保密集合相交问题的高效计算

周素芳[1] 李顺东[1] 郭奕旻[2] 窦家维[3] 陈振华[4]

[1](陕西师范大学计算机科学学院 西安 710062)

[2](中国科学院软件研究所可信计算与信息保障实验室 北京 100190)

[3](陕西师范大学数学信息与科学学院 西安 710062)

[4](西安科技大学计算机科学与技术学院 西安 710054)

摘 要 安全多方计算作为网络空间安全的关键技术,是密码学的一个重要研究方向,是近年来国际密码学界研究的热点.科学计算是安全多方计算的一个重要分支.集合论是现代数学最重要的基础,许多数学分支都是以集合论为基础建立的.由于许多问题都可以抽象成集合问题,集合论及其数学思想被运用到越来越多的领域.因此保密的集合计算成为安全多方计算的一个重要方向.集合相交的保密计算是集合保密计算的一个重要问题,得到了广泛的关注.该问题在隐私保护方面有许多应用,如保密的数据挖掘、保密的数据外包、医疗敏感数据分析、个人财产数据及其他隐私数据的安全共享等.现有的关于集合相交保密计算的研究可以分为两个方面.一方面是研究有两个参与者且他们的集合都取自于一个无限大集合的情况.尽管该情况下研究者较多,但是该情况下的解决方案仅是计算性安全的而且存在计算效率较低的问题.另一方面是研究有多个参与者的情况,在这种情况下现有的解决方案比较少,且效率较低.该文针对在不同适用情况下集合相交存在的问题,设计了不同的解决方案.在有多个参与者的情况下,该文首先利用将集合表示成多项式的方法,设计了一个不需要借助密码学原语的、具有信息论安全的、计算复杂性低且通信效率高的安全多方交集计算方案.通过对该方案的改进,作者给出了另一个计算复杂性更低的方案,但该方案需要牺牲少量的通信效率.接下来,对于有两个参与者且参与者的集合取自于一个无限大集合的情况,该文利用单向散列函数的性质设计了一个高效的交集计算方案.此外,对于两个参与者的集合取自于一个有限集合子集的场合,该文利用离散对数困难性假设提出了高效的解决方案.同时,作者给出的解决方案经过简单改造可以用来保密地计算集合交集和并集的势以及认证的集合保密计算问题.最后,作为方案的应用,该文用多方集合相交的方案解决了求多个数最大公约数的保密计算问题.作者使用安全多方计算普遍采用的模拟范例证明方法证明了这些方案在半诚实模型下是安全的.

关键词 密码学;安全多方计算;交集;多项式;最大公约数

中图法分类号 TP309 **DOI 号** 10.11897/SP.J.1016.2018.00464

Efficient Secure Set Intersection Problem Computation

ZHOU Su-Fang[1] LI Shun-Dong[1] GUO Yi-Min[2] DOU Jia-Wei[3] CHEN Zhen-Hua[4]

[1](*School of Computer Science, Shaanxi Normal University, Xi'an* 710062)

[2](*Trusted Computing and Information Assurance Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing* 100190)

[3](*School of Mathematics and Information Science, Shaanxi Normal University, Xi'an* 710062)

[4](*School of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an* 710054)

Abstract Secure multi-party computation, which is a key technology of the information security in the cyberspace, is an important field of research in cryptography, and it is a research focus in the international cryptographic community in recent years. Scientific computation is a branch of secure multiparty computation. Set theory is the most important base of modern mathematics,

收稿日期:2014-11-10;在线出版日期:2017-05-23.本课题得到国家自然科学基金面上项目(61272435)、中央高校基本科研业务费专项资金(2016TS061)和国家留学基金委资助.**周素芳**,女,1990 年生,博士研究生,主要研究方向为信息安全.E-mail:zhousufang@snnu.edu.cn.**李顺东**,男,1963 年生,博士,教授、博士生导师,中国计算机学会(CCF)会员,主要研究领域为密码学与信息安全.**郭奕旻**,女,1992 年生,博士研究生,主要研究方向为信息安全.**窦家维**(通信作者),女,1963 年生,博士,副教授,主要研究方向为应用数学与应用密码学.E-mail:jiawei@snnu.edu.cn.**陈振华**,女,1976 年生,博士,副教授,主要研究方向为信息安全.

and many mathematical branches are based on set theory. Since many problems can be abstracted as set problems, set theory and its mathematical thought are applied in more and more fields. The secure set computation is a highly important problem in the secure multiparty computation. Secure set intersection computation is an important problem within the secure set computation and attracts many attentions. The secure set intersection computation has many applications in the privacy preservation, such as the secure data mining, secure data outsourcing, analysis of the sensitive medical data, and secret sharing of the personal property data and other private data, etc. At present, the research of the secure set intersection computation has two aspects. On one hand, researchers research on the protocols that there are two parties, and their sets are taken from an infinite set. Even though most of researches focus on this circumstance, the solution for this circumstance is only computationally secure and not so efficient in term of computational complexity. On the other hand, the research for the secure multi-party set computation is quite few, and they are not efficient either. This paper designs different solutions for the different situations where the researchers have not well solved. In the multi-party set intersection computation, based on the polynomial representation of a finite set, this study first constructs a secure multi-party set intersection protocol which is not based on any primitives of cryptography, and it is information theoretically secure and has a low overhead of the computation and the communication. Based on this multi-party protocol, we offer another protocol that has less computational complexity, while it sacrifices a little communication complexity. In the next, for the two-party set intersection computation and the sets of these two parties are taken from an infinite set, this manuscript presents an efficient protocol based on the one-way property of the one-way hash function. In addition, for the situation that the sets of two parties are subsets of a finite set, this work introduces an efficient protocol based on the assumption of the hardness of computing discrete logarithm. At the same time, the protocol we present for the two-party set intersection computation can be used to either the secure computation of the cardinality of the set intersection or the set union, and the authenticating of the secure set computation with a little changes. Finally, as an application of the first protocol, we demonstrate how to use the protocol to privately compute the greatest common divisor of several private numbers. The protocols that we present in this paper are proven to be secure in the semi-honest model using the simulation paradigm which is widely used in secure multiparty computation.

Keywords　cryptography; secure multi-party computation; set intersection; polynomial; greatest common divisor

1　引　言

网络的迅速发展为多方合作计算提供了巨大的机会，使安全多方计算(Secure Multi-party Computation，SMC)成为国际密码学界研究的热点问题. 安全多方计算是指两个或多个参与者联合进行的秘密计算，计算结束后，各个参与者除了得到既定的输出结果外，其输入信息没有任何泄露. 如果可以借助于可信的第三方，安全多方计算将变得非常简单. 但在复杂的网络中，找到一个公认的可信第三者是不可能的，因而需要使用其他的方法解决该问题.

两个参与者的安全多方计算最早由姚期智教授[1]在 1982 年提出，随后 Ben-Or 和 Goldwasser 等人[2]在 1988 年给出了多个参与者的安全多方计算. 安全多方计算是密码学中的一个基本问题，许多密码学问题都可以看作是安全多方计算的一个特例. Goldreich 等人[3-4]从理论上证明了任意的安全多方计算问题都是可解的，并给出了通用的解决方案，但指出从计算效率考虑，对具体的问题应该研究具体的解决方案. 同时 Goldreich 利用比特承诺[5]和零知识证明[6]设计了一个编译器，证明了给定一

个对于半诚实参与者安全的安全多方计算协议，借助于这个编译器，可以自动生成一个对于恶意参与者也是安全的安全多方计算协议，该编译器可以强迫恶意参与者以半诚实的方式参与协议的执行过程，这使得多数的安全多方计算研究都只研究基于半诚实模型下的保密计算问题. Goldwasser[7] 预言安全多方计算将成为计算科学中一个必不可少的组成部分. Goldreich 和 Goldwasser 等的研究激励着人们研究各种各样的安全多方计算问题，这些问题包括百万富翁问题、保密的数据挖掘、保密的信息比较、保密的统计分析、保密的远程访问、保密拍卖、保密的计算几何等[8-10].

集合是一个非常重要的概念，现实中的很多问题都可以用集合表示，集合问题的研究是安全多方计算研究的一个重要方面，其中集合相交问题在保密的数据挖掘[11]、数据外包[12]、医疗敏感数据分析[13]、个人财产数据及其他隐私数据的安全共享[14]等方面有重要的应用.

现有集合相交问题的解决方案多是基于一些密码学算法，如 Kissner 等人[15] 提出的方案基于 Paillier[16] 同态加密算法和多项式求值，Soled 等人[17] 提出的方案基于 ElGamal[18] 同态加密算法和多项式求值，文献[19]给出的方案基于 Naor-Reingold 的类随机函数，文献[20]给出的方案基于可交换的密码算法，文献[21]给出的方案基于秘密共享[22]等. 最著名的是 Freedman[23] 利用多项式不经意求值并借助于 Paillier 同态加密算法和平衡哈希函数提出的一种高效求集合交集的解决方案，其安全性是语义安全的. 但该方案主要适用于两个参与者的情况，当扩展到多个参与者的情况时其计算复杂性和通信复杂性都非常高；而且该方案只适用于参与者的集合是一个无限大集合子集的情况（其论文中明确指出参与者集合取自于指数大的域），但实际应用中这种情况并不多. 在其后的发展中，求集合交集的解决方案多趋向于两个参与者的情况，对多个参与者的情况研究比较少. Li 等人[21] 提出了第一个信息论安全保密求集合交集的方案，其主要借助于 Freedman 方案中将集合表示成多项式根的形式和秘密共享方案，虽然不需要使用加密算法，但其仍然需要借助于比较低效的密码学原语（primitive），计算复杂性仍然比较高. 其后提出的一些具有信息论安全的保密求集合交集的方案如文献[24-27]等也都是基于一些密码学原语.

密码学算法与协议的安全性可分为计算性安全和信息论安全（或无条件安全）. 计算性安全是指算法或协议对于有多项式时间计算能力的敌手来说是安全的，其安全性是基于某些计算困难性假设的，如大整数因式分解假设、离散对数假设、二次剩余假设等. 这些困难性假设只是对计算能力有限的敌手来说是安全的，但对于有无限计算能力的敌手的情况来说并不安全（语义安全是计算性安全的一种）. 信息论安全的算法或协议的安全性是基于信息论安全的，这类算法或者协议使得攻击者无法获得足够的信息来破坏其安全性[28]，信息论安全的算法或协议即使对于有无限计算能力的敌手来说也是安全的. 在量子计算环境下，计算安全的算法与协议都不再安全，而信息论安全的算法与协议即使对于量子计算攻击仍然是安全的，因此研究具有信息论安全的集合交集问题的解决方案是非常有意义的.

本文根据多项式的性质和回路分区的思想[29]，提出了一种计算复杂性和通信复杂性都很低，且不需要借助于密码学原语求多个参与者集合交集的方案，同时其安全性是信息论安全的. 该方案对于参与者集合中的元素取自于无限集合和有限集合都适用，但该方案不适用于两个参与者的情况. 接着本文又针对两个参与者的情况分别提出了适合于参与者的元素取自于有限集合和无限集合的方案. 最后用本文中提出的新方案解决了多个参与者保密求最大公约数问题.

