# Private Two-Party Set Intersection Protocol in Rational Model

Atsuko Miyaji
School of Information Science,
Japan Advanced Institute of Science and Technology,
Ishikawa, Japan.
miyaji@jaist.ac.jp

Mohammad Shahriar Rahman*
Department of Computer Science and Engineering,
University of Asia Pacific,
Dhaka, Bangladesh.
shahriar.rahman@uap-bd.edu

**Abstract**

Many data mining algorithms use privacy preserving set intersection operations. Private set operations have considered semi-honest and malicious adversarial models in cryptographic settings. Protocols in semi-honest model, requiring light computations, provide weak security. Protocols in malicious model guarantee strong security at the price of expensive computations like homomorphic encryption and zero-knowledge proof. However, practical implementations require robust and efficient protocols. In this paper, we build efficient and private set intersection avoiding the use of expensive tools like homomorphic encryption and zero-knowledge proof. Our proposed set intersection protocol is constructed in game-theoretic model. In our model, the parties are viewed as rational whereby they are assumed (only) to act in their self-interest. Our protocol satisfies computational Nash equilibrium.

**Keywords**: Privacy, Set Intersection, Rational Cryptography, Computational Nash Equilibrium

## 1 Introduction

In data mining area, private set intersection protocol allows two parties interact on their respective input sets. These protocols address several realistic privacy issues. For example, in healthcare industry, insurance companies often need to obtain information about their insured patients from other parties, such as other insurance carriers or hospitals. The insurance carriers cannot disclose the identity of inquired patients, whereas, the hospitals cannot provide any information on other patients. Fig.1 explains a set intersection operation.

Privacy-preserving set intersection protocols use different models based on the adversarial behavior assumptions. Semi-honest and malicious are the two categories of adversaries that have been considered in cryptography literature. Protocols secure in the presence of semi-honest or honest-but-curious adversaries assume that parties honestly follow all protocol specifications and do not misrepresent any information related to their inputs, e.g., set size and content (according to Goldreich's definition [11]). But, any party might passively attempt to infer additional information about the other party's input during or after protocol execution. To formalize this model, it is required that the parties involved in the protocol do not learn more information that they would in an ideal scenario assuming a trusted third party (TTP). Security in the presence of malicious parties allows arbitrary deviations from the protocol.

*Journal of Internet Services and Information Security (JISIS)*, volume: 2, number: 1/2, pp. 93-104

*Corresponding author: Dept of Computer Science and Engineering, University of Asia Pacific, House 52/1, Road 4/A, Dhanmondi, Dhaka-1209, Bangladesh, Tel: +88-01742092172

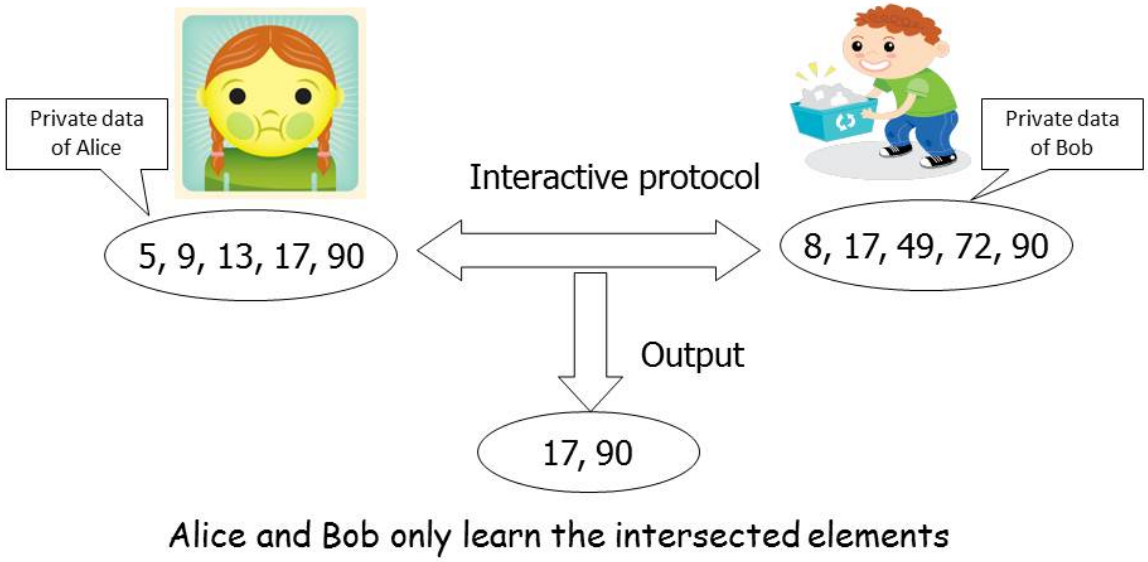



Figure 1: Privacy-preserving set intersection

In general, however, it does not prevent parties from refusing to participate in the protocol, modifying their private input sets, or prematurely aborting the protocol. Security in the malicious model is achieved if the adversary (interacting in the real protocol, without the TTP) can learn no more information than it could in the ideal scenario.

Protocols for some cryptographic tasks (e.g., secret sharing, multi-party computation) have begun to be re-evaluated in a game-theoretic setting since the work of Halpern and Teague [12] (for an overview of work in this direction, see [7, 19]). In game theoretic setting, parties are neither honest nor corrupt/malicious but are viewed as rational and are assumed to act in their self-interest. This feature is particularly interesting for data mining operations where huge collection of data is used, since parties will not deviate (i.e., abort) as there is no incentive to do so.

