Computer Science and Application 计算机科学与应用, 2021, 11(6), 1649-1661
Published Online June 2021 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2021.116170

Hans汉斯

复数域上的隐私保护集合交集势计算

唐　桐[1,2]，李　轶[1]，吴文渊[1]，冯　勇[1]

[1]中国科学院重庆绿色智能技术研究院，重庆
[2]中国科学院大学，北京
Email: tangtong@cigit.an.cn

收稿日期：2021年5月7日；录用日期：2021年6月3日；发布日期：2021年6月10日

摘　要

安全多方计算是密码学领域的研究热点，集合交集问题是其重要研究方向之一。由于现实中众多应用场景可以抽象成数学上的集合关系计算，故集合保密计算具有重要的实际意义。整数环上的集合交集保密计算目前已存在较多研究成果，但针对复数域上集合交集保密计算的研究却很少。本文主要研究复数域上的集合保密计算问题。由于复数上的集合交集计算与整数上的集合交集计算存在差异，故本文引入代数计算中的结式来解决该问题。首先使用牛顿公式来构造复数域上的多项式，再将保护隐私的两方求解集合交集势问题转化为安全两方求解结式中参变元最低次数问题。最后，利用随机哈希和西尔维斯特结式的性质降低计算开销。

关键词

安全多方计算，隐私保护集合交集计算，复数，结式

The Computation of Private Set Intersection Cardinality over Complex Number

Tong Tang[1,2], Yi Li[1], Wenyuan Wu[1], Yong Feng[1]

[1]Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing
[2]University of Chinese Academy Science, Beijing
Email: tangtong@cigit.an.cn

Received: May 7th, 2021; accepted: Jun. 3rd, 2021; published: Jun. 10th, 2021

Abstract

Secure multi-party computation is a hotspot in the field of cryptography, and the problem of private set intersection is a significant research direction. Since many practical problems can be ab-

文章引用: 唐桐, 李轶, 吴文渊, 冯勇. 复数域上的隐私保护集合交集势计算[J]. 计算机科学与应用, 2021, 11(6): 1649-1661. DOI: 10.12677/csa.2021.116170

stracted as set relation computation in mathematics, the computation of private set intersection has important practical significance. There are many works to study secure integer set computation, but there are little researches in the complex field. This paper focuses on the computation of private set intersection in the complex number. Because of the difference between complex numbers and integers, we need to find a method to compute complex numbers. For this reason, we use the polynomial theory to construct the related polynomial according to the set elements, and introduce resultant in algebraic calculations to transform the two-party private set intersection cardinality problem into a problem of calculating the lowest order of the parameter argument, finally, use random hashing and the properties of Sylvester resultant to calculate it quickly, further reducing the computational overhead.

Keywords

Secure Multi-Party Computation, Private Set Intersection Computation, Complex Number, Resultant

1. 引言

安全多方计算(Secure Multi-Party Computation, SMC)指的是在不泄露各方私有数据的前提下，正确完成需要各方合作计算的函数值。安全多方计算问题是由姚期智在 1982 年提出的[1]。后来，Goldreich 对该问题进行了研究[2] [3]，证明一般的安全多方计算问题是可解的，并给出了解决方案。但是同时，他也指出这些具有一般性的理论在很多情况下由于效率低下无法应用在实际中，而应该根据研究问题的不同设计不同的解决方案。

隐私保护集合交集(Private Set Intersection, PSI)计算是安全多方计算领域的一个重要研究方向。它指的是各自拥有集合的多方协议参与者希望保密计算出这些集合之间的关系，如集合交集、交集势[4]-[10]和并集、并集势[11] [12] [13]等计算问题。

随着社会的发展，人们对个人数据安全越来越重视，隐私保护集合交集计算也随之受到更多的关注、被应用在更多的现实场景中。如：在商业应用场景中，APP 想要通过用户手机通讯录中的联系人信息了解到这些人中曾经多少人下载并使用过本 APP，这需要计算通讯录联系人与 APP 服务器数据库中用户信息的交集势。为了保护用户隐私，APP 不可以将用户手机通讯录中的联系人信息进行上传，故需要寻求一种能够在不泄露用户联系人信息的情况下，计算出该交集势的方法。

由于隐私保护集合交集计算受到越来越多的关注，一些学者也提出了自己的解决方案。这些方案大致可以分为以下几类。

第一类是基于公钥加密体系的 PSI。2002 年，Clifton 等人在文献[14]中使用可交换加密算法 Pohlig-Hellman [15]对协议双方集合元素均加密两次，将问题转化为集合元素匹配问题。2004 年，Freedman 等人在文献[6] [16]利用插值法将原集合元素作为根构造出多项式，将集合交集问题转化为对方集合元素是否为多项式的根的问题，在此基础上，利用 Paillier [17]或 El Gamal [18]同态加密算法对多项式系数加密，从而求解集合的交、集合交集势等问题。2005 年，Kissner 等人在文献[19]中根据集合元素构造多项式，使用门限同态加密方法求解集合交集、并集势和集合包含问题，但由于使用门限解密，导致计算量

较高。2012 年，Xia 等人在文献[20]中使用基于 LWE 的加密算法判断元素与集合的关系，并基于该方法，用遍历元素的思路求解了集合交集问题。2020 年，窦家维等人在文献[21]中，借助三角形面积公式的计算，将有理数域上的元素与集合关系问题转化为整数范围内向量内积问题，并进一步结合 Paillier [17]加密算法设计了集合运算的保密计算协议。2020 年，巩明林等人在文献[22]中基于非对称加密，提出了一种面向有理数集合的、无匹配差错的两方保密集合交集势计算协议。

