云外包计算中空间位置关系的保密判定

陈振华[1),2)] 李顺东[3)] 黄 琼[4)] 丁 勇[5)] 孙 嫚[1)]

[1)](西安科技大学计算机科学与技术学院 西安 710054)

[2)](中国科学院信息工程研究所信息安全国家重点实验室 北京 100093)

[3)](陕西师范大学计算机科学学院 西安 710062)

[4)](华南农业大学数学与信息学院 广州 510642)

[5)](桂林电子科技大学广西密码学与信息安全重点实验室 广西 桂林 541004)

摘 要 目前安全多方几何计算问题都是在传统模式下依靠参与方交互完成,文中首次将安全多方几何计算问题转移到云计算平台下借助不可信第三方云服务器参与完成,这为安全多方计算提供了一个新的研究方向.传统模式下空间位置关系的保密判定已存方案,大多是把原问题转化为距离问题或数据对应成比例问题,造成了用户计算成本过大,或能判断的位置关系有限,而且这些方法只能由参与方相互交互完成,因此在云计算平台下并不适用.针对这些问题,文中首先将原问题转化为夹角问题,接着设计了适用于云外包计算的内积协议,然后基于此内积协议在云平台下解决了点线、线线、点面、线面、面面五种空间位置关系的保密判定,并用模拟范例证明了协议的安全性.最后的分析和比较显示,文中不但首次设计了云计算平台下空间位置关系的保密判断协议,并且设计的方案能判断的位置关系更加广泛,也为用户节省了更多的计算成本.此外,我们设计的内积协议可以作为一种新的云计算技术的基础协议,可以被其他协议调用.

关键词 安全多方计算;位置关系;空间几何;内积协议;云计算

中图法分类号 TP309 **DOI 号** 10.11897/SP.J.1016.2017.00351

Privacy-Preserving Determination of Spatial Location-Relation in Cloud Computing

CHEN Zhen-Hua[1),2)] LI Shun-Dong[3)] HUANG Qiong[4)] DING Yong[5)] SUN Man[1)]

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

[2)](*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093)

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

[4)](*College of Mathematics and Informatics, South China Agricultural University, Guangzhou* 510642)

[5)](*Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi* 541004)

Abstract Up to now, all secure multi-party geometric computation problems are conducted under the conditional pattern, which is handled only by the interaction of participants. In this paper, we first transform this traditional pattern into the cloud computing pattern, which allows the untrusted third party (i.e., the cloud server) involved in the procedure of calculation. This manner provides a new research direction for secure multi-party computation. The most existing schemes about privacy-preserving determination of spatial location-relation transform the original problem into either the distance problem or the correspondingly proportional data problem and

收稿日期:2016-05-29;在线出版日期:2016-09-11.本课题得到国家自然科学基金(61272435,61472146)、西安科技大学博士启动基金(2015QDJ008)、信息安全国家重点实验室开放课题基金(2016-MS-19)、广东省自然科学基金杰出青年基金(2014A030306021)、广东特支计划科技青年拔尖人才(2015TQ01X796)、广州市珠江科技新星专项(201610010037)、江苏省优势学科 PAPD 和江苏省大气环境与装备技术协同创新中心 CICAEET(KJR1615)资助.**陈振华**,女,1976 年生,博士,副教授,研究方向为秘密共享和安全多方计算.E-mail:chenzhenhua@snnu.edu.cn.**李顺东**,男,1963 年生,教授,博士生导师,主要研究方向为信息隐藏和安全多方计算.**黄 琼**,男,1981 年生,教授,硕士生导师,主要研究领域为信息隐藏和密文上的搜索.**丁 勇**,男,1976 年生,教授,硕士生导师,主要研究领域为信息隐藏和密文上的搜索.**孙 嫚**,女,1990 年生,硕士研究生,主要研究方向为信息安全和安全多方计算.

solve it with traditional pattern. These approaches burden the user's computation overhead or limit the range of determining location-relation, moreover, with which the participants accomplish the computation task only by their interactive with each other. It is not suitable for secure multi-party computation in cloud computing. Aiming at these problems, we first transform the original problem into the included angle problem, then design a novel protocol of scalar product available for cloud computing, and further determine the five spatial location-relation: point and line, point and plane, line and line, line and plane, and plane and plane in cloud computing. Lastly, we prove the security of our protocols with simulation paradigm. The analysis indicates it is the first that we design the protocol for preserving-privacy determination of spatial location-relation in cloud computing, which can not only save more computation cost for users but also determine more location-relation than the known schemes. In addition, the proposed scalar protocol in this work as the new technique of cloud computing can be used a building-block, which can be invoked by other schemes.

Keywords secure multi-party computation; location-relation; spatial geometry; scalar product protocol; cloud computing

1 引 言

安全多方计算最早由 Yao[1] 提出,是指在不泄漏各方的输入数据(隐私性)的条件下,能正确合作完成输入数据的函数计算(正确性).现实问题中涉及到保护隐私的合作计算都可以归结到安全多方计算的研究范围.因此它在保护隐私的质量评估[2]、定位查找[3-4]、数据挖掘[5]、数据查询[6-7]、外包计算[8]等方面有着广泛的应用.

安全计算几何问题属于安全多方计算中的一个分支,是指各参与者都处在几何图形中,各自拥有几何图形中的一些参量,在不泄露各参与方数据隐私的条件下,合作计算一些几何问题.无论是在理论还是实际中,安全计算几何问题是经常会碰到的问题.针对于此,Yang 等人[9]研究了平面上线和圆、圆和圆的位置关系.Liu 等人[10]研究了更普遍意义上的点和一般曲线的位置关系.Qin 等人[11]研究了平面曲线相交问题.Liu 等人[12]研究了平面上三角形面积的计算问题.He 等人[13]研究了平面上线和圆、线和双曲线的位置关系.但是这些问题都集中在平面几何上,而空间几何问题很少有人涉及.由于空间几何比平面几何更能形象的刻画现实世界,因此研究立体几何,意义更加重大.例如以下的场景:

A 国准备向 C 国发射航空导弹,但 C 国的领空上有 A 国的盟国 B 国的一部分秘密基地(如太空站),因此 A 国不能伤害自己的盟国,即 A 国的航空导弹不能穿过 B 国的秘密基地.但是为了军事机密,A 国不能向 B 国透露自己的航空导弹轨道,而 B 国也不能向 A 国透露自己太空站的具体位置.那么双方如何在不暴露自己隐私的情况下,解决这个问题呢?

以上场景属于保密军事问题,转化成数学模型就是保密判定空间中一个点是否在空间一条直线上,属于立体几何中空间位置关系的安全多方计算问题.但目前关于这方面的研究文献并不多.由于现实中很多问题都可以归结为此类问题,因此研究其理论意义对现实问题有着重要的应用价值.

1.1 相关工作

针对立体几何中点、线、面等空间位置关系的安全多方计算问题,以往的学者们提出了一些解决方案.2006 年,罗永龙等人[14]利用法向量与方向向量,将原问题转化成数据对应成比例的问题,从而解决了线线、线面、面面位置关系的判定.由于该方案调用了多种基本协议:百万富翁协议、点积协议,数据对应成比例协议.其中的数据对应成比例协议的复杂性,使得方案的计算成本和通信成本极高,成为了制约效率的瓶颈(后文表 4 有分析).2014 年,Li 等人[15]利用四面体的体积,将原问题转化为距离来判定点面、线面、面面的位置关系.此方法非常简单巧妙而且效率较高,但是由于该方法要利用四面体的体积,再通过距离解决空间位置关系.而点和直线、直线和直线,无法组成四面体,因此该方法要判断点与线、线与线的位置,并不适用.即,该方法能解决的

问题比较局限，能判断出的位置关系也比较单一(后文表 5 有分析).

由于以上方案都是在传统模式下解决，并且由于使用的转化方法，使得用户的计算开销过大(罗永龙等人[14]的方法)，或者解决的问题比较局限、能判断出的位置关系也比较单一(Li 等人[15]的方法).

传统的安全多方计算中不存在可信的第三方，都是由参与方相互交互合作完成计算任务. 因此计算能力仅仅依靠用户本身，这在解决复杂问题时往往效率较低. 云计算是一种新的服务交付和计算模式，具有强大的计算能力，因此人们想到可以将安全多方计算中的复杂计算任务以按需付费的形式外包给云，从而为用户节省大量的计算成本. 但由于云计算服务器(第三方)是不可信的，因此将传统的安全多方计算模式转移到云计算平台下时，就有较大的区别，所用到的技术手段也有不同. 若把本文空间位置关系的保密判定问题转移到云计算平台下进行研究，以往的协议[14-15]由于都是依靠参与方本身交互完成，并不适用于云计算平台. 那么如何在不可信第三方存在的云平台下解决空间位置关系的保密判定正是本文要解决的问题.

1.2　本文贡献

本文首次将传统的安全计算几何问题转移到云计算平台下，并利用内积解决了空间位置关系的保密判定. 具体贡献如下：

(1) 提供了一种新的转化方法.

本文将空间位置关系的判定问题转化为夹角问题. 避免了前人多次求距离或数据对应成比例的方法. 使得能判断的位置关系多样化、解决的问题更加广泛.

(2) 提供了新技术.

本文设计的适用于云计算外包的内积协议，可单独作为一种新的云计算技术.

