论 文

排序问题的安全多方计算协议

唐春明^{①②*}, 石桂花^①, 姚正安^③^① 广州大学数学与信息科学学院, 广州 510006^② 中国科学院软件研究所信息安全国家重点实验室, 北京 100080^③ 中山大学数学与计算科学学院, 广州 510275

* 通信作者. E-mail: ctang@gzhu.edu.cn

收稿日期: 2010-03-12; 接受日期: 2010-10-19

国家自然科学基金 (批准号: 10871222) 资助项目

摘要 为某些特殊问题构造有效的安全多方计算协议是多方计算研究领域的一个重要分支. 文中使用密钥共享方案为排序问题构造了一个有效的安全多方计算协议, 该协议在 n 个参与者中至多只有 t 个不诚实者, 若 $t \leq (n-1)/2$, 则它在被动攻击下是完全安全的; 若 $t < n/3$ 时, 它在主动攻击下是完全安全的. 排序问题的最简单形式就是“百万富翁”问题.

关键词 安全多方计算 排序问题 密钥分享

1 引言

假设两个人各自拥有一个数据, 他们希望在没有泄露自己数据任何信息的前提下双方能共同知道谁拥有的数据大, 即双方都不知道对方数据的任何信息但是能知道谁的数据大. Yao^[1] 首先将该类问题描述为“百万富翁”问题, 并提出了一个解决方法, 从而开辟了安全多方计算这一研究领域. 自从 Yao 提出解决“百万富翁”问题的协议以后, 许多类似的协议被提出^[2–10], 特别在文献 [10] 中, 提出了利用计算几何设计的能够判断任意两个正实数之间的大小或相等关系的协议, 而且该协议只需要常数轮.

Yao 在文献 [1] 中为“百万富翁”问题构造了一个有效的协议, 但缺乏安全性证明. Lindell 和 Pinkas 在文献 [11] 中最早给出了 Yao 协议的安全性证明, 但他们的协议 (包括 Yao 的协议) 都是假设参与者是半诚实的 (即协议参与者按照预定的协议步骤执行). 还有一种参与者, 也许不按照协议的预定步骤执行, 它可能篡改数据, 更可能停止协议的执行, 这种参与者在安全多方计算领域中被称为恶意的参与者. 如果参与者是恶意的, 那么在文献 [1, 11] 中的协议就无法安全实现. 为了保证在有恶意参与者也能安全实现解决“百万富翁”问题的协议, Goldreich 等^[12] 使用零知识证明协议实现了它, 并且, 还使用零知识证明为任意概率函数 (有 $n \geq 2$ 个输入) 构造了安全多方计算协议. 正是他们的这些里程碑工作, 给安全多方计算提供了更广阔的应用空间, 从而吸引了更多的研究者来从事这方面的研究.

然而, Goldreich 等^[12] 提出的协议使用了零知识证明协议, 因此只有在特殊单向函数存在等密码学假设下才能实现, 也就是说, 所有参与者的计算能力是有限的, 因此得到的协议只能是计算安全的, 即如果某参与者具有无限的计算能力, 那么协议不再安全. 另外, Goldreich 等人提出的协议的实用性

引用格式: 唐春明, 石桂花, 姚正安. 排序问题的安全多方计算协议. 中国科学: 信息科学, 2011, 41: 789–797

非常小, 因为它们是基于通讯复杂性很高的零知识证明协议、茫然传输协议、以及位承诺协议. 除此之外, 恶意参与者的数量 t 不能是任意的, Goldreich 的协议在恶意参与者的情形下是安全多方计算协议的前提是 $t < n/2$.

为了去掉单向函数等密码学假设和参与者的计算能力有限的限制, Ben-Or 等^[13]以及 Chaum 等^[14]几乎同时提出了没有任何密码学假设的完全安全的多方计算协议, 也就是说, 即使参与者计算能力无界, 该协议仍然是安全的. 他们的协议适合于任意的概率函数, 但要保证协议是安全的, 若参与者是半诚实的, 那么半诚实的参与者数量 $t < n/2$, 若参与者是恶意的, 那么恶意参与者的数量 $t < n/3$. 这些完全安全的多方计算协议, 只能是门限形式的, 而对于非门限形式的完全安全的多方计算协议被 Cramer 等人^[15]基于扩张规划 (span program) 构造出来, 这些协议具有更好的实用性, 主要是他们的密钥共享方案是非门限的, 但缺点也有, 即不是理想的密钥共享方案.

目前, 所有的安全多方计算协议都是以文献 [1, 11–15] 为基础的, 只不过是基于的工具不一样, 或考虑的模型不一样, 或要求协议所具有的性质不一样. 而且, 所有的这些协议差不多都只是理论上能实现, 而要把它付诸于实践, 还有很多困难需要解决. 另外, 也不可能存在某个安全多方计算的模型能适合所有的问题. 因此为不同的问题构造有效的安全多方计算协议便成为多方计算领域的一个重要研究内容.