第二类是基于电路的 PSI。2012 年，Huang 等人在文献[23]中提出了三种基于姚氏混乱电路 PSI 协议的实现，分别适用于不同的集合规模和集合定义域大小的计算。2014 年 Pinkas 等人对 Huang 等人提出的协议分别使用随机 OT 协议和哈希表进行优化[8] [24]。

第三类是基于不经意传输(OT)协议的 PSI。2013 年 Dong 等人在文献[9]中使用布隆过滤器(Bloom Filter, BF)、扩展的 OT 协议来构造可以处理集合元素达到亿级别大小的 PSI 协议。2020 年，Lv 等人在文献[25]中解决集合交集势问题时，用可交换加密算法对明文加密，利用 Bloom 过滤器在两方参与者间传递密文，并证明在发送方数据集足够大时，协议是安全的，并且计算成本较低。

综上可以看出，众多学者使用不同的方法解决 PSI 问题。尽管这些方案协议各不相同，但是大多数协议均是将集合元素视为一维整数。目前关于多维数据的 PSI 问题研究较少，而日常生活中，关于多维数据的保密计算应用场景十分广泛。例如，保险公司迎来几位想要买疾病险的客户，而公司规定曾经有过某几种疾病史的人不可以购买该保险，此时，保险公司需要将客户信息以及规定的几种疾病与医院数据库进行隐私保护的数据求交；又例如，商家想要知道，自己通过广告公司在互联网上投放的不同商品的广告效益如何，需要将购买过商品的用户信息、商品信息与广告公司通过用户点击商品链接获得的用户信息、商品信息进行隐私保护的数据交集势求解；再例如，同一平面上，Alice 和 Bob 各自拥有点集，他们想要知道双方是否存在相同的点。

正因为多维数据的应用具有广泛的应用场景，故本文提出一种求解二维数据的 PSI 协议，首先将二维数据视为复数，根据这些复数使用 2.2 节的牛顿公式构造多项式，将问题转化为复数域上的多项式公根问题，再利用可以求解复数多项式公根的结式，将问题转化为复数域上的两方结式保密计算，以此来求解该问题。

2. 基本概念

2.1. 问题描述

集合成员关系的多方安全计算

Alice 拥有集合 $S_a=\{\alpha_1,\alpha_2,\cdots,\alpha_m\}$，Bob 拥有集合 $S_b=\{\beta_1,\beta_2,\cdots,\beta_n\}$。Alice 和 Bob 双方希望可以协作计算出 S_a 与 S_b 两集合交集的势，即 $|S_a\cap S_b|$，但要求在协作计算的过程中任何一方不可以从计算过程和结果中获取对方的私有集合元素信息。

2.2. 牛顿公式

设

$$f(x)=a_0x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n$$

是域 K 上的 n 次多项式，如果 $a_0\neq 0$，则在复数域中它有 n 个根，记为(其中重根出现的次数等于其重数)：

$$\alpha_1,\alpha_2,\cdots,\alpha_n$$

用 s_p 来记这些根的 p 次等幂和：

$$s_p=\sum_{j=1}^{n}\alpha_j^p,\quad p=0,1,2,\cdots,n$$

特别的，$s_0 = n$。

设当$k < 0$或$k > n$时，$a_k = 0$，则牛顿公式[26]为：

$$a_{k-1}s_1 + a_{k-2}s_2 + \cdots + a_0 s_k = -ka_k,\ k = 1,2,\cdots$$

可以通过上式看出，牛顿公式能够有效描述多项式根与系数的关系。

2.3. 结式

2.3.1. 西尔维斯特结式

给定两个多项式：

$$f(x) = a_0 x^m + a_1 x^{m-1} + \cdots + a_m$$

$$g(x) = b_0 x^n + b_1 x^{n-1} + \cdots + b_n$$

记

$$M_0 = \begin{pmatrix} a_0 & a_1 & \cdots & a_m & & & \\ & a_0 & a_1 & \cdots & a_m & & \\ & & \cdots & \cdots & \cdots & \cdots & \\ & & & a_0 & a_1 & \cdots & a_m \\ b_0 & b_1 & \cdots & \cdots & b_n & & \\ & b_0 & b_1 & \cdots & \cdots & b_n & \\ & & \cdots & \cdots & \cdots & \cdots & \cdots \\ & & & b_0 & b_1 & \cdots & \cdots & b_n \end{pmatrix}$$

M_0是一个$m+n$阶的方阵，成为多项式$f(x)$与$g(x)$关于x的西尔维斯特结式矩阵[26]，记为$Sylvester(f,g,x)$。它的行列式记为$res(f,g,x)$，即：

$$res(f,g,x) = det(Sylvester(f,g,x)) = det(M_0)$$

称为多项式$f(x)$与$g(x)$关于x的结式。其中，$det(M_0)$表示计算矩阵M_0的行列式。

2.3.2. 西尔维斯特结式的性质

多项式$f(x)$与$g(x)$、$res(f,g,x)$如上定义。

设多项式$f(x)$的m个根分别为：

$$\alpha_1, \alpha_2, \cdots, \alpha_m$$

定理 1：$res(f,g,x) = a_0^n \prod_{i=1}^m g(\alpha_i)$。

定理 2：如果$a_0 \neq 0$，则$f(x)$与$g(x)$有公根的充分必要条件是$res(f,g,x) = 0$。

定理 3：设λ是一个参变元，结式$res(f,g+\lambda,x)$称为$f(x)$与$g(x)$的λ结式。它是关于λ的一个多项式。那么，$f(x)$的m个根中恰好有k个使得$g(x) = 0$的充要条件是结式$res(f,g+\lambda,x)$关于λ的最低次数为k。

