保护私有信息的图形相似判定

李顺东[1],杨晓莉[1],左祥建[1],周素芳[1],亢 佳[1],刘 新[1,2]
(1.陕西师范大学计算机科学学院,陕西西安 710119;2.内蒙古科技大学信息工程学院,内蒙古包头 014010)

摘 要: 目前,关于几何图形的相似问题仅限于多边形的相似,而一般几何图形相似的问题还没有研究.本文利用单向散列函数首先设计了保密判断两个数是否相等的协议、保密矩阵和向量是否相等的协议;最终,利用矩阵和向量相等的协议设计了保密判断图形是否同构和图形是否相似的协议.给出了以上协议的安全性证明、仿真实验与效率分析,实验数据表明本文保密的图形相似判定协议效率是两个多边形相似协议效率的889倍.图形相似的保密判定问题是一个全新的安全多方计算几何问题,本文研究成果可应用在分子生物学、机械工程和地形匹配等领域.

关键词: 密码学;安全多方计算;计算几何;图形相似;图形同构
中图分类号: TP302 **文献标识码:** A **文章编号:** 0372-2112 (2017)09-2184-06
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2017.09.019

Privacy-Preserving Graphical Similarity Determination

LI Shun-dong[1], YANG Xiao-li[1], ZUO Xiang-jian[1], ZHOU Su-fang[1], KANG Jia[1], LIU Xin[1,2]
(1. *School of Computer Science, Shaanxi Normal University, Xi'an, Shaanxi* 710119, *China*;
2. *School of Information Engineering, Inner Mongolia University of Science and Technology, Baotou, Inner Mongolia* 014010, *China*)

Abstract: At present, graphical similarity is limited to polygonal similarity, but the problem of general graphical similarity has not been studied. We first present protocols for privately determining whether two numbers, matrices or vectors are equal based on one-way hash function. Finally, we design protocols to privately determine whether two special graphics are isomorphic, and whether two graphics are similar. We prove the security of the protocols, implement them on a personal computer and analyze their efficiency. The simulation shows that the protocol of two similar graphics is 889 times as fast as the protocol of two similar polygons. Privately determining whether two graphs are similar is completely a new secure multiparty computation problem. It has application prospects in the field of the molecular biology, mechanical engineering and terrestrial matching, etc.

Key words: cryptography; secure multi-party computation; computational geometry; graphics similar; graphics isomorphism

1 引言

安全多方计算由图灵奖获得者姚期智教授首次提出[1],经过了Goldreich,Micali等人的发展[2,3],成为近年密码学研究的热点问题之一[4~6].安全多方计算是信息社会隐私保护的重要技术,它使拥有私有数据的参与者能够合作利用这些私有数据进行某些联合计算,同时又不泄露自己的私有数据,因而使人们能够最大限度地利用私有数据而不破坏数据的隐私性.

保护隐私的计算几何(Secure Multiparty Computational Geometry,SMCG)是安全多方计算的新领域[7],主要研究计算几何的信息安全问题,尤其是分布式系统中合作用户的隐私数据保护问题.具体地说,SMCG问题的研究就是要设计出相应的协议,使用户能够保密地解决某些计算几何问题.

在保密的计算几何方面,文献[7,8]提出了安全多方点包含问题、两个凸多边形相交问题;文献[9]研究了两方的点包含问题;文献[10]研究了保密的几何距离计算问题;文献[11]研究了多边形的点包含问题;文献[12]研究了安全的多边形面积计算问题;文献[13]

收稿日期:2016-08-07;修回日期:2016-10-19;责任编辑:覃怀银
基金项目:国家自然科学基金(No. 61272435);内蒙古自然科学基金(No. 2017MS0602);中央高校基本科研业务费专项资金资助(No. 2016TS061);内蒙古自治区高等学校科学研究项目(No. NJZY17164)

研究了多边形相似问题. 但关于相似问题的安全多方计算研究仍然是初步的;关于几何图形的相似问题仅限于多边形的相似[13]. 而一般几何图形相似的问题还没有研究. 但一般几何图形的相似有更广泛的应用.

随着几何图形检索技术迅速发展,其实用价值逐步显现,在生物分子、机械零件、地形匹配等领域都得到了广泛应用[14],都用到了图形相似问题的判定. 要进行保密几何图形检索,就要保密判断图形是否相似,用判断是否全等来判断是否为同一个对象将得出非同一对象的错误结论. 因此,研究保密判断图形相似的问题是十分必要的.

2　预备知识

2.1　字母表与连接运算[15]

字母表是对象的有穷非空集合 M,M 中符号的 n 元组称作 M 上的字或字符串,M 中所有字符串的全体记作 M^*,例如,所有的自然数都是 0,1,2,・・・,9 字母表上的字符串,所有的二进制数都是 0,1 字母表上的字符串,字符串的连接定义如下:

$$\begin{cases}\mathrm{Concat}_n^1(u)=u,\\ \mathrm{Concat}_n^{m+1}(u_1,\cdots,u_m,u_{m+1})=zu_{m+1}.\end{cases}$$

其中 $z=\mathrm{Concat}_n^m(u_1,\cdots,u_m)$,对于给定的字符串 $u_1,\cdots,u_m\in M^*$,$\mathrm{Concat}_n^m(u_1,\cdots,u_m)$ 就是把字符串 $u_1,\cdots,u_m$ 一个接着一个放在一起所得到的新的字符串.