“百万富翁”问题 (或秘密数据比较问题) 一个很自然的推广就是在 $n \geq 2$ 个数中寻找一个最大 (或最小) 的数, 这样的推广显然是很有现实和理论意义的. 理论意义在于, 若能解决推广的问题, 则百万富翁问题也能解决. 现实意义就更多, 比如在电子投标中, 每个投标者 (投标者总数 > 2) 都不想泄露自己的投标价格, 然而主办方需要知道投标价格最大的投标者, 该问题的数学模型就是推广的秘密数据比较. 显然, 已有的解决数据秘密比较的安全多方计算协议都不能简单解决推广的秘密数据比较问题, 除非连续使用 $n - 1$ 次双方秘密数据比较协议.

我们可以把推广的秘密数据比较问题进行更一般的描述, 即描述成数组的排序问题. 假设一个 n 元数组 $(s_1, s_2, \dots, s_n)$, 在不泄露任何数据的情形下, 我们需要对该数组根据某种性质进行排序, 得到新的数组 $(s'_1, s'_2, \dots, s'_n)$. 如果我们把“性质”定义为“数据的大小”, 即根据数据的大小进行排序, 则我们的排序问题能够解决推广的秘密数据比较问题.

本文为排序问题构造一个完全安全的多方计算协议, 通过运行该协议, 我们能得到 n 个秘密数据的正确排序. 此协议具有如下的性质: 该协议在 n 个参与者中若至多只有 t 个不诚实者, 其中 $t \leq (n - 1)/2$, 则它在被动攻击下是完全安全的, 而 $t < n/3$ 时, 它在主动攻击下是完全安全的.

文中将使用秘密共享方案作为主要工具, 特别当攻击者为被动攻击者时, 只需要使用 Shamir 的秘密共享方案^[16], 然而, 当攻击者是主动攻击者时, 需要使用基于纠错码理论的可验证秘密共享方案.

2 基本知识

2.1 安全多方计算的基本概念

设 $P_1, P_2, \dots, P_n$ 是 n 个参与者, 他们通过一个密码学协议 (即多方计算) 共同计算某个给定的函数 $f(s_1, s_2, \dots, s_n) = (y_1, y_2, \dots, y_n)$, 其中, 函数 f 是一个概率函数^[13], $s_1, \dots, s_n$ 分别为参与者 $P_1, P_2, \dots, P_n$ 的秘密输入, 在协议结束后, $P_1, P_2, \dots, P_n$ 分别得到 $y_1, y_2, \dots, y_n$. 通常说一个多方计算协议是安全的, 如果它具有正确性和秘密性. 我们说协议是正确的, 如果即使在某些参与者有欺骗行为的情况下, 仍然保证诚实参与者计算结果的正确性, 即计算结束后每个诚实的参与者 P_i 都能得到正

确的输出 y_i . 我们说协议具有秘密性, 如果每个参与者 P_i 除了知道 (s_i, y_i) 以及从中推导出的信息之外, 得不到任何其他信息.

多方计算的安全性要求是部分参与者有欺骗行为的情况下是满足的, 事实上, 这也正是安全多方计算理论的核心. 显然, 有欺骗行为的成员数目以及他们的欺骗能力对欺骗成功与否, 即对安全性的满足有密切联系. 对于有欺骗行为的成员数目与协议设计的方法、使用的工具、以及安全性的程度有关. 在这我们需要描述欺骗能力, 即攻击者的能力. 在安全多方计算中攻击者的能力通常分为两种, 即被动的 (passive) 和主动的 (active). 被动的攻击者只是能得到被收买者 (多方计算中的参与方) 的全部信息, 但被收买者仍然忠实地执行协议 (即被收买者是半诚实的参与者); 主动攻击者则会完全控制收买者, 这意味着被收买者将不会忠实地执行协议 (即被收买者是恶意的参与者).

2.2 排序问题

假设参与者 $P_1, P_2, \dots, P_n$ 各自拥有秘密 $s_1, s_2, \dots, s_n$, 其中 $s_i \in \mathbb{Q}^+, i = 1, \dots, n$. 他们希望能比较 $s_1, s_2, \dots, s_n$, 其结果是 $P_1, P_2, \dots, P_n$ 知道 $s_1, s_2, \dots, s_n$ 的大小顺序关系, 但除了知道大小顺序和各自的秘密 s_i 外, 不知道其他任何的信息.

2.3 秘密共享方案

下面具体介绍一个秘密共享方案^[13]. 这个秘密共享方案是 n 个 Shamir 门限秘密共享方案的组合. 秘密共享方案一般分为两个阶段, 分发阶段和重构阶段. 假设这个秘密共享协议中有 n 个参与者, $n > t$.

初始化: 每个参与者 p_i 分别拥有一个秘密 s_i 和一个公开身份 $x_i, i = 1, 2, \dots, n$.