### 1.1 Related Work

In general, there are two types of assumptions on data distribution: vertical and horizontal partitioning. Secure distributed protocols have been developed for horizontally partitioned data for mining decision trees [22], k-means clustering [21]. Secure protocols for the vertically partitioned case have been developed for mining association rules [30], and k-means clusters [14, 29]. All of those protocols claimed to be secure only in the semi-honest model. In [8, 17], authors present two-party secure protocols in the malicious model for data mining. They follow the generic malicious model definitions from the cryptographic literature, and also focus on the security issues in the malicious model, and provide the malicious versions of the subprotocols commonly used in previous privacy-preserving data mining algorithms. Assuming that at least one party behaves in semi-honest model, they use threshold homomorphic encryption for malicious adversaries presented by Cramer et al. [5]. Since homomorphic encryption is considered too expensive [23] and zero-knowledge proof is often one of the most expensive parts of cryptographic protocols, the protocols proposed in malicious adversarial model are not very practical for operations on large data items. Set operations using commutative encryption have been proposed in [2], where the adversaries have been considered as semi-honest parties. Game theory and data mining, in

general, have been combined in [15, 18] for constructing various data mining algorithms. Rational adversaries have also been considered in privacy-preserving set operations [31, 3]. These protocols consider Nash equilibrium to analyze the rational behavior of the participating entities. As in all of cryptography, computational relaxations are meaningful and should be considered; doing so allows us to get around the limitations of the information-theoretic setting. So, analyzing set operations from the viewpoint of computational Nash equilibrium is interesting, since it gives a more realistic results. There have been several works on game theory based MPC/secret sharing schemes [1, 12, 20, 24, 9, 28, 13]. But [12, 28] require the continual involvement of the dealer even after the initial shares have been distributed or assume that sufficiently many parties behave honestly during the computation phase. Some schemes [1, 20, 24] rely on multiple invocations of protocols. Other work [13] relies on physical assumptions such as secure envelopes and ballot boxes. [9] proposed efficient protocols for rational secret sharing. But secret sharing schemes cannot be directly used for our purpose since they require much heavier computation, the existence of TTP, and their set up is different.

### 1.2 Our Contribution

In this work, we build two-party private set intersection protocol in game-theoretic setting using cryptographic primitives [1]. It is assumed that parties are neither honest nor corrupt but are instead rational and are assumed to act only in their self-interest. Our construction avoids the use of expensive cryptographic tools like homomorphic encryption and zero-knowledge proof. We have used commutative encryption as the underlying cryptographic primitive which is simple and efficient. The parties run the protocol in a sequence of $r$ rounds and learn the complete result at the end of the $r$-th round. Also, our construction does not rely on the existence of any trusted third party. It is also possible to use our protocol for computing set-union operations. We also show that our protocol satisfies computational version of strict Nash equilibrium. In short, our protocol achieves the following:

- Either of the parties may cheat with incorrect input. But cheating does not help any party to win the game.
- At any round earlier than $r$, aborting the protocol does not give any higher pay off to the aborting party than following the protocol.

**Organization of the paper:** The remainder of the paper is organized as follows: Section 2 presents the background and preliminaries. Section 3 describes the protocol model. Section 4 includes protocol construction. In Section 5, we analyze the protocol formally. Performance analysis in Section 6 is followed by some concluding remarks in Section 7.

## 2 Background and Preliminary

In this section, we will state the definitions of computational Nash equilibrium and Commutative encryption. A protocol is in Nash equilibrium if no deviations are advantageous. In other words, there is no incentive to deviate in the case of a Nash equilibrium. We assume that a party exhibits its malicious behavior by aborting early or sending non-participate message. However, a malicious party does not manipulate its own datasets to provide wrong data. Preventing malicious parties from sharing false data is difficult since the data are private and non-verifiable information. To prevent such malicious behavior, there can be auditing mechanism where a TTP can verify the integrity of data. We denote the security

[1] A preliminary version of this paper appears at IEEE WAINA 2012 [27]. This is the full version. Full proof of Theorem 1, performance analysis, and figures have been added in this version.

parameter by $n$. A function $\varepsilon$ is negligible if for all $c > 0$ there is a $n_c > 0$ such that $\varepsilon(n) < 1/n^c$ for all $n > n_c$; let *negl* denote a generic negligible function. We say $\varepsilon$ is noticeable if there exist $c, n_c$ such that $\varepsilon(n) > 1/n^c$ for all $n > n_c$.
We consider the strategies in our work as the PPT interactive Turing machines. Given a vector of strategies $\vec{\sigma}$ for two parties in the computation phase, let $u_j(\vec{\sigma})$ denote the expected utility of $P_j$, where the expected utility is a function of the security parameter $n$. This expectation is taken over the randomness of the players' strategies. Following the standard game-theoretic notation, $(\sigma'_j, \vec{\sigma}_{-j})$ denotes the strategy vector $\vec{\sigma}$ with $P_j$'s strategy changed to $\sigma'_j$.

**Definition 1.** $\Pi$ *induces a computational Nash equilibrium if for any PPT strategy* $\sigma'_1$ *of* $P_1$ *we have* $u_1(\sigma'_1, \sigma_2) \leq u_1(\sigma_1, \sigma_2) + negl(n)$*, and similarly for* $P_2$.