2.2　安全性定义

半诚实参与者[16]本文方案均假设安全多方计算的参与者为半诚实参与者. 简单地说,所谓半诚实参与者是指参与者在协议执行过程中将不折不扣地执行协议,但他们也会保留计算的中间结果试图推导出其他参与者的输入.

定义 1[3]**(半诚实参与者的保密性)**　对于一个函数 f,如果存在概率多项式时间算法 S_1 与 S_2(也称这样的多项式时间算法为模拟器)使得

$$\{S_1(x,f_1(x,y)),f_2(x,y)\}_{x,y}\overset{c}{\equiv}\{\mathrm{view}_1^{\pi}(x,y),\mathrm{output}_2^{\pi}(x,y)\}_{x,y}\tag{1}$$

$$\{f_1(x,y),S_2(y,f_2(x,y))\}_{x,y}\overset{c}{\equiv}\{\mathrm{output}_1^{\pi}(x,y),\mathrm{view}_2^{\pi}(x,y)\}_{x,y}\tag{2}$$

其中 $\overset{c}{\equiv}$ 表示计算上不可区分. 则认为 π 保密地计算 f.

要证明一个多方计算方案是保密的,就必须构造满足式(1)和(2)的模拟器 S_1 与 S_2.

3　图形相似协议

3.1　基于对称加密算法的保密数相等判定协议

比较两个数是否相等问题　Alice 有一个数 a,Bob 有一个数 b,他们想保密比较 a 是否等于 b. 即如果 $a\neq b$,比较过程不泄露 a,b 的任何信息.

具体方案如下:Alice 和 Bob 分别任意选择随机数 $r\in\{0,1\}^m$,$s\in\{0,1\}^n(m,n>64)$,Alice 计算 $c=a\oplus r$,发送给 Bob;Bob 计算 $d=b\oplus s$,发送给 Alice;Alice 计算 $a'=d\oplus r=b\oplus s\oplus r$,Bob 计算 $b'=c\oplus s=a\oplus r\oplus s$. Alice 和 Bob 分别对 a',b' 进行哈希运算,得到 $\mathrm{hash}(a')$,$\mathrm{hash}(b')$ 并交换. 如果 $\mathrm{hash}(a')=\mathrm{hash}(b')$,那么 $a=b$;否则,$a\neq b$. 为叙述简单定义一个二元谓词如下:

$$P(a,b)=\begin{cases}0,\text{如果 } a=b;\\ 1,\text{如果 } a\neq b.\end{cases}$$

协议 1　保密数相等判定协议

输入:Alice 输入 a,Bob 输入 b.

输出:$P(a,b)$.

(1) Alice 和 Bob 分别任意选择随机数 $r\in\{0,1\}^m$,$s\in\{0,1\}^n(m,n>64)$,计算 $c=a\oplus r$,$d=b\oplus s$,并交换 c,d.

(2) Alice 和 Bob 分别计算 $a'=d\oplus r=b\oplus s\oplus r$,$b'=c\oplus s=a\oplus r\oplus s$.

(3) Alice 和 Bob 分别对 a',b' 进行哈希运算,得到 $\mathrm{hash}(a')$,$\mathrm{hash}(b')$ 并交换.

(4) 如果 $\mathrm{hash}(a')=\mathrm{hash}(b')$,输出 $P(a,b)=0$;否则,输出 $P(a,b)=1$.

定理 1　保密数相等判定协议 1 是安全的.

证明　通过构造使等式(1)和(2)成立的模拟器 S_1、S_2 证明本定理.

在本方案中

$$\mathrm{view}_1^{\pi}(a,b)=(a,r,d,a',(\mathrm{hash}(a'),\mathrm{hash}(b')),P(a,b))$$

$$f_1(a,b)=f_2(a,b)=\mathrm{output}_1^{\pi}(a,b)=\mathrm{output}_2^{\pi}(a,b)=P(a,b)$$

其中:a,b 分别是 Alice 和 Bob 的输入,r 是 Alice 选择的随机数,a',$\mathrm{hash}(a')$ 是 Alice 计算的结果,d,$\mathrm{hash}(b')$ 是 Bob 发送给 Alice 的数. 首先构造 S_1 来模拟 $\mathrm{view}_1^{\pi}(a,b)$ 使得式(1)成立. 模拟过程如下:

(1)接受输入 $(a,f_1(a,b))$,根据 $f_1(a,b)$ 的值,选定数 b_1,使得 $f_1(a,b_1)=f_1(a,b)=P(a,b)$,选择随机数 $s'\in\{0,1\}^m$,S_1 计算 $c=a\oplus r$,$d'=b_1\oplus s'$.

(2)S_1 计算 $a_1'=d'\oplus r=b_1\oplus s'\oplus r$,$\mathrm{hash}(a_1')$,$b_1'=c\oplus s'=a\oplus r\oplus s'$,$\mathrm{hash}(b_1')$.

