DE-RSTC: A rational secure two-party computation protocol based on direction entropy

Yuling Chen¹,² | Juan Ma¹ | Xianmin Wang³ | Xinyu Zhang¹ | Huiyu Zhou⁴

¹State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guizhou University, Guiyang, China
²Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, China
³Institute of Artificial Intelligence and Blockchain, Guangzhou University, Guangzhou, China
⁴School of Informatics, University of Leicester, England, United Kingdom

Correspondence
Xianmin Wang, Institute of Artificial Intelligence and Blockchain, Guangzhou University, 510002 Guangzhou, China. Email: xianmin@gzhu.edu.cn

Funding information
National Key Project of China, Grant/Award Number: 2020YFB1005700; Natural Science Foundation, Grant/Award Numbers: 61962009, 62072127, 62002076; National Key Research and Development Program of China, Grant/Award Number: 2021YFA1000600; National Key Research and Development Program of Guangdong Province, Grant/Award Number: 2020B0101090002; Major Scientific and Technological Special Project of Guizhou Province, Grant/Award Numbers: 20183001, 20193003; Foundation of Guangxi Key Laboratory of Cryptography and Information Security, Grant/Award Number: GCIS202118; Science and Technology Support Plan of Guizhou Province, Grant/Award Number: [2020]2Y011; Key Research and Development Program of Guangzhou,

Abstract
Rational secure multi-party computation means two or more rational parties complete a function on private inputs. Unfortunately, players sending false information can prevent the protocol from executing correctly, which will destroy the fairness of the protocol. To ensure the fairness of the protocol, the existing works on achieving fairness by specific utility functions. In this paper, we leverage game theory to propose the direction entropy-based solution. To this end, we utilize the direction entropy to examine the player's strategy uncertainty and quantify its strategy from different dimensions. Then, we provide mutual information to construct a new utility for the players. What's more, we measure the mutual information of players to appraise their strategies. By analyzing and proofing of protocol, we show that the protocol reaches a Nash equilibrium when players choose a cooperative strategy. Furthermore, we
solve the fairness of the protocol. Compared to the previous approaches, our protocol is not required deposits and design-specific utility functions.

**KEYWORDS**
direction entropy, direction vector, mutual information, Nash equilibrium, rational two-party computation

## 1 | INTRODUCTION

With the rapid development of big data, the application of cryptography has become more and more popular. In cryptography, secure multiparty computation (SMC) refers to methods for parties to jointly compute a function over their inputs without revealing these values. In SMC, the multiparty protocol assumes that parties are honest or malicious. The honest players abide by the rules of the protocol, and the malicious players deviate from the protocol. However, parties make strategic choices to maximize their utility. Therefore, how to ensure the correct implementation of the protocol is an issue that cannot be ignored. To solve this problem, Halpern first proposed the concept of rational parties and used rational secure multiparty computation (RSMC).

RSMC combines game theory and SMC, which is different than SMC. The reason is that it is based on the utility execution protocol. During the execution of the protocol, players have multiple behaviors, which will lead to uncertainty of the protocol and the uncertainty of events can be measured by information entropy. However, when using information entropy to measure the player’s strategy, it cannot measure the positive or negative of the amount of information transmitted. Thus, we increase the positive and negative quantification of information based on information entropy, and we measure the player’s strategy from different dimensions. In effect, we exploit the direction entropy to dynamically quantify the behaviors of the parties to reach the optimal state of the protocol.

Our rational secure two-party computation protocol is a solution based on direction entropy. More concretely, we utilize the direction vector of direction entropy to measure the player’s strategy from different dimensions. Next, we employ mutual information to construct the player’s utility function, and analyze the player’s interaction in each round to measure the amount of information. Furthermore, we establish a communication channel for the player, and provide the channel capacity to express the maximum utility that the player obtains. Finally, we give a fairness proof of the protocol. Also, we prove that the protocol has a Nash equilibrium.

### 1.1 | Related works

SMC is an emerging topic which has been drawing growing attention during recent decades. The “millionaire” problem proposed by Yao has considered the beginning of SMC in 1982. It is
the first one that solved the problem of secure two-party computation. Subsequently, Yao\(^9\) expanded it to solve other problems such as secret sharing. Goldreich et al.\(^10\) extended the two parties to multiple parties and defined the security of SMC. So far, there have been many studies on SMC based on cryptography.\(^11\) In particular, to solve the fairness in SMC, more research like Gordon and colleagues.\(^12\)-\(^14\) However, in these schemes, although extremely reliable execution can be guaranteed, the cost caused by the cryptographic proof is still expensive.

Game theory is always used to predict the behavior of rational players, where the players are no longer viewed as being honest or corrupt and will not choose a strategy without motivation.\(^15\) Specifically, players who follow the protocol are honest, and players who deviate from the protocol are corrupt. By contrast, all parties are simply rational motivated by some utility function. Since Halpern et al.\(^6\) first proposed the rational player and applied it to the field of secret sharing, lots of research on bridging game theory and cryptography have been done. For instance, Wang et al.\(^16\) proposed a rational fair calculation protocol that satisfies a computable sequential equilibrium. Subsequently, they introduced the current research status of RSMC and discussed the advantages and disadvantages of the previous protocols.\(^17\) In addition, more research on RSMC such as Dou and colleagues.\(^18\),\(^19\)

Since Shannon created information theory, information entropy has been used to measure the uncertainty of events. Since then, lots of research on the uncertainty of measuring events by information entropy have been done.\(^20\) In addition, the researchers studied the fairness,\(^21\) security,\(^22\),\(^23\) and privacy\(^24\) issues in SMC based on information entropy. In particular, Yang et al.\(^25\) proposed the idea of direction entropy based on information entropy. Specifically, it is based on the information entropy to increase the judgment of the positive and negative of the information. However, existing works guarantee the player’s honest strategy by predepositing deposits and defining specific utility functions, which is a lack of persuasion since they elaborate designed against set targets.

Recently, there are many kinds of research that use mutual information to study different fields. Tian et al.\(^26\) combined mutual information with rational commissioned calculations to achieve the optimal offensive and defensive strategy of rational commissioned calculations, but only in one-to-one situations. Subsequently, Li et al.\(^27\) extended the one-to-one rational commission calculation to the one-to-many rational delegating computation. In addition, Liu et al.\(^28\) utilized mutual information to obtain the optimal interval of the \(k\) value under the \(k\)-anonymous model. However, in these schemes, the researcher converts the two or more parties' protocol into a communication problem between players. Based on these schemes, we present mutual information to construct a rational player’s utility function. Concurrently, when the player's mutual information reaches the channel capacity, its utility is also maximized.

### 1.2 Our contribution

The main contribution of this study is to design a new RSTC protocol by using the direction entropy to measure the player’s behavior from different dimensions and redefine the player’s utility function. Compared with the existing work, we not require deposits and design-specific utility functions. The fairness of the protocol can be guaranteed. Therefore, compared to the previous solutions, our main contributions are as follows.
• Direction entropy. Based on the direction entropy, we propose a direction entropy algorithm for rational secure two-party computation. More specifically, we provide the direction vector to investigate the player’s strategy uncertainty from different dimensions. If the player chooses the cooperation strategy, and its direction vector is positive, which means that the information transmitted by the player is positive and vice versa.