分发阶段: 对于 $i = 1, 2, \dots, n$, 每个 P_i 分别随机选择有理数域上的一个 t 次多项式 $f_i(x) = s_i + a_{i1}x + a_{i2}x^2 + \dots + a_{it}x^i$, 并把它发给 $f_i(x_j)$. 设 $g(x) = \sum_{i=1}^n f_i(x)$, 每个参与者 P_j 都可以计算 $g(x_j) = \sum_{i=1}^n f_i(x_j)$, 其中 $j = 1, 2, \dots, n$.

重构阶段: 假设 $t+1$ 个任意参与者, 不妨就假设为 $P_1, \dots, P_{t+1}$, 因为他们能够分别计算 $g(x_1), g(x_2), \dots, g(x_{t+1})$, 所以他们可以利用 Lagrange 插值公式计算:

$$\sum_{i=1}^{t+1} \left(\prod_{j=1, j \neq i}^{t+1} \frac{-x_j}{x_i - x_j} \right) g(x_i) = g(0) = s_1 + s_2 + \dots + s_n.$$

在以上协议中, 任意的 $t+1$ 个参与者能重构出 $s_1 + s_2 + \dots + s_n$, 然而任何 $t' (< t+1)$ 个参与者没有 $s_1 + s_2 + \dots + s_n$ 的任何信息^[12].

3 排序问题的安全多方计算协议及安全性分析

假设有 n 参与者 $P_1, P_2, \dots, P_n$, 他们分别拥有秘密数据 $s_1, s_2, \dots, s_n$. 在本章, 我们将为排序问题设计一个安全多方计算协议. 通过该协议, 参与者 $P_1, P_2, \dots, P_n$ 能够知道秘密数据 $s_1, s_2, \dots, s_n$ 的大小顺序, 然而, 对于每个参与者 P_i , 除了知道 s_i 和 $s_1, s_2, \dots, s_n$ 的大小顺序外, 没有其他 $s_j (i \neq j)$ 的任何信息.

我们使用的基本思想是对于每一个 $i = 1, 2, \dots, n$, 令 $s'_i = r_1 s_i + r_2 s_i^2 + \dots + r_{2t+1} s_i^{2t+1}$, 其中 $s_1, s_2, \dots, s_n$ 都是大于或等于 0 的有理数, $r_1, r_2, \dots, r_{2t+1}$ 也都是大于或等于 0 的有理数, 那么有理数 $s_1, s_2, \dots, s_n$ 和 $s'_1, s'_2, \dots, s'_n$ 有相同的大小顺序.

因此, 我们将设计一个协议, 即通过 $s_1, s_2, \dots, s_n$ 计算出 $s'_1, s'_2, \dots, s'_n$, 从 $s'_1, s'_2, \dots, s'_n$ 的大小顺序得到 $s_1, s_2, \dots, s_n$ 的大小顺序. 但是, 我们的协议必须保证从 $s'_1, s'_2, \dots, s'_n$ 中的任意 t 个值 $s'_{i_1}, \dots, s'_{i_t}$ 得不到 $s_j \in \{s_1, \dots, s_n\} \setminus \{s_{i_1}, \dots, s_{i_t}\}$ 的任何信息.

3.1 排序问题的安全多方计算协议

初始化阶段: 假设协议有 n 个参与者, 分别为 $P_1, P_2, \dots, P_n$, 每个参与者 P_i 拥有一个公开的身份 x_i , 以及一个秘密数据 $s_i \in \mathbb{Q}^+, i = 1, 2, \dots, n$. 任意两个参与者之间存在一个秘密通道, 每个参与者有一个广播渠道.

输入阶段: 对于 $i = 1, 2, \dots, n$, 每个 P_i 随机独立选择 $2t + 1$ 个 t 次多项式分别在有理数域上共享 $s_i, s_i^2, \dots, s_i^{2t}, s_i^{2t+1}$,

$$\begin{aligned} f_{i_1} &= s_i + a_{i_1 1}x + a_{i_1 2}x^2 + \dots + a_{i_1 t}x^t, \\ f_{i_2} &= s_i^2 + a_{i_2 1}x + a_{i_2 2}x^2 + \dots + a_{i_2 t}x^t, \\ &\vdots \\ f_{i_{2t+1}} &= s_i^{2t+1} + a_{i_{2t+1} 1}x + a_{i_{2t+1} 2}x^2 + \dots + a_{i_{2t+1} t}x^t. \end{aligned}$$

计算 $f_{i_1}(x_j), f_{i_2}(x_j), \dots, f_{i_{2t+1}}(x_j)$, 并把 $f_{i_1}(x_j), f_{i_2}(x_j), \dots, f_{i_{2t+1}}(x_j)$ 发给 P_j . 另外 P_i 生成一随机数 $r_i \in \mathbb{Q}^+$, 并随机选择一个有理数域上的 t 次多项式 $r_i(x)$, $r_i(x) = r_i + a_{i1}x + a_{i2}x^2 + \dots + a_{it}x^t$. 计算 $r_i(x_j)$, 并把 $r_i(x_j)$ 发送给 P_j , 其中 $j = 1, 2, \dots, n$.