(3)S_1 通过比较 $\mathrm{hash}(a_1')$ 和 $\mathrm{hash}(b_1')$ 是否相等来比较 a 和 b_1 是否相等. 令

$$S_1(a,f_1(a,b))=\{a,r,d',a_1',(\mathrm{hash}(a_1'),\mathrm{hash}(b_1')),P(a,b)\},$$

因为

$$d\overset{c}{\equiv}d',a'\overset{c}{\equiv}a_1',\mathrm{hash}(a')\overset{c}{\equiv}\mathrm{hash}(a_1'),\mathrm{hash}(b')\overset{c}{\equiv}\mathrm{hash}(b_1'),$$

所以

$\{S_1(a,f_1(a,b)),f_2(a,b)\}\overset{c}{\equiv}\{\text{view}_1^{\pi}(a,b),\text{output}_2^{\pi}(a,b)\}$

类似地,还可以构造 S_2 使

$\{f_1(a,b),S_2(b,f_2(a,b))\}\overset{c}{\equiv}\{\text{output}_1^{\pi}(a,b),\text{view}_2^{\pi}(a,b)\}$

定理 1 证毕.

3.2 保密矩阵相等判定协议

假设 Alice 和 Bob 分别有 $m\times n$ 维矩阵 $\boldsymbol{A}=(a_{ij})_{m\times n}$,$B=(b_{ij})_{m\times n}$,两人想在不暴露自己信息的情况下,比较两个矩阵是否相等. 该方案的思想是:将矩阵的每一行连接起来构成一个较长的序列,通过哈希函数比较这两个序列的哈希值相等就可以保密判断矩阵相等. 具体做法如下:

以矩阵 $\boldsymbol{A}=(a_{ij})_{m\times n}$ 为例,将矩阵第 1 行元素 $a_{11},\cdots,a_{1n}$、…、第 m 行元素 $a_{m1},\cdots,a_{mn}$ 连起来,构成一个序列数,Alice 通过在高位数的位置补零的方法使序列的每个数的十进制位数与最大数的位数相同,得到

$$x_A=a_{11}'\cdots a_{1n}'a_{21}'\cdots a_{2n}'\cdots a_{m1}'\cdots a_{mn}' \tag{3}$$

Bob 用同样的方法得到 $x_B=b_{11}'\cdots b_{mn}'$,如果 x_A,x_B 的比特数大于 64,直接比较这两个哈希值相等即可判断两个矩阵相等. 但是当矩阵中的元素比较少且全是 0,1 元素时,哈希函数对穷举攻击是敏感的. 为了避免信息泄露,调用协议 1 保密比较这两个哈希值. 该方法思想类似于协议 1. 为叙述简单定义二元谓词如下:

$$P(\boldsymbol{A},\boldsymbol{B})=\begin{cases}0,\text{如果 }\boldsymbol{A},\boldsymbol{B}\text{ 相等;}\\1,\text{如果 }\boldsymbol{A},\boldsymbol{B}\text{ 不相等.}\end{cases}$$

协议 2 保密的矩阵相等判定协议

输入:Alice 输入矩阵 $\boldsymbol{A}=(a_{ij})_{m\times n}$,Bob 输入矩阵 $\boldsymbol{B}=(b_{ij})_{m\times n}$.

输出:$P(\boldsymbol{A},\boldsymbol{B})$.

(1) Alice 和 Bob 分别将矩阵 $\boldsymbol{A},\boldsymbol{B}$ 转换成形如式(3)的序列数 $x_A=a_{11}'\cdots a_{mn}'$($x_B=b_{11}'\cdots b_{mn}'$).

(2) Alice 和 Bob 分别选择随机数 $r\in\{0,1\}^k$,$s\in\{0,1\}^t$($k,t>m\times n$),并计算 $C=x_A\oplus r$,$D=x_B\oplus s$,交换 C,D.

(3) Alice 和 Bob 分别计算 $E=D\oplus r=x_B\oplus s\oplus r$,$\text{hash}(E)$ 与 $F=C\oplus s=x_A\oplus r\oplus s$,$\text{hash}(F)$.

(4) 如果 $\text{hash}(E)=\text{hash}(F)$,输出 $P(\boldsymbol{A},\boldsymbol{B})=0$;否则,输出 $P(\boldsymbol{A},\boldsymbol{B})=1$.

正确性分析 根据哈希函数的特点,该协议会出现单边错误,即如果 $\boldsymbol{A}=\boldsymbol{B}$,$\text{hash}(E)$ 一定等于 $\text{hash}(F)$;但 $\text{hash}(E)=\text{hash}(F)$ 不一定 $\boldsymbol{A}=\boldsymbol{B}$,如果 $\text{hash}(E)=\text{hash}(F)$,但 $\boldsymbol{A}\neq\boldsymbol{B}$,表明协议出现了错误. 该协议给出错的概率非常小,设哈希函数为 128 位的二进制数,即 $|\text{hash}(\cdot)|=128$,那么 $\text{hash}(E)=\text{hash}(F)$,而 $\boldsymbol{A}\neq\boldsymbol{B}$ 概率

$$\Pr[\text{hash}(E)=\text{hash}(F)\wedge\boldsymbol{A}\neq\boldsymbol{B}]=2^{-128},$$

因此,可以说出现单边错误的情况是可以忽略的,即在概率意义下,$\boldsymbol{A}=\boldsymbol{B}\Leftrightarrow\text{hash}(E)=\text{hash}(F)$.

定理 2 保密矩阵相等判定协议 2 是安全的.