(3) 提供了新的研究方向.

① 将传统的无可信第三方的安全多方计算转移到有不可信第三方存在的云计算平台下进行研究.

② 借助云计算的能力为安全多方计算中的用户节省了计算成本.

2　预备知识

2.1　安全多方计算的安全性定义

由于本文所涉及的参与方和云服务器都是半诚实模型，恶意模型下的协议可以根据 Goldreich 等人[16-17]的通用转化方法直接由半诚实模型下的协议转化得到. 因此这里只给出半诚实模型下的协议和相应的安全性模拟范例.

(1) 半诚实参与者

安全多方计算的协议运行环境分为半诚实参与者模型和恶意参与者模型[16-17]，半诚实参与者指协议方将诚实地执行协议，不会篡改输入和输出信息，但可能会保留计算的中间结果，试图推导出协议之外的信息或者他人的信息.

(2) 半诚实模型下的安全性定义

云计算平台下的安全两方计算，由于不可信第三方(云服务器)参与了计算，因此和传统模式下安全两方计算的安全性模拟范例略有不同.

设 Alice 拥有 x，Bob 拥有 y，Server 是云服务器，$f(x,y)$为概率多项式函数，π 是计算 f 的协议. 三方要在不暴露 x,y 的前提下，合作计算函数 $f(x,y)=(f_1(x,y),f_2(x,y))$. 计算的目的是为了让 Alice 和 Bob 分别得到函数 f 的两个分量 $f_1(x,y)$，$f_2(x,y)$. 在整个计算过程中，要保证 Alice(Bob)没有得到对方的隐私输入；Server 也没有得到 Alice 和 Bob 的隐私输入；并且即使 Server 和其中一方合谋，也不能得到另一方的隐私.

将 Alice 在执行协议 π 的过程中得到的视图记为 $view_1(x,y)$；Bob 的视图记为 $view_2(x,y)$；Server 的视图记为 $view_3(x,y)$；Sever 和 Alice 合谋的视图记为 $view_4(x,y)=(view_1(x,y),view_3(x,y))$；Sever 和 Bob 合谋的视图记为 $view_5(x,y)=(view_2(x,y),view_3(x,y))$. $E(x)$，$E(y)$分别表示 Alice 和 Bob 发送给 Server 的数据(即 Server 获得的输入)，$E(x,y)$表示 Server 的最后计算结果(即 Server 获得的输出)，$\subseteq_c$ 表示计算上可区分.

定义 1.　在云计算环境下，我们说协议 π 保密地计算了 $f(x,y)$，如果存在概率多项式时间的模拟器 S_1、S_2、S_3、S_4、S_5 使得表 1 中的 5 个式子同时成立.

表 1 中的 5 个式子同时成立，说明了参与者视图中的信息视图中不包含额外的信息，这样就保证了在协议执行过程中，无论合谋与否，都得不到其他方的私有信息. 因此，要证明一个云计算环境下两方计算协议是保密的，就必须构造使得表 1 中的(1)(2)(3)(4)(5)式同时成立的模拟器 S_1、S_2、S_3、S_4、S_5.

表 1 云计算平台下两方计算的安全性

类型	安全性	模拟器	
不合谋	Bob 的隐私性(相对于 Alice)	$S_1(x,f_1(x,y)) \subseteq view_1(x,y)$	(1)
	Alice 的隐私性(相对于 Bob)	$S_2(y,f_2(x,y)) \subseteq view_2(x,y)$	(2)
	Alice 和 Bob 的隐私性(相对于 Server)	$S_3(E(x),E(y),E(x,y)) \subseteq view_3(x,y)$	(3)
合谋	Bob 的隐私性(相对于 Server 和 Alice 合谋)	$S_4(x,E(x),E(y),f_1(x,y),E(x,y)) \subseteq view_4(x,y)$	(4)
	Alice 的隐私性(相对于 Server 和 Bob 合谋)	$S_5(y,E(x),E(y),f_2(x,y),E(x,y)) \subseteq view_5(x,y)$	(5)

2.2 双线性对

G_1,G 为两个同为素数阶 q 的乘法群,e 为一个线性映射,$e:G_1\times G_1\rightarrow G$,$g$ 为 G_1 的一个生成元,若 e 满足以下性质:

(1) 双线性性. 对于任意的 $a,b\in Z$,有等式 $e(g^a,g^b)=e(g,g)^{ab}$;

(2) 非退化性. $e(g,g)\neq 1$;

(3) 可计算性. 对于任意的 $P,Q\in G_1$,$e(P,Q)$ 可有效计算. 则 e 为群 G_1,G 上的双线性对.

2.3 Boneh 的 2-DNF 同态加密算法[18]

设 E 是一个加密算法,$C_1=E(m_1,r_1)$ 是对 m_1 的加密,$C_2=E(m_2,r_2)$ 是对 m_2 的加密,其中 r_1,r_2 是两个随机数. 若对于某个随机数 r,有 $C_1C_2=E(m_1m_2,r)$,则称 E 是一个乘法同态加密算法;若有 $C_1C_2=E(m_1+m_2,r)$,则称 E 是一个加法同态加密算法.

Boneh 在文献[18]中提出了一种加密方案如下(图 1).

系统建立:解密者选取 $N=pq$,其中 p,q 为两个素数,双线性对 $e:G_1\times G_1\rightarrow G$,$G_1$ 和 G 同为阶数为 N 的群,g,u 为 G_1 的两个随机生成元,且 $h=u^p$. 公钥:(G_1,G,e,g,h,N),私钥:q.
加密过程:明文 $m<p$,任取随机数 $r\in Z_N$,密文 $c=g^mh^r$.
解密过程:密文 $c<N$,明文 $m'=c^q$,若 $m'=1$,则明文 $m=0$;若 $m'\neq 1$,则说明明文 $m=1$.

图 1 Boneh 的同态加密方案

在以下公式中,令 $e(g,g)=g_1$,$e(g,h)=h_1$,$r_1+r_2=r$,$m_2r_1+m_1r_2+\alpha pr_1r_2=r'$. 由于 $h=u^p$,因此 $h=g^{ap}$,α 为一个为未知的整数. 则有下式:

$$\begin{aligned} c_1c_2&=g^{m_1}h^{r_1}.g^{m_2}h^{r_2}\\ &=g^{m_1+m_2}h^{r_1+r_2}\\ &=g^{m_1+m_2}h^{r} \end{aligned} \tag{1}$$

$$\begin{aligned} e(c_1,c_2)&=e(g^{m_1}h^{r_1},g^{m_2}h^{r_2})\\ &=e(g,g)^{m_1m_2}e(g,h)^{m_2r_1+m_1r_2}e(h,h)^{r_1r_2}\\ &=e(g,g)^{m_1m_2}e(g,h)^{m_2r_1+m_1r_2}e(g^{ap},h)^{r_1r_2}\\ &=e(g,g)^{m_1m_2}e(g,h)^{m_2r_1+m_1r_2+\alpha pr_1r_2}\\ &=g_1^{m_1m_2}h_1^{r'} \end{aligned} \tag{2}$$

即

$$E(m_1)E(m_2)=E(m_1+m_2,r) \tag{3}$$

$$e(E(m_1),E(m_2))=E(m_1m_2,r') \tag{4}$$

因此 Boneh 加密方案具有多次加法同态和一次乘法同态性质.

备注 1. 对于式(2)中的乘法同态,由于 $g_1^{m_1m_2}h_1^{r'}\bmod N\in G$,但 $\notin G_1$,不能再进行双线性对运算,即只能进行一次乘法同态运算.

3 问题的描述和转化

3.1 本文研究的问题

问题 1. 安全计算空间点与空间直线的位置关系. Alice 有一个秘密的空间点 $p_0(x_0,y_0,z_0)$,Bob 有一条秘密的空间直线

$$L:\begin{cases}A_1x+B_1y+C_1z+D_1=0\\A_2x+B_2y+C_2z+D_2=0\end{cases}.$$

Alice 和 Bob 想在不泄露对方信息的条件下,判定点 p_0 和直线 L 的位置关系(属于或不属于).

问题 2. 安全计算空间点与空间平面的位置关系. Alice 有一个秘密的空间点 $p_0(x_0,y_0,z_0)$,Bob 有一个秘密的空间平面 $\Pi:Ax+By+Cz+D=0$. Alice 和 Bob 想在不泄露对方信息的条件下,判定点 p_0 和平面 Π 的位置关系(属于或不属于).

问题 3. 安全计算空间两直线的位置关系. Alice 有一条秘密的空间直线

$$L_1:\begin{cases}A_1x+B_1y+C_1z+D_1=0\\A_2x+B_2y+C_2z+D_2=0\end{cases},$$

Bob 有一条秘密的空间直线

$$L_2:\begin{cases}A_3x+B_3y+C_3z+D_3=0\\A_4x+B_4y+C_4z+D_4=0\end{cases}.$$

Alice 和 Bob 想在不泄露对方信息的条件下,判定直线 L_1 与直线 L_2 的位置关系(垂直、平行、重合或相交).