2.4. 安全矩阵乘积协议

Du 在文献[27]中提出了一种安全矩阵乘积协议。该协议在双方私有数据以矩阵形式表示时，提供了一种安全可靠的方法来帮助双方完成矩阵计算并保证信息的安全，可将其描述为：

Alice 拥有私有数据$m \times p$大小的矩阵A，Bob 拥有私有数据$p \times n$大小的矩阵B，执行安全矩阵乘积协议后，Alice 可得到V_A，Bob 可得到V_B，满足$V_A + V_B = A \cdot B$，并且需要保证在计算的过程中不泄露 Alice、

Bob 双方私有信息矩阵中的数值给对方。

定义协议 1 如下：

输入：Alice 拥有私有数据 $m\times p$ 大小的矩阵 A，Bob 拥有私有数据 $p\times n$ 大小的矩阵 B。

输出：Alice 获得 V_A，Bob 获得 V_B，满足 $V_A+V_B=A\cdot B$。

1) Alice、Bob 联合生成一个随机可逆 $p\times p$ 大小的矩阵 M。

2) 在垂直方向上，将 M 从中间分割成左子矩阵 M_{left} 与右子矩阵 M_{right}。

Alice 计算 $A_1=A\cdot M_{left}$，$A_2=A\cdot M_{right}$。

然后将 A_1 发送给 Bob。

3) 在水平方向上，将逆矩阵 M^{-1} 从中间分割成上子矩阵 M_{top}^{-1} 与下子矩阵 M_{bottom}^{-1}。

Bob 计算 $B_1=M_{top}^{-1}\cdot B$，$B_2=M_{bottom}^{-1}\cdot B$，$V_B=A_1\cdot B_1$。

然后将 V_B 和 B_2 发送给 Alice。

Alice 计算 $V_A=A_2\cdot B_2$，由 $A\cdot B=V_A+V_B$，即可得到 $A\cdot B$ 的结果，最后 Alice 将结果发送给 Bob。

2.5. 安全多方计算的安全性定义

2.5.1. 半诚实参与者

半诚实参与者指的是，其在参与执行安全多方计算协议的过程中会忠实履行该协议，不会恶意篡改输入和输出的信息，但其可能会保留协议中间过程的计算结果，并尝试推导出他人的私有信息。

实际上，安全多方计算的协议运行环境分为半诚实参与者模型与恶意攻击者模型，但 Goldreich 在[3]中利用比特承诺和零知识证明设计了一种编译器。该编译器可以将在半诚实参与者条件下保密计算函数 f 的协议 π 自动生成在恶意参与者条件下保密计算函数 f 的协议 π'。π' 将迫使恶意参与者以半诚实方式参与协议的执行，否则就会被发现。故大多数时候，只需要设计半诚实参与者模型下的安全协议即可。本文协议的参与者都是半诚实的。

2.5.2. 半诚实模型下的安全性定义

设 $f(x,y)$ 是概率多项式函数，π 是计算 f 的协议。假设 Alice 拥有 x，Bob 拥有 y，如果他们需要在不向对方暴露 x、y 的前提下合作计算函数 $f(x,y)=(f_1(x,y),f_2(x,y))$。计算目的是为了让 Alice 和 Bob 分别得到 f 的两个分量 $f_1(x,y)$ 与 $f_2(x,y)$。Alice 在执行计算的过程中所得到的视图记作 $view_1(x,y)$，输出记作 $output_1(x,y)$；Bob 得到的视图记为 $view_2(x,y)$，输出记为 $output_2(x,y)$。Goldreich 在[3]中给出了半诚实参与者模型下的安全两方计算的计算不可区分性的形式化定义，如下：

定义 1.对于概率多项式函数 f，如果存在概率多项式时间模拟器算法 S_1 与 S_2 使得：

$$\left(S_1(x,f_1(x,y)),f_2(x,y)\right)\overset{c}{=}\left(view_1(x,y),output_2(x,y)\right)$$

$$\left(f_1(x,y),S_2(x,f_2(x,y))\right)\overset{c}{=}\left(output_1(x,y),view_2(x,y)\right)$$

上两式同时成立。其中，$\overset{c}{=}$ 表示计算不可分。

该定义说明了两方参与者视图中的信息均只能来自于自己的输入和获得的输出。这样可以保证任何一方得不到其他方的私有信息。

3. 安全协议

3.1. 问题转化

集合交集势问题定义如下：

Receiver 拥有集合 $S_a=\{\alpha_1,\alpha_2,\cdots,\alpha_m\}$，Sender 拥有集合 $S_b=\{\beta_1,\beta_2,\cdots,\beta_n\}$，其中，集合元素均为复数。需要在不泄露 S_a 与 S_b 的前提下，计算出 $|S_a\cap S_b|$ 的结果。

该问题可以进行如下转化。

Receiver 拥有集合 $S_a=\{\alpha_1,\alpha_2,\cdots,\alpha_m\}$，此集合可以通过 2.2 节的牛顿公式计算为一个 m 次多项式：

$$f(x)=(x-\alpha_1)(x-\alpha_2)\cdots(x-\alpha_m)=a_0x^m+a_1x^{m-1}+\cdots+a_m$$

该多项式的根分别为 $\alpha_1,\alpha_2,\cdots,\alpha_m$。显然，$a_0=1$。

同理，Sender 拥有集合 $S_b=\{b_1,b_2,\cdots,b_n\}$ 也可以通过 2.2 节的牛顿公式计算为一个 n 次多项式：

$$g(x)=(x-\beta_1)(x-\beta_2)\cdots(x-\beta_n)=b_0x^n+b_1x^{n-1}+\cdots+b_n$$

上述通过根来求多项式系数可以通过则 S_a 与 S_b 的集合交集势问题可以转化为多项式 $f(x)$ 与 $g(x)$ 存在几个公共根的问题，而两个多项式的公根问题可以借助西尔维斯特结式来进行求解。

