

# 安全计算协议

姚期智

加利福尼亚大学伯克利分校

翻译：李晓峰 (cy\_lxf@163.com)

译文来自于经典文献翻译项目 <https://gitee.com/uisu/InfSecClaT>

译者单位：北京联合大学智慧城市学院

2024 年 6 月 25 日

## 摘要

此文是对姚期智老师 Protocols for secure computations 文章的翻译。

## 1 引言

## 2 安全计算的统一视图

## 3 确定性计算

### 3.1 百万富翁问题的解决方案

在这个摘要中，我们将详细描述我们所拥有的三种解决方案中的一种。

为了明确起见，设 Alice 有  $i$  百万，Bob 有  $j$  百万，并且  $1 \leq i, j \leq 10$ ，我们需要一个协议来判断 “ $i$  是否小于  $j$ ”，并且最后我们只能得到这个信息（也就是说没有获得多余的有关  $i, j$  的信息），设  $M$  是所有  $N$  比特非负整数集合， $Q_N$  是所有  $M$  到  $M$  的 1-1 满射函数 (onto function) 集合， $E_a$  是 Alice 的公钥，从  $Q_N$  中随机选取的。<sup>1</sup>

协议处理过程如下：

---

<sup>1</sup>译者注： $E_a$  逆函数  $D_a$  只有 Alice 知道，对于 Bob 同样有  $E_b$  和  $D_b$ 。

1. Bob 取一个 N 比特随机数  $x$ , 计算  $k = E_a(x)$ 。
2. Bob 把  $k - j$  发给 A.<sup>2</sup>。
3. Alice 计算  $y_u = D_a(k - j + u), u = 1, 2, \dots, 10$ 。<sup>3</sup>
4. Alice 随机选一个  $N/2$  比特随机素数  $p$ , 计算  $z_u = y_u \pmod{p}, u = 1, 2, \dots, 10$ , 如果所有  $z_u$  在模  $p$  下至少相差 2, 则停止, 否则产生新的  $p$ , 直至条件成立, 也就是说直至  $|z_u - z_v| \geq 2, u, v \in \{1, 2, \dots, 10\}, u \neq v$ 。<sup>4</sup>
5. Alice 把  $p$  和 10 个数:  $z_1, z_2, \dots, z_i$  和  $z_{i+1} + 1, z_{i+2} + 1, \dots, z_{10} + 1$  发送给 Bob。以上数都是在模  $p$  的运算。
6. Bob 取 Alice 发来的第  $j$  个数  $w$ , 如  $w = x \pmod{p}$ , 那么  $i \geq j$ , 否则  $i < j$ 。<sup>5</sup>
7. Bob 告诉 Alice 比较结果。

该协议显然能够使 Alice 和 Bob 正确地决定谁是更富有的人。为了证明该协议符合他们无法获取对方财富的任何更多信息的要求, 我们将在第 3.2 节中给出一个精确的模型。在这里, 我们将非正式地论证为什么该要求能够得到满足。

首先, Alice 对 Bob 的财富  $j$  一无所知, 除了 Bob 告诉她的最终结果所隐含的  $j$  约束, 因为来自 Bob 的唯一其他信息是 Bob 知道  $k-j+1$  到  $k-j+10$  之间某个  $s$  的  $D_a(s)$  的值。由于函数  $E_a$  是随机的所有 10 种可能性都是等概率的。

Bob 知道什么? 他知道  $y_j$ (也就是  $x$ ) 因此也知道  $z_j$ 。然而, 他没有关于其他  $z_u$  值的信息, 并且通过查看 Alice 发送给他的数字, 他无法判断它们是  $z_u$  还是  $z_u + 1$ 。

这还没有结束争论, 因为 Alice 或 Bob 可能会试图通过更多的计算来计算对方的价值。例如, Bob 可能会尝试随机选择一个数字  $t$  并检查

---

<sup>2</sup>译者注: 原文中是  $k - j + 1$

<sup>3</sup>译者注: 这里计算的是  $y_1, \dots, y_j, \dots, y_{10}$ , 特别注意  $y_j = D_a(k - j + j) = D_a(k) = x$ 。而此时 Alice 并不知道这里就是 Bob 的值。

<sup>4</sup>译者注: 此处要求相差为 2, 是因为在下面的步骤中通过加一改变了原数值, 而这种改变应该是能够与原值进行区分的。

<sup>5</sup>译者注: 如果  $j$  处的数没有变化, 则说明  $j$  是在  $i$  前面或者就是  $i$ , 没有  $+1$ 。

$E_a(t) = k-j+9$  是否成立; 如果他成功了, 他就知道  $y_9$  的值是  $t$ , 并且知道  $z_9$  的值, 这样他就可以知道是否  $i \geq 9$ 。如果  $i \geq j$  是前一个结论的结果, 那么这将是 Bob 不应该发现的额外信息。因此, 我们还必须在正式定义中包括, 参与者不仅没有通过协议指定的交换获得信息, 而且他们也不能在合理的时间内执行计算以获得该信息。在 3.2 节给出的正式定义中, 我们将对此进行精确的定义。