本文贡献如下：(1) 利用多项式的性质，提出了一种不需要借助于密码学原语保密计算多个集合相交问题的解决方案，并证明了它是信息论安全的，该方案的计算效率和通信效率都比较高，且具有广泛的适用范围；(2) 对于两个参与者的情况，用单向散列函数给出了参与者的集合取自于无限集合的高效解决方案；当参与者的集合取自于一个有限集合时，在现有的方案中没有发现解决办法，本文基于离散对数困难性假设给出了解决方案，给出的解决方案经过改造可以用来保密地求集合交集和并集的势以及认证的集合保密计算问题；(3) 研究了最大公约数的保密计算问题，利用求多个集合相交的方案解决了该问题.

本文在第 2 节给出本论文中用到的一些基本知识. 多个参与者高效保密计算集合交集的方案在第 3 节给出，并证明方案的安全性；在第 4 节，关于两个参与者的情况，给出参与者集合中的元素取自于有限集合和无限集合两种方案，同时对它们的安全性进行分析；在第 5 节中，将多个参与者高效保密求

交集的方案经过改造应用到高效保密求多个参与者的最大公约数中，并运用模拟范例证明了方案的安全性；第 6 节是对本文的总结和进一步研究工作的展望.

2　预备知识

2.1　理想模型

假设有一个可信的第三者(Trusted Third Party，TTP)，他在任何情况下都不会撒谎，也绝不会泄露任何不该泄露的信息. 理想的安全多方计算即为多个参与者需要借助于 TTP 进行的保密计算. 若有 n 个参与者 $P_1,\cdots,P_n$，他们分别将各自的秘密数据 $x_1,\cdots,x_n$ 发送给 TTP，TTP 自己单独计算函数 f：$f(x_1,\cdots,x_n)=(f_1(x_1,\cdots,x_n),\cdots,f_n(x_1,\cdots,x_n))$，然后将计算结果

$$f_1(x_1,\cdots,x_n),\cdots,f_n(x_1,\cdots,x_n)$$

分别发送给参与者 $P_1,\cdots,P_n$. 因此 $P_1,\cdots,P_n$ 除了从协议得到 TTP 发送给自己的计算结果外得不到任何其他信息. 该协议是一个理想的安全多方计算协议. 理想的安全多方计算协议虽然简单，但其安全性却是最高的安全多方计算协议，任何一个计算函数 $f(x_1,\cdots,x_n)$ 的实际安全多方计算协议的安全性都不可能超过这个协议，因此其他保密协议可以通过和理想模型比较来检验其安全性.

2.2　半诚实模型

本文假设所有的参与者都是半诚实的. 对于半诚实的参与者，在协议的执行过程中，所有参与者都会按照协议要求忠实地履行协议，执行协议后，除协议的计算结果外没有人知道其他参与者的输入，但他们同时可能会记录下来协议执行过程中收集到的所有信息，并试图根据收集到的信息(多个参与者时我们应该考虑多个参与者在执行协议中收集到的信息，而不是一个参与者收集到的信息)推算出其他参与者的输入. 所以半诚实模型又称为诚实且好奇模型或被动模型.

2.3　模拟范例

设 $f:(\{0,1\}_1^*,\cdots,\{0,1\}_n^*)\to(\{0,1\}_1^*,\cdots,\{0,1\}_n^*)$ 是一个 n 元概率多项式时间函数，令

$$\bar{x}=(x_1,\cdots,x_n),\ f(\bar{x})=(f_1(\bar{x}),\cdots,f_n(\bar{x})),$$

其中 $f_i(\bar{x})$ 表示 $f(\bar{x})$ 的第 i 个输出元素. 参与者 P_i $(i=1,\cdots,n)$ 参与保密计算的目的是要在不泄露 x_i 的情况下得到 $f_i(\bar{x})$. 设 Π 是计算 f 的协议，在协议执行过程中，P_i 得到的信息序列为

$$view_i^{\Pi}(\bar{x})=(x_i,r_i,m_i^1,m_i^2,\cdots,m_i^t),$$

其中 $m_i^j(j=1,2,\cdots,t)$ 表示 P_i 第 j 次收到的信息. $output^{\Pi}(\bar{x})$ 记为协议 Π 执行后的输出结果.

在参与者都是半诚实的情况下，协议 Π 计算的函数 f 是保密的，如果存在概率多项式时间算法 S 使得下式成立：

$$\{(S(x_i,f_i(\bar{x})),f(\bar{x}))\}\overset{c}{\equiv}\{(view_i^{\Pi}(\bar{x}),output^{\Pi}(\bar{x}))\}.$$

其中，$\overset{c}{\equiv}$ 表示计算上不可区分.

2.4　离散对数假设

一些记号：自然数 n 模 p 的最小剩余集记作 $Z_p=\{0,1,2,\cdots,p-1\}$，Z_p 中与 p 互素的元素的个数为 $\phi(p)$，称为欧拉函数. Z_p^* 为 Z_p 中与 p 互素的元素组成的乘法群. 如果 p 是一个素数，则 $Z_p^*=\{1,2,\cdots,p-1\}$，$\phi(p)=p-1$. 如果 $\alpha\in Z_p^*$，则使等式 $\alpha^t \bmod p=1$ 成立的最小正整数 t 为 α 的阶 $\mathrm{ord}(\alpha)$. 如果 $\mathrm{ord}(\alpha)=\phi(p)=p-1$，那么 α 是 Z_p^* 的生成元. α 是生成元意味着 $\{\alpha^1,\alpha^2,\cdots,\alpha^{\phi(p)}\}$ 是 Z_p^* 中元素的一个置换.

离散对数假设：给定一组自然数 α,β,p，其中 $\alpha\in Z_p^*$ 是一个生产元，p 是一个大素数，$\beta\in Z_p^*$，则对于任何多项式时间算法 $\mathcal{A}$，任意的多项式 $p(\cdot)$ 和充分大的 l 都有

$$\Pr[(\mathcal{A}(\alpha,\beta,p)=x)\wedge(\alpha^x \bmod p=\beta)]<\frac{1}{p(l)}.$$

2.5　集合相交问题

设有 n 个参与者 $P_1,\cdots,P_n$ 分别拥有秘密集合

$$X_1=\{x_{11},x_{12},\cdots,x_{1l_1}\},$$

$$\cdots,$$

$$X_n=\{x_{n1},x_{n2},\cdots,x_{1l_n}\},$$

他们需要计算这些集合的交集 $X=X_1\cap\cdots\cap X_n$. 保密的集合相交需要各个参与者知道他们集合的交集，但对其他参与者的集合没有任何信息.

最简单的保密求集合相交问题是有两个参与者，他们各自只有一个元素，然后比较这两个元素是否相等，即社会主义百万富翁问题[30].

现有保密求集合交集的方案都是基于某种密码学原语给出的解决方案，所有的密码学原语相对于普通四则运算的计算复杂性都非常高，所以导致求交集的计算复杂性也非常高. 在求集合交集的保密计算方案中，Freedman 的方案是最有创意的方案，其后的一些方案的设计思想也来源于此方案，方案比较好地解决了集合相交的问题. 但方案中存在一些不足.

首先，Freedman 提出的解决方案主要基于 Paillier 公钥加密算法. 一方面 Paillier 公钥加密算法的效率非常低，每次加密或解密需要 $2\log N$(模 N^2)次模乘运算. 另一方面 Paillier 公钥加密算法的安全性是语义安全的，语义安全是基于计算性困难假设的，对于有多项式时间计算能力的敌手来说是困难的，但对于有无限计算能力的敌手来说并不安全. 随着计算能力的发展，如果参与者的计算能力较强时，可以破解该加密算法，方案将不再安全. 同时，每个参与者还需要知道其余 $n-1$ 个参与者的公钥，公钥的获得，无疑又增加了计算负担.

其次，在 Freedman 的方案中，如果参与者的集合是一个有 r 个元素的有限集合的子集，$\mathcal{S}$随机地选取一个不是自身集合内的元素 R 计算后发送给 $\mathcal{C}$，R 是$\mathcal{C}$拥有集合内的元素的概率为 $1/r$，这个概率是不可忽略的，所以其方案要求所有参与者集合中的元素取自于一个指数大的集合，而大多数实际应用场合都不符合这个假设前提，因而该解决方案在大多数实际应用中都不适用.

最后，在 Freedman 的方案中，通过使用 Horner 算法和平衡哈希函数，将两个参与者方案的通信复杂度降低到 $O(l)$，计算复杂度降低到 $O(l\ln\ln l)$，其中 l 为参与者集合中元素的个数. 但是当有 n 个参与者时，需要做 $l(n-1)(2n+l)$次的加密或解密运算，需要 $2n^2l$ 的通信量，同时该方案的通信复杂度和计算复杂度都是基于 Paillier 公钥加密算法的密文长度和计算量的，所以 Freedman 的方案主要适用于两个参与者的情况，当有多个参与者时，其计算复杂性和通信复杂性都随着参与者的增加而迅速提高.

本文提出了一种新的解决方案，解决了 Freedman 方案中存在的问题.

首先，本文提出的方案不需要各个参与者使用加密算法加密自己集合中的元素. 由于现有加密算法的计算复杂性都比较高，对于普通用户的计算能力并不适用，而本文中的方案不再需要加密算法，可以在普通设备中实现保密计算，本文中的方案也不需要获得和存储其他参与者的公钥，减少了计算量. 该方案是信息论安全的，即使有无限计算能力的敌手，在没有完全获得某个参与者构造的秘密多项式时，也不能计算出关于该参与者集合中的元素.

其次，由于多项式的性质，新方案对参与者集合的取值范围没有限制，既适用于无限集合，也适用于有限集合，这样更适合在实际计算环境中使用.

最后，本文中提出的方案由于不需要加密，且每个参与者不需要给其他参与者发送加密的随机数，共需要 $2l\cdot l_n$次普通乘法运算和发送 $2n$ 个 $2l$ 阶多项式. 由于普通乘法计算与 Paillier 公钥加密算法的加、解密运算相比，其运算非常简单，且一般集合中元素的位数远小于 Paillier 公钥加密算法密文的长度，所以本文中给出的协议在有多个参与者时效率比较高.

3 多个参与者保密计算交集

3.1 具体方案

假设有 n 个参与者 $P_1,\cdots,P_n$，每个参与者拥有集合元素的个数为 $l_1,\cdots,l_n$，对应于 n 个参与者的集合分别为

$$X_1=\{x_{11},x_{12},\cdots,x_{1l_1}\},$$

$$\cdots,$$

$$X_n=\{x_{n1},x_{n2},\cdots,x_{1l_n}\},$$

求 n 个参与者的交集 $X=X_1\cap\cdots\cap X_n$.

在 Freedman 的方案中，每个参与者将自己的集合用多项式的形式表示，集合中的每个元素是多项式的一个根. 在本文给出的方案中，表示集合的方法与 Freedman 方案中的方法类似，每个参与者也将自己的集合用多项式的形式表示，不同的是每个参与者将自己集合中的每个元素作为多项式的根分别构造属于自己的 2 重秘密多项式. 如果对于每个参与者 $P_i(i=1,\cdots,n)$拥有集合元素的个数 l_i 都有 $l_i\leq l$，则 P_i 为自己构造的多项式增加份额 $x^{2(l-l_i)}$，使每个参与者多项式的次数都保证为 $2l$ 次，参与者 $P_1,\cdots,P_n$构造的秘密多项式分别为

$$f_1(x)=x^{2(l-l_1)}(x-x_{11})^2(x-x_{12})^2\cdots(x-x_{1l_1})^2,$$

$$\cdots,$$

$$f_n(x)=x^{2(l-l_n)}(x-x_{n1})^2(x-x_{n2})^2\cdots(x-x_{nl_n})^2.$$

由此可知，若 $x=a$，$f_i(a)=0$ 当且仅当 $a\in X_i$.