首先，根据 2.2.2 节定理 3，通过 $f(x)$ 与 $g(x)$ 的表达式，构造出西尔维斯特结式 $res(f,g+\lambda,x)$。

第二步，根据定理 1 可以得到 f 与 $g+\lambda$ 关于西尔维斯特结式的计算结果：

$$res(f,g+\lambda,x)=a_0^n\prod_{i=1}^{m}(g(\alpha_i)+\lambda)=a_0^n\prod_{i=1}^{m}(b_0\alpha_i^n+b_1\alpha_i^{n-1}+\cdots+b_n+\lambda)$$

第三步，由于 $a_0=1$，故只需使用安全两方协议计算出 $\prod_{i=1}^{m}(g(\alpha_i)+\lambda)$ 的每一项，即 $g(\alpha_i)+\lambda$，$i=1,\cdots,m$。由于

$$\prod_{i=1}^{m}(g(\alpha_i)+\lambda)=\prod_{i=1}^{m}(b_0\alpha_i^n+b_1\alpha_i^{n-1}+\cdots+b_n+\lambda)$$

故考虑构造下列矩阵和向量：

$$A=\begin{bmatrix}1&\alpha_1&\alpha_1^2&\alpha_1^3&\cdots&\alpha_1^n\\1&\alpha_2&\alpha_2^2&\alpha_2^3&\cdots&\alpha_2^n\\\vdots&\vdots&\vdots&\vdots&&\vdots\\1&\alpha_m&\alpha_m^2&\alpha_m^3&\cdots&\alpha_m^n\end{bmatrix}$$

$$B=\begin{bmatrix}b_n+\lambda\\b_{n-1}\\b_{n-2}\\\vdots\\b_0\end{bmatrix}$$

用安全矩阵乘积协议计算出 AB：

$$AB=\begin{bmatrix}b_0\alpha_1^n+b_1\alpha_1^{n-1}+\cdots+b_n+\lambda\\b_0\alpha_2^n+b_1\alpha_2^{n-1}+\cdots+b_n+\lambda\\b_0\alpha_3^n+b_1\alpha_3^{n-1}+\cdots+b_n+\lambda\\\vdots\\b_0\alpha_m^n+b_1\alpha_m^{n-1}+\cdots+b_n+\lambda\end{bmatrix}$$

再将 AB 中的每个多项式相乘，就得到了 $\prod_{i=1}^{m}(g(\alpha_i)+\lambda)$，最后查看关于 λ 的最低次数 k 是多少。此时的 k，即为 $|S_a\cap S_b|$。

3.2. 具体协议

下面给出在半诚实模型下，安全计算两集合交集势的具体协议。

协议 2. 安全计算集合交集势协议

输入：Receiver 保密输入集合 $S_a=\{\alpha_1,\alpha_2,\cdots,\alpha_m\}$，Sender 保密输入集合 $S_b=\{\beta_1,\beta_2,\cdots,\beta_n\}$，其中集合元素均为复数。

输出：Receiver 和 Sender 都知道 $|S_a\cap S_b|$。

1) Receiver 构造矩阵 A 如下：

$$A=\begin{bmatrix}1 & \alpha_1 & \alpha_1^2 & \alpha_1^3 & \cdots & \alpha_1^n\\ 1 & \alpha_2 & \alpha_2^2 & \alpha_2^3 & \cdots & \alpha_2^n\\ \vdots & \vdots & \vdots & \vdots & & \vdots\\ 1 & \alpha_m & \alpha_m^2 & \alpha_m^3 & \cdots & \alpha_m^n\end{bmatrix}$$

并构造随机非奇异矩阵 R，计算出 RA。

2) Sender 通过 2.2 节的牛顿公式构造多项式 $g(x)+\lambda$ 如下：

$$\begin{aligned}g(x)+\lambda&=(x-\beta_1)(x-\beta_2)\cdots(x-\beta_n)+\lambda\\&=b_0x^n+b_1x_{n-1}+\cdots+b_n+\lambda\end{aligned}$$

根据 $g(x)+\lambda$ 的系数 $b_i\ (i=1,\cdots,n)$ 构造向量 B 如下：

$$B=\begin{bmatrix}b_n+\lambda\\ b_{n-1}\\ b_{n-2}\\ \vdots\\ b_0\end{bmatrix}$$

3) 使用 2.3 节安全矩阵乘积协议计算出 RAB 的结果。

Receiver 得到 RAB 后左乘 R^{-1}，即可得到 AB。再将 AB 结果中的每项相乘，查看关于 λ 的最低次数 k 的值。

4) Receiver 将结果告诉 Sender。

分析：在协议 2 中，Receiver 根据 S_a 构造出矩阵 A，Sender 根据 S_b 构造多项式 $g(x)$，根据 $g(x)$ 的系数进而构造向量 B。为了保证矩阵 A 与 B 的私有信息不被泄露，构造出随机非奇异矩阵 R 左乘矩阵 A，并使用安全矩阵乘积协议来进一步提高其安全性。另外，在协议开始前，Receiver 需要获取 Sender 集合 S_b 的势，即 n 的值，但泄露该信息并不能够影响集合交集势计算的安全性，因为 Receiver 无法通过该信息计算出 S_b 中的任何元素。这一点，Du 在[28]中进行过论证：当需要设计安全协议时，如果在降低完美安全性的前提下，泄露的信息并不影响协议的有效性，则认为是可接受安全。

