高效的集合安全多方计算协议及应用

窦家维[1)] 刘旭红[1)] 周素芳[2)] 李顺东[2)]

[1)](陕西师范大学数学与信息科学学院 西安 710062)

[2)](陕西师范大学计算机科学学院 西安 710062)

摘 要 集合的安全多方计算是一个重要的科学问题，在秘密分享、保密投票、保密的数据挖掘等领域有广泛的应用. 现有的解决方案基本上是关于两方集合的安全计算，该文主要研究多个参与者集合的安全计算问题. 不同于现有的关于集合安全计算的研究方法，该文提出了全新的数学方法框架，通过应用编码方法并结合具有一定同态性的加密算法，将集合安全计算问题转化为数组的安全计算问题. 研究构造关于一些集合基本运算的安全计算协议，包括集合的交集/并集及其势的计算，有关阈值并集的计算. 该文所设计的集合安全计算协议具有以下特点：(1) 与现有方案比较，该文的协议具有计算效率高的优势，并且适合于多个集合的安全计算；(2) 能够应用标准的模拟范例方法对协议的安全性进行严格证明，协议能够抵抗任意的合谋攻击；(3) 综合应用该文所设计的协议或应用其设计思想，能够解决广泛的实际应用问题.

关键词 安全多方计算；集合运算；同态加密系统；编码方法；安全性

中图法分类号 TP309 **DOI 号** 10.11897/SP.J.1016.2018.01844

Efficient Secure Multiparty Set Operations Protocols and Their Application

DOU Jia-Wei[1)] LIU Xu-Hong[1)] ZHOU Su-Fang[2)] LI Shun-Dong[2)]

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

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

Abstract In the modern information-driven society, every identity (a person, an organization, an enterprise, a government sector etc.) has numerous private data which is the most important strategic resource and digital wealth of the identity. To release the great value of the data, they must share the data. In some important applications, some mutually distrustful parties must perform some computation over their private data to obtain some important information for decision-making, but they do not want to disclose the privacy of their data. In some cases, the input to the computation is each party's private set. To protect the privacy of these private sets, the parties must perform a privacy-preserving computation, that is, no party can, by performing the computation, learns more information about other parties' private sets than what can be deduced from the result. Secure multiparty computation is a key technology to protect the parties' privacy in such cooperative computation and a research focus in the international cryptographic community. Secure set operations are extensively used in fields such as secret sharing, secure voting, privacy-preserving data mining, and electronic commerce. The existing protocols for set operations focus mainly on the two-party computation, and are difficult to be extended directly to multi-party cases. At present, there hardly are researches on the secure set computations for multiple parties, and the few secure multiple set computation protocols are very

收稿日期：2017-01-13；在线出版日期：2017-12-27. 本课题得到国家自然科学基金(61272435)资助.**窦家维**，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. **刘旭红**，女，1992 年生，硕士研究生，主要研究方向为应用数学与应用密码学. **周素芳**，女，1990 年生，博士研究生，主要研究方向为密码学与信息安全. **李顺东**，男，1963 年生，博士，教授，博士生导师，主要研究领域为密码学与信息安全.

complex and very inefficient. This paper studies secure set operations for multiple parties. Different from the research methodology that the existing secure set operation protocols use, that is, to represent a private set as a polynomial, by skillfully constructing various coding methods and using additively or multiplicatively homomorphic cryptosystems, we develop a completely new mathematical framework that transforms secure set operations to secure array operations. Based on this framework, we put forward several secure and efficient basic set operation protocols for some basic set operation such as intersection set, the union set, the cardinality of intersection set and union set, and the threshold union set of some private sets. The contributions of this work are as follows: (1) we put forward a new encoding method to encode a set into an array such that each parties' private set is hidden in a special array. This encoding method plays an important role in secure set computations, and provides a new way for solving other multiparty secure computation problems; (2) using the new encoding method and the encryption algorithm with multiplicative or additive homomorphism, several secure and efficient computing protocols for some basic set operations are designed in the semi-honest model. These protocols can resist collision attack of arbitrary parties and we prove that they are secure by using the standard simulation paradigm; (3) the ideas and the protocols can be composed to solve extensive practical problems, and achieve more efficient results than those of previous work; (4) to resist malicious parties' behavior, we further design a protocol which is secure in the malicious model. We provide an experimental result to show that our protocols are efficient, and therefore are practical. As the applications of these protocols, we show how to use them in secure electronic voting, how to use them to privately determine whether multiple private numbers are equal, etc.

Keywords　secure multiparty computation; set operation; homomorphic encryption system; encoding scheme; security

1　引　言

网络的迅速发展为多个参与者利用各自的保密数据联合进行数据挖掘、知识发现、信息搜索以及寻求数据之间的各种统计规律、合作进行科学研究等提供了巨大的机会，同时也给参与者的信息安全带来了巨大的挑战. 在互不信任的网络环境中，参与者需要保护各自所拥有数据的隐私，在联合计算过程中稍有不慎就可能导致数据的机密性丧失与隐私泄露. 运用安全多方计算技术，既能充分发挥机密数据的作用，又能保护数据的机密性与隐私，这使得安全多方计算越来越受到人们的关注.

1982 年 Yao[1] 提出了两个参与者的安全计算问题，1988 年 Ben 和 Goldwasser[2] 引入了多个参与者的安全多方计算问题. 安全多方计算是指两个或更多参与者利用各自的保密数据(作为计算的输入)，联合进行的保密计算. 计算结束后，没有参与方能够获得多于规定输出的信息. 安全多方计算是网络空间信息安全与隐私保护的关键技术，它对于计算科学、密码学和信息安全的理论与实践都有重要的意义，是国际密码学界近年来的研究热点[3-5]. Goldwasser[6] 预言具有丰富理论基础和广泛应用前景的安全多方计算将成为计算科学一个必不可少的工具. Cramer 等人[7] 也指出安全多方计算将成为计算科学一个威力强大的工具.

Yao[8] 用混淆电路(garbled circuit)的方法证明了所有安全多方计算问题都是可解的，并给出了解决方案. Goldreich 等人[9-10] 对安全多方计算进行了深入的理论研究，通过把安全多方计算归约到智力游戏(mental game)上，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. 这两种解决方案共同的优势是它们的通用性，但存在的共同问题是它们的效率都太低. 因此 Goldreich 指出利用这些通用的解决方案来解决具体的安全多方计算问题是不实际的，从计算效率方面考虑，对具体的问题应该研究具体的解决方案.

Goldwasser，Goldreich 和 Cramer 关于安全多方计算的研究与论述激励着人们研究各种各样具体的安全多方计算问题. 这些问题包括保密的科学计

算[1,11-14],保密的计算几何[15-18],保密的统计分析[19-21],保密的数据挖掘[22-24],安全多方计算应用[25-27]等.

研究设计各种集合运算的保密计算协议是保密的科学计算中非常重要的问题,在自然科学、工程技术、社会科学等各个方面都有广泛的应用.比如,考虑下面问题:(1)n 个商业机构 $P_1,\cdots,P_n$,每个机构都拥有自己的 VIP 客户群,各个群的成员构成私密集合 $S_1,\cdots,S_n$.他们要合作进行一项营销活动,希望获知大家共同拥有的客户人数(或客户名单),又不想泄露各企业的私密客户群,这就需要保密计算这些集合交集的势(或交集本身);(2)在上面问题中,如果在保护每个客户群隐私的前提下,希望获得所有机构所拥有客户的综合信息,即客户群中出现的所有客户的总数量(或综合名单,每个客户只能出现一次),这就需要保密计算这些集合并集的势(或并集).我们再考虑下面的问题:(3)一个群体要对若干个候选人进行保密的投票选举,投票结果可能要求仅显示票数达到或超过某个规定阈值的候选人名单(比如有些选举需要选出多名代表,为了能代表民意,要求至少获得一定数量选票才能当选),或同时要求这些人的具体票数(有些选举不仅要选出多名代表,而且这些代表的职位高低也不同,因此同时根据具体票数使这些候选人能适得其所),这类问题可归结为有关阈值并集的保密计算问题(详细叙述参看第 8.3 节).

近年来,关于各种集合运算的安全计算问题受到广泛的关注,取得了很多重要的研究成果.本文的主要目的是应用新的技巧研究构造关于一些基本集合运算的高效安全计算协议,包括集合的交集/并集及其势的计算,有关阈值并集的计算.与本文所研究问题密切相关的工作有:

两方集合保密计算.在集合保密计算问题研究中,大多数的工作是关于两方计算的,文献[28]最先应用不经意多项式计算获得了集合交集及其势的两方保密计算协议.如果两参与者的集合所含元素个数分别为 w 和 v,协议具有 $O(w\log\log v)$ 的计算复杂性和 $O(w+v)$ 的通信复杂性.在文献[28]的基础上,文献[29]研究了恶意模型下的两方交集计算协议,协议的复杂性与文献[28]类似.文献[30]改进了文献[28]的复杂性,获得了关于输入集合规模的准线性(almost linear)的复杂性,文献[31-33]利用更多的密码学工具对于两方交集/并集的计算问题设计了具有线性复杂性的高效协议.在文献[34]中,作者首次获得了具有线性复杂性的集合交集/并集势的计算协议.

多方集合保密计算.文献[35]基于文献[28]的研究思想,应用多项式表示集合和各种集合运算以及多项式不经意计算的方法,首次研究设计了有 $n(n\geq 2)$ 个参与者的多重集的交集/并集及其势的计算协议,若每个参与方具有的集合元素个数为 k,其设计的交集/并集协议具有二次计算复杂性和线性通信复杂性,交集/并集势的计算协议具有二次计算复杂性和 $O(n^2k)$ 的通信复杂性;文献[35]也构造了有关阈值问题的集合计算协议,并具有类似的复杂性.文献[36-37]推进了文献[35]的工作,使得计算复杂性降低为关于参与者人数是线性的,而文献[38]应用多项式的点表示方法保密计算交集,其计算复杂性关于集合规模是拟线性的.

在文献[39]中,作者应用可交换的单向散列函数以及 Pohlig-Hellman 密码系统构造了一个交集势的 $n(n\geq 2)$ 方计算协议,其计算复杂性和通信复杂性分别为 $O(nv)$ 和 $O(n^2v)$.在文献[40]中作者以不经意排序以及数据比较为基础对于集合和多重集的多种运算的保密计算进行研究,如果所有参与者的元素之和为 m,文中所设计的协议具有 $O(m\log m)$ 的计算复杂性和通信复杂性,但协议要求各参与者相互之间具有安全认证信道.文献[41]应用 Bloom 过滤器的方法获得了集合交集/并集势的计算协议,由于避免应用公钥加密系统,所设计协议的复杂性较低,但要求参与者之间有私密的安全信道,其计算结果也仅是近似结果.

据我们所知,已有的多方集合保密计算协议关于集合规模基本上都具有二次或拟线性的计算复杂性.如文献[38]所指出的,设计关于集合规模具有线性复杂性的多方计算协议仍是一个公开问题.

已有的关于集合的保密计算大多是针对两个集合的计算问题,很难将其直接推广到多方的情形.目前,关于多个集合的保密计算的研究还很少,已有的一些方案设计都很复杂,计算复杂性较高.集合的多方保密计算在实际中具有重要的应用,比如利用网络系统在互不信任的多个成员之间进行保密的信息分享,或进行匿名的集体决策等都需要转化为一定的集合保密计算问题.关于多个集合的保密计算问题最直接的想法是将其归约到已有的两方计算协议,即对于集合两两进行保密计算来解决,但这样的方法一般都会泄露很多不应泄露的信息.其次,因为两方保密计算不需要考虑协议执行过程中的合谋攻

击，而在多方情形下能否抵抗合谋攻击是协议安全性的一个重要方面. 一般来说，把安全两方计算方案直接推广到安全多方计算是不可行的[42]，因此对于多个集合的保密计算问题需要寻求新的解决方案.