证明方法与定理 1 的证明类似,采用构造模拟器的方法.

3.3 保密向量相等判定协议

假设 Alice 有一个向量 $\boldsymbol{X}=(x_1,\cdots,x_m)$,Bob 有一个向量 $\boldsymbol{Y}=(y_1,\cdots,y_m)$. 他们想保密地判断这两个向量是否相等,而不泄露向量的任何其他信息,这就是向量相等的安全多方计算问题. 解决该问题的思想是:Alice 和 Bob 首先用协议 2 的方法,把 $\boldsymbol{X},\boldsymbol{Y}$ 转换为分量位数相同的 $\boldsymbol{X}'=(x_1',\cdots,x_m')$ 和 $\boldsymbol{Y}'=(y_1',\cdots,y_m')$. 把 $\boldsymbol{X}',\boldsymbol{Y}'$ 用字母表和连接运算表示为 $S=\text{Concat}_{10}^m(x_1',\cdots,x_m')$,$T=\text{Concat}_{10}^m(y_1',\cdots,y_m')$. 若 S,T 的比特数大于 64,分别用 hash( )计算并比较 $\text{hash}(S)$ 和 $\text{hash}(T)$ 是否相等;否则,调用协议 1. 如果 $\text{hash}(S)=\text{hash}(T)$,那么 $\boldsymbol{X}=\boldsymbol{Y}$;否则,$\boldsymbol{X}\neq\boldsymbol{Y}$. 为叙述简单定义二元谓词如下:

$$P(\boldsymbol{X},\boldsymbol{Y})=\begin{cases}0,\text{如果 }\boldsymbol{X},\boldsymbol{Y}\text{ 相等;}\\1,\text{如果 }\boldsymbol{X},\boldsymbol{Y}\text{ 不相等.}\end{cases}$$

协议 3 保密的向量相等判定协议

输入:Alice 输入向量 $\boldsymbol{X}=(x_1,\cdots,x_m)$,Bob 输入向量 $Y=(y_1,\cdots,y_m)$.

输出:$P(\boldsymbol{X},\boldsymbol{Y})$.

(1) Alice 和 Bob 分别将向量 $\boldsymbol{X},\boldsymbol{Y}$ 通过在高位数的位置补零的方法使向量的每个分量的十进制位数与最大的分量位数相同,得到 $\boldsymbol{X}'=(x_1',\cdots,x_m')$ 和 $\boldsymbol{Y}'=(y_1',\cdots,y_m')$.

(2) Alice 和 Bob 分别用字母表及连接运算表示 $\boldsymbol{X}'$ 和 $\boldsymbol{Y}'$,得到 $S=\text{Concat}_{10}^m(x_1',\cdots,x_m')$ 和 $T=\text{Concat}_{10}^m(y'_1,\cdots,y'_m)$.

(3) Alice 和 Bob 商定一个 hash( ),若 S,T 的比特数大于 64,直接计算 $\text{hash}(S)$,$\text{hash}(T)$ 并比较;否则,调用协议 1. 如果 $\text{hash}(S)=\text{hash}(T)$,输出 $P(\boldsymbol{X},\boldsymbol{Y})=0$;否则,输出 $P(\boldsymbol{X},\boldsymbol{Y})=1$.

定理 3 保密向量相等判定协议 3 是安全的.

证明方法与定理 1 的证明类似,采用构造模拟器的方法.

3.4 保密图形同构判定协议

图形同构问题描述如下:Alice 有一个图形 S,Bob 有一个图形 T,双方想判断 S 和 T 是否同构,但不泄露双方的私有信息. 本文研究一类特殊图形的同构问题:图形的所有顶点构成一个多边形,多边形的内部没有顶点.

解决问题的思想是:两个图形同构的条件是两个图形的邻接矩阵相等,这就要先确定两个图形的顶点的对应关系,按照以下方法确定两个图形顶点的对应关系:Alice 按照自己图形的逆时针方向给图形的每条边确定一个方向,这样的图形就成为一个有向图,有向图的每条边都是一个向量. 沿着逆时针方向给最长的向量(边)的起点编号为 1,终点编号为 2,然后按逆时

针给所有顶点依次编号,并计算相应的邻接矩阵$(\boldsymbol{S}_{ij})_{m\times n}$. 同样, Bob 对自己的图按逆时针方向和顺时针方向各做一次,得到两个邻接矩阵$(\boldsymbol{T}_{ij})_{m\times n}$,$(\boldsymbol{T}_{ij})'_{m\times n}$. 如果两个图形同构,那么$((\boldsymbol{S}_{ij})_{m\times n}=(\boldsymbol{T}_{ij})_{m\times n})\vee((\boldsymbol{S}_{ij})_{m\times n}=(\boldsymbol{T}_{ij})'_{m\times n})$(如果两图形镜面对称$(\boldsymbol{S}_{ij})_{m\times n}=(\boldsymbol{T}_{ij})'_{m\times n}$);否则,$((\boldsymbol{S}_{ij})_{m\times n}\neq(\boldsymbol{T}_{ij})_{m\times n})\wedge((\boldsymbol{S}_{ij})_{m\times n}\neq(\boldsymbol{T}_{ij})'_{m\times n})$.