计算阶段:

1) 所有参与者约定在 $r_1(x), r_2(x), \dots, r_n(x)$ 取定 $2t + 1$ 个随机的多项式, 不妨设这 $2t + 1$ 个多项式为 $r_1(x), r_2(x), \dots, r_{2t+1}(x)$, 其对应的参与者分别是 $P_1, P_2, \dots, P_{2t+1}$. 每个参与者 P_j 计算 $\lambda_j g_i(x_j) (1 \leq i \leq n, 1 \leq j \leq 2t + 1)$, 其中

$$g_i(x_j) = r_1(x_j)f_{i_1}(x_j) + r_2(x_j)f_{i_2}(x_j) + \dots + r_{2t+1}(x_j)f_{i_{2t+1}}(x_j). \quad (3.1)$$

$$A = \begin{pmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{2t} \\ 1 & x_2 & x_2^2 & \dots & x_2^{2t} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_{2t+1} & x_{2t+1}^2 & \dots & x_{2t+1}^{2t} \end{pmatrix}, \quad A^{-1} = \begin{pmatrix} \lambda_1 & \lambda_2 & \lambda_3 & \dots & \lambda_{2t+1} \\ * & * & * & \dots & * \\ \dots & \dots & \dots & \dots & \dots \\ * & * & * & \dots & * \end{pmatrix}.$$

注意: 若约定的 $2t + 1$ 个参与者 $P_1, P_2, \dots, P_{2t+1}$ 不一样, 则 $\lambda_1, \lambda_2, \dots, \lambda_{2t+1}$ 也不一样, 主要是 A 发生了变化.

2) 对于 $P_1, P_2, \dots, P_{2t+1}$ 中的每个 P_j 随机选择 n 个 t 次多项式 $h_{j1}(x), h_{j2}(x), \dots, h_{jn}(x)$ 分别用来共享 $\lambda_j g_1(x_j), \lambda_j g_2(x_j), \dots, \lambda_j g_n(x_j)$, 其中 $h_{ji}(0) = \lambda_j g_i(x_j), 1 \leq i \leq n$. 然后 P_j 计算 $h_{j1}(x_k), h_{j2}(x_k), \dots, h_{jn}(x_k)$ 并将它们作为秘密发送给 $P_k, 1 \leq k \leq n$.

输出阶段: 对于每个 P_j , 计算并输出 $y_j = (y_{j1}, \dots, y_{jn}) = (\sum_{i=1}^{2t+1} h_{i1}(x_j), \dots, \sum_{i=1}^{2t+1} h_{in}(x_j))$, 其中 $j = 1, 2, \dots, n$.

排序的实现: 假设每个 P_j 具有输出 $y_j (1 \leq j \leq n)$. 使用 Lagrange 插值公式, 任意 $t+1$ 个参与者都可以恢复 $s'_i = r_1 s_i + r_2 s_i^2 + \cdots + r_{2t+1} s_i^{2t+1}$, $1 \leq i \leq n$, 同时广播 (P_i, s'_i) , $1 \leq i \leq n$. 因此任何人都可以根据 (P_i, s'_i) , $1 \leq i \leq n$, 得出 n 个参与者的秘密数据的排序.

3.2 安全性分析

假设多项式

$$g_i(x) = r_1(x)f_{i1}(x) + r_2(x)f_{i2}(x) + \cdots + r_{2t+1}(x)f_{i2t+1}(x), \quad (3.2)$$

则由 $r_j(x)$ 和 $f_{ij}(x)$, $1 \leq i \leq n, 1 \leq j \leq 2t+1$ 都是 t 次的多项式, 我们有

$$g_i(x) = s'_i + b_{i1}x + b_{i2}x^2 + \cdots + b_{i2t}x^{2t}. \quad (3.3)$$

引理 1 $s'_i = r_1 s_i + r_2 s_i^2 + \cdots + r_{2t+1} s_i^{2t+1}$.

证明 显然

$$\begin{aligned} s'_i &= g_i(0) = r_1(0)f_{i1}(0) + r_2(0)f_{i2}(0) + \cdots + r_{2t+1}(0)f_{i2t+1}(0) \\ &= r_1 s_i + r_2 s_i^2 + \cdots + r_{2t+1} s_i^{2t+1}. \end{aligned}$$

定理 1 任意 $t+1$ 个参与者能计算 $s'_i, i = 1, 2, \dots, n$.