由于目前关于集合的多方保密计算研究结果还较少，而所应用的方法主要有混淆电路方法[8-9]，这类方法一般情况下复杂性都很高；应用公钥加密及多项式不经意计算的方法[35,38]，这类结果计算复杂性也较高(二次或拟线性)；还有一些文献应用信息分享或一些其它方法[40-41,43-44]，这些方法大多需要私密的通信信道.

我们注意到，对于很多实际应用问题能够事先确定集合元素在一定范围内取值(文献[35,38]也假设了这种情形)，在这种情形下，本文利用将集合编码成一些特殊数组的新技巧，并结合具有一定同态性质的公钥加密系统解决集合多方保密计算问题，研究设计了包括集合交集/并集以及势的计算，有关阈值并集计算等问题的高效计算协议. 本文的贡献如下：

(1) 提出了新的编码方法，使每个参与者的保密集合都隐藏在一个特殊数组中. 这类编码方法在集合保密计算中具有重要的作用，也可为解决其它安全多方计算问题提供一种新的途径.

(2) 利用所设计的新编码方法与具有乘法或加法同态性的加密算法，对于几种基本的集合运算在半诚实模型下设计了安全高效的计算协议. 对于输入集合含于一个全集而全集的势不太大的情形，所设计的协议具有很高的计算效率. 我们的方案适用于两方或多方集合保密计算，而且是高效的方案. 能够应用标准的模拟范例方法证明这些方案对半诚实参与者是安全的，可以抵抗任意的合谋攻击.

(3) 对于所设计的协议进行适当修改或者对其中几个进行适当的组合应用，对于更广泛的集合计算问题可设计构造高效安全的解决方案.

(4) 以交集保密计算问题为例，通过具体实例操作说明本文所设计协议是实用的和高效的. 并在半诚实模型下交集保密计算协议的基础上设计了在恶意模型下也安全的交集计算协议.

2　预备知识

2.1　半诚实模型

安全多方计算中通常应用的模型有半诚实模型和恶意模型. 所谓半诚实参与者是指那些在协议的执行过程中按照协议要求忠实地履行协议的参与者，但他们可能会记录下协议执行过程中收集到的所有信息，在协议执行后试图根据记录的信息推算出其他参与者的输入. 如果所有的参与者均为半诚实参与者，这样的计算模型称为半诚实模型. 由于半诚实参与者不对协议实施主动攻击，所以半诚实模型又称为诚实但好奇(honest-but-curious)模型或被动模型[10].

设有 n 个参与者 $P_1,\cdots,P_n$，分别具有保密数据 $x_1,\cdots,x_n$，记 $X=(x_1,\cdots,x_n)$. 他们利用协议 Π 保密地计算 $f(X)=(f_1(X),\cdots,f_n(X))$，其中 $f_i(X)$ $(i\in[n]=\{1,\cdots,n\})$ 为参与者 P_i 得到的输出结果. 在协议执行过程中，P_i 得到的信息序列记为

$$view_i^{\Pi}(X)=(x_i,r_i,M_i^1,\cdots,M_i^t),$$

其中 $M_i^j(j=1,\cdots,t)$ 表示 P_i 收到的第 j 个信息. 对于部分参与者构成的子集 $I=\{P_{i_1},\cdots,P_{i_s}\}\subseteq\{P_1,\cdots,P_n\}$，记

$$view_I^{\Pi}(X)=(I,view_{i_1}^{\Pi}(X),\cdots,view_{i_s}^{\Pi}(X)).$$

定义 1(半诚实模型下协议的安全性[10]). 在参与者都是半诚实的情况下，如果存在概率多项式时间算法 S，使得对于任意的 $I=\{P_{i_1},\cdots,P_{i_s}\}\subseteq\{P_1,\cdots,P_n\}$，均有下式成立：

$$\{S(I,(x_{i_1},\cdots,x_{i_s}),f_I(X))\}_{X\in(\{0,1\}^*)^n}\overset{c}{\equiv}\{view_I^{\Pi}(X)\}_{X\in(\{0,1\}^*)^n}\quad(1)$$

其中，$\overset{c}{\equiv}$ 表示计算上不可区分，则称协议 Π 保密地计算了 n 元函数 $f(X)$.

显然，如果对于任意 $n-1$ 个参与者构成的集合 Γ，都存在满足式(1)的 S，则协议 Π 能够抵抗任意的合谋攻击.

2.2　恶意模型

关于恶意模型下协议的安全性定义以及如何由半诚实模型下的保密计算协议编译获得恶意模型下的安全计算协议的具体方案，可参看文献[10]详细了解.

恶意模型下安全的多方计算协议应迫使各参与者像半诚实参与者一样按协议要求执行协议. 但有三种恶意行为无法避免(在任意协议中)，即参与者拒绝参加协议，参与者修改其原本规定的输入数据而用其它数据替代，以及参与者在协议执行过程中可能随时中止协议. 因此在恶意模型下考虑协议的安全性时这几种恶意行为原则上不予考虑[10].

2.3　ElGamal 同态加密系统

同态性是某些公钥加密系统所具有的重要性

质,在多方保密计算中有重要的应用.一般的公钥加密系统包括密钥生成算法,生成私钥/公钥 sk/pk;加密算法 E_{pk} 以及解密算法 D_{sk}.一个同态加密系统除了包括上述三个算法外还要满足一定的同态性质,表示为

$$E_{pk}(M_1 \oplus M_2) = E_{pk}(M_1) \oplus_{pk} E_{pk}(M_2) \quad (2)$$

其中$\oplus$和$\oplus_{pk}$分别表示明文及密文之间的某种代数运算,如果$\oplus$表示乘法(加法)运算,则表明加密系统具有乘法(加法)同态性.

ElGamal 加密系统是一种具有乘法同态性的公钥加密系统.具体描述如下[45]:

密钥生成. 给定安全参数 k,密钥生成算法生成一个 k 比特的大素数 p 以及 Z_p^* 的一个生成元 g,随机选取 x 作为私钥,对应的公钥为 $h=g^x \bmod p$.

加密. 为加密消息 $M(M\in Z_p^*)$,选择随机数 r,密文为

$$E_{pk}(M) = (c_1, c_2) = (g^r \bmod p, Mh^r \bmod p).$$

解密. 对于密文 $E_{pk}(M)=(c_1,c_2)$ 解密为

$$M = c_2 \cdot c_1^{-x} \bmod p.$$

同态性质. 加密系统具有乘法同态性:

$$\begin{aligned} E_{pk}(M_1) \times E_{pk}(M_2) &= (g^{r_1}, M_1 h^{r_1}) \times (g^{r_2}, M_2 h^{r_2}) \\ &= (g^{r_1+r_2}, M_1 \times M_2 h^{r_1+r_2}) \\ &= E_{pk}(M_1 \times M_2). \end{aligned}$$

对 ElGamal 加密系统稍加修改,可使其成为加法同态加密系统[32,38].我们将修改后的加密系统称为 ElGamal 修改系统.具体描述如下:密钥生成算法保持不变,修改后的加密算法和解密算法分别记为 $\hat{E}_{pk}$ 和 $\hat{D}_{pk}$.

加密. 为加密消息 $M(M\in Z_p^*)$,选择随机数 r,密文为

$$\hat{E}_{pk}(M) = (c_1, c_2) = (g^r \bmod p, g^M h^r \bmod p).$$

解密. 对于密文 $\hat{E}_{pk}(M)=(c_1,c_2)$,解密为

$$g^M = c_2 \cdot c_1^{-x} \bmod p.$$

同态性质. 加法同态性:

$$\begin{aligned} \hat{E}_{pk}(M_1) \times \hat{E}_{pk}(M_2) &= (g^{r_1}, g^{M_1} h^{r_1}) \times (g^{r_2}, g^{M_2} h^{r_2}) \\ &= (g^{r_1+r_2}, g^{M_1+M_2} h^{r_1+r_2}) \\ &= \hat{E}_{pk}(M_1 + M_2). \end{aligned}$$

ElGamal 密码系统及其修改系统都是语义安全的.一个密码系统是语义安全的,意味着同一明文可以加密成多个不同的密文形式,并且这些密文在计算上都是不可区分的.我们也注意到,ElGamal 修改系统解密仅可得到 g^M,因此不具有完全解密性.

2.4 门限密码体制

门限密码体制[46-47]是安全多方计算中对抗合谋攻击的一个重要工具.在门限密码体制中,n 个参与者联合生成一个公钥,解密密钥由这 n 个参与者联合持有.公钥可以直接用来加密消息,但解密一个密文需要 n 个参与者中一定数量人员合作才能完成.如果至少需要 t 个人合作才能解密,少于 t 个人合作时得不到明文的任何信息,这样的密码体制称为(t,n)门限密码体制.本文需要的是一种朴素的门限密码体制,即(n,n)门限密码体制,它是抵抗合谋攻击的一种有效方法.本文所应用的密码系统需要具有乘法(或加法)同态性以及语义安全性,并能应用其构造(n,n)门限密码体制.应用 ElGamal 密码系统可构造(n,n)门限密码体制,具体构造如下:

联合生成公钥:n 个参与者 $P_i(i\in[n])$ 选取 ElGamal公钥系统的参数 g,p.每个参与者 P_i 选取私钥 x_i,公布 $h_i=g^{x_i} \bmod p$,如此即可联合生成公钥 $pk: h=g^{x_1+\cdots+x_n} \bmod p$.这里所有参与者联合持有私钥 $sk=x_1+\cdots+x_n$.

联合解密:为了解密应用公钥 $h=g^{x_1+\cdots+x_n} \bmod p$ 所加密的密文$(u,v)=(g^r \bmod p, Mh^r \bmod p)$,每个参与者 P_i 计算 $w_i=u^{x_i} \bmod p$,并公布.并进行进一步计算:

$$M \equiv v\Big[\prod_{i=1}^{n} w_i\Big]^{-1} \bmod p.$$

上面所构造的门限密码体制具有乘法同态性,将在本文协议 1~4 的构造中应用.

如果把上面构造中的 M 以 g^M 代替,即可获得具有加法同态性的门限密码体制,它可应用于本文协议 5,6 的构造中.

与 ElGamal 密码系统类似,由离散对数问题的困难性假设可保证上面所构造的门限密码体制是语义安全的[38,45].在公钥生成(或联合解密)过程中每个参与方需要进行两个模指数运算,仅需进行一轮交互即可完成.

3 集合交集/并集保密计算

3.1 集合交集计算协议

问题描述. 考虑 $n\geq 2$ 个参与者 $P_i, i\in[n]$,P_i 具有私密输入集合 S_i.他们希望合作计算所有集合的交集 $\bigcap_{i=1}^{n} S_i$ 而不泄露任何 S_i.假设 $S_i \subseteq Z=\{e_1,\cdots,e_m\}$,其中 $e_1<e_2<\cdots<e_m$.

计算原理. 设计一个新的编码方法,即 $1-r$ 编码,使得每个 P_i 能将自己的保密集合 S_i 与一个 m 维数组 $X_i=(x_{i1},\cdots,x_{im})$ 相对应.具体构造方法如下:对于 $i\in[n], k\in[m]$,令

$$x_{ik}=\begin{cases}1, & \text{如果 } e_k\in S_i\\ r_{ik}, & \text{否则}\end{cases} \tag{3}$$

式(3)中，r_{ik}为大于 1 的随机整数.

命题 1.

$$e_k\in\bigcap_{i=1}^{n}S_i\Longleftrightarrow\prod_{i=1}^{n}x_{ik}=1.$$

证明.　如果 $e_k\in\bigcap_{i=1}^{n}S_i$，则对于所有 $i\in[n]$，有 $e_k\in S_i$，因此 $x_{ik}=1$. 即有 $\prod_{i=1}^{n}x_{ik}=1$.