The following definition is stated for the case of a deviating $P_1$ (definition for a deviating $P_2$ is analogous). Let $P_1$ and $P_2$ interact, following $\sigma_1$ and $\sigma_2$, respectively. Let *mes* denote the messages sent by $P_1$, but not including any messages sent by $P_1$ after it writes to its (write-once) output tape. Then $view_2^{\Pi}$ includes the information given by the trusted party to $P_2$, the random coins of $P_2$, and the (partial) transcript *mes*. We fix a strategy $\gamma_1$ and an algorithm $A$. Now, let $P_1$ and $P_2$ interact, following $\gamma_1$ and $\sigma_2$, respectively. Given the entire view of $P_1$, algorithm $A$ outputs an arbitrary part *mes*$'$ of *mes*. Then $view_2^{A,\gamma_1}$ includes the information given by the trusted party to $P_2$, the random coins of $P_2$, and the (partial) transcript *mes*$'$.

**Definition 2.** *Strategy* $\gamma_1$ *yields equivalent play with respect to* $\Pi$*, denoted* $\gamma_1 \approx \Pi$*, if there exists a PPT algorithm A such that for all PPT distinguishers D*
$| Pr[D(1^n, view_2^{A,\gamma_1}) = 1] - Pr[D(1^n, view_2^{\Pi}) = 1] | \leq negl(n)$

**Commutative Encryption:** Our definition of commutative encryption below is similar to the constructions used in [4, 6, 10] and others. Informally, a commutative encryption is a pair of encryption functions $f$ and $g$ such that $f(g(v)) = g(f(v))$.

**Definition 3.** *Let* $\omega_k \in \{0,1\}^n$ *be a finite domain of n-bit numbers. Let* $D_1 = D_1(\omega_n)$ *and* $D_2 = D_2(\omega_n)$ *be distributions over n. Let* $A_n(x)$ *be an algorithm that, given* $x \in \omega_n$*, returns either true or false. We define distribution* $D_1$ *of random variable* $x \in \omega_n$ *to be computationally indistinguishable from distribution* $D_2$ *if for any family of PPT algorithms* $A_n(x)$*, any polynomial* $p(n)$*, and all sufficiently large n*
$Pr[A_n(x)|x \in D_1] - Pr[A_n(x)|x \in D_2] < \frac{1}{p(n)}$
*where x is distributed according to* $D_1$ *or* $D_2$*, and* $Pr[A_n(x)]$ *is the probability that* $A_n(x)$ *returns true.*

**Definition 4.** *A commutative encryption F is a computable (in polynomial time) function* $f : KeyF \times DomF \rightarrow DomF$*, defined on finite computable domains, that satisfies all properties listed below. We denote* $f_e(x) \equiv f(e,x)$.

*(1) Commutativity: For all* $e, e' \in KeyF$ *we have* $f_e \circ f_{e'} = f_{e'} \circ f_e$

*(2) Each* $f_e : DomF \rightarrow DomF$ *is a bijection.*

*(3) The inverse* $f_e^{-1}$ *is also computable in polynomial time given e.*

*(4) The distribution of* $\langle x, f_e(x), y, f_e(y) \rangle$ *is indistinguishable from the distribution of* $\langle x, f_e(x), y, z \rangle$*, where* $x, y, z \in_r DomF$ *and* $e \in_r KeyF$.

Informally, Property 1 says that when we compositely encrypt with two different keys, the result is the same irrespective of the order of encryption. Property 2 says that two different values will never have the same encrypted value. Property 3 says that given an encrypted value $f_e(x)$ and the encryption key $e$,

we can find $x$ in polynomial time. Property 4 says that given a value $x$ and its encryption $f_e(x)$ (but not the key $e$), for a new value $y$, we cannot distinguish between $f_e(y)$ and a random value $z$ in polynomial time. Thus we can neither encrypt $y$ nor decrypt $f_e(y)$ in polynomial time. Note that this property holds only if $x$ is a random value from $DomF$, i.e., the adversary does not control the choice of $x$.

*Remark:* One-way functions exist under the discrete log-type hardness assumption; namely, exponentiation modulo a prime $p$. To be precise, given that $DomF$ is all quadratic residues modulo $p$ where $p$ is a safe prime and $q = (p-1)/2$ such that $p$ and $q$ are primes, and $KeyF$ is $\{1,2,\ldots q-1\}$, the exponentiation function $f_e(x) = x^e \bmod p$ has the properties of commmutative encryption. That is, the powers commute, each of the powers $f_e$ is a bijection with its inverse, and indistinguishability is satisfied under the discrete log-type hard problem.

# 3 Model

In a typical protocol, parties are viewed as either honest or semi-honest/malicious. To model rationality, we consider players' utilities. Here we assume that $\mathscr{F} = \{f : X \times Y \to Z\}$ is a functionality where $\mid X \mid = \mid Y \mid$ and their domain is polynomial in size ($poly(n)$). Let $\mathscr{D}$ be the domain of output which is polynomial in size. The function returns a vector $I$ that represents the set intersection where $I_t$ is set to one if item $t$ is in the set intersection. In other words, for all the data items of the parties (i.e., $X$ and $Y$), we will compute $X \cap Y$, and we get $I$ as the output of the function. Clearly for calculating set intersection, we need to calculate $x_l \wedge y_l$ for each $l$ where $x_l \in X$ and $y_l \in Y$. Similarly, for set-union, we need to calculate $x_l \vee y_l$ for all $l$. This can be rewritten as $\neg(\neg x_l \wedge \neg y_l)$. Computing the set-union is thus straight forward.