• New utility function. Perhaps more interestingly, we analyze the player’s behavioral strategies, and we employ mutual information to represent the player’s new utility function. Next, when the information transmitted by the player is positive, the number of mutual information increases. In other words, if the player chooses a cooperative strategy, its utility is raised. Conversely, the utility is reduced. Then, we construct a communication channel for the player and measure the player’s channel capacity to determine the player’s behavior strategy.

• Achieve fairness. We prove in detail the protocol we gave, and the result show the fairness problem in rational secure two-party computation. We prove that the proposed protocol reaches the Nash equilibrium. Moreover, we analyze our entire scheme and conclude that the protocol we design maximizes utility for players. In a nutshell, the protocol reaches the Nash equilibrium when the probability of players choosing to cooperate approaches 1.

1.3 | Roadmap

The rest of the paper is structured as follows. Section 1 presents the related work and our contribution. Section 2 introduces concepts such as game theory, information entropy, direction entropy, mutual information, and channel capacity. Section 3 proposes an information exchange protocol based on direction entropy. Section 4 constructs the player’s utility function based on mutual information. Section 5 proves and analyses the protocol. Section 6 simulates the player’s utility and analyzes the performance of the protocol. The paper concludes with a summary and future directions in Section 7.

2 | PRELIMINARIES

In this section, we introduce the basic knowledge needed for our scheme. First, we introduce the concept of the standard game. Furthermore, we review the most important concept in game theory, that is, the Nash equilibrium. Then, we give the definition of information entropy and direction entropy. Finally, we introduce some basic knowledge in information theory.

2.1 | Game theory

Definition 1 (Standard game). The standard form of a n-player game is composed of three elements: player set P, strategy space S and utility function u, denoted
as $G = \{P, S, u\}$, where $P = \{P_1, ..., P_n\}$, $S = \{S_1, ..., S_n\}$, $u = \{u_1, ..., u_n\}$. Any specific strategy $s_i \in S_i$ indicates that strategy $s_i$ is the key element of the strategy set $S_i$, and utility function $u : S \rightarrow R$ denotes the profits of the players $i$ under different strategy profiles.

**Definition 2** (Nash equilibrium). A strategy profile $s^* = \{s_1^*, ..., s_n^*\}$ is a Nash equilibrium of game $G = \{P, S, u\}$, if $u_i(s_i^*, s_{-i}^*) \geq u_j(s_j^*, s_{-j}^*)$ holds for each player $P_i (i = 1, ..., n)$ and all $s_j \in S_j$.

Obviously, if player $i \neq j$ complies with the strategy $s_i^*$, then the player will not deviate from the strategy $s_j^*$, as it will not benefit at all. In principle, there may be multiple Nash equilibrium in a game.

### 3 | ENTROPY

**Definition 3** (Information entropy). Information entropy is a measure of the degree of uncertainty. If a thing $X$ has $n$ mutually exclusive possible outcomes, and the probability of occurrence of the $n$ outcomes is $\{p_1, ..., p_n\}$ respectively, the entropy to measure the degree of uncertainty of the thing $X$ is defined as:

$$ H(X) = \sum_{i=1}^{n} p_i \log \frac{1}{p_i} \quad (1) $$

**Definition 4** (Direction entropy). Direction entropy is based on the information entropy to increase the measurement of the positive and negative information. Specifically, it refers to a random variable $X$ composed of $n$ random events with probabilities $\{p_1, ..., p_n, p_1 +, ..., +p_n = 1\}$, and the total amount of information contained is:

$$ H(a, X) = \sum_{i=1}^{n} a_i p_i \log \frac{1}{p_i} \quad (2) $$

### 4 | INFORMATION THEORY

**Definition 5** (Mutual information). Another random variable causes to reduce the decrement of uncertainty of the original random variable. The mutual information denotes the decrement of uncertainty about $X$ between before and after receiving $Y$. It is also the metric of independent degree between two random variables. It is symmetrical about $X$ and $Y$ and nonnegative, the mutual information is 0 if and only if $X$ and $Y$ are mutually independent.

$$ H(X; Y) = H(X) - H(X|Y) = \sum_{x,y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (3) $$
Definition 6 (Channel capacity). For the communication channel where the input signal is $X$ and the output signal is $Y$, the channel capacity $C$ defined as:

$$C = \max_{p(x)} I(x;Y) = \max_{p(x)} \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$  \hspace{1cm} (4)$$

5  | INFORMATION EXCHANGE PROTOCOL BASED ON DIRECTION ENTROPY

In this section, we design a new protocol to measure rational player strategy based on direction entropy. First, we examine the player’s strategy uncertainty from different dimensions. Moreover, we exploit the direction vector of the direction entropy to quantify the positive and negative of the information transmitted by the player.

5.1  | Rational secure two-party computation framework

In this section, we first describe the parameters and concepts required in this solution. Specifically, we use the following notations in Table 1.

As illustrated in the framework of the protocol, our protocol consists of three main phases described in detail in the following.

Phase 1: At this stage, the initialization of the protocol is mainly completed (Initialization of direction entropy is given in Section 3.2).

Phase 2: In the second phase, this phase completes the information exchange of the protocol (Information exchange based on direction entropy is given in Section 3.3).

| Serial | Symbol | Description |
|--------|--------|-------------|
| (1)    | $\mu$  | $P_1$ probability of choosing cooperation |
| (2)    | $\nu$  | $P_2$ probability of choosing cooperation |
| (3)    | $C$    | The cooperative strategy |
| (4)    | $D$    | The non-cooperative strategy |
| (5)    | $S$    | The secret |
| (6)    | $H(dir_t)$ | Directional entropy at round $t$ |
| (7)    | $H(dir_{t-1})$ | Directional entropy at round $t - 1$ |
| (8)    | $\mathcal{A} \in \{C,D\}$ | The strategy set |
| (9)    | $s_1^1, ..., s_1^m$ | Sub-secret of player $P_1$ |
| (10)   | $s_2^1, ..., s_2^m$ | Sub-secret of player $P_2$ |
| (11)   | $P_i \in \mathcal{P}(i \in \{1, 2\})$ | The player set |
Phase 3: At this stage, this phase completes the information computation of the protocol (Computation based on direction entropy is given in Section 3.4).

Phase 4: Finally, the integration phase is where the participant integrates all received sub-secrets to recover the secret.

Three main phases of the proposed protocol are summarized in the following.

---

**Protocol π**

**Phase 1: Initialization phase**
Step 1: TTP divides the $S$ into $m$ shares, i.e., $s_1^1, ..., s_1^m$ and $s_2^1, ..., s_2^m$.
Step 2: TTP sends the $s_1^1, ..., s_1^m$ to $P_1$ and $s_2^1, ..., s_2^m$ to $P_2$.

**Phase 2: Exchange phase**
During the $t$-th round of secret exchange, there are the following steps:
Step 3: $P_1$ sends sub-secret $s_t^1$ to $P_2$.
Step 3.1: If $P_1$ chooses $C$, then $a_t = 1$, there is $H_d(i_r)(> 0)$ $P_1$;
Step 3.2: Otherwise, $a_t = -1$, there is $H_d(i_r)(0)$$P_1$$\leq$.
Step 4: $P_2$ sends sub-secret $s_t^2$ to $P_1$.
Step 4.1: If $P_2$ chooses $C$, then $b_t = 1$, there is $H_d(i_r)(> 0)$ $P_2$;
Step 4.2: Otherwise, $b_t = -1$, there is $H_d(i_r)(0)$$P_2$$\leq$.