由编码方式(3)，如果 $e_k\notin\bigcap_{i=1}^{n}S_i$，则至少有一个 x_{ik}为大于 1 的随机数，$\prod_{i=1}^{n}x_{ik}$也是大于 1 的随机数. 这意味着如果 $\prod_{i=1}^{n}x_{ik}=1$，则对于所有的 $i\in[n]$，$x_{ik}=1$，即 $e_k\in S_i$，因此 $e_k\in\bigcap_{i=1}^{n}S_i$.　证毕.

命题 1 是我们计算若干个集合交集的基本原理，但直接这样做没有秘密可言. 如果能够保密计算 $\prod_{i=1}^{n}x_{ik}$，根据乘积的值就能判别 e_k是否属于交集 $\bigcap_{i=1}^{n}S_i$. 集合交集的具体协议参看协议 1.

协议 1.　集合交集计算协议.

输入：有 n 个半诚实参与者 $P_i(i\in[n])$，每个 P_i拥有私密集合 S_i，在 ElGamal 具有乘法同态性的门限密码体制中，所有参与者联合持有私钥 sk，其对应的公钥为 pk

输出：所有参与者集合的交集 $f_\wedge(S_1,\cdots,S_n)=\bigcap_{i=1}^{n}S_i$

1. 参与者 P_i，$i\in[n]$将集合 S_i按照式(3)转化成数组 X_i；

2. 参与者 P_i，$i\in[n]$加密数组中的元素 1，使其与随机数不可区分，加密后的数组记为 $C_i=(c_{i1},\cdots,c_{im})$；

3. P_1将 $W_1=C_1$发送给 P_2；

4. 每一个 P_i，$i=2,\cdots,n$；

(a) 从 P_{i-1}处接收到 $W_{i-1}=C_1\cdots C_{i-1}$，

(b) 将 C_i与 W_{i-1}的对应分量相乘，得到新的乘积数组 $W_i=C_1\cdots C_{i-1}\cdot C_i$，

(c) 发送密文乘积数组 W_i给 $P_{(i+1)\bmod n}$；

5. 参与者 P_1公开 $W=C_1\cdots C_n=(w_1,\cdots,w_m)$；

6. 所有的参与者联合解密 W 的每一个分量 w_k，得到一个数值 $a_k(1\le k\le m)$；

7. 如果 $a_k=1$，则 e_k是所求交集中的一个元素；

8. 输出交集 $f_\wedge(S_1,\cdots,S_n)=\{e_k\mid a_k=1,k=1,\cdots,m\}$.

3.2　正确性

协议 1 是正确的意味着对于任意的输入集合 $S_i\subseteq Z(i\in[n])$，协议能够正确地计算得到交集 $\bigcap_{i=1}^{n}S_i$. 协议 1 的正确性由命题 1 以及加密系统所具有的乘法同态性质很容易得到. 具体地，如果 $e_k\in\bigcap_{i=1}^{n}S_i$，则对于所有的 $i\in[n]$，$x_{ik}=1$，因此 w_k为 1 的密文，即有 $D_{sk}(w_k)=1$. 如果 $e_k\notin\bigcap_{i=1}^{n}S_i$，则至少有一个 x_{ik}为大于 1 的随机数，所以 $D_{sk}(w_k)$也是大于 1 的随机数. 因此协议 1 是正确的.

3.3　安全性

关于协议 1 的安全性，我们有下面的结论.

定理 1.　协议 1 是安全的，能够抵抗任意的合谋攻击.

证明.　应用文献[10]中的模拟范例严格证明定理 1. 对于任意 $n-1$ 个参与者构成的合谋者集合 Γ，需要构造相应的模拟器 S，使得式(1)成立.

由于各参与者地位的平等性，不妨设 $\Gamma=\{P_1,\cdots,P_{n-1}\}$，他们合谋想获知 P_n的私密集合 S_n中的元素. S 按如下方式运行：

(1) S 运行具有乘法同态性的 ElGamal 加密系统生成私钥 sk，其对应的公钥为 pk.

(2) 对于输入 $(S_1,\cdots,S_{n-1},f_\wedge(S_1,\cdots,S_{n-1},S_n))$，$S$ 构造一个集合 S_n'，使得

$$f_\wedge(S_1,\cdots,S_{n-1},S_n)=f_\wedge(S_1,\cdots,S_{n-1},S_n'),$$

并由式(3)构造 S_n'对应的数组 X_n'.

(3) S 按照协议 1 第 2 步的方法对 X_n'进行加密，得到 $C_n'=(c_{n1}',\cdots,c_{nm}')$.

(4) S 计算得到密文乘积数组

$$W'=C_1\cdots C_{n-1}\cdot C_n'=(w_1',\cdots,w_m').$$

(5) S 解密得到 $a_k'=D_{sk}(w_k')$，$k=1,\cdots,m$.

在协议的执行中，

$$view_\Gamma^\pi(S_1,\cdots,S_{n-1},S_n)=$$
$$\{S_1,\cdots,S_{n-1},(w_1,\cdots,w_m),(a_1,\cdots,a_m)\}.$$

令

$$S(S_1,\cdots,S_{n-1},f_\wedge(S_1,\cdots,S_{n-1},S_n))=$$
$$\{S_1,\cdots,S_{n-1},(w_1',\cdots,w_m'),(a_1',\cdots,a_m')\}.$$

因为 ElGamal 公钥加密系统是语义安全的，并且所有参与者合作才能正确解密，即使 Γ 中的所有参与者合谋也无法解密任何密文，因此对于 Γ 中的参与者而言，$C_n=(c_{n1},\cdots,c_{nm})$与 $C_n'=(c_{n1}',\cdots,c_{nm}')$是计算不可区分的，进而有

$$(w_1,\cdots,w_m)\overset{c}{\equiv}(w_1',\cdots,w_m').$$

最后，对于每一个 $k=1,\cdots,m$，要么 $a_k'=a_k$，要么 a_k'与 a_k同为大于 1 的随机数，因此 $(a_1,\cdots,a_m)\overset{c}{\equiv}(a_1',\cdots,a_m')$. 所以，

$$\{view_\Gamma^\pi(S_1,\cdots,S_n)\}_{S_i\subseteq Z,i\in[n]}\overset{c}{\equiv}$$
$$\{S(S_1,\cdots,S_{n-1},f_\wedge(S_1,\cdots,S_n))\}_{S_i\subseteq Z,i\in[n]}.$$

因此定理 1 得证.　证毕.

3.4　集合并集保密计算

问题描述. 考虑 $n\ge2$ 个参与者 P_i，$i\in[n]$，每个 P_i具有一个私密输入集合 S_i. 他们希望合作计算

所有集合的并集$\bigcup_{i=1}^{n}S_i$而不泄露任何S_i.假设$S_i\subseteq Z$.计算的基本原理非常类似于集合交集计算,但编码方式要相应改变.

计算原理. 设计一个新的编码方法,即$r-1$编码,使得每个P_i能将自己的私密集合与一个m维数组$Y_i=(y_{i1},\cdots,y_{im})$相对应.具体构造方法如下:对于$i\in[n],k\in[m]$,令

$$y_{ik}=\begin{cases}r_{ik}, & \text{如果 } e_k\in S_i\\ 1, & \text{否则}\end{cases} \tag{4}$$

式中,r_{ik}为大于1的随机整数.

命题2.

$e_k\in\bigcup_{i=1}^{n}S_i\Longleftrightarrow\prod_{i=1}^{n}y_{ik}$为大于1的随机数.

证明. 如果$e_k\in\bigcup_{i=1}^{n}S_i$,则至少有一个$i\in[n]$,使得$e_k\in S_i$,即$y_{ik}$为大于1的随机数,因此$\prod_{i=1}^{n}y_{ik}$为大于1的随机数.反之,由编码方式(4),如果$e_k\notin\bigcup_{i=1}^{n}S_i$,则$\prod_{i=1}^{n}y_{ik}=1$.这意味着如果$\prod_{i=1}^{n}y_{ik}$为大于1的随机数,则至少有一个$i\in[n]$,$y_{ik}$为大于1的随机数,这保证了$e_k\in\bigcup_{i=1}^{n}S_i$.证毕.

命题2是计算集合并集的基本原理,以此为基础可设计并集保密计算协议,并集计算协议与交集的计算协议非常类似,简单描述如下.

协议2. 集合并集计算协议.

集合并集计算协议只需将协议1中第1步和第7步做少量修改,其它保持不变即可,其输出结果为所有参与者集合的并集$\bigcup_{i=1}^{n}S_i$.协议1中第1步和第7步应修改为:

1. 参与者$P_i,i\in[n]$将集合S_i按照式(4)转化成数组Y_i;

7. 如果$a_k\neq1$为随机数,则e_k是所求并集$\bigcup_{i=1}^{n}S_i$中的一个元素.

关于协议2的正确性与安全性,我们有下面的定理2.由于定理2的证明与定理1完全类似,故省略.

定理2. 协议2是正确的和安全的,能够抵抗任意的合谋攻击.

4 集合交集/并集势的保密计算

4.1 协议设计

问题描述及计算原理. 考虑$n\geq2$个参与者$P_i,i\in[n]$,每个P_i具有一个私密输入集合S_i.他们希望合作计算所有集合的交集(并集)的势$|\bigcap_{i=1}^{n}S_i|$($|\bigcup_{i=1}^{n}S_i|$)而不泄露任何S_i.这里我们假设$S_i\subseteq Z$.对于交集(并集)势的保密计算问题,协议设计类似于相应的集合交集(并集)计算协议,只是在协议最后需要对乘积密文数组进行进一步的处理,在保证交集(并集)中的元素个数不变的条件下,隐藏所含元素的具体信息.集合交集势的计算协议参看协议3.

协议3. 集合交集势的计算协议.

输入:有n个半诚实参与者$P_i(i\in[n])$,每个P_i拥有私密集合S_i,在ElGamal具有乘法同态性的门限密码体制中,所有参与者联合持有私钥sk,其对应的公钥为pk

输出:所有参与者集合交集的势$|\bigcap_{i=1}^{n}S_i|$

1. 参与者$P_i,i\in[n]$将S_i按照式(3)转化成数组X_i;

2. 参与者$P_i,i\in[n]$加密数组中的元素1,使其与随机数不可区分,加密后的数组记为$C_i=(c_{i1},\cdots,c_{im})$;

3. P_1将$W_1=C_1$发送给P_2;

4. 每一个$P_i,i=2,\cdots,n-1$

(a) 从P_{i-1}处接收到$W_{i-1}=C_1\cdots C_{i-1}$,

(b) 将C_i与W_{i-1}的对应分量相乘,得到新的乘积数组$W_i=C_1\cdots C_{i-1}\cdot C_i$,

(c) 发送密文乘积数组W_i给P_{i+1};

5. 参与者P_n计算$W_n=C_1\cdot C_2\cdots C_n=(w_1,\cdots,w_m)$,并将其元素进行随机置换(记置换为$\pi_n$),得到$V_n=\pi_n(W_n)=(v_{n1},\cdots,v_{nm})$;

6. 每一个$P_i,i=n-1,\cdots,2$

(a) 从P_{i+1}处接收到$V_{i+1}=(v_{(i+1)1},\cdots,v_{(i+1)m})$,

(b) 对V_{i+1}的每一个元素乘以1的不同密文,再进行随机置换(记置换为π_i),得到

$V_i=(v_{i1},\cdots,v_{im})=\pi_i(v_{(i+1)1}E_{pk}(1),\cdots,v_{(i+1)m}E_{pk}(1))$,

(c) 发送密文乘积数组V_i给P_{i-1};

7. P_1将接收到的数组V_2的每一个元素乘以1的不同密文,再进行随机置换(记置换为π_1),得到

$V=(v_1,\cdots,v_m)=\pi_1(v_{21}E_{pk}(1),\cdots,v_{2m}E_{pk}(1))$;

8. P_1公开V.所有的参与者解密V的每一个元素,得到m个数值$a_k(1\leq k\leq m)$;

9. 其中取值为1的a_k的数目即为所求交集的势$|\bigcap_{i=1}^{n}S_i|$.

