# 基于无匹配差错的 PSI 计算

巩林明[1,2] 王道顺[3] 刘沫萌[1] 高全力[1] 邵连合[1] 王明明[1]

[1](西安工程大学计算机科学学院陕西省服装设计智能化重点实验室/新型网络智能信息服务国家地方联合工程研究中心 西安 710048)

[2](西安工程大学陕西省功能性服装面料重点实验室 西安 710048)

[3](清华大学计算机科学与技术系 北京 100084)

**摘 要** 分布式计算有很多应用需要参与各方协同执行集合的一些计算但不泄露各自数据集的信息. 保密集合交集(private set intersection, PSI)计算已经成为数据匹配、数据挖掘、推荐系统等应用中保护用户隐私的一个重要工具. 本文的主要工作是构造无匹配差错的安全两方保密集合交集运算协议. 着重探讨三个问题:(1)开发构造无匹配差错的两方保密集合交集计算所需要的工具(①面向有理数且具有语义安全性的加密方案,②便于集合匹配计算的称之为集合的定长向量编码方法);(2)无匹配差错的两方保密集合交集计算问题;(3)元素为有理数的保密集合交集计算问题. 首先在标准模型下设计了一个能够加密有理数的方案,并证明了该方案能抗自适应性地选择明文攻击;而后又提出了一种便于集合匹配计算的,称之为集合的定长向量编码方法;最后基于有理数加密方案和集合的定长向量编码方法构造了两个面向有理数的、无匹配差错的两方保密集合交集协议. 与先前的两方保密集合交集协议相较之,这两个协议不仅解决了无匹配差错的两方保密集合交集计算,还拓展了保密集合交集问题中隐私保护的范畴:除了可以保护各参与方的隐私数据外,还可以保护各参与方隐私数据的数量.

**关键词** 保密集合交集;有理数加密;语义安全;安全两方计算;集合的定长向量编码

**中图法分类号** TP301 **DOI 号** 10.11897/SP.J.1016.2020.01769

# PSI Computation Based on No Matching Errors

GONG Lin-Ming[1,2] WANG Dao-Shun[3] LIU Mo-Meng[1] GAO Quan-Li[1]
SHAO Lian-He[1] WANG Ming- Ming[1]

[1] (*The National and Local Joint Engineering Research Center for Advanced Networking & Intelligent Information Service/ Shaanxi Key Laboratory of Clothing Intelligence, School of Computer Science, Xi'an Polytechnic University, Xi'an* 710048)

[2] (*Shaanxi Key Laboratory on Functional Cloths, Xi'an Polytechnic University, Xi'an* 710048)

[3](*Department of Computer Science and Technology, Tsinghua University, Beijing* 100084)

**Abstract** In a distributed computing, many applications require multi-parties to jointly perform set operations. After the collaborative computing, except the result of the operation (equal to the result of the plaintext operation), the participating parties should not get the private information about other parties. This collaboratively distributed computation is called private set computation. Private set intersection (PSI) is an important research field of private set computation. In recent years, PSI has become an important tool for protecting user privacy in applications such as data matching, data mining, and recommendation systems, etc.. In this paper, the goal is to construct private set intersection protocols using encryption scheme with

收稿日期:2018-09-25;在线发布日期:2020-02-07. 本课题得到国家自然科学基金(61972225, 61902164, 61601358, 61672426, 61902300, 61902303, 11847101)、国家科技支撑计划子课题(2018YFB1004501)、西安工程大学博士科研启动基金(107020331)、陕西省教育厅重点科学研究计划项目(20JS052)、陕西省2020年技术创新引导专项计划(2020CGXNG-012)资助. **巩林明**(通信作者),博士,讲师,主要研究领域为密码学与信息安全. E-mail: glmxinjing@163.com. **王道顺**(通信作者),博士,副教授,主要研究领域为密码算法、视频智能行为分析、多媒体安全与取证,E-mail: daoshun@mail.tsinghua.edu.cn. **刘沫萌**,博士,讲师,主要研究领域为密码学与信息安全. **高全力**,博士,讲师,主要研究领域为网络与信息安全. **邵连合**,博士,副教授,主要研究领域为信息安全与量子智能信息计算. **王明明**,博士,副教授,主要研究领域为密码学与信息安全.

rational numbers. Three specific issues are covered. The first issue is developing tools required to construct the two-party PSI protocols without matching errors (a rational number-oriented encryption scheme with semantic security and a set encoding method called set encoding with a fixed-length vector that is convenient for set matching calculation). The second issue is studying the problem of precise evaluation in the two-party private set intersection. Both participants have a private set respectively. After implementing a protocol for computing private set intersection, they aim to achieve (1) the result of the collaborative computation is 100% equal to the result calculated directly in plaintext. Unlike some previous confidential two-party intersection protocols, the results of collaborative calculation may have errors, and the scope of errors is difficult to be defined; (2) after the collaborative calculation, the two sides do not disclose the elements of their respective sets, nor do they disclose the potential of their respective sets. That is to say, after completing the confidential calculation, the results of the collaborative calculation by these two participants must be correct, and both participants can not get any other information about each other (the elements of the set, the cardinality of the set) except for the common elements of their private set. The third issue is studying the problem of two-party private intersection computing whose elements are rational numbers: Both participants respectively have a private set, in which the elements are rational numbers. By jointly implementing a PSI calculation protocol, they aim to achieve (1) the result of the collaborative calculation is 100% equal to the result calculated directly in plaintext; (2) the two sides do not disclose the elements of their respective sets, nor do they disclose the cardinality of their respective sets. To answer those questions, firstly, a scheme capable of encrypting rational numbers is designed under the standard model, and it is proved that the scheme can resist adaptive selection plaintext attack. Then, a set encoding called set encoding with a fix-length vector is proposed to facilitate set matching calculation. Finally, based on the proposed encryption scheme with rational numbers and the proposed set encoding with a fix-length vector, we present two two-parties private intersection protocols with rational elements without matching errors. Compared with the previous two-party private intersection protocols, these two protocols not only extend the category of privacy protection in the problem of PSI but also solve the precise evaluation of two-party private intersection: in addition to protecting the private data of each party, the amount of private data of each participant can also be protected.

**Keywords** private set intersection; encryption with rational numbers; semantic security; two-party secure computation; set encoding with fix-length vector

## 1 引 言

随着分布式计算在各领域的深入发展，越来越多的应用需要参与各方利用他们的私有数据集合协同执行一些集合运算，协同运算结束后，除了运算结果(等于明文运算的结果)外，参与各方不会得到有关其它各方的私有信息，这种分布式协同计算被称为保密集合计算. 保密集合交集计算是保密集合计算领域的一个重要研究方向. 它在保密服务预约、保密信息匹配[1]、保密数据挖掘[2]、保密推荐系统[3]，保密计算广告转化率[4]等领域有着广泛应用，已经成为这些领域保护用户隐私的一个重要工具.

从可计算的角度来讲，基于“多项式验根”法、“多个函数间提取公因式”方法、集合元素用数字签名技术实施认证以及“不经意伪随机函数(Pseu-Dorandom Function, PRF)计算”思想构造的 PSI 协议，例如协议[1,5-7]分别为分布式环境下的保密集合计算问题开辟了一种可以解决问题的新方法. 然而采用前三种方法设计的保密集合交集计算协议，协议执行结束计算结果可能存在差错，且差错的范围难以界定；基于不经意伪随机函数计算的保密集合交集计算协议虽然不会泄露双方集合中的私密数据，但会泄露拥有 PRF 密钥 $\mathcal{K}_{prf}$ 一方集合的势.

较基于“多项式验根”法、“多个函数间提取公因式”方法、数字签名技术所构造的保密集合交集计算协议，采用协同计算结果必定等于交集思想所

构造的保密集合交集计算(被称为保密集合交集的精确计算)协议更具有研究价值、更具有应用价值：从安全方面讲，保密集合交集的精确计算可以为分布式用户提供更安全的隐私保护(既保护参与方的数据元素又保护参与方的数据元素个数)；从效率方面讲，保密集合交集的精确计算因为协同计算无差错，一次协同计算结束后参与各方必定会实现协同计算的目的.

本文着手于解决安全两方集合交集的精确计算问题：参与双方 $P_1$ 和 $P_2$ 分别拥有集合 $S_1$ 与 $S_2$，在执行保密计算 $S_1 \cap S_2$ 的协议后他们要实现：(1) 协同计算的结果百分百等于 $S_1 \cap S_2$，不再像先前诸如协议[1,5,8-10]等一些保密两方交集协议那样，协同计算的结果可能存在差错，且差错的范围难于界定；(2) 在协同完成计算 $S_1 \cap S_2$ 后，双方在不泄露各自集合元素的同时，也不泄露各自集合的势. 也就说，$P_1$ 和 $P_2$ 协同完成保密计算 $S_1 \cap S_2$ 后，二者的计算结果一定正确，他们除了得到 $S_1 \cap S_2$ 外，再也得不到有关对方的任何其它信息(集合的元素、集合的势).

近些年来,因保密集合交集计算问题在保密服务预约、保密信息匹配[1]、保密数据挖掘[2]、保密推荐系统[3]、安全事件信息的安全共享[4]等领域有着广泛应用，得到了学者们的广泛关注，产出了相当丰硕的成果. 这些成果按照协议构造时所采用的保密工具大致可以分成五类：基于某个数学难解问题构造的协议[5,6,12-22], 基于对称密码方案构造的协议[23,24], 基于混淆电路构造的协议[23-26]，基于不经意传输构造的协议[27-30], 基于不经意函数计算构造的协议[1,4,8,11,31]. 其中，基于某个数学难解问题构造的一类协议大都采用保密验证元素的方法实现交集的计算. 保密验证元素的方法大致可以分为四类：多项式验根，函数间提取公因式，不经意函数计算法，用数字签名认证元素法.

1. 采用“多项式验根”法构造协议[1,8]. 两方保密集合交集运算最初由 Freedman 等人提出[1]. 此协议采用如下方式实现保密集合运算(不失一般性，我们假定 Alice 和 Bob 为参与保密计算的两个互不信任的协作方)：

(1) Alice 运行 Paillier 同态加密方案密钥生成算法 $\mathcal{G}(1^k)$，获取公私密钥对 $(\mathrm{K}_{pub}, \mathrm{K}_{pri})$；

(2) Alice 将集合中的元素作为多项式的根构造一个多项式：$f(x)=(x-a_1)(x-a_2)\cdots(x-a_{k_a})=\hat{a}_{k_a}x^k+\hat{a}_{k_a-1}x^{k-1}+\cdots+\hat{a}_0$，然后将多项式的系数 $\hat{a}_{k_a},\hat{a}_{k_a-1},\cdots,\hat{a}_0$ 用 Paillier 同态加密方案加密：$E(\hat{a}_{k_a}),E(\hat{a}_{k_a-1}),\cdots,E(\hat{a}_0)$，并按序传送给 Bob；

(3) Bob 收到 $E(\hat{a}_{k_a}),E(\hat{a}_{k_a-1}),\cdots,E(\hat{a}_0)$ 后，在密文上执行同态运算，进而求得 $r\cdot f(b_i)+b_i$ (其中 $r$ 是 Bob 随机选取的，$b_i$ 是 Bob 集合中的元素)对应的密文 $c_i=E(r\cdot f(b_i)+b_i)$，并将密文 $C=(c_i)_{i=1,\cdots,k_b}$ 发送给 Alice；

(4) Alice 收到 $C=(c_i)_{i=1,\cdots,k_b}$ 后进行解密；如果解密结果落在自己的集合中，则认为解密结果为保密集合交集的元素之一. 此后该协议被拓展成高效的协议[17,32]、对外包数据可验证的协议[18]和抗恶意敌手的协议[8-10,12,13].

2. 采用“多个函数间提取公因式”方法构造协议[5,19,33]. Kissner, Song 采用多项式间提取公因式法设计了一个适应于多方环境的保密集合交集计算协议[5]. 该协议的朴素思想如下：

(1) Alice(假定 Alice 拥有 Paillier 解密密钥)和 Bob 分别将各自的集合 $A$ 与 $B$ 表示成多项式 $Q_A(\cdot)$ 和 $Q_B(\cdot)$；

(2) 如果令 $r_A(\cdot)$ 和 $r_B(\cdot)$ 分别是 Alice 和 Bob 选择的随机多项式，则多项式 $r_A(\cdot)\cdot Q_A(\cdot)+r_B(\cdot)\cdot Q_B(\cdot)$ 的根所构成的集合等于 $A\cap B$ 的概率很大. 这是因为如果 $x_i,x_{i+1},\cdots,x_{i+j}$ ( $j\geqslant 1$ ) 是 $Q_A(\cdot)$ 和 $Q_B(\cdot)$ 的公共零点，$Q_A(\cdot)$ 和 $Q_B(\cdot)$ 提取因式 $(x-x_i)(x-x_{i+1})\cdots(x-x_{i+j})$ ($i\leqslant j$)后的剩余多项式分别记作 $Q_RA(\cdot)$ 和 $Q_RB(\cdot)$，则 $r_A(\cdot)\cdot Q_A(\cdot)+r_B(\cdot)\cdot Q_B(\cdot)$ 可以表示成 $(x-x_i)(x-x_{i+1})\cdots(x-x_{i+j})(r_A(\cdot)\cdot Q_RA(\cdot)+r_B(\cdot)\cdot Q_RB(\cdot))$。

2011 年，Kim M 等人将该协议拓展成服务器与客户都可以得到保密交集的协议[19]；2013 年，Dan Boneh 等人将该协议拓展成三方协议[34]；2018 年 Zhou 等人采用多项式间提取公因式法设计了一个适应于多方环境、具有信息论安全的保密集合交集计算协议[33]. 此协议的朴素思想如下：① $n$ 个参与者 $\{P_i\}_{1\leqslant i\leqslant n}$ 分别将各自集合表示成保密多项式 $f_i(x)=\sum_{q=0}^{q=2l}a_{iq}x^q=x^{2(l-l_i)}(x-x_{i1})^2(x-x_{i2})^2\cdots(x-x_{il_i})^2$，其中 $l_i\leqslant l$，$i\in\{1,2,\cdots,n\}$；② 参与者 $P_i, 1\leqslant i\leqslant n-1$ 按照如下方式执行：1) 将各自的保密多项式随机地分成非零的 $k$ 份 $f_{i1}(x), f_{i2}(x),\cdots, f_{ik}(x)$ 满足 $f_i(x)=\sum_{j=1}^{k}f_{ij}(x)$，为每一份加入随机数后发送给 $n$ 个参与者中的 $k$ 个；2) 将各自收到的多项式相加，组成新的多项式 $g_i(x), i=1,\cdots,n-1$，并分别发送给 $P_{\mathrm{n}}$；3) $P_{\mathrm{n}}$ 计算交集.

3. 采用“不经意伪随机函数计算”思想构造协议[7,14,15,35]. Michael 借助不经意伪随机函数计算实现了保密集合交集计算[7]. 其思想如下(此处仍假定 Alice 和 Bob 为协议的两个参与方,他们分别持有集合 $X$ 与 $Y$ ):

(1) Alice 选取一个 PRF 密钥 $\mathcal{K}_{prf}$ 并计算集合 $PRF_X=\{PRF_{\mathcal{K}_{prf}}(x)\}_{x\in X}$ ;

(2) Alice 和 Bob 联合执行不经意随机函数计算协议,其中 Alice 在协议中的输入为 $\mathcal{K}_{prf}$ , Bob 的输入为其私有集合 $Y$ ,协议执行完后,Bob 得到集合 $PRF_Y=\{PRF_{\mathcal{K}_{prf}}(x)\}_{y\in Y}$ ;

(3) Alice 将集合 $PRF_X=\{PRF_{\mathcal{K}_{prf}}(x)\}_{x\in X}$ 发给 Bob,Bob 计算 $PRF_X\cap PRF_Y$ 并据此提取 $X\cap Y$ .

此后,该协议被拓展成一些高效的协议[14,15,35]. 近年来,有些学者在该方法的基础上结合其他方法实现了满足某些实际应用需求的协议[16,25,27,28,36].

4. 采用数字签名技术构造带认证的协议[6,20,37-39]. 2009 年,De Cristofaro[6] 采用数字签名技术设计了一个带认证的保密集合交集计算协议. 其思想如下(此处仍假定 Alice 和 Bob 为协议的两个参与方,他们分别持有集合 $X=\{x_i\}_{1\leqslant i\leqslant l_a}$ 与 $Y=\{y_i\}_{1\leqslant i\leqslant l_b}$ , $1\leqslant l_a,l_b\leqslant n$ ,分别表示 Alice 和 Bob 各自集合中元素的数目):

(1) Alice 运行加密方案密钥生成算法,获取公私密钥对 $(K_{pub},K_{pri},g)$ ;

(2) Alice 为集合中的所有元素计算一个指纹 $hc_i=H(x_i)$ ,并对指纹进行签名: $\sigma_i=(hc_i)^{K_{pub}} \bmod n$ ,然后随机选取一个 $R_{ci}\leftarrow Z_{n/4}$ 并计算: $\mu_i=(\sigma_i)^2\cdot g^{R_{ci}} \bmod n$ ,最后将 $\{\mu_i\}_{1\leqslant i\leqslant n}$ 传送给 Bob;

(3) Bob 收到 $\{\mu_i\}_{1\leqslant i\leqslant n}$ 后按照如下方式执行:① 随机选取一个 $R_s\leftarrow Z_{n/4}$ 并计算 $Z=g^{K_{pri}\cdot R_s} \bmod n$ ;② 对于 $\forall i\in Z^+_{l_a},\forall j\in Z^+_{l_b}$ 计算 $K_{s,i,j}=(\mu_i)^{K_{pri}\cdot R_s}\cdot(H(y_i))^{-2R_s} \bmod n$ , $t_{i,j}=H(K_{s,i,j})$ ;③将 $Z,\{t_{1,1},\cdots,t_{i,j}\}$ 发送给 Alice;

(4) Alice 收到 $Z,\{t_{1,1},\cdots,t_{i,j}\}$ 后按照如下方式执行: ① 对于 $\forall j\in Z^+_{l_a}$ 计算 $K_{c,j}=(Z)^{R_{cj}} \bmod n$ , $t_j=H(K_{c,j})$ ; ② 计算 $\{t_1,\cdots,t_{l_a}\}\cap\{t_{1,1},\cdots,t_{l_a,l_b}\}$ .

2010 年,De Cristofaro 对上述协议的安全性进行了改良,提出了一个能够抗恶意敌手攻击的保密集合交集协议[20];随后,De Cristofaro 对上述协议的隐私保护条件做了进一步约束(协议应在在保护双方数据集元素的同时还不泄露客户数据集的大小,即客户数据集的势),提出了两个快速保密集合交集计算协议[37,38]. 2017,Kiss Á 对上述协议进行了改进,提出了一个面向移动应用的轻量级保密集合交集协议[39].