**Phase 3: Computation phase**
Step 5: Once $P_i$ receives the subsecret from $P_i$, we use the public function $f$ to calculate: $f(s_t^i) \rightarrow y_i(I(S; s_t^i))$.

We use $s_j$ to denote the subsecrets that have been successfully exchanged. The specific steps are as follows:
Step 5.1: If $I(S; s_t^i) \neq I(S; s_t^j)$, then $P_i$ successfully obtains a part of $P_j$’s secret;
Step 5.2: If $I(S; s_t^i) = I(S; s_t^j)$, then $P_i$ obtains a duplicate sub-secret of $P_j$;
Step 5.3: If $I(S; s_t^i) = 0$, then $P_i$ obtains an incorrect sub-secret of $P_j$.

**Phase 4: Integration phase**
When $n$ rounds of protocol interaction is completed, if no player terminates the protocol during the execution of the protocol, the result of the protocol is that both players obtain all correct sub-secrets, and the secret is restored by integrating all sub-secrets, which is also called the correct result value. Subsequently, both players withdrew from the protocol.

---

5.2 Initialization of direction entropy

As mentioned Definition 4, $n$ is the number of factors in the agreement that are all independent of each other and affect the player’s strategy, and $p_i$ is the probability of an event triggered by the $i$-th factor that affects the player’s strategy. This includes all the factors that affect the player’s strategy, so there is a probability normalization condition:

$$\sum_{i=1}^{n} a_i p_i = 1$$ (5)

We assume that the factors that affect the strategy in the agreement are $p_1, ..., p_n$ (such as complexity; utility; time, etc.), and the direction entropy at round $t$ is $H(dir_t)$. Meanwhile, $H(dir_t)$ increases with the number of rounds, that is, $\frac{dH(dir_t)}{dt} > 0$. It means that the player
complied with the protocol during the \( t \) round. Conversely, \( H(dir_t) \) decreases with the number of rounds, that is, \( \frac{dH(dir_t)}{dt} < 0 \). It indicates that the player deviated from the protocol during the \( t \) round.

(1) First, we assume that a trusted third-party TTP divides the secret \( S \) into \( m \) shares, and sends the \( S \) shares to the player \( P_i \in \mathcal{P}(i \in 1, 2) \).

(2) TTP chooses \( s_1^1, ..., s_1^m \) and \( s_2^1, ..., s_2^m \) as the exchange secret \( S \), and they are held by \( P_1 \) and \( P_2 \), respectively.

(3) \( P_1 \) and \( P_2 \) exchange secrets \( s_1^1, ..., s_1^m \) and \( s_2^1, ..., s_2^m \) for \( n(m < n) \) rounds, and finally \( P_1 \) and \( P_2 \) integrate the \( m \) secrets received to form \( S \).

In the exchange phase, player \( P_i \in \mathcal{P}(i \in 1, 2) \) has two behavior strategies, namely cooperative strategy \( C \) or non-cooperative strategy \( D \). The strategy set is \( \mathcal{A} \in \{C, D\} \). We combine the strategies of the players to get the strategy combination matrix. The details are as follows:

\[
\mathcal{A}_{2 \times 2} = \begin{pmatrix}
(C, C) & (C, D) \\
(D, C) & (D, D)
\end{pmatrix}
\]

The above strategy matrix has the following four situations:

**Case 1:** (\( C, C \)) means that both \( P_1 \) and \( P_2 \) choose a cooperation strategy.

**Case 2:** (\( C, D \)) means that the \( P_1 \) chooses a cooperative strategy, while \( P_2 \) chooses a non-cooperative strategy.

**Case 3:** (\( D, C \)) means that the \( P_1 \) chooses a non-cooperative strategy, while \( P_2 \) chooses a cooperative strategy.

**Case 4:** (\( D, D \)) means that both \( P_1 \) and \( P_2 \) choose a non-cooperative strategy.

In addition, there are two cases for the direction vectors \( \mathbf{a} = (a_1, a_2, ..., a_n) \) and \( \mathbf{b} = (b_1, b_2, ..., b_n) \) of the players \( P_1 \) and \( P_2 \). Specifically, when the player chooses the cooperative strategy \( C \), the direction vector is positive, and when the player chooses the non-cooperative strategy \( D \), the direction vector is negative. We have the following assumptions:

(1) The probability that the \( P_1 \) chooses \( C \) is \( \mu \), its direction vector is positive, the probability of choosing \( D \) is \( 1 - \mu \), its direction vector is negative, and satisfies \( \mu + (1 - \mu) = 1 \).

(2) The probability that the \( P_2 \) chooses \( C \) is \( \nu \), its direction vector is positive, the probability of choosing \( D \) is \( 1 - \nu \), its direction vector is negative, and satisfies \( \nu + (1 - \nu) = 1 \).

### 5.3 Information exchange based on direction entropy

In the information exchange process in the secure two-party computation, a total of \( n \) rounds of information needs to be exchanged. In each round of information exchange, it is assumed that the player has a current round direction entropy \( H_{P_t}(dir_{cur}) \). In round \( t \in \{1, 2, ..., n\} \), we exploit the direction entropy to evaluate participants' strategies, which is divided into two stage.
Phase 1: Player $P_i \in P (i \in 1, 2)$ (assuming player $P_1$) chooses a strategy from strategy set $A$. If chooses the cooperation strategy $C$, then $P_1$ will send $s_i^1$ to $P_2$. In other words, the direction entropy of $P_1$ is $H_{P_1}(\text{dir}_t)$ at round $t$. If chooses the non-cooperative strategy $D$, then $P_1$ will not send $s_i^1$ to $P_2$, the protocol is terminated, and $P_2$ can only choose $D$. That is, neither $P_1$ nor $P_2$ sends a secret, and their direction entropy is $H_{P_1}(\text{dir}_{t-1})$.

Phase 2: If $P_1$ chooses strategy $C$ in the first stage, $P_2$ chooses a strategy from strategy set $A$. If $P_2$ chooses the cooperation strategy $C$, then $P_2$ will send $s_i^2$ to $P_1$. In other words, the direction entropy of $P_1$ and $P_2$ are $H_{P_1}(\text{dir}_t)$ and $H_{P_2}(\text{dir}_t)$, respectively. If $P_2$ chooses the non-cooperative strategy $D$, then $P_2$ will not send $s_i^2$ to $P_1$, and the protocol is terminated, that is, the direction entropy $P_2$ of is $H_{P_2}(\text{dir}_{t-1})$. If no participant adopts a non-cooperative strategy $D$ and $t < n$, the protocol enters the next round. Otherwise, the protocol terminates.

Specifically, in round $t \in \{1, 2, ..., n\}$, there are the following four cases:

Case 1: If $P_1$ chooses a cooperative strategy, the direction vector of its direction entropy is positive. At the same time, events with probability $\mu$ provide positive information with a quantity of $\mu \log \frac{1}{\mu}$ because $P_1$ chooses a cooperative strategy with probability $\mu$. In short, in the $t$-th round, the direction entropy value of $P_1$ in the $t$-th round is greater than the direction entropy value in the $t-1$ round, that is, $H_{P_1}(\text{dir}_t) > H_{P_1}(\text{dir}_{t-1})$.