Given that $j$ parties are active during the computation phase, let the outcome $o$ of the computation phase be a vector of length $j$ with $o_j = 1$ iff the output of $P_j$ is equal to the exact intersection (i.e., $P_j$ learns the correct output). Let $\nu_j(o)$ be the utility of player $P_j$ for the outcome $o$. Following [12, 9], we make the following assumptions about the utility functions of the players:
- If $o_j > o'_j$, then $\nu(o_j) > \nu(o'_j)$
- If $o_j = o'_j$ and $\sum_j o_j < \sum_j o'_j$, then $\nu(o_j) > \nu(o'_j)$

In other words, player $P_j$ first prefers outcomes in which he learns the output; otherwise, $P_j$ prefers strategies in which the fewest number of other players learn the result (in our two-party case, the other player learns). From the point of view of $P_j$, we consider the following three cases of utilities for the outcome $o$ where $U^* > U > U'$:
- If only $P_j$ learns the output, then $\nu_j(o) = U^*$.
- If $P_j$ learns the output and the other player does also, then $\nu_j(o) = U$.
- If $P_j$ does not learn the output, then $\nu_j(o) = U'$.

So, we have the expected utility of a party who outputs a random guess for the output (assuming other party aborts without any output, or with the wrong output) as follows: $U_{rand} = \frac{1}{|\mathscr{D}|} \cdot U^* + (1 - \frac{1}{|\mathscr{D}|}) \cdot U'$. Also, we assume that $U > U_{rand}$; else players have almost no incentive to run the computation phase at all. As in [9], we make no distinction between outputting the wrong secret and outputting a special 'don't know' symbol- both are considered as a failure to output the correct output.

# 4 Rational Set Intersection Protocol

## 4.1 An Overview of the Protocol

Let $x$ denote the input of $P_1$, let $y$ denote the input of $P_2$, and let $f$ denote the set intersection function they are trying to compute. Our protocol is composed of two stages, where the first stage can be viewed

as a key generation stage and the second stage that computes the intersection takes place in a sequence of $r = r(n)$ iterations. More specifically, in the key generation stage the parties generate their encryption keys. They also choose $i^* \in r$ according to some random distribution $\alpha$ in which step they can learn the complete intersection result. In every round $i \in \{1,\ldots,r\}$ the parties exchange the encrypted data for the current round, which enables $P_1$ and $P_2$ to perform the Intersection Computation. Clearly, when both parties are honest, the parties produce the same output result which is uniformly distributed. Briefly speaking, the stages have the following form:
**Key Generation Stage**:

- Each party randomly chooses a secret key for itself, i.e. $e_S \in KeyF$ for $P_1$ and $e_R \in KeyF$ for $P_2$, for commutative encryption.
- A value $i^* \in \{1,\ldots,r\}$ is chosen according to some random distribution $0 < \alpha < 1$ where $\alpha$ depends on the players' utilities (discussed later). This represents the iteration, in which parties will learn the complete result.

**Intersection Computation Stage**:
In each iteration $i$, for $i = 1,\ldots,r$, the parties do the following: First, $P_2$ sends $c_1$ to $P_1$ and then $P_1$ sends $c_2$ to $P_2$, where $c_1$ and $c_2$ are the ciphertexts computed by party $P_1$ and $P_2$ respectively. After receiving the ciphertexts, $P_2$ and $P_1$ compute the set intersection using commutative property of the encryption scheme. If a party aborts in some iteration $i$, then the other party outputs the value computed in the previous iteration. If some party fails to follow the protocol, the other party aborts. In fact, it is rational for $P_j$ to follow the protocol as long as the expected gain of deviating is positive only if $P_j$ aborts exactly in iteration $i^*$; and is outweighed by the expected loss if $P_j$ aborts before iteration $i^*$. The intersection computation phase proceeds in a series of iterations, where each iteration consists of one message sent by each party.

## 4.2 Protocol Construction

As described above, our protocol $\Pi$ consists of two stages. Let $p$ be an arbitrary polynomial, and set $r = p\cdot \mid Y \mid$. We implement the first stage of $\Pi$ using a key generation algorithm. This functionality returns required keys to each party. In the second stage of $\Pi$, the parties exchange their ciphertexts in a sequence of $r$ iterations. The protocol returns $I$ at the end of the operations on all the data items as follows:

**Key Generation Stage:**

- Each party randomly chooses a secret key $e_1 \in KeyF$ for $P_1$ and $e_2 \in KeyF$ for $P_2$ for commutative encryption.
- A value $i^* \in \{1,\ldots,r\}$ is chosen according to some random distribution $0 < \alpha < 1$ where $\alpha$ depends on the players' utilities. This represents the iteration, in which parties will learn the complete result.

**Set Intersection Computation Stage:**
for all $i$ do

(1) $P_2$ encrypts its input dataset $Z_2 = f_{e_2}(Y)$ and sends $Z_2$ to $P_1$.

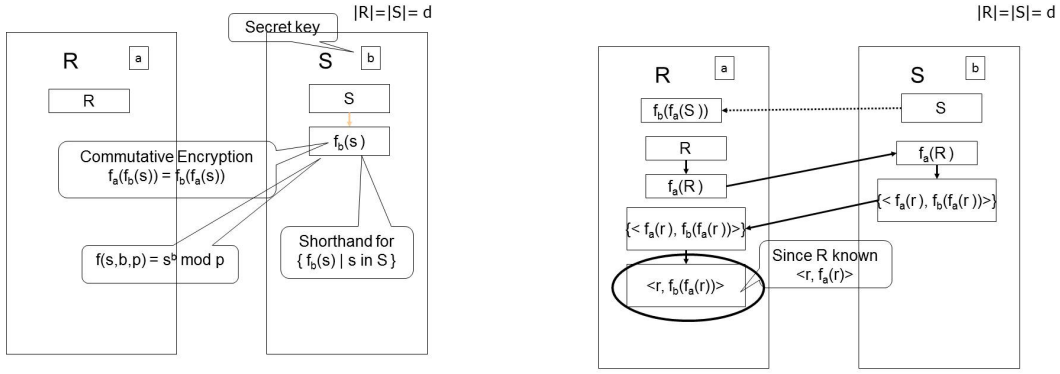
(2) $P_1$ encrypts its input dataset $Z_1 = f_{e_1}(X)$ and sends $Z_1$ to $P_2$.

