of the Advances in Cryptology - EUROCRYPT 2004 - International Conference on the Theory and Application of Cryptographic Techniques*, pp. 19–38, Interlaken, Switzerland, May 2004.

[47] A. Abadi, S. J. Murdoch, and T. Zacharias, “Polynomial representation is tricky: maliciously secure private set intersection revisited,” in *Proceedings of the 26th European Symposium on Research in Computer Security*, pp. 721–742, Virtual Event, October 2021.

[48] T. Elgamal, “A public key cryptosystem and a signature scheme based on discrete logarithms,” *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[49] R. Canetti, “Universally composable security: a new paradigm for cryptographic protocols,” in *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science*, pp. 136–145, Las Vegas, NV, USA, October 2001.

[50] Lattigo, “Tunesight,” 2022, https://github.com/tuneinsight/lattigo.

[51] S. Halevi, Y. Polyakov, and V. Shoup, “An improved RNS variant of the BFV homomorphic encryption scheme,” in *Proceedings of the Topics in Cryptology - CT-RSA 2019 - the Cryptographers’ Track at the RSA Conference*, pp. 83–105, San Francisco, CA, USA, March 2019.