4.2 协议3的正确性

由于协议3的前4步与协议1相同,首先得到密文乘积数组$W_n=(w_1,\cdots,w_m)$,且可知如果$e_k\in\bigcap_{i=1}^{n}S_i$,则$w_k$为1的密文.协议3在此基础上对乘积数组的每个元素$w_k$乘了若干个1的密文$E_{pk}(1)$,并对数组进行若干次随机置换,这些操作均不改变$w_k$的解密结果,仅仅改变$w_k$在原数组中的排列位置,因此满足$D_{sk}(w_k)=1$的数组元素$w_k$的总个数没有改变,所以对于任意的输入集合$S_i\subseteq Z,i\in[n]$,协议3能够正确地计算得到交集的势$|\bigcap_{i=1}^{n}S_i|$.

4.3 协议3的安全性

定理3. 协议3是安全的,能够抵抗任意的合谋攻击.

证明.　类似于定理 1 的证明，容易证明任一参与者 P_i 的私密集合对于其他 $n-1$ 个参与者构成的合谋攻击都是安全的. 在定理 1 的基础上我们仅需再证明除了规定的输出结果外，协议 3 对于交集中所含元素的其它信息都是保密的.

在协议 3 中，由于交集中所含元素的有关信息是由 w_k 的位置信息所确定的，为了对所有参与者隐藏 w_k 的位置信息，每个 P_i 首先对 P_{i+1} 发来的数组元素乘以 $E_{pk}(1)$，并进一步进行随机置换，这样得到的新数组中各元素的排列位置对于其他参与者来说是完全保密的（注意到 P_n 发送给 P_{n-1} 的数组也经过了随机置换，也具有同样的保密性），因此最后得到的保密数组 $(v_1,\cdots,v_m)$ 中元素的排列位置完全掩盖了原数组 $(w_1,\cdots,w_m)$ 中元素的排列位置. 又由于每个 P_i 都参与了这样的保密计算及置换过程，所以协议 3 能够抵抗任意 $n-1$ 个参与者的合谋攻击.

集合并集势的计算协议与交集势的计算协议非常类似，简单描述如下：

协议 4.　集合并集势的计算协议.

集合并集势的计算协议只需将协议 3 中第 1 步和第 9 步做少量修改，其它保持不变，其输出结果为所有参与者集合并集的势 $|\bigcup_{i=1}^{n} S_i|$. 协议 3 中第 1 步和第 9 步应修改为：

1. 参与者 $P_i, i\in[n]$ 将集合 S_i 按照式(4)转化成数组 Y_i；

9. 其中取值不为 1 的 a_k 的数目即为所求并集的势 $|\bigcup_{i=1}^{n} S_i|$.

以协议 2 的正确性和安全性为基础，协议 4 的正确性和安全性可完全类似于交集势的相关结论进行证明，仅叙述下面定理，证明省略.

定理 4.　协议 4 是正确的和安全的，能够抵抗任意的合谋攻击.

5　阈值并集问题保密计算

本部分主要研究阈值并集以及阈值多重并集的保密计算问题. 多重集（或称多重集合）是集合概念的推广. 在一个集合中，相同的元素只能出现一次，因此只能显示出有或无的属性. 在多重集之中，同一个元素可以出现多次.

5.1　阈值并集保密计算

问题描述及计算原理. 考虑 $n\geq 2$ 个参与者 P_i，$i\in[n]$，每个 P_i 具有一个私密输入集合 S_i. 假设 $S_i\subseteq Z$. 在不泄露 S_i 的条件下，参与者合作计算在 n 个私密集合中出现次数合计达到阈值 t 的所有元素构成的并集，要求对这些元素出现的具体次数保密. 我们将此类问题称为阈值并集问题，并将其记为 $f_T(S_1,\cdots,S_n)$.

假如有 5 个集合

$$\begin{aligned} &S_1=\{1,3,6,8\},\ S_2=\{1,2,4,6,7\},\\ &S_3=\{2,3,6,7\},\ S_4=\{1,3,6,9\},\ S_5=\{3,6,8\} \end{aligned} \tag{5}$$

如果设置阈值为 $t=3$，那么这 5 个集合的阈值并集为 $\{1,3,6\}$.

为解决阈值并集保密计算问题，需要构造新的编码方法，使得每个 P_i 能将自己的私密集合与一个 m 维数组 $U_i=(u_{i1},\cdots,u_{im})$ 相对应. 具体构造方法如下：

$$u_{ik}=\begin{cases}1, & \text{如果 } e_k\in S_i\\ 0, & \text{否则}\end{cases} \tag{6}$$

容易证明下面结论：

命题 3.

$$e_k\in f_T(S_1,\cdots,S_n)\Longleftrightarrow\sum_{i=1}^{n}u_{ik}\geq t.$$

证明.　如果 $e_k\in f_T(S_1,\cdots,S_n)$，则存在 $j_1,\cdots,j_t\in[n]$，使得 $e_k\in S_{j_s}$ $(s=1,\cdots,t)$. 根据编码方式(6)，相应地有 $u_{j_sk}=1$，因此 $\sum_{i=1}^{n}u_{ik}\geq t$ 成立. 反之，如果 $\sum_{i=1}^{n}u_{ik}\geq t$，$S_1,\cdots,S_n$ 中至少有 t 个集合含有元素 e_k，因此 $e_k\in f_T(S_1,\cdots,S_n)$.　证毕.

命题 3 是我们计算阈值并集的基本原理，对于每一个 $k\in[m]$，首先应用具有加法同态性的加密系统保密计算 $\sum_{i=1}^{n}u_{ik}$，进一步根据和式的值保密判别 e_k 是否属于 $f_T(S_1,\cdots,S_n)$. 协议中应用了文献[35,48]中给出的 IsEq 协议. 根据 IsEq 协议，对于应用公钥 pk 加密的两个密文 C 和 C'，如果 C,C' 为同一明文的密文，则 $IsEq(C,C')=1$；如果 C,C' 为不同明文的密文，则 $IsEq(C,C')=0$（这个计算很容易由文献[48]中的技巧得到，具体请参看文献[35,48]）. 在文献[35]所设计的阈值并集（perfect threshold set-union）保密计算协议中，为了隐藏超阈值并集（over threshold set-union，为多重集）中各元素出现的具体次数，需要将密文并集中所有元素 $U'_1,\cdots,U'_{nk}$ 应用 IsEq 协议顺次比较，如果密文 U'_j 与其前面的某个密文为同一明文所加密的密文，则需要将 U'_j 随机化，如果密文 U'_j 与其前面的所有密文比较均为不同明文所加密的密文，则保留 U'_j. 在本文的阈值并集保密计算协议 5 中，应用 IsEq 协议的思想，将每个数值 e_k 在所有私密集合 S_i 中出现的总次数的密文 w_k 与已有密文 $E_l=\hat{E}_{pk}(l)$ $(l=1,\cdots,t-1)$ 进

行比较，容易看到 e_k 属于阈值并集的充要条件是对于所有 $l=1,\cdots,t-1$，均有 $IsEq(w_k,E_l)=0$. 我们根据这个原理构造阈值并集协议. 具体协议参看协议 5.

协议 5. 阈值并集计算协议.

输入：有 n 个半诚实参与者 $P_i(i\in[n])$，每个 P_i 拥有私密集合 S_i，在具有加法同态性的门限密码体制中，所有参与者联合持有私钥 sk，其对应的公钥为 pk. 并有密文 $E_l=\hat{E}_{pk}(l)(l=1,\cdots,t-1)$

输出：参与者集合的阈值并集 $f_T(S_1,\cdots,S_n)$

1. 参与者 $P_i,i\in[n]$ 将 S_i 按照式(6)转化成数组 U_i；

2. 参与者 $P_i,i\in[n]$ 加密数组中的各个元素，加密后的数组记为 $C_i=(c_{i1},\cdots,c_{im})$；

3. P_1 将 $W_1=C_1$ 发送给 P_2；

4. 每一个 $P_i,i=2,\cdots,n$

(a) 从 P_{i-1} 处接收到 $W_{i-1}=C_1\cdots C_{i-1}$，

(b) 将 C_i 与 W_{i-1} 的对应分量相乘，得到新的乘积数组 $W_i=C_1\cdots C_{i-1}\cdot C_i$，

(c) 发送密文乘积数组 W_i 给 $P_{(i+1)\bmod n}$；

5. 参与者 P_1 公开 $W=C_1\cdot C_2\cdots C_n=(w_1,\cdots,w_m)$；

6. 对于每一个 $i\in[n],k\in[m]$

(a) P_i 选择随机数 $q_{ik}\in N^+=\{1,2,\cdots\}$，

(b) 计算 $v_k=\hat{E}_{pk}\Big(\big(\sum_{i=1}^{n}q_{ik}\big)(IsEq(w_k,E_1)+\cdots+IsEq(w_k,E_{t-1}))\Big)$；

7. 所有的参与者解密每一个 v_k，得到一个数值 $a_k(1\leq k\leq m)$；

8. 如果 $a_k=0$，则 e_k 是所求阈值并集 $f_T(S_1,\cdots,S_n)$ 中的一个元素；

9. 输出 $f_T(S_1,\cdots,S_n)$.

5.2 协议 5 的正确性

协议 5 是正确的意味着对于任意的输入集合 $S_i\subset Z(i\in[n])$，协议能够正确地计算得到阈值并集 $f_T(S_1,\cdots,S_n)$.

协议 5 的正确性由命题 3 以及加密系统所具有的加法同态性质立刻得到. 具体地，如果 $e_k\in f_T(S_1,\cdots,S_n)$，则 $S_1,\cdots,S_n$ 中至少有 t 个集合含有元素 e_k，对应这些集合的数组的第 k 个元素取值为 1，因此相应地有 $\hat{D}_{sk}(w_k)\geq t$.

如果 $\hat{D}_{sk}(w_k)\geq t$，则在参与者构造的 n 个保密数组中，其中至少有 t 个数组的第 k 个元素取值为 1，而与这些数组对应的集合都含有元素 e_k，因此 $e_k\in f_T(S_1,\cdots,S_n)$. 故协议 5 是正确的.

5.3 协议 5 的安全性

关于协议 5 的安全性，有下面的结论.

定理 5. 如果所用的公钥加密系统是语义安全的，则协议 5 是安全的，能够抵抗任意的合谋攻击.