(3) For $P_2$, if it has not received any message from $P_1$ then output the result of iteration $i-1$ and halt. Otherwise, compute $Z_2' = f_{e_2}(f_{e_1}(X))$ and sends the pairs $\langle Z_1, Z_2' \rangle$ to $P_1$.

(4) For $P_1$, if it has not received any message from $P_2$ then output the result of iteration $i-1$ and halt.. Otherwise, compute $Z_1' = f_{e_1}(f_{e_2}(Y))$. Also, from pairs $\langle f_{e_1}(x), f_{e_2}(f_{e_1}(x))\rangle$ obtained in step 5 for each $x \in X$, it creates pair $\langle x, f_{e_2}(f_{e_1}(x))\rangle$ replacing $f_{e_1}(x)$ with corresponding $x$.

(5) For $P_1$, for $x \in X$ for which $(f_{e_2}(f_{e_1}(x)) \in Z_1'$, these values form the intersection result $I = X \cap Y$.

(6) $P_2$ computes and outputs $I$ similarly.




(a) Commutative Encryption (b) Exchange of Ciphertexts

Figure 2: Proposed Intersection Protocol for One Round

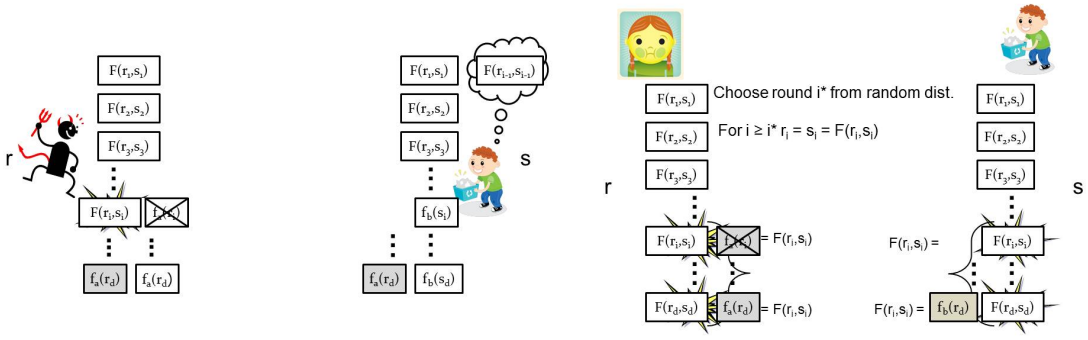
Fig. 1 simply describes the proposed protocol. Two parties, having R and S as their data sets, and $a$ and $b$ as their secret keys, perform the intersection protocol using commutative encryption. We provide an intuitive description of the computation phase. Let us assume that $P_1$ has a set of data items $\{Tokyo, London, Washington, Beijing\}$ and $P_2$ has $\{Tokyo, Paris, Toronto, Rome\}$. At first, they encrypt each of the items with their secret keys and exchange the ciphertexts with each other at each round (here, we will have 4 rounds at most to complete the whole protocol). After a party receives the ciphertext from the other party, it reencrypts the ciphertext using its own secret key. After they exchange such data items at each round, due to the commutative property of the underlying encryption scheme, they will come to know the intersection output (1 if the otems match, 0 otherwise). For this example, they will come to know that *Tokyo* is the intersected result from round 1, and all the subsequent rounds will output 0. So, the final result they will know only is the intersected value.

# 5 Protocol Analysis

Here we will give some intuition as to why the reconstruction phase of $\Pi$ is a computational Nash equilibrium for an appropriate choice of $\alpha$. Let us assume that $P_2$ follows the protocol, and $P_1$ deviates from the protocol. (It is easier to analyze the deviations by $P_2$ since $P_2$ starts in every iteration.) When $P_1$ aborts in some iteration $i < i^*$, the best strategy $P_1$ can adopt is to output $Z_1^{i^*}$ hoping that $i = i^*$. Fig. 3 shows us the protocol in many rounds.

Thus, following this strategy, the expected utility that $P_1$ obtains can be calculated as follows:

- $P_1$ aborts exactly in iteration $i = i^*$. In this case, the utility that $P_1$ gets is at most $U^*$.
- When $i < i^*$, $P_1$ has 'no information' about correct $I$ and so the best it can do is guess. In this case, the expected utility of $P_1$ is at most $U_{rand}$.




(a) Protocol in Rounds with Deviation Attempt (b) Learning from Iteration

Figure 3: Protocol for Many Rounds

Considering the above, $P_1$'s expected utility of following this strategy is at most:

$$\alpha \times U^* + (1-\alpha) \times U_{rand}$$

Now, it is possible to set the value of $\alpha$ such that the expected utility of this strategy is strictly less than $U$, since $U_{rand} < U$ by assumption. In such a case, $P_1$ has no incentive to deviate. Since there is always a unique valid message a party can send and anything else is treated as an abort, it follows that the protocol $\Pi$ induces a computational Nash equilibrium.