为判断上述条件是否满足,把不易判断的图形同构问题转换成判断邻接矩阵相等问题,因为图形邻接矩阵都是用0,1表示的,若邻接矩阵元素的比特数大于64,直接用选择的 hash( )就可以保密判断图形是否同构;在图形的顶点数比较小时,调用协议1,保密比较两组哈希值的一组相等,即得到这两个邻接矩阵相等. 为叙述简单定义二元谓词如下:

$$P(S,T)=\begin{cases}0,\text{如果 } S,T \text{ 同构};\\1,\text{如果 } S,T \text{ 不同构}.\end{cases}$$

协议4　保密的图形同构判定协议

输入:Alice 输入图 S 的邻接矩阵$(\boldsymbol{S}_{ij})_{m\times n}$,Bob 输入图 T 的邻接矩阵$(\boldsymbol{T}_{ij})_{m\times n}$和$(T_{ij})'_{m\times n}$.

输出:$P(S,T)$.

(1) Alice 和 Bob 分别使用协议2的方法将矩阵$(\boldsymbol{S}_{ij})_{m\times n}$和$(\boldsymbol{T}_{ij})_{m\times n}$,$(\boldsymbol{T}_{ij})'_{m\times n}$按式(3)的方式排列得到$x_{S_{ij}}=s_{11}\cdots s_{mn}$,$x_{T_{ij}}=t_{11}\cdots t_{mn}$,$x'_{T_{ij}}=t'_{11}\cdots t'_{mn}$.

(2) Alice 和 Bob 商定一个 hash( ),若$x_{S_{ij}}$,$x_{T_{ij}}$,$x'_{T_{ij}}$的比特数大于64,计算并比较$(\text{hash}(x_{S_{ij}})=\text{hash}(x_{T_{ij}}))\vee(\text{hash}(x_{S_{ij}})=\text{hash}(x'_{T_{ij}}))$是否成立,否则进行下一步.

(3) Alice 和 Bob 调用协议1比较 hash$(x_{S_{ij}})$,hash$(x_{T_{ij}})$和hash$(x_{S_{ij}})$,hash$(x'_{T_{ij}})$是否相等,如果$(\text{hash}(x_{S_{ij}})=\text{hash}(x_{T_{ij}}))\vee(\text{hash}(x_{S_{ij}})=\text{hash}(x'_{T_{ij}}))$,输出 $P(S,T)=0$;否则,输出 $P(S,T)=1$.

定理4　保密的图形同构判定协议4是安全的.

证明方法与定理1的证明类似,采用构造模拟器的方法.

3.5　保密图形相似判定协议

图形相似问题描述如下:Alice 有一个图形 G,Bob 有一个图形 H,Alice 和 Bob 想在不泄露任何信息的同时判断这两个图形是否相似. 方案的基本思想是:首先利用协议3的方法判断两个图形的对应内角相等和对应边成比例;其次利用协议4判断两个几何图形同构. 下面给出该问题的详细描述和解决方案.

Alice 按照协议4的方法给图形编号,计算相应的边向量为$\boldsymbol{X}=(x_1,\cdots,x_m)$,角向量为$\boldsymbol{A}=(\alpha_1,\cdots,\alpha_n)$,邻接矩阵$(\boldsymbol{G}_{ij})_{m\times n}$. Bob 对自己的图做同样的事情两次,一次逆时针方向,一次顺时针方向,分别得到相应的两组边向量为$\boldsymbol{Y}=(y_1,\cdots,y_m)$,$\boldsymbol{Y}_1=(a_1,\cdots,a_m)$,两组角向量为$\boldsymbol{B}=(\beta_1,\cdots,\beta_n)$,$\boldsymbol{B}_1=(b_1,\cdots,b_n)$,两个邻接矩阵为$(\boldsymbol{H}_{ij})_{m\times n}$,$(\boldsymbol{H}_{ij})'_{m\times n}$. 如果两个图形相似,那么

$$((\frac{\boldsymbol{X}}{\boldsymbol{Y}}=C)\vee(\frac{\boldsymbol{X}}{\boldsymbol{Y}_1}=C))\wedge((\boldsymbol{A}=\boldsymbol{B})\vee(\boldsymbol{A}=\boldsymbol{B}_1))$$
$$\wedge(((\boldsymbol{G}_{ij})_{m\times n}=(\boldsymbol{H}_{ij})_{m\times n})\vee((\boldsymbol{G}_{ij})_{m\times n}=(\boldsymbol{H}_{ij})'_{m\times n}))$$

(C 是常数)(如果两图形镜面对称

$$(\frac{\boldsymbol{X}}{\boldsymbol{Y}_1}=C)\wedge(\boldsymbol{A}=\boldsymbol{B}_1)\wedge((\boldsymbol{G}_{ij})_{m\times n}=(\boldsymbol{H}_{ij})'_{m\times n}));$$

否则,

$$((\frac{\boldsymbol{X}}{\boldsymbol{Y}}\neq C)\wedge(\frac{\boldsymbol{X}}{\boldsymbol{Y}_1}\neq C))\vee((\boldsymbol{A}\neq\boldsymbol{B})\wedge(\boldsymbol{A}\neq\boldsymbol{B}_1))$$
$$\vee((\boldsymbol{G}_{ij})_{m\times n}\neq(\boldsymbol{H}_{ij})_{m\times n})\wedge((\boldsymbol{G}_{ij})_{m\times n}\neq(\boldsymbol{H}_{ij})'_{m\times n}).$$