参与者 $P_i(i=1,\cdots,n-1)$将自己的多项式整理成如下形式：

$$f_i(x)=\sum_{q=0}^{2l}a_{iq}x^q,$$

并将多项式随机地分成非零的 $k(k\leq n)$份，即 $f_{i1}(x),f_{i2}(x),\cdots,f_{ik}(x)$，使

$$f_i(x)=f_{i1}(x)+f_{i2}(x)+\cdots+f_{ik}(x).$$

即使敌手$\mathcal{A}$获得关于 P_i多项式 $f_i(x)$的 $k-1$ 份多项式

$$f_{i1}(x),\cdots,f_{i(k-1)}(x),$$

但$\mathcal{A}$不知道第 k 份多项式 $f_{ik}(x)$，也无法得到多项式 $f_i(x)$. 因为

$$f_i(x)=f_{i1}(x)+\cdots+f_{i(k-1)}(x)+f_{ik}(x),$$

多项式 $f_{ik}(x)\neq 0$ 而且是随机的，在这一个方程中有两个未知量(后面通过加随机量强化的方法中更导致一个方程三个未知量)，是一个不定方程，所以$\mathcal{A}$无法得到多项式 $f_i(x)$，因而无法求出 $f_i(x)$的根，也就无法得到 P_i拥有集合中的元素.

为了增加多项式的安全性，参与者 $P_i(i=1,\cdots,n-1)$为多项式 $f_i(x)$的每个份额 $f_{ij}(x)(j=1,2,\cdots,k)$加上一个随机数 r_{ij}，为了降低后期消除随机数的计算量，需要保证

$$r_{i1}+r_{i2}+\cdots+r_{ik}=0.$$

P_i将自己 k 份加入随机数的多项式 $f_{ij}(x)+r_{ij}(j=1,2,\cdots,k)$分别发送给 n 个参与者中的 k 个(这 k 个参与者可以包括 P_i自己，也可以不包括，其他人不知道 P_i分发的 k 份包括不包括自己，即这个 k 是不固定的)，其他参与者不知道具体分发给了哪几个参与者. 当每个参与者 P_i将其多项式都分发后，P_i也会收到其他参与者发送过来的多项式份额，P_i将其收到的所有多项式份额相加，组成新的多项式 $g_i(x)$，然后将 $g_i(x)$发送给 P_n.

参与者 P_n将收到的所有多项式份额相加并加上自己原有的秘密多项式，构成一个新的多项式

$$f(x)=g_1(x)+g_2(x)+\cdots+g_n(x).$$

任何一个参与者 $P_i(i=1,\cdots,n)$的多项式 $f_i(x)$被分成 k 份发送给 k 个参与者，这 k 份之和仍是多项式 $f_i(x)$. 所有参与者都是半诚实的，会严格按照协议执行. P_i的 k 份多项式包含在收到它们的 k 个参与者最终发给 P_n的多项式中，因此 P_n最后计算的多项式包含 $f_i(x)$. 对每一个 $P_i(i=1,\cdots,n)$都是如此，所以 $f(x)$包含 $f_i(x)(i=1,\cdots,n)$；由于半城实的参与者不会在发给 P_n的多项式中添加任何额外的信息，$f(x)$中只包含 $f_i(x)(i=1,\cdots,n)$. 所以参与者 P_n最终构成的多项式之和 $f(x)$是而且只是 n 个参与者秘密多项式的和，即

$$f(x)=f_1(x)+f_2(x)+\cdots+f_n(x).$$

P_n将自己的 l_n个元素 $x_{n1},x_{n2},\cdots,x_{1l_n}$分别代入多项式 $f(x)$中，若 $f(x_{nj})=0(j=1,2,\cdots,l_n)$，则元素 x_{nj}为 n 个参与者交集中的元素. 由于 n 个参与者集合的交集是每个参与者集合的子集，所以 P_n将自己秘密集合中的所有元素代入后，可以确定 n 个参与者的交集 X. P_n将交集 X 公布.

定理 1.　参与者 P_n将自己的元素 x_{nj}代入多项式 $f(x)$中，$f(x_{nj})=0$ 当且仅当 x_{nj}是所有参与者交集中的元素.

证明.　必要性. 当 x_{nj}是所有参与者交集中的元素时，每个参与者构造的多项式 $f_i(x)(i=1,2,\cdots,n)$中都含有$(x-x_{nj})^2$即

$$f_i(x)=(x-x_{nj})^2 f_i^*(x),$$

其中 $f_i^*(x)$为多项式 $f_i(x)$除去$(x-x_{nj})^2$剩余的部分多项式. 当 P_n将 x_{nj}代入 $f(x)$中时，由于

$$f_1(x_{nj})=0,f_2(x_{nj})=0,\cdots,f_n(x_{nj})=0,$$

则

$$f(x_{nj})=f_1(x_{nj})+f_2(x_{nj})+\cdots+f_n(x_{nj})=0.$$

所以当 x_{nj}是参与者交集中的元素时，$f(x_{nj})=0$.

充分性. 对于一般的由 n 个多项式 $f_i(x)(i=1,2,\cdots,n)$之和组成的多项式 $f(x)$来说，$f(x)=0$ 时组成 $f(x)$的每个多项式 $f_i(x)$不一定等于 0，即当参与者 P_n将自己的元素 x_{nj}代入多项式 $f(x)$中时，会出现 $f(x_{nj})=0$ 而组成 $f(x)$的多项式 $f_i(x_{nj})$不都等于 0 的情况. 但在本方案中，每个多项式 $f_i(x)$是由相应参与者 P_i拥有集合的元素为根组成的 2 重多项式，这样当参与者 P_n将自己拥有集合的元素 x_{nj}代入 $f(x)$中时，组成 $f(x)$的每个多项式 $f_i(x)$的值只有是该多项式的根时才等于 0，其余情况均大于 0，从而可以保证当 $f(x)=0$ 时组成 $f(x)$的每个多项式 $f_i(x)=0$，即

$$y=x_1^2+x_2^2+\cdots+x_n^2$$

形式的公式，$y=0$ 当且仅当 $x_1=x_2=\cdots=x_n=0$. 由于所有参与者的交集是每个参与者集合的子集，不会出现 x'_{nj}是交集中的元素，但不是参与者 P_n集合中的元素. 所以，参与者 P_n将自己的元素 x_{nj}代入多项式 $f(x)$中，$f(x_{nj})=0$ 时，x_{nj}是所有参与者交集中的元素.　　证毕.

在此基础上构造多个参与者保密求交集的解决方案，具体如协议 1.

协议 1.　多个参与者求交集的安全多方计算.

输入：$P_1,P_2,\cdots,P_n$各自的秘密集合 $X_1,X_2,\cdots,X_n$

输出：$X=X_1\cap X_2\cap\cdots\cap X_n$

1. $P_1,P_2,\cdots,P_n$以各自集合 $X_1,X_2,\cdots,X_n$中的元素为根，分别构造秘密多项式 $f_1(x),f_2(x),\cdots,f_n(x)$.
2. 参与者 $P_i(i=1,2,\cdots,n-1)$计算如下：
   (a) $P_1,P_2,\cdots,P_{n-1}$将自己的秘密多项式随机地分成非零的 k 份并加入随机数，然后发送给 n 个参与者中的 k 个.

(b) $P_1,P_2,\cdots,P_{n-1}$ 将各自收到的多项式份额相加，组成新的多项式 $g_1(x),g_2(x),\cdots,g_{n-1}(x)$，并分别发送给 P_n.

3. P_n 计算如下：

(a) SETUP：$f(x)\leftarrow g_1(x)+g_2(x)+\cdots+g_{n-1}(x)+g_n(x)$，$X\leftarrow\varnothing$.

(b) FOR $j=1$ to l_n {

computes $f(x_{nj})$;

IF $f(x_{nj})=0$

THEN $X\leftarrow X\cup\{x_{nj}\}$; }

(c) OUTPUTS：X.

3.2 方案分析

正确性分析.

由于参与者 P_n 最终组成的多项 $f(x)$ 一定等于参与者 $P_1,P_2,\cdots,P_n$ 的秘密多项式之和 $f_1(x)+f_2(x)+\cdots+f_n(x)$，同时，由定理 1 可知该协议中参与者 P_n 将秘密集合 X_n 中的元素代入多项式 $f(x)$ 中求出的交集 X 是 n 个参与者的交集，所以协议 1 是正确的.

抗合谋性分析.

为了便于理解，这里以五个参与者，每个参与者将自己构造的多项式分为 2 份为例进行说明.

当五个参与者 A,B,C,D,E 共同求其拥有集合的交集时，A,B,C,D,E 将各自集合的元素作为多项式的根构造相应的 2 重多项式 f_A,f_B,f_C,f_D,f_E.

参与者 A,B,C,D 将各自的多项式随机地分成非零的两份

$$f_A=f_{A_1}+f_{A_2},\ f_B=f_{B_1}+f_{B_2},$$

$$f_C=f_{C_1}+f_{C_2},\ f_D=f_{D_1}+f_{D_2},$$

然后分别发送给五个参与者中的两个参与者. 不失一般性，不妨令参与者 A,B,C,D 各自分别保存多项式 $f_{A_1},\cdots,f_{D_1}$，然后将剩下的那份多项式 $f_{A_2},\cdots,f_{D_2}$ 分别发送给其余的 4 个参与者中的一个，如 A 将 f_{A_2} 发送给 B，B 将 f_{B_2} 发送给 C，C 将 f_{C_2} 发送给 D，D 将 f_{D_2} 发送给 A，而其他参与者不知道某个参与者具体将自己的两份多项式分发给了哪两个参与者. 然后每个参与者将自己保留的多项式和收到来自其他参与者的多项式相加组成新的多项式

$$g_A=f_{A_1}+f_{D_2},\ g_B=f_{B_1}+f_{A_2},$$

$$g_C=f_{C_1}+f_{B_2},\ g_D=f_{D_1}+f_{C_2},$$

并发送给 E.

参与者 E 令所有得到的多项式之和为 f，即

$$\begin{aligned}f&=g_A+g_B+g_C+g_D+f_E\\&=f_A+f_B+f_C+f_D+f_E,\end{aligned}$$

然后 E 将自己集合中的元素代入 f，求出所有参与者的交集 X，并将其公布. 具体情况如图 1.

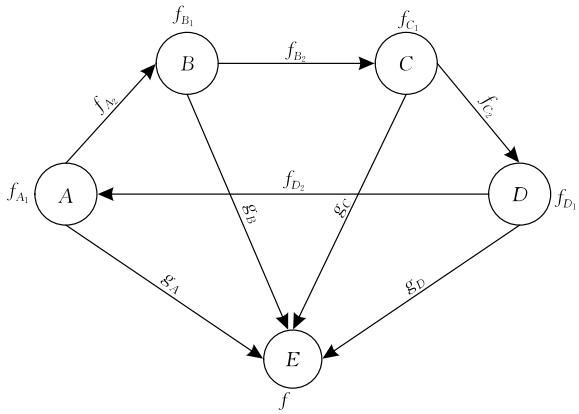

图 1 5 个参与者示例图

当 C 想要根据自己的输入和收到 B 的部分多项式 f_{B_2} 来计算 B 秘密集合的元素时，由于 C 只知道 B 的部分多项式 f_{B_2}，显然，即使 C 有无限的计算能力也不能计算出 B 拥有集合中的元素. 但是 C 不放弃，仍然希望得到 B 的另一半多项式 f_{B_1}. 由于 B 将自己另一半多项式 f_{B_1} 加上收到来自 A 的多项式 f_{A_2}（即 $g_B=f_{A_2}+f_{B_1}$）发送给了 E，若 C 想要获得 f_{B_1}，则他需要和 A、E 两个参与者合作才可以得到，即 $f_{B_1}=g_B-f_{A_2}$. 若 A 和 E 中的任何一个人拒绝合作，C 都不能得到多项式 f_{B_1}，即不能计算出 B 所拥有的元素. 若内部参与者 D 或外部敌手想要知道 B 的输入，情况更复杂，需要和更多的参与者合谋. 如果 B 没有收到来自 A 的多项式 f_{A_2}，B 发送给参与者 E 的多项式只是自己的部分多项式，则 C 只需要和参与者 E 合谋，就可以知道 B 的多项式.