**Theorem 1.** *The protocol* $\Pi$ *induces a computational Nash equilibrium given that* $0 < \alpha < 1$, $U > \alpha \times U^* + (1-\alpha) \times U_{rand}$, *and the properties of commutative encryption.*

*Proof:* We first show that $\Pi$ is a valid set intersection protocol. The proof method is similar to that of [9]. Computational secrecy follows from the proof, below, that the intersection computation is a computational Nash equilibrium. Because if secrecy did not hold then computing the output locally and not participating in the intersection computation phase at all would be a profitable deviation. We next focus on correctness. Assuming both parties run the protocol honestly, the output is computed correctly if the properties of commutative encryption are not achieved, which has negligible probability. We next show that $\Pi$ induces a computational Nash equilibrium. Assume $P_2$ follows the strategy $\sigma_2$ prescribed by the protocol, and let $\sigma_1'$ denote any PPT strategy followed by $P_1$. (The other case, where $P_1$ follows the protocol and we look at deviations by $P_2$, follows similarly with an even simpler approach.) In a given execution of the computation phase, let $i$ denote the iteration in which $P_1$ aborts (where an incorrect message is viewed as an abort); if $P_1$ never aborts then set $i = 1$. Let *early* be the event that $i < i^*$; let *exact* be the event that $i = i^*$; and let *late* be the event that $i > i^*$. Let *correct* be the event that $P_1$ outputs the correct output. We will consider the probabilities of these events in two experiments: the experiment defined by running the actual intersection computation scheme, and a second experiment where $P_1$ is given $Z_1$, $t_1$ chosen uniformly at random from the appropriate ranges. We denote the probabilities in the first experiment by $Pr_{real}[\cdot]$, and the probabilities in the second experiment by $Pr_{ideal}[\cdot]$. We have the following equation using the fact (as discussed above) that whenever late occurs $P_2$ outputs the correct result. Since when both parties follow the protocol $P_1$ gets utility $U$, we need to show that there exists a negligible function $\varepsilon$ such that $u_1(\sigma_1', \sigma_2) \leq U + \varepsilon(n)$:

$u_1(\sigma_1', \sigma_2) \leq U^* \times Pr_{real}[exact] + U^* \times Pr_{real}[correct \wedge early] + U' \times Pr_{real}[\overline{correct} \wedge early] + U \times Pr_{real}[late]$

Now we have the following claim that follows from the indistinguishability property of commutative encryption:
*Claim 1*: There exists a negligible function $\varepsilon$ such that

$$\begin{aligned} \mid Pr_{real}[exact] &- Pr_{ideal}[exact] \mid \leq \varepsilon(n) \\ \mid Pr_{real}[late] &- Pr_{ideal}[late] \mid \leq \varepsilon(n) \\ \mid Pr_{real}[correct \wedge early] &- Pr_{ideal}[correct \wedge early] \mid \leq \varepsilon(n) \\ \mid Pr_{real}[\overline{correct} \wedge early] &- Pr_{ideal}[\overline{correct} \wedge early] \mid \leq \varepsilon(n) \end{aligned}$$

Now, we have $U_{ideal} = U^* \cdot Pr_{ideal}[exact] + U^* \cdot Pr_{ideal}[correct \wedge early] + U' \cdot Pr_{ideal}[\overline{correct} \wedge early] + U \cdot Pr_{ideal}[late]$

From Claim 1 we get that $\mid u_1(\sigma_1', \sigma_2) - U_{ideal} \mid \leq \varepsilon(n)$ for some negligible $\varepsilon$. We bound $U_{ideal}$ as follows: Let $abort = exact \vee early$, so that *abort* is the event that $P_1$ aborts before iteration $i^* + 1$. We have $Pr_{ideal}[exact \mid abort] = \alpha$ and $Pr_{ideal}[correct \mid early] = 1/\mathscr{D}$. It is easy to find that $U_{ideal} = U + (\alpha \cdot U^* + (1-\alpha) \cdot U_{rand} - U) \cdot Pr_{ideal}[abort] \leq U$ given that $\alpha \cdot U^* + (1-\alpha) \cdot U_{rand} - U < 0$. This shows that $\Pi$ induces a computational Nash equilibrium.

## 6 Efficiency Analysis

The complexity of secure intercetion protocol proposed in Kantarcioglu can be expressed as $O(n)$ where $n$ is the size of the input dataset, in other words the size of data items to be processed. Therefore, the efficiency of the protocol is highly dependent on the size of the input dataset. The complexity displayed in Table 1 involves both communication and computation times, and the difference can be explained with the impact of communication and computation overhead that the ZK proof brings. While the overhead of ciphertext computation and communication in our protocol are similar to that in Kantarcioglu's work, we do not need any computation and communication of ZK proof in our protocol. This is a drastic reduction in computation and communication cost. In our work, the round complexity is linear to the number of items and the inverse of geometric distribution $\alpha$. As discussed earlier, use of ZK proof and homomorphic encryption leads to inefficiency in practical world and we want to avoid using the expensive tool like ZK proofs and homomorphic encryptions. Note that in [26], the use of Verifiable Random Function (VRF) has made the protocol an expensive one.