综上所述,协议虽然都很漂亮地解决了保密集合交集问题. 但是我们发现保密集合交集计算领域依然存在如下一些尚未彻底解决的问题:

(1) 集合交集元素的无差错匹配问题依然没有得到解决,具体体现在如下两个方面:

①采用“多项式验根”法所构造的保密集合交集计算协议,其匹配计算结果可能存在差错,且差错的范围难以界定. 这是因为这些协议只关注了 $f(b_i)=0$ 时 $r\cdot f(b_i)+b_i=b_i$ 这一合理性,而忽略了利用同态加密运算获得密文 $g^{r\cdot f(b_i)+b_i} \bmod n^2=g^{r\cdot(b_i)+b_i \bmod n} \bmod n^2$ c( $g=1+kn,k\in Z^+$ )过程中,指数上所隐含的模 $n$ 运算: $r\cdot f(b_i)+b_i \bmod n$ . $f(b_i)\neq 0$ 时, $r\cdot f(b_i)+b_i \bmod n$ 很可能是 Alice 集合中不等于 $b_i$ 的元素 $a_j$ (只属于用户数据库而不是服务器数据库中的元素),即 $r\cdot f(b_i)+b_i \bmod n=a_j$ (如图 1 所示). 如果 $r\cdot f(b_i)+b_i \bmod n=a_j$ 这种情况发生了,即用户 Alice 所求结果中包含了两方数据交集中本应该没有的元素(此元素只属于 Alice 而不属于 Bob),即 Alice(用户)所求结果是错误的. 遗憾的是 Alice 无法发现这种匹配错误. 事实上,只有匹配正确的 PSI 计算才有意义,这违背了保密集合交集计算的初衷(计算结果正确且能保证参与双方的隐私数据). 特别是用户数据库集合包含的元素越多时,则用户数据库集合中本来不属于交集的元素 $a_j$ ,即 $r\cdot f(b_i)+b_i \bmod n=a_j$ 出现在交集中的概率就越大,即保密信息匹配结果出现错误的概率就越大.

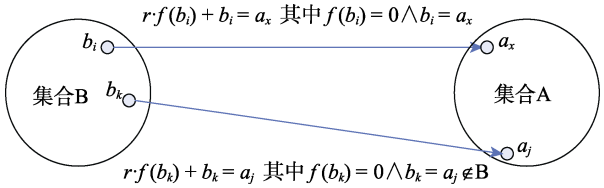


图 1 不可忽略的误差

②在采用多个函数间提取公因式方法所构造的保密集合交集计算协议中, $r_A(\cdot)\cdot Q_R A(\cdot)$ 与 $r_B(\cdot)\cdot Q_R B(\cdot)$ 很可能还存在公共因式 $(x-x_{i'})(x-x_{i'+1})\cdots(x-x_{i+j'})$ ( $i'\leqslant j'$ ). 显然,这很可能出现 $\{x_i,x_{i+1},\cdots,x_{i+j'}\}\cup\{x_{i'},x_{i'+1},x_{i+j'}\}\neq\{x_i,x_{i+1},\cdots,x_{i+j}\}$ 的情形,并且这种情形发生的概率是无法界定的. 而 $\{x_i,x_{i+1},\cdots,x_{i+j}\}\cup\{x_{i'},x_{i'+1},x_{i+j'}\}\neq\{x_i,x_{i+1},\cdots,x_{i+j}\}$ 意味着产生匹配计算差错,即 $A\cap B\neq\{x_i,x_{i+1},\cdots,x_{i+j}\}$ . 从而导致实际计算出的交集 $A\cap B$ 中存在元素要么只属于 Alice,要么只属于 Bob. 在该方法的拓展协议[33]中, $P_n$ 在计算 $\sum_{i=1}^{n}g_i(x)$ 时,除了可以提取完

全平方构成的公因式外还很可能引入公共因式 $(x-x_{i'})(x-x_{i'+1})\cdots(x-x_{i+j'})$ ($i'\leqslant j'$)，例如，$(x-6)^2(x-21)^2$ 与 $(x-6)^2(x-21)^2(x-2)^2$ 相加的结果中不仅有公因式 $(x-6)^2(x-21)^2$ 还有公因式 $(x-5)(x+4)$. 这显然会导致保密计算结果产生错误.

(2) 基于不经意函数计算的保密集合交集计算协议虽然不会泄露双方集合中的私密数据，但会泄露拥有 PRF 密钥 $\mathcal{K}_{prf}$ 一方集合的势.

为了丰富保密集合交集计算的研究方法，同时实现无匹配差错且保密范畴更广的(既不会泄露参与双方集合中的私密数据，也不会泄露参与双方私密数据的数量)保密集合交集计算. 在两方参与的环境下，构建了无匹配差错的保密集合交集计算协议. 该协议在保证保密交集计算 100%正确的前提下，能够用于保密计算元素为有理数集合的交集，且不泄露各参与方私有集合的势. 除此之外，本文还做了如下三个工作：

(1) 提出一个称之为二元组型高阶剩余类判定性(Decisional Tuple Composite Residuosity)问题的难解问题，并证明了其难解性；

(2) 基于二元组型高阶剩余类判定性问题，设计了一个高效的有理数加密方案，该方案既可以用于解决集合元素为整数域上的保密集合交集问题也可以用于解决集合元素为有理数域上的保密集合交集问题，拓展了解决问题的范围；

(3) 提出了一种适用于无匹配差错的保密集合交集计算的集合编码方法：集合的定长向量编码.

## 2　预备知识

### 2.1　非对称加密系统安全性定义[40,41]

非对称加密系统具有不可区分安全性的充分必要条件. 加密语义安全的概念通常由一个系统构建者和一个敌手参与的思维实验来刻画[40]，该思维实验通常被称作不可区分安全游戏. 对于任意一个安全参数为 $k$ 的公钥加密系统 $\mathcal{E}$，任何攻击系统 $\mathcal{E}$ 的敌手 $\mathcal{A}$ (在多项时间可被执行完的算法)，它参与不可区分游戏时，能够获胜的优势函数记作 $Adv_{\mathcal{A},\mathcal{E}}(k)$，其用选择性明文攻击方法攻击 $\mathcal{E}$ 的事件记作 $PubK^{cpa}_{\mathcal{A},\mathcal{E}}(k)$. $\mathcal{E}$ 具有选择明文不可区分(indistinguishability under chosen-plaintext attack, IND-CPA)安全性，当且仅当存在一个可忽略的函数 $\delta$，满足：

$$Adv^{cpa}_{\mathcal{A},\mathcal{E}}(k)=\left|Pr[PubK^{cpa}_{\mathcal{A},\mathcal{E}}(k)=1]-\frac{1}{2}\right|\leqslant\delta(k).$$

其中，IND-CPA 安全性游戏在文献[40,41]被定义为：

(1) 系统构建者生成系统 $\mathcal{E}$，并产生公私钥对 ($K_{Pub}$, $K_{Pri}$)，敌手 $\mathcal{A}$ 获得公钥 $K_{Pub}$；

(2) $\mathcal{A}$ 生成若干明文消息，并得到它们对应的密文；

(3) $\mathcal{A}$ 输出两个等长的消息 $m_b$, $b\in\{0,1\}$，系统构建者随机选取 $b\in\{0,1\}$，将 $m_b$ 加密成挑战密文 $C^*$，并将 $C^*$ 发送给 $\mathcal{A}$；

(4) $\mathcal{A}$ 输出 $b'\in\{0,1\}$，如果 $b'=b$，则敌手攻击 $\mathcal{E}$ 成功.

在该游戏中，敌手的优势被定义成关于安全参数 $k$ 的函数：

$$Adv^{cpa}_{\mathcal{A},\mathcal{E}}(k)=\left|Pr[b'=b]-\frac{1}{2}\right|.$$

### 2.2　抗半诚实敌手的安全多方计算模型

我们采用文献[42,43]给出的半诚实模型下两方计算协议的安全性定义：设分别拥有 $x_1$ 与 $x_2$ 的两个参与者采用协议 $\Pi$ 协同计算确定性概率多项式函数 $f=(f_i,f_{1-i}):\Psi^*\times\Psi^*\to\Psi^*\times\Psi^*, i\in\Psi$，$\Psi=\{0,1\}$. 如果存在多项式时间的敌手，记作 $\mathcal{S}_1$ 和 $\mathcal{S}_2$，分别控制着参与者 $P_1$ 和 $P_2$ 并满足

$$\begin{cases}\{\mathcal{S}_1(x_1,f_1(x_1,x_2))\}_{x_1,x_2\in\{0,1\}^*}\overset{c}{\equiv}\{\text{view}_1^{\Pi}(x_1,x_2)_{x_1,x_2\in\{0,1\}^*}\} & (1a)\\ \{\mathcal{S}_2(x_2,f_1(x_1,x_2))\}_{x_1,x_2\in\{0,1\}^*}\overset{c}{\equiv}\{\text{view}_2^{\Pi}(x_1,x_2)_{x_1,x_2\in\{0,1\}^*}\} & (1b)\end{cases}.$$

则称协议 $\Pi$ 在半诚实模型下能够安全计算函数 $f=(f_i,f_{1-i}), i\in\{0,1\}$. 其中 $\{\mathcal{S}_1(x_1,f_1(x_1,x_2))\}_{x_1,x_2\in\{0,1\}^*}$ 与 $\{\mathcal{S}_2(x_2,f_1(x_1,x_2))\}_{x_1,x_2\in\{0,1\}^*}$ 分别代表参与方 $P_1$ 和 $P_2$ 的视图，“$\overset{c}{\equiv}$”表示计算上不可区分. 敌手视图包括相应参与者的输入、模拟协议时的模拟抛硬币结果和模拟协议时收到的消息.

Goldreich 采用比特承诺与零知识证明理论构造了一个编译器，如有必要，就可以借助该编译器将抗半诚实敌手攻击的协议自动转换成抗恶意敌手攻击的协议. 其核心思想是迫使恶意的参与者以半诚实方式参与协议的执行，否则就会被发现. 如果我们需要一个抗恶意敌手攻击的协议，只要将事先设计好的抗半诚实敌手攻击的协议 $\Pi$ 作为编译器的输入，编译器就可以为我们输出一个抗恶意敌手攻击的协议 $\Pi'$. 出于工程实际的需要，文中假定保密集合交集计算协议的参与者都是半诚实的.

### 2.3　Paillier同态加密方案

Paillier 加密方案[44] (如图 2 所示)具有良好的加法同态性：

$$\begin{cases}E(M_0+M_1)=E(M_0)\cdot E(M_1) & (2a)\\ E(M_0\cdot M_1)=(E(M_0))^{M_1}=(E(M_1))^{M_0} & (2b)\end{cases}.$$

加密: 明文 $M<n$， $n=pq$， $p$、$q$ 是等长的大素数

选择随机数 $r<n$

密文 $C=g^{M}r^{n} \bmod n^{2}$， $g=1+kn$ $(k\in Z_n^*)$

解密: 密文 $C<n^2$

明文 $M=\dfrac{L(C^{\lambda} \bmod n^2)}{L(g^{\lambda} \bmod n^2)} \bmod n$； $L(\mu)=\dfrac{\mu-1}{n}$；

$\lambda=\mathrm{lcm}(p-1,q-1)$

图 2 Paillier 加密方案

该方案在高阶剩余类判定性困难假设下具有语义安全性，即任意两个明文 $M_0$、$M_1$ ($|M_0|=|M_1|$) 用此方案加密后所得的对应消息 $C_0$、$C_1$ 是两个计算不可区分的量：$C_0 \overset{c}{\equiv} C_1$.

### 2.4 高阶剩余类判定性问题

高阶剩余类判定性(Decisional Composite Residuosity, DCR )问题[41,44]. 简单地讲，高阶剩余类判定性(Decisional Composite Residuosity, DCR )问题[4,8]就是判定是否存在一个 $y$ 使得 $z\equiv y^n \bmod n^2$，其中 $n$（合数）与 $z\in Z_{n^2}$ 为事先给定的量. 其形式化语言描述[8]如下：

假定区分算法 $D$ 在区分实验中，能够区分两个分布 $D_{\mathcal{E}}=\{(n,\mathrm{R})\,|\,\mathrm{R}\leftarrow\{r^n \bmod n^2\,|\,r\in Z_n\}\}$ 和 $D_{ran}=\{(n,\mathrm{R})\,|\,\mathrm{R}\overset{R}{\leftarrow}Z_{n^2}\}$ 的优势记作关于系统安全参数 $\tau$ 的函数 $Adv_D^{(D_{ran},D_{\mathcal{E}})}(\tau)$. 给定一个随机分布 $(n,\mathrm{R})\in\{D_{ran},D_{\mathcal{E}}\}$，区分算法 $D$ 能够区分出 $(n,\mathrm{R})$ 是 $D_{ran}$ 和 $D_{\mathcal{E}}$ 中哪一个的优势函数 $Adv_D(\tau)$ 可以表示为

$$Adv_D^{(D_{ran},D_{\mathcal{E}})}(\tau)=\left|Pr\left[\mathrm{D}(n,R)=D_{ran}\right]-Pr\left[\mathrm{D}(n,R)=D_{\mathcal{E}}\right]\right|.$$

事实上，高阶剩余类判定性问题是公认的难解问题，所以必定存在一个可忽略的函数 $\delta(\tau)$ 满足

$$Adv_D^{(D_{ran},D_{\mathcal{E}})}(\tau)\leqslant\delta(\tau).$$

二元组型高阶剩余类判定性(Decisional Tuple Composite Residuosity , DTCR)问题. 现将上面两个分布 $D_{ran}$ 和 $D_{\mathcal{E}}$ 中的 $R$ 分别替换成 $(\hat{R},\check{R})$ 和 $(\hat{r}^n \bmod n^2,\check{r}^n \bmod n^2)$，得到两个新分布 $\overline{D}_{ran}$ 和 $\overline{D}_{\mathcal{E}}$：

$$\begin{cases}\overline{D}_{ran}=\{(n,R)=(n,(\hat{R},\check{R}))\,|\,R\xleftarrow{(\hat{R},\check{R})}Z_{n^2}\times Z_{n^2}\} & (3a)\\ \overline{D}_{\mathcal{E}}=\{(n,R)=(n,(\hat{r}^n \bmod n^2,\check{r}^n \bmod n^2))\,|\,R\leftarrow\{(\hat{r}^n \bmod n^2,\check{r}^n \bmod n^2)\,|\,\hat{r},\check{r}\in Z_n\}\} & (3b)\end{cases}.$$

区分 $(n,R)\in\{\overline{D}_{ran},\overline{D}_{\mathcal{E}}\}$ 是 $\overline{D}_{ran}$ 和 $\overline{D}_{\mathcal{E}}$ 中哪一个的问题被称作 DTCR 问题.

**定理 1.** 如果 DCR 是多项式时间的难解问题，则 DTCR 问题也是多项式时间的难解问题.

证明. 假定区分算法 $D'$ 区分出 $(n,R)\in\{\overline{D}_{ran},\overline{D}_{\mathcal{E}}\}$ 是分布 $\overline{D}_{ran}$ 和 $\overline{D}_{\mathcal{E}}$ 中哪一个的优势函数记作 $Adv_{D'}^{(\overline{D}_{ran},\overline{D}_{\mathcal{E}})}(\tau)$.

显然如果算法 $D'$ 能够以很大的优势区分出 $(n,R)\in\{\overline{D}_{ran},\overline{D}_{\mathcal{E}}\}$ 是分布 $\overline{D}_{ran}$ 和 $\overline{D}_{\mathcal{E}}$ 中的哪一个，那么就可以用区分算法 $D'$ 解决 DCR 问题.

假定算法 $D'$ 能以不可忽略的优势解决 DCR 问题，则用区分算法 $D'$ 解决 DCR 问题的成功概率可用贝叶斯全概率公式表示为：

$$\frac{1}{2}\times\left(\frac{1}{2}+Adv_{\boldsymbol{D}}^{(D_{ran},D_{\mathcal{E}})}(\tau)\right)+\frac{1}{2}\times\frac{1}{2}=\frac{1}{2}+\frac{Adv_{\boldsymbol{D}}^{(D_{ran},D_{\mathcal{E}})}(\tau)}{2}.$$

因此，区分算法 $D'$ 能够区分出 $(n,R)\in\{\overline{D}_{ran},\overline{D}_{\mathcal{E}}\}$ 是 $\overline{D}_{ran}$ 和 $\overline{D}_{\mathcal{E}}$ 中哪一个的优势函数 $Adv_{D'}^{(\overline{D}_{ran},\overline{D}_{\mathcal{E}})}(\tau)$ 满足

$$Adv_{D'}^{(\overline{D}_{ran},\overline{D}_{\mathcal{E}})}(\tau)=\left(\frac{1}{2}+\frac{Adv_D^{(D_{ran},D_{\mathcal{E}})}(\tau)}{2}\right)-\frac{1}{2}\leqslant\frac{\delta(\tau)}{2}.$$

因为 $\delta(\tau)$ 是可忽略的函数，所以 $\dfrac{\delta(\tau)}{2}$ 也是可忽略的. 这与假设“算法 $D'$ 能以不可忽略的优势解决 DCR 问题”矛盾，也就说，二元组型高阶剩余类判定性问题是难解的.

证毕.

## 3 有理数的加密

Paillier 加密方案具有同态加的特性，为解决一些保密多方计算问题提供了便利. 但是它只能用于解决 $Z_n$ 上的函数保密计算问题. 为了拓展解决问题的范围，本文设计了一个基于二元组型高阶剩余类判定性问题的且能用于加密有理数的加密方案. 该方案把一个有理数加密成一个二维向量，其中表示分子和分母的密文分量可以通过密文分量间(同为分子的密文分量或同为分母的密文分量)的模乘运算实现同类型密文分量对应的明文分量在指数上的加同态；并且在某些需要保护无密钥一方隐私的计算分布式计算中，作为无密钥的参与方，计算包含隐私的脱敏数据时所用的参数 $g_x$，可以由其根据实际需要适应性地选取. 用于有理数的加密方案及其性能描述如下：

### 3.1 有理数的加密方案

有理数的加密系统由密钥生成算法(Key-Generation)、换底算法(Base-Transformation)、有理

加密(Encryption)和有理解密(Decryption)四个随机算法组成，记作 $\mathcal{E}$ (Key-Generation, Base-Transformation, Encryption, Decryption)：

Key-Generation：生成两个大素数 $p$、$q$ 使得 $|p|=|q|$ ($|\cdot|$ 表示数“·”的二进制码长)，计算 $n=pq$，$\lambda=\mathrm{lcm}(p-1,q-1)$，$g=1+kn$ $(k\in Z_n^*)$，发布公钥 $K_{pub}=(n,g)$；保留私钥 $K_{pri}=\lambda$.