问题 4. 安全计算空间直线与空间平面的位置关系. Alice 有一条秘密的空间直线

$$L:\begin{cases}A_1x+B_1y+C_1z+D_1=0\\A_2x+B_2y+C_2z+D_2=0\end{cases},$$

Bob 有一个秘密的空间平面 $\Pi:A_3x+B_3y+C_3z+$

$D_3=0$. Alice 和 Bob 想在不泄露对方信息的条件下，判定直线 L 与平面 Π 的位置关系（垂直、平行、重合或相交）.

问题 5. 安全计算空间平面与空间平面的位置关系. Alice 有一个秘密的空间平面 $\Pi_1: A_1x+B_1y+C_1z+D_1=0$，Bob 有一个秘密的空间平面 $\Pi_2: A_2x+B_2y+C_2z+D_2=0$. Alice 和 Bob 想在不泄露对方信息的条件下，判定平面 Π_1 与平面 Π_2 的位置关系（垂直、平行、重合或相交）.

3.2　问题的转化

本文中点与线、点与面的位置关系可以转化为一个点是否为一个方程的解；而线与线、线与面、面与面的位置关系可以转化为方向向量和法向量的夹角，最后利用保密内积协议解决.

例如，要解决本文的问题 4：判断一条直线 L 和一个平面 Π 的位置关系（图 2）. 等价于已知直线 L 的方向向量 $\boldsymbol{e}$，平面 Π 的法向量 $\boldsymbol{d}$，那么两者之间的夹角为 $\cos\theta=\dfrac{\langle \boldsymbol{e},\boldsymbol{d}\rangle}{|\boldsymbol{e}||\boldsymbol{d}|}$. 夹角 $\cos\theta$ 的情况分为以下 3 种：

(1) 若 $\cos\theta=0$，则直线和平面平行或重合；

(2) 若 $\cos\theta=\pm1$，则直线和平面垂直；

(3) 其他则为相交.

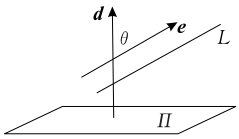

图 2　直线和平面

因此要判断直线和平面的位置关系，只需保密判断 $\cos\theta$ 是否为 0、±1 或其他即可，而要求夹角，可借助内积协议求出 $\langle \boldsymbol{e},\boldsymbol{d}\rangle$.

3.3　云计算外包下的内积协议

根据 3.2 节的转化方法，原问题转化为夹角问题后，需要借助内积协议. 但目前大多内积协议多是基于同态加密的方法[19-20]，不可避免的要计算多个模幂运算，若处理的内积问题很大的话，会给用户带来较大的计算开销. 借助云计算的能力，将内积协议中大多数的计算任务外包给云，用户只要付出少量费用，就可以让云完成用户所需的计算. 但传统模式下的内积计算并不存在第三方，都是由参与方相互交互完成. 而云计算平台下的内积计算需要不可信第三方（云服务器）参与计算过程. 这种情况下，不可信第三方（云服务器）可能会将用户的数据信息泄露. 因此用户需要在对云服务器保持输入和输出隐私的情况下完成所需要的计算任务.

综上分析，由于云计算平台下的内积计算并不同于传统模式下的内积计算，此时传统的内积协议[19-20]都已经失效. 因此基于 Boneh 的 2-DNF 同态加密算法[18]，本文重新设计了一种适用于云外包计算下保密计算内积的协议. 在本文场景中，假设所有的参与方和云服务器都是半诚实模型，网络之间传输都是公开信道.

3.3.1　具体内积协议

协议 1.　云计算外包下的内积协议.

输入：Alice 保密输入 $\boldsymbol{X}=(x_1,x_2,\cdots,x_n)$，Bob 保密输入 $\boldsymbol{Y}=(y_1,y_2,\cdots,y_n)$，且 $n\geq2$

输出：Alice 和 Bob 都知道 $\langle \boldsymbol{X},\boldsymbol{Y}\rangle$ 的值，但云服务器不知内积结果

1. Alice 选择 Boneh 加密方案的参数，公布公钥 $(\boldsymbol{e},G_1,G,n=pq,g,h)$，保密私钥 (p,q). 计算 $a_1=E(x_1),a_2=E(x_2),\cdots,a_n=E(x_n)$，发送给云服务器.

2. Bob 选择随机数 $r(r\neq0,1)$，用同样的 Boneh 加密算法计算 $b_1=E(y_1+r),b_2=E(y_2+r),\cdots,b_n=E(y_n+r)$，并发送给云服务器.

3. 云服务器收到 Alice 和 Bob 发来的密文后，利用 Boneh 加密算法的同态性计算 2-DNF 的多项式，计算以下两式：

$$\begin{aligned}c&=e(a_1,b_1)\cdots e(a_n,b_n)\\&=e(E(x_1),E(y_1+r))\cdots e(E(x_n),E(y_n+r))\\&=E(x_1y_1+\cdots+x_ny_n+rx_1+\cdots+rx_n)\\&=E(\langle \boldsymbol{X},\boldsymbol{Y}\rangle+r(x_1+\cdots+x_n)),\\c_1&=E(x_1)\cdots E(x_n)\\&=E(x_1+\cdots+x_n).\end{aligned}$$

云服务器将 c,c_1 发送给 Bob.

4. Bob 收到密文 c,c_1 后，利用 Boneh 加密算法的加法同态性计算下式：

$$\begin{aligned}c_2=c_1^r&=(E(x_1+x_2+\cdots+x_n))^r\\&=E(r(x_1+x_2+\cdots+x_n))\end{aligned}$$

并计算 $c_3=\dfrac{c}{c_2}=E(\langle \boldsymbol{X},\boldsymbol{Y}\rangle)$，将 c_3 发送给 Alice.

5. Alice 解密 c_3，得到内积 $\langle \boldsymbol{X},\boldsymbol{Y}\rangle$，并将结果告诉 Bob.

分析：在协议 1 中，从直觉上可以看出，Alice 利用加密算法保护了持有的向量 $\boldsymbol{X}$ 的隐私性. Bob 利用同样的加密算法保护了持有的向量 $\boldsymbol{Y}$ 的隐私性. 但由于云服务器可以和任何一方合谋，而 Alice 具有解密密钥，若云服务器收到 Bob 发来的密文后和 Alice 合谋解密，这样 Bob 的隐私性就得到了破坏. 为了抵抗合谋，在 Bob 的明文中加入随机数 r，使得云服务器收到的密文即使交给 Alice 解密，也有 $n+1$ 个未知数，却只有 n 个方程，仍然无法获得 Bob 的信息，因此保护了 Bob 的隐私. 下面给出模

拟范例的形式化安全性证明.

3.3.2 安全性证明

定理 1. 协议 1 保密地计算了向量 $\boldsymbol{X}$ 和向量 $\boldsymbol{Y}$ 的内积$\langle \boldsymbol{X},\boldsymbol{Y}\rangle$.

证明. 根据 2.1 节中的表 1,需要构造 5 个模拟器 S_1,S_2,S_3,S_4,S_5. 由于模拟器 S_1,S_2 的构造方法类似;S_4,S_5 的构造情况类似. 为了节省篇幅,只给出模拟器 S_1,S_3,S_4 的构造方法,其余的构造过程省略. 因此分为以下 3 种情况进行证明.

(1) 构造模拟器 S_1

在本协议中

$$f_1(\boldsymbol{X},\boldsymbol{Y})=f_2(\boldsymbol{X},\boldsymbol{Y})=\langle \boldsymbol{X},\boldsymbol{Y}\rangle$$

或者 $$f_1(\boldsymbol{X},\boldsymbol{Y})=f_2(\boldsymbol{X},\boldsymbol{Y})\neq\langle \boldsymbol{X},\boldsymbol{Y}\rangle.$$

假设 $f_1(\boldsymbol{X},\boldsymbol{Y})=f_2(\boldsymbol{X},\boldsymbol{Y})=\langle \boldsymbol{X},\boldsymbol{Y}\rangle$,构造模拟器 S_1. S_1 接受$(\boldsymbol{X},f_1(\boldsymbol{X},\boldsymbol{Y}))$作为输入,按如下方式工作:

第 1 步. S_1 接受输入$(\boldsymbol{X},f_1(\boldsymbol{X},\boldsymbol{Y}))$后,先随机选取一个向量$\boldsymbol{Y}'=(y_1',y_2',\cdots,y_m')$,使得 $f_1(\boldsymbol{X},\boldsymbol{Y})=f_1(\boldsymbol{X},\boldsymbol{Y}')$. 然后用$(\boldsymbol{X},\boldsymbol{Y}')$进行模拟. 按照协议 1,将向量 $\boldsymbol{X}=(x_1,x_2,\cdots,x_n)$加密,得到下式,记作 A.

$$\{a_1=E(x_1),a_2=E(x_2),\cdots,a_n=E(x_n)\}.$$

第 2 步. S_1 选取较大随机数 $r'(r'\neq 0,1)$,加密向量$\boldsymbol{Y}'$,得到下式,记作 B'.

$$\{b_1'=E(y_1'+r'),\cdots,b_n'=E(y_n'+r')\}.$$

第 3 步. S_1 得到 A 和 B'后,计算

$$\begin{aligned}\boldsymbol{c}'&=e(a_1,b_1')e(a_2,b_2')\cdots e(a_n,b_n')\\&=E(\langle \boldsymbol{X},\boldsymbol{Y}'\rangle+r'(x_1+x_2+\cdots+x_n)).\end{aligned}$$

第 4 步. S_1 得到$\boldsymbol{c}'$后,计算 $E(x_1+x_2+\cdots+x_n)$,并利用 Boneh 加密算法的同态性计算:

$$\begin{aligned}c_1'&=(E(x_1+x_2+\cdots+x_n))^{r'}\\&=E(r'(x_1+x_2+\cdots+x_n)),\end{aligned}$$

然后计算$\dfrac{\boldsymbol{c}'}{c_1'}=E(\langle \boldsymbol{X},\boldsymbol{Y}'\rangle)$,记作 c_2'.