Case 2: If $P_1$ chooses a non-cooperative strategy, the direction vector of its direction entropy is negative. At the same time, events with probability $1 - \mu$ provide positive information with a quantity of $-(1 - \mu) \log \frac{1}{1 - \mu}$ because $P_1$ chooses a cooperative strategy with probability $1 - \mu$. In short, in the $t$-th round, the direction entropy value of $P_1$ in the $t$-th round is less than the direction entropy value in the $t-1$ round, that is, $H_{P_1}(\text{dir}_t) < H_{P_1}(\text{dir}_{t-1})$.

Case 3: If $P_2$ chooses a cooperative strategy, the direction vector of its direction entropy is positive. At the same time, events with probability $\nu$ provide positive information with a quantity of $\nu \log \frac{1}{\nu}$ because $P_2$ chooses a cooperative strategy with probability $\nu$. In short, in the $t$-th round, the direction entropy value of $P_2$ in the $t$-th round is greater than the direction entropy value in the $t-1$ round, that is, $H_{P_2}(\text{dir}_t) > H_{P_2}(\text{dir}_{t-1})$.

Case 4: If $P_2$ chooses a non-cooperative strategy, the direction vector of its direction entropy is negative. At the same time, events with probability $1 - \nu$ provide positive information with a quantity of $-(1 - \nu) \log \frac{1}{1 - \nu}$ because $P_2$ chooses a cooperative strategy with probability $1 - \nu$. In short, in the $t$-th round, the direction entropy value of $P_2$ in the $t$-th round is less than the direction entropy value in the $t-1$ round, that is, $H_{P_2}(\text{dir}_t) < H_{P_2}(\text{dir}_{t-1})$.

For the proposed protocol, we design an algorithm to measure the player’s behavioral uncertainty and changes in direction entropy. The specific algorithm details are shown in Algorithm 1:
Algorithm 1 The change of directional entropy

**Input:** $t$, $H(dir_{t-1})$.

**Output:** $H(dir)$.  

1: Initialize $s_1^1, \ldots, s_1^n \in \text{random}(S_1)$  
2: Initialize $s_2^1, \ldots, s_2^n \in \text{random}(S_2)$  
3: Initialize $A = \{C, D\}$.  
4: Initialize $t \leftarrow 1$.  
5: \textbf{for} $t < n$ \textbf{do}  
6: \hspace{2em} $P_2 \leftarrow P_1$ chooses sub-secret $s_1^i$.  
7: \hspace{2em} \textbf{if} $s_1^i \in \text{random}(S_1)$ \textbf{then}  
8: \hspace{4em} $a_i = 1$;  
9: \hspace{4em} $H_{P_1}(dir_i) > H_{P_1}(dir_{t-1})$;  
10: \hspace{4em} \textbf{else}  
11: \hspace{6em} $a_i = -1$;  
12: \hspace{6em} $H_{P_1}(dir_i) \leq H_{P_1}(dir_{t-1})$.  
13: \hspace{2em} $t \leftarrow t + 1$.  
14: \hspace{2em} \textbf{end if}  
15: \hspace{2em} $P_1 \leftarrow P_2$ chooses sub-secret $s_2^i$.  
16: \hspace{2em} \textbf{if} $s_2^i \in \text{random}(S_2)$ \textbf{then}  
17: \hspace{4em} $b_i = 1$;  
18: \hspace{4em} $H_{P_2}(dir_i) > H_{P_2}(dir_{t-1})$;  
19: \hspace{4em} \textbf{else}  
20: \hspace{6em} $b_i = -1$;  
21: \hspace{6em} $H_{P_2}(dir_i) \leq H_{P_2}(dir_{t-1})$.  
22: \hspace{2em} $t \leftarrow t + 1$.  
23: \hspace{2em} \textbf{end if}  
24: \textbf{end for}

According to the above Algorithm 1, we know that players have different strategies. For players’ different strategies, we use direction entropy to measure players’ strategies in multiple dimensions. Specifically, when the players cooperate, the player’s direction vector is positive. Otherwise, the player’s direction vector is negative.

5.4 Computation of the player’s direction entropy

After the end of round $t$, we calculate the direction entropy difference of player $P_i$. We use the direction entropy difference to indicate the change of the direction entropy value of the players after a round of exchange of information, which can help the players to understand each other better in the subsequent exchange process. We assume that the player’s direction entropy at round $t-1$ is $H_{P_i}(dir_{t-1})$, and after the $t$-th round of exchange, $P_i$’s direction entropy is $H_{P_i}(dir_t)$. There are the following computations:

$$\Delta H_{P_i}(dir) = H_{P_i}(dir_t) + H_{P_i}(dir_t - 1)$$ (6)
After each round of information exchange, the player’s directional entropy will change. Specifically, we have the following two situations:

**Case 1:** \( \Delta H_{P_i}(dir > 0) \) indicates that the direction entropy of \( P_i \) in the \( t \)-th round is greater than the direction entropy of the \( t-1 \) round.

**Case 2:** \( \Delta H_{P_i}(dir \leq 0) \) means that the direction entropy of \( P_i \) in the \( t \)-th round is less than the direction entropy of the \( t-1 \) round.

After the \( t \)-th exchange, the direction entropy of \( P_i \) is expressed as follows:

\[
\Delta H_{P_i}(dir_t) = \begin{cases} 
H_{P_i}(dir_{t-1}), & P_i \in D \\
H_{P_i}(dir_{t-1}) + H_{P_{cur}}(dir_{cur}), & P_i \in C
\end{cases}
\]  

(7)

Additionally, we give a detailed algorithm for information exchange based on direction entropy, which is described as follows:

**Algorithm 2** Computation of direction entropy

**Input:** \( H_{P_i}(dir_{t-1}), H_{P_{cur}}(dir) \).

**Output:** \( \Delta H_{P_i}(dir) \).

1: Initialize \( t \leftarrow 1 \).
2: Initialize \( P_i(i \in 1, 2) \).
3: \( P_i \) Compute \( \Delta H_{P_i}(dir) = H_{P_i}(dir_t) - H_{P_i}(dir_{t-1}) \).
4: for \( t < n \) do
5:   if \( \Delta H_{P_i}(dir) > 0 \) then
6:     \( P_i \) sends \( s_i^t \);
7:     \( t \leftarrow t + 1 \);
8:   else
9:     \( P_i \) refuses to send \( s_i^t \);
10: Protocol termination.
11: end if
12: end for

As can be seen from Algorithm 2, each player quantifies the direction entropy of the opposing player in round \( t \). Players are more inclined to trust the opposing participants with higher direction entropy to choose cooperation strategies to exchange information. For players with low direction entropy, they are more inclined to distrust that they will choose a cooperative strategy to exchange information.