Base-Transformation: 随机选择一个 $k'\in Z_n$，为某一时间段内的加密运算计算新的底数：

$$g_x=1+k'(g-1)\bmod n^2.$$

Encryption：发送者首先将要加密的有理数对应的分数表示成序偶 $(\hat{M},\check{M})$ ($\hat{M},\check{M}\in Z_n$ 分别为分子和分母并且满足 $\gcd(\hat{M},\check{M})=1$)，然后随机选择 $r_0\in Z_n^+, r_1, r_2\in Z_n^*$，对于 $M<n$，计算：

$$\hat{c}=(1+\hat{M}\cdot(g_x-1))\hat{r}^n \bmod n^2=g_x^{\hat{M}}\hat{r}^n \bmod n^2,$$

$$\check{c}=(1+\check{M}\cdot(g_x-1))\check{r}^n \bmod n^2=g_x^{\check{M}}\check{r}^n \bmod n^2.$$

Decryption：接收方执行解密运算：

$$\frac{\hat{M}}{\check{M}}=\frac{\mathcal{L}(\hat{c}^{\lambda} \bmod n^2)}{\mathcal{L}(\check{c}^{\lambda} \bmod n^2)}=\frac{(1+\lambda\cdot\hat{M}\cdot(g_x-1))-1}{(1+\lambda\cdot\check{M}\cdot(g_x-1))-1},$$

其中 $\mathcal{L}(\mu)=\mu-1,(\mu=1+xn)\wedge(\mu<n^2)$.

### 3.2　解密正确性验证

证明.

$$\frac{\mathcal{L}(\hat{c}^{\lambda} \bmod n^2)}{\mathcal{L}(\check{c}^{\lambda} \bmod n^2)}=\frac{\mathcal{L}(g_x^{\lambda\hat{M}} \bmod n^2)}{\mathcal{L}(g_x^{\lambda\check{M}} \bmod n^2)}=\frac{\mathcal{L}((1+kn)^{k'\lambda\hat{M}} \bmod n^2)}{\mathcal{L}((1+kn)^{k'\lambda\check{M}} \bmod n^2)}=\frac{((1+k'\lambda\hat{M}\,kn)\bmod n^2)-1}{((1+k'\lambda\check{M}\,kn)\bmod n^2)-1}=\frac{k'k\lambda\hat{M}\,n\bmod n^2}{k'k\lambda\check{M}\,n\bmod n^2}=\frac{\hat{M}}{\check{M}}.$$

证毕.

### 3.3　有理数加密方案的可计算性

假定本文方案和 Paillier 方案选用的模数相同，皆为 $n^2$，并将模 $n^2$ 意义下的一次自模乘运算的复杂度($O(\log^2 n)$)定义为衡量本文方案和 Paillier 方案中算法复杂度的基本单位. 将 $g_k=(1+n)^k \bmod n^2$，$k\leqslant n-1$ 按照二项式展开计算可得 $g_k=(1+n)^k \bmod n^2=1+kn$，因此，$g_k$ 可以由简单计算“$1+kn$”得到. 因为 $k\leqslant n-1$，所以 $1+kn$ 的计算复杂度为 $O(\log^2 n)$；从而可得：$\mathcal{E}$ 中的加、解密算法的复杂度分别为 $O(4\log^3 n+\log^2 n)$ 和 $O(2\log^3 n)$. 而 Paillier 方案中的加、解密算法的计算复杂度皆为 $O(2\log^3 n)$. 显然，就本文方案与 Paillier 方案在加、解密算法的计算复杂性方面而言，二者属于同一级别.

### 3.4　有理数加密方案的性质

**性质 1.** 盲化性. 假定有理数 $M_A=\frac{\hat{M}_A}{\check{M}_A}$ 在方案 $\mathcal{E}$ 作用下对应的密文序偶为 $(\hat{c}_A=g_x^{\hat{M}_A}(\hat{r}_A)^n \bmod n^2,\ \check{c}_A=g_x^{\check{M}_A}(\check{r}_A)^n \bmod n^2)$，则任意参与者可对其实施盲化运算：

$$(\hat{c}_A)^\alpha\equiv g_x^{\alpha\cdot\hat{M}_A}(\hat{r}_A)^{\alpha\cdot n} \bmod n^2=g_\alpha^{\hat{M}_A}(\hat{r}_A)^{\alpha\cdot n} \bmod n^2,$$

$$(\check{c}_A)^\alpha\equiv g_x^{\alpha\cdot\check{M}_A}(\check{r}_A)^{\alpha\cdot n} \bmod n^2=g_\alpha^{\check{M}_A}(\check{r}_A)^{\alpha\cdot n} \bmod n^2.$$

或

$$(\hat{c}_A)^\alpha(\hat{R})^n\equiv g_x^{\alpha\cdot\hat{M}_A}(\hat{R}(\hat{r}_A)^\alpha)^n \bmod n^2=g_\alpha^{\hat{M}_A}(\hat{R}_A)^n \bmod n^2,$$

$$(\check{c}_A)^\alpha(\check{R})^n\equiv g_x^{\alpha\cdot\check{M}_A}(\check{R}(\check{r}_A)^\alpha)^n \bmod n^2=g_\alpha^{\check{M}_A}(\check{R}_A)^n \bmod n^2,$$

其中 $\hat{R},\check{R},\hat{R}_A,\check{R}_A\in Z_n^*$.

对于解密者而言，解密 $(\hat{c}_A,\check{c}_A)$、$((\hat{c}_A)^\alpha \bmod n^2$，$(\check{c}_A)^\alpha \bmod n^2)$以及$((\hat{c}_A)^\alpha(\hat{R})^n \bmod n^2, (\check{c}_A)^\alpha(\check{R})^n \cdot \bmod n^2)$，结果都是 $M_A=\frac{\hat{M}_A}{\check{M}_A}$. 这是因为

$$g_x^\alpha \bmod n^2=(1+k'(g-1)\bmod n^2)^\alpha \bmod n^2$$

$$=(1+\alpha k'(g-1)) \bmod n^2$$
$$=1+\alpha(g_x-1) \bmod n^2$$
$$=g_\alpha,$$

$$((\hat{r}_A)^{\alpha\cdot n} \bmod n^2)^\lambda \bmod n^2=((\hat{r}_A)^{\lambda\cdot n})^\alpha \bmod n^2=1^\alpha \bmod n^2,$$

$$((\check{r}_A)^{\alpha\cdot n} \bmod n^2)^\lambda \bmod n^2=((\check{r}_A)^{\lambda\cdot n})^\alpha \bmod n^2=1^\alpha \bmod n^2,$$

$$(\hat{c}_A)^\alpha \equiv (g_x^{\hat{M}_A}(\hat{r}_A)^n \bmod n^2)^\alpha \bmod n^2$$
$$=g_x^{\alpha\cdot\hat{M}_A}(\hat{r}_A)^{\alpha\cdot n} \bmod n^2$$
$$=((g_x)^\alpha)^{M_A}(\hat{r}_A)^{\alpha\cdot n} \bmod n^2$$
$$=(g_x)^\alpha)^{\hat{M}_A}(\hat{r}_A)^{\alpha\cdot n} \bmod n^2$$
$$=g_\alpha^{\hat{M}_A}(\hat{r}_A)^{\alpha\cdot n} \bmod n^2,$$

$$(\check{c}_A)^\alpha \equiv (g_x^{\check{M}_A}(\check{r}_A)^n \bmod n^2)^\alpha \bmod n^2$$
$$=g_x^{\alpha\cdot\check{M}_A}(\check{r}_A)^{\alpha\cdot n} \bmod n^2$$
$$=((g_x)^\alpha)^{\check{M}_A}(\check{r}_A)^{\alpha\cdot n} \bmod n^2$$
$$=(g_x)^\alpha)^{\check{M}_A}(\check{r}_A)^{\alpha\cdot n} \bmod n^2$$
$$=g_\alpha^{\check{M}_A}(\check{r}_A)^{\alpha\cdot n} \bmod n^2,$$

因 $\hat{R},\hat{r}_A\in Z_n^*$，则 $\hat{R}(\hat{r}_A)^\alpha$ 可以表示成 $\hat{R}(\hat{r}_A)^\alpha=\hat{R}_A+\omega n$ (因为 $\hat{R},\hat{r}_A\in Z_n^*$，所以 $\hat{R}(\hat{r}_A)^\alpha$ 不能被 $n$ 整除，从而可得 $\hat{R}_A\neq 0$)，其中 $1\leqslant\omega\leqslant n-1$. 从而有

$$(\hat{R}(\hat{r}_A)^\alpha)^n \bmod n^2$$
$$=(\hat{R}_A+\omega n)^n \bmod n^2$$
$$=\sum_{k=0}^{n}\binom{n}{k}(\hat{R}_A)^{n-k}(\omega n)^k \bmod n^2$$
$$=(\hat{R}_A)^n \bmod n^2,$$

同理可得

$$(\check{R}(\check{r}_A)^\alpha)^n \bmod n^2$$
$$=(\check{R}_A+\omega n)^n \bmod n^2$$
$$=\sum_{k=0}^{n}\binom{n}{k}(\check{R}_A)^{n-k}(\omega n)^k \bmod n^2$$
$$=(\hat{R}_A)^n \bmod n^2,$$

$$(\hat{c}_A)^\alpha(\hat{R})^n \equiv (g_x^{\hat{M}_A}(\hat{r}_A)^n \bmod n^2)^\alpha(\hat{R})^n \bmod n^2$$
$$=g_x^{\alpha\cdot\hat{M}_A}(\hat{R}(\hat{r}_A)^\alpha)^n \bmod n^2$$
$$=((g_x)^\alpha)^{M_A}(\hat{R}_A)^n \bmod n^2$$
$$=g_\alpha^{\hat{M}_A}(\hat{R}_A)^n \bmod n^2,$$

$$(\check{c}_A)^\alpha(\check{R})^n \equiv (g_x^{\check{M}_A}(\check{r}_A)^n \bmod n^2)^\alpha(\check{R})^n \bmod n^2$$
$$=g_x^{\alpha\cdot\check{M}_A}(\check{R}(\check{r}_A)^\alpha)^n \bmod n^2$$
$$=((g_x)^\alpha)^{M_A}(\check{R}_A)^n \bmod n^2$$
$$=g_\alpha^{\check{M}_A}(\check{R}_A)^n \bmod n^2.$$

**性质 2.** 密文对中的两个分量都各自保持了 Paillier 加密方案的加法同态性.

假定有理数 $M_A=\dfrac{\hat{M}_A}{\check{M}_A}$ 与 $M_B=\dfrac{\hat{M}_B}{\check{M}_B}$ 在方案 $\mathcal{E}$ 作用下对应的密文对分别为 $(\hat{c}_A=g_x^{\hat{M}_A}(\hat{r}_A)^n \bmod n^2,\ \check{c}_A=g_x^{\check{M}_A}(\check{r}_A)^n \bmod n^2)$ 与 $(\hat{c}_B=g_x^{\hat{M}_B}(\hat{r}_B)^n \bmod n^2,\ \check{c}_B=g_x^{\check{M}_B}(\check{r}_B)^n \bmod n^2)$. 令 $\hat{r}_A\cdot\hat{r}_B=\hat{r}_C+kn$，由已知 $\hat{r}_A,\hat{r}_B\in Z_n^*$，可得 $\hat{r}_A\cdot\hat{r}_B$ 不能被 $n$ 整除且 $\hat{r}_C\neq 0$，进而有

$$(\hat{r}_A\cdot\hat{r}_B)^n \bmod n^2=(\hat{r}_C+kn)^n \bmod n^2$$
$$=\sum_{\alpha=0}^{n}C_n^\alpha(\hat{r}_C)^{n-\alpha}(kn)^\alpha \bmod n^2$$
$$=(\hat{r}_C)^n \bmod n^2 \text{ (二项式展开定理).}$$

所以有

$$\hat{c}_A\cdot\hat{c}_B=[g_x^{\hat{M}_A}(\hat{r}_A)^n \bmod n^2]\cdot[g_x^{\hat{M}_B}(\hat{r}_B)^n \bmod n^2]$$
$$=g_x^{\hat{M}_B+\hat{M}_A}(\hat{r}_A\cdot\hat{r}_B)^n \bmod n^2$$
$$=g_x^{\hat{M}_B+\hat{M}_A}(\hat{r}_C)^n \bmod n^2.$$

同理，可得

$$\check{c}_A\cdot\check{c}_B=[g_x^{\check{M}_A}(\check{r}_A)^n \bmod n^2]\cdot[g_x^{\check{M}_B}(\check{r}_B)^n \bmod n^2]$$
$$=g_x^{\check{M}_A+\check{M}_B}(\check{r}_A\cdot\check{r}_B)^n \bmod n^2$$
$$=g_x^{\check{M}_A+\check{M}_B}(\check{r}_C)^n \bmod n^2.$$

所以，由加密方案 $\mathcal{E}$ 产生的密文对中的两个分量都各自保持了 Paillier 加密方案的加法同态性.

$\mathcal{E}$ 性质 1 与性质 2 的应用场景：A、B 两方各自拥有一个秘密 $a,b$，他们想利用 A 方的加密系统 $\mathcal{E}$ 通过安全计算 $\dfrac{r_1\cdot a+r_2}{r_1\cdot b+r_2}$ 或 $\dfrac{r_1\cdot b+r_2}{r_1\cdot a+r_2}$ ($r_1$, $r_2$ 为 B 方在

协议执行过程随机选取的随机数)实现安全计算 $a,b$ 是否相等，而当 $a\neq b$ 时，不能泄露双方的大小关系.

设 $(\hat{C},\check{C})$ 是对应于 $(r_1\cdot a+r_2,\ r_1\cdot b+r_2)$ 或 $(r_1\cdot b+r_2,\ r_1\cdot a+r_2)$ 的密文对，由 B 方利用 $\mathcal{E}$ 的性质 1 及性质 2 计算而来的. 因 B 方利用性质 1 将加密底数秘密地变换了，加之不知道 B 方发来的密文对 $(\hat{C},\check{C})$ 对应于 $(r_1\cdot a+r_2,\ r_1\cdot b+r_2)$ 或 $(r_1\cdot b+r_2,r_1\cdot a+r_2)$ 中哪一个，即便拥有密钥，A 方利用解密密钥，除了可以计算出 $a,b$ 是否相等外，再也得不到其它任何额外的信息.

### 3.5　有理数加密系统的安全性

**定理 2.** 如若 DTCR 在多项式时间内无法被计算求解，则称 $\mathcal{E}$ 是一个抗 IND-CPA 攻击的有理数加密方案.

证明. DTCR 挑战者按照 2.1 节中非对称加密系统 IND-CPA 安全性游戏执行的操作为：

1. 执行 Key-Generate 算法产生公钥 $(n,1+kn)$；
2. 均匀地选取 $(k\in Z_n)\wedge(k\neq 0)$，并计算：
$$g_x=1+k'(g-1)\bmod n^2.$$
3. $d\xleftarrow{R}\{0,1\}$；
4. 如果 $d=0$，则置 $T=(\hat{T},\check{T})=(\hat{r}^n \bmod n^2,\ \check{r}^n \bmod n^2)$，否则置 $T=R=(\hat{R},\check{R})$；
5. 将 $(n,1+n,(\hat{T}\,g_x^{\hat{M}_b} \bmod n^2,\check{T}\,g_x^{\check{M}_b} \bmod n^2),T)$ 发送给攻击者.

设 $\mathcal{E}$ (Key-Generation, Base-Transformation Encryption, Decryption)为 3.1 节中构造的加密方案、$\mathcal{A}$ 为攻击 $\mathcal{E}$ 的概率多项式时间内的任意的敌手，并将 $\mathcal{A}$ 在 $PubK_{\mathcal{A},\mathcal{E}}^{cpa}(n)$ 游戏中获胜的优势函数记作 $\epsilon$. 为了将 DTCR 问题的难解性与优势函数 $\epsilon$ 联系在一起，下面将在构造性证明框架下，以算法 $\mathcal{A}$ 为基元构建一个拟用于解决 DTCR 问题的算法 $\mathcal{B}$.

**算法 $\mathcal{B}$.**

1. 作为敌手，$\mathcal{B}$ 接收由 DTCR 挑战者发送的参数 $(n,1+kn,(n,R),T)$ (它并不知道 $(n,R)$ 具体来自分布 $D_{Ran}$ 还是分布 $D_{\mathcal{E}}$)；
2. $K_{Pub}\leftarrow(n,g=1+kn)$；
3. 将 $K_{Pub}$ 连同系统安全参数 $1^n$ 发给敌手 $\mathcal{A}$；
4. 接收来自 $\mathcal{A}$ 的消息 $M_b$ (其中 $M_b=\dfrac{\hat{M}_b}{\check{M}_b}\wedge M_b$, $b\in\{0,1\},|\hat{M}_0|=|\hat{M}_1|,|\check{M}_0|=|\check{M}_1|$；
5. $b\xleftarrow{R}\{0,1\}$；
6. $c^*\leftarrow(\hat{T}\,g_x^{\hat{M}_b} \bmod n^2,\check{T}\,g_x^{\check{M}_b} \bmod n^2)$ 并把 $c^*$ 转发至敌手 $\mathcal{A}$；
7. 接收 $\mathcal{A}$ 对于参数 $b$ 取值的猜测值 $b'\in\{0,1\}$；
8. 根据 $\mathcal{A}$ 的输出结果，生成一个输出 $d'$ 如果 $b=b'$，则令 $d'=0$；如果 $b\neq b'$，则令 $d'=1$ ）.