Alice(Bob)将角向量$\boldsymbol{A}(\boldsymbol{B},\boldsymbol{B}_1)$保留两位小数并扩大100倍得到$\boldsymbol{A}'=(\alpha'_1,\cdots,\alpha'_n)$($\boldsymbol{B}'=(\beta'_1,\cdots,\beta'_n)$,$B'_1=(b'_1,\cdots,b'_n)$);Alice(Bob)选择自己的最小边$x_i(y_i,a_j(y_i=a_j))$,计算$(\frac{x_1}{x_i},\cdots,\frac{x_m}{x_i})$($(\frac{y_1}{y_i},\cdots,\frac{y_m}{y_i})$,$(\frac{a_1}{a_j},\cdots,\frac{a_m}{a_j})$)(商定比值保留两位小数),分别将这三个向量都扩大100倍得到向量

$\boldsymbol{X}'=(x'_1,\cdots,x'_m)$,$Y'=(y'_1,\cdots,y'_m)$,$Y'_1=(a'_1,\cdots,a'_m)$,

Alice(Bob)分别将$\boldsymbol{A}'$,$\boldsymbol{X}'$($\boldsymbol{B}'$,$\boldsymbol{Y}'$和$\boldsymbol{B}'_1$,$\boldsymbol{Y}'_1$)连接得到向量

$\boldsymbol{W}=(\alpha'_1,\cdots,\alpha'_n,x'_1,\cdots,x'_m)$($\boldsymbol{Z}=(a'_1,\cdots,\beta'_n,y'_1,\cdots,y'_m)$,$\boldsymbol{Q}=(b'_1,\cdots,b'_n,a'_1,\cdots,a'_m)$),

字母表连接$\boldsymbol{W}(\boldsymbol{Z},\boldsymbol{Q})$的元素得到

$$S=\text{Concat}_{10}^{m+n}(\alpha'_1,\cdots,\alpha'_n,x'_1,\cdots,x'_m)$$
$$(T=\text{Concat}_{10}^{m+n}(\beta'_1,\cdots,\beta'_n,y'_1,\cdots,y'_m),$$
$$T'=\text{Concat}_{10}^{m+n}(b'_1,\cdots,b'_n,a'_1,\cdots,a'_m)).$$

如果S,T,T'的比特数大于64,用协商好的 hash( ),计算并比较$(\text{hash}(S)=\text{hash}(T))\vee(\text{hash}(S)=\text{hash}(T'))$是否成立;否则,调用协议1. 如果$(\text{hash}(S)=\text{hash}(T))\vee(\text{hash}(S)=\text{hash}(T'))$,则两个图形的对应角相等对应边成比例. 之后利用协议4判断图形是否同构,如果同构,那么两个图形相似;否则,两个图形不相似. 为叙述简单定义二元谓词如下:

$$P(G,H)=\begin{cases}0,\text{如果 } G,H \text{ 相似};\\1,\text{如果 } G,H \text{ 不相似}.\end{cases}$$

协议5　保密的图形相似判定协议

输入:Alice 输入图 G 的边向量为$\boldsymbol{X}=(x_1,\cdots,x_m)$,角向量为$\boldsymbol{A}=(\alpha_1,\cdots,\alpha_n)$和图的邻接矩阵$(\boldsymbol{G}_{ij})_{m\times n}$;Bob 输入图 H 的边向量为$\boldsymbol{Y}=(y_1,\cdots,y_m)$、$\boldsymbol{Y}_1=(a_1,\cdots,a_m)$,角向量为$\boldsymbol{B}=(\beta_1,\cdots,\beta_n)$、$\boldsymbol{B}_1=(b_1,\cdots,b_n)$和图的邻接矩阵$(\boldsymbol{H}_{ij})_{m\times n}$、$(\boldsymbol{H}_{ij})'_{m\times n}$.

输出:$P(G,H)$.

(1) Alice(Bob)将$\boldsymbol{A}(\boldsymbol{B},\boldsymbol{B}_1)$保留两位小数,扩大100倍得到$\boldsymbol{A}'=$

$(\alpha_1',\cdots,\alpha_n')$($\boldsymbol{B}'=(\beta_1',\cdots,\beta_n')$、$\boldsymbol{B}_1'=(b_1',\cdots,b_n')$).

(2) Alice(Bob)选择边 $\boldsymbol{X}(\boldsymbol{Y},\boldsymbol{Y}_1)$ 中的最小边 x_i,进行比值运算保留两位小数,将这两个向量都扩大 100 倍得 $\boldsymbol{X}'=(x_1',\cdots,x_m')$,($\boldsymbol{Y}'=(y_1',\cdots,y_m')$,$\boldsymbol{Y}_1'=(a_1',\cdots,a_m')$).

(3) Alice 将 $\boldsymbol{A}'$、$\boldsymbol{X}'$ 连接得到向量 $\boldsymbol{W}=(\alpha_1',\cdots,\alpha_n',x_1',\cdots,x_m')$,Bob 分别将 $\boldsymbol{B}'$、$\boldsymbol{Y}'$ 和 $\boldsymbol{B}_1'$、$\boldsymbol{Y}_1'$ 连接得到向量 $\boldsymbol{Z}=(\beta_1',\cdots,\beta_n',y_1',\cdots,y_m')$、$\boldsymbol{Q}=(b_1',\cdots,b_n',a_1',\cdots,a_m')$.