证明 令 $X_i = (s'_i, b_{i1}, \dots, b_{i2t})^T$. 让 (3.3) 式分别在 $x_1, x_2, \dots, x_{2t+1}$ 取值, 这 $2t+1$ 个表达式可以记为

$$AX_i = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^{2t} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{2t} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{2t+1} & x_{2t+1}^2 & \cdots & x_{2t+1}^{2t} \end{pmatrix} \begin{pmatrix} s'_i \\ b_{i1} \\ \vdots \\ b_{i2t} \end{pmatrix} = \begin{pmatrix} g_i(x_1) \\ g_i(x_2) \\ \vdots \\ g_i(x_{2t+1}) \end{pmatrix}, \quad (3.4)$$

所以

$$\begin{pmatrix} s'_i \\ b_{i1} \\ \vdots \\ b_{i2t} \end{pmatrix} = A^{-1} \begin{pmatrix} g_i(x_1) \\ g_i(x_2) \\ \vdots \\ g_i(x_{2t+1}) \end{pmatrix} = \begin{pmatrix} \lambda_1 & \lambda_2 & \lambda_3 & \cdots & \lambda_{2t+1} \\ * & * & * & \cdots & * \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ * & * & * & \cdots & * \end{pmatrix} \begin{pmatrix} g_i(x_1) \\ g_i(x_2) \\ \vdots \\ g_i(x_{2t+1}) \end{pmatrix},$$

即

$$s'_i = \lambda_1 g_i(x_1) + \lambda_2 g_i(x_2) + \cdots + \lambda_{2t+1} g_i(x_{2t+1}). \quad (3.5)$$

令 $h_i(x) = h_{1i}(x) + h_{2i}(x) + \cdots + h_{2t+1i}(x)$, 其中 $h_{1i}(x), h_{2i}(x), \dots, h_{2t+1i}(x)$ 是协议计算阶段第 2 步中 $P_1, P_2, \dots, P_{2t+1}$ 分别选择的随机多项式.

则 $h_i(0) = h_{1i}(0) + h_{2i}(0) + \cdots + h_{2t+1i}(0) = \lambda_1 g_i(x_1) + \lambda_2 g_i(x_2) + \cdots + \lambda_{2t+1} g_i(x_{2t+1}) = s'_i$, 而 $h_i(x)$ 是一个 $t+1$ 次的多项式, 根据 Lagrange 插值, 任何 $t+1$ 个不同的点都能计算出 $h_i(0)$, 即 $s'_i, i = 1, 2, \dots, n$.

定理 2 若不诚实的参与者个数 t 与参与者总数 n 满足 $t \leq (n-1)/2$, 则对于 n 个参与者的排序问题存在一个在被动攻击下完全安全的多方计算协议.

证明 显然, 为了保证在 3.1 小节所构造的协议的合理性, 根据计算阶段的第一步必须有 $2t+1 \leq n$, 而根据式 (3.2) 必须有 $2t < n$, 因此 $t \leq (n-1)/2$.

为了证明在 3.1 小节所构造的协议是一个安全多方计算协议, 我们只需要证明该协议满足正确性和秘密性.

1) 正确性. 因为我们假设的攻击者是被动攻击者, 所以所有的参与者 (包括所有攻击者和诚实者) 都是遵循协议的步骤. 因此, 我们只需要验证此协议是否能给秘密数据数组 $s_1, s_2, \dots, s_n$ 进行排序.

由引理 1, 我们可以知道 $s'_i = r_1 s_i + r_2 s_i^2 + \dots + r_{2t+1} s_i^{2t+1}$, 其中 $1 \leq i \leq n$. 而由定理 1, 我们知道任意 $t+1$ 参与者都能计算所有的 $s'_i, i = 1, 2, \dots, n$. 显然, $s'_1, s'_2, \dots, s'_n$ 的排序就是 $s_1, s_2, \dots, s_n$ 的正确排序. 也就是说, 我们能够得到 $s_1, s_2, \dots, s_n$ 的正确排序, 即正确性满足.

2) 秘密性. 根据文献 [13] 中多方计算协议秘密性的定义, 如果协议满足下面的 (a) 和 (b), 则协议的秘密性成立: (a) 每个 P_i 观察到的信息串与等长的随机串具有相同的概率分布, 即 P_i 观察到的信息串与等长的随机串不可区分, (b) 任意 $t \leq (n-1)/2$ 个参与者联合也不能得到任意其他参与者输入的任何信息, 即任意 $P_{i_1}, \dots, P_{i_t}$ 使用它们的 $s_{i_1}, \dots, s_{i_t}$ 和所有的输出, 也不可能得到 $s_i \notin \{s_{i_1}, \dots, s_{i_t}\}$ 的任何信息. 假设 P_i 是被动攻击者之一, 则它观察到的数据为:

输入阶段中 P_i 生成的数据串 $f_{i1}(x_j), f_{i2}(x_j), \dots, f_{i_{2t+1}}(x_j), r_i(x_j), 1 \leq j \leq n$, 其他参与者发送给它的串 $f_{j1}(x_i), f_{j2}(x_i), \dots, f_{j_{2t+1}}(x_i), 1 \leq j \leq n$.