显然，按照上述方式利用算法 $\mathcal{A}$ 构造的算法 $\mathcal{B}$ 在多项式时间内可被完成（因为算法 $\mathcal{A}$ 是多项式时间内可被完成的算法），因此算法 $\mathcal{B}$ 在多项式时间内解决 DTCR 问题的概率可以用贝叶斯公式计算：

$$\begin{cases} Pr[d=d'] \\ =Pr[d=0]Pr[d=d'\,|\,d=0]+ \\ \quad Pr[d=1]Pr[d=d'\,|\,d=1] \\ =\dfrac{1}{2}Pr[d'=0\,|\,d=0]+\dfrac{1}{2}Pr[d'=1\,|\,d=1] \\ =\dfrac{1}{2}Pr[b=b'\,|\,d=0]+\dfrac{1}{2}Pr[b\neq b'\,|\,d=1] \end{cases}. \quad (4)$$

$d=0$ 时，据 DTCR 挑战者工作方式规定，DTCR 挑战者将置 $T=(\hat{T},\check{T})=(\hat{r}^n \bmod n^2,\check{r}^n \bmod n^2)$. 此时，因为在算法 $\mathcal{B}$ 执行过程中，由它向算法 $\mathcal{A}$ 提交的视图(view)与 $\mathcal{A}$ 在实际安全游戏 $PubK_{\mathcal{A},\mathcal{E}}^{cpa}$ 中的视图是计算不可区分的，所以在 $d=0$ 的前提下，$b=b'$ 的条件概率为 0.5 与敌手 $\mathcal{A}$ 在安全游戏 $PubK_{\mathcal{A},\mathcal{E}}^{cpa}$ 中获胜的优势之和，即

$$Pr[b=b'\,|\,d=0]=\frac{1}{2}+\epsilon. \quad (5)$$

$d=1$，据 DTCR 挑战者工作方式规定，DTCR 挑战者将置 $T=R=(\hat{R},\check{R})$. 在 $Z_{n^2}^*$ 上，参变量 $R$ 服从均匀分布，显然易得：$c^*\leftarrow(\hat{R}\,g_x^{\hat{M}_b} \bmod n^2,\check{R}\,g_x^{\check{M}_b} \bmod n^2)$ 在 $(Z_{n^2}^*,Z_{n^2}^*)$ 上也是一个服从均匀分布的参变量，且独立于参变量 $n,(\hat{M}_0,\check{M}_0),(\hat{M}_1,\check{M}_1)$ 与 $b$，又因 $n$、$g_x$、$\hat{R}\,g_x^{\hat{M}_b} \bmod n^2$、$\check{R}\,g_x^{\check{M}_b} \bmod n^2$ 和 $b$ 也是两两相互独立于彼此的随机参变量，所以敌手 $\mathcal{B}$ 无法借助参数 $K_{Pub}$ 与构造变量 $c^*$ 推演或计算出任何有关参变量 $b$ 取值的信息. 因此，我们可得出结论：事件 $b'$ ( $\mathcal{A}$ 输送的对参变量 $b$ 的猜测值)与事件 $b$ (由挑战者随机选择明文的下标)必为两个相互独立的事件. 又因参变量 $b$ 的取值是 $\mathcal{A}$ 猜测的，所以猜测参变量取值 $b=0$ 和 $b=1$ 的概率均等，由此可计算条件概率：

$$Pr[b=b'\,|\,d=1]=\frac{1}{2}. \quad (6)$$

由算式(4)、(5)和(6)得

$$Pr[d=d']=\frac{1}{2}\times\left(\frac{1}{2}+\epsilon\right)+\frac{1}{2}\times\frac{1}{2}=\frac{1}{2}+\frac{1}{2}\epsilon. \quad (7)$$

因此，算法 $\mathcal{B}$ 在解决 DTCR 过程中的成功优势为：

$$\begin{cases}\left|Pr[b=b']-Pr[b\neq b']\right|=\left|Pr[PubK^{cpa}_{\mathcal{B},\Pi}(n)=1]-\dfrac{1}{2}\right| \\ =\dfrac{\epsilon}{2}\end{cases}. \tag{8}$$

事实上，DTCR 问题在第 2.4 节中已被证明是一个难解问题，这意味着算法 $\mathcal{B}$ 在解决 DTCR 过程中取得的成功优势是可忽略的，即 $\frac{\epsilon}{2}$ 是一个可忽略的量. 这蕴含着 $\epsilon$ 也是一个可忽略量. 这说明敌手 $\mathcal{A}$ 在攻击 $\mathcal{E}$ 的游戏 $PubK^{cpa}_{\mathcal{A},\mathcal{E}}$ 中获胜的优势 $\epsilon$ 是可忽略的. 因此，方案 $\mathcal{E}$ 具有 IND-CPA 安全性.

如果利用方案 $\mathcal{E}$ 将 $M_0=(\hat{M}_0,\check{M}_0)$ 与 $M_1=(\hat{M}_1,\check{M}_1)$ ($|\hat{M}_0|=|\hat{M}_1|$ 并且 $|\check{M}_0|=|\check{M}_1|$)分别为加密成 $c_0=(\hat{c}_0,\check{c}_0)$ 和 $c_1=(\hat{c}_1,\check{c}_1)$，那么 $c_0=(\hat{c}_0,\check{c}_0)$ 和 $c_1=(\hat{c}_1,\check{c}_1)$ 是两个计算上不可区分的量，即 $(\hat{c}_0,\check{c}_0)\overset{c}{\equiv}(\hat{c}_1,\check{c}_1)$.

证毕.

# 4 无匹配差错的两方 PSI 计算思想

为了实现交集元素保密求解差错为“0”的目的，同时也为了方便保密集合交集计算，我们在构造无匹配差错的两方 PSI 计算协议采用了有理数加密方案 $\mathcal{E}$ 中构造保密比值的思想和一种称之为集合的定长向量编码方法.

## 4.1 用 $\mathcal{E}$ 构造两个集合对应元素的保密比值

将有理数加密方案 $\mathcal{E}$ 用于构造多方保密计算协议时，参与者可以根据需要，利用性质 1 和性质 2 改变一个密文的加密底数，这使得本来由参与各方自由选择的加密底数统一变为相同的加密底数，从而使得解密者只有同时拥有底数相同的密文对才可以正确求解出一个有理数. 不失一般性，在此将 Alice 和 Bob 视作安全多方计算协议中的两个参与者. 假定整数 $1\leqslant m_A,m_B\leqslant n$ 分别属于 Alice 和 Bob，并且 Alice 拥有公钥加密方案的私钥，Bob 只知道该公钥方案的公钥，则结合上述性质 2，Alice 和 Bob 按照如下方式可以协同计算 $(m_A+m_x)$ 与 $(m_B+m_x)$ 两个量的比值，而不会泄露自己的信息：

(1) Alice 计算 $m_A$ 对应的密文 $c_A=g^{m_A}r_A^n \bmod n^2$，计算完成后将其发送给 Bob.

(2) Bob 得到 Alice 的密文 $c_A$ 后，随机选择 $r_{B1}\in Z_n^*$，$k'\in Z_n$ (满足 $k'\cdot(g-1)>n$)，然后计算：

$$\begin{aligned}c_{A_{g_x}}&=c_A^{k'} \bmod n^2\\&=(1+kn)^{k'm_A} \bmod n^2\\&=(1+k'kn)^{m_A} \bmod n^2\\&=g_x^{m_A} \bmod n^2,\end{aligned}$$

$$c_{m_x}=(1+k'\cdot(g-1)\cdot m_x)r_{B1}^n \bmod n^2,$$

$$c'_A=c_{A_{g_x}}\cdot c_{m_x} \bmod n^2.$$

(3) 随机选择 $r_{B1},r_{B2}\in Z_n^*$，对于 $m_B<n$，计算：

$$c_B=(1+k'\cdot(g-1)\cdot m_B+m_x)r_B^n \bmod n^2.$$

并将 $(c'_A,c_B)$ 发送给解密方.

(4) 收到 $(c'_A,c_B)$ 后，Alice 通过执行运算：

$$\frac{\mathcal{L}((c'_A)^{\lambda} \bmod n^2)}{\mathcal{L}(c_B^{\lambda} \bmod n^2)},$$

就可以得到 $(m_A+m_x)$ 与 $(m_B+m_x)$ 两个量的比值.

我们注意到：(1)用 $(m_A+m_x)$ 与 $(m_B+m_x)$ 两个量的比值可以替代 $\frac{m_A}{m_B}$ 用于安全比较 $m_A$ 与 $m_B$ 的大小，并且当 $\frac{m_A+m_x}{m_B+m_x}=1$，则必有 $m_A=m_B$；(2)若 $m_A\in A$ 并且 $m_B\in B$，则 $\frac{m_A+m_x}{m_B+m_x}=1$ 蕴含 $m_A=m_B$ 且 $m_A$ 为集合 $A$ 与 $B$ 的公共元素.

## 4.2 集合的定长向量编码

1. 编码思想. 以全集 $E$ 为参照为其子集 $E_i$ 构造一个新的映象集合 $E'$，满足：

(1) $|E|=|E'|=m$；

(2) 如果全集中的元素 $e_i$ 在子集 $E_i$ 中，则 $E'$ 与 $E$ 的第 $i$ 个元素相同，否则 $E'$ 中的其他元素映射为一个随机数. 例如：如果 $E_i=\{\overline{e}_1,\overline{e}_2,\overline{e}_3,\overline{e}_4,\overline{e}_5,\overline{e}_6\}=\{e_1,e_4,e_i,e_{m-4},e_{m-2},e_{m-1}\}$，则 $E_i$ 的定长向量编码如图 3 所示.

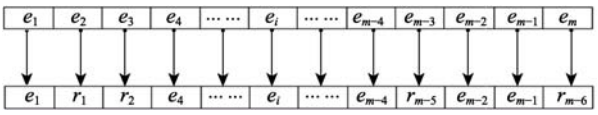


图 3 集合 $E_i$ 参照全集 $E$ 的定长向量编码

2. 区分映射值域的集合定长向量编码. 假定保密集合交集计算协议中两个参与方分别持有保密集合 $E_i$ 与 $E_j$，$E_i$ 与 $E_j$ 由两个参与者采用定长向量编码方法编码后的集合分别为 $E'$ 与 $E''$，并假定编码时集合编码映射的原象和象值都取自相同的集合 $\{1,2,\cdots,n\}$. 我们注意到：如果两个集合在编码

时，集合编码映射的原象和象值都取自集合 $\{1,2,\cdots,n\}$，则很可能会出现如下情形：$(E'\cap E'')\neq \mathrm{E}_i\cap \mathrm{E}_j$ 即 $(E'\cap E'')\supseteq E_i\cap E_j$.

这是因为：一方面，如果 $E_i$ 与 $E_j$ 都不具有全集 E 中的某个元素，则采用定长向量编码由两个不同的人分别对它们进行编码时可能会映射为同一个随机数，即两方所选择的随机数也会以 $\frac{1}{n^2}$ 的概率发生碰撞. 另一方面，如果 $E_i$ 与 $E_j$ 只有一方具有全集 $E$ 中的某个元素，则采用定长向量编码由两个不同的人分别对它们进行编码时，选择随机数映射的一方所选择的随机数会以 $\frac{1}{n}$ 的概率撞上另一方的某个元素.

为了避免在两方保密集合交集计算中产生上述碰撞，达到无匹配差错的保密集合交集计算目的，我们将 $Z_n$ 分成三段：$Z_{\lfloor \frac{n}{3}\rfloor}$、$(Z_{\lceil \frac{5n}{6}\rceil}-Z_{\lfloor \frac{2n}{3}\rfloor})\cup(Z_{\lceil \frac{n}{2}\rceil}-Z_{\lceil \frac{n}{3}\rceil})$ 和 $(Z_{\lceil \frac{2n}{3}\rceil}-Z_{\lfloor \frac{n}{2}\rfloor})\cup(Z_n-Z_{\lfloor \frac{5n}{6}\rfloor})$，其中 $Z_{\lfloor \frac{n}{3}\rfloor}$ 表示集合 $E$ 及其子集 $E_i$ 中元素(或者数据库中数据项)的取值范围，$A_0=(Z_{\lceil \frac{5n}{6}\rceil}-Z_{\lfloor \frac{2n}{3}\rfloor})\cup(Z_{\lceil \frac{n}{2}\rceil}-Z_{\lceil \frac{n}{3}\rceil})$，$A_1=(Z_{\lceil \frac{2n}{3}\rceil}-Z_{\lfloor \frac{n}{2}\rfloor})\cup(Z_n-Z_{\lfloor \frac{5n}{6}\rfloor})$ 为编码时可供参与方选取随机数的集合，简称编码随机可选集合.

设 $X\subseteq E=\{e_i\}_{i=1,\cdots,m}$ 并且规定：

（1）$|E|\leqslant\left\lfloor\frac{n}{3}\right\rfloor$ 或 $m\leqslant\left\lfloor\frac{n}{3}\right\rfloor$，其中 $\lfloor\cdot\rfloor$ 为下取整函数；

（2）$e_i\in Z_{\lfloor \frac{n}{3}\rfloor}$；

（3）$\{e_i\}_{i=1,\cdots,m}$ 中的元素互不相同.

以集合 $E$ 为参照，为集合 $X=\{x_i\}_{i=1,\cdots,k}$ $(k\leq m)$ 构造定长的向量编码 $\boldsymbol{x}=(x_1,x_2,\cdots,x_m)$：

(1) 若集合的元素都为整数，则按照如下方式对集合 $X=\{x_i\}_{i=1,\cdots,k}$ 编码

$$x_i=\begin{cases} r_i\in A_d \text{或} A_{\bar{d}}(d\in\{0,1\}), & e_i\notin X \\ e_i, & e_i\in X \end{cases};$$

(2) 若集合的元素为有理数，则按照如下方式编码：

①将元素 $e_i$ 表示成 $(\hat{e_i},\check{e_i})$ 使其满足：$e_i=\frac{\hat{e_i}}{\check{e_i}}$，$\gcd(\hat{e_i},\check{e_i})=1$；

② 按照如下方式对集合 $X=\{x_i\}_{i=1,\cdots,k}$ $(k\leqslant m)$ 构造定长的向量编码 $\boldsymbol{x}=(x_1,x_2,\cdots,x_m)$：

$$x_i=\begin{cases} \hat{r_i},\check{r_i}\in A_d \text{ 或} A_{\bar{d}}(d\in\{0,1\}), & \\ \qquad ((\hat{r_i}\neq\check{r_i})\wedge(\gcd(\hat{r_i},\check{r_i})=1)) & e_i\notin X \\ (\hat{e_i},\check{e_i}), & e_i\in X \end{cases}$$

### 4.3　无匹配差错的PSI计算思路

我们采用区分映射值域的集合定长向量编码方法和有理数加密方案 $\mathcal{E}$ 实现无匹配差错的 PSI 计算，其中区分映射值域的集合定长向量编码方法的主要作用是避免匹配过程中的数值碰撞问题，有理数加密方案 $\mathcal{E}$ 主要作用是无匹配差错的 PSI 计算. 实现无匹配差错的 PSI 计算思路如下：

(1) Alice 和 Bob 分别先将各自的私有集合采用定长向量编码方法编码成 $\boldsymbol{x}=(x_1,x_2,\cdots,x_m)$ 与 $\boldsymbol{y}=(y_1,y_2,\cdots,y_m)$；

(2) 双方采用有理数加密方案 $\mathcal{E}$，并按照如下方式将 $\boldsymbol{x}=(x_1,x_2,\cdots,x_m)$ 与 $\boldsymbol{y}=(y_1,y_2,\cdots,y_m)$ 构造成密文对 $c_i=(\hat{c_i},\check{c_i})$，其中 $(\hat{c_i},\check{c_i})$ 是 $(x_i+\omega_i,\ y_i+\omega_i)$ 或者 $(y_i+\omega_i,x_i+\omega_i)$ 对应的密文：

①Alice 用自己的公钥为每个 $x_i$ 计算：$Enc(x_i)=g^{x_i}r_{Ai}^n \bmod n^2$，并按序发送给 Bob；

②Bob 得到 $Enc(x_i)$ 后，随机选择 $\hat{r}_{Bi},\check{r}_{Bi}\in Z_n^*$，$\omega_i,k_i'\in Z_n$ (满足 $k'\cdot(g-1)>n$)，然后通过计算：

$$Enc(x_i+\omega_i)=$$

$$Enc^{k'}(x_i)\cdot(1+k_i'(g-1))^{\omega_i}\cdot(\hat{r}_{Bi})^n \bmod n^2,$$

$$Enc(y_i+\omega_i)=(1+k_i'\cdot(g-1)\cdot(y_i+\omega_i))\cdot(\check{r}_{Bi})^n \bmod n^2$$

随机选择 $(\hat{c_i},\check{c_i})\in\{(Enc(x_i+\omega_i),Enc(y_i+\omega_i)),(Enc(y_i+\omega_i),Enc(x_i+\omega_i))\}$ 并按序发送给 Alice.

③Bob 执行解密运算 $Dec(c_i)$，如果 $Dec(c_i)=1$，则表示 $x_i=y_i$，即 $e_i$ 为 Alice 和 Bob 私有集合交集的元素，否则，$e_i$ 不是 Alice 和 Bob 私有集合交集的元素.

## 5　无匹配差错的两方 PSI 计算

### 5.1　集合交集的相关计算问题

我们研究两类情形下的保密集合交集计算问题(不失一般性，本文假定 Alice 和 Bob 是两方保密协议的参与方)：

情形 A：Alice 和 Bob 分别拥有一个集合 $S_a$ 和

$S_b$，他们想知道他们所持有的集合中有哪些元素是相同的,但不泄露其他任何信息,即 Alice 和 Bob 在不泄漏 $S_a$、$S_b$ 的前提下，协同计算：

$$S_a \cap S_b .$$

情形 B：Alice 和 Bob 分别拥有一个集合 $S_a$ 和 $S_b$，二人只想测试集合 $S_a$ 和 $S_b$ 是否有交集，如果有交集则求交集的势，但不泄露其他任何信息，即 Alice 和 Bob 在不泄漏 $S_a$、$S_b$ 的前提下协同计算：

$$S_a \cap S_b \overset{?}{=} \varnothing, \qquad |S_a \cap S_b| .$$

### 5.2 针对情形A和情形B的PSI协议

5.2.1 针对情形 A、面向整数元素的 PSI 计算协议 $\Pi_1$

1. 具体协议

输入：Alice 和 Bob 各自的保密集合 $S_a$ 和 $S_b$、编码随机可选集合标识码 $\{d,\bar{d}\}$ ($d \in \{0,1\}$).

输出：$S_a \cap S_b$.

(1) Alice 先运行方案 $\mathcal{E}$ 的密钥生成算法产生公私钥：公钥为 $(n,1+n)$，私钥为 $\lambda$；并按照集合的定长向量编码方法将自己的集合 $S_a$ 编码成 $m$ 长的向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$；

(2) Alice 对向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 各分量 $a_i$ ($i=1,2,\cdots,m$) 做如下计算：

$$c_{a_i}=(1+kn)^{a_i} r_{a_i}^n \bmod n^2. \tag{9}$$

并将得到的密文向量 $\boldsymbol{C_A}=(c_{a_1},\cdots,c_{a_m})$ 连同自己的随机可选集的标识符 $d \in \{0,1\}$ 发送给 Bob；

(3) Bob 收到 $C_A$ 与 $d \in \{0,1\}$ 后按照如下方式工作：

① 利用集合的定长编码方法 1,借助集合 $A_{\bar{d}}$ 将自己的集合 $S_b$ 编码成 $m$ 长的向量 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$；

② 随机选择 $4m$ 个不等的随机数 $k_{b_1},k_{b_2},\cdots,k_{b_m} \in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m} \in Z_n$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m} \in Z_n^*$，$r_{b\check{x}_1},r_{b\check{x}_2},\cdots,r_{b\check{x}_m} \in Z_n^*$，利用有理数加密方案 $\mathcal{E}$ 加密性质计算：

$$\begin{cases} c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times \\ \qquad (1+(g-1)k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2 & \text{(10a)}, \\ c_{(r_{b_i}+b_i)}=(1+(g-1)k_{b_i}r_{b_i}n)r_{b\check{x}_i}^n \bmod n^2 & \text{(10b)} \end{cases}$$

得到 $m$ 个密文对；

③ 将 $m$ 个密文对 $(c_{(r_{b_i}+a_i)},c_{(r_{b_i}+b_i)}),1\leqslant i\leqslant m$ 分别做对内分量随机置换得到密文对序列：$(c_{L_i},c_{R_i})$, $1\leqslant i\leqslant m$，并发给 Alice.