证明. 我们应用模拟范例[10]严格证明定理 5. 对于任意 $n-1$ 个参与者构成的合谋者集合 Γ，需要构造相应的模拟器 S，使得式(1)成立. 由于各参与者地位的平等性，不妨设 $\Gamma=\{P_1,\cdots,P_{n-1}\}$，他们合谋想获知 P_n 的私密集合 S_n 中的元素. S 按如下方式运行：

(1) S 运行具有加法同态性的公钥加密系统产生私钥 sk，其对应的公钥为 pk. 并计算密文 $E'_1=\hat{E}_{pk}(1),\cdots,E'_{t-1}=\hat{E}_{pk}(t-1)$；

(2) 对于输入 $(S_1,\cdots,S_{n-1},f_T(S_1,\cdots,S_{n-1},S_n))$，$S$ 构造一个集合 S'_n，使得

$$f_T(S_1,\cdots,S_{n-1},S_n)=f_T(S_1,\cdots,S_{n-1},S'_n),$$

并根据式(6)构造 S'_n 对应的数组 U'_n；

(3) S 按照协议 5 第 2 步的方法对 U'_n 进行加密，获得 $C'_n=(c'_{n1},\cdots,c'_{nm})$；

(4) S 计算得到乘积数组

$$W'=C_1\cdots C_{n-1}\cdot C'_n=(w'_1,\cdots,w'_n);$$

(5) S 选择 mn 个随机数 $q'_{ik}\in N^+,i\in[n],k\in[m]$，并计算

$$v'_k=\hat{E}_{pk}\Big(\big(\sum_{i=1}^{n}q'_{ik}\big)(IsEq(w'_k,E'_1)+\cdots+IsEq(w'_k,E'_{t-1}))\Big);$$

(6) S 解密每一个 v'_k，得到一个数值 $a'_k(1\leq k\leq m)$，并得到阈值并集

$$f_T(S_1,\cdots,S_{n-1},S'_n)=\{e_k\mid a'_k=0,k=1,\cdots,m\}.$$

在协议的执行中，

$$view_\Gamma^\pi(S_1,\cdots,S_{n-1},S_n)=\{S_1,\cdots,S_{n-1},(w_1,\cdots,w_m),(q_{n1},\cdots,q_{nm}),(v_1,\cdots,v_m),f_T(S_1,\cdots,S_{n-1},S_n)\},$$

令

$$S(S_1,\cdots,S_{n-1},f_T(S_1,\cdots,S_{n-1},S_n))=\{S_1,\cdots,S_{n-1},(w'_1,\cdots,w'_m),(q'_{n1},\cdots,q'_{nm}),(v'_1,\cdots,v'_m),f_T(S_1,\cdots,S_{n-1},S'_n)\}.$$

因为公钥加密系统是语义安全的，并且所有参与者合作才能正确解密，即使 Γ 中的所有参与者合谋也无法解密任何密文，因此对于 Γ 中的参与者而言，密文数组 $C_n=(c_{n1},\cdots,c_{nm})$ 与 $C'_n=(c'_{n1},\cdots,c'_{nm})$ 是计算不可区分的，进而有

$$(w_1,\cdots,w_m)\overset{c}{\equiv}(w'_1,\cdots,w'_m).$$

又由于 $(v_1,\cdots,v_m)$ 和 $(v'_1,\cdots,v'_m)$ 都是公钥 pk 所加密的密文，对于 Γ 中的参与者而言两组密文在计算上是不可区分的. 最后，由于随机数组在计算上

都是不可区分的，即有 $(q_{n1},\cdots,q_{nm})\overset{c}{\equiv}(q'_{n1},\cdots,q'_{nm})$，而 $f_T(S_1,\cdots,S_{n-1},S_n)=f_T(S_1,\cdots,S_{n-1},S'_n)$，因此

$$\{view_T^{\pi}(S_1,\cdots,S_{n-1},S_n)\}_{S_i\subseteq Z,i\in[n]}\overset{c}{\equiv}$$

$$\{S(S_1,\cdots,S_{n-1},f_T(S_1,\cdots,S_{n-1},S_n))\}_{S_i\subseteq Z,i\in[n]},$$

即式(1)成立，因此协议 5 是安全的.　　证毕.

5.4　阈值多重并集保密计算

问题描述. 考虑 $n\geq 2$ 个参与者 $P_i,i\in[n]$，每个 P_i 具有一个私密输入集合 S_i，假设 $S_i\subseteq Z$. 与前面的阈值并集问题不同的是，在这里参与者不但要保密计算 n 个私密集合中出现达到阈值 t 次的所有元素构成的并集，还要同时给出该并集中各个元素出现的具体次数，这样的阈值并集可由一个多重集描述. 我们将此类问题称为阈值多重并集问题，并将其记为 $f_{TM}(S_1,\cdots,S_n)$. 如果 $a_k\in f_{TM}(S_1,\cdots,S_n)$，进一步将 a_k 出现的总次数记为 $TM(a_k)$.

例如，对于式(5)中给出的 5 个集合，如果设置阈值为 $t=3$，那么这些集合的阈值多重并集为 $f_{TM}(S_1,\cdots,S_5)=\{1,1,1,3,3,3,3,6,6,6,6,6\}$，而 $TM(1)=3,TM(3)=4,TM(6)=5$.

计算原理. 为解决阈值多重并集问题，仍可按照式(6)中的 1－0 编码方式，使得参与者 P_i 将自己的私密集合对应于 m 维数组 $U_i=(u_{i1},\cdots,u_{im})$. 并容易证明下面结论：

命题 4.

$$[e_k\in f_{TM}(S_1,\cdots,S_n)]\wedge[TM(e_k)=n_k]$$

$$\Longleftrightarrow\sum_{i=1}^{n}u_{ik}=n_k.$$

证明. 如果 $e_k\in f_{TM}(S_1,\cdots,S_n)$ 并且 $TM(e_k)=n_k$，则有且仅有 $j_1,\cdots,j_{n_k}\in[n]$，使得 $e_k\in S_{j_s}(s=1,\cdots,n_k)$. 根据编码方式(6)，如果 $s=1,\cdots,n_k$，则有 $u_{j_sk}=1$，否则 $u_{j_sk}=0$. 因此 $\sum_{i=1}^{n}u_{ik}=n_k$ 成立. 反之，当 $\sum_{i=1}^{n}u_{ik}=n_k$ 时，恰有 $S_1,\cdots,S_n$ 中 n_k 个集合含有元素 e_k，因此 $e_k\in f_{TM}(S_1,\cdots,S_n)$，并且 $TM(e_k)=n_k$.　　证毕.

根据命题 4 给出的计算原理，并结合具有加法同态性的加密系统设计保密计算协议，协议的前 5 步与阈值并集计算协议完全相同. 首先计算出保密乘积数组 $(w_1,\cdots,w_n)$. 由于要保留阈值并集中每个元素在 $f_T(S_1,\cdots,S_n)$ 中出现的具体次数，对于协议 5 第 6 步的计算需要进行修改，在确定 e_k 属于阈值多重并集的同时，还要保留 w_k 的有关信息. 具体协议参看协议 6.

协议 6.　阈值多重并集计算协议.

输入：有 n 个半诚实参与者 $P_i(i\in[n])$，每个 P_i 拥有私密集合 S_i. 在具有加法同态性的门限密码体制中，所有参与者联合持有私钥 sk，其对应的公钥为 pk，并有密文 $E_l=\hat{E}_{pk}(l),l=1,\cdots,t-1$

输出：阈值多重并集 $f_{TM}(S_1,\cdots,S_n)$

1. 参与者 $P_i,i\in[n]$ 将 S_i 按照式(6)转化成数组 U_i；

2. 参与者 $P_i,i\in[n]$ 加密数组中的每个元素，加密后的数组记为 $C_i=(c_{i1},\cdots,c_{im})$；

3. P_1 将 $W_1=C_1$ 发送给 P_2；

4. 每一个 $P_i,i=2,\cdots,n$

(a) 从 P_{i-1} 处接收到 $W_{i-1}=C_1\cdots C_{i-1}$，

(b) 将 C_i 与 W_{i-1} 的对应分量相乘，得到新的乘积数组 $W_i=C_1\cdots C_{i-1}\cdot C_i$，

(c) 发送密文乘积数组 W_i 给 $P_{(i+1)\bmod n}$；

5. 参与者 P_1 公开 $W=C_1\cdot C_2\cdots C_n=(w_1,\cdots,w_m)$；

6. 对于每一个 $i\in[n],k\in[m]$

(a) P_i 选择随机数 q_{ik}，使得 $q_{ik}>n$，

(b) 计算 $v_k=w_k\cdot\hat{E}_{pk}\left(\left(\sum_{i=1}^{n}q_{ik}\right)(IsEq(w_k,E_1)+\cdots+IsEq(w_k,E_{t-1}))\right)$

7. 所有的参与者解密每一个 v_k，得到一个数值 $a_k(1\leq k\leq m)$；

8. 如果 $t\leq a_k\leq n$，则 e_k 是所求阈值多重并集 $f_{TM}(S_1,\cdots,S_n)$ 中的一个元素，且 $TM(e_k)=a_k$；

9. 输出阈值多重并集 $f_{TM}(S_1,\cdots,S_n)$.

5.5　协议 6 的正确性和安全性

关于协议 6 的正确性和安全性，我们有下面的定理.

定理 6.　协议 6 是正确的. 如果所用的公钥加密系统是语义安全的，协议 6 还是安全的，能够抵抗任意的合谋攻击.

定理 6 的证明与定理 5 类似，仅有的区别是在协议第 6 步的计算中，保证了当 e_k 在 n 个输入集合中出现的次数达到或超过阈值 t 时通过解密可得到 $\hat{D}_{sk}(w_k)$ 的具体数值，而如果 e_k 在 n 个输入集合中出现的次数低于阈值 t，对应的 $\hat{D}_{sk}(w_k)$ 将被随机化，从而无法获知这些元素在 n 个输入集合中出现的次数. 这样做既保证了协议 6 的正确性，又保证了安全性. 详细的证明过程省略.

6　协议的效率分析

计算效率分析. 首先分析上面各协议的计算复杂性，分析中忽略各协议执行中需要的数组乘法运算，只考虑最费时的模指数运算. 应用 ElGamal 公钥密码系统加密一次需要进行两次模指数运算，所有 n 个参与者共同解密一个密文需要进行 $n+1$ 次

模指数运算.我们也注意到,本文所有的编码过程都是参与者把集合转化成一个数组的过程,数组元素或为1,或为0,或为随机数.这些数组转化过程的计算复杂性与模指数运算相比也可忽略不计.

在协议1(或协议2)中,每个参与者首先要将自己的数据转化成一个 m 维数组,并对自己数组中取值为1的元素进行ElGamal加密,n 个参与者最多需要 $2nk$ 次模指数运算(假设各参与者集合最多含有 k 个元素);其次,每个参与者对前一个参与者传送过来的密文数组和自己的密文数组做乘法运算;最后,所有参与者对乘积密文数组中 m 个元素解密,需要进行 $(n+1)m$ 次模指数运算,所以协议1(或协议2)最多共需要 $(n+1)m+2nk$ 次模指数运算.

在协议3(或协议4)中,在参与者私密数组加密以及解密阶段参与者所作的运算与协议1(或协议2)相同,n 个参与者最多需要 $(n+1)m+2nk$ 次模指数运算;在协议3(或协议4)中,参与者 $P_{n-1},\cdots,P_1$ 对前一个参与者传送过来的密文数组元素要重新随机化(即乘以1的不同密文 $E_{pk}(1)$),并进行随机置换,共需要进行 $2(n-1)m$ 次模指数运算.所以协议3(或协议4)最多共需要 $(3n-1)m+2nk$ 次模指数运算.

在协议5(或协议6)中,在数组加密阶段参与者所作的运算与协议2相同,n 个参与者最多需要 $2nk$ 次模指数运算;在协议5(或协议6)中,参与者需要调用 $m(t-1)$ 次IsEq计算协议,由文献[48]知,当 t 为定数时需要模指数运算为 $O(nm)$;最后对 m 个密文 $v_1,\cdots,v_m$ 进行联合解密,共需要 $(n+1)m$ 次模指数运算.所以协议5(或协议6)需要模指数运算为 $O(nm)$.

通信效率分析. 协议1(或协议2)中,每个参与者将数组加密后的密文发送给下一个参与者需要一轮通信.在解密过程中,参与者合作解密数组元素也需要一轮通信,因此总共需要两轮通信.通信复杂性为 $O(nm)$.

在协议3(或协议4)中,每个参与者将数组加密后的密文发送给下一个参与者之后,还要将乘积数组随机化,共需要两轮通信.再加上参与者需要合作解密数组元素,因此协议3(或协议4)共需要3轮通信.通信复杂性为 $O(nm)$.