(4) Alice 和 Bob 分别计算 $S=\mathrm{Concat}_{10}^{m+n}(\alpha_1',\cdots,\alpha_n',x_1',\cdots,x_m')$ 与 $T=\mathrm{Concat}_{10}^{m+n}(\beta_1',\cdots,\beta_n',y_1',\cdots,y_m')$,$T'=\mathrm{Concat}_{10}^{m+n}(b_1',\cdots,b_n',a_1',\cdots,a_m')$,如果 S,T,T' 的比特数大于 64,用协商好的 hash(),计算并比较($\mathrm{hash}(S)=\mathrm{hash}(T)$)∨($\mathrm{hash}(S)=\mathrm{hash}(T')$)是否成立;否则,调用协议 1. 如果($\mathrm{hash}(S)\neq\mathrm{hash}(T)$)∧($\mathrm{hash}(S)\neq\mathrm{hash}(T')$),输出 $P(G,H)=1$;否则,继续执行下一步.

(5) Alice 和 Bob 调用协议 4 判断 G,H 的邻接矩阵是否相等,如果相等,输出 $P(G,H)=0$;否则,输出 $P(G,H)=1$.

定理 5 保密的图形相似判定协议 5 是安全的.

证明方法与定理 1 的证明类似,采用构造模拟器的方法.

3.6 实例验证

为了更好地理解协议 5,下面给出一个简单的实例. Alice 有一个 5 边形 S,边向量为 $\boldsymbol{X}=(x_1,\cdots,x_5)$,角向量为 $\boldsymbol{A}=(\alpha_1,\cdots,\alpha_5)$和图的邻接矩阵 $\boldsymbol{S}_{5\times5}$,Bob 有一个 5 边形 T,边向量为 $\boldsymbol{Y}=(y_1,\cdots,y_5)$,角向量为 $\boldsymbol{B}=(\beta_1,\cdots,\beta_5)$和图的邻接矩阵 $\boldsymbol{T}_{5\times5}$. 如图 1 所示,Alice 首先用协议 3 保密地判断两个向量是否相等的方法判断图形的外部多边形是否相似,其次用协议 4 的方法判断图形的邻接矩阵相等,Alice 写出自己拥有图的邻接矩阵为:

$$\boldsymbol{S}_{5\times5}=\begin{bmatrix}0&1&1&1&1\\1&0&1&0&0\\1&1&0&1&1\\1&0&1&0&1\\1&0&1&1&0\end{bmatrix},$$

Bob 用同样的方法写出邻接矩阵. 之后用哈希函数的方法比较这两个矩阵是否相等,如果这两个矩阵相等也就是两个图形同构. 如果符合以上条件,则这两个图形相似.

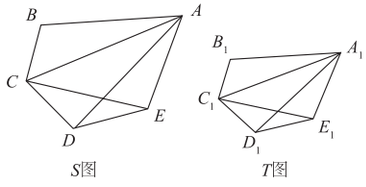

图1 S图与T图

4 性能分析

4.1 效率分析

计算复杂性分析 本文提出的协议通过转换问题来降低计算复杂性,这是本文的主要工作. 因为本文方案以哈希运算为基础,所以主要通过比较哈希运算的次数来衡量计算复杂性,规定执行一次哈希运算的运算量是 h. 本文协议 1 执行 2 次哈希运算,计算复杂度是 $2h$;协议 2 执行 2 次哈希运算,计算复杂度是 $2h$;协议 3 执行 2 次哈希运算,计算复杂度是 $2h$;协议 4 执行 2 次哈希运算,计算复杂度是 $3h$;协议 5 执行 6 次哈希运算,计算复杂度是 $6h$.

通信复杂性分析 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,在安全多方计算研究中通常用轮数. 本论文中协议 1 的通信复杂度是 2 轮;协议 2 通信复杂度是 2 轮;协议 3 的通信复杂度是 2 轮;协议 4 的通信复杂度是 2 轮;协议 5 的通信复杂度是 4 轮,所以本论文协议的计算复杂性和通信复杂度都比较低. 各方案计算复杂性和通信复杂性比较见表 1.

表 1 各方案计算复杂性和通信复杂性的比较

	协议 1	协议 2	协议 3	协议 4	协议 5
计算复杂性	$2h$	$2h$	$2h$	$3h$	$6h$
通讯复杂性	2	2	2	2	4

4.2 实验测试与分析

通过两个模拟实验来测试执行协议 5 的效率. 算法采用 C++实现,并在一台 Intel(R) Pentium(R) Dual CPU E2200 的 PC 机上运行. 当两个图形的顶点数不等时,程序运行结果会显示不同的哈希值,本实验只考虑相同顶点数图形的情况.

实验方法 (1)假定两个图形是相似的,分别设定图形的顶点 n 为 10 到 200 间隔为 10 的值时,随机选取两个相似(不相邻顶点没有连线)图形 S 和 T,规定一个逆时针方向,Alice 分别以 S 图形最大边的起点开始逆时针,Bob 以 T 图形最大边的起点开始逆时针排序和顺时针排序,对顶点 n 的每个设定值进行 10 次模拟实验测试,统计协议执行时间的平均值. 图 2 描述了图形外部多边形相似的执行时间随顶点数增长的变化规律.