4. 安全性分析

定理 4.1：协议 2 保密地计算了集合交集势。

证明：通过构造满足定义 1 中的模拟器 S_1 与 S_2 证明本定理。S_1 的工作过程如下：

第一步：给定输入 $(S_a,|S_a\cap S_b|)$，S_1 随机选取一个具有 n 个元素的集合 $S_b'=\{\beta_1',\beta_2',\cdots,\beta_n'\}$，使得 $f_2(S_a,S_b)=f_2(S_a,S_b')$，用 S_a、S_b' 来模拟。按照协议，先根据 S_a 构造出矩阵 A；

第二步：根据 S_b' 构造向量 B'。

第三步：使用安全矩阵乘积协议，根据 RA 构造出 RA_1、RA_2，根据 B' 构造出 B'_1、B'_2，使用该基础协议得到的结果为 RAB'。

在本协议中：

$$view_1(S_a, S_b) = \{S_a, RA, RA_1, RA_2, B_2, RAB\}$$

$$S_1(S_a, f_1(S_a, S_b)) = \{S_a, RA, RA_1, RA_2, B'_2, RAB'\}$$

由于 S'_b 是具有 n 个随机元素的集合，故 B_2、RAB 与 B'_2、RAB' 在计算上是不可区分的，所以有

$$view_1(S_a, S_b) \overset{c}{=} S_1(S_a, f_1(S_a, S_b))$$

又因为第一步中

$$f_2(S_a, S_b) = |S_a \cap S_b|$$

$$f_2(S_a, S'_b) = |S_a \cap S'_b|$$

$$f_2(S_a, S_b) = f_2(S_a, S'_b)$$

故

$$|S_a \cap S_b| = |S_a \cap S'_b|$$

又因为

$$output_2(S_a, S_b) = f_2(S_a, S_b) = |S_a \cap S_b|$$

所以：

$$\{S_1(S_a, f_1(S_a, S_b)), f_2(S_a, S_b)\} \overset{c}{=} \{view_1(S_a, S_b), output_2(S_a, S_b)\}$$

同理，可以构造出 S_2，满足：

$$f_1(S_a, S_b), S_2(S_a, f_2(S_a, S_b)) \overset{c}{=} output_1(S_a, S_b), view_2(S_a, S_b)$$ □

5. 实验与分析

5.1. 效率分析

本节将给出本文协议 2 与文献[14] [21] [29]中集合交集势协议的分析对比。

计算复杂度：为了便于比较，统一各个协议中两方集合的势分别为 m 与 n，并设模幂运算的开销为 M_p，幂运算开销为 M_e，点积协议运算开销为 M_d，安全矩阵乘积协议运算开销为 M_m。

在上述条件下，文献[14]基于可交换加密 Pohlig-Hellman 算法进行了 $2m+2n$ 次模幂运算；文献[29]没有进行公钥加密，只进行了 m 次幂运算，但使用了 n 次点积协议[30]；文献[21]基于 Paillier 加密方案，共需要 $3L+2n+2$ 次模幂运算(L 为大于 m 的数)；本文协议 2 同样没有进行公钥加密，也进行了 m 次幂运算，但是只使用了一次基础协议，即 2.3 节的安全矩阵乘积协议。

通信复杂度：在安全多方计算问题研究中，常用参与者在协议运行中交换信息的次数来代表通信复杂度。

故通信开销上，文献[14]进行了 4 次通信，文献[29]进行了 $5n$ 次通信，文献[21]进行了 3 次通信，本文协议 2 也进行了 3 次通信。通信开销上，本文协议具有一定优势。

这样得到各个协议的具体比较如表 1。

在表 1 中，每个协议的计算开销是模幂、幂运算与基础协议开销的和。通过对比可以看出，文献[14] [21]的模幂开销较大，分别是$(2m+2n)M_p$和$(3L+2n+2)M_p$，而文献[29]与本文协议 2 的幂开销均只有mM_e。同时，文献[14] [21]并没有使用额外的基础协议，文献[29]使用 n 次点积协议，协议 2 仅使用了一次安全矩阵乘积协议。与文献[29]相比，由于安全矩阵乘积协议的计算中仅需计算四个矩阵相乘，时间消耗并不高(这一点将通过 5.2 节实验部分展示)，故协议 2 计算量上占优；与文献[14] [21]相比，尽管他们没有使用基础协议，但是使用了大量模幂运算，导致计算量较大。

Table 1. Comparison
表 1. 比较

协议	计算开销	通信轮数	是否支持二维数据
[14]	$(2m+2n)M_p$	4	否
[29]	mM_e+nM_d	$5n$	否
[21]	$(3L+2n+2)M_p$	3	是
协议 2	mM_e+M_m	3	是

安全性上，文献[14]使用的是 Pohlig-Hellman 加密方案[15]，该方案基于离散对数困难问题；文献[29]使用的是点积协议，该协议基于茫然传输；文献[21]使用的是 Paillier 同态加密方案，该方案基于大整数因数分解困难问题。协议 2 使用的是安全矩阵乘积协议，双方在该协议的过程中，将得到的中间量构造方程组并尝试解出对方的私有信息时会发现该方程组的解有无穷个，该协议以此来保证双方私有信息的安全性。

应用范围上，文献[14] [29]中的集合元素均是整数，并不支持二维数据。尽管文献[21]提出了一种将二维数据通过哥德尔编码变为一维数据再进行计算，但是由于存在编码解码过程，该环节也会造成额外计算开销。