第 5 步. S_1 解密 c_2'得到内积 C'.

于是得到

$$view_1(\boldsymbol{X},\boldsymbol{Y})=\{\boldsymbol{X},A,E(x_1+x_2+\cdots+x_n),c_2,C\},$$

$$S_1(\boldsymbol{X},f_1(\boldsymbol{X},\boldsymbol{Y}))=\{\boldsymbol{X},A,E(x_1+x_2+\cdots+x_n),c_2',C'\}.$$

由于 $C=f_1(\boldsymbol{X},\boldsymbol{Y})=\langle \boldsymbol{X},\boldsymbol{Y}\rangle$, $f_1(\boldsymbol{X},\boldsymbol{Y})=f_1(\boldsymbol{X},\boldsymbol{Y}')$, 而 $C'=f_1(\boldsymbol{X},\boldsymbol{Y}')=\langle \boldsymbol{X},\boldsymbol{Y}'\rangle$,因此,$C=C'$.

$$c_2'=\frac{\boldsymbol{c}'}{c_1'}=\frac{E(\langle \boldsymbol{X},\boldsymbol{Y}'\rangle+r'(x_1+x_2+\cdots+x_n))}{E(r'(x_1+x_2+\cdots+x_n))},$$

而 $$c_2=\frac{c}{c_1}=\frac{E(\langle \boldsymbol{X},\boldsymbol{Y}\rangle+r(x_1+x_2+\cdots+x_n))}{E(r(x_1+x_2+\cdots+x_n))}.$$

由于$\langle \boldsymbol{X},\boldsymbol{Y}\rangle=\langle \boldsymbol{X},\boldsymbol{Y}'\rangle$,$r\subseteq r'$, 因此,$c_2'\subseteq c_2$. 所以:

$$S_1(x,f_1(x,y))\subseteq view_1(x,y).$$

同理,用类似的方法可构造模拟器 S_2 使得

$$S_2(y,f_2(x,y))\subseteq view_2(x,y).$$

(2) 构造模拟器 S_3

在本协议中 Server 的输入为 Alice 和 Bob 发送的数据:$E(\boldsymbol{X}),E(\boldsymbol{Y})$;Server 的输出为 $E(\boldsymbol{X}),E(\boldsymbol{Y})$的计算结果:$E(\boldsymbol{X},\boldsymbol{Y})$. S_3 接受$(E(\boldsymbol{X}),E(\boldsymbol{Y}),E(\boldsymbol{X},\boldsymbol{Y}))$作为输入,按如下方式工作:

第 1 步. S_3 接受输入$(E(\boldsymbol{X}),E(\boldsymbol{Y}),E(\boldsymbol{X},\boldsymbol{Y}))$后,首先随机选取两个向量 $\boldsymbol{X}'=(x_1',x_2',\cdots,x_m')$, $\boldsymbol{Y}'=(y_1',y_2',\cdots,y_m')$,使得 $E(\boldsymbol{X},\boldsymbol{Y})=E(\boldsymbol{X}',\boldsymbol{Y}')$. 然后用$(\boldsymbol{X}',\boldsymbol{Y}')$进行模拟. 按照协议 1,将向量 $\boldsymbol{X}'=(x_1',x_2',\cdots,x_m')$加密,得到下式,记作 A'.

$$\{a_1'=E(x_1'),a_2'=E(x_2'),\cdots,a_n'=E(x_n')\}.$$

第 2 步. S_3 选取较大随机数 $r'(r'\neq 0,1)$,加密向量$\boldsymbol{Y}'$,得到下式,记作 B'.

$$\{b_1'=E(y_1'+r'),\cdots,b_n'=E(y_n'+r')\}.$$

第 3 步. S_3 得到 A'和 B'后,利用 Boneh 加密方案的同态性计算

$$\begin{aligned}\boldsymbol{c}'&=\boldsymbol{e}(a_1',b_1')\boldsymbol{e}(a_2',b_2')\cdots\boldsymbol{e}(a_n',b_n')\\&=E(\langle \boldsymbol{X}',\boldsymbol{Y}'\rangle+r'(x_1'+x_2'+\cdots+x_n')).\end{aligned}$$

第 4 步. S_3 得到$\boldsymbol{c}'$后,计算 $E(x_1'+x_2'+\cdots+x_n')$,并利用 Boneh 加密方案的加法同态性计算

$$\begin{aligned}c_1'&=(E(x_1+x_2+\cdots+x_n))^{r'}\\&=E(r'(x_1+x_2+\cdots+x_n)),\end{aligned}$$

然后计算$\dfrac{\boldsymbol{c}'}{c_1'}=E(\langle \boldsymbol{X}',\boldsymbol{Y}'\rangle)$,记作 c_2'.

第 5 步. S_3 解密 c_2'得到内积$\langle \boldsymbol{X}',\boldsymbol{Y}'\rangle$.

于是得到

$$view_3(E(\boldsymbol{X}),E(\boldsymbol{Y}))=\{A,B,c\},$$

$$S_3(E(\boldsymbol{X}),E(\boldsymbol{Y}),E(\boldsymbol{X},\boldsymbol{Y}))=\{A',B',\boldsymbol{c}'\}.$$

由于 $c=E(\boldsymbol{X},\boldsymbol{Y})$,而 $E(\boldsymbol{X},\boldsymbol{Y})=E(\boldsymbol{X}',\boldsymbol{Y}')$,$\boldsymbol{c}'=E(\boldsymbol{X}',\boldsymbol{Y}')$,因此,$\boldsymbol{c}=\boldsymbol{c}'$. 由于 $A=E(\boldsymbol{X})$,$B=E(\boldsymbol{Y})$, 而 $A'=E(\boldsymbol{X}')$,$B'=E(\boldsymbol{Y}')$. 因此,$A\subseteq A'$. $B\subseteq B'$. 所以:

$$S_3(E(x),E(y),E(x,y))\subseteq view_3(x,y).$$

此过程证明了云服务器Server的视图$view_3(x,y)$只能从自己的输入 $E(x),E(y)$和输出的计算结果 $E(x,y)$中得到. $E(x),E(y)$分别为 Alice 和 Bob 加密的数据,Server 无法解密,因此得不到 x,y 的隐私,而 Server 所得到的输出计算结果 $E(x,y)$,为关于 x,y 的加密密文,Server 仍然无法解密. 因此,Server 的视图 $view_3(x,y)$中不包含 Alice 和 Bob

的任何信息，即整个云计算过程都得不到 Alice 和 Bob 的隐私.

(3) 构造模拟器 S_4

在本协议中，Server 和 Alice 合谋，此时两者获得的输入为 $\boldsymbol{X}, E(\boldsymbol{X}), E(\boldsymbol{Y})$，获得的输出为 $f_1(\boldsymbol{X},\boldsymbol{Y}), E(\boldsymbol{X},\boldsymbol{Y})$. 模拟器 S_3 接受 $\boldsymbol{X}, E(\boldsymbol{X}), E(\boldsymbol{Y})$ 和 $f_1(\boldsymbol{X},\boldsymbol{Y}), E(\boldsymbol{X},\boldsymbol{Y})$ 作为输入，综合(1)、(2)情况，用类似的方法进行工作(此处省略具体构造过程)，得到 $S_4(x,E(x),E(y),f_1(x,y),E(x,y)) \underline{c} view_4(x,y)$. 同理，用类似的方法可构造模拟器 S_5 使得 $S_5(y,E(x),E(y),f_2(x,y),E(x,y)) \underline{c} view_5(x,y)$.

此过程证明了即使云服务器 Server 和 Alice 合谋，但是合谋者子集的视图 $view_4(x,y)$，只能从 Server 和 Alice 的自身的输入 $x, E(x), E(y)$ 和自身所获得的输出 $f_1(x,y), E(x,y)$ 中得到，并不包含 Bob 的任何信息；对于 $view_5(x,y)$ 也是同样的道理. 因此整个过程，即使云服务器和一方合谋，也都不得另一方的隐私.　　证毕.

3.3.3　效率分析比较

本文设计的云外包计算下的内积协议(协议 1)，将同态运算过程(Evaluation)的计算任务交付给了云服务器，而两个用户只进行了最起初的加密和唯一的一个模幂运算、模乘运算和最后解密，大大减少了传统模式下内积运算中需要用户自身操作的同态运算(Evaluation)，因此可以为用户节省计算成本.

为了显示我们设计的云平台下内积协议确实比传统模式下内积协议为用户节省了计算成本，将本文内积协议(协议 1)和传统模式下的内积协议(文献[19-20])的计算复杂性做出对比分析. 由于文献[19-20]利用的都是同态加密思路，不过利用的同态加密算法不同而已. 文献[19]利用 Damgärd 等人[21]的同态算法，文献[20]利用 Paillier[22]的同态加密算法. 文献[19]只是把文献[20]的数域范围推广到负数范围内，但两者设计的方案几乎完全一致. 因此这里只给出本文的内积协议(协议 1)和文献[20]的内积协议比较.