(4) Alice 收到 $(c_{L_i},c_{R_i})$, $1\leqslant i\leqslant m$ 后计算：

$$\partial_i=\frac{c_{L_i}^{\lambda} \bmod n^2}{c_{R_i}^{\lambda} \bmod n^2} \tag{11}$$

记录 $\partial_i=1$ 的下标 $i$，并将它们发送给 Bob.

2. 协议 $\Pi_1$ 的正确性

因为集合 $S_a=\{x_i\}_{i=1,\cdots,k_1}$ 与 $S_b=\{x'_1,x'_2,\cdots,x'_{k_2}\}$ 对应的定长编码向量分别为 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$，其中

$$a_i=\begin{cases} \hat{r}_i \in A_d, & e_i \notin S_a \\ e_i, & e_i \in S_a \end{cases},$$

$$b_i=\begin{cases} \check{r}_i \in A_{\bar{d}}, & e_i \notin S_b \\ e_i, & e_i \in S_b \end{cases},$$

又因为 $A_0 \cap A_1=\varnothing$，所以 $S_a \cap S_b=\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$，即如果 $e_i$ 是集合 $S_a$ 和 $S_b$ 的公共元素，则集合 $S_a$ 和 $S_b$ 的定长编码向量对应的第 $i$ 个元素都为 $e_i$. 下面从元素 $e_i$，$a_i$ 与 $b_i$ 三者的关系论证 $S_a \cap S_b=\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$ 的正确性：

(1) 当 $e_i \notin S_a \wedge e_i \notin S_b$ 时,因为 Alice 和 Bob 编码时选用到的随机数 $\hat{r}_i$ 和 $\check{r}_i$ 选自两个不同的编码随机可选集合，所以 $a_i \neq b_i$，从而必有：

$$\frac{\rho a_i+\omega}{\rho b_i+\omega}\neq 1, \quad \frac{\rho b_i+\omega}{\rho a_i+\omega}\neq 1 \ (\rho,\omega \in Z_n^+);$$

(2) 当 $e_i \notin S_a \wedge e_i \in S_b$ 或者 $e_i \in S_a \wedge e_i \notin S_b$ 时，因为 Alice 和 Bob 编码时选用的随机数 $\hat{r}_i$、$\check{r}_i$ 以及 $e_i$ 分别选自三个不同的编码随机可选集合，所以 $a_i \neq b_i$，从而必有：$\dfrac{\rho a_i+\omega}{\rho e_i+\omega}\neq 1$，$\dfrac{\rho e_i+\omega}{\rho a_i+\omega}\neq 1$，$\dfrac{\rho b_i+\omega}{\rho e_i+\omega}\neq 1$，$\dfrac{\rho e_i+\omega}{\rho b_i+\omega}\neq 1$；

(3) 因为当 $e_i \in S_a \wedge e_i \in S_b$ 时元素 $e_i$，$a_i$ 与 $b_i$ 两两彼此相等，所以必有：$\dfrac{\rho a_i+\omega}{\rho b_i+\omega}=\dfrac{\rho e_i+\omega}{\rho e_i+\omega}=1$.

显然 $\dfrac{\rho a_i+\omega}{\rho b_i+\omega}=1$ 必然有 $e_i$ 在交集 $\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$ 中,并且协议中 Bob 利用同态操作和 Alice 的密文向量 $\boldsymbol{C_A}$ 构造的密文对向量 $((c_{(r_{b_1}+a_1)},c_{(r_{b_1}+b_1)}),(c_{(r_{b_2}+a_2)},c_{(r_{b_2}+b_2)}),\cdots,(c_{(r_{b_m}+a_m)},c_{(r_{b_m}+b_m)}))$ 的下标与编码向量集合 $\{a_i\}_{i=1,\cdots,m}$ 和 $\{b_i\}_{i=1,\cdots,m}$ 完全一致，因此，通过记录

$$\partial_i=\frac{c_{L_i}^{\lambda} \bmod n^2}{c_{R_i}^{\lambda} \bmod n^2}=1$$

的下标 $i$，Alice 和 Bob 就能够精确地计算出两个集合 $S_a=\{x_i\}_{i=1,\cdots,k_1}$、$S_b=\{x'_1,x'_2,\cdots,x'_{k_2}\}$ 的交集 $S_a\cap S_b$. 因为 $\partial_i=1$时，集合 $S_a$ 对应的定长向量编码向量的第 $i$ 个分量和 $S_b$ 对应的定长向量编码向量的第 $i$ 个分量相等，并且等于集合 $\mathrm{E}=\{e_i\}_{i=1,\cdots,m}$ 中第 $i$ 元素 $e_i$.

3. 协议 $\Pi_1$ 的安全性

**定理 3.** Alice 和 Bob 采用协议 $\Pi_1$ 可以安全地实现保密计算 $S_a\cap S_b$，其中 $S_a$ 与 $S_b$ 都是由整型元素组成的.

证明. 协议 $\Pi_1$ 能否实现保密计算 $S_a\cap S_b$ 的关键是协议执行后有没有造成 Alice 与 Bob 私有信息的泄露. 下面将严格按照 2.2 节中抗半诚实敌手攻击安全多方计算模型声明的安全标准和方法证明：在保密计算 $S_a\cap S_b$ 的过程中，Alice 与 Bob 都不会得到除计算结果 $S_a\cap S_b$ 外任何有关对方的其他私有信息.

对于 Bob 私有信息的安全性：如果在 Alice 受控于模拟器 $\mathcal{S}_1^{\Pi_1}$ 的情形下，多项式时间内的敌手 $\mathcal{S}_1^{\Pi_1}$ 获得的信息并不多于 Alice 在实际执行协议中的视图内容，则称协议 $\Pi_1$ 完成后 Bob 私有信息依然是安全的.

构造一个控制 Alice、能够在多项式时间内模拟协议 $\Pi_1$ 整个执行过程的模拟器 $\mathcal{S}_1^{\Pi_1}$，$\mathcal{S}_1^{\Pi_1}$ 的输入为：根据 Alice 的私有输入 $S_a=\{a_i\}_{i=1,\cdots,k_a}$ 和 $S_a\cap S_b$ 构造的利于攻击 Bob 的集合 $S'_a=\{a'_1,a'_2,\ldots,a'_{k_a}\}$，Bob 随机选择的 $4m$ 个不等的随机数 $k_{b_1},k_{b_2},\cdots,k_{b_m}\in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m}\in Z_n^*$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m}\in Z_n^*$，$r_{b\vee x_1},r_{b\vee x_2},\cdots,r_{b\vee x_m}\in Z_n^*$，以及 Bob 的私有集合 $S_b=\{b_i\}_{i=1,\cdots,k}$. 作为敌手，模拟器 $\mathcal{S}_1^{\Pi_1}$ 产生的视图为 $(\{i,c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2,(c'_{L_i},c'_{R_i})\}_{1\leqslant i\leqslant m})$；而保密集合交集协议 $\Pi_1$ 的实际执行中，Alice 的实际视图为 $(\{i,c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2,(c_{L_i},c_{R_i})\}_{1\leqslant i\leqslant m})$. 因为密文 $(c'_{L_i},c'_{R_i})_{1\leqslant i\leqslant m}$ 是 Bob 经过下述方式构造的：

(1) 由密文$(1\leqslant i\leqslant m)$利用方案 $\mathcal{E}$ 的性质 1 和性质 2 经运算(10a)与(10b)计算得到 $m$ 个密文对 $(c_{(r_{b_i}+a_i)},c_{(r_{b_i}+b_i)})$ $c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2$；

(2) 对 $m$ 个密文对 $(c_{(r_{b_i}+a_i)},c_{(r_{b_i}+b_i)})$ 分别做对内分量随机置换得到密文对序列 $C'_B=(c'_{L_i},c'_{R_i})_{1\leqslant i\leqslant m}$.

对于$1\leqslant i\leqslant m$，Alice 获得 $C'_B$ 后通过解密运算后最多只能得到由两个方程(其中每个方程各包含 3 个不同的未知数)组成的方程组，不可能通过联立方程组计算出具体的 $b_i$. 同理对于$1\leqslant i\leqslant m$，Alice 获得 $(c_{L_i},c_{R_i})$ 也不可能通过联立方程组计算出具体的 $b_i$. 这也就是说 $\mathcal{S}_1^{\Pi_1}$ 满足安全定义关系式(1a).

对于 Alice 私有信息的安全性：假定控制 Bob 的敌手 $\mathcal{S}_2^{\Pi_1}$ 在 Bob 不参与的情况下，能够在多项时间内模拟出协议 $\Pi_1$ 的执行过程. 如果在该假定条件下，多项式时间内的敌手 $\mathcal{S}_2^{\Pi_1}$ 获得的信息并不多于 Bob 在实际执行协议中的视图内容，则 Alice 的私有信息是安全的.

首先构造一个控制 Bob 且能在多项式时间内模拟执行协议 $\Pi_1$ 的模拟器 $\mathcal{S}_2^{\Pi_1}$. 该模拟器的输入为：Alice 的私有集合 $S_a=\{a_i\}_{i=1,\cdots,k_a}$、模拟器 $\mathcal{S}_2^{\Pi_1}$ 根据 Bob 的私有集合 $S_b=\{b_i\}_{i=1,\cdots,k_b}$ 和 $S_a\cap S_b$ 构造的有利于获取 Alice 私有信息的集合 $S'_b=\{b'_1,b'_2,\ldots,b'_{k_b}\}$. 作为敌手，模拟器 $\mathcal{S}_2^{\Pi_1}$ 产生的视图为$(c_{a'_i}=(1+n)^{a'_i}r_{a'_i}^n \bmod n^2$，$c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)\times r_{b\hat{x}'_i}^n \bmod n^2, c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}n)r_{b\vee x'_i}^n \bmod n^2, i)$，其中$1\leqslant i\leqslant m$；而 Bob 在协议 $\Pi_1$ 的实际执行中产生的视图为$(c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2, c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{a_i}^n \bmod n^2$，$c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}\cdot n)r_{b_i}^n \bmod n^2, i)$，其中$1\leqslant i\leqslant m$. 无论在模拟协议还是实际协议中 Bob 接收到的皆为另一个参与者(Alice 或 $\mathcal{S}_2^{\Pi_1}$)私有信息在 $\mathcal{E}$ 作用下的密文，因为 Bob 没有 $\mathcal{E}$ 的解密密钥，并且方案 $\mathcal{E}$ 已被证明在选择明文攻击下具有语义不可区分安全性，因此，对于 Bob 或者控制着 Bob 的敌手而言，$c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2$ 与 $c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2$ 是计算不可区分的，并且 $c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)\cdot r_{b\hat{x}'_i}^n \bmod n^2$ 与 $c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2$ 也是计算不可区分的. 从而可得：$\mathcal{S}_2(c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2$，$c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2$，$c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}'_i}^n \bmod n^2$，$c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}n)r_{b\vee x'_i}^n \bmod n^2, 1\leqslant i\leqslant m)$与真实协议中视图 $\mathrm{View}_{\mathbf{B}}^{\Pi_1}(c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2$，$c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2$，$c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}n)\ r_{b\vee x_i}^n \bmod n^2, 1\leqslant i\leqslant m)$ 计算不可区分，即 $\mathcal{S}_2^{\Pi_1}$ 符合式(1b)定义的安全条件.

综上可得：Alice 和 Bob 在协议执行中无信息泄

漏，即协议 $\Pi_1$ 满足两方保密计算定义.

证毕.

5.2.2 针对情形 B 的、面向整数的保密集合交集测试和交集势的计算协议 $\Pi_2$

1. 具体协议

输入：Alice 和 Bob 各自的保密集合 $S_a$ 和 $S_b$、编码随机可选集合标识码 $\{d,\bar{d}\}$ ($d\in\{0,1\}$).

输出：如果 $S_a\cap S_b=\varnothing_a$，输出 0；如果 $S_a\cap S_b\neq\varnothing_a$，输出 $|S_a\cap S_b|$.

(1) 准备阶段：Alice 先运行方案 $\mathcal{E}$ 的密钥生成算法产生公私钥：公布公钥 $(n,1+n)$，保留私钥 $(n,1+n)$；Alice 和 Bob 按照集合的定长向量编码方法，将各自的集合 $S_a$ 和 $S_b$ 分别编码成 $m$ 长的向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$；

(2) 保密集合交集测试和交集势的计算：

①Alice 采用方案 $\mathcal{E}$ 对向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 的各分量 $a_i$ ($i=1,2,\cdots,m$)进行加密得到向量 $\boldsymbol{a}$ 对应的密文向量 $\boldsymbol{C_A}=(c_{a_i})_{i=1,\cdots,m}$，并将它发送给 Bob；

②收到 $\boldsymbol{C_A}=(c_{a_i})_{i=1,\cdots,m}$ 后，Bob 随机选择 $4m$ 个不等的随机数 $k_{b_1},k_{b_2},\cdots,k_{b_m}\in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m}\in Z_n^*$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m}\in Z_n^*$，$r_{b\check{x}_1},r_{b\check{x}_2},\cdots,r_{b\check{x}_m}\in Z_n^*$，按照下式

$$\begin{cases} c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times \\ \qquad (1+(g-1)k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2 & (12a)\\ c_{(r_{b_i}+b_i)}=(1+(g-1)k_{b_i}r_{b_i}n)r_{b\check{x}_i}^n \bmod n^2 & (12b)\end{cases}$$

计算出 $m$ 个密文对 $(c_{(r_{b_i}+a_i)},c_{(r_{b_i}+b_i)}),1\leqslant i\leqslant m$ 后，先将这 $m$ 个密文对做对内分量间的随机置换，然后再对这 $m$ 个密文对做对间的随机置换，将随机置换后的密文对序列记作：$(c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m})$，并把它发给 Alice.

③ Alice 收到 $(c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m})$ 后计算：

$$\theta=\sum_{i=1}^{m}P\left(\frac{c_{L_i}^{\lambda} \bmod n^2}{c_{R_i}^{\lambda} \bmod n^2}\right),\ \text{其中}P(y)=\begin{cases}1, & y=1\\ 0, & y\neq 1\end{cases}. \tag{13}$$

然后将 $\theta$ 发送给 Bob.

2. 协议 $\Pi_2$ 的正确性

因为集合 $S_a=\{x_i\}_{i=1,\cdots,k_1}$ 与 $S_b=\{x_1',x_2',\cdots,x_{k_2}'\}$ 对应的定长编码向量分别为 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$，其中

$$a_i=\begin{cases}\hat{r}_i\in A_d, & e_i\notin S_a\\ e_i, & e_i\in S_a\end{cases},$$

$$b_i=\begin{cases}\check{r}_i\in A_{\bar{d}}, & e_i\notin S_b\\ e_i, & e_i\in S_b\end{cases},$$

又因为 $A_0\cap A_1=\varnothing$，所以 $S_a\cap S_b=\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$，即如果 $e_i$ 为集合 $S_a$ 和 $S_b$ 的公共元素，则集合 $S_a$ 和 $S_b$ 的定长编码向量对应的第 $i$ 个元素都为 $e_i$. 下面从元素 $e_i$，$a_i$ 与 $b_i$ 三者的关系论证 $S_a\cap S_b=\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$ 的正确性：

(1) 当 $e_i\notin S_a\wedge e_i\notin S_b$ 时，因为 Alice 和 Bob 编码时选用到的随机数 $\hat{r}_i$ 和 $\check{r}_i$ 选自两个不同的编码随机可选集合，所以 $a_i\neq b_i$，从而必有：

$$\frac{\rho a_i+\omega}{\rho b_i+\omega}\neq 1,\ \frac{\rho b_i+\omega}{\rho a_i+\omega}\neq 1\ (\rho,\omega\in Z_n^+);$$

(2) 当 $e_i\notin S_a\wedge e_i\in S_b$ 或者 $e_i\in S_a\wedge e_i\notin S_b$ 时，因为 Alice 和 Bob 编码时选用到的随机数 $\hat{r}_i$、$\check{r}_i$ 以及 $e_i$ 分别选自三个不同的编码随机可选集合，所以 $a_i\neq b_i$，从而必有：$\frac{\rho a_i+\omega}{\rho e_i+\omega}\neq 1$，$\frac{\rho e_i+\omega}{\rho a_i+\omega}\neq 1$，$\frac{\rho b_i+\omega}{\rho e_i+\omega}\neq 1$，$\frac{\rho e_i+\omega}{\rho b_i+\omega}\neq 1$；

(3) 因为当 $e_i\in S_a\wedge e_i\in S_b$ 时元素 $e_i$，$a_i$ 与 $b_i$ 两两彼此相等，所以必有 $\frac{\rho a_i+\omega}{\rho b_i+\omega}=\frac{\rho e_i+\omega}{\rho e_i+\omega}=1$.

如果 $\frac{\rho a_i+\omega}{\rho b_i+\omega}=1$，则 $e_i$ 必在交集 $\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$ 中，其中 $i$ 用于指示集合 $\{1,2,\cdots,m\}$ 的一个随机置换集合的第 $i$ 个元素，因此通过统计结果为 1 的个数：

$$\theta=\sum_{i=1}^{m}P\left(\frac{c_{L_i}^{\lambda} \bmod n^2}{c_{R_i}^{\lambda} \bmod n^2}\right),\ \text{其中}P(y)=\begin{cases}1, & y=1\\ 0, & y\neq 1\end{cases}. \tag{14}$$

能够精确地计算出集合 $\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$ 的势，即 $|S_a\cap S_b|$.

3. 协议 $\Pi_2$ 的安全性

**定理 4.** Alice 和 Bob 采用协议 $\Pi_2$ 可以安全地实现保密计算 $|S_a\cap S_b|$，其中 $S_a$ 与 $S_b$ 都是由整型元素组成的.

证明. Alice 和 Bob 能否利用 $\Pi_2$ 安全地实现保密计算 $|S_a\cap S_b|$ 的关键是协议完成后有没有造成 Alice 与 Bob 私有信息的泄露. 而 Alice 与 Bob 私有信息有无泄露重点是协议完成后拥有私钥的 Alice 能否得到 $|S_a\cap S_b|$ 中的元素. 下面将严格按照 2.2 节中抗半诚实敌手攻击安全多方计算模型声明的安全标准和方法进行证明：在安全计算交集势的过程

中，Alice 与 Bob 都不会得到除协议输出 $|S_a \cap S_b|$ 外的、有关对方的其他任何私有信息.