Table 1: Secure Set-Intersection: Performance Comparison

| Schemes | Computation | | Communication | | Round | Tools |
|---|---|---|---|---|---|---|
| | ciphertext | ZKP | ciphertext | ZKP | | |
| Malicious ([17]) | $O(n)$ | $O(n)$ | $O(n)$ | $O(n)$ | $O(n)$ | Homomorphic, ZKP |
| Malicious ([8]) | $O(n)$ | $O(1)$ | $O(n)$ | $O(1)$ | $O(n)$ | Homomorphic, ZKP |
| Covert ([25]) | $O(n)$ | – | $O(n)$ | – | $O(1)$ | Homomorphic |
| Rational ([26]) | $O(n)$ | – | $O(n)$ | – | $O(n\alpha^{-1})$ | VRF |
| Rational (This work) | $O(n)$ | – | $O(n)$ | – | $O(n\alpha^{-1})$ | Commutative |

While it is difficult to compare our protocols in malicious and rational models due to their construction methods, here our table shows the comparison based on the number of data items to be processed. As for the other parameters in rational model, the share size is $|t| + O(k)$, where $t$ is the size of data items and $k$ is the security parameter. The round complexity of the protocol for each item is $O(\alpha^{-1})$, where $\alpha$ is the geometric distribution used to pick up the value of $i^*$ (typically, we will need only two rounds for

each items in our protocol). Clearly, the our model requires much lighter computation than the protocol designed in malicious, model and performs even better than the covert or rational model ([26]) in terms of computational overhead due to the fact that VRF, homomorphic encryption and ZKP are expensive tools.

# 7 Conclusion

We have proposed a private two-party set intersection protocol using commutative encryption as the underlying cryptographic primitive. Our protocol is in rational model whereby parties are viewed neither malicious nor semi-honest. We avoid relying on expensive cryptographic tools like homomorphic encryption and zero knowledge proof. Our protocol satisfies computational Nash equilibrium. As for the future work achieving a more advanced gane-theoretic property 'strict computational Nash equilibrium' in multi-party environment would be interesting.

# References

[1] I. Abraham, D. Dolev, R. Gonen, and J. Halpern. Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation. In *Proc. of the 25th annual ACM symposium on Principles of distributed computing (PODC'06), Denver, Colorado, USA*, pages 53–62. ACM Press, 2006.

[2] R. Agrawal, A. Evfimievski, and R. Srikant. Information sharing across private databases. In *Proc. of the 2003 ACM SIGMOD international conference on Management of data (SIGMOD'03), San Diego, California*, pages 86–97. ACM Press, 2003.

[3] R. Agrawal and E. Terzi. On honesty in sovereign information sharing. In *Proc. of the 10th International Conference on Extending Database Technology (EBDT'06), Munich, Germany, LNCS*, volume 3896, pages 240–256. Springer-Verlag, March 2006.

[4] J. C. Benaloh and M. de Mare. One-way accumulators: A decentralized alternative to digital sinatures (extended abstract). In *Proc. of the 1993 International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT'93), Lofthus, Norway, LNCS*, volume 765, pages 274–285. Springer-Verlag, May 1993.

[5] R. Cramer, I. Damgård, and J. B. Nielsen. Multiparty computation from threshold homomorphic encryption. In *Proc. of the 2001 International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT'01), Innsbruck, Austria, LNCS*, volume 2045, pages 280–299. Springer-Verlag, May 2001.

[6] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[7] Y. Dodis and T. Rabin. *Algorithmic Game Theory*. Cambridge University Press, 2001.

[8] K. Emura, A. Miyaji, and M. S. Rahman. Efficient privacy-preserving data mining in malicious model. In *Proc. of the 6th International Conference on Advanced Data Mining and Applications (ADMA'10), Chongqing, China, LNCS*, volume 6440, pages 370–382. Springer-Verlag, November 2010.

[9] G. Fuchsbauer, J. Katz, and D. Naccache. Efficient rational secret sharing in standard communication networks. In *Proc. of the 7th Theory of Cryptography Conference (TCC'10), Zurich, Switzerland, LNCS*, volume 5978, pages 419–436. Springer-Verlag, February 2010.

[10] T. E. Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469–472, 1985.

[11] O. Goldreich. *Foundations of Cryptography: Basic Applications*. Cambridge University Press, 2004.

[12] J. Y. Halpern and V. Teague. Rational secret sharing and multiparty computation: extended abstract. In *Proc. of the 36th Annual ACM Symposium on Theory of Computing (STOC'04), Chicago, IL, USA*, pages 623–632. ACM Press, June 2004.

[13] S. Izmalkov, S. Micali, and M. Lepinski. Rational secure computation and ideal mechanism design. In *Proc. of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05), Pittsburgh, Pennsylvania, USA*, pages 585–595. IEEE, October 2005.

[14] G. Jagannathan and R. N. Wright. Privacy-preserving distributed k-means clustering over arbitrarily partitioned data. In *Proc. of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'05), Chicago, IL, USA*, pages 593–599. ACM Press, August 2005.

[15] W. Jiang, C. Clifton, and M. Kantarcioglu. Transforming semi-honest protocols to ensure accountability. *Data & Knowledge Engineering*, 65(1):57–74, 2008.

[16] M. Kantarcioglu and C. Clifton. Privately computing a distributed k-nn classifier. In *Proc. of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'04), Pisa, Italy, LNCS*, volume 3202, pages 279–290. Springer-Verlag, September 2004.

[17] M. Kantarcioglu and O. Kardes. Privacy-preserving data mining in the malicious model. *International Journal of Information and Computer Security*, 2(4):353–375, 2009.

[18] M. Kantarcioglu and R. Nix. Incentive compatible distributed data mining. In *Proc. of the 2nd IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT'10), Minnesota, USA*, pages 735–742. IEEE, August 2010.