为了便于两个协议比较，假设两个用户各自执有的向量都为 n 维. 将文献[20]利用的 Paillier[22]的模乘运算记为 M_{N_1}，模幂运算记为 M_{e_1}；本文协议 1 利用的 Boneh[18]的模乘运算记为 M_{N_2}，模幂运算记为 M_{e_2}. 每个内积协议都经历 3 个阶段，加密阶段(Encryption)，同态运算阶段(Evaluation)和解密阶段(Decryption). 因此将这 3 个阶段的用户操作运算次数作为衡量用户计算复杂性的标准，得到表 2.

表 2　本文内积协议与现有内积协议用户的计算成本比较

方案	加密阶段	同态运算		解密阶段
	模幂运算次数	模幂运算次数	模乘运算次数	模幂运算次数
文献[20]	nM_{e_1}	nM_{e_1}	$1M_{N_1}$	$1M_{e_1}$
我们的协议 1	$2nM_{e_2}$	$1M_{e_2}$	$1M_{N_2}$	$1M_{e_2}$

从表 2 可以看出，在同态运算阶段，由于文献[20]的内积协议是在传统模式下进行，这个阶段耗时最多的模幂运算都是依靠用户自身完成，因此用户需要操作 n 次；而我们的协议 1 由于在平计算平台下进行，所需要的模幂运算都由第三方(云服务器)完成，用户只需要操作 1 次即可，而模乘运算次数两者都是相同的，即这个阶段我们为用户减少了 $n-1$ 次模幂运算. 此外，虽然从表 2 看到我们协议 1 在加密阶段比文献[20]增加了 n 个模幂运算，但我们模幂运算的模数 $N_2=pq$，底数为 g；而文献[20]模幂运算的模数 $N_1=(N_2)^2=p^2q^2$，底数为 g^m，由于 $N_1 \gg N_2, g^m > g$. 即我们加密阶段增加的 n 个模幂运算都是低模数低指数运算，而文献[20]在同态运算阶段增加的 $n-1$ 个模幂运算却是高模数高指数运算，其余阶段两者都是相同的. 因此从总体上说，利用云外包技术，我们的内积协议可以极大的减少高数量级的模幂运算次数，为用户节省计算成本.

此外，我们这篇论文旨在为安全多方计算几何问题提供新的研究方向，意将安全多方几何计算架构在云平台下. 文献[20]是利用 Paillier[22]的加法同态性质设计的内积协议，而 Paillier 加密方案并不具有类同态的性质，因此并不能直接改造成为云计算平台下的内积协议.

3.3.4　实验仿真

为了进一步证实 3.3.3 节的理论分析结果，我们又给出了以下的仿真实验，从耗时上给读者一个更直观的观察. 我们的实验平台为 PC 机 3.0GHz Pentium (IV) system，用 VC 6.0 版本进行编译，用标准的/O2 编译器优化①. 分别取不同的模数 N 为 512 bits、1024 bits、2048 bits 进行实验，得到了不同模数和不同指数尺寸下运行一个模幂运算所需要的时间(单位为 ms)，如表 3 所示.

① Scott M. Multi-precision integer and rational arithmetic C/C++ library(MIRACL). URL: http://www.shamus.ie, 2003

表 3 模幂运算需要的时间

模数 N 的尺寸/bits	指数的尺寸/bits	3.0GHz Pentium(IV) /ms
512	160	0.64
1024	160	2.30
2024	160	8.23
2048	256	12.38

从表 3 可以看出，若固定模幂运算的指数大小不变，随着模数的增大，模幂运算需要的时间(单位为 ms)也在增大，增长倍数约为 3.5. 在 3.3.3 节中，将协议 1 在加密阶段增加的 n 个模幂运算所需要的时间记为 $T_2=nM_{e_2}$，模幂运算 M_{e_2} 的模数为 N_2；将文献[20]在同态运算阶段增加的 $n-1$ 个模幂运算所需要的时间记为 $T_1=(n-1)M_{e_1}$，模幂运算 M_{e_1} 的模数为 $N_1=(N_2)^2=p^2q^2$. 假设两个模幂运算 M_{e_1}，M_{e_2} 的指数大小相同(实际上 M_{e_2} 要比 M_{e_1} 的指数尺寸小)，得到两个阶段的耗时差：

$$\Delta T=T_1-T_2\approx(n-1)\frac{N_2}{2}3.5M_{e_2}-nM_{e_2}.$$

即我们的协议 1 比文献[20]在耗时上为用户至少节省了 ΔT ms.

若统一取两种模幂运算中的指数大小为160 bits，向量维数 $n=100$，模数大小分别取为 $N=N_2=$ 512 bits 或1024 bits 或 2048 bits. 得到不同的模数下，文献[20]中的 T_1 和本文协议 1 中的 T_2 的耗时及对应的耗时差 ΔT(时间都以 ms 记)，如图 3 所示.

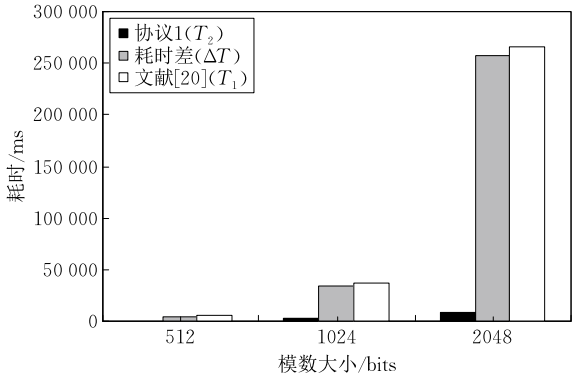

图 3 不同模数下 T_1 和 T_2 的耗时和对应的耗时差 ΔT

从图 3 可以看出，即使两个内积协议的指数大小相同，但由于两个内积的模数不同，而我们的协议 1(黑色柱状)利用了云计算平台，确实比文献[20](白色柱状)为用户节省了计算时间，并且随着模数的增大，节省的时间 ΔT(灰色柱状)更加明显. 由于模数和处理的明文范围有关，因此我们设计的云计算平台下的内积协议为处理大数据的问题提供了有力的工具.

4 具体协议

在第 3 节的基础上，本节给出云外包计算中 5 种空间位置关系的保密判定协议. 以下协议假设所有的参与者和云服务器都是在半诚实模型下，网络之间传输都是公开信道.

协议 2. 保密判断点与线的位置关系.

输入：Alice 保密输入空间点 $p_0(x_0,y_0,z_0)$，Bob 保密输入空间直线 $L:\begin{cases}A_1x+B_1y+C_1z+D_1=0\\A_2x+B_2y+C_2z+D_2=0\end{cases}$

输出：Alice 和 Bob 都知道点 p_0 是否在直线 L 上

1. Alice 持有向量 $\boldsymbol{a}=(x_0,y_0,z_0,1)$，Bob 持有向量 $\boldsymbol{b}=(A_1,B_1,C_1,D_1)$，$\boldsymbol{c}=(A_2,B_2,C_2,D_2)$.

2. Alice 选择 Boneh 加密方案，公布公钥$(\boldsymbol{e},G_1,G,n=pq,g,h)$，保密私钥$(p,q)$，并和 Bob 借助云外包计算，执行两次 3.3.1 节的保密内积协议，可计算出向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{b}$ 的内积、向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{c}$ 的内积.

3. 若$(\langle\boldsymbol{a},\boldsymbol{b}\rangle=0)\wedge(\langle\boldsymbol{a},\boldsymbol{c}\rangle=0)$，则点 p_0 在线 L 上，否则点 p_0 在线 L 外.

4. Alice 将结果告诉 Bob.

协议 3. 保密判断点与面的位置关系.

输入：Alice 保密输入空间点 $p_0(x_0,y_0,z_0)$，Bob 保密输入空间平面 $\Pi:Ax+By+Cz+D=0$

输出：Alice 和 Bob 都知道点 p_0 是否在平面 Π 上

1. Alice 持有向量 $\boldsymbol{a}=(x_0,y_0,z_0,1)$，Bob 持有向量 $\boldsymbol{b}=(A,B,C,D)$.

2. Alice 选择 Boneh 加密方案，公布公钥$(\boldsymbol{e},G_1,G,n=pq,g,h)$，保密私钥$(p,q)$. 并和 Bob 借助云外包计算，执行一次 3.3.1 节的保密内积协议，可计算出向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{b}$ 的内积.

3. 若$\langle\boldsymbol{a},\boldsymbol{b}\rangle=0$，则点 p_0 在面 Π 上，否则点 p_0 在面 Π 外.

4. Alice 将结果告诉 Bob.

协议 4. 保密判断线与线的位置关系.