另外，本文协议与基于 Bloom 过滤器的方案[25]不同，本文协议的计算是准确的，而基于 Bloom 过滤器的集合交集计算是有小概率出错的。最后，本文协议不要求集合元素取自某个全集，而文献[29]要求集合取自一个约定的全集，因此本文协议具有更为广泛的应用范围。

5.2. 实验与优化

5.2.1. 实验

本节将对协议 2 进行实验测试。

实验环境如下：操作系统为 Windows10 专业版，处理器为 Intel(R) Core(TM) i7-9700 CPU @ 3.00 GHz，RAM 为 40.0 GB，64 位操作系统。使用 Python 对协议 2 进行编程实现。

由于涉及到矩阵计算，故代码使用 numpy 库中相关函数来对矩阵进行处理。但由于 numpy 中能够处理的数据大小有限，为了防止计算溢出，需要限制两方集合势的大小，所以实验将发送方一方的集合势固定为 8，在此基础上，改变 Receiver 一方集合势的大小。以下实验数据均为重复 10 次实验求其平均值得到的数据。其结果如图 1。

通过图 1 的实验数据可以发现，各方的计算时间随着 Receiver 集合势的增长而增长，该增长趋近于线性增长。

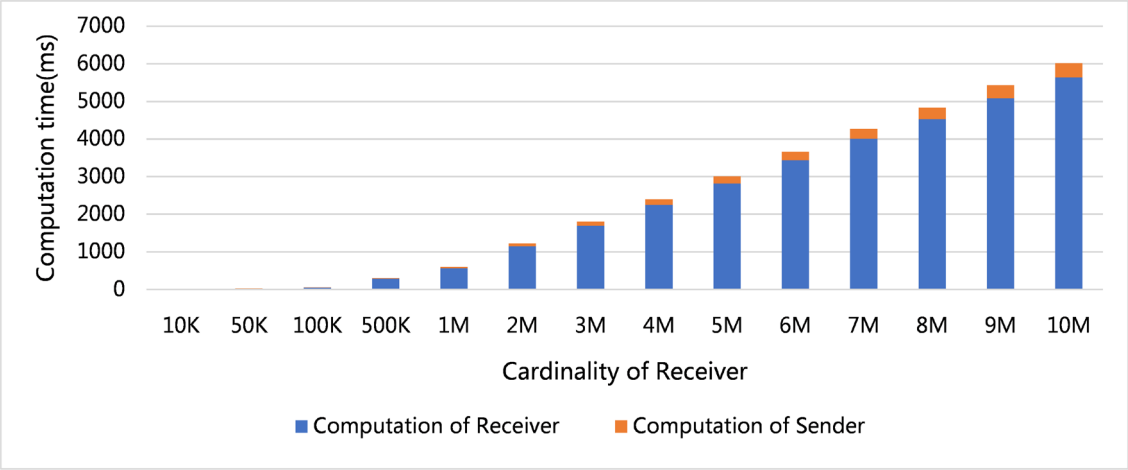

Figure 1. The computation time of protocol 2
图 1. 协议 2 的计算时间

5.2.2. 优化

本节将采用随机哈希桶的方式对协议 2 实验进行优化，降低其计算开销。具体方法是：进行实验前，Receiver、Sender 双方使用哈希函数 h 来对各自的集合元素进行哈希，将每个元素放入对应的哈希桶中，设哈希桶数量为 B。等到分配完所有的元素后，双方再将对应哈希桶中的元素执行协议 2 进行交集计算，最后将所有哈希桶中交集势求和，即可得到正确的结果。