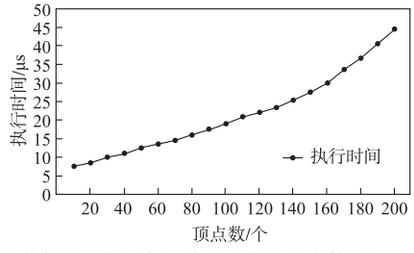

图2 图形外部多边形相似的执行时间随顶点数增长的变化规律

(2)在实验(1)的基础上,任意连接图形 S 和 T 中不相邻的若干对点,用协议 4 的方法根据不相邻点之间的连线得到一个邻接矩阵,同样设定图形的顶点 n 为 10 到 200 间隔为 10 的值时,对顶点 n 的每个设定值进行 10 次

模拟实验测试,统计协议执行时间的平均值. 图 3 描述了图形同构的执行时间随顶点数增长的变化规律.

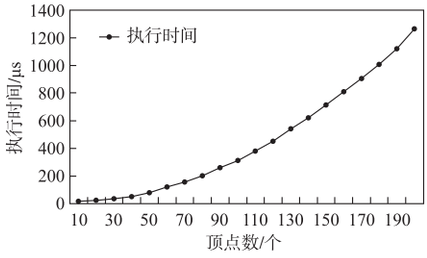

图3　图形同构的执行时间随顶点数增长的变化规律

从实验 1 和实验 2 的结果可知两个图形相似时,测出协议执行时间随顶点数的增长大致呈线性增长. 综上所述,协议 5 的计算复杂度比较低.

5　结论

关于图形相似问题的安全多方计算协议的构造一直是密码学领域中的一个重要问题. 本文基于哈希函数算法首先提出了保密数相等判定协议,在此基础上提出了保密矩阵相等判定协议和保密向量相等判定协议,最后提出了保密图形同构判定协议和保密图形相似判定协议. 本文研究的这些问题都是基于半诚实模型的,对于安全多方计算的研究与应用有重要的理论意义.

参考文献

[1] Yao A. Protocols for secure computations[A]. E Kushilevitz. Proceedings of the 23th IEEE Symposium on Foundations of Computer Science[C]. Chicago:IEEE Press,1982. 160 – 164.

[2] Goldreich O,et al. How to play any mental game[A]. Alfred V. Aho. Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing[C]. New York : ACM,1987. 218 – 229.

[3] Goldreich O. Foundations of Cryptography:Basic Applications[M]. London:Cambridge University Press,2004. 599 – 764.

[4] Yasin S,et al. Cryptography based e-commerce security:a review[J]. International Journal of Computer Science Issues,2012,9(2):132 – 137.

[5] Sharma R. Review paper on cryptography[J]. International Journal of Research,2015,2(5):141 – 142.

[6] Kumar S N. Review on network security and cryptography [J]. Science and Education,2015,3(1):1 – 11.

[7] Du W L, Atallah J. Secure multi-party computation problems and their applications:A review and open problems [A]. B Timmerman. New Security Paradigms Workshop 2001[C]. New York,USA:ACM,2001. 11 – 20.

[8] Atallah M J,Du W. Secure Multi-party Computational Geometry[M]. Berlin:Springer Berlin Heidelberg,2001. 165 – 179.

[9] Li S D,DAI Y Q. Secure two-party computational geometry[J]. Journal of Computer Science and Technology, 2005,20(2):258 – 263.

[10] Li S D,WU C Y,et al. Secure multiparty computation of solid geometric problems and their applications[J]. Information Sciences,2014,282:401 – 413.

[11] Li S D,Dai Y Q,et al. A secure multi-party computation solution to intersection problems of sets and rectangles [J]. Progress in Natural Science,2006,16(5):538 – 545.

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

[13] 王涛春,罗永龙,等. 多边形相似判定中的私有信息保护[J]. 小型微型计算机系统,2012,33(2):383 – 387.
Wang T C,Luo Y L,et al. Privacy-preserving in the determination of polygonal similarity[J]. Journal of Chinese Computer Systems,2012,33(2):383 – 387. (in Chinese)

[14] 祝晓晖. 基于几何图形相似仿真系统的设计与实现[D]. 四川:电子科技大学,2012.
Zhu X H. The Design and Implementation of Simulation System Based on Geometry Similarity[D]. Sichuan:University of Electronic Science and Technology of China,2012.

[15] Ionescu A,Leiss E L. On the role of complementation in implicit language equations and relations[J]. Journal of Computer and System Sciences,2014,80(2):457 – 467.

[16] Li S D,Wang D S,Dai Y Q. Efficient secure multiparty computational geometry[J]. Chinese Journal of Electronics,2010,19(2):324 – 328.

作者简介

李顺东　男,1963 年生于河南平顶山. 现为陕西师范大学计算机科学学院教授、博士生导师. 主要研究方向为密码学与信息安全.
E-mail:shundong@ snnu. edu. cn

杨晓莉　女,1989 年生于陕西商洛. 现为陕西师范大学计算机科学学院硕士研究生. 主要研究方向为密码学与信息安全.
E-mail:xiaoliyang@ snnu. edu. cn