在协议5(或协议6)中,在数组加密阶段参与者所作的运算与协议2相同,而在协议5(或协议6)中,参与者还需要合作执行IsEq计算协议,最后对密文 $v_1,\cdots,v_m$ 进行解密,所以执行协议5(或协议6)共需要3轮通信.通信复杂性为 $O(nm)$.

我们将对本文协议和与本文关系密切的一些已有的较好结果进行比较,见表1.

表1 本文结果与已有结果的效率比较

文献	计算功能	计算复杂性	通信复杂性	参与方数量
[34]	PSI-CA PSU-CA	$O(m)$	$O(m)$	2
[32]	PSI,PSI-CA	$O(m)$	$O(m)$	2
[35]	PSI,PSU PSI-CA PSU-CA TPSU TMPSU	$O(n^2k^2)$	$O(n^2k)$	n
[38]	PSI	$O(nk+k(\log k)^2)$	$O(n^2k)$	n
[40]	PSI,PSU,PSI-CA PSU-CA	$O(nk\log(nk))$	$O(nk\log(nk))$	n
本文	PSI,PSU,PSI-CA PSU-CA TPSU TMPSU	$O(nm)$	$O(nm)$	n

在表1中,计算复杂性为模指数运算次数,通信复杂性为交互数据量.计算功能一栏为所列文献研究的具体内容,各符号分别表示:PSI为保密计算集合交集,PSU为保密计算集合并集,PSI-CA(PSU-CA)为保密计算交集(并集)的势,TPSU为保密计算阈值并集,TMPSU为保密计算阈值多重并集.

由于在很多应用场景(如投票选举,客户群求交/并等),有 $m\ll k^2$,$m\ll k\log(nk)$,在此情形下本文所设计协议的计算复杂性及通信复杂性都较低.特别地,如果有 $m=O(k)(|S_i|\leq k)$,则本文协议具有线性复杂性.另外,由于在本文协议中所涉及到的加密运算全部都是对数字1或0的加密,这些加密运算都可以离线进行,本文协议的在线复杂性可进一步降低,因此本文的协议是高效的协议.

协议效率实验测试. 前面已从理论上对本文所设计协议的效率进行了全面分析,并与已有相关结果进行了比较.下面进一步对于一个具体的交集计算问题进行实验测试,并将本文协议的执行结果与已有的效率较高的协议(文献[38]的交集协议)的执行结果进行比较.

(1)问题描述. 某单位有三位高层领导,计划在10个候选人中确定几个中层领导,三位领导心中各有自己信任的秘密人选,他们共同推荐的候选人首先当选.为了保密确定三位领导都信任的候选人,对10位候选人由1~10编号,并假设三位高层领导(记为参与者 P_1,P_2,P_3)所信任的候选人名单集合为 A,B,C,这就需要保密计算 $A\cap B\cap C$.下面计算中假设:

$A=\{1,2,3,4,5,6\}$, $B=\{3,4,5,6,7,8\}$,

$C=\{4,5,6,7,8,9\}$.

我们将应用本文所设计的协议 1 以及文献[38]在半诚实模型下的交集计算协议分别计算三个集合的交集，并比较执行过程的计算复杂性.

(2) 实验平台. Intel(R)Core(TM)M-5Y51 CPU @1.1GHz,1.3GHz、内存 8.00GB 64 位操作系统、安装 Windows10 家庭中文版 Java 版本 Myeclipse10.7.1.

首先生成 ElGamal 系统参数：素数 p：1024 bits，Z_p^* 的生成元 g，选取密钥 sk，求出对应的公钥 pk. 下面执行中的所有加密及解密运算都是在同一组参数下进行的，执行结果为 20 次计算结果的平均值.

(3) 实验结果. 详细的执行过程和操作说明请参看文后的附录 1. 由执行结果可知，应用本文协议 1 计算，整个协议执行过程平均需要 162.73 ms. 而应用文献[38]中半诚实模型下的交集协议，应用一些简化执行步骤后，执行过程的时间仍高于 301.9ms. 实验结果说明本文的协议是高效的协议.

7　恶意模型下的交集计算问题

7.1　协议设计

文献[10]利用比特承诺和零知识证明设计了一个协议编译器. 给定一个对于半诚实参与者安全的多方计算协议，该编译器可以自动生成一个对于恶意参与者也安全的多方计算协议. 编译器的基本原理是设法防止各种恶意行为，迫使各参与者像半诚实参与者一样按协议要求执行协议. 本部分将利用编译器的基本思想，以半诚实模型下保密计算交集的协议 1 为例，利用认证计算函数[10]以及零知识证明系统[49]，构造在恶意模型下也安全的交集计算协议.

为迫使各参与者像半诚实参与者一样按协议要求执行协议，根据协议 1 的构造，需要做到：(1) 防止 P_i 在进行数组乘积过程中偏离协议 1；(2) 保证在联合解密过程中各 P_i 提供正确的解密信息.

根据上面的要求，我们将在协议 1 的基础上构造恶意模型下的交集安全计算协议.

协议 7.　恶意模型下的交集计算协议.

输入：$P_1,\cdots,P_n$ 各自的秘密数据 $S_1,\cdots,S_n$；ElGamal 门限密码系统：每个 P_i 选择 x_i，并公布 $h_i=g^{x_i} \bmod p$，公钥为 $h=g^{x_1+\cdots+x_n} \bmod p$

输出：所有参与者集合的交集 $f_\wedge(S_1,\cdots,S_n)=\bigcap_{i=1}^n S_i$

1. 参与者 $P_i, i\in[n]$ 将 S_i 按照式(3)转化成数组 X_i；

2. 参与者 $P_i, i\in[n]$ 加密数组中的元素 1，使其与随机数不可区分，并公布加密后的数组 $C_i=(c_{i1},\cdots,c_{im})$；

3. 所有参与者计算 $W=C_1\cdots C_n=(w_1,\cdots,w_m)$；

4. 所有参与者联合解密 W 的每一个分量 $w_k=(u_k,v_k)$. 解密过程如下：对于 $k=1,\cdots,m$

(a) P_i 计算 $y_i=u_k^{x_i} \bmod p$，并公布，

(b) 应用零知识证明系统证明 y_i 的有效性，即验证 $y_i=u_k^{x_i} \bmod p$ 和 $h_i=g^{x_i} \bmod p$ 中的 x_i 是否相同. 若验证未全部通过，则中止协议. 若验证全部通过，则继续，

(c) P_i 计算 $a_k\equiv v_k\left[\prod_{i=1}^n y_i\right]^{-1} \bmod p$；

5. 如果 $a_k=1$，则 e_k 是所求交集中的一个元素；

6. 输出交集

$$f_\wedge(S_1,\cdots,S_n)=\{e_k \mid a_k=1, k=1,\cdots,m\}.$$

7.2　安全性分析

(1) 首先要求所有参与者 $P_i, i\in[n]$ 将 S_i 按照式(3)转化成数组 X_i 并对其中的元素 1 进行加密；如果在这个过程中有参与者未按原协议要求执行，这完全等同于 P_i 修改其原本规定的输入集合 S_i，而修改原始输入的行为本身是无法避免的，故不予考虑.

(2) 其次要求所有参与者公开他们所加密的数组 $C_i=(c_{i1},\cdots,c_{im}), i\in[n]$，所有参与者可以明确计算 n 个加密数组的乘积(对应分量相乘)，这样做要比协议 1 稍增加一些计算复杂性. 但在恶意模型下，如果按协议 1 中第 3～5 步的方法求乘积数组，若要防止参与者偏离协议，按文献[10]中编译器的设计，需要各参与者 P_i 对其加密数组 C_i 进行承诺，并进一步应用认证计算函数保证每个 P_i 所做乘积运算的正确性，如此做法计算复杂性更高.

(3) 最后阶段还需要正确执行解密. 这里需要应用零知识证明系统，根据 g,p 以及 h_i,y_i 的值证明 y_i 的有效性，若验证通过，则进一步由协议 7 第 4(c)步进行解密即可，若验证未能全部通过，则中止协议.

综上所述，协议 7 能迫使参与者以半诚实的方式正确计算集合交集，如有任何不当行为都会被发现而导致协议中止，在未中止情形下各方均可获得正确的输出 $f_\wedge(S_1,\cdots,S_n)=\bigcap_{i=1}^n S_i$. 限于篇幅，关于协议 7 安全性的严格证明在此省略. 仅叙述下面结论.

定理 7.　协议 7 是恶意模型下关于集合交集问题的安全计算协议.

8　协议的推广及实际应用系统举例

对于前面所设计的协议进行适当修改或者组合，或直接应用前面各协议的设计思想，对于更广泛的科学计算问题或实际应用问题，能够设计构造高

效安全的解决方案.下面举例进行说明.

8.1 多数据相等判别问题

参与者 $P_1,\cdots,P_n$ 各人分别拥有私密数据 $s_1,\cdots,s_n$,他们想合作保密判断这些数据是否相等.这个问题可直接应用上面的集合交集计算协议(协议 1)解决,这时可把每个数据 s_i 看成一个单元素集合,求其交集,若交集非空即说明这些数相等,否则不相等.

这个问题也可以利用前面的设计思想,将协议 1 进行适当修改,构造新的更简洁的解决方案.对于新的方案,仅叙述基本思路:

(1)参与者 P_1 将其具有的数据集合 $S_1=\{s_1\}$ 按照式(3)转化成数组 X_1,并加密数组中的元素 1,使其与随机数不可区分,加密后的数组记为 $C_1=(c_1,\cdots,c_m)$,并公开.

(2)对于每一个 $i=2,\cdots,n$,假设 P_i 具有的数据为 $s_i=z_{k_i}\in Z(k_i\in[m])$,$P_i$ 计算 $b_i=c_{k_i}E_{pk}(1)$.

(3)参与者合作计算 $B=b_2\cdots b_n$,并解密 B 得到 $D_{sk}(B)$.

(4)由于 $s_1=\cdots=s_n$ 当且仅当 $D_{sk}(B)=1$,由此得到判定结果.

8.2 集合包含判定问题

两个参与者 Alice 和 Bob 各有一个私密集合 A 和 B,双方需要合作保密判定 $A\subseteq B$ 是否成立,而不泄露 A,B 的信息.

注意到,$A\subseteq B$ 当且仅当 $A\cap B=A$ 或 $A\cup B=B$ 成立,因此这个问题可直接应用前面的集合交集或并集协议(协议 1 或协议 2)解决.

也可应用本文的思想方法,对协议 1 进行适当的修改,重新构造更高效的解决方案.对于新的方案,仅叙述基本思路:

(1)Alice 将其具有的集合 A 按照式(3)转化成数组 X,并加密数组中的元素 1,使其与随机数不可区分,加密后的数组记为 $C=(c_1,\cdots,c_m)$,并发送给 Bob.

(2)记 Bob 具有的私密集合为 $B=\{b_1,\cdots,b_v\}$,其中 $b_i=z_{k_i}\in Z(i\in[v],k_i\in[m])$,Bob 计算 $W=c_{k_1}\cdots c_{k_v}E_{pk}(1)$,并发送给 Alice.

(3)Alice 解密密文 W 得到 $D_{sk}(W)$.由于 $A\subseteq B$ 当且仅当 $D_{sk}(W)=1$,由此得到判定结果.

8.3 电子保密选举问题

匿名投票选举是一种很重要的选举形式,现场匿名选举相对简单.随着网络系统的广泛应用,电子保密投票选举问题的研究具有重要的理论意义和实际应用价值.下面简要说明如何应用本文所构造的保密计算协议,从技术层面研究解决两类电子匿名选举中的保密投票问题.