对于 Bob 私有信息的保密性：如果在完全控制 Alice 的情形下，多项式时间内的敌手 $\mathcal{S}_1^{\Pi_2}$ 获得的信息并不多于 Alice 在实际执行协议中的视图内容，则称协议 $\Pi_2$ 完成后 Bob 私有信息依然是保密的.

构造一个控制 Alice、能够在多项式时间内模拟协议 $\Pi_2$ 整个执行过程的模拟器 $\mathcal{S}_1^{\Pi_2}$，$\mathcal{S}_1^{\Pi_2}$ 的输入为：根据 Alice 的私有输入 $S_a=\{a_i\}_{i=1,\cdots,k_a}$ 和 $|S_a \cap S_b|$ 构造的利于攻击 Bob 的集合 $S'_a=\{a'_1,a'_2,\ldots,a'_{k_a}\}$，Bob 随机选择的 $4m$ 个不等的随机数 $k_{b_1},k_{b_2},\cdots,k_{b_m}\in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m}\in Z_n^*$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m}\in Z_n^*$，$r_{b\vee x_1},r_{b\vee x_2},\cdots,r_{b\vee x_m}\in Z_n^*$，以及 Bob 的私有集合 $S_b=\{b_i\}_{i=1,\cdots,k_b}$. 作为敌手，$\mathcal{S}_1^{\Pi_2}$ 产生的视图为 $\{c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2,\ (c'_{L_i},c'_{R_i})_{i=1,\cdots,m},i\}$ $(1\leqslant i\leqslant m)$，其中 $(c'_{L_i},c'_{R_i})_{i=1,\cdots,m}$ 是 Bob 按照如下方式产生的：

(1) 通过计算

$$\begin{cases} c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times \\ \qquad (1+(g-1)k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2 \qquad (15a)\\ c_{(r_{b_i}+b_i)}=(1+(g-1)k_{b_i}r_{b_i}n)r_{b\vee x_i}^n \bmod n^2 \qquad (15b)\end{cases}$$

得到 $m$ 个密文对 $(c_{(r_{b_i}+a'_i)},c_{(r_{b_i}+b_i)})$，$1\leqslant i\leqslant m$；

(2) 先对这 $m$ 个密文对做对内分量间的随机置换，然后再对这 $m$ 个密文对做对间的随机置换，将随机置换后的密文对序列记作：$(c'_{L_1},c'_{R_1}),(c'_{L_2},c'_{R_2}),\cdots,(c'_{L_m},c'_{R_m})$. 而在实际保密计算协议 $\Pi_2$ 的执行中，Alice 的实际视图为 $(c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2,(c_{L_i},c_{R_i})_{i=1,\cdots,m},i)$，其中 $1\leqslant i\leqslant m$. 因为密文 $(c_{L_i},c_{R_i})_{i=1,\cdots,m}$ 是 Bob 经过下述方式构造的：

①由密文 $c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2$ $(1\leqslant i\leqslant m)$ 利用方案 $\mathcal{E}$ 的性质 1 和性质 2 经运算(12a)与(12b)计算得到 $m$ 个密文对 $(c_{(r_{b_i}+a_i)},c_{(r_{b_i}+b_i)})$；

②先对这 $m$ 个密文对做对内分量间的随机置换，然后再对这 $m$ 个密文对做对间的随机置换，将随机置换后的密文对序列记作：$(c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m})$.

对于 $1\leqslant i\leqslant m$，即便模拟器 $\mathcal{S}_1^{\Pi_2}$ 通过控制 Alice 获得了解密密钥，在得到 $(c'_{L_i},c'_{R_i})_{i=1,\cdots,m}$ 后，它通过解密运算后最多也只能得到由两个方程(其中每个方程各包含 3 个不同的未知数)组成的方程组，不可能通过联立方程组计算出具体的 $b_i$. 因为 $(c'_{L_i},c'_{R_i})_{i=1,\cdots,m}$ 的下标是集合定长向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$ 中的下标构成集合的一个随机置换，因此解密者无法再根据 $\frac{\mathcal{L}(c'^{\lambda}_{L_i} \bmod n^2)}{\mathcal{L}(c'^{\lambda}_{R_i} \bmod n^2)}=1$ 的下标确定交集元素. 同理对于 $1\leqslant i\leqslant m$，Alice 获得 $(c_{L_i},c_{R_i})$，$1\leqslant i\leqslant m$ 后，想通过联立方程组计算出具体的 $b_i$，根据 $\frac{\mathcal{L}(c^{\lambda}_{L_i} \bmod n^2)}{\mathcal{L}(c^{\lambda}_{R_i} \bmod n^2)}$ 的下标确定交集元素也都是不可行的. 这也就说 $\mathcal{S}_1^{\Pi_2}$ 满足安全定义关系式(1a).

对于 Alice 私有信息的安全性：假定 Bob 完全在敌手 $\mathcal{S}_2^{\Pi_2}$ 控制下，即使 Bob 不参与协议 $\Pi_2$，$\mathcal{S}_2^{\Pi_2}$ 也能够在多项时间内模拟出 $\Pi_2$ 的执行过程. 如果在该假设下，多项式时间内的敌手 $\mathcal{S}_2^{\Pi_2}$ 所得信息并不多于 Bob 在实际执行 $\Pi_2$ 中的视图内容，则称 Alice 私有信息是安全的.

首先构造一个完全控制 Bob 且能在多项式时间内模拟执行 $\Pi_2$ 的模拟器 $\mathcal{S}_2^{\Pi_2}$. 该模拟器的输入为：Alice 的私有集合 $S_a=\{a_i\}_{i=1,\cdots,k_a}$、模拟器 $\mathcal{S}_2^{\Pi_2}$ 根据 Bob 的私有集合 $S_b=\{b_i\}_{i=1,\cdots,k_b}$ 和协议输出结果 $|S_a \cap S_b|$ 构造的有利于获取 Alice 私有信息的集合 $S'_b=\{b'_1,b'_2,\ldots,b'_{k_b}\}$. 作为敌手，$\mathcal{S}_2^{\Pi_2}$ 产生的视图为 $(c_{a'_i}=(1+n)^{a'_i}r_{a'_i}^n \bmod n^2,\ c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}'_i}^n \bmod n^2,\ c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}n)r_{b\vee x_i}^n \bmod n^2,i)$，其中 $1\leqslant i\leqslant m$；而 Bob 在协议 $\Pi_2$ 的实际执行中产生的视图为 $(c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2,c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{a_i}^n \bmod n^2,c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}n)r_{b_i}^n \bmod n^2,i)$，其中 $1\leqslant i\leqslant m$. 一方面，Bob 没有加密方案 $\mathcal{E}$ 的解密密钥，并且他从 Alice 或者模拟器 $\mathcal{S}_2^{\Pi_2}$ 接收的经加密方案 $\mathcal{E}$ 作用后的密文信息；另一方面，方案 $\mathcal{E}$ 已被证明在选择明文攻击下具有语义不可区分安全性，即由加密方案 $\mathcal{E}$ 产生的密文是语义不可区分的. 因此对于 Bob 或者控制着 Bob 的敌手而言，$c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2$ 与 $c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2$ 是计算不可区分的；并且 $c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}'_i}^n \bmod n^2$ 与 $c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2$ 也是计算不可区分的. 从而可得结论：$\mathcal{S}_2$ $(c_{a'_i}=(1+(g-1)n)^{a'_i}r_{a'_i}^n \bmod n^2,c_{(r_{b_i}+a'_i)}=((c_{a'_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)\cdot r_{b\hat{x}'_i}^n \bmod n^2,c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}n)r_{b\vee x_i}^n \bmod n^2,i)$ 与真实协议执行中视图 $\mathrm{View}_{\mathbf{B}}^{\Pi}$ $(c_{a_i}=(1+(g-1)n)^{a_i}r_{a_i}^n \bmod n^2,\ c_{(r_{b_i}+a_i)}=((c_{a_i})^{k_{b_i}} \bmod n^2)\times(1+k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n \bmod n^2,\ c_{(r_{b_i}+b_i)}=(1+k_{b_i}r_{b_i}$

$n)r_{\check{b}x_i}^{n} \bmod n^2, i)$ 计算不可区分，即 $\mathcal{S}_2^{\Pi_2}$ 符合式(1b)定义的安全条件.

综上可得出：Alice 和 Bob 的私密性满足安全定义的形式化等式(1a)与(1b). 所以协议 $\Pi_2$ 满足两方保密计算 $|S_a \cap S_b|$ 的定义.

证毕.

5.2.3 针对情形 A、面向有理数的保密集合交集计算协议 $\Pi_3$

1. 具体协议

输入：Alice 和 Bob 各自的保密集合 $S_a$ 和 $S_b$、编码随机可选集合标识码 $\{d,\bar{d}\}$ ($d \in \{0,1\}$).

输出：$S_a \cap S_b$.

(1) Alice 先运行方案 $\mathcal{E}$ 的密钥生成算法产生公私钥：公钥为 $(n,1+kn)$，私钥为 $\lambda$；并按照集合的定长向量编码方法将自己的集合 $S_a$ 编码成 $m$ 长的有理向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$；

(2) 将 $m$ 长的有理向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 中的每个分量 $a_i$ 表示成有序整数对 $(\hat{a}_i,\check{a}_i)$，其中 $\hat{a}_i$ 与 $\check{a}_i$ 分别表示分子与分母；

(3) 对于 $i=1,2,\cdots,m$ 和 $((\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m))$ Alice 做如下计算：

$$\begin{cases} c_{\hat{a}_i} = (1+kn)^{\hat{a}_i} r_{\hat{a}_i}^{n} \bmod n^2 & \text{(16a)} \\ c_{\check{a}_i} = (1+kn)^{\check{a}_i} r_{\check{a}_i}^{n} \bmod n^2 & \text{(16b)} \end{cases}.$$

并向 Bob 发送 $((\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m))$；

(4) Bob 获得 $((\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m))$ 后，执行下述工作：

① 利用集合的定长编码方法 1，将自己的集合 $S_b$ 编码成 $m$ 长的有理向量 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$；

②将 $m$ 长的有理向量 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$ 中的每个分量 $b_i$ 表示成有序整数对 $(\hat{b}_i,\check{b}_i)$，其中 $\hat{a}_i$ 与 $\check{a}_i$ 分别表示分子与分母；

③ 随机选择 $4m$ 个不等的随机数 $k_{b_1},k_{b_2},\cdots,k_{b_m} \in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m} \in Z_n^*$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m} \in Z_n^*$，$r_{\check{b}x_1},r_{\check{b}x_2},\cdots,r_{\check{b}x_m} \in Z_n^*$，利用有理数加密方案 $\mathcal{E}$ 加密及其同态性通过计算：

$$\begin{cases} c_{(r_{b_i}+\check{b}_i\hat{a}_i)} = ((c_{\hat{a}_i})^{\check{b}_i k_{b_i}} \bmod n^2) \\ \quad \times (1+(g-1)k_{b_i} r_{b_i} n) r_{b\hat{x}_i}^{n} \bmod n^2 & \text{(17a)} \\ c_{(r_{b_i}+\hat{b}_i\check{a}_i)} = ((c_{\check{a}_i})^{\hat{b}_i k_{b_i}} \bmod n^2) \times \\ \quad (1+(g-1)k_{b_i} r_{b_i} n) r_{\check{b}x_i}^{n} \bmod n^2 & \text{(17b)} \end{cases}.$$

得到 $m$ 个密文对；

④ 将 $m$ 个密文对 $(c_{(r_{b_i}+\check{b}_i\hat{a}_i)}, c_{(r_{b_i}+\hat{b}_i\check{a}_i)})$ 做对内分量随机置换得到密文对序列：$((c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m}))$，并发给 Alice.

(5) Alice 收到 $((c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m}))$ 后计算：

$$\partial_i = \mathrm{P}\left(\frac{c_{L_i}^{\lambda} \bmod n^2}{c_{R_i}^{\lambda} \bmod n^2}\right), \text{ 其中 } \mathrm{P(X)} = \begin{cases} 1 & \mathrm{X}=1 \\ 0 & \mathrm{X}\neq 1 \end{cases}. \quad (18)$$

然后将 $\partial_i = 1$ 的下标 $i$ 发送给 Bob.

2. 协议 $\Pi_3$ 的正确性

因为集合 $S_a=\{x_i\}_{i=1,\cdots,k_1}$ 与 $S_b=\{x'_1,x'_2,\cdots,x'_{k_2}\}$ 对应的定长编码向量分别为 $\boldsymbol{a}=((\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m))$ 和 $\boldsymbol{b}=((\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m))$，其中

$$(\hat{a}_i,\check{a}_i) = \begin{cases} \hat{a}_i,\check{a}_i \in A_d \ ((\hat{a}_i \neq \check{a}_i) \wedge (\gcd(\hat{a}_i,\check{a}_i)=1)), \\ \qquad\qquad e_i \notin X \\ (\hat{e}_i,\check{e}_i), \\ \qquad\qquad e_i \in X \end{cases},$$

$$(\hat{b}_i,\check{b}_i) = \begin{cases} \hat{b}_i,\check{b}_i \in A_{\bar{d}} \ ((\hat{b}_i \neq \check{b}_i) \wedge (\gcd(\hat{b}_i,\check{b}_i)=1)), \\ \qquad\qquad e_i \notin X \\ (\hat{e}_i,\check{e}_i), \\ \qquad\qquad e_i \in X \end{cases},$$

$d \in \{0,1\}$，又因为 $A_d \cap A_{\bar{d}} = \varnothing$，所以如果 $(\hat{e}_i,\check{e}_i) \in \{(\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m)\} \cap \{(\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m)\}$ 则必有 $e_i \in (S_a \cap S_b)$，即如果 $e_i \in (S_a \cap S_b)$ 则集合 $S_a$ 和 $S_b$ 的定长编码向量对应的第 $i$ 个元组都为 $e_i$. 下面从元素 $e_i,(\hat{a}_i,\check{a}_i),(\hat{b}_i,\check{b}_i)$ 三者的关系论证 $\{(\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m)\} \cap \{(\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m)\}$ 计算 $S_a \cap S_b$ 的正确性：

**关系 1.** 当 $e_i \notin S_a \wedge e_i \notin S_b$ 时，因为 Alice 和 Bob 编码时，元组 $(\hat{a}_i,\check{a}_i)$ 和 $(\hat{b}_i,\check{b}_i)$ 的分量分别取自两个不同的编码随机可选集合 $\{A_d, A_{\bar{d}}\}$ ($d \in \{0,1\}$)，所以 $r_i \neq r_i'$，从而必有：$\frac{\rho\hat{a}_i\check{b}_i+\omega}{\rho\check{a}_i\hat{b}_i+\omega} \neq 1$、$\frac{\rho\check{a}_i\hat{b}_i+\omega}{\rho\hat{a}_i\check{b}_i+\omega} \neq 1$ ($\rho,\omega \in Z_n^+$)；

**关系 2.** 当 $e_i \notin S_a \wedge e_i \in S_b$ 或者 $e_i \in S_a \wedge e_i \notin S_b$

时，因为 $e_i$，$\hat{a}_i,\check{a}_i$ 与 $\hat{b}_i,\check{b}_i$ 分别取自三个不同的域，所以必有 $\hat{e}_i\neq\hat{a}_i$，$\hat{e}_i\neq\hat{b}_i$，$\check{e}_i\neq\check{a}_i$，$\check{e}_i\neq\check{b}_i$. 从而必有 $\dfrac{\rho\hat{a}_i\check{e}_i+\omega}{\rho\check{a}_i\hat{e}_i+\omega}\neq1$，$\dfrac{\rho\check{a}_i\hat{e}_i+\omega}{\rho\hat{a}_i\check{e}_i+\omega}\neq1$，$\dfrac{\rho\hat{b}_i\check{e}_i+\omega}{\rho\check{b}_i\hat{e}_i+\omega}\neq1$，$\dfrac{\rho\check{a}_i\hat{e}_i+\omega}{\rho\hat{a}_i\check{e}_i+\omega}\neq1$；

**关系 3.** 当 $e_i\in S_a\wedge e_i\in S_b$ 时，因为 $\hat{e}_i=\hat{a}_i$，$\hat{e}_i=\hat{b}_i$，$\check{e}_i=\check{a}_i$，$\check{e}_i=\check{b}_i$，所以有 $\dfrac{\rho\hat{a}_i\check{b}_i+\omega}{\rho\check{a}_i\hat{b}_i+\omega}=\dfrac{\rho\check{a}_i\hat{b}_i+\omega}{\rho\hat{a}_i\check{b}_i+\omega}=1$.

综上所述，协议 $\Pi_3$ 利用集合的定长向量编码方法，通过计算

$$\{(\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m)\}\cap\{(\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m)\}$$

能够精确地计算出集合 $S_a=\{x_i\}_{i=1,\cdots k_1}$ 与 $S_b=\{x'_1,x'_2,\cdots,x'_{k_2}\}$ 的交集 $S_a\cap S_b$. 从而在协议 $\Pi_3$ 中，如果

$$\partial_i=\frac{\mathcal{L}(c_{L_i}^{\lambda}\bmod n^2)}{\mathcal{L}(c_{R_i}^{\lambda}\bmod n^2)}=1. \tag{19}$$

时，则集合 $\mathrm{E}=\{e_i\}_{i=1,\cdots m}$ 中对应的第 $i$ 元素 $e_i$ 必为交集 $S_a\cap S_b$ 的元素.

3. 协议 $\Pi_3$ 的安全性

**定理 5.** Alice 和 Bob 利用协议 $\Pi_3$ 可以安全地实现保密计算 $S_a\cap S_b$，其中 $S_a$ 与 $S_b$ 中的元素都是有理数.

证明. 关于 $\Pi_3$ 的安全性需要考量如下两种情况：

(1) Alice 完全受控于敌手时 Bob 私有信息的安全性，如果多项式时间内的敌手 $\mathcal{S}_1^{\Pi_3}$ 在完全控制 Alice 情形下，它在代替 Alice 模拟执行协议 $\Pi_3$ 过程中所能获取的信息并不多于 Alice 在实际执行协议中的视图内容，则称协议 $\Pi_3$ 能够保证 Bob 私有信息的安全性；

(2) Bob 完全受控于敌手时 Alice 私有信息的安全性，如果多项式时间内的敌手 $\mathcal{S}_1^{\Pi_3}$ 在完全控制 Bob 情形下，它在代替 Bob 模拟执行协议 $\Pi_3$ 过程中所能获取的信息并不多于 Bob 在实际执行协议中的视图内容，则称协议 $\Pi_3$ 能够保证 Alice 私有信息的安全性.