计算阶段中如果 P_i 是参与计算的 $2t+1$ 个参与者之一, 则 P_i 生成的数据串为 $h_{i1}(x_j), h_{i2}(x_j), \dots, h_{in}(x_j), 1 \leq j \leq n$, 其他参与计算的 $2t+1$ 个参与者 (包括它自己) 发送给它的串为 $h_{j1}(x_i), h_{j2}(x_i), \dots, h_{jn}(x_i), 1 \leq j \leq 2t+1$. 如果 P_i 不是参与计算的 $2t+1$ 个参与者之一, 则它没有生成任何数据串, 仅仅得到 $2t+1$ 个参与者发送给它的串为 $h_{j1}(x_i), h_{j2}(x_i), \dots, h_{jn}(x_i), 1 \leq j \leq 2t+1$.

输出阶段中 P_i 所观察到的数据 $y_i, 1 \leq i \leq n$.

所有的 $2t+1$ 个参与计算的参与者选择的多项式是随机的, 所以它们生成的所有串, 即随机多项式上的点, 也是随机的. 因此, P_i 观察到的所有数据串与相同长的随机串具有同样的概率, (a) 满足.

如果 Shamir 的密钥共享方案中密钥 s 共享使用的多项式是 t 阶多项式, 则任意 t 个秘密分块是独立的^[16], 也就是说, 对于任意 t 个随机串与任意的 s 都能唯一决定一个 t 阶多项式. 因为被动攻击者的总数最多是 t 个, 也就是说, 被动攻击者拥有在 3.1 小节的协议中每个随机多项式上的至多 t 个点, 因此对于任意被共享的 s , 被动攻击者都不能得到任意的信息. 另外, 假设我们的所有 $s'_1, s'_2, \dots, s'_n$ 已经被构造出. 由 $s'_i = r_1 s_i + r_2 s_i^2 + \dots + r_{2t+1} s_i^{2t+1}, 1 \leq i \leq n$, t 个被动参与者能得到 t 个方程组, 这 t 个方程组中含有 $t+1$ 个未知数, 因此无法推出其他参与者的这 $t+1$ 个随机数, 从而也无法推出其他参与者的秘密值, 即 t 个被动参与者使用它们的 $s_{i_1}, \dots, s_{i_t}$ 和所有的输出, 也不可能得到 $s_i \notin \{s_{i_1}, \dots, s_{i_t}\}$ 的任何信息. 这样, (b) 成立.

因此, 秘密性成立, 而且, 即使参与者计算能力无界, 秘密性依然成立^[13].

3.3 主动攻击情形下的排序问题的安全多方计算

在 3.1 小节中构造的协议中, 只有至多有 $t \leq (n-1)/2$ 参与者是被动参与者的情形下, 协议才是安全的, 否则, 参与者也许通过数据的替换或者终止协议的执行来进行欺骗. 为了防止类似的欺骗, 即防止不诚实参与者的主动攻击, 我们不得不使用可验证的秘密共享方案和纠错码的理论^[13].

显然, 在 3.1 小节中的安全多方计算协议主要是为了计算 $s'_i = r_1 s_i + r_2 s_i^2 + \dots + r_{2t+1} s_i^{2t+1}$, 其中 $1 \leq i \leq n$. 而这 n 个函数都是多项式可计算的函数, 根据文献 [13] 中的定理 3, 只要使用可验证秘密

共享方案和纠错码原理, 我们能为任意的多项式可计算函数构造一个在主动攻击下的安全多方计算协议. 该类协议有一个限制条件, 即主动攻击者控制的参与者数 t 必须满足 $t < n/3$. 因此, 我们有如下的结论成立.

定理 3 若不诚实的参与者个数 t 与参与者总数 n 满足 $t < n/3$, 则对于 n 个参与者的排序问题存在一个在主动攻击下完全安全的多方计算协议.

3.4 补充

在我们的排序协议中, 因为 $1 \leq t \leq (n-1)/2$, 所以 $n > 2$.

强调的是, 在本文中我们考虑秘密值和随机数都是任意正有理数, 所有的计算都在有理数域中进行. 若在整数或者有限域内取值和计算, 则本文中的协议用穷举法可被成功攻击, 任意 t 各不诚实参与者可计算出其他参与者的秘密值.

事实上, 如果秘密数 $s_1, s_2, \dots, s_n$ 和随机数 $r_1, r_2, \dots, r_{2t+1}$ 都是正实数, 所有运行都在实数域中进行, 则使用我们的安全多方计算协议仍然能够实现秘密的排序.

4 排序问题的安全多方计算协议的一个实例

我们在这里例举 4 个秘密数据排序的例子. 由初始化阶段知 $t = 1$, 即至多只有一个参与者不诚实. 4 个参与者分别记为 P_1, P_2, P_3, P_4 , 他们拥有的秘密数据分别为 5, 15, 14, 8.