6 | NEW UTILITY BASED ON MUTUAL INFORMATION

In this part, we construct utility functions for rational players based on mutual information. First, we define the player’s utility function, which is mutual information. Then, we construct a standard game and communication channel for the player. Finally, we calculate the mutual information of each player in detail.
6.1 | New definition of utility

In this section, we first provide a standard game theory model $G = \{P_1, P_2, S_1, S_2; u_1, u_2\}$, where the direction vectors of players $P_1$ and $P_2$ are $a = (a_1, a_2, ..., a_n)$ and $b = (b_1, b_2, ..., b_n)$ respectively. Next, we assume that a transition matrix $A = [A_{ij}]$, which is equivalent to determining the transition matrix of a certain channel, namely $A_{ij} = p(x = i, y = j), 1 \leq i \leq n, 1 \leq j \leq m$. If $P_1$ and $P_2$ are respectively $n$ and $m$ random variables. The strategic space $S_1$ of $P_1$ is defined as $S_1 = \{0 \leq x_1 \leq 1; 1 \leq i \leq n, x_1 + x_2 + \ldots + x_n = 1\}$, and the strategic space $S_2$ of $P_2$ is defined as $S_2 = \{0 \leq y_1 \leq 1; 1 \leq i \leq m, y_1 + y_2 + \ldots + y_m = 1\}$. $P_1$ and $P_2$ are arbitrary pure strategies any $s_1 \in S_1$ (i.e., $s_1 = (p_1, p_2, ..., p_n)$, $p_1 + p_2 + \ldots + p_n$) and $s_2 \in S_2$ (i.e., $s_2 = (q_1, q_2, ..., q_n)$, $q_1 + q_2 + \ldots + q_n$). In this setting, the utility function $u_1(s_1, s_2)$ of player is defined as:

$$u_1(s_1, s_2) = \sum_{j=1}^{m} p_j \sum_{i=1}^{n} A_{ij} \log \frac{A_{ij}}{p_i}$$

Similarly, the utility function $u_2(s_1, s_2)$ of player $P_2$ is defined as:

$$u_2(s_1, s_2) = \sum_{i=1}^{n} p_i \sum_{j=1}^{m} B_{ij} \log \frac{B_{ij}}{q_j}$$

In summary, the utility function of player $P_1$ is $u_1(s_1, s_2)$, i.e., $I(X; Y)$. Similarly, the utility function of $P_2$ is $u_2(s_1, s_2)$, i.e., $I(Y; X)$. Where the probability distribution functions of $X$ and $Y$ are defined by $s_1$ and $s_2$ as: $P(X = i) = p_i, 1 \leq i \leq n, 0 \leq p_i \leq 1$ and $P(Y = j) = q_j, 1 \leq j \leq m, 0 \leq q_j \leq 1$.

Among the operations, we use mutual information to represent the player’s utility function. Simultaneously, we provide direction entropy to measure the player’s behavior uncertainty. Players choose different strategies, the direction vector of the direction entropy is different, and the amount of positive and negative information transmitted is also different. Therefore, the player’s utility function is defined separately based on the direction entropy.

In the following, we give mutual information to represent the player’s new utility function described in detail.

(1) The utility function $u_1(s_1, s_2)$ of the player $P_1$ is defined as $u_1(s_1, s_2) = ||I(a, X; b, Y)||$, which is the absolute value of the direction mutual information $|I(a, X; b, Y)|$ of $X$ and $Y$.

(2) The utility function $u_2(s_1, s_2)$ of the player $P_2$ is defined as $u_2(s_1, s_2) = ||I(b, Y; a, X)||$, which is the absolute value of the direction mutual information $|I(b, Y; a, X)|$ of $Y$ and $X$.

6.2 | New utility function of the players

In this section, we first utilize $X = 0$ to represent the player $P_1$ chooses a non-cooperative strategy, and $X = 1$ to represent the player $P_1$ chooses a cooperative strategy. Similarly, we use $Y = 0$ to represent the player $P_2$ chooses a non-cooperative strategy, and $Y = 1$ to represent the player $P_2$ chooses a cooperative strategy. Then the behavior strategies of $P_1$
and $P_2$ can be described by the probability distribution of $X$ and $Y$ respectively, as shown in Equation (10):

$$
P_r(X = 0) = 1 - \mu; \quad P_r(X = 1) = \mu
$$

$$
P_r(Y = 0) = 1 - \nu; \quad P_r(Y = 1) = \nu
$$

(10)

Where $0 \leq \mu \leq 1, 0 \leq \nu \leq 1$. According to the probability distribution of $P_1$ and $P_2$, we suppose the joint probability distribution of random variable $(X, Y)$ is as follows:

$$
P_r(X = 0, Y = 0) = a
$$

$$
P_r(X = 0, Y = 1) = b
$$

$$
P_r(X = 1, Y = 0) = c
$$

$$
P_r(X = 1, Y = 1) = d
$$

(11)

$0 < a, b, c, d, \mu, \nu < 1$

Where $a, b, c, d, \mu, \nu$ satisfy the following three linear relationship equations:

$$
a + b + c + d = 1
$$

$$
\mu = P_r(X = 0, Y = 0) + P_r(X = 0, Y = 1) = a + b
$$

$$
\nu = P_r(X = 0, Y = 0) + P_r(X = 1, Y = 0) = a + c
$$

(12)

Next, we construct a communication channel $(X; Y)$ for the player $P_1$ with $X$ as input and $Y$ as output. Specifically, we have the following theorem:

**Theorem 1.** We assume the channel capacity $M$ of the player $P_1$ channel composed of a random variable $(X; Y)$.

(1) If the player wants to obtain the subsecret share of the $t$th round, then there must be some kind of skill that enables it to achieve the goal with a probability close to 1 during the $t/M$ exchange.

(2) If the player wins $S$ times during $n$ rounds of exchange, then there must be $S \leq nM$.

From the above theorem, we only require the channel capacity $M$ of the player’s $P_1$ channel, then the player’s utility limit is determined, that is, the maximum utility function of $P_1$. Its $2 \times 2$ order transition probability matrix $A = [A(i, j)]$, $i, j = 0, 1$, the details are as follows:

$$
A(0, 0) = P_r(Y = 0|X = 0) = \frac{P_r(Y = 0, X = 0)}{P_r(X = 0)} = \frac{a}{\mu}
$$

$$
A(0, 1) = P_r(Y = 1|X = 0) = \frac{P_r(Y = 1, X = 0)}{P_r(X = 0)} = \frac{b}{\mu} = 1 - \frac{a}{\mu}
$$

$$
A(1, 0) = P_r(Y = 0|X = 1) = \frac{P_r(Y = 0, X = 1)}{P_r(X = 1)} = \frac{c}{1 - \mu} = \frac{\nu - a}{1 - \mu}
$$

$$
A(1, 1) = P_r(Y = 1|X = 1) = \frac{P_r(Y = 1, X = 1)}{P_r(X = 1)} = \frac{d}{1 - \nu} = 1 - \frac{\nu - a}{1 - \mu}
$$
Then, we get the transfer matrix of $P_1$ is as follows:

$$A_{2 \times 2} = \begin{bmatrix} A(0, 0) & A(0, 1) \\ A(1, 0) & A(1, 1) \end{bmatrix} = \begin{bmatrix} \frac{a}{\mu} & 1 - \frac{a}{\mu} \\ \frac{\nu - a}{1 - \mu} & 1 - \frac{\nu - a}{1 - \mu} \end{bmatrix}$$

According to the transition probability matrix of player $P_1$, we define the utility function $u_1(s_1, s_2)$ of $P_1$ as the following formula:

$$u_1(s_1, s_2) = |I(a, X; b, Y)|$$

$$= \sum_{a, x, b, y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}$$

$$= a \log \frac{a}{\mu\nu} - b \log \frac{b}{\mu(1 - \nu)} - c \log \frac{c}{\nu(1 - \mu)} - d \log \frac{d}{(1 - \mu)(1 - \nu)}$$

$$= a \log \frac{a}{\mu\nu} - (\mu - a) \log \frac{\mu - a}{\mu(1 - \nu)} - (\nu - a) \log \frac{\nu - a}{\nu(1 - \mu)}$$

$$- (1 + a - \mu - \nu) \log \frac{1 + a - \mu - \nu}{(1 - \mu)(1 - \nu)}$$

(13)

Similar to Theorem 1, we construct a communication channel $(Y; X)$ for the player $P_2$ with $Y$ as input and $X$ as output. To this end, we have the following Theorem 2.