构造模拟 Alice 视图的模拟器 $\mathcal{S}_1^{\Pi_3}$：假设 $\mathcal{S}_1^{\Pi_3}$ 能够在多项式时间内模拟协议 $\Pi_3$ 的执行过程，并令它的输入为：根据 Alice 的私有输入 $S_a=\{a_i\}_{i=1,\cdots k_a}$ 和 $S_a\cap S_b$ 构造的利于攻击 Bob 的集合 $S'_a=\{a'_1,a'_2,\ldots,a'_{k_a}\}$、Bob 选择的 $4m$ 个随机数 $k_{b_1}$，$k_{b_2},\cdots,k_{b_m}\in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m}\in Z_n^*$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m}\in Z_n^*$，$r_{b\check{x}_1},r_{b\check{x}_2},\cdots,r_{b\check{x}_m}\in Z_n^*$，以及 Bob 的私有集合 $S_b=\{b_i\}_{i=1,\cdots k_b}$. 作为敌手，$\mathcal{S}_1^{\Pi_3}$ 模拟产生的视图为 $(\{(c'_{\hat{a}_i},c'_{\check{a}_i})\}_{i=1,2,\cdots,m},\{(c'_{L_i},c'_{R_i})\}_{i=1,2,\cdots,m},i)$；而协议 $\Pi_3$ 的实际执行中，Alice 的实际视图为 $(\{(c_{\hat{a}_i},c_{\check{a}_i})\}_{i=1,2,\cdots,m},\{(c_{L_i},c_{R_i})\}_{i=1,2,\cdots,m},i)$. 因为密文 $(c'_{L_i},c'_{R_i})_{i=1,\cdots m}$ $(1\leqslant i\leqslant m)$ 是 Bob 经过下述方式构造的：

(1) 由密文对 $(c'_{\hat{a}_i},c'_{\check{a}_i})$（其中 $c_{\hat{a}_i}=(1+kn)^{\hat{a}_i}r_{\hat{a}_i}^n\bmod n^2$，$c_{\check{a}_i}=(1+kn)^{\check{a}_i}r_{\check{a}_i}^n\bmod n^2$）利用方案 $\mathcal{E}$ 的性质 1 和性质 2 经运算：

$$\begin{cases}c_{(r_{b_i}+\check{b}_i\hat{a}_i)}=\\((c_{\hat{a}_i})^{\check{b}_i k_{b_i}}\bmod n^2)\times(1+(g-1)k_{b_i}r_{b_i}n)r_{b\hat{x}_i}^n\bmod n^2\\c_{(r_{b_i}+\hat{b}_i\check{a}_i)}=\\((c_{\check{a}_i})^{\hat{b}_i k_{b_i}}\bmod n^2)\times(1+(g-1)k_{b_i}r_{b_i}n)r_{b\check{x}_i}^n\bmod n^2\end{cases}.$$

计算得到 $m$ 个密文对 $(c'_{(r_{b_i}+a_i)},c'_{(r_{b_i}+b_i)})$；

(2) 对 $m$ 个密文对 $(c'_{L_i},c'_{R_i})_{i=1,\cdots m}$ 分别做对内分量随机置换得到密文对序列 $\{(c'_{L_i},c'_{R_i})\}_{i=1,2,\cdots,m}$. 对于 $1\leqslant i\leqslant m$，Alice 获得 $(c'_{L_i},c'_{R_i})_{i=1,\cdots m}$ 后通过解密运算后最多只能得到由两个方程(其中每个方程各包含 3 个不同的未知数)组成的方程组，不可能通过联立方程组计算出具体的 $(\hat{b}_i,\check{b}_i)$. 同理在实际协议 $\Pi_3$ 中，对于 $i=1,2,\cdots,m$，Alice 获得 $(c_{L_i},c_{R_i})$ 后也不可能通过联立方程组计算出具体的 $(\hat{b}_i,\check{b}_i)$. 这也就说敌手在模拟协议和 Alice 在实际协议中一样无法通过额外的计算推算出比协议 $\Pi_3$ 的实际输出 $S_a\cap S_b$ 更多的信息. 因此，模拟器 $\mathcal{S}_1^{\Pi_3}$ 满足定义关系式(1a).

构造模拟 Bob 视图的模拟器 $\mathcal{S}_2^{\Pi_3}$：假定敌手 $\mathcal{S}_2^{\Pi_3}$ 完全控制着 Bob 且能在多项式时间内模拟执行协议 $\Pi_3$. 其中该模拟器的输入为：Alice 的私有集合 $S_a=\{a_i\}_{i=1,\cdots,k_a}$、模拟器 $\mathcal{S}_2^{\Pi_3}$ 根据 Bob 的私有集合 $S_b=\{b_i\}_{i=1,\cdots,k_a}$ 和 $S_a\cap S_b$ 构造的有利于获取 Alice

私有信息的集合 $S'_b=\{b'_1,b'_2,\ldots,b'_{k_b}\}$. 作为敌手，模拟器 $\mathcal{S}_2^{\Pi_3}$ 产生的视图为

$$(\{(c'_{\overset{\wedge}{a_i}},c'_{\overset{\vee}{a_i}})\}_{i=1,2,\cdots,m},\{(c'_{L_i},c'_{R_i})\}_{i=1,2,\cdots,m},i),$$

对于 $i=1,2,\cdots,m$，$(c'_{\overset{\wedge}{a_i}},c'_{\overset{\vee}{a_i}})$ 是 Alice 在模拟协议中用自己的公钥对有理数 $a_i$ 对应的整型表示元组的重新加密：

$$\begin{cases}c_{\overset{\wedge}{a_i}}=(1+kn)^{\hat{a}_i}r'^{n}_{\hat{a}_i}\bmod n^2\\ c_{\overset{\vee}{a_i}}=(1+kn)^{\overset{\vee}{a_i}}r'^{n}_{\overset{\vee}{a_i}}\bmod n^2\end{cases}.$$

生成的. 而 Bob 在实际执行协议 $\Pi_3$ 的过程中产生的视图为

$$(\{(c_{\overset{\wedge}{a_i}},c_{\overset{\vee}{a_i}})\}_{i=1,2,\cdots,m},\{(c_{L_i},c_{R_i})\}_{i=1,2,\cdots,m},i),$$

对于 $i=1,2,\cdots,m$，$(c_{\overset{\wedge}{a_i}},c_{\overset{\vee}{a_i}})$ 是 Alice 在实际协议 $\Pi_3$ 中用自己公钥对自己集合元素 $a_i$ 对应的整型表示元组的加密：

$$\begin{cases}c_{\overset{\wedge}{a_i}}=(1+kn)^{\hat{a}_i}r^{n}_{\hat{a}_i}\bmod n^2\\ c_{\overset{\vee}{a_i}}=(1+kn)^{\overset{\vee}{a_i}}r^{n}_{\overset{\vee}{a_i}}\bmod n^2\end{cases}.$$

无论在模拟协议还是实际协议中 Bob 接收到的信息都是另一个参与者(Alice 或 $\mathcal{S}_2^{\Pi_3}$)私有信息在 $\mathcal{E}$ 作用下的密文，一方面因为 Bob 没有 $\mathcal{E}$ 的解密密钥；另一方面又因方案 $\mathcal{E}$ 已被证明在选择明文攻击下具有语义不可区分安全，即由加密方案 $\mathcal{E}$ 产生的密文是语义不可区分的，也就说 $(c'_{\overset{\wedge}{a_i}},c'_{\overset{\vee}{a_i}})$ 与 $(c_{\overset{\wedge}{a_i}},c_{\overset{\vee}{a_i}})$ 是计算不可区分的并且 $(c'_{L_i},c'_{R_i})_{i=1,\cdots m}$ 与 $(c_{L_i},c_{R_i})_{i=1,\cdots,m}$ 也是计算不可区分的. 从而可得：模拟视图 $(\{(c'_{\overset{\wedge}{a_i}},c'_{\overset{\vee}{a_i}})\}_{i=1,2,\cdots,m},\{(c'_{L_i},c'_{R_i})\}_{i=1,2,\cdots,m},i)$ 与真实视图 $(\{(c_{\overset{\wedge}{a_i}},c_{\overset{\vee}{a_i}})\}_{i=1,2,\cdots,m},\{(c_{L_i},c_{R_i})\}_{i=1,2,\cdots,m},i)$ 是计算不可区分的. 因此模拟器 $\mathcal{S}_2^{\Pi_3}$ 满足安全定义关系式(1b).

综上得出：Alice 和 Bob 在协议执行中无信息泄漏，即协议 $\Pi_3$ 满足两方保密计算定义条件(1a)与(1b). 所以任意两方都可以利用协议 $\Pi_3$ 安全地实现两方集合交集的保密计算.

证毕.

5.2.4 针对问题 B、面向有理数的 PSI 测试和交集势的计算协议 $\Pi_4$

1. 具体协议

输入：Alice 和 Bob 各自的保密集合 $S_a$ 和 $S_b$、编码随机可选集合标识码 $\{d,\bar{d}\}$ ($d\in\{0,1\}$).

输出：如果 $S_a$ 与 $S_b$ 无公共元素，输出 0；如果 $S_a$ 与 $S_b$ 有公共元素，输出 $|S_a\cap S_b|$.

(1) 准备阶段. Alice 和 Bob 按照如下方式进行准备工作：

① Alice 先运行方案 $\mathcal{E}$ 的密钥生成算法产生公私钥：公布公钥 $(n,1+n)$，保留私钥 $\lambda$；

②Alice 和 Bob 按照集合的定长向量编码方法，将各自的集合 $S_a$ 和 $S_b$ 分别编码成 $m$ 长的有理向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$；

③ Alice 和 Bob 分别将各自的有理向量 $\boldsymbol{a}=(a_1,a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$ 表示成分量为整型有序对的向量 $((\overset{\wedge}{a_1},\overset{\vee}{a_1}),(\overset{\wedge}{a_2},\overset{\vee}{a_2}),\cdots,(\overset{\wedge}{a_m},\overset{\vee}{a_m}))$ 和 $((\overset{\wedge}{b_1},\overset{\vee}{b_1}),(\overset{\wedge}{b_2},\overset{\vee}{b_2}),\cdots,(\overset{\wedge}{b_m},\overset{\vee}{b_m}))$

Alice 先运行方案 $\mathcal{E}$ 的密钥生成算法产生公私钥：公布公钥 $(n,1+n)$，保留私钥 $\lambda$；Alice 和 Bob 按照集合的定长向量编码方法，将各自的集合 $S_a$ 和 $S_b$ 分别编码成 $m$ 长的有理向量 $\boldsymbol{a}=(a_1,\ a_2,\cdots,a_m)$ 和 $\boldsymbol{b}=(b_1,b_2,\cdots,b_m)$；

(2) 保密集合交集测试和交集势的计算：

① Alice 采用方案 $\mathcal{E}$ 对向量 $\boldsymbol{a}=((\overset{\wedge}{a_1},\overset{\vee}{a_1}),(\overset{\wedge}{a_2},\overset{\vee}{a_2}),\cdots,(\overset{\wedge}{a_m},\overset{\vee}{a_m}))$ 的各分量 $(\overset{\wedge}{a_i},\overset{\vee}{a_i})$ $(i=1,2,\cdots,m)$ 进行加密得到向量 $\boldsymbol{a}$ 对应的密文向量 $((c_{\overset{\wedge}{a_1}},c_{\overset{\vee}{a_1}}),(c_{\overset{\wedge}{a_2}},c_{\overset{\vee}{a_2}}),\cdots,(c_{\overset{\wedge}{a_m}},c_{\overset{\vee}{a_m}}))$，并将它发送给 Bob；

② 收到 $((c_{\overset{\wedge}{a_1}},c_{\overset{\vee}{a_1}}),(c_{\overset{\wedge}{a_2}},c_{\overset{\vee}{a_2}}),\cdots,(c_{\overset{\wedge}{a_m}},c_{\overset{\vee}{a_m}}))$ 后，Bob 随机选择 $4m$ 个不等的随机数 $k_{b_1},k_{b_2},\cdots,k_{b_m}\in Z_n$，$r_{b_1},r_{b_2},\cdots,r_{b_m}\in Z_n^*$，$r_{b\hat{x}_1},r_{b\hat{x}_2},\cdots,r_{b\hat{x}_m}\in Z_n^*$，$r_{b\overset{\vee}{x}_1},r_{b\overset{\vee}{x}_2},\cdots,r_{b\overset{\vee}{x}_m}\in Z_n^*$，按照式(17a)、(17b)计算出 $m$ 个密文对 $(c_{(r_{b_i}+b_i\overset{\vee}{a_i})},c_{(r_{b_i}+b_i\overset{\wedge}{a_i})})(1\leqslant i\leqslant m)$ 后，先将这 $m$ 个密文对做对内分量间的随机置换，然后再对这 $m$ 个密文对做对间的随机置换，将随机置换后的密文对序列记作：$(c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m})$，并把它发给 Alice.

③ Alice 收到 $(c_{L_1},c_{R_1}),(c_{L_2},c_{R_2}),\cdots,(c_{L_m},c_{R_m})$ 后计算：

$$\theta=\sum_{i=1}^{m}P\left(\frac{\mathcal{L}(c_{L_i}^{\lambda}\bmod n^2)}{\mathcal{L}(c_{R_i}^{\lambda}\bmod n^2)}\right),\ \text{其中}P(y)=\begin{cases}1,&y=1\\0,&y\neq1\end{cases}\tag{20}$$

并将 $\theta$ 发送给 Bob.

2. 协议 $\Pi_4$ 的正确性

因为集合 $S_a=\{x_i\}_{i=1,\cdots,k_1}$ 与 $S_b=\{x'_1,x'_2,\cdots,x'_{k_2}\}$

对应的定长编码向量分别为 $\boldsymbol{a}=((\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m))$ 和 $\boldsymbol{b}=((\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m))$，其中

$$(\hat{a}_i,\check{a}_i)=\begin{cases}\hat{a}_i,\check{a}_i\in A_d\ ((\hat{a}_i\neq\check{a}_i)\wedge(\gcd(\hat{a}_i,\check{a}_i)=1)), & e_i\notin X\\ (\hat{e}_i,\check{e}_i), & e_i\in X\end{cases},$$

$$(\hat{b}_i,\check{b}_i)=\begin{cases}\hat{b}_i,\check{b}_i\in A_{\bar{d}}\ ((\hat{b}_i\neq\check{b}_i)\wedge(\gcd(\hat{b}_i,\check{b}_i)=1)), & e_i\notin X\\ (\hat{e}_i,\check{e}_i), & e_i\in X\end{cases},$$

$d\in\{0,1\}$，又因为 $A_d\cap A_{\bar{d}}=\varnothing$，所以如果 $(\hat{e}_i,\check{e}_i)\in\{(\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m)\}\cap\{(\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m)\}$则 $e_i$ 必为 $S_a$ 和 $S_b$ 的公共元素，即如果 $e_i$ 为 $S_a$ 和 $S_b$ 的公共元素，则集合 $S_a$ 和 $S_b$ 的定长编码向量对应的第 $i$ 个元组都为 $e_i$．下面从元素 $e_i$,$(\hat{a}_i,\check{a}_i)$,$(\hat{b}_i,\check{b}_i)$ 三者的关系论证

$$|\{(\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m)\}\cap\{(\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m)\}|=|S_a\cap S_b|$$

的正确性：

**关系 4.** 当 $e_i\notin S_a\wedge e_i\notin S_b$ 时，因为 Alice 和 Bob 编码时，元组 $(\hat{a}_i,\check{a}_i)$ 和 $(\hat{b}_i,\check{b}_i)$ 的分量分别取自两个不同的编码随机可选集合 $\{A_d,A_{\bar{d}}\}$ $(d\in\{0,1\})$，所以 $r_i\neq r_i'$，从而必有 $\dfrac{\rho\hat{a}_i\check{b}_i+\omega}{\rho\check{a}_i\hat{b}_i+\omega}\neq1$，$\dfrac{\rho\check{a}_i\hat{b}_i+\omega}{\rho\hat{a}_i\check{b}_i+\omega}\neq1$ $(\rho,\omega\in Z_n^+)$；

**关系 5.** 当 $e_i\notin S_a\wedge e_i\in S_b$ 或者 $e_i\in S_a\wedge e_i\notin S_b$ 时，因为 $e_i$，$\hat{a}_i,\check{a}_i$ 与 $\hat{b}_i,\check{b}_i$ 分别取自三个不同的域，所以必有 $\hat{e}_i\neq\hat{a}_i$，$\hat{e}_i\neq\hat{b}_i$，$\check{e}_i\neq\check{a}_i$，$\check{e}_i\neq\check{b}_i$．从而必有 $\dfrac{\rho\hat{a}_i\check{e}_i+\omega}{\rho\check{a}_i\hat{e}_i+\omega}\neq1$，$\dfrac{\rho\check{a}_i\hat{e}_i+\omega}{\rho\hat{a}_i\check{e}_i+\omega}\neq1$，$\dfrac{\rho\hat{b}_i\check{e}_i+\omega}{\rho\check{b}_i\hat{e}_i+\omega}\neq1$，$\dfrac{\rho\check{a}_i\hat{e}_i+\omega}{\rho\hat{a}_i\check{e}_i+\omega}\neq1$；

**关系 6.** 当 $e_i\in S_a\wedge e_i\in S_b$ 时，因为 $\hat{e}_i=\hat{a}_i$，$\hat{e}_i=\hat{b}_i$，$\check{e}_i=\check{a}_i$，$\check{e}_i=\check{b}_i$ 所以有 $\dfrac{\rho\hat{a}_i\check{b}_i+\omega}{\rho\check{a}_i\hat{b}_i+\omega}=\dfrac{\rho\check{a}_i\hat{b}_i+\omega}{\rho\hat{a}_i\check{b}_i+\omega}=1$．

综上所述，如果 $\dfrac{\rho\hat{a}_i\check{b}_i+\omega}{\rho\check{a}_i\hat{b}_i+\omega}=1$ 或 $\dfrac{\rho\check{a}_i\hat{b}_i+\omega}{\rho\hat{a}_i\check{b}_i+\omega}=1$，即 $\dfrac{c_{L_i}^{\lambda}\bmod n^2}{c_{R_i}^{\lambda}\bmod n^2}=1$，则 $e_i=a_i=\dfrac{\hat{a}_i}{\check{a}_i}=b_i=\dfrac{\hat{b}_i}{\check{b}_i}$ 必在交集 $\{a_1,a_2,\cdots,a_m\}\cap\{b_1,b_2,\cdots,b_m\}$ 中，其中 $i$ 用于指示集合 $\{1,2,\cdots,m\}$ 的一个随机置换集合的第 $i$ 个元素，因此通过统计计算结果为 1 的个数：

$$\theta=\sum_{i=1}^{m}P\left(\frac{c_{L_i}^{\lambda}\bmod n^2}{c_{R_i}^{\lambda}\bmod n^2}\right),\text{其中}P(y)=\begin{cases}1, & y=1\\ 0, & y\neq1\end{cases}.$$

能够精确地计算出集合 $\{(\hat{a}_1,\check{a}_1),(\hat{a}_2,\check{a}_2),\cdots,(\hat{a}_m,\check{a}_m)\}\cap\{(\hat{b}_1,\check{b}_1),(\hat{b}_2,\check{b}_2),\cdots,(\hat{b}_m,\check{b}_m)\}$ 的势，即 $|S_a\cap S_b|$．

3. 协议 $\Pi_4$ 的安全性

**定理 6**. Alice 和 Bob 利用协议 $\Pi_4$ 可以安全地实现保密计算 $|S_a\cap S_b|$，其中 $S_a$ 和 $S_b$ 中的元素都是有理数.

证明．协议 $\Pi_4$ 和 $\Pi_2$ 在安全性证明方面与保密计算 $|S_a\cap S_b|$ 的机理完全一样，只是集合构成元素的数据类型不同(前者针对集合元素是有理数的集合，后者针对集合元素是整数的集合)，因此该定理的证明与定理 3 的证明完全一样．为了节省篇幅在此不再赘述.