输入阶段: P_1 随机独立选择 4 个 1 阶多项式: $f_{11}(x) = 5 + 2x, f_{12}(x) = 5^2 + 12x, f_{13}(x) = 5^3 + 10x, r_1(x) = 4 + 7x$; P_2 随机独立选择 4 个 1 阶多项式: $f_{21}(x) = 15 + x, f_{22}(x) = 15^2 + 7x, f_{23}(x) = 15^3 + 6x, r_2(x) = 6 + 9x$; P_3 随机独立选择 4 个一阶多项式: $f_{31}(x) = 14 + 8x, f_{32}(x) = 14^2 + 11x, f_{33}(x) = 14^3 + 4x, r_3(x) = 7 + 8x$; P_4 随机独立选择 4 个一阶多项式: $f_{41}(x) = 8 + 14x, f_{42}(x) = 8^2 + 20x, f_{43}(x) = 8^3 + 12x, r_4(x) = 5 + 3x$; 表 1 为 4 个参与者分发分享值的表格.

计算阶段: 4 个参与者共同选定 P_1, P_2, P_4 , 那么

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 4 & 16 \end{pmatrix}, \quad A^{-1} = \begin{pmatrix} 8/3 & -2 & 1/3 \\ * & * & * \\ * & * & * \end{pmatrix}.$$

P_1 计算并选择 4 个 1 阶多项式 $h_{11}(x), h_{12}(x), h_{13}(x), h_{14}(x)$ 分别共享:

$$\begin{aligned} \frac{8}{3}g_1(1) &= \frac{8}{3}(11 \times 7 + 15 \times 37 + 8 \times 135) = \frac{8}{3} \times 1712, \\ \frac{8}{3}g_2(1) &= \frac{8}{3}(11 \times 16 + 15 \times 232 + 8 \times 3381) = \frac{8}{3} \times 30704, \\ \frac{8}{3}g_3(1) &= \frac{8}{3}(11 \times 22 + 15 \times 207 + 8 \times 2748) = \frac{8}{3} \times 25331, \\ \frac{8}{3}g_4(1) &= \frac{8}{3}(11 \times 22 + 15 \times 84 + 8 \times 524) = \frac{8}{3} \times 5694; \end{aligned}$$

同样 P_2 计算并选择 4 个 1 阶多项式 $h_{21}(x), h_{22}(x), h_{23}(x), h_{24}(x)$ 分别共享:

$$-2g_1(2) = -2 \times 2933, -2g_2(2) = -2 \times 43299,$$

表 1 4 个参与者分发分享值

Table 1 All shares owned by four participants respectively in this example

	P_1	P_2	P_3	P_4
P_1	...	(2,9),(2,49), (2,145),(2,18)	(3,11),(3,61), (3,155),(3,25)	(4,13),(4,73), (4,165),(4,32)
P_2	(1,16),(1,232), (1,3381),(1,15)	...	(3,18),(3,246), (3,3393),(3,33)	(4,19),(4,253), (4,3399),(4,42)
P_3	(1,22),(1,207), (1,2748),(1,15)	(2,30),(2,218), (2,2752),(2,23)	...	(4,46),(4,240), (4,2760),(4,39)
P_4	(1,22),(1,84), (1,524),(1,8)	(2,36),(2,104), (2,536),(2,11)	(3,50),(3,124), (2,536),(2,11)	...

$$-2g_3(2) = -2 \times 36044, -2g_4(2) = -2 \times 9040.$$

P_4 计算并选择 4 个 1 阶多项式 $h_{41}(x), h_{42}(x), h_{43}(x), h_{44}(x)$ 分别共享:

$$\begin{aligned}\frac{1}{3}g_1(4) &= \frac{1}{3} \times 6287, \frac{1}{3}g_2(4) = \frac{1}{3} \times 69017, \\ \frac{1}{3}g_3(4) &= \frac{1}{3} \times 58472, \frac{1}{3}g_4(4) = \frac{1}{3} \times 17616.\end{aligned}$$

输出阶段:

$$y_1 = (y_{11}(1), y_{12}(1), y_{13}(1), y_{14}(1)), y_2 = (y_{21}(2), y_{22}(2), y_{23}(2), y_{24}(2)),$$

$$y_3 = (y_{31}(3), y_{32}(3), y_{33}(3), y_{34}(3)), y_4 = (y_{41}(4), y_{42}(4), y_{43}(4), y_{44}(4)),$$

其中, $y_{ij}(i) = h_{ij}(i) + h_{2j}(i) + h_{4j}(i), 1 \leq i \leq 4, 1 \leq j \leq 4$.