由于哈希函数的不确定性，每个哈希桶中存放的元素个数是不一样的，此时为了安全考虑，需要往哈希桶中添加额外元素 0，使得每个哈希桶中元素个数一样。

在对协议 2 进行优化后，依然将 Sender 一方的集合势固定为 8，并改变哈希桶的个数进行实验。结果如图 2。

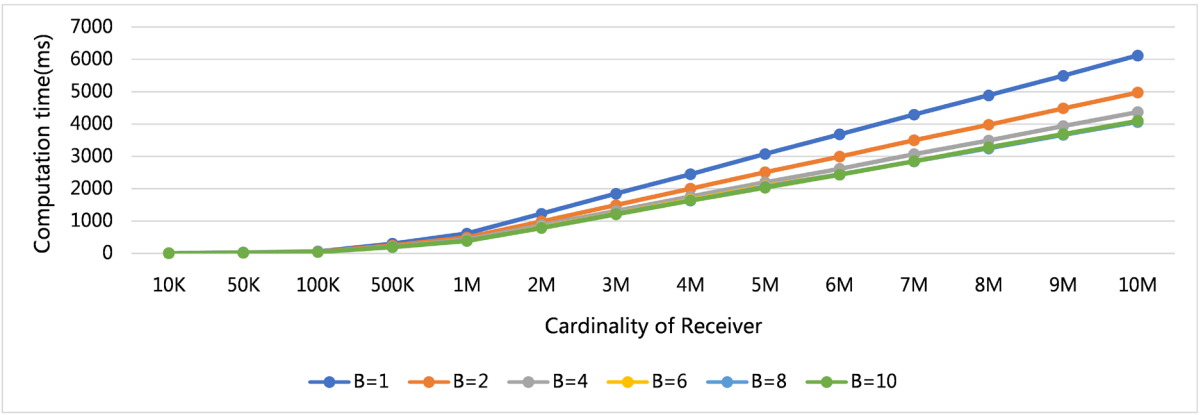

Figure 2. The comparison of different B
图 2. 不同哈希桶数量下的比较

$B=1$的折线与其他折线相比可以看出，添加哈希桶确实能够有效降低计算上的时间开销，最多能够降低大约 30%左右的时间。

最后，在设定哈希桶 $B=4$、Alice 集合元素的势为 100 万的条件下，改变 Sender 集合元素势的大小，其结果如图 3。

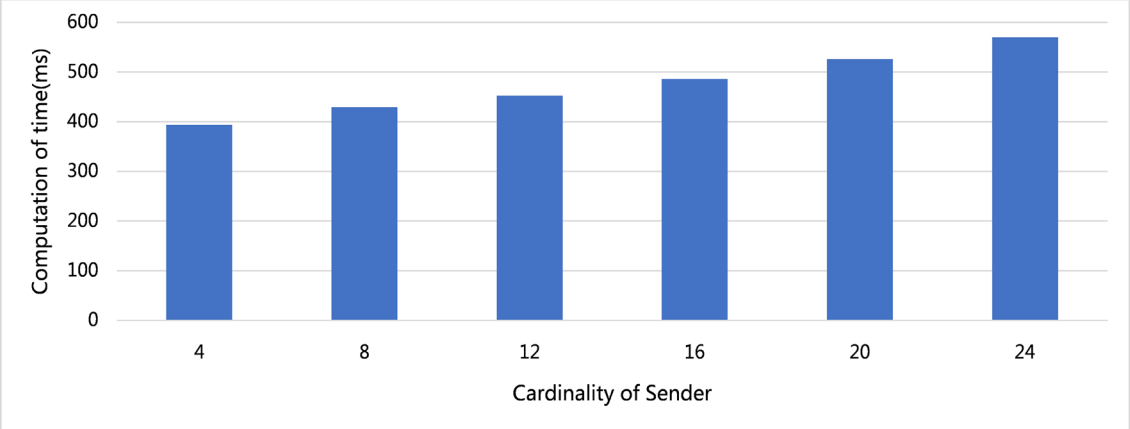

Figure 3. The comparison of different cardinality of Sender
图 3. 不同 Sender 势的比较

从图 3 中可以看出，Sender 集合元素势的大小成倍增加时，所需的计算时间并没有成倍增加，这表明 Sender 集合势的改变对于协议计算时间的影响是比较小的。

从以上实验可以看出，本文协议的计算时间随着双方势的大小增加而增加，趋于线性增长，增长幅度较慢。另外，使用随机哈希可以有效降低计算开销，在 Sender 集合势为 8、Receiver 集合势为一千万、哈希桶个数为 8 时，双方总计算时间为 4 秒左右。最后，Sender 端的计算时间远远小于 Receiver 端的计算时间，而 Sender 代表着拥有数据数量较少的用户，Receiver 代表着拥有庞大数据量的服务器，这意味着大部分的计算开销集中于服务器端，用户本身承担的计算开销较少。

6. 结束语

本文设计了一种二维数据上的求解两方集合交集势的安全协议。该协议将二维数据视为复数，再利用 2.2 节的牛顿公式构造多项式，将集合相交问题转化为两方多项式系数构造出的结式求解问题，再利用矩阵乘积协议来求解该结式。

通过实验结果可以看出，本文协议当一方集合势大小为 100 万、另一方为 20，哈希桶大小为 4 时，其计算消耗的时间为 500 ms 左右。而与同样可以应用在二维数据上的文献[21]中的方案相比，后者在计算双方集合的势均为 30 个时，所消耗的时间为 500 ms 左右。比较后可以发现，本协议在优化之后的计算时间是比较少的。本协议适用于一方集合较大、另一方集合较小的多维数据集合求交的应用场景，如引言中描述的保险公司想查询几位新客户是否存在病史问题、手机 APP 想要知道用户手机通讯录中有多少人曾经下载使用过本 APP 的问题；也由于任意一个有理数都可以写成两个整数相除的最简形式，将这两个整数视为二维数据的话，本协议也可以解决一些场景下的有理数集合求交问题。另外，本协议的计算是准确的，而非有小概率出错的，并且本协议不要求集合取自一个约定的全集，这使本协议具有更广泛的应用范围。但是本文协议也存在不足。本文协议只适用于两方集合的场景，并不容易推广到多方。如果想要解决该问题，还需要进一步研究。

基金项目

国家重点研发计划“变革性技术关键科学问题”(2020YFA0712300)；重庆市自然科学基金(cstc2019jcyj-msxmX0638)；国家自然科学基金(11771421)；中国科学院“西部之光”；重庆市科技创新引导专项(cstc2018jcyj-yszxX0002，cstc2019yszx-jcyjX0003)。

参考文献

[1] Yao, A.C. (1982) Protocols for Secure Computation. 23*rd Annual Symposium on Foundations of Computer Science*, Chicago, 3-5 November 1982, 160-164. https://doi.org/10.1109/SFCS.1982.38

[2] Goldreich, O., Micali, S. and Wigderson, A. (1987) How to Play ANY Mental Game. *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, New York, January 1987, 218-229. https://doi.org/10.1145/28395.28420

[3] Goldreich, O. (2004) Foundations of Cryptography II: Basic Applications. *Journal of the ACM*, **10**, 359-364. https://doi.org/10.1017/CBO9780511721656

[4] Cristofaro, E.D., Kim, J. and Tsudik, G. (2010) Linear-Complexity Private Set Intersection Protocols Secure in Malicious Model. 16*th International Conference on the Theory and Application of Cryptology and Information Security*, Singapore, 5-9 December 2010, 213-231. https://doi.org/10.1007/978-3-642-17373-8_13

[5] Jarecki, S. and Liu, X. (2010) Fast Secure Computation of Set Intersection. *International Conference on Security & Cryptography for Networks*, Amalfi, 13-15 September 2010, 418-435. https://doi.org/10.1007/978-3-642-15317-4_26