输入：Alice 保密输入空间直线

$$L_1:\begin{cases}A_1x+B_1y+C_1z+D_1=0\\A_2x+B_2y+C_2z+D_2=0\end{cases},$$

Bob 保密输入空间直线

$$L_2:\begin{cases}A_3x+B_3y+C_3z+D_3=0\\A_4x+B_4y+C_4z+D_4=0\end{cases}$$

输出：Alice 和 Bob 都知道直线 L_1 与直线 L_2 是否平行、垂直、相交或重合

1. 利用叉积公式，Alice 可以得到直线 L_1 的方向向量 $\boldsymbol{e}_1=(m_1,n_1,l_1)=\begin{vmatrix}i&j&k\\A_1&B_1&C_1\\A_2&B_2&C_2\end{vmatrix}$. 同理，Bob 也可以得到直

线 L_2 的方向向量 $\boldsymbol{e}_2=(m_2,n_2,l_2)=\begin{vmatrix} i & j & k \\ A_3 & B_3 & C_3 \\ A_4 & B_4 & C_4 \end{vmatrix}$.

2. Alice 选择 Boneh 加密方案，公布公钥$(\boldsymbol{e},G_1,G,n=pq,g,h)$，保密私钥$(p,q)$，并和 Bob 借助云外包计算，调用 3.3.1 节的保密内积协议，求得两者内积$\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle$. 若$\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle=0$ 根据夹角公式$\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle=|\boldsymbol{e}_1||\boldsymbol{e}_2|\cos\theta$，则 $\cos\theta=0$，此时两条直线垂直；否则 Alice 可以得到 $\left|\frac{\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle}{|\boldsymbol{e}_1|}\right|$，记该数为 x. Bob 可以得到$|\boldsymbol{e}_2|$，记该数为 y. 利用 Hash 函数判断 $h(x)$ 与 $h(y)$是否相等. 若 $h(x)\neq h(y)$，则 $x\neq y$，即 $\cos\theta\neq\pm1$，则两条直线相交. 若 $h(x)=h(y)$，则 $x=y$，即 $\cos\theta=\pm1$，则直线 L_1 与直线 L_2 平行或重合，转入下面的步骤.

3. Alice 在 L_1 上任取一点$\{x_0,y_0,z_0\}$，得到向量 $\boldsymbol{a}=(x_0,y_0,z_0,1)$. Bob 持有向量 $\boldsymbol{b}=(A_1,B_1,C_1,D_1)$，$\boldsymbol{c}=(A_2,B_2,C_2,D_2)$. Alice 和 Bob 调用两次 3.3.1 节的保密内积协议，可计算出向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{b}$ 的内积、向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{c}$ 的内积. 若$(\langle\boldsymbol{a},\boldsymbol{b}\rangle=0)\wedge(\langle\boldsymbol{a},\boldsymbol{c}\rangle=0)$，则说明两直线 L_1 与 L_2 重合，否则平行.

4. Alice 将结果告诉 Bob.

协议 5.　保密判断线与面的位置关系.

输入：Alice 保密输入直线

$$L:\begin{cases}A_1x+B_1y+C_1z+D_1=0\\A_2x+B_2y+C_2z+D_2=0\end{cases},$$

Bob 保密输入平面 Π：$A_3x+B_3y+C_3z+D_3=0$

输出：Alice 和 Bob 都知道直线 L 与平面 Π 是否相交、平行垂直或重合

1. 利用叉积公式，Alice 可以得到直线 L 的方向向量

$$\boldsymbol{e}=(m,n,l)=\begin{vmatrix} i & j & k \\ A_1 & B_1 & C_1 \\ A_2 & B_2 & C_2 \end{vmatrix}.$$

Bob 持有平面 Π 的法向量 $\boldsymbol{d}=(A_3,B_3,C_3)$.

2. Alice 选择 Boneh 加密方案，公布公钥$(\boldsymbol{e},G_1,G,n=pq,g,h)$，保密私钥$(p,q)$，并和 Bob 借助云外包计算，调用 3.3.1 节的保密内积协议，可求得两者内积$\langle\boldsymbol{e},\boldsymbol{d}\rangle$. 根据夹角公式$\langle\boldsymbol{e},\boldsymbol{d}\rangle=|\boldsymbol{e}||\boldsymbol{d}|\cos\theta$，Alice 可以得到 $\left|\frac{\langle\boldsymbol{e},\boldsymbol{d}\rangle}{|\boldsymbol{e}|}\right|$，记该数为 x. Bob 可以得到$|\boldsymbol{d}|$，记该数为 y. 利用 Hash 函数判断 $h(x)$与 $h(y)$是否相等，若 $h(x)\neq h(y)$，则 $x\neq y$，即 $\cos\theta\neq\pm1$，则直线 L 与平面 Π 相交；若 $h(x)=h(y)$，则 $x=y$，即 $\cos\theta=\pm1$，即直线 L 与平面 Π 垂直；若$\langle\boldsymbol{e},\boldsymbol{d}\rangle=0$，那么 $\cos\theta=0$，则直线 L 与平面 Π 平行或重合；转入下面步骤.

3. Alice 在直线 L 上任取一点(x_1,y_1,z_1)，得到向量 $\boldsymbol{a}=(x_1,y_1,z_1,1)$，Bob 持有向量 $\boldsymbol{b}=(A_3,B_3,C_3,D_3)$，再次调用 3.3.1 节的保密内积协议，可求得内积$\langle\boldsymbol{a},\boldsymbol{b}\rangle$，若$\langle\boldsymbol{a},\boldsymbol{b}\rangle=0$，则直线 L 与平面 Π 重合，否则平行.

4. Alice 将结果告诉 Bob.

协议 6.　保密判断面与面的位置关系.

输入：Alice 保密输入平面：Π_1：$A_1x+B_1y+C_1z+D_1=0$，Bob 保密输入平面：Π_2：$A_2x+B_2y+C_2z+D_2=0$

输出：Alice 和 Bob 都知道平面 Π_1 与平面 Π_2 是否平行、相交、垂直或重合

1. Alice 得到平面 Π_1 的法向量 $\boldsymbol{d}_1=(A_1,B_1,C_1)$，Bob 得到平面 Π_2 的法向量 $\boldsymbol{d}_2=(A_2,B_2,C_2)$.

2. Alice 选择 Boneh 加密方案，公布公钥$(\boldsymbol{e},G_1,G,n=pq,g,h)$，保密私钥$(p,q)$，并和 Bob 借助云外包计算，调用 3.3.1 节的保密内积协议求得两者内积$\langle\boldsymbol{d}_1,\boldsymbol{d}_2\rangle$. 根据夹角公式$\langle\boldsymbol{d}_1,\boldsymbol{d}_2\rangle=|\boldsymbol{d}_1||\boldsymbol{d}_2|\cos\theta$，Alice 可得到 $\left|\frac{\langle\boldsymbol{d}_1,\boldsymbol{d}_2\rangle}{|\boldsymbol{d}_1|}\right|$，记该数为 x. Bob 可以得到$|\boldsymbol{d}_2|$，记该数为 y. 利用 Hash 函数判断 $h(x)$与 $h(y)$是否相等，若 $h(x)\neq h(y)$，则 $x\neq y$，即 $\cos\theta\neq\pm1$，平面 Π_1 与平面 Π_2 相交；若$\langle\boldsymbol{d}_1,\boldsymbol{d}_2\rangle=0$，那么 $\cos\theta=0$，即平面 Π_1 与平面 Π_2 垂直；若 $h(x)=h(y)$，则 $x=y$，那么 $\cos\theta=\pm1$，则平面 Π_1 与平面 Π_2 平行或重合；转入下面步骤.

3. Alice 在平面 Π_1 上任取一点$\{x_0,y_0,z_0\}$，得到向量 $\boldsymbol{a}=(x_0,y_0,z_0,1)$. Bob 持有向量 $\boldsymbol{b}=(A_2,B_2,C_2,D_2)$，再次调用 3.3.1 节的保密内积协议，可求得内积$\langle\boldsymbol{a},\boldsymbol{b}\rangle$. 若$\langle\boldsymbol{a},\boldsymbol{b}\rangle=0$，则平面 Π_1 与平面 Π_2 重合，否则平行.

注意：在以上 5 个协议中，由于我们的协议都是通过夹角完成的. 若 hash 值相等，那么一方会得到另一方向量的模长. 但这泄露的唯一信息，不能推出向量的任何信息. 关于这一点，Du 等人在文献[23]中专门有论证：在设计协议时，如果在降低完美安全性的程度上，允许泄露的信息并不影响方案的有效性，那么这就是可接受性安全. 而可接受性安全要根据具体问题，具体设定.

5　安全性分析

在本节，我们应用 2.1 节的模拟范例给出本文 5 个协议的安全性证明. 由于协议 2、协议 3 的安全性依靠基本协议 1，因此由定理 1 可以直接得到定理 2、定理 3 的结论.

定理 2.　协议 2 保密地判断了点与线的位置关系.

定理 3.　协议 3 保密地判断了点与面的位置关系.

由于协议 4～6 证明过程类似. 因此为了节省篇幅，我们以证明协议 4 安全性为主，而对于其余协议，只给出安全性结论.

定理 4.　协议 4 保密地判断了线与线的位置关系.

证明.　协议 4 的证明过程类似基本协议 1，也

需要构造 5 个模拟器 S_1,S_2,S_3,S_4,S_5，分为 3 种情况进行证明. 由于模拟器的构造过程类似，为了节省篇幅，只给出模拟器 S_1 的构造过程，其余可以参照基本协议 1 进行构造.