由上述例子可知，每个参与者将多项式分成两份，若要恢复某个参与者的多项式，至少需要两个参与者的合谋. 扩展到一般情况，若有 n 个参与者，每个参与者 $P_i(i=1,2,\cdots,n)$ 将自己的秘密多项式分成 $k(k\leq n)$ 份，然后随机地发送给 n 个参与者中的 k 个参与者，如 $P_{i+1\bmod n},P_{i+2\bmod n},\cdots,P_{i+k\bmod n}$，其中 $(P_{i+1\bmod n},P_{i+2\bmod n},\cdots,P_{i+k\bmod n})\subseteq(P_1,P_2,\cdots,P_n)$. 如果某个参与者如 $P_{i+1\bmod n}$ 想要知道参与者 P_i 的秘密多项式，必须要和其他的 $k-1$ 个参与者 $P_{i+2\bmod n},\cdots,P_{i+k\bmod n}$ 合谋（如果 $P_i(i=1,2,\cdots,n)$ 分发 k 份时给自己留了一份，即分发的 k 个参与者中包含自己，而 $P_i(i=1,2,\cdots,n)$ 又不参与合谋，那么其他人永远无法得到他的多项式）. 所以每个参与者将多项式分成 k 份，即可以抵抗 $k-1$ 个参与者的合谋攻击（在许多保密计算场合只要 $n-1$ 个人合谋，一般都可以得到另一个人的全部数据，但在保密计算集合交集的场合，即使 $n-1$ 个人合谋，也只能得到另一个人数

据的很少一部分).最多需要将多项式分成 n 份,则可抵抗除 P_i 之外所有剩余参与者的合谋攻击.具体的实际应用系统可以根据其对安全性要求的高低来改变 k 的值.

安全性分析.

协议 1 的安全性通过定理 2 给出.

定理 2. 协议 1 是基于信息论安全的,即使有无限计算能力的敌手,在没有完全获得某个参与者构造的秘密多项式时,也不能计算出关于该参与者集合中的元素.

证明. 我们分下面两种情况证明协议 1 是信息论安全的.

(1)每个参与者 $P_i(i=1,2,\cdots,n-1)$ 对于其他参与者是信息论安全的.收到 P_i 多项式的所有参与者合谋才可以恢复出 P_i 集合中的元素,而任何少于这个数量的参与者合谋即使有无限计算能力也不能得到 P_i 集合中的元素.可以分为两种类型:

(a)参与者 P_i 将其 k 份多项式发送给其余 $n-1$ 个参与者中的 k 个(该情况下 P_n 没有特殊性).

参与者 $P_i(i=1,2,\cdots,n-1)$ 将其多项式 $f_i(x)$ 分成 k 份并加入随机数,即 $f_{ij}(x)+r_{ij}(j=1,2,\cdots,k)$,发送给其余 $n-1$ 个参与者中的 k 个,每个参与者最多得到关于 P_i 的 1 份多项式,收到 P_i 多项式的 k 个参与者合谋可以恢复 $f_i(x)$,因为

$$f_{i1}(x)+r_{i1}+\cdots+f_{ik}(x)+r_{ik}=f_i(x).$$

从而可以计算出 P_i 集合中的元素.如果有 $k-1$ 个有无限计算能力的参与者合谋,假设收到 $f_i(x)$ 前 $k-1$ 份 $f_{ij}(x)+r_{ij}(j=1,2,\cdots,k-1)$ 的 $k-1$ 个参与者合谋,要得到关于 P_i 的多项式 $f_i(x)$.他们最终得到方程

$$g(x)=\sum_{j=1}^{k}(f_{ij}(x)+r_{ij})=f_i(x)-f_{ik}(x)-r_{ik},$$

而多项式 $f_{ik}(x)\neq 0$ 与 $r_{ik}\neq 0$ 是随机的,这是一个含有 3 个未知量的不定方程,因此无法从这个多项式中获得关于 P_i 多项式的足够信息.所以 $k-1$ 个参与者合谋即使有无限的计算能力也无法得到多项式 $f_i(x)$,从而得不到 P_i 集合中的元素.

(b)参与者 P_i 将其 k 份多项式自己保留一份,然后将剩下的 $k-1$ 份发送给其余 $n-1$ 个参与者中的 $k-1$ 个.

在协议 1 中,P_i 除了需要自己保留一份并将剩余的 $k-1$ 份多项式发送给其余 $n-1$ 个参与者中的 $k-1$ 个(设为集合 I)外,还需要将其收到的来自其他参与者的多项式和自己保留的多项式组成多项式 $g_i(x)$ 发送给参与者 P_n,相当于把 $f_i(x)$ 发给了 $I\cup P_n$. P_n 在该情况下对其他参与者能否得到 $f_i(x)$ 有着决定性的作用.当 P_n 不参与合谋时,其余的 $n-2$ 个参与者合谋即使有无限的计算能力也不能得到 P_i 集合中的元素.当 P_n 和一些参与者 I^* 合谋,只有当 $I\subseteq I^*$ 时,才可以计算出 $f_i(x)$,所以至少需要收到 $k-1$ 份多项式的 $k-1$ 个参与者合谋才可以恢复出 P_i 集合中的元素.这仍相当于需要收到 P_i 多项式的所有参与者合谋才可以恢复出 P_i 集合中的元素.

(2)参与者 P_n 对于其他所有参与者 $P_i(i=1,2,\cdots,n-1)$ 是信息论安全的.由于参与者 P_n 在计算集合交集的过程中,只是得到了参与者 $P_i(i=1,2,\cdots,n-1)$ 发送过来的多项式 $g_i(x)$,而自己集合构成的多项式 $f_n(x)$ 的一部分没有和外界进行交互,其他参与者 $P_i(i=1,2,\cdots,n-1)$ 即使有无限的计算能力也不能根据 $f_n(x)$ 得到 P_n 的集合;同时,在 P_n 将自己的元素 x_{nj} 代入多项式 $f(x)$ 中求交集时,也只有 P_n 自己参与,$P_i(i=1,2,\cdots,n-1)$ 即使有无限的计算能力也不能得到 P_n 集合中的元素.计算结束后 P_n 将交集 X 公布,这与 P_n 将自己的集合发送给一个可信第三者计算的理想模型相同,其他参与者合谋即使有无限的计算能力也不能计算出关于 P_n 集合中的元素.

综上所述,协议 1 是信息论安全的. 证毕.

效率分析.

Freedman 首先借助于 Paillier 同态加密算法和集合的多项式表示给出了多个参与者保密求集合交集的协议,对于 n 个参与者,每个参与者集合中有 l 个元素,共需要 $l(n-1)(2n+l)$ 次加密或解密运算和 $2ln^2$ 的通信开销,且这些都是基于 Paillier 公钥加密算法的.文献[15]给出的多个参与者保密求集合交集的方案的计算效率和通信效率分别为 $O(n^2l^2)$,$O(n^2l)$ 且也是基于 Paillier 同态加密算法.文献[21]第一次给出了具有信息论安全保密求集合交集的方案,并指出其通信复杂度为 $O(n^4l^2)$,但随后文献[26]指出其通信复杂度实为 $O(n^7l)$,通信复杂度比较高,其后的方案[24-27]主要降低了具有信息论安全保密求集合交集方案的通信复杂度,文献[27]给出了比较高效的具有信息论安全的保密求集合交集方案,将通信量降低到 $n^3l\log(nl)$.

由于协议 1 是在 Freedman 方案基础上进行的改造,文献[27]给出了比较高效的具有信息论安全的方案,现将该方案与 Freedman 方案、文献[27]方案的效率进行比较.

计算复杂性分析.

忽略方案中构造多项式的开销.

Freedman 方案使用的是 Paillier 公钥加密算法,参与者 $P_1,P_2,\cdots,P_{n-1}$需要加密一个 l 行 $n-1$ 列的随机数矩阵和构造的多项式系数,共需要 $(n-1)nl$ 次加密运算,P_n在将自己的每个元素代入 $P_1,P_2,\cdots,P_{n-1}$构造的多项式时,需要做 $l^2(n-1)$ 次加密运算,$P_1,P_2,\cdots,P_{n-1}$每个人需要解密 n 个 l 行向量,需要解密 $nl(n-1)$次,所以在其方案中共用到 $l(n-1)(2n+l)$次加密或解密运算,由于 Paillier 的公钥加密算法每次加密或解密时需要 $2\log N$ 次模 N^2 的模乘运算,所以该方案共需要 $2\log N\cdot l(n-1)(2n+l)$次模 N^2的模乘运算.

文献[27]方案中,主要计算量在于将 n 个参与者中每个参与者拥有的 l 个元素进行不经意排序,共需要进行 $nl\log(nl)$次的比较交换,每次比较交换需要 $4q$ 次基本操作(q 为参与者元素的长度),同时要借助于 Shamir 秘密共享的方法保证信息论安全,需要 n 个参与者将拥有的每个元素以(t,n)门限共享给其他参与者,每次共享需要 t 次普通乘法,n 个参与者共需要进行 n^2次共享,所以共需要 $4qtn^3l\log(nl)$次普通乘法运算.

本方案只需要 P_n将拥有的 l_n个元素代入多项式 $f(x)$中进行求值运算,其中 $f(x)$是由 n 个 $2l$ 阶多项式组成的 $2l$ 阶多项式,对 $2l$ 阶多项式求值最多需要 $l(2l-1)$次普通乘法运算,如果借助于秦九韶算法(Horner 算法)①,将多项式

$$f(x)=a_0+a_1x+\cdots+a_{2l}x^{2l}$$

写成如下形式:

$$f(x)=a_0+x(a_1+x(a_2+\cdots x(a_{2l-1}+a_{2l}x)\cdots)),$$

对 $2l$ 阶多项式求值最多只需要做 $2l$ 次乘法运算和一些加法运算,则共需要 $2l\cdot l_n$次普通乘法运算.对于两个 m 位数的普通乘法运算,借助于快速傅里叶运算可以实现 $O(m\log m)$的计算效率.而 Paillier 公钥加密算法中的模乘运算需要先将两个数相乘,然后再除以 N^2求余数.设 N^2的位数为$|N^2|$,由于乘法运算是除法运算的逆运算,相当于多做 $O(|N^2|\log|N^2|)$次运算(其中 $N\approx2^{1024}$,$|N^2|\approx2048$).而且本文中的普通乘法运算的两个乘数都不大,位数 m 都远远小于 1024,所以普通乘法运算与 Paillier 公钥加密算法中的模乘运算相比可以忽略,所以本方案大大降低了原有方案的计算复杂性,同时也高于文献[27]的方案.