人们可能已经注意到, 在这个过程中, 某些方面可能会通过偏离商定的协议而作弊。例如, Bob 可能在最后一步对爱丽丝撒谎, 告诉爱丽丝错误的结论。是否有一种设计协议的方法, 使得成功作弊的机会变得非常小, 而不暴露  $i$  和  $j$  的值? 我们将在 3.3 节展示这是可能的。(请注意, 这是一个比 Shamir 等人 [5] 在心理扑克协议中使用的可验证性要求更强的要求。)

针对百万富翁的问题, 我们有另外两种基于不同原则的解决方案。第一个方案假设 Alice 和 Bob 各自拥有一个私有的单向函数, 其中这些函数满足交换性, 即  $E_a E_b(x) = E_b E_a(x)$ 。另一个方案利用 Goldwasser 和 Micali[2] 发明的概率加密方法。

### 3.2 一般性问题的模型

Alice 有个秘密数  $i$ , Bob 有个秘密数  $j$ , 假设 Alice 有一个公共单向函数  $E_a$ , 其逆函数是  $D_a$ , 逆函数只有 Alice 知道, 对于 Bob 同样有函数  $E_b, D_b$ , 假设  $E_a, E_b$  相互独立并且是从  $Q_N$  中随机选取,  $Q_N$  是 N 比特整数的 1-1 满射函数集合, 下面我们精确地描述 Alice 和 Bob 如何通过协议  $\Lambda$  计算  $f(i, j)$ 。

Alice 和 Bob 交替给对方发送字符串。

Bob 每次发送完成, Alice 检查她所拥有的信息:

- 1、字符串序列  $\alpha_1, \alpha_2, \dots, \alpha_t$
- 2、这些字符串之间的关系, 比如  $E_b(\alpha_3) = \alpha_9, \alpha_8$  有奇数个 1.
- 3、根据 Alice 和 Bob 至此已经传输过的比特, 协议说明 Alice 如何计算隐私字符串  $\alpha_{t+1}, \alpha_{t+2}, \dots, \alpha_s$ , 此处每一个新的字符串  $\alpha_u, u \in \{t+1, \dots, s\}$  都是以前字符串的函数, 或者说新字符串都是这样的形式  $E_a(y), E_b(y)$  或  $D_a(y)$ , 此处  $y$  是 Alice 已经获得的字符串。A 随机选择使用哪个函数, 例如, Alice 投币决定使用  $E(4)$  或者计算  $\alpha_2 + 3\alpha_8$ 。
- 4、Alice 计算完后, 她将发一个字符串给 Bob, 选择发送哪个字符串也是随机的。

Bob 收到字符串后，他也按 Alice 的方法计算一些字符串，并且根据协议发送一个字符串。

Alice 和 Bob 达成一致，当收到一个特殊的字符时，协议执行结束，这时，协议有一条指令，就是每个参与者都秘密计算函数  $f$  的值，最后，在协议中，我们要求 Bob 和 Alice 计算 E 和 D 的数量受  $O(N^k)$  的限制，此处  $k$  是一个事先选择好的整数。

#### 隐私限制 (Privacy Constraint)

设  $\epsilon, \delta > 0$ ,  $f(i, j)$  函数值为 0 或 1，假定初始时所有  $(i, j)$  取值可能性都是一样的，并且假定 Bob 和 Alice 根据协议忠实第计算，最后 Alice 原则上可以根据她计算的函数值  $v$  和她拥有的字符串，计算  $j$  值的概率分布  $p_i(j)$ . 一个协议如果满足以下条件，我们就说此协议满足  $(\epsilon, \delta)$  隐私限制：

1.  $p_i(j) = \frac{1}{\|G_i\|}(1 + O(\epsilon)), j \in G_i$ , 此处  $G_i$  是使  $f(i, j) = v$  等式成立的所有  $j$  组成的集合，如果  $j \notin G_i$ , 则  $p_i(j) = 0$ .

2. 如果 Alice 之后尝试执行更多计算计算 E 和 D，但计算的次数不超过  $O(N^k)$  次，那么她会以至少  $1 - \delta$  的概率仍然得到  $j$  上的上述概率分布。

3. 对于 Bob 也有以上同样要求。

**Theorem 1** 对于任何  $\epsilon, \delta > 0$  和任何函数  $f$ , 存在一个用于计算  $f$  的协议满足  $(\epsilon, \delta)$  隐私限制。

### 3.3 增加的需求

#### 复杂性 (complexity)

文章中给出的百万富翁算法并不实用，因为决定  $i, j$  范围的  $n$  如果很大，那么传输的比特也会很多，因为传输的比特数与  $n$  是一个正比关系，那么一个有意思的问题就出现了：

对于满足  $(\epsilon, \delta)$  隐私限制的用于计算  $f$  的任一协议来说，所学传输的最小比特数是多少？

可以想象，在没有因私限制时，有一些函数很容易计算，但是当有额外的隐私限制时，就变得很不容易。幸运的是，我们可以证明事实并非如此。假设  $\Lambda$  是一个协议，当使用此协议时，Alice 和 Bob 之间传输的最大比特数记为  $T(\Lambda)$ .

**Theorem 2** 设  $1 > \epsilon, \delta > 0, f(i, j)$  是一个 0-1 函数，如果  $f$  可以被一个规模为  $C(f)$  的布尔电路计算，那么这里就有一个计算  $f$  的协议  $\Lambda$  满足  $(\epsilon, \delta)$

隐私限制，并且  $T(\Lambda) = O(C(f) \log \frac{1}{\epsilon