问题描述:(1)某个团体要以电子匿名投票方式保密选举有多个委员组成的委员会,要求候选委员必须得到规定的票数才能当选,由于该选举属于多人同职选举,选举结果要求只公布当选人名单,而对所有候选委员的得票数量保密;(2)如果进一步要由委员会成员匿名投票选出一个主任和若干个副主任.同样要求候选者达到一定票数方可当选,这时由于当选者的职位高低不同,要使当选者适得其所,选举结果要求公布当选者所获得的票数,而对其他落选者的票数保密.

为了应用本文的方法解决上述问题,首先将候选人编号,构成集合 $Z=\{z_1,\cdots,z_m\}$.所有投票成员记为 $P_1,\cdots,P_n$,他们对候选人编号进行投票,记 P_i 所投的多个候选人(编号)构成集合 $X_i\subseteq Z$,并且规定当选者所得票数不能少于阈值 t.

如此,问题(1)和问题(2)可分别应用前面构造的阈值并集以及阈值多重并集保密计算协议(协议 5 和协议 6)获得解决.

8.4 协议组合应用

考虑下面的问题:Alice 和 Bob 想知道他们两人共同的朋友而又不泄露各自的朋友圈.如果记 Alice 和 Bob 的朋友圈集合分别为 A、B,这个问题可以归约到保密计算 $A\cap B$.但 Alice 可能会担心,如果交集的势 $|A\cap B|$ 接近于自己集合的势 $|A|$,她的朋友圈信息将会泄露.因此事先约定只有当 $|A\cap B|$ 满足一定条件 T 时,才允许 B 获得交集结果.这个问题需要将交集势的计算和交集计算结合在一起进行解决.

解决思路如下:

(1)Alice 构造私钥/公钥对 sk/pk;

(2)Alice 将自己的集合 A 按照 $1-r$ 编码方式(3)编码成数组 X,加密其中的 1.发送给 Bob;

(3)Bob 也将自己的集合按照 $1-r$ 编码方式(3)编码成数组,并将 Alice 发来的数组中对应于自己 1 元素位置的元素挑选出来,并各乘以 $E_{pk}(1)$,构成一个新的密文数组(Bob 为隐藏自己集合的势,新密文数组中可添加一些随机数元素),随机置换该数组后将结果发给 Alice;

(4)Alice 解密 Bob 发来的每一个数组元素,解密结果为 1 的元素个数为两集合交集的势;

(5)Alice 根据自己集合的势和交集势的数值判断是否符合条件 T,若不符合,协议终止;若符合,把解密为 1 的元素位置告诉 Bob,Bob 即可获知

交集中的具体元素.

8.5 实际应用系统举例

本部分以交集的实际保密计算为例，简要说明前面所描述各协议的实际应用性. 为了使加密及解密过程更加清晰并易于验算，我们选取较小的模数，并对简单的输入集合进行运算，如此更加明了协议的实际操作过程.

假设有三个半诚实参与者 P_1,P_2,P_3，分别拥有集合 $A_1=\{2,3,5\},A_2=\{2,5,7\},A_3=\{1,2,5,6\}$，并令集合 $Z=\{z_1,\cdots,z_8\}=\{1,2,\cdots,8\}$. 现要保密计算三个集合的交集.

为此应用具有乘法同态性的 ElGamal 门限密码体制实现计算. 选取 ElGamal 密码系统的参数为：$p=19,g=2$.

(1) P_1,P_2,P_3 分别秘密选择私钥：$x_1=20,x_2=11,x_3=10$. 并公开：

$$g^{x_1}\equiv 11\bmod 19,\ g^{x_2}\equiv 7\bmod 19,\ g^{x_3}\equiv 8\bmod 19.$$

(2) P_1,P_2,P_3 计算得到公钥为

$$h\equiv 11\cdot 7\cdot 8\equiv 13\bmod 19.$$

(3) P_1,P_2,P_3 分别将其具有的集合按 1，r 方式编码成数组，并加密其中的 1，得到相应的加密数组为

$$C_1=[(3,7),(8,12),(8,7),(6,2),(16,4),(11,8),(17,4),(5,14)],$$

$$C_2=[(4,12),(13,14),(7,3),(5,4),(7,11),(2,6),(12,8),(5,9)],$$

$$C_3=[(15,2),(14,10),(9,3),(5,2),(3,15),(4,7),(6,11),(16,7)],$$

其中数组 C_1 中第 2,3,5 个数对为 1 的密文，加密选用的随机数分别为 $r=3,9,4$；数组 C_2 中第 2,5,7 个数对为 1 的密文，加密选用的随机数分别为 $r=5,6,15$；数组 C_3 中第 1,2,5,6 个数对为 1 的密文，加密选用的随机数分别为 $r=11,7,13,2$. 三个数组中其它数对均为随机数对.

(4) 参与者将数组顺次发送并相乘，最终得到乘积数组为

$$W=[(9,16),(12,8),(10,6),(17,16),(13,14),(12,13),(8,10),(1,8)].$$

(5) 联合解密过程

① 为了解密 $w_2=(u_2,v_2)=(12,8)$，P_1,P_2,P_3 首先分别公布：

$$u_2^{x_1}\equiv 11\bmod 19,\ u_2^{x_2}\equiv 8\bmod 19,\ u_2^{x_3}\equiv 7\bmod 19,$$

并计算得到：

$$a_2\equiv v_2\cdot(11\cdot 8\cdot 7)^{-1}\equiv 8\cdot 8^{-1}\equiv 1\bmod 19.$$

② 为了解密 $w_5=(u_5,v_5)=(13,14)$，P_1,P_2,P_3 首先分别公布：

$$u_5^{x_1}\equiv 17\bmod 19,\ u_5^{x_2}\equiv 2\bmod 19,\ u_5^{x_3}\equiv 6\bmod 19.$$

并计算得到：

$$a_5\equiv v_5\cdot(17\cdot 2\cdot 6)^{-1}\equiv 14\cdot 14^{-1}\equiv 1\bmod 19.$$

③ 类似解密其它 $w_k=(u_k,v_k)$，$k\neq 2,5$，得到的解密结果 $a_k\neq 1\bmod 19$.

因此，所求交集为 $A_1\cap A_2\cap A_3=\{2,5\}$.

9 结论与讨论

本文通过巧妙构造各种编码方法，将参与者的集合转化成一个数组，并结合具有加法或乘法同态性的加密算法，分别构造了包括集合的交集/并集及其势的计算，有关阈值并集计算等集合基本运算的保密计算协议. 应用严格的模拟范例方法证明了所有协议是安全的，并能抵抗任意的合谋攻击. 效率分析及实例验证都表明我们的协议是高效的协议.

由于本文所设计的协议是针对一般的集合运算设计的，这里应用的方法也完全能够处理多重集的有关计算问题. 如果某一元素 a 在一个多重集 A 中重复出现 k 次，我们可以约定以下面 k 个数据（‖ 为连字符）：

$$a,a\|1,a\|2,\cdots,a\|(k-1)$$

来表示在多重集 A 中所出现的 k 个重复元素 a. 通过这样的转化，本文中关于集合计算问题所设计的协议，可以推广到多重集的有关计算. 而其它文献中关于多重集运算所设计的很多计算协议却无法适用于一般的集合运算. 因此我们的协议具有更广泛的适用性.

本文提出的方案适合保密数据的范围已知，即各参与者集合含于某个确定的集合 Z，而 Z 的势不太大的情形，当 Z 的势很大时，协议的效率比较低. 恶意模型下的安全计算协议基本上是应用文献[10]中所提供的一般方法来设计的，复杂性较高. 今后将进一步研究保密数据的范围未知或者其所在范围很大（集合 Z 的势很大）情形下的多集合保密运算等科学计算问题，并设计恶意模型下高效的安全计算方案.

参 考 文 献

[1] Yao A C. Protocols for secure computations//Proceedings of the 23rd IEEE Symposium on 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] Gennaro R, Robshaw M. Advances in Cryptology-CRYPTO 2015, Part II. LNCS 9216. Heidelberg: Springer, 2015

[4] Robshaw M, Katz J. Advances in Cryptology-CRYPTO 2016. LNCS 9815. Heidelberg: Springer, 2016

[5] Fischlin M, Coron J S. Advances in Cryptology-EUROCRYPT 2016, Part II. LNCS 9666. Heidelberg: Springer, 2016

[6] 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

[7] Cramer R, Damgard I B, Nielsen J B. Secure Multiparty Computation. London: Cambridge University Press, 2015

[8] Yao A. How to generate and exchange secrets//Proceedings of the 23rd IEEE Symposiumon Foundations of Computer Science. Toronto, Canada, 1986: 162-167

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

[10] Goldreich O. The Fundamental of Cryptography: Basic Applications. London: Cambridge University Press, 2009

[11] Larraia E, Orsini E, Smart N P. Dishonest majority multi-party computation for binary circuits. Advances in Cryptology—CRYPTO 2014. LNCS 8617: 495-512

[12] Fagin R, Naor M, Winklery P. Comparing information without leaking it. Communications of the ACM, 1996, 39(4): 77-85

[13] Zhou Su-Fang, Dou Jia-Wei, Guo Yi-Min, et al. Secure multiparty vector computation. Chinese Journal of Computers, 2017, 40(5): 1134-1150(in Chinese)
(周素芳, 窦家维, 郭奕旻等. 安全多方向量计算. 计算机学报, 2017, 40(5): 1134-1150)

[14] Beimel A, Gabizon A, Ishai Y, et al. Non-interactive secure multiparty computation. Advances in Cryptology-CRYPTO 2014, LNCS 8617: 387-404

[15] Atallah M J, Du W. Secure multi-party computational geometry. Workshop on Algorithms and Data Structures 2001, LNCS 2125: 165-179

[16] Guo Yi-Min, Zhou Su-Fang, Dou Jia-Wei, et al. Efficient privacy-preserving interval computation and its applications. Chinese Journal of Computers, 2017, 40(7): 1664-1679(in Chinese)
(郭奕旻, 周素芳, 窦家维等. 高效的区间保密计算及应用. 计算机学报, 2017, 40(7): 1664-1679)

[17] Li S D, Wu C Y, Wang D S, Dai Y Q. Secure multiparty computation of solid geometric problems and their applications. Information Sciences, 2014, 282: 401-413

[18] Qin J, Duan H, Zhao H, et al. A new Lagrange solution to the privacy-preserving general geometric intersection problem. Journal of Network and Computer Applications, 2014, 46: 94-99

[19] Du W, Atallah M J. Privacy-preserving cooperative statistical analysis. IEEE Annual Computer Security Applications Conference. New Orleans, Louisiana, USA, 2001: 102-110

[20] Bogdanov D, Kamm L, Laur S, et al. Privacy preserving statistical data analysis on federated databases//Proceedings of the Privacy Technologies and Policy. Athens, Greece, 2014: 30-55

[21] Lin B R, Wang Y, Rane S. A framework for privacy preserving statistical analysis on distributed databases. IEEE International Workshop on Information Forensics and Security (WIFS). Costa Adeje-Tenerife, Spain, 2012: 61-66

[22] Agrawal R, Srikant R. Privacy-preserving data mining. ACM Sigmod Record, 2000, 29(2): 439-450

[23] Lindell Y, Pinkas B. Privacy preserving data mining. Advances in Cryptology-CRYPTO 2000, LNCS 1880: 36-54

[24] Yi X, Rao F Y, Bertino E, Bouguettaya A. Privacy-preserving association rule mining in cloud computing//Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security. Singapore, 2015: 439-450

[25] Luo Yong-Long, Huang Liu-Sheng, Jing Wei-Wei, et al. Privacy-preserving cross product protocol and its applications. Chinese Journal of Computers, 2007, 30(2): 248-254 (in Chinese)
(罗永龙, 黄刘生, 荆巍巍等. 保护私有信息的叉积协议及其应用. 计算机学报, 2007, 30(2): 248-254)