通信复杂性分析.

在 Freedman 的方案中,每个参与者都需要将自己构造多项式的系数和一个 l 行 $n-1$ 列的随机数矩阵经过 Paillier 公钥加密算法加密后发送给参与者 P_n或是在公告板上公开,需要的通信开销为 $nl(n-1)$.P_n根据 Paillier 公钥加密算法的同态性构造每个参与者的加密多项式,然后将每个元素分别代入他们的多项式中并随机化后再公布到公告板上,需要 nl 的通信开销.每个参与者仍需要从公告板上取回自己相应的份额进行解密计算,仍需要通信开销 n^2l.该方案中的密文空间为$|N^2|$,则总的通信开销为 $2n^2l\cdot|N^2|$.

文献[27]中,需要进行 $nl\log(nl)$次的比较交换,n 个参与者将秘密共享需要传递 n^2个元素,则总的通信量为 $n^3l\log(nl)\cdot q$(q 为元素的长度).

本方案只需要参与者将自己的多项式传递给其他参与者,并将收到来自其他参与者的多项式传递给 P_n,共需要传递 $2n$ 次 $2l$ 阶多项式,由于多项式表示的是参与者集合中的元素,所以相当于传递了 $4nl$ 个集合中的元素,若交集中元素的长度为 q,则总的通信开销为 $4nl\cdot q$.因为 N 一般为 2^{1024},而 $q\ll2^{1024}$,所以本方案的通信开销远小于文献[23],同时也小于文献[27]的方案.

Freedman 的方案、文献[27]的方案与本文提出的方案具体比较如表 1.

表 1 多个参与者集合相交协议的比较

	Freedman 方案	文献[27]	本方案
计算开销	$2\log N\cdot l(n-1)(2n+l)$次模 N^2模乘	$4qtn^3l\log(nl)$次普通乘法	$2l\cdot l_n$次普通乘法
通信开销	$2n^2l\cdot\|N^2\|$	$n^3l\log(nl)\cdot q$	$4nl\cdot q$
安全性	语义安全	信息论安全	信息论安全

3.3 改进方案

在上述方案中,将每个参与者拥有的集合用多项式表示时,每个元素表示成多项式的 2 重根,这样可以保证定理 1 成立,但这样比直接将参与者的元素表示成多项式的根的计算量增加了一倍.为了降低计算量,可以将上述方案进行一些修改,改成一种计算量少的方案,具体如协议 2.

协议 2. 多个参与者求交集的安全多方计算.

输入:$P_1,P_2,\cdots,P_n$各自的秘密集合 $X_1,X_2,\cdots,X_n$

输出:$X=X_1\cap X_2\cap\cdots\cap X_n$

1. 每个参与者 $P_i(i=1,2,\cdots,n)$将自己拥有的集合表示成多项式的形式:

   $f_i(x)=x^{(l-l_i)}(x-x_{i1})(x-x_{i2})\cdots(x-x_{il_i})$.

2. 参与者 $P_1,P_2,\cdots,P_{n-1}$的计算如协议 1 中的步骤 2.

① Horner's methord. http://en.wikipedia.org/wiki/Horner%27s_method

3. P_n计算如下：

(a) 先按照协议 1 中步骤 3 的(a)、(b)求出集合 X；

(b) 随机地选择一个集合 $X_r(X_r \cap X=\varnothing)$，令

$$X_n^* = X \cup X_r;$$

(c) 将伪交集 X_n^* 公布.

4. 参与者 $P_i(i=1,\cdots,n-1)$计算：

$$X_i^* \leftarrow X_i \cap X_n^*.$$

5. 参与者 $P_1,\cdots,P_{n-1},P_n$ 根据各自新的秘密集合 $X_1^*,\cdots,X_{n-1}^*,X$ 重复步骤 1、2.

6. P_n得到多项式 $f^*(x)=f_1^*(x)+f_2^*(x)+\cdots+f_n^*(x)$，然后将集合 X 中的每个元素 x_i^* 分别代入 $f^*(x)$中，若 $f^*(x_i^*)=0$，则 x_i^* 是交集 X^* 中的元素.

7. P_n将交集 X^* 公布.

定理 3.　在协议 2 中，参与者 P_n将元素 x_{nj} 代入多项式 $f(x)$中求得的集合 X 不一定是 n 个参与者的交集.

证明.　由于参与者 P_n将元素 x_{nj} 代入 $f(x)$中时，使 $f(x_{nj})=0$ 并不能保证每个多项式 $f_i(x_{nj})=0$，即不能保证 x_{nj} 是交集中的元素，所以此时求出的集合 X 并不是 n 个参与者的交集. 以参与者 A、B、C、D 分别拥有集合$\{2,12\}$，$\{3,8\}$，$\{4,6\}$，$\{5,9\}$为例，他们分别构造自己的多项式

$f_A(x)=(x-2)(x-12)$，$f_B(x)=(x-3)(x-8)$，

$f_C(x)=(x-4)(x-6)$，$f_D(x)=(x-5)(x-9)$，

经协议 2 中的步骤 2，D 得到多项式

$$f(x)=(x-2)(x-12)+(x-3)(x-8)+(x-4)(x-6)+(x-5)(x-9),$$

然后将集合$\{5,9\}$中的元素分别代入 $f(x)$中，得到 $f(9)=0$，但 9 不是交集中的元素. 由此可知，参与者 P_n求得的集合并不一定是所有参与者的交集，但协议 2 经过后面的计算可以保密地求出所有参与者的交集.　　证毕.

正确性分析.

由定理 3 可知，协议 2 中求出的集合 X 不一定是所有参与者的交集，如果求 n 个参与者的交集，仍需继续计算. 参与者 P_n将随机数集合 $X_r(X_r \cap X=\varnothing)$加入集合 X 中组成集合 X_n^*，然后将 X_n^* 公布. 由于 P_n加入了集合 X_r，P_n不会泄露集合 X_n中的信息，参与者 $P_1,\cdots,P_{n-1},P_n$分别用集合 $X_1^*,\cdots,X_{n-1}^*,X$ 中的元素构造自己的秘密多项式，经协议 2 中的步骤 2 计算，P_n得到多项式$f^*(x)$，P_n将集合 X 中的元素代入 $f^*(x)$中求得交集 X^*. 由于 $|X_n^*|\le l$，伪交集 X_n^* 不可能包含所有的参与者集合中的元素，新组成的多项式 $f^*(x)\neq f(x)$，集合 X 中除所有参与者交集中的元素外其他元素不能保证 $f^*(x)=0$，所以集合 X^* 是 $P_1,P_2,\cdots,P_n$的交集.

安全性分析.

由协议 1 的安全性可知，参与者 $P_1,P_2,\cdots,P_n$ 在 P_n求出 X 之前与在步骤 5、6、7 中是安全的，在此省略该部分证明. 由于集合 X 是 P_n将自己的集合 X_n代入求得的，P_n为了保护自己集合 X_n的隐私性，随机地选取集合 $X_r(X_r \cap X=\varnothing)$，然后令 $X_n^*=X\cup X_r$并将 X_n^* 公布. 由于集合 X_r的加入，参与者 $P_i(i=1,\cdots,n-1)$不知道 X_n^* 中哪些元素是 X_n中的，所以 P_n公布 X_n^* 是安全的. 所以协议 2 是安全的.

效率分析.

协议 2 的方案和协议 1 中的方案忽略构造多项式的开销. 协议 1 中的方案，需要进行 $2l\cdot l_n$次普通乘法运算，需要进行 $2n$ 轮通信. 协议 2 的方案中，在求出 X 之前，P_n将拥有集合 X_n中的 l_n个元素代入多项式 $f(x)$中进行计算，每个元素需要做 l 次乘法运算，共需要做 $l\cdot l_n$次乘法运算；在求出 X 到求出最终的交集 X^* 之间中，若 X 中有 l_n^* 个元素，每个参与者构造的多项式为 l^* 阶，需要做 l^* 次乘法运算，协议 2 共需要做 $l_n l+l_n^* l^*$ 次乘法运算. 在求出集合 X 之前需要进行 $2n$ 轮通信，在求出集合 X 之后仍需要 $2n$ 轮通信，则共需要 $4n$ 轮通信. 协议 1 与协议 2 的比较如表 2.

表 2　协议 1 与协议 2 的比较

	协议 1	协议 2
计算开销	$2l\cdot l_n$	$l_n l+l_n^* l^*$
通信轮数	$2n$	$4n$

密码学中通信复杂性和计算复杂性往往是一对矛盾的问题，经常为了提高某一性能而牺牲另一性能，我们可以根据具体的系统要求选择具体的算法. 协议 1 的计算量几乎是协议 2 的 2 倍，但协议 1 的通信轮数是协议 2 的 1/2. 所以协议 1 适用于对通信要求较高但计算能力较强的网络，如无线传感器网络等，而协议 2 适用于通信能力较强而计算能力相对较弱的网络，如一般用户的有线网络.

4　两个参与者保密计算交集

在 Freedman 的方案中，两个参与者拥有的集合取自于一个无限大的集合，且主要基于 Paillier 的公钥加密算法. 一方面，由于 Paillier 的公钥加密算法效率非常低，不适合在实际场合中使用. 另一方面，在实际的计算中，参与者拥有的集合很可能不是

取自于一个无限大的集合,而是有限集合.所以本文针对两个参与者的情况,根据集合取值范围的不同设计了两种方案.

4.1 适用于无限集合子集的交集方案

由于单向散列函数 Hash 具有如下性质:

(1)给定消息 M,很容易计算 $h=\mathrm{Hash}(M)$.

(2)给定 $h=\mathrm{Hash}(M)$,根据 h 计算其逆 $M=\mathrm{Hash}^{-1}(M)$非常困难.

(3)给定M,要找到另一个消息M',使得$\mathrm{Hash}(M)=\mathrm{Hash}(M')$很难.

(4)找出两个随机的消息M,M',使得$\mathrm{Hash}(M)=\mathrm{Hash}(M')$是困难的.

(5)如果对 M 作微小的改变,即使只改变一个比特为 M',$\mathrm{Hash}(M')$与 $\mathrm{Hash}(M)$相比也会发生惊人的改变,至少改变一半位数的值.

对于 Freedman 方案中适合的场景,即对于两个参与者 Alice 和 Bob 的集合 A 和 B 取自于一个无限集合的情况,本文利用单向散列函数的性质,设计如下协议.

协议 3. 求两个无限集合子集的交集.

输入:Alice 输入:$A=\{a_1,a_2,\cdots,a_m\}$,Bob 输入:$B=\{b_1,b_2,\cdots,b_n\}$

输出:$C=A\cap B$

1. Alice 和 Bob 共同协商一个单向散列函数 $\mathrm{Hash}(x)$.
2. Alice 将自己的集合 A 中的每个元素代入 $\mathrm{Hash}(x)$中,求出单向散列函数值,记为 $\mathrm{Hash}(A)$,$\mathrm{Hash}(A)=\{\mathrm{Hash}(a_1),\mathrm{Hash}(a_2),\cdots,\mathrm{Hash}(a_m)\}$,并将 $\mathrm{Hash}(A)$发送给 Bob.
3. Bob 计算如下:
   (a) SETUP: $C\leftarrow\varnothing$.
   (b) FOR $j=1$ to n {
   computes $\mathrm{Hash}(b_j)$;
   IF $\mathrm{Hash}(b_j)\in\mathrm{Hash}(A)$;
   $C\leftarrow C\cup\{b_j\}$;
   }
   (c) OUTPUTS: C.