[6] Freedman, M.J., *et al*. (2016) Efficient Set Intersection with Simulation-Based Security. *Journal of Cryptology*, **29**, 115-155. https://doi.org/10.1007/s00145-014-9190-0

[7] Cristofaro, E.D., Gasti, P. and Tsudik, G. (2012) Fast and Private Computation of Cardinality of Set Intersection and Union. *International Conference on Cryptology and Network Security*, Darmstadt, 12-14 December 2012, 218-231. https://doi.org/10.1007/978-3-642-35404-5_17

[8] Pinkas, B., Schneider, T. and Zohner, M. (2018) Scalable Private Set Intersection Based on OT Extension. *ACM Transactions on Privacy and Security*, **21**, 7:1-7:35. https://doi.org/10.1145/3154794

[9] Dong, C.Y., Chen, L.Q. and Wen, Z.K. (2013) When Private Set Intersection Meets Big Data: An Efficient and Scalable Protocol. *Proceedings of the* 2013 *ACM SIGSAC Conference on Computer & Communications Security*, Berlin, 4-8 November 2013, 128-144.

[10] Kolesnikov, V., *et al*. (2016) Efficient Batched Oblivious PRF with Applications to Private Set Intersection. In: *Proceedings of the* 2016 *ACM SIGSAC Conference on Computer and Communications Security*, Association for Computing Machinery, New York, 818-829. https://doi.org/10.1145/2976749.2978381

[11] Seo, J.H., Cheon, J.H. and Katz, J. (2012) Constant-Round Multi-Party Private Set Union Using Reversed Laurent Series. *International Conference on Practice & Theory in Public Key Cryptography*, Darmstadt, 21-23 May 2012, 398-412. https://doi.org/10.1007/978-3-642-30057-8_24

[12] Blanton, M. and Aguiar, E. (2016) Private and Oblivious Set and Multiset Operations. *International Journal of Information Security*, **15**, 493-518. https://doi.org/10.1007/s10207-015-0301-1

[13] Chun, J.Y., *et al*. (2013) Privacy-Preserving Disjunctive Normal Form Operations on Distributed Sets. *Information Sciences*, **231**, 113-122. https://doi.org/10.1016/j.ins.2011.07.003

[14] Clifton, C., *et al*. (2002) Tools for Privacy Preserving Distributed Data Mining. *ACM SIGKDD Explorations Newsletter*, **4**, 28-34. https://doi.org/10.1145/772862.772867

[15] Pohlig, S. and Hellman, M. (1978) An Improved Algorithm for Computing Logarithms over GF(p) and Its Cryptographic Significance. *IEEE Transactions on Information Theory*, **24**, 106-110. https://doi.org/10.1109/TIT.1978.1055817

[16] Freedman, M.J., Nissim, K. and Pinkas, B. (2004) Efficient Private Matching and Set Intersection. 23*rd Annual Eurocrypt Conference*, Interlaken, 2-6 May 2004, 1-19. https://doi.org/10.1007/978-3-540-24676-3_1

[17] Paillier, P. (1999) Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. *International Conference on the Theory and Application of Cryptographic Techniques*, Prague, 2-6 May 1999, 223-238.

[18] Elgamal, T. (1985) A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Trans on Information Theory*, **31**, 469-472. https://doi.org/10.1109/TIT.1985.1057074

[19] Kissner, L. and Song, D. (2005) Privacy-Preserving Set Operations. In: Shoup, V., Ed., *Advances in Cryptology-Crypto'*05, Vol. 3621 of LNCS, Springer-Verlag, Berlin, 241-257. https://doi.org/10.1007/11535218_15

[20] Xia, F. (2012) Secure Two-Party Computation for Set Intersection and Set Equality Problems Based on LWE. *Journal of Electronics & Information Technology*, **34**, 462-467.

[21] 窦家维, 刘旭红, 王文丽. 有理数域上两方集合的高效保密计算[J]. 计算机学报, 2020, 43(8): 1397-1413.

[22] 巩林明, 等. 基于无匹配差错的 PSI 计算[J]. 计算机学报, 2020, 43(9): 1769-1790.

[23] Huang, Y., Evans, D. and Katz, J. (2012) Private Set Intersection: Are Garbled Circuits Better than Custom Protocols? *NDSS* 2012, San Diego, 5-8 February 2012.

[24] Pinkas, B., Schneider, T. and Zohner, M. (2014) Faster Private Set Intersection Based on OT Extension. *Proceedings of the* 23*rd USENIX Security*, San Diego, 20-22 August 2014, 797-812.

[25] Lv, S., *et al*. (2020) Unbalanced Private Set Intersection Cardinality Protocol with Low Communication Cost. *Future Generation Computer Systems*, **102**, 1054-1061. https://doi.org/10.1016/j.future.2019.09.022

[26] 侯晓荣, 杨张. 非线性代数方程组与定理机器证明[M]. 上海: 上海科技教育出版社, 2001.

[27] Du, W., Han, Y. and Chen, S. (2004) Privacy-Preserving Multivariate Statistical Analysis: Linear Regression and Classification. https://doi.org/10.1137/1.9781611972740.21

[28] Du, W. and Zhan, Z. (2003) A Practical Approach to Solve Secure Multi-Party Computation Problems. *Proceedings of the* 2002 *Workshop on New Security Paradigms*, Virginia Beach, 23-26 September 2002, 127-135. https://doi.org/10.1145/844102.844125

[29] 孙彦飞, 等. 保护私有信息的集合交集协议[J]. 计算机应用, 2010, 30(2): 506-509.

[30] Tzeng, W. (2001) Efficient Oblivious Transfer Schemes. IACR Cryptology ePrint Archive, 73.