证毕.

## 6　性能分析

计算复杂度方面(为了公平地比较，在此只对模数都采用 $n^2$ 的协议进行比较分析)，协议 $\Pi_1$、$\Pi_2$、$\Pi_3$、$\Pi_4$ 和基于多项式验根法构造的协议[9,10]属于同一级别：在协议 $\Pi_1$ 和 $\Pi_2$ 中，Alice 和 Bob 都需要运行 $2m$ 次加密、$m$ 次解密、$m$ 次同态运算；在协议 $\Pi_3$ 和 $\Pi_4$ 中，Alice 和 Bob 都需要运行 $2m$ 次加密、$m$ 次解密、$2m$ 次同态运算；而 2016 年，Freedman 等基于 Paillier 加密方案设计的保密集合交协议，该协议运行一次总计需要执行 $k_a+k_b$ 次加密、$k_b$ 次解密、$k_b\sum_{i=1}^{\left\lfloor\frac{k_a}{B}\right\rfloor}$ 次同态运算．如果把衡量算法复杂度的基础单位定义成两个量的一次自模乘运算 $((x\bmod n^2\cdot y\bmod n^2)\bmod n^2$，$x,\ y<n^2)$所消耗的

计算资源，记作($O(\log^2 n)$)，则协议$\Pi_1$的计算总计需要花费$2ml+m\lambda l+ml$次自模乘运算；而基于多项式验根法构造的协议[1,8]总计需要$(k_a+k_b)l+\lambda k_b l+(k_b\sum_{i=1}^{\lfloor \frac{k_a}{B} \rfloor})l$次自模乘运算. 因为$m<\frac{n}{3}$，$1\leqslant k_a$，$k_b\leqslant n$，所以上述5个协议的计算复杂度都属于$O(n^2\lg n)$级别的.

在集合交集计算差错方面，协议$\Pi_1$、$\Pi_2$、$\Pi_3$、$\Pi_4$对于集合交集元素的计算差错为零，而基于多项式验根法构造的协议[1,8]、基于多项式间提取公因式法构造的协议[5,33]、采用数字签名技术构造带元素认证的协议[6]对于集合交元素的计算存在差错并且无法界定差错大小.

在保护隐私的范畴方面，协议$\Pi_1$、$\Pi_2$、$\Pi_3$、$\Pi_4$不仅保护了协议参与双方集合中的数据元素还保护了参与双方私有集合的势，而协议[1,5,8]中作为服务器一方私有集合的势是公开的信息；基于不经意伪随机函数计算的保密集合交集计算协议[7]虽然不会泄露双方集合中的私密数据，但会泄露拥有PRF密钥$\mathcal{K}_{prf}$一方集合的势.

表1是协议$\Pi_1$、$\Pi_2$、$\Pi_3$、$\Pi_4$与协议[1,8]在协议计算复杂度(用自模乘运算最大次数作为比较标准)、集合交集计算是否存在差错以及作为服务器一方私有集合的势是否保密三个方面的比较.

表2是协议$\Pi_1$、$\Pi_2$、$\Pi_3$、$\Pi_4$与协议[1,5,33]在保密集合交集(势)计算是否存在差错方面以及隐私保护范畴方面，除了保护双方私有集合元素外，还保护双方私有集合的势两个方面的比较.

**表1 模数都采用$n^2$的保密计算协议间的对比分析**

| 类型 | 计算复杂度 | 交集(或交集势)的计算是否无差错 | 服务器一方集合的势是否保密 |
|---|---|---|---|
| 协议[9,10] | $O(n^2\lg n)$ | × | × |
| 协议$\Pi_1$ | $O(n^2\lg n)$ | √ | √ |
| 协议$\Pi_2$ | $O(n^2\lg n)$ | √ | √ |
| 协议$\Pi_3$ | $O(n^2\lg n)$ | √ | √ |
| 协议$\Pi_4$ | $O(n^2\lg n)$ | √ | √ |

其中，√表示具有某种性能，×表示不具有某种性能

**表2 模数不等的保密计算协议间的对比分析**

| 类型 | 交集(势)的计算是否无差错 | 隐私保护范畴 |
|---|---|---|
| 协议[1,5,33] | × | √ |
| 协议$\Pi_1$ | √ | √ |
| 协议$\Pi_2$ | √ | √ |
| 协议$\Pi_3$ | √ | √ |
| 协议$\Pi_4$ | √ | √ |

其中，√表示具有某种性能，×表示只保护保护双方私有集合元素，不能保护双方私有集合的势

## 7 结束语

本文首先提出了一个面向有理数的加密方案，并在标准模型下证明了它具有语义安全性. 然后利用该方案和集合的定长向量编码技术设计了两个保密集合交集计算协议和两个保密集合交集势计算协议，协议$\Pi_1$和$\Pi_3$对于相交元素的保密计算不存在差错；协议$\Pi_2$和$\Pi_4$对于相交元素个数的保密计算不存在差错；此外，协议$\Pi_1$、$\Pi_2$、$\Pi_3$、$\Pi_4$在保护隐私的范畴方面，不仅保护了双方的私有集合元素，还保护了双方私有集合的势. 这丰富了保密集合交集计算的研究方法，同时实现无匹配差错且保密范畴更广的(既不会泄露双方集合中的私密数据，也不会泄露双方私密数据的数量)保密集合交集计算.

## 参 考 文 献


[1] Freedman M J, Nissim K, Pinkas B. Efficient private matching and set intersection//Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques. Interlaken, Switzerland, 2004: 1-19

[2] Aggarwal C C, Philip S Y. A general survey of privacy-preserving data mining models and algorithms. Privacy-preserving data mining. Berlin Germany: Springer, 2008: 11-52

[3] Vatsalan D, Sehili Z, Christen P, et al. Privacy-preserving record linkage for big data: Current approaches and research challenges. Handbook of Big Data Technologies. Berlin Germany: Springer, 2017: 851-895

[4] Egert R, Fischlin M, Gens D, et al. Privately computing set-union and set-intersection cardinality via bloom filters//Proceedings of the Australasian Conference on Information Security and Privacy. Brisbane, Australia, 2015: 413-430.

[5] Kissner L, Song D. Privacy-preserving set operations//Proceedings of the annual international cryptology conference. Santa Barbara, USA, 2005: 241-257

[6] De Cristofaro E, Tsudik G. Practical private set intersection protocols with linear complexity//Proceedings of the International Conference on Financial Cryptography and Data Security. Springer, Tenerife, Canary Islands, Spain, 2010: 143-159

[7] Michael J. Freedman, Yuval Ishai, Benny Pinkas, and Omer Reingold. Keyword search and oblivious pseudorandom functions//Proceedings of the Theory of Cryptography Conference. Cambridge, USA, 2005: 303-324

[8] Freedman M J, Hazay C, Nissim K, et al. Efficient set intersection with simulation-based security. Journal of Cryptology, 2016, 29(1): 115-155

[9] Hazay C, Mikkelsen G L, Rabin T, et al. Efficient RSA Key Generation and Threshold Paillier in the Two-Party Setting. Journal of Cryptology, 2019, 32(2): 265-323

[10] Dong C, Chen L, Camenisch J, et al. Fair private set intersection with a semi-trusted arbiter//Proceedings of the IFIP Annual Conference on Data and Applications Security and Privacy. Newark, USA, 2013: 128-144

[11] Bradley T, Faber S, Tsudik G. Bounded size-hiding private set intersection//Proceedings of the International Conference on Security and Cryptography for Networks. Amalfi, Italy, 2016: 449-467.

[12] Dachman-Soled D, Malkin T, Raykova M, et al. Efficient robust private set intersection. International Journal of Applied Cryptography, 2012, 2(4): 289-303

[13] Hazay C, Nissim K. Efficient set operations in the presence of malicious adversaries. Journal of Cryptology, 2012, 25(3): 383-433

[14] Carmit Hazay and Yehuda Lindell. Efficient oblivious polynomial evaluation with simulation-based security. IACR Cryptology ePrint Archive, 2009:459, 2009. https://eprint. iacr.org/2009/459.pdf

[15] Carmit Hazay and Yehuda Lindell. Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries. Journal of Cryptology, 2010, 23(3): 422-456

[16] Kolesnikov V, Kumaresan R, Rosulek M, et al. Efficient batched oblivious PRF with applications to private set intersection// Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. Vienna, Austria, 2016: 818-829

[17] Kerschbaum F. Outsourced private set intersection using homomorphic encryption//Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security. Seoul Korea, 2012: 85-86

[18] Abadi A, Terzis S, Dong C. VD-PSI: Verifiable delegated private set intersection on outsourced private datasets//Proceedings of the International Conference on Financial Cryptography and Data Security. Christ Church, Barbados, 2016: 149-168

[19] Kim M, Lee H T, Cheon J H. Mutual private set intersection with linear complexity//Proceedings of the International Workshop on Information Security Applications. Jeju Island, Korea, 2011: 219-231

[20] De Cristofaro E, Kim J, Tsudik G. Linear-complexity private set intersection protocols secure in malicious model//Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security. Singapore, 2010: 213-231

[21] Hemenway Falk B, Noble D, Ostrovsky R. Private set intersection with linear communication from general assumptions//Proceedings of the 18th ACM Workshop on Privacy in the Electronic Society. London, UK, 2019: 14-25

[22] Rindal P, Rosulek M. Improved private set intersection against malicious adversaries//Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques. Paris, France, 2017: 235-259

[23] Kolesnikov V, Matania N, Pinkas B, et al. Practical multi-party private set intersection from symmetric-key techniques//Proceedings of the ACM Conference on Computer and Communications Security. London, UK, 2017: 1257-1272

[24] Kolesnikov V, Rosulek M, Trieu N, et al. Scalable private set union from symmetric-key techniques//Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security. Brisbane, Australia, 2019: 636-666

[25] Pinkas B, Schneider T, Segev G, et al. Phasing: Private set intersection using permutation-based hashing//Proceedings of the 24th Conference on USENIX Security Symposium. Washington, USA, 2015: 515-530

[26] Pinkas B, Schneider T, Tkachenko O, et al. Efficient circuit-based psi with linear communication//Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques. Zagreb, Croatia, 2019: 122-153

[27] Pinkas B, Schneider T, Zohner M. Faster private set intersection based on OT extension//Proceedings of the 23th Conference on USENIX Security Symposium. San Diego, USA, 2014: 797-812

[28] Pinkas B, Schneider T, Zohner M. Scalable private set intersection based on OT extension. ACM Transactions on Privacy and Security (TOPS), 2018, 21(2): 7

[29] Orrù M, Orsini E, Scholl P. Actively secure 1-out-of-N OT extension with application to private set intersection//Proceedings of the Cryptographers' Track at the RSA Conference. San Francisco, USA, 2017: 381-396

[30] Pinkas, Benny & Rosulek, Mike & Trieu, Ni & Yanai, Avishay. SpOT-Light: Lightweight private set intersection from sparse OT//Proceedings of the CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, USA, 2019: 401-431

[31] Ghosh S, Nilges T. An algebraic approach to maliciously secure private set intersection//Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques. Zagreb, Croatia, 2019: 154-185

[32] D. Many, M. Burkhart, and X. Dimitropoulos. Fast private set operations with sepia. Zurich, Switzerland: Communication Systems Group TIK-Report No. 345, Mar 2012

[33] Zhou Su-Fang, Li Shun-Dong, Guo Yi-Min, Dou Jia-Wei, Chen Zhen-Hua. Efficient secure set intersection problem computation. Chinese Journal of Computers, 2018, 41(2): 464-480(in Chinese) (周素芳, 李顺东, 郭奕旻, 窦家维,王道顺. 保密集合相交问题的高效计算. 计算机学报, 2018, 41(2): 464-480)

[34] Boneh D, Gentry C, Halevi S, et al. Private database queries using somewhat homomorphic encryption//Proceedings of the International Conference on Applied Cryptography and Network Security. Banff, Canada, 2013: 102-118

[35] Carmit Hazay and Yehuda Lindell. Efficient Secure Two-Party Protocols-Techniques and Constructions. Berlin, Germany: Springer-Verlag, 2010

[36] Changyu Dong, Liqun Chen, and Zikai Wen. When private set intersection meets big data: An efcient and scalable protocol// Proceedings of the 2013 ACM SIGSAC conference on computer & communications security. Berlin, Germany, 2013: 789-800

[37] De Cristofaro E, Gasti P, Tsudik G. Fast and private computation of cardinality of set intersection and union//Proceedings of the International Conference on Cryptology and Network Security. Darmstadt, Germany, 2012: 218-231

[38] Ateniese G, De Cristofaro E, Tsudik G. (If) size matters: size-hiding private set intersection//Proceedings of the Interna-

tional Workshop on Public Key Cryptography. Taormina, Italy, 2011: 156-173

[39] Kiss Á, Liu J, Schneider T, et al. Private set intersection for unequal set sizes with mobile applications//Proceedings of the Privacy Enhancing Technologies. Minneapolis, USA, 2017: 177-197

[40] Katz J, Lindell Y. Introduction to modern cryptography. Second Edition. Boca Raton: CRC Press, 2014

[41] Gong Lin-Ming, Li Shun-Dong, Dou Jia-Wei, Guo Yi-Min, Wang Dao-shun. Homomorphic encryption scheme and a protocol on secure computing a line by two private point. Journal of Software, 2017, 28(12) :3274-3292(in Chinese)
(巩林明, 李顺东, 窦家维, 郭奕旻, 王道顺. 同态加密方案及安全两点直线计算协议. 软件学报, 2017, 28(12) :3274-3292)

[42] Goldreich O. Foundations of cryptography: volume 2, basic applications. Cambridge, UK: Cambridge University press, 2009

[43] Cramer R, Damgaård I B. Secure multiparty computation. Cambridge, UK: Cambridge University Press, 2015

[44] Paillier P. Public-key cryptosystems based on composite degree residuosity classes//Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques. Prague, Czech Republic, 1999: 223-238

**GONG Lin-Ming**, Ph.D., lecturer. His current research interests include information security and cryptography.

**WANG Dao-Shun**, Ph.D., associate professor. His current research interests include multimedia security and forensics, cryptographic algorithms and video intelligent behavior analysis.

**LIU Mo-Meng**, Ph.D., lecturer. Her research interests include information security and cryptography.

**GAO Quan-Li**, Ph.D., lecturer. His current research interests include network and information security.

**SHAO Lian-He**, Ph.D., associate professor. His research interests include information security and quantum intelligence information calculation.

**WANG Ming-Ming**, Ph.D., associate professor. His research interests include information security and cryptography.

## Background

In distributed secure multi-party computation setting, private set intersection is an important tool for protecting users' privacy in data matching, data mining, recommendation, and so on. Up to now, the existing protocols for evaluating private set intersection can be divided into four categories according to the construction methods (polynomial root testing, common factor extracting between functions, oblivious function calculating, and set elements authenticating via digital signature): (1) protocols constructed by polynomial root-checking method, (2) protocols constructed by "extracting Common Factor between multiple functions", (3) protocols constructed with the idea of "inadvertent pseudorandom function calculation", and (4) protocols constructed with the idea of "authentication of set elements with digital signature technology". From the view of computable point, PSI protocols constructed using "polynomial root testing", "common factor extracting among multiple functions", "set elements authenticating via digital signature technology" and "inadvertent pseudorandom function (pseudo-andom function, PRF) calculating", such as protocols[1,5-7] have opened up four methods to solve the problem of PSI in distributed environment. However, for the PSI protocols designed by the first three methods ("polynomial root testing", "common factor extracting among multiple functions", "set elements authenticating via digital signature technology"), there may be errors in the calculation results after the execution, and the scope of the errors is difficult to define. The PSI protocol based on oblivious pseudorandom function calculation will not disclose the private data in the two sets, but they will reveal the size of the set of the participant that has PRF keys.

In this paper, we aim to solve the problem that constructs precisely private set intersection protocols using an encryption scheme with rational numbers and set encoding with a fixed-length vector. Assume that both parties $P_1$ and $P_2$ have sets $S_1$ and $S_2$, respectively. After executing the protocol for confidential calculation of $S_1 \cap S_2$, they want to implement as follows. (1) The result of the collaborative calculation is 100% equal to $S_1 \cap S_2$, unlike some previous private set intersection protocols, such as protocols[1,5,8-10], there may be errors in the results of collaborative computing, and the scope of the errors is difficult to define; (2) After collaborative completion of calculation of $S_1 \cap S_2$, $P_1$ and $P_2$ do not disclose the elements of their respective sets, nor do they disclose the cardinality of their respective sets. That is, after $P_1$ and $P_2$ complete the calculation of $S_1 \cap S_2$ together, the calculation result of $S_1 \cap S_2$ must be correct, they cannot get any other information about each other except $S_1 \cap S_2$ (the elements of the sets, the cardinality of the sets).

This work is supported by the National Nature Science Foundation of China (Nso. 61972225, 61902164, 61601358, 61672426, 61902300, 61902303, 11847101), the National Science and Technology Support Plan (2018YFB1004501), the Key Scientific Research Program Project of Department of Education of Shaanxi Province(No.20JS052), the Special Plan for technological Innovation guidance of Shaanxi Province in 2020 (2020CGXNG-012) and the Research Fund for the Doctoral Program of Xi'an Polytechnic University (No. 107020331).