**Theorem 2.** We assume the channel capacity $N$ of the player $P_2$ channel composed of a random variable $(Y; X)$.

(1) If the player wants to obtain the sub-secret share of the $t$-th round, then there must be some kind of skill that enables it to achieve the goal with a probability close to 1 during the $tN$ exchange.

(2) If the player wins $S$ times during $n$ rounds of exchange, then there must be $S \leq nN$.

From the above theorem, we only require the channel capacity $N$ of the player’s $P_2$ channel, then the player’s utility limit is determined, that is, the maximum utility function of $P_2$. The transition probability matrix $B = [B(i, j)]$, $i, j = 0, 1$, the details are as follows:

$$B(0, 0) = P_t(X = 0|Y = 0) = \frac{P_t(X = 0, Y = 0)}{P_t(Y = 0)} = \frac{a}{\nu}$$

$$B(0, 1) = P_t(X = 1|Y = 0) = \frac{P_t(X = 1, Y = 0)}{P_t(Y = 0)} = \frac{c}{1 - \nu} = \frac{\nu - a}{1 - \nu}$$

$$B(1, 0) = P_t(X = 0|Y = 1) = \frac{P_t(X = 0, Y = 1)}{P_t(Y = 1)} = \frac{b}{\nu} = 1 - \frac{a}{\nu}$$

$$B(1, 1) = P_t(X = 1|Y = 1) = \frac{P_t(X = 1, Y = 1)}{P_t(Y = 1)} = \frac{d}{1 - \nu} = 1 - \frac{\nu - a}{1 - \nu}$$
Similar to $P_1$, we get the transfer matrix of $P_2$ is as follows:

$$B_{2 \times 2} = \begin{bmatrix} B(0, 0) & B(0, 1) \\ B(1, 0) & B(1, 1) \end{bmatrix} = \begin{bmatrix} a & 1 - a \\ \nu & \nu - a \\ \nu - a & 1 - \nu \\ 1 - \nu & 1 - \nu \end{bmatrix}$$

Through the transition matrix we obtained above, we show the utility function $u_2(s_1, s_2)$ of $P_2$:

$$u_2(s_1, s_2) = I(b, X; a, Y) = \sum_{b,y} \sum_{a,x} p(x,y) \log \frac{p(y,x)}{p(y)p(x)}$$

$$= \left| a \log \frac{a}{\mu \nu} - b \log \frac{b}{\nu(1-\mu)} - c \log \frac{c}{\mu(1-\nu)} - d \log \frac{d}{(1-\mu)(1-\nu)} \right|$$

$$= \left| a \log \frac{a}{\mu \nu} - (\mu - a) \log \frac{\mu - a}{\mu(1-\nu)} - (\nu - a) \log \frac{\nu - a}{\nu(1-\mu)} - (1 + a - \mu - \nu) \log \frac{1 + a - \mu - \nu}{(1-\mu)(1-\nu)} \right|$$

(14)

7 | PROOF

At first, we prove that the player realizes the Nash equilibrium. The player reaches the Nash equilibrium when the player's mutual information reaches the channel capacity. We also demonstrate the security and fairness of the proposed protocol in detail.

7.1 | Nash equilibrium proof

**Theorem 3.** There is a pair of strategies $(s_1^*, s_2^*)$ in the game that makes the protocol reach the Nash equilibrium.

In a standard game $G = \{P_1, ..., P_n; S_1, ..., S_n; u_1, ..., u_n\}$ with $n$ players, if $n$ is bounded, and for each $i$, $S_i$ is a nonempty convex set in euclidean space, the utility function $u_i$ is continuous with respect to $s_i$, and the pair of $s_i$ is quasi-convex Concave, the game has a Nash equilibrium.

**Proof.** In standard game $G = \{P_1, ..., P_n; S_1, ..., S_n; u_1, ..., u_n\}$, there are two players, that is, $n$ is bounded. Simultaneously, for strategy $S_1 = [0 \leq x_i \leq 1 : 1 \leq i \leq n, x_1 + x_2 + \cdots + x_n = 1]$, it is an $n - 1$ dimensional subcube in an $n$ dimensional closed cube with side length 1, which is a nonempty convex set in Euclidean space. Similarly, for strategy $S_2 = [0 \leq y_i \leq 1 : 1 \leq i \leq m, y_1 + y_2 + \cdots + y_n = 1]$, it is an $m - 1$ dimensional subcube in an $m$ dimensional closed cube of side length 1, which is a nonempty convex set in Euclidean space. In addition, according to the concave-convexity theorem of mutual information, it can be known that when the channel $p(x|y)$ is fixed, $u_i$ is continuous to $s_i (s_i \in S_i)$, and is a concave function to $s_1$. Likewise, under the condition
that the channel \( p(x|y) \) is fixed, \( u_2 \) is continuous with respect to \( s_2(s_2 \in S_2) \) and a concave function with respect to \( s_2 \).

Since all the conditions in the game \( G = \{ P_1, ..., P_n; S_1, ..., S_n; u_1, ..., u_n \} \) are satisfied, that is, there is a pure strategy Nash equilibrium in the constructed standard game \( G = \{ P_1, ..., P_n; S_1, ..., S_n; u_1, ..., u_n \} \). In other words, there is a pair of pure strategies \( s_1^* = (p_1^*, p_2^*, ..., p_n^*) \) and \( s_2^* = (q_1^*, q_2^*, ..., q_n^*) \) in a standard game, which correspond to a pair of input and output variables \( X^* \) and \( Y^* \). When \( P(X^* = i) = p_i^*, 1 \leq i \leq n, p_1^* + p_2^* + ... + p_n^* = 1 \) and \( P(Y^* = j) = q_j^*, 1 \leq j \leq m, q_1^* + q_2^* + ... + q_m^* = 1 \), it makes any given \( s_2 \in S_2 \) must have \( u_1(s_1^*, s_2) \geq u_1(s_1, s_2) \) and any given \( s_1 \in S_1 \) and \( u_2(s_1, s_2^*) \geq u_2(s_1, s_2) \) must be established at the same time. Furthermore, according to the definition of channel capacity, we know of channel \( C = u_1(s_1^*, s_2^*) = u_2(s_1^*, s_2^*) \) of channel \( p(y|x) \), it means that the \( P_1 \) and \( P_2 \) reach the Nash equilibrium during the game, which is equal to the channel capacity when the player \( P_1 \) is the sender of the channel and the player \( P_2 \) is the receiver of the channels, i.e., \( u_t = C = \max_{p(x)} I(a, X; b, Y) = \max_{q(y)} I(b, Y; a, X) \).

### 7.2 Security proof

**Theorem 4.** The rational and secure two-party computation protocol based on direction entropy is secure.

**Proof.** The protocol performs security proofs in terms of safe abort and correct delivery of outputs.

Safe abort: Since both parties do not know the specific value of \( t \), the two parties can only finalize the secret after \( n \) interactions. Therefore, neither party can terminate the agreement at will, and if it is aborts at will, it cannot acquire the secret itself.