排序的实现: 4 个参与者中任意 2 个参与者可以计算出

$$\begin{aligned}\frac{8}{3}g_1(1) - 2g_1(2) + \frac{1}{3}g_1(4) &= 795, \frac{8}{3}g_2(1) - 2g_2(2) + \frac{1}{3}g_2(4) = 18285, \\ \frac{8}{3}g_3(1) - 2g_3(2) + \frac{1}{3}g_3(4) &= 14952, \frac{8}{3}g_4(1) - 2g_4(2) + \frac{1}{3}g_4(4) = 2976.\end{aligned}$$

参与者广播 $(P_1, 795), (P_2, 18285), (P_3, 14952), (P_4, 2976)$. 任何人都可以对其进行排序, 显然 $(P_2, 18285) > (P_3, 14952) > (P_4, 2976) > (P_1, 795)$. 对照前面的秘密值 5, 15, 14, 8, 知道执行协议得到了正确的排序.

5 结论

关于排序问题的安全多方计算协议构造一直是密码学领域中一个难解的问题. 本文使用秘密共享方案为排序问题构造了一个有效的安全多方计算协议, 不仅在理论上还是在应用上都具有重要的价值.

参考文献

- 1 Yao A C. Protocols for secure computations. In: Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science, 1982. 160–164
- 2 Fagin R, Naor M, Winkler P. Comparing information without leaking it. *Commun ACM*, 1996, 39: 77–85
- 3 Schoenmakers B, Tuyls P. Practical two-party computation based on the conditional gate. In: Proceedings of Advances in Cryptology-ASIACRYPT'04 LNCS 3329. Berlin/Heidelberg: Springer-Verlag, 2004. 119–136
- 4 Qin J, Zhang Z F, Feng D G, et al. A protocol of comparing information without leaking. *J Softw*, 2004, 15: 421–427
- 5 Blake I F, Kolesnikov V. Strong condition oblivious transfer and computing on intervals. In: Proceedings of Advances in Cryptology-ASIACRYPT'04. LNCS 3329. Berlin/Heidelberg: Springer-Verlag, 2004. 515–529
- 6 Ioannidis I, Grama A. An efficient protocol for Yao's millionaires' problem. In: Proceedings of the 36th Hawaii International Conference on System Sciences, 2003
- 7 Fischlin M. A cost-effective pay-per-multiplication comparison method for millionaires. In: Progress on Cryptology-CT-RSA 2001: the Cryptographers' Track at RSA Conference 2001 San Francisco, 2001. 457–472
- 8 Lin H Y, Tzeng W G. An efficient solution to the millionaires problem based on homomorphic encryption. In: ASIACRYPT 2005. LNCS 3531. Berlin/Heidelberg: Springer-Verlag, 2005. 456–466
- 9 Cachin C. Efficient private bidding and auctions with an oblivious third party. In: Proceedings of the 6th ACM Conference on Computer and Communications Security, 1999. 120–127
- 10 Luo Y L. Key issues and application of secure multiparty computation. PhD Dissertation Hefei: Chinese University of Science and Technology, 2005
- 11 Lindell Y, Pinkas B. A proof of yao's protocol for secure two-party computation. *J Cryptol*, 2009, 22: 161–188
- 12 Goldreich O, Micali S, Wigderson A. How to play any mental game—a completeness theorem for protocols with honest majority. In: Proceedings of the 19th ACM Symposium on Theory of Computing, 1987. 218–229
- 13 Ben-Or M, Goldwasser S, Wigderson A. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In: Proceedings of the 20th ACM Symposium on Theory of Computing, 1988. 1–10
- 14 Chaum D, Crepeau C, Damgard I. Multi-party unconditionally secure protocols. In: Proceedings of the 20th ACM Symposium on Theory of Computing, 1988. 11–19
- 15 Cramer R, Damgard I, Maurer U. General secure multi-party computation from linear secret sharing scheme. In: Advances in Crypto-EUROCRYPT 2000. LNCS 1807. Berlin/Heidelberg: Springer-Verlag, 2000. 316–334
- 16 Shamir A. How to share a secret. *Commun ACM*, 1979 22: 612–613

Secure multi-party computation protocol for sequencing problem

TANG ChunMing^{1,2*}, SHI GuiHua¹ & YAO ZhengAn³¹ School of Mathematics and Information Science, Guangzhou University, Guangzhou 510006, China;² State Key Laboratory of Information Security, Institute of Software Science, Chinese Academy of Sciences, Beijing 100080, China;³ School of Mathematics and Computational Science, Sun Yat-Sen University, Guangzhou 510275, China

*E-mail: ctang@gzhu.edu.cn

Abstract In multi-party computation field, one important branch is to construct an efficient and secure multi-party computation for some special problems. In this paper, we make use of secret sharing scheme to construct an efficient and secure multi-party computation for sequencing problem. Our protocols are perfectly secure against passive adversary if it can corrupt at most $t \leq (n-1)/2$ participants, and is perfectly secure against active adversary if it can corrupt at most $t < n/3$ participants. The simplest sequencing problem is millionaire problem.

Keywords secure multi-party computation, sequencing problem, secret sharing