正确性分析.

协议中的单向散列函数为 $\mathrm{Hash}(x):\{0,1\}^*\rightarrow\{0,1\}^w$,将无限大取值范围的数映射到长度为 w 的值.单项散列函数会出现单边性错误,即如果 $a_i=b_j$,协议 3 绝对不会给出 $\mathrm{Hash}(a_i)\neq\mathrm{Hash}(b_j)$,不会出现错误结论;但是 $\mathrm{Hash}(a_i)=\mathrm{Hash}(b_j)$不能保证 $a_i=b_j$绝对成立,因为无限大取值范围的数经置换映射到长度为 w 的值,会出现 $\mathrm{Hash}(a_i)=\mathrm{Hash}(b_j)$而 $a_i\neq b_j$的情况,但该协议出现这种错误的概率非常小.现有的单向散列函数值一般为 128 位的二进制数,即 $w=128$,那么 $\mathrm{Hash}(a_i)=\mathrm{Hash}(b_j)$而 $a_i\neq b_j$ 的概率为

$$\Pr[(\mathrm{Hash}(a_i)=\mathrm{Hash}(b_j))\wedge(a_i\neq b_j)]<2^{-128},$$

即协议 3 给出的解决方案出错的概率小于 2^{-128},这在科学计算中可以忽略不计,所以该方案是正确的.

安全性分析.

根据单项散列函数的性质,对于任何有多项式时间计算能力的敌手来说,即使知道 $\mathrm{Hash}(x)$但要求出 x 仍然是困难的,所以对于 Alice 的集合 A,无论对于内部参与者 Bob 还是其他的外部敌手都是安全的,所以 Alice 是安全的.Bob 的集合在整个求交集的过程中没有和任何人进行交互,只有他自己知道,所以 Bob 是安全的,从而协议 3 是安全的.

效率分析.

在 Freedman 的方案中,两个参与者需要使用 Paillier 公钥加密算法进行计算,他们通过使用平衡哈希函数实现计算和通信的并行执行,这样只是提高了计算效率和通信效率,但是总的计算量和通信量并没有降低.本方案利用单向散列函数的性质.首先,Alice 和 Bob 使用的单向散列函数可以公开,在协商单向散列函数时不需要使用加密算法.其次,Alice 在计算自己集合中元素的单向散列函数值后发送给 Bob 的这个过程中也不需要进行加密.最后,整个协议只需要一轮通信,通信复杂度已经达到最低.所以本方案在计算开销和通信轮数都达到了最优,至少接近最优.Freedman 两个参与者的方案与协议 3 的比较结果如表 3.

表 3 适用于无限集合子集相交协议的比较

	Freedman 方案	协议 3
计算开销	$2(m+n+mn)\log N$ 次模 N^2 运算	$(m+n)$次单向散列函数求值
通信轮数	2	1

该方案中,如果 Alice 和 Bob 集合中的元素取自于一个有限集合,Bob 可以利用穷举攻击的方法获得 Alice 的秘密集合 A,即分别计算集合元素取值范围内所有元素关于单向散列函数 $\mathrm{Hash}(x)$的值,然后和 Alice 发给 Bob 的集合 $\mathrm{Hash}(A)$中的元素对比,进而求出集合 A.所以该方案只适用于参与者集合中的元素取自于无限集合的情况.

4.2 适用于有限集合子集的交集方案

在实际应用中,保密求两个集合的交集时,有时参与者的集合并不是无限集合的子集,而是某个有限集合的子集,对于这类问题还没有看到该问题的

解决方案. 为了解决 Alice 和 Bob 集合中的元素取自于有限集合的问题，本文根据 Diffie-Hellman[31] 的密钥分配方案基于离散对数困难性设计了一个解决方案，具体如协议 4.

协议 4.　求适用于有限集合子集的交集.

输入：Alice 的集合 $A=\{a_1,a_2,\cdots,a_m\}$，Bob 的集合 $B=\{b_1,b_2,\cdots,b_n\}$，$A,B\subseteq D$

输出：$C=A\cap B$

1. Alice 和 Bob 共同确定一个大素数 p.
2. Alice 选择一个大的随机数 r_1，对元素 a_i（$i=1,2,\cdots,m$）计算其相应的离散对数，即
$$y_{a_i}=a_i^{r_1} \bmod p,$$
然后将 $Y_A=\{y_{a_1},y_{a_2},\cdots,y_{a_m}\}$ 发送给 Bob.
3. Bob 选择一个大的随机数 r_2，对元素 b_j（$j=1,2,\cdots,n$）计算其相应的离散对数，即
$$y_{b_j}=b_j^{r_2} \bmod p,$$
然后将 $Y_B=\{y_{b_1},y_{b_2},\cdots,y_{b_n}\}$ 发送给 Alice，同时记下集合 Y_B 中元素与集合 B 中元素的对应关系.
4. Alice 对 Bob 集合 Y_B 中的元素 y_{b_j} 计算
$$y_{b_j}^*=y_{b_j}^{r_1} \bmod p,$$
然后按照 Bob 发给她的集合顺序将 $Y_B^*=\{y_{b_1}^*,y_{b_2}^*,\cdots,y_{b_n}^*\}$ 发送给 Bob.
5. Bob 对 Alice 集合 Y_A 中的元素 y_{a_i} 计算
$$y_{a_i}^*=y_{a_i}^{r_2} \bmod p,$$
得到集合 $Y_A^*=\{y_{a_1}^*,y_{a_2}^*,\cdots,y_{a_m}^*\}$，Bob 求出 $Y=Y_A^*\cap Y_B^*$，然后根据 Y 中元素在 Y_B^* 中的位置，找出交集中的元素，进而求出交集 C.
6. Bob 将交集 C 公布.

正确性分析.

本协议中，因为集合 Y_B^* 中的元素为
$$y_{b_j}^*=y_{b_j}^{r_1} \bmod p=b_j^{r_1r_2} \bmod p,$$
集合 Y_A^* 中的元素为
$$y_{a_i}^*=y_{a_i}^{r_2} \bmod p=a_i^{r_1r_2} \bmod p,$$
所以当 $a_i=b_j$ 时，则 $y_{a_i}^*=y_{b_j}^*$. 由于 Alice 和 Bob 集合中的元素取自于一个有限集合 D，同时 Alice 和 Bob 选择大素数 p 作模数，所以各个元素在经过模 p 运算后出现重复的概率足够小，可以忽略. Bob 在将 Y_B 发送给 Alice 时记下了自己集合 B 中的元素与 Y_B 中的元素的对应关系，Alice 对 Y_B 中的每个元素加入 r_1 后，将 Y_B^* 中的元素仍然保持其在 Y_B 中的顺序发送给 Bob，则 Bob 仍知道自己集合 B 中的元素在集合 Y_B^* 中的对应位置，所以 Bob 可以根据 Y_B^* 与 Y_A^* 相等的元素找到其与 Alice 交集中的元素.

安全性分析.

在本协议中，Alice 对自己集合中的元素 a_i 计算 $y_{a_i}=a_i^{r_1} \bmod p$，则对于有限计算能力的敌手 $\mathcal{B}$ 来说无法根据 y_{a_i} 计算出 a_i；同时由于 r_1 是 Alice 随机选择的随机数，任何人也无法通过将元素 a_i 取值范围内的元素一个个代入进行穷举攻击，所以 Alice 将 Y_A 发送给 Bob 是安全的. 与 Alice 相同，Bob 将 Y_B 发送给 Alice 也不会泄露自己集合内元素的任何信息. 由离散对数困难性假设可知，对于等式 $y=\alpha^x \bmod p$，即使知道 y,α,p，求 x，对于有概率多项式时间计算能力的敌手 $\mathcal{A}$ 来说仍然是困难性问题. 任意概率多项式时间计算能力的敌手 $\mathcal{A}$ 根据 y_{b_j} 和 $y_{b_j}^*$ 都无法求出 r_1，Alice 将计算的
$$y_{b_j}^*=y_{b_j}^{r_1} \bmod p$$
发送给 Bob 是安全的. 所以协议 4 是安全的.

效率分析.

该协议中，Alice 将自己集合的元素计算随机数 r_1 次方后发送给 Bob，需要做 m 次指数运算，同时 Bob 将自己拥有的元素计算随机数 r_2 次方后发送给 Alice，需要做 n 次指数运算. 然后 Alice 将收到来自 Bob 集合中的元素计算 r_1 次方后发送给 Bob，同时 Bob 也将 Alice 第一次发送过来的集合中的元素计算 r_2 次方，Alice 和 Bob 分别需要做 n,m 次指数运算. 整个过程中，Alice 做了 $m+n$ 次指数运算，Bob 做了 $m+n$ 次指数运算，所以在该协议中 Alice 和 Bob 共做了 $2(m+n)$ 次指数运算，进行了 3 轮的通信. 该协议中计算复杂性与元素个数成线性关系，通信轮数为常数，所以该方案的效率比较高.

协议的变形可保密计算两个集合交集的势.

集合 A 与集合 B 交集的势为 A 与 B 交集中元素的个数，记为 $|A\cap B|$. 保密求集合交集的势可应用到隐私匹配[32]、语义定位[33]、对样本集相似度的评价[34]、隐私保护的基因测试[35]等问题中，同时也可以在现实生活中得到很好的应用. 如 Alice 公司和 Bob 公司是合作公司，现在都想要扩大在一个地区的投资范围，各自拟定了计划扩大投资的城市，但是为了避免竞争，Alice、Bob 公司不想在同一城市同时扩大投资，为了保证各自公司的利益，他们不想让对方知道各自计划投资的城市，他们可以将这一地区的所有城市编码构成集合 D，若 Alice 公司将计划投资的城市集合为 A，Bob 公司将计划投资城市的集合为 B，则 $A,B\subset D$，然后求 A,B 两个集合是否有交集，即求 $|A\cap B|$ 是否为零，若 $|A\cap B|=0$，说明两个公司可以执行各自的计划，否则说明两个公司的计划有冲突，需要修改各自的计划.

协议 4 经过简单的改造可以实现保密求集合 A 与集合 B 交集的势，即 Alice 和 Bob 只知道 $|A\cap B|$，而对于两个集合内具体有哪些元素相等没有任何信

息.在协议 4 中,对 Alice 的行为做微小的改变可以实现保密求集合交集的势.在协议 4 的步骤 4 中,Alice 收到 Bob 发送过来的集合 Y_B 时,为每个元素加入随机数后,不直接将集合 Y_B^* 发送给 Bob,而是将 Y_B^* 中的每个元素进行随机置换后再发送给 Bob,这样 Bob 将对集合 B 和集合 Y_B^* 中元素的对应关系不再有任何信息.所以 Bob 仅可求出 $|Y_B^* \cap Y_A^*|$,但却得不到与交集中元素的原值的相关任何信息,同时该方案与协议 4 中的方案相比没有计算量的增加.

对于保密计算集合交集势的方案,潜在的可以保密的计算集合并集的势.因为集合 A 与集合 B 并集的势可用如下形式表示:

$$|A \cup B| = |A| + |B| - |A \cap B|,$$

所以协议 4 经改造过的保密求集合交集势的协议可以用来实现求集合并集的势.Bob 根据收到 Alice 发送过来的集合 Y_A 可以知道 Alice 拥有集合的势 $|A|$,同时 Bob 知道自己集合的势 $|B|$,所以 Bob 求出 $|A \cap B|$ 之后可以得到 $|A \cup B|$.