Correct delivery of output: (1) Suppose there is an adversary \( A \) who can forge a random sub-secret \( s_i' \) and send it to the participant during the secret exchange process, then the player calculates \( P[I(S; s_i') = I(S; s_i)] \leq \negl \) for the subsecret \( s_i' \). That is, the probability that the player cannot distinguish the random value is negligible.

(2) Assuming that there is an adversary \( A \) who forges a duplicate sub-secret \( s'' \) during the secret exchange process and send it to the player, then the player calculates \( P[I(S; s'') = I(S; s_i)] \leq \negl \) for the sub-secret \( s'' \). That is, the probability that the player cannot distinguish the duplicate subsecret is negligible.

(3) After the protocol is executed, all participating parties can integrate their own secrets. The adversary’s advantage is \( \text{Adv}^2_{\pi}(S, y) = P_r[y \leftarrow \oplus_{\pi}[\hat{s}] \leq \negl \), where \( \hat{s} \) represents the sub-secret forged by the adversary. That is, the probability of integrating the secret when the adversary \( A \) forges the sub-secret input is negligible.

### 7.3 Proof of fairness

**Theorem 5.** The rational and secure two-party computation protocol based on direction entropy is fair.
Proof. Since both players reach the Nash equilibrium when the channel capacity is reached. At the same time, their utility is maximized when they reach the channel capacity. In the secret exchange phase, players judge the amount of information transmitted according to their own channel capacity. In each round of information exchange, players judge the amount of information transferred based on their channel capacity. Specifically, we obtain the utility function as shown in Table 2. to get the following three situations:

**Case 1:** When $M(q) > N(p)$, it means that the probability of player $P_2$ chooses $C$ is greater than the probability of player $P_1$ chooses $C$ during the $t \in \{1, 2, ..., n\}$ round of information interaction. Overall, the mutual information of the player $P_2$ is greater than the mutual information of the player $P_1$, that is, the utility of the player $P_2$ is greater than the utility of the player $P_1$. At this time, player $P_2$ regards the information delivered by player $P_1$ as incorrect information, and $P_2$ terminates the protocol.

**Case 2:** When $M(q) < N(p)$, it means that the probability of player $P_2$ chooses $C$ is less than the probability of player $P_1$ chooses $C$ during the $t \in \{1, 2, ..., n\}$ round of information interaction. Overall, the mutual information of the player $P_1$ is greater than the mutual information of the player $P_2$, that is, the utility of the player $P_1$ is greater than the utility of the player $P_2$. At this time, player $P_1$ regards the information delivered by player $P_2$ as incorrect information, and $P_1$ terminates the protocol.

**Case 3:** When $M(q) = N(p)$, it means that the probability of player $P_1$ chooses $C$ is equal to the probability of player $P_2$ chooses $D$ during the $t \in \{1, 2, ..., n\}$ round of information interaction. The mutual information obtained by players $P_1$ and $P_2$ is equal. In other words, the utility of players $P_1$ and $P_2$ are equal. When the player chooses to cooperate with a probability close to 1, the player maximizes the utility and the protocol reaches the Nash equilibrium. In short, if the player chooses $C$ with a probability close to 1, the player successfully exchanges information and the protocol reaches the optimal state.

\[\square\]

8 | SIMULATIONS AND RESULTS

In this section, we first utilize the MATLAB software to simulate and analyze the player's utility. Subsequently, we also analyze the utility function of players when they choose cooperative strategies with different probabilities. We then analyze the performance of the protocol.

TABLE 2  Direction entropy and utility of players

| Serial | Strategy | Direction vector | Direction entropy | Utility |
|--------|----------|------------------|-------------------|---------|
| (1)    | {C, C}   | $a_i = 1, b_i = 1$ | $H(\text{dir})_1, H(\text{dir})_1$ | $[I(a, X; b, Y)|l, I(b, Y; a, X)|l]$ |
| (2)    | {C, D}   | $a_i = 1, b_i = -1$ | $H(\text{dir})_1, H(\text{dir})_{-1}$ | $[I(a, X; b, Y)|l, I(b, Y; a, X)|_{-1}]$ |
| (3)    | {D, C}   | $a_i = -1, b_i = 1$ | $H(\text{dir})_{-1}, H(\text{dir})_1$ | $[I(a, X; b, Y)|_{-1}, I(b, Y; a, X)|l]$ |
| (4)    | {D, D}   | $a_i = -1, b_i = -1$ | $H(\text{dir})_{-1}, H(\text{dir})_{-1}$ | $[I(a, X; b, Y)|_{-1}, I(b, Y; a, X)|_{-1}]$ |
8.1 Experiments

In Figure 1, we shows a graph of a player’s utility as a function of the probability of cooperation. Where $\mu$ represents the probability of player $P_1$ choosing a cooperative strategy, $\nu$ represents the probability of player $P_2$ choosing a cooperative strategy, and $u$ represents the player’s utility. It can be seen from the figure that the utility $u$ of the player gradually increases with the increase of $\mu$ and $\nu$, that is, the greater the probability that the player chooses the cooperative strategy, the more utility he obtains. In addition, when the probability of players $P_1$ and $P_2$ choosing a cooperative strategy is 1, their utility is maximized.

In Figure 2, we show a graph of the player’s probability of choosing a cooperative strategy as a function of the utility gained. Among them, we use $\mu$ to denote the probability of player $P_1$ choosing a cooperative strategy, and $u$ to denote the utility of player $P_1$. Specifically, when the $\nu$ values are 0.1, 0.3, 0.5, 0.7, and 0.9 respectively, it can be seen from the figure that the probability of player $P_1$ choosing a cooperative strategy is positively correlated with the utility function. At the same time, the probability of player $P_2$ choosing a cooperative strategy also has a positive correlation with the maximum utility value of player $P_1$. That is, player $P_2$ determines the maximum utility value of player $P_1$, and vice versa.

8.2 Performance analysis

Table 3 shows the comparison of the RSTC in the proposed scheme and the existing work. Comparing from the privacy, fairness and utility. Here, “Yes” satisfies the performance and “No” dissatisfies the performance.

We leverage game theory to propose the direction entropy-based solution. Our protocol realize the security and fairness of RSTC. Wang et al. construct the player’s utility function through information entropy, and use belief to solve the player’s fairness problem. Zhang et al. predeposit deposits and construct agreements to make players choose honest strategies to achieve the fairness of the agreement. Based on TTP, the protocol proposed by Ah-Fat et al. also achieves fairness, but due to the existence of TTP, this will lead to leakage of the privacy of

![Figure 1](wileyonlinelibrary.com)

FIGURE 1 The utility changes of player [Color figure can be viewed at wileyonlinelibrary.com]
the agreement. Different from these protocols, our protocol constructs a utility function through mutual information, and is based on directional entropy to ensure the fairness of the protocol. In addition, both players reach a Nash equilibrium when the channel capacity is reached.