[26] Zhou Shui-Geng, Li Feng, Tao Yu-Fei, Xiao Xiao-Kui. Privacy preservation in database applications: A survey. Chinese Journal of Computers, 2009, 32(4): 847-861 (in Chinese)
(周水庚, 李丰, 陶宇飞, 肖小奎. 面向数据库应用的隐私保护研究综述. 计算机学报, 2009, 32(4): 847-861)

[27] Li S D, Dai Y Q, Wang D S, et al. Comparing two sets without disclosing them. Science in China Series F: Information Sciences, 2008, 51(9): 1231-1238

[28] Freedman M J, Nissim K, Pinkas B. Efficient private matching and set intersection. Advances in Cryptology-EUROCRYPT 2004, LNCS 3027: 1-19

[29] Dachman-Soled D, Malkin T, Raykova M, et al. Efficient robust private set intersection. Applied Cryptography and Network Security 2009, LNCS 5536: 125-142

[30] Hazay C, Nissim K. Efficient set operations in the presence of malicious adversaries. Public Key Cryptography 2010, LNCS 6056: 312-331

[31] De Cristofaro E, Kim J, Tsudik G. Linear-complexity private set intersection protocols secure in malicious model. Advances in Cryptology—ASIACRYPT 2010, LNCS 6477: 213-231

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

[33] Jarecki S, Liu X. Fast secure computation of set intersection. Security and Cryptography for Networks 2010, LNCS 6280: 418-435

[34] De Cristofaro E, Gasti P, Tsudik G. Fast and private computation of cardinality of set intersection and union. Cryptology and Network Security 2012, LNCS 7712: 218-231

[35] Kissner L, Song D. Privacy-preserving set operations. Advances in Cryptology-CRYPTO 2005, LNCS 3621: 241-257

[36] Sang Y, Shen H. Privacy preserving set intersection based

on bilinear groups//Proceedings of the 31st Australasian Computer Science Conference. Wollongong, Australia, 2008: 47-54

[37] Sang Y, Shen H. Efficient and secure protocols for privacy-preserving set operations. ACM Transactions on Information and System Security (TISSEC), 2009, 13(1): 9:1-9:35

[38] 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

[39] Vaidya J, Clifton C. Secure set intersection cardinality with application to association rule mining. Journal of Computer Security, 2005, 13(4): 593-622

[40] Blanton M, Aguiar E. Private and oblivious set and multiset operations. International Journal of Information Security, 2016, 15(4): 493-518

[41] Egert R, Fischlin M, Gens D, et al. Privately computing set-union and set-intersection cardinality via bloom filters. Australasian Conference on Information Security and Privacy. 2015, LNCS 9144: 413-430

[42] Du W L. A Study of Several Specific Secure Two-Party Computation Problems [Ph. D. dissertation]. Purdue University, 2000

[43] Cramer R, Damgård I, Maurer U. General secure multi-party computation from any linear secret-sharing scheme. Advances in Cryptology-EUROCRYPT 2000, LNCS 1807: 316-334

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

[45] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory, 1985, 31(3): 469-472

[46] Desmedt Y, Frankel Y. Threshold cryptosystems. Advances in Cryptology—CRYPTO'89 Proceedings. 1989, LNCS 435: 307-315

[47] Long Y, Chen K, Mao X. New constructions of dynamic threshold cryptosystem. Journal of Shanghai Jiaotong University (Science), 2014, 19: 431-435

[48] Kissner L, Oprea A, Reiter M K, et al. Private keyword-based push and pull with applications to anonymous communication. Applied Cryptography and Network Security 2004, LNCS 3089: 16-30

[49] Chaum D, et al. An improved protocol for demonstrating possession of discrete logarithms and some generalizations. Workshop on the Theory and Application of Cryptographic Techniques, Amsterdam, the Netherlands, 1987: 127-141

附录 1.

应用本文的协议 1 以及文献[38]的交集计算协议分别对集合 $A=\{1,2,3,4,5,6\}$, $B=\{3,4,5,6,7,8\}$ 和 $C=\{4,5,6,7,8,9\}$ 计算交集(为简单起见,两个协议中的阈值解密都应用 sk 直接解密代替,执行结果为 20 次计算的平均值).

(1) 本文半诚实模型下交集计算协议 1 的执行情况

① 令 $Z=\{1,2,\cdots,10\}$,首先将三个集合按式(3)编码成 10 维数组(简单运算,忽略计算花费);

②将各数组分别应用公钥 pk 加密,需要 3 组加密运算,每组加密获得 6 个 1 的密文 $E_{pk}(1)$,共耗时 28.83+28.89+28.71=86.43(ms);

③将三个密文数组对应项相乘,得到一个 10 维乘积数组(乘法运算,忽略计算花费);

④分别解密上面 10 个数组元素,耗时 76.3 ms.

⑤解密结果为 1 的元素位置为交集中的元素,因此获得交集 $A\cap B\cap C=\{4,5,6\}$.

执行结果. 整个协议执行过程近似花费为 86.43+76.3=162.73(ms).

(2) 文献[38]中半诚实模型下交集计算协议的执行情况(简化计算情形)

①应用集合 A,B,C 构造多项式分别为(忽略计算花费)

$$f_1(x)=(x-1)(x-2)(x-3)(x-4)(x-5)(x-6),$$

$$f_2(x)=(x-3)(x-4)(x-5)(x-6)(x-7)(x-8),$$

$$f_3(x)=(x-4)(x-5)(x-6)(x-7)(x-8)(x-9),$$

并记 $I(x)=f_1(x)+f_2(x)+f_3(x)$.

②将上面多项式用点表示法进行表示(忽略计算花费)

令 $S=\{s_1,\cdots,s_7\}=\{11,\cdots,17\}$,对于 $i=1,2,3$,根据 $PR_i=[(s_1,f_i(s_1)),\cdots,(s_7,f_i(s_7))]$,将多项式用点表示法分别表示为

$PR_1=[(11,151200),(12,332640),(13,665280),(14,1235520),$
$(15,2162160),(16,3603600),(17,5765760)]$,

$PR_2=[(11,20160),(12,60480),(13,151200),(14,332640),$
$(15,665280),(16,1235520),(17,2162160)]$,

$PR_3=[(11,5040),(12,20160),(13,60480),(14,151200),$
$(15,332640),(16,665280),(17,1235520)]$.

③应用 ElGamal 修改系统加密 $f_i(s_k)$,记其密文为 $c_{ik}=\hat{E}_{pk}(f_i(s_k))$,得到

$$\hat{E}_{pk}(PR_i):=[(s_1,c_{i1}),\cdots,(s_7,c_{i7})],$$

这部分运算花费时间为 142.8 ms.

④将上面密文数组按下面方式作乘积(忽略计算花费)

$$U=((s_1,C_1),\cdots,(s_7,C_7))$$

$$=\left[\left(s_1,\prod_{i=1}^{3}c_{i1}\right),\cdots,\left(s_7,\prod_{i=1}^{3}c_{i7}\right)\right].$$

⑤解密 $C_1,\cdots,C_7$,得到的相应明文为

$$g^{I(s_1)},\cdots,g^{I(s_7)},$$

这部分运算花费时间为 159.1 ms.

⑥利用 $[(s_1,g^{I(s_1)}),\cdots,(s_7,g^{I(s_7)})]$ 重构 $F(x)=g^{I(x)}$(这部分未计时).

⑦参与者 P_i 需要将其集合中的每个元素 x_{ik} 代入 $F(x)$ 中进行计算,根据

$$F(x_{ik})=1\Leftrightarrow I(x_{ik})=0\Leftrightarrow x_{ik}\in A\cap B\cap C$$

求得交集 $A\cap B\cap C=\{4,5,6\}$(这部分未计时).

执行结果. 仅考虑第 3,5 步的计算就需要花费 142.8+159.1=301.9(ms).

补充说明. 在执行文献[38]中半诚实模型下交集计算

协议时很多步骤用了简化计算.原计算协议的计算原理是:各参与者首先将其集合表示成多项式(集合元素为多项式的根,为 6 次多项式),其次再将所构造的多项式分别乘一个随机多项式(同样为 6 次),之后再求所得乘积多项式的和,结果为一个 12 次多项式(这样做是为了防止在第 7 步出现"增根"的问题,即有 x_{ik} 满足 $F(x_{ik})=1$(等价于 $I(x_{ik})=0$),但 x_{ik} 却不属于交集),如此得到的多项式相当于我们上面简化执行中的 $I(x)$.进一步,如果 $I(x)$为 12 次多项式,协议中选择点集 S 至少应包含 13 个元素,这样的结果导致第 2 步中应用点表示时的数据更大,并且在第 3 步中需要的加密运算以及第 5 步需要的解密运算次数几乎都翻倍.又由于文献[38]中协议需要应用具有加法同态性的 ElGamal 修改系统进行加密及解密(在 ElGamal 修改系统下对 M 加密实质上是应用 ElGamal 加密系统对 g^M 加密,如果 M 比较大,这相当于又增加一次模指数运算).而文献[38]中最后第 6,7 两步的工作也较复杂,上面协议执行中没有考虑此两项花费.

从上面的执行结果及补充说明可看到,即使对文献[38]的协议进行简化计算,并且仅考虑第 3,5 步的计算花费,其计算复杂性就远高于本文协议 1 的执行结果.此例说明,本文的协议是高效实用的协议.

DOU Jia-Wei, born in 1963, Ph. D., associate professor. Her main research interests are applied mathematics and applied cryptography.

LIU Xu-Hong, born in 1992, M. S. candidate. Her main research interests are applied mathematics and applied cryptography.

ZHOU Su-Fang, born in 1990, Ph. D. candidate. Her main research interests are cryptography and information security.

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

Background

Secure multiparty computation (SMC) was first introduced by Yao and is currently a research focus in the international cryptographic community. SMC is a crucial privacy-preserving technology in cyberspace. Since SMC was introduced, cryptographic scholars have studied many SMC problems that have arisen in various fields such as scientific computation, statistical analysis, data mining, computational geometry. But many solutions to these problems are computationally inefficient. There are many new problems either requiring study or appealing more efficient solutions, or both.

This paper studies secure set operations. These problems have been investigated in the literatures. The famous methodology to solve these problems is to represent a private set as a polynomial and to reduce the secure set operations to secure polynomial operations. These solutions are very innovative, but they are very inefficient and are difficult to extend to more than two private sets. In this paper we develop a completely new framework to efficiently solve secure set operations.

By introducing new encoding schemes and using the ElGamal homomorphic encryption, we develop protocols for the intersection set and the union set of private sets, the cardinality of the intersection set and the union set of private sets, and the threshold set of private sets. We proved that these protocols are secure in the semi-honest model using the simulation paradigm, and that they can resist collusion attack of any number of parties. If all the private sets are subsets of a universal set which is known to all the parties, our protocols are suitable for more than two private sets, and are very efficient. These protocols are of theoretical importance and have practical importance in SMC and other privacy-preserving computations.

Analyses indicate that our protocols are computationally efficient compared with the existing protocols. We give some examples to illustrate how the new protocols can be used to construct protocols for other secure multiparty computation problems such as privately determining (1) whether more than two numbers are equal, (2) whether one private set is a subset of another private set, (3) how to implement secure voting, and (4) how to compose these protocols to solve some complicate problems. The simulation results of the protocols also show that our protocols are efficient.

This study is sponsored by the National Natural Science Foundation of China under Grant No. 61272435. The aim of these projects is to address various privacy problems arising in cyberspace. Our team has been engaged in the design and analysis of cryptographic protocols, such as SMC, secret sharing, bit commitment, zero-knowledge proof, digital signature, and secure scientific computation, secure computational geometry over 10 years. We have published over 50 papers, of which near 30 have been indexed by SCI.