4.3 认证的集合交集的保密计算

根据 RSA 签名算法[36]协议 4 可以改造成认证的保密求集合交集协议.如果协议 4 中 Alice 和 Bob 对彼此不信任,担心对方给出的集合不是自己拥有的,而是随意给出的,但 Alice 对 Bob 的上级 Fred 信任,而 Bob 对 Alice 的上司 Ella 信任,他们在求集合交集之前 Alice、Bob 分别需要 Ella、Fred 对集合 A、集合 B 进行签名认证.Alice 和 Bob 经认证的保密求集合交集的协议具体如协议 5.

协议 5. 求两个参与者认证的集合交集.

输入:Alice 经认证的集合 $A=\{(a_1,\sigma_1),\cdots,(a_m,\sigma_m)\}$,Bob 经认证的集合 $B=\{(b_1,\delta_1),\cdots,(b_n,\delta_n)\}$.其中 $\sigma_i=a_i^{d_E}$ $(i=1,\cdots,m)$,$\delta_j=b_j^{d_F}$ $(j=1,\cdots,n)$ 分别为 Ella 和 Fred 相应的签名

输出:$C=A \cap B$

1. Alice 和 Bob 共同协商确定一个大素数 p.
2. Alice 选择一个大随机数 r_1,对元素 σ_i $(i=1,2,\cdots,m)$ 计算

$$y_{a_i}=\sigma_i^{r_1} \bmod p,$$

然后将 $Y_A=\{y_{a_1},y_{a_2},\cdots,y_{a_m}\}$ 发送给 Bob.

3. Bob 选择一个大的随机数 r_2,对元素 δ_j $(j=1,2,\cdots,n)$ 计算

$$y_{b_j}=\delta_j^{r_2} \bmod p,$$

然后将 $Y_B=\{y_{b_1},y_{b_2},\cdots,y_{b_n}\}$ 发送给 Alice,同时记下集合 Y_B 中的元素与集合 B 中元素的对应关系.

4. Alice 对 Bob 集合 Y_B 中的每个元素 y_{b_j} 计算

$$y_{b_j}^*=y_{b_j}^{e_F r_1} \bmod p,$$

然后按照 Bob 发给她的集合顺序将 $Y_B^*=\{y_{b_1}^*,y_{b_2}^*,\cdots,y_{b_n}^*\}$ 发送给 Bob.

5. Bob 对 Alice 集合 Y_A 中的每个元素 y_{a_i} 计算

$$y_{a_i}^*=y_{a_i}^{e_E r_2} \bmod p,$$

得到集合 $Y_A^*=\{y_{a_1}^*,y_{a_2}^*,\cdots,y_{a_m}^*\}$,Bob 求出 $Y=Y_A^* \cap Y_B^*$,然后根据 Y 中元素在 Y_B^* 中的位置,找出交集中的元素,进而求出交集 C.Bob 将交集 C 公布.

正确性分析.

本协议中,因为集合 Y_B^* 中的元素为

$$y_{b_j}^*=y_{b_j}^{e_F r_1}=\delta_{b_j}^{e_E r_1 r_2}=b_j^{r_1 r_2},$$

集合 Y_A^* 中的元素为

$$y_{a_i}^*=y_{a_i}^{e_E r_2}=\delta_{a_i}^{e_E r_1 r_2}=a_i^{r_1 r_2},$$

当 $a_i=b_j$ 时,$y_{a_i}^*=y_{b_j}^*$.所以协议 5 的方案是正确的.

根据协议 4,该协议的安全性和效率很容易证明分析,在此省略协议 5 的证明分析.与协议 4 相同,协议 5 经改造后也可以实现保密地求认证集合并集的势和认证集合交集的势.

5 保密求多个数的最大公约数

假设有 n 个参与者 $P_1,P_2,\cdots,P_n$,各自拥有秘密整数 $x_1,x_2,\cdots,x_n$,他们想知道共同的最大公约数 x,但又不想泄露自己拥有的秘密数.求多个数的最大公约数不仅可以应用到隐私保护数据挖掘等密码学问题,而且可以在现实生活中实现原材料或花费等问题的最优化.

参与者 $P_1,\cdots,P_n$ 将各自的秘密数 $x_1,\cdots,x_n$ 用算术基本定理表示:

$$x_1=p_1^{e_{11}} p_2^{e_{12}} \cdots p_m^{e_{1m}},$$

$$\cdots,$$

$$x_n=p_1^{e_{n1}} p_2^{e_{n2}} \cdots p_m^{e_{nm}},$$

每个等式中的 $p_1,p_2,\cdots,p_m$ 分别代表第 1 个,第 2 个,…,第 m 个素数,即素数 2,3,5,….

每个参与者 P_i $(i=1,2,\cdots,n)$ 根据自己拥有的秘密数构造多项式 $f_i(x)$,即构造 $p_1,p_2,\cdots,p_m$ 的 $2e_{i1},2e_{i2},\cdots,2e_{im}$ 重根多项式 $f_i(x)$,

$$f_i(x)=(x-p_1)^{2e_{i1}}(x-p_2)^{2e_{i2}}\cdots(x-p_m)^{2e_{im}}.$$

参与者 P_i $(i=1,2,\cdots,n-1)$ 将自己的多项式整理成如下形式:

$$f_i(x)=\sum_{q=0}^{l_i} a_{iq} x^q,$$

然后将其随机地分成非零的 k $(k \le n)$ 份,即 $f_{i1}(x)$,$f_{i2}(x)$,…,$f_{ik}(x)$,使

$$f_i(x)=f_{i1}(x)+f_{i2}(x)+\cdots+f_{ik}(x).$$

与上述多个参与者求集合交集的情况类似，为了增加安全性，为多项式 $f_i(x)$ 的每份多项式 $f_{ij}(x)$ $(j=1,2,\cdots,k)$ 加上一个随机数 r_{ij}，并且使

$$r_{i1}+r_{i2}+\cdots+r_{ik}=0.$$

P_i 将自己 k 份加入随机数的多项式 $f_{ij}(x)+r_{ij}$ $(j=1,2,\cdots,k)$ 分别发送给 n 个参与者中的 k 个参与者（这 k 个参与者可以包括 P_i 自己，也可以不包括，其他人不知道 P_i 分发的 k 份包括不包括自己，即这个 k 是变的），其他参与者不知道具体分发给了几个参与者. 当每个参与者 P_i 将其多项式都分发后，参与者 P_i 将其收到的所有多项式份额相加，组成新的多项式 $g_i(x)$，P_i 将多项式 $g_i(x)$ 发送给参与者 P_n.

P_n 将收到的多项式加上自己的多项式组成新的多项式 $h_0(x)$，即

$$h_0(x)=g_1(x)+\cdots+g_n(x).$$

因为每个参与者都是半诚实的，会严格按照协议执行，每个参与者发送出去的多项式和收到的多项式都是来自 n 个参与者内部，且加入的随机数之和为 0，所以

$$h_0(x)=f_1(x)+\cdots+f_n(x),$$

即 $h_0(x)$ 是 n 个参与者的多项式之和.

如果在 P_n 的秘密数中，素数底 p_s $(s=1,2,\cdots,m)$ 对应的指数 $e_{ns}=0$，说明 n 个参与者没有关于 p_s 的公因子，否则 n 个参与者可能存在关于 p_s 的公因子. P_n 将 p_s 代入多项式 $h_0(x)$ 中，若 $h_0(p_s)\neq 0$，说明 p_s 不是所有参与者的一个公因子，且以素数 p_s 为底的数 $p_s^2,p_s^3,\cdots,p_s^{e_{ns}}$ 都不是参与者的公因子，不需要继续计算关于 p_s 的公因子. 如果 $h_0(p_s)=0$，说明 p_s 是所有参与者的一个公因子，将 p_s 记为多重集 X 中的一个元素，然后令

$$h_1(x)=h_0(x)/(x-p_s)^2.$$

P_n 将 p_s 代入多项式 $h_1(x)$ 中，若 $h_1(p_s)\neq 0$，则 $h_1(x)$ 中没有 n 个参与者关于 p_s 的公因子，否则说明 $h_1(x)$ 中有公因子 p_s，并将 p_s 加入多重集 X 中. 由于 n 个参与者的公因子是每个参与者的因子，依次类推，直到 $h_t(p_s)\neq 0$ 或 $t>e_{ns}$，此时多重集 X 中有 $t-1$ 个 p_s，说明所有参与者的整数中含有 $t-1$ 个公因子 p_s.

重复此过程，直到找到所有参与者关于每个素数底 p_s 的公因子并放入多重集 X 中，然后将 X 中的所有元素相乘，乘积则为所有参与者的最大公约数 x. P_n 将所有参与者的最大公约数 x 公布. 多个参与者求最大公约数的安全多方计算协议如下.

协议 6. 多个参与者求最大公约数的安全多方计算.

输入：$P_1,P_2,\cdots,P_n$ 各自拥有的秘密整数 $x_1,x_2,\cdots,x_n$

输出：$x=\gcd(x_1,x_2,\cdots,x_n)$

1. 每个参与者 P_i $(i=1,2,\cdots,n)$ 将各自的秘密数 x_i 用算术基本定理表示，并构造 $p_1,p_2,\cdots,p_m$ 的 $2e_{i1},2e_{i2},\cdots,2e_{im}$ 重根多项式 $f_i(x)$.
2. 参与者 P_i $(i=1,2,\cdots,n-1)$ 计算如下：
   (a) 将自己的多项式分成 k 份，并为每份加入随机数，然后随机地将 k 份多项式发送给 n 个参与者中的 k 个.
   (b) 将自己收到的多项式份额相加，组成一个新的多项式 $g_i(x)$ 并发送给 P_n.
3. 参与者 P_n 进行如下操作：
   (a) SETUP: $X\leftarrow\varnothing$,
   $h_0(x)\leftarrow g_1(x)+g_2(x)+\cdots+g_n(x)$
   $=f_1(x)+f_2(x)+\cdots+f_n(x)$.
   (b) FOR ($s=1$ to m) {
   IF ($e_{ns}\neq 0$) {
   FOR $j=0$ to $t(t<e_{ns})$ {
   computes $h_j(p_s)$;
   IF $h_j(p_s)=0$ {
   $X\leftarrow X\cup\{p_s\}$;
   $h_{j+1}(x)\leftarrow h_j(x)/(x-p_s)^2$; }
   } } }
   (c) $x=\prod_{p_s\in X}p_s$;
   OUTPUTS: x.

定理 4. 计算多个参与者最大公约数的协议 6（记为 Π）是保密的.

证明. 通过构造满足

$$\{(S(x_i,f_i(\bar{x})),f(\bar{x}))\}\overset{c}{\equiv}\{(view_i^{\Pi}(\bar{x}),output^{\Pi}(\bar{x}))\}$$

的模拟器 S 来证明本定理. S 工作如下：

(1) 给定输入

$$(x_i,P(x_i,(x_1,\cdots,x_{i-1},x_{i+1},\cdots,x_n))),$$

S 随机地选择 $x_1',\cdots,x_{i-1}',x_{i+1}',\cdots,x_n'$ 使得

$$P(x_i,(x_1,\cdots,x_{i-1},x_{i+1},\cdots,x_n))=$$
$$P(x_i,(x_1',\cdots,x_{i-1}',x_{i+1}',\cdots,x_n')) \quad (i)$$

用 $x_1',\cdots,x_{i-1}',x_{i+1}',\cdots,x_n'$ 进行模拟 $x_1,\cdots,x_{i-1},x_{i+1},\cdots,x_n$.