在本协议中

$$\begin{aligned}f_1(\boldsymbol{X},\boldsymbol{Y})&=f_2(\boldsymbol{X},\boldsymbol{Y})\\&=(L_1\perp L_2)\vee(L_1//L_2)\vee\\&\quad(L_1=L_2)\vee(L_1\text{相交 }L_2)\end{aligned}$$

或者

$$\begin{aligned}f_1(\boldsymbol{X},\boldsymbol{Y})&=f_2(\boldsymbol{X},\boldsymbol{Y})\\&=(\neg(L_1\perp L_2))\vee(\neg(L_1//L_2))\vee\\&\quad(\neg(L_1=L_2))\vee(\neg(L_1\text{相交 }L_2)).\end{aligned}$$

假设 $f_1(\boldsymbol{X},\boldsymbol{Y})=f_2(\boldsymbol{X},\boldsymbol{Y})=(L_1//L_2)$，构造模拟器 S_1. S_1 接受 $(\boldsymbol{X},f_1(\boldsymbol{X},\boldsymbol{Y}))$ 作为输入，按如下方式工作：

第 1 步. S_1 接受输入 $(\boldsymbol{X},f_1(\boldsymbol{X},\boldsymbol{Y}))=(L_1,(L_1//L_2))$ 后，首先随机选取一条直线 L_2'：

$$\begin{cases}A_3'x+B_3'y+C_3'z+D_3'=0\\A_4'x+B_4'y+C_4'z+D_4'=0\end{cases}$$

使得 $f_1(L_1,L_2)=f_1(L_1,L_2')$，然后用 (L_1,L_2') 进行模拟. 按照协议 4，S_1 分别得到直线 L_1、直线 L_2' 的方向向量 $\boldsymbol{e}_1$、$\boldsymbol{e}_2'$.

第 2 步. S_1 调用 3.3.1 节的保密内积协议，得到内积 $\langle\boldsymbol{e}_1,\boldsymbol{e}_2'\rangle$.

第 3 步. S_1 计算 $\left|\frac{\langle\boldsymbol{e}_1,\boldsymbol{e}_2'\rangle}{|\boldsymbol{e}_1|}\right|$，记该数为 x'，并计算 $|\boldsymbol{e}_2'|$，记该数为 y'. 利用 Hash 函数计算 $h(x')$ 与 $h(y')$，比较两者的值是否相等，将结果记为 H'. S_1 在 L_1 上任取一点 $\{x_0,y_0,z_0\}$，得到向量 $\boldsymbol{a}=(x_0,y_0,z_0,1)$. 在 L_2' 上得到向量 $\boldsymbol{b}'=(A_3',B_3',C_3',D_3')$，$\boldsymbol{c}'=(A_4',B_4',C_4',D_4')$，$S_1$ 调用两次 3.3.1 节的保密内积协议，计算向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{b}'$ 的内积 $\langle\boldsymbol{a},\boldsymbol{b}'\rangle$、向量 $\boldsymbol{a}$ 与向量 $\boldsymbol{c}'$ 的内积 $\langle\boldsymbol{a},\boldsymbol{c}'\rangle$，得到结果 C'. 于是得到

$$\begin{aligned}view_1(\boldsymbol{X},\boldsymbol{Y})&=\{L_1,\boldsymbol{e}_1,\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle,x,h(x),\\&\quad h(y),H,\boldsymbol{a},\langle\boldsymbol{a},\boldsymbol{b}\rangle,\langle\boldsymbol{a},\boldsymbol{c}\rangle,C\},\\S_1(\boldsymbol{X},f_1(\boldsymbol{X},\boldsymbol{Y}))&=\{L_1,\boldsymbol{e}_1,\langle\boldsymbol{e}_1,\boldsymbol{e}_2'\rangle,x',h(x'),h(y'),\\&\quad H',\boldsymbol{a},\langle\boldsymbol{a},\boldsymbol{b}'\rangle,\langle\boldsymbol{a},\boldsymbol{c}'\rangle,C'\}.\end{aligned}$$

由于 $C=f_1(L_1,L_2)=(L_1//L_2)$，而 $C'=f_1(L_1,L_2')=(L_1//L_2')$，$f_1(L_1,L_2)=f_1(L_1,L_2')$，因此 $C=C'$. 由于 $x'=\left|\frac{\langle\boldsymbol{e}_1,\boldsymbol{e}_2'\rangle}{|\boldsymbol{e}_1|}\right|$，$x=\left|\frac{\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle}{|\boldsymbol{e}_1|}\right|$，而在定理 1 中已经证明内积 $\langle\boldsymbol{e}_1,\boldsymbol{e}_2'\rangle\subseteq\langle\boldsymbol{e}_1,\boldsymbol{e}_2\rangle$，因此 $h(x')\subseteq h(x)$，$x'\subseteq x$. L_2' 为 S_1 随机取的一条直线，而 S_1 不知道 L_2 的具体位置，且 L_2 为随机分布. 因此 $\boldsymbol{e}_2$ 和 $\boldsymbol{e}_2'$ 计算不可区分，即得到 $h(y)=h(\boldsymbol{e}_2)$，$h(y')=h(\boldsymbol{e}_2')$，因此 $h(y)\subseteq h(y')$，$\langle\boldsymbol{a},\boldsymbol{b}\rangle\subseteq\langle\boldsymbol{a},\boldsymbol{b}'\rangle$，$\langle\boldsymbol{a},\boldsymbol{c}\rangle\subseteq\langle\boldsymbol{a},\boldsymbol{c}'\rangle$，$H\subseteq H'$. 又因为 $output_2(\boldsymbol{X},\boldsymbol{Y})=f_2(\boldsymbol{X},\boldsymbol{Y})=(L_1//L_2')$，所以：

$$S_1(x,f_1(x,y))\subseteq view_1(x,y).$$

用类似的构造方法(参照基本协议 1)可以构造其他 4 个模拟器，分别得到下式：

$$S_2(y,f_2(x,y))\subseteq view_2(x,y),$$

$$S_3(E(x),E(y),E(x,y))\subseteq view_3(x,y),$$

$$S_4(x,E(x),E(y),f_1(x,y),E(x,y))\subseteq view_4(x,y),$$

$$S_5(y,E(x),E(y),f_2(x,y),E(x,y))\subseteq view_5(x,y).$$

证毕.

定理 5. 协议 5 保密地判断了线与面的位置关系.

定理 6. 协议 6 保密地判断了面与面的位置关系.

定理 5、定理 6 的证明过程类似定理 4，为了节省篇幅，这里不再一一赘述.

6 效率分析与比较

本节给出本文协议和引言中的相关文献[14-15]在效率以及性能方面的分析和比较. 由于这些文献涉及的空间位置问题不全相同，为了便于做出比较，在效率方面，统一用本文和相关文献[14-15]中都出现的面与面位置问题保密判断协议做出对比. 模乘运算记为 M_{N_2}，模幂运算记为 M_{e_2}，乘法运算记为 M.

(1) 计算复杂度

以上方案利用的基础协议中，比较复杂的是内积协议，而内积协议都是通过同态加密算法得到. 为了便于比较，假设所有方案中的内积协议都采用本文的 Boneh[18] 同态加密方案. 在整个执行过程中，计算开销较大的是模幂运算、模乘运算、乘法运算. 因此把各个方案中用户需要调用的内积协议总次数，以及用户需要的模幂运算、模乘运算、乘法运算总个数作为衡量计算复杂性的指标，其他忽略不计.

文献[14]的面面位置关系保密判断(协议 4)：使用了 1 次内积协议和 2 次数据对应成比例协议. 而数据对应成比例协议每调用一次，就需要循环语句调用内积协议，因此使用的内积协议总次数为 6 次. 按照原文[14]的分析，需要 $34m$ 个模幂运算、$8m$ 个模乘运算，其中，m 为安全参数. 因此该协议的计算开销为 $8mM_{N_2}+34mM_{e_2}$.

文献[15]的面面位置关系保密判断(协议 4)：计算了 3 次内积，但并没有调用内积协议，用户主要

使用了矩阵运算.进行了 3 个 3 阶的矩阵运算,9 个数乘运算,因此乘法运算总个数为 $36M$.

本文的面面位置关系保密判断(协议 6):协议调用的 2 次内积协议被外包计算给云,用户本身并没有计算内积.用户只计算了 2 个幂运算、2 个模乘运算、14 个加密运算、2 个解密运算.因此该协议总的计算开销为 $2M_{N_2}+18M_{e_2}$.

(2)通信复杂度

衡量通信复杂度的指标用协议交换信息的比特数,或者用通信轮数,在多方保密计算研究中通常用轮数(round).

文献[14]的面面位置关系保密判断(协议 4):使用了 1 次内积协议、2 次数据对应成比例、1 次比较协议.第 1 次调用数据对应成比例协议,交互轮数为 $6m$,第 2 次调用数据对应成比例协议,交互轮数为 $8m$,其中 m 为安全参数.总共进行了 $14m+4$ 轮交互.

文献[15]的面面位置关系保密判断(协议 4):总共进行了 2 轮交互.

本文的面面位置关系保密判断(协议 6):使用了 2 次内积协议和 1 次比较协议,内积协议每调用一次就进行了 3 轮交互.因此两个用户包括和云服务器,总共进行了 8 轮交互.

(3)性能

以协议判断的空间位置关系多少、解决的问题多少、是否适合云计算平台作为衡量性能的指标,其中×表示不具有此性能,√表示具有此性能.