[19] J. Katz. Bridging game theory and cryptography: Recent results and future directions. In *Proc. of the 5th Theory of Cryptography Conference (TCC'08), New York, USA, LNCS*, volume 4948, pages 251–272. Springer-Verlag, March 2008.

[20] G. Kol and M. Naor. Cryptography and game theory: Designing protocols for exchanging information. In *Proc. of the 5th Theory of Cryptography Conference (TCC'08), New York, USA, LNCS*, volume 4948, pages 320–339. Springer-Verlag, March 2008.

[21] X. Lin, C. Clifton, and M. Y. Zhu. Privacy-preserving clustering with distributed em mixture modeling. *Knowledge and Information Systems*, 8(1):68–81, 2005.

[22] Y. Lindell and B. Pinkas. Privacy preserving data mining. *Journal of Cryptology*, 15(3):177–206, 2002.

[23] J. Liu, Y.-H. Lu, and C.-K. Koh. Performance analysis of arithmetic operations in homomorphic encryption. `http://www.http://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=1403&context=ecetr`, last viewed May 2012, 2010.

[24] A. Lysyanskaya. Rationality and adversarial behavior in multi-party computation. In *Proc. of the 26th Annual International Cryptology Conference (CRYPTO'06), Santa Barbara, California, USA, LNCS*, volume 4117, pages 180–197. Springer-Verlag, August 2006.

[25] A. Miyaji and M. S. Rahman. Privacy-preserving data mining in presence of covert adversaries. In *Proc. of the 6th International Conference on Advanced Data Mining and Applications (ADMA'10) Part I, Chongqing, China, LNCS*, volume 6440, pages 429–440. Springer-Verlag, November 2010.

[26] A. Miyaji and M. S. Rahman. Privacy-preserving data mining: A game-theoretic approach. In *Proceedings of the Data and Applications Security and Privacy XXV - 25th Annual IFIP WG 11.3 Conference (DBSec 2011), VA, USA, (LNCS)*, volume 6818, pages 186–200. Springer-Verlag, July 2011.

[27] A. Miyaji and M. S. Rahman. Privacy-preserving set operations in the presence of rational parties. In *Proc. of the 26th International Conference on Advanced Information Networking and Applications Workshops (WINA'12), Fukuoka, Japan*, pages 869–874. IEEE, March 2012.

[28] S. J. Ong, D. C. Parkes, A. Rosen, and S. P. Vadhan. Fairness with an honest minority and a rational majority. In *Proc. of the 6th Theory of Cryptography Conference (TCC'09), San Francisco, CA, USA, LNCS*, volume 5444, pages 36–53. Springer-Verlag, March 2009.

[29] C. Su, F. Bao, J. Zhou, T. Takagi, and K. Sakurai. Security and correctness analysis on privacy-preserving *k*-means clustering schemes. *IEICE Transactions*, 92-A(4):1246–1250, 2009.

[30] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proc. of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'02), Edmonton, Alberta, Canada*, pages 639–644. ACM Press, July 2002.

[31] N. Zhang and W. Zhao. Distributed privacy preserving information sharing. In *Proc. of the 31st International Conference on Very Large Data Bases (VLDB'05), Trondheim, Norway*, pages 889–900. ACM Press, August 2005.

**Atsuko Miyaji** received the B. Sc., the M. Sc., and the Dr. Sci. degrees in mathematics from Osaka University, Osaka, Japan in 1988, 1990, and 1997 respectively. She worked at Panasonic Co., LTD from 1990 to 1998 and engaged in research and development for secure communication. She joined as an associate professor at the Japan Advanced Institute of Science and Technology (JAIST) in 1998. She has also joined the computer science department of the University of California, Davis since 2002. She has been a professor at the Japan Advanced Institute of Science and Technology (JAIST) since 2007 and the director of Library of JAIST since 2008. Her research interests include the application of number theory into cryptography and information security. She received Young Paper Award of SCIS'93 in 1993, Notable Invention Award of the Science and Technology Agency in 1997, the IPSJ Sakai Special Researcher Award in 2002, the Standardization Contribution Award in 2003, Engineering Sciences Society: Certificate of Appreciation in 2005, the AWARD for the contribution to CULTURE of SECURITY in 2007, IPSJ/ITSCJ Project Editor Award in 2007, 2008, 2009, and 2010, the Director-General of Industrial Science and Technology Policy and Environment Bureau Award in 2007, Editorial Committee of Engineering Sciences Society: Certificate of Appreciation in 2007, DoCoMo Mobile Science Awards in 2008, Advanced Data Mining and Applications (ADMA 2010) Best Paper Award, and The chief of air staff: Letter of Appreciation Award. She is a member of the International Association for Cryptologic Research, the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, and the Mathematical Society of Japan.

**Mohammad Shahriar Rahman** is currently working as an assistant professor at the department of computer science and engineering of the University of Asia Pacific, Bangladesh. He completed B.Sc. (Hons) from the University of Dhaka, Bangladesh, in 2006. He received M.Sc. and Ph.D. from Japan Advanced Institute of Science and Technology (JAIST) in 2009 and 2012, respectively. His research interests include applied cryptography, privacy preserving data mining, privacy in resource constrained devices, and game theory. Rahman is the recipient of Japan Government's Monbukagakusho scholarship from October 2006 to March 2012 for M.Sc. and Ph.D. studies, the IEICE Excellent Student Award 2008, the Best Paper Award at the 6th International Conference on Advanced Data Mining and Applications (ADMA'10), and Outstanding Performance Award from JAIST for Ph.D. studies.