9 | CONCLUSION

In this study, we proposed a rational secure two-party computation protocol based on direction entropy, which has a game nature. Although our protocol does not set a deposit or a specific utility function as the existing rational protocol, our protocol still achieves its fairness, as well as guaranteeing the correctness of the final output. We adopt the direction vector of direction entropy to quantify the player’s strategy from different dimensions. Specifically, the player chooses a cooperation strategy, and its direction vector is positive, which means that the information delivered is positive. Conversely, it means that the information transmitted is negative. Furthermore, according to the player’s behavioral preferences, we define the player’s utility based on mutual information. The proof and analysis results show that players gain utility when they comply with the protocol. What’s more, when the player chooses a

**TABLE 3** The comparison of protocol

| Serial | Literature       | Security                | Fairness  | Utility       |
|--------|------------------|-------------------------|-----------|---------------|
| (1)    | Wang et al.²¹    | Yes                     | Yes       | Entropy       |
| (2)    | Zhang et al.²³   | Safety within threshold | Yes       | Deposit       |
| (3)    | Ah-Fat et al.²⁴  | Based on TTP            | Based on TTP | /             |
| (4)    | Our protocol     | Yes                     | Yes       | Mutual information |

![FIGURE 2](Color figure can be viewed at wileyonlinelibrary.com)
cooperation strategy with a probability close to 1, its mutual information reaches the maximum, which is equal to the channel capacity.

In future work, one of our directions is to apply our protocol to a specific scenario like in Yuan et al., which will bring more valuable effects. Furthermore, in this paper, based on the direction entropy, we propose a rational secure two-party computation protocol. However, in real scenarios there are usually multiple players. Therefore, another direction is to extend our protocol to one-to-many or even many-to-many protocols.

ACKNOWLEDGMENTS
This article is supported by the National Key Project of China (No. 2020YFB1005700); Natural Science Foundation under Grant (No. 61962009, 62072127, and 62002076); National Key Research and Development Program of China (No. 2021YFA1000600); National Key Research and Development Program of Guangdong Province under Grant (2020B0101090002); Major Scientific and Technological Special Project of Guizhou Province under Grant (20183001 and 20193003); Foundation of Guangxi Key Laboratory of Cryptography and Information Security (GCIS202118); Science and Technology Support Plan of Guizhou Province ([2020]2Y011) and the Key Research and Development Program of Guangzhou (No. 202103050003); Science and Technology Program of Guangzhou, China (No. 202002030131), Guangdong basic and applied basic research fund joint fund Youth Fund (No. 2019A1515110213).

CONFLICT OF INTEREST
The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT
Data sharing is not applicable to this article as no new data were created or analyzed in this study.

ORCID
Yuling Chen http://orcid.org/0000-0002-8674-8356
Xianmin Wang http://orcid.org/0000-0003-3480-8780

REFERENCES
1. Chen Y, Dong S, Li T, Wang Y, Zhou H. Dynamic multi-key FHE in asymmetric key setting from LWE. IEEE Trans Inform Forensics Security. 2021;16:5239-5249.
2. Zhao C, Zhao S, Zhao M, et al. Secure multi-party computation: theory, practice and applications. Inform Sci. 2019;476:357-372.
3. Vu DH, Luong TD, Ho TB. An efficient approach for secure multi-party computation without authenticated channel. Inform Sci. 2020;527:356-368.
4. Li T, Wang Z, Yang G, Cui Y, Chen Y, Yu X. Semi-selfish mining based on hidden Markov decision process. Int J Intell Syst. 2021;36(7):3596-3612.
5. Chen Y, Sun J, Yang Y, Li T, Niu X, Zhou H. PSSPR: a source location privacy protection scheme based on sector phantom routing in WSNs. Int J Intell Syst. 2022;37(2):1204-1221.
6. Halpern J, Teague V. Rational secret sharing and multiparty computation. In: Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing; 2004:623-632.
7. Goldreich O. Secure multi-party computation. Manuscript Preliminary version. 1998;78:110.
8. Yao AC. Protocols for secure computations. 23rd Annual Symposium on Foundations of Computer Science (SFCS 1982), IEEE; 1982:160-164.
9. Yao ACC. How to generate and exchange secrets. 27th Annual Symposium on Foundations of Computer Science (SFCS 1986), IEEE; 1986:162-167.
10. Goldreich O, Micali S, Wigderson A. How to play any mental game, or a completeness theorem for protocols with honest majority. *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*; 2019:307-328.

11. Wang X, Ranellucci S, Katz J. Authenticated garbling and efficient maliciously secure two-party computation. *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*; 2017:21-37.

12. Gordon SD, Katz J. Partial fairness in secure two-party computation. *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer; 2010:157-176.

13. Gordon SD, Hazay C, Katz J, Lindell Y. Complete fairness in secure two-party computation. *J ACM*. 2011;58(6):1-37.

14. Groce A, Katz J. Fair computation with rational players. Annual 525 International Conference on the Theory and Applications of Cryptographic Techniques, Springer; 2012:81-98.

15. Chen Z, Tian Y, Xiong J, Peng C, Ma J. Towards reducing delegation overhead in replication-based verification: an incentive-compatible rational delegation computing scheme. *Inform Sci*. 2021;568:286-316.

16. Wang Y, Zheng Z, Wang H, Xu Q. Rational fair computation with computational sequential equilibrium. *J Comput Res Devel*. 2014;51(7):1527-1537.

17. Wang Y, Li T, Qin H, et al. A brief survey on secure multi-party computing in the presence of rational parties. *J Ambient Intell Human Comput*. 2015;6(6):807-824.

18. Dou Z, Xu G, Chen XB, Niu XX, Yang YX. Rational protocol of quantum secure multi-party computation. *Quantum Inform Process*. 2018;17(8):102641.

19. Wang Y, Liu Z, Wang H, Xu Q. Social rational secure multi-party computation. *Concurrency Computat Practice Expe*. 2014;26(5):1067-1083.

20. Garay J, Katz J, Maurer U, Tackmann B, Zikas V. Rational protocol design: Cryptography against incentive-driven adversaries. 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, IEEE; 2013:648-657.

21. Wang Y, Yang G, Li T, Li F, Tian Y, Yu X. Belief and fairness: a secure two-party protocol toward the view of entropy for IoT devices. *J Netw Comput Appl*. 2020;161(1):102641.

22. Tang W, Liu YJ, Chen YL, Yang YX, Niu XX. SLBRS: network virus propagation model based on safety entropy. *Appl Soft Comput*. 2020;97:106784.

23. Zhang X, Liu Y, Chen Y. A new entropic criterion model in rational secure two-party computation. *J Amb Intell Human Comput*. 2022;13(3):1363-1372.

24. Ah-Fat P, Huth M. Optimal accuracy-privacy trade-off for secure computations. *IEEE Trans Inform Theory*. 2018;65(5):3165-3182.

25. Yang Y, Niu X. *The general theory of information security*. Publishing house of electronics industry; 2018.

26. Tian Y, Guo J, Wu Y, Lin H. Towards attack and defense views of rational delegation of computation. *IEEE Access*. 2019;7:44037-44049.

27. Li Q, Tian Y. Rational delegation computing using information theory and game theory approach. International Conference on Multimedia Modeling, Springer; 2020:669-680.

28. Liu C, Tian Y, Xiong J, Lu Y, Li Q, Peng C. Towards attack and defense views to k-anonymous using information theory approach. *IEEE Access*. 2019;7:156025-156032.

29. Yuan F, Chen S, Liang K, Xu L. *Research on the Coordination Mechanism of Traditional Chinese Medicine Medical Record Data Standardization and Characteristic Protection Under Big Data Environment*. Shandong People's Publishing House; 2021.

**How to cite this article:** Chen Y, Ma J, Wang X, Zhang X, Zhou H. DE-RSTC: a rational secure two-party computation protocol based on direction entropy. *Int J Intell Syst*. 2022;37:8947-8967. doi:10.1002/int.22975