综合以上分析,本文与现有文献[14-15]在面面保密判定协议的效率对比如表 4.本文与现有文献[14-15]方案的性能对比如表 5.

表 4　本文协议与现有协议的效率比较

方案	计算开销		通信开销
	内积协议	操作运算	
文献[14](协议 4)	$2n+1$	$8mM_{N_2}+34mM_{e_2}$	$(14m+4)$ rounds
文献[15](协议 4)	0	$36M$	2 rounds
我们的协议 6	0	$2M_{N_2}+18M_{e_2}$	8 rounds

表 5　本文方案与现有方案的性能比较

方案	文献[14]					文献[15]			Ours				
解决的问题	点线	点面	线线	线面	面面	点面	线面	面面	点线	点面	线线	线面	面面
判断的位置关系	属于	属于	异面	垂直	垂直	属于	×	×	属于	属于	垂直	垂直	垂直
	×	×	平行	平行	平行	×	平行	平行	×	×	平行	平行	平行
	×	×	相交	×	相交	×	相交	相交	×	×	相交	相交	相交
	×	×	×	重合	重合	×	×	×	×	×	重合	重合	重合
云计算平台	×					×			√				

从表 4 可以看出,文献[15]的效率是最优的,但从表 5 可以看出,该文献能解决的问题最少(只能解决 3 种),而且能判断的位置关系最单一,因此该文献虽然效率较高,但性能最差,应用比较受限.若要解决和我们的方案以及文献[14]同样多的问题,判断更多的位置关系,还需要再寻找其他的方法.

从表 5 可以看出,文献[14]和我们的方案能解决的问题相同,但文献[14]由于利用了复杂的数据对应成比例协议,需要使用循环语句不断调用内积协议.而我们的方案由于云计算技术的使用,将内积协议外包给云,为用户节省了计算成本,因此在解决同样多的问题下,从表 4 可以看出我们的效率优于文献[14].此外,从表 5 还可以看出,虽然我们的方案和文献[14]能处理的问题相同,但能判断的位置关系却比其他已存方案都多,并且我们的方案第一次给出了云计算平台下处理空间立体几何问题的解决方法.

7　总结和开放问题

空间位置关系的保密计算属于安全多方计算中的空间几何问题,现实中很多问题都能归结于此.已存方案由于转化方法的原因使得用户的计算复杂性较高,或者解决的问题有限、能判断的位置关系单一,并且都是在传统模式下进行的.本文首次探索了云计算平台下安全多方计算几何问题的解决方法,首先将原问题转化成夹角问题,接着设计了适用于云外包计算下的内积协议,然后基于该内积协议,解决了空间几何中 5 种位置关系的保密判断问题.由于问题的巧妙转化,使得我们的方案能解决的问题也更多,判断的位置关系也更加多样化.并且由于云计算技术的使用,为用户节省了大量的计算成本.

本文的协议为了提高效率,以接受性安全作为牺牲代价,有一定的信息泄露(如向量的模长),虽然不影响协议的执行和有效性,但并未取得完美性安

全.因此,可以进一步考虑如何设计关于此问题的完美安全性下的高效协议.此外,将传统安全多方计算中的大数据问题,借助云计算的能力有效解决,是安全多方计算的一个研究新方向,但将安全多方计算问题架构在云平台下时,需要不同以往的新技术,这也将是未来研究的工作之一.

致 谢 在此,我们向对本文的工作给予支持和建议的同行,尤其是陕西师范大学计算科学学院 614 实验室的老师和同学表示感谢!

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] Freudiger J, Rane S, Brito A E, Uzun E. Privacy preserving data quality assessment for high-fidelity data sharing// Proceedings of the ACM Workshop on Information Sharing & Collaborative Security. Scottsdale, USA, 2014: 21-29

[3] Li X Y, Jung T. Search me if you can: Privacy-preserving location query service//Proceedings of the IEEE International Conference on Computer Communications. Turin, Italy, 2013: 2760-2768

[4] Song D, Sim J, Park K, et al. A privacy-preserving continuous location monitoring system for location-based services. International Journal of Distributed Sensor Networks, 2015, 2015(14): 137-142

[5] Yang Jing, Zhao Jia-Shi, Zhang Jian-Pei. A privacy preservation method for high dimensional data mining. Acta Electronica Sinica, 2013, 41(11): 2187-2192(in Chinese)
(杨静, 赵家石, 张健沛. 一种面向高维数据挖掘的隐私保护法. 电子学报, 2013, 41(11): 2187-2192)

[6] Samanthula B K, Elmehdwi Y, Howser G, et al. A secure data sharing and query processing framework via federation of cloud computing. Information Systems, 2015, 48(3): 196-212

[7] Campos R, Dias G, Jorge A M, et al. Survey of temporal information retrieval and related applications. ACM Computing Surveys, 2015, 47(2): 1-14

[8] Kerschbaum F. Privacy-preserving computation//Preneel B, Ikonomou D eds. Privacy Technologies and Policy. Berlin Heidelberg: Springer, 2014: 41-54

[9] Yang B, Sun A, Zhang W. Secure two-party protocols on planar circles. Journal of Information, 2011, 8(1): 29-40

[10] Liu L, Wu C, Li S. Two privacy-preserving protocols for point-curve relation. Journal of Electronics, 2012, 29(5): 422-430

[11] 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(11): 94-99

[12] Liu L, Chen X, Lou W. A secure three-party computational protocol for triangle area. International Journal of Information Security, 2016, 15(1): 1-13

[13] He F, Wang T. Research and application of secure multi-party computation in several computational geometry problems// Proceedings of the International Conference on IEEE Control and Electronics Engineering. Xi'an, China, 2012: 1434-1437

[14] Luo Yong-Long, Huang Liu-Sheng, Jing Wei-Wei, et al. Privacy protection in the relative position determination for two spatial geometric objects. Journal of Computer Research and Development, 2006, 43(3): 410-416(in Chinese)
(罗永龙, 黄刘生, 荆巍巍等. 空间几何对象相对位置判定中的私有信息保护. 计算机研究与发展, 2006, 43(3): 410-416)

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

[16] Goldreich O. Foundations of cryptography: Basic applications. London, UK: Cambridge University Press, 2004: 599-729

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

[18] Boneh D, Goh E J, Nissim K. Evaluating 2-DNF formulas on ciphertexts//Proceedings of the 2nd Theory of Cryptography Conference. Cambridge, USA, 2005: 325-341

[19] Yang B, Yang C H, Yu Y, et al. A secure scalar product protocol and its applications to computational Geometry. Journal of Computers, 2013, 8(8): 2018-2026

[20] Goethals B, Laur S, Lipmaa H, et al. On private scalar product computation for privacy-preserving data mining// Proceedings of the International Conference on Information Security and Cryptology. Seoul, Korea, 2005: 104-120

[21] Damgärd I, Jurik M. A length-flexible threshold cryptosystem with applications//Proceedings of the 8th Australasian Conference on Information Security and Privacy (ACISP'03). Wollongong, Australia, 2003: 350-364

[22] Paillier P. Public-key cryptosystems based on composite degree residue classes//Proceedings of the Advances in Cryptology-EuroCrypt 1999. Prague, Czech Republic, 1999: 223-238

[23] Du W, Zhan Z. A practical approach to solve secure multi-party computation problems//Proceedings of the Workshop on New Security Paradigms. Virginia Beach, USA, 2002: 127-135

CHEN Zhen-Hua, born in 1976, associate professor, Ph. D. Her research interests include secret sharing and secure multi-party computation, etc.

LI Shun-Dong, born in 1963, professor, Ph.D. supervisor. His research interests include information hiding and secure multi-party computation, etc.

HUANG Qiong, born in 1981, professor, M.S. supervisor. His research interests include information hiding and search on encrypted data, etc.

DING Yong, born in 1976, professor, M. S. supervisor. His research interests include information hiding and search on encrypted data, etc.

SUN Man, born in 1990, M. S. candidate. Her research interests include information security and secure multi-party computation, etc.

Background

Transferring the traditional pattern of secure multi-party computation to the cloud computing pattern is a new research direction. However, there are significant differences between both patterns because of the existence of distrusted cloud server (third party) in cloud computing. Compared with traditional pattern, this pattern raised many new problems, which appeal new techniques. Aiming to this field, in this paper transferring privacy-preserving determination of spatial location-relation in traditional manner into one in cloud computing pattern has important significance to study secure multi-party computation. Up to now, the most existing schemes about privacy-preserving determination of spatial location-relation transform the original problem into the distance problem or the correspondingly proportional data problem and solve it with traditional pattern. These approaches burden the user's computation overhead or limit the range of determining location-relation. So, the research on how to improve its efficiency and universality has important significance.

This research is supported by the National Natural Science Foundation of China (Nos. 61272435, 61472146), the Research Fund for the Doctoral Program of Xi'an University of Science and Technology (No. 2015QDJ008), the Open Foundation of State Key Laboratory of Information Security (No. 2016-MS-19), the Guangdong Natural Science Funds for Distinguished Young Scholar (No. 2014A030306021), the Guangdong Program for Special Support of Top-Notch Young Professionals (No. 2015TQ01X796), the Pearl River Nova Program of Guangzhou (No. 201610010037), and the CICAEET Fund and the PAPD Fund (No. KJR1615).