(2) 首先按照协议 Π 根据算术基本定理构造多项式

$$f_i(\bar{x}),f_1(\bar{x})',\cdots,f_{i-1}(\bar{x})',f_{i+1}(\bar{x})',\cdots,f_n(\bar{x})'.$$

每个多项式随机地分成 k 份，并为每份加入随机数，然后随机地将 k 份多项式发送给 n 个参与者中的 k 个.

(3) 在协议 Π 中，

$$view_i^{\Pi}(\bar{x})=view_i^{\Pi}(x_i,(x_1,\cdots,x_{i-1},x_{i+1},\cdots,x_n),\bar{x})$$
$$=\{x_i,f_i(\bar{x}),(f_{i1}^*(\bar{x}),\cdots,f_{i(i-1)}^*(\bar{x}),$$
$$f_{i(i+1)}^*(\bar{x}),f_{in}^*(\bar{x})),\bar{x}\},$$

其中 $f_{ij}^*(\bar{x})(j=1,\cdots,i-1,i+1,\cdots,n)$表示 P_i从参与者 $P_1,\cdots,P_{i-1},P_{i+1},\cdots,P_n$得到的多项式,令

$$S(x_i,f_i(\bar{x}))=\{x_i,f_i(\bar{x}),(f_{i1}^{*\prime}(\bar{x}),\cdots,f_{i(i-1)}^{*\prime}(\bar{x}),$$
$$f_{i(i+1)}^{*\prime}(\bar{x}),f_{in}^{*\prime}(\bar{x})),\bar{x}'\},$$

因为等式(i)成立,根据 $\bar{x}$的计算方法可知 $\bar{x}=\bar{x}'$,即

$$(f_{i1}^*(\bar{x}),\cdots,f_{i(i-1)}^*(\bar{x}),f_{i(i+1)}^*(\bar{x}),f_{in}^*(\bar{x}))=$$
$$(f_{i1}^{*\prime}(\bar{x}),\cdots,f_{i(i-1)}^{*\prime}(\bar{x}),f_{i(i+1)}^{*\prime}(\bar{x}),f_{in}^{*\prime}(\bar{x})),$$

所以

$$\{(S(x_i,f_i(\bar{x})),f(\bar{x}))\}\overset{c}{\equiv}\{(view_i^{\Pi}(\bar{x}),output^{\Pi}(\bar{x}))\}$$

成立.则可证协议 6 求多个参与者的最大公因数是安全的. 证毕.

6 结 语

保密求集合相交问题是安全多方计算的一个重要模块,现有的解决方案都是基于某种密码学原语,其计算复杂性和通信复杂性都比较高,且主要是针对于两个参与者的集合是一个无限大集合子集的情况进行的研究.本文针对多个参与者的情况,利用多项式的性质和回路分区的思想,提出了一种信息论安全、不需要借助于密码学原语、高效且适用范围广泛地求集合相交保密计算协议.但新协议不适用于两个参与者的情况,本文又给出了两个参与者的集合是无限集合的子集和有限集合的子集的情况.最后,在新协议的基础上,本文解决了多个参与者保密求最大公约数的问题.本文提出的保密求集合交集方案虽然高效,但是均有各自的适用范围,没有设计出普遍适用的协议,有待进一步深入研究,来给出普遍适用的集合交集的高效保密计算方案.本文设计的最大公约数的保密计算,当私有数据的素因子很大时,效率较低,探索更高效的保密计算算法也值得进一步研究.

参 考 文 献

[1] Yao A C. Protocols for secure computations//Proceedings of the 23rd IEEE Symposiumon Foundations of Computer Science. Chicago, USA, 1982: 160-164

[2] Ben-Or M, Goldwasser S, Wigderson A. Completeness theorems for non-cryptographic fault-tolerant distributed computation// Proceedings of the 20th Annual ACM Symposium on Theory of Computing. Chicago, USA, 1988: 1-10

[3] Goldreich O, Micali S, Wigderson A. How to play any mental game//Proceedings of the 19th Annual ACM Conference on Theory of Computing. New York, USA, 1987: 218-229

[4] Goldreich O. Foundations of Cryptography: Basic Applications. Volume 2. London, England: Cambridge University Press, 2009

[5] Brassard G, Chaum D, Crépeau C. Minimum disclosure proofs of knowledge. Journal of Computer and System Sciences, 1988, 37(2): 156-189

[6] Goldwasser S, Micali S, Rackoff C. The knowledge complexity of interactive proof systems. SIAM Journal on Computing, 1989, 18(1): 186-208

[7] Goldwasser S. Multi-party computations: Past and present// Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing. Santa Barbara, USA, 1997: 1-6

[8] Prabhakaran M M, Sahai A. Secure Multi-Party Computation. Amsterdam, The Netherlands: IOS Press, 2013

[9] Li Shun-Dong, Wang Dao-Shun. Efficient secure multiparty computation based on homomorphic encryption. Acta Electronica Sinica, 2013, 41(4): 798-803(in Chinese)
(李顺东, 王道顺. 基于同态加密的高效多方保密计算. 电子学报, 2013, 41(4): 798-803)

[10] Shundong L, Chunying W, Daoshun W, et al. Secure multi-party computation of solid geometric problems and their applications. Information Sciences, 2014, 282(20): 401-413

[11] Kantardzic M. Data Mining: Concepts, Models, Methods, and Algorithms. Hoboken, USA: John Wiley & Sons, 2011

[12] Liu F, Ng W K, Zhang W, et al. Encrypted set intersection protocol for outsourced datasets//Proceedings of the 2014 IEEE International Conference on Cloud Engineering. Boston, USA, 2014: 135-140

[13] Drosatos G, Efraimidis P S. Privacy-preserving statistical analysis on ubiquitous health data//Furnell S, Lambrinoudakis C, Pernul G eds. Trust, Privacy and Security in Digital Business 2011. Heidelberg, Germany: Springer, 2011: 24-36

[14] De Cristofaro E, Tsudik G. Practical private set intersection protocols with linear complexity//Sion R eds. Financial Cryptography and Data Security 2010. Heidelberg, Germany: Springer, 2010: 143-159

[15] Kissner L, Song D. Privacy-preserving set operations// Proceedings of the 25th Annual International Cryptology Conference. Santa Barbara, USA, 2005: 241-257

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

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

[18] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory, 1985, 31(4): 469-472

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

[20] Li Shun-Dong, Dou Jia-Wei, Jia Xiao-Lin. Secure two-party computation for set intersection problem. Journal of Xi'an Jiaotong University, 2006, 40(10): 1091-1093(in Chinese)
(李顺东，窦家维，贾晓林. 集合相交问题的双方保密计算. 西安交通大学学报，2006，40(10)：1091-1093)

[21] Li R, Wu C. An unconditionally secure protocol for multi-party set intersection//Proceedings of the 5th International Conference on the Applied Cryptography and Network Security. Zhuhai, China, 2007: 226-236

[22] Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612-613

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

[24] Patra A, Choudhary A, Rangan C P. Round efficient unconditionally secure MPC and multiparty set intersection with optimal resilience//Proceedings of the 10th International Conference on Cryptology in India: Progress in Cryptology New Delhi, India, 2009: 398-417

[25] Cheon J H, Jarecki S, Seo J H. Multi-party privacy-preserving set intersection with quasi-linear complexity. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2012, 95(8): 1366-1378

[26] Patra A, Choudhary A, Rangan C P. Information theoretically secure multi party set intersection re-visited//Proceedings of the 16th Annual International Workshop on the Selected Areas in Cryptography. Calgary, Canada, 2009: 71-91

[27] Blanton M, Aguiar E. Private and oblivious set and multiset operations. International Journal of Information Security, 2016, 15(5): 493-518

[28] Shannon C E. Communication theory of secrecy systems. Bell System Technical Journal, 1949, 28(4): 656-715

[29] Urabe S, Wang J, Kodama E, Takata T. A high collusion-resistant approach to distributed privacy-preserving data mining. Information and Media Technologies, 2007, 2(3): 821-834

[30] Liu W, Wang Y B, Jiang Z T, et al. A protocol for the quantum private comparison of equality with χ-type state. International Journal of Theoretical Physics, 2012, 51(1): 69-77

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

[32] Narayanan G S, Aishwarya T, Agrawal A, et al. Multi party distributed private matching, set disjointness and cardinality of set intersection with information theoretic security//Proceedings of the 8th International Conference Cryptology and Network Security. Kanazawa, Japan, 2009: 21-40

[33] Destino G, Macagnano D. Semantic positioning via structured sparsity models//Proceedings of the 2014 IEEE World Forum on Internet of Things(WF-IoT). Seoul, South Korea, 2014: 106-110

[34] Blundo C, De Cristofaro E, Gasti P. EsPRESSo: Efficient privacy-preserving evaluation of sample set similarity. Journal of Computer Security, 2014, 22(3): 355-381

[35] Baldi P, Baronio R, De Cristofaro E, et al. Countering GATTACA: Efficient and secure testing of fully-sequenced human genomes//Proceedings of the 18th ACM conference on Computer and communications security. Chicago, USA, 2011: 691-702

[36] Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 1978, 21(2): 120-126

ZHOU Su-Fang, born in 1990, Ph. D. candidate. Her research interests is information security.

LI Shun-Dong, born in 1963, Ph. D., professor. His research interests include cryptography and information security.

GUO Yi-Min, born in 1992, Ph. D. candidate. Her research interests focus on information security.

DOU Jia-Wei, born in 1963, Ph.D., associate professor. Her main research interests include application mathematics and application of cryptograpy.

CHEN Zhen-Hua, born in 1976, Ph. D., associate professor. Her research interests focus on information security.

Background

Secure multi-party computation (SMC) is one of the most important research fields studied by the international cryptographic community. Secure set intersection is an important aspect of SMC and has many applications in such

fields as privacy-preserving data mining, data outsourcing, medical data analysis and private data sharing.

Existing protocols for secure set operation are based on several cryptographic primitives. Freedman posited an interesting and relatively efficient solution to the set intersection problem using Paillier's homomorphic encryption scheme. His solution mainly works when there are two parties and when private sets are subsets of an exponentially large set. Even in cases where it can be successfully applied, the implementation of this solution is inefficient. Furthermore, the cases where the private sets are subsets of an infinite set are not very common in practical applications.

We present several efficient solutions to the set intersection problem. Using properties of polynomials, we design an information-theoretically secure and efficient multiparty set intersection computation protocol. The protocol does not rely on any public key cryptosystems. The parties only perform ordinary multiplication. Protocol 1 is efficacious when there are more than two parties and when the private sets are subsets of a finite set. In cases where private sets are subsets of an infinite set, we design a very efficient protocol (Protocol 3) using the properties of hash functions and prove that it is secure in the semi-honest model. In cases where only two parties are involved, we propose a solution (Protocol 4) that exploits the discrete logarithm assumption to produce satisfactory results when the private sets are subsets of either a finite or infinite set. Additionally, Protocol 4 can be simply modified to privately compute the cardinality of intersection and union set of private sets. As an application of the first protocol, we show a new scheme to privately compute the greatest common divisor of several private integers.

This work is supported by the National Natural Science Foundation of China (General Program) (No. 61272435) and the Fundamental Research Funds for the Central Universities of China (No. 2016TS061) and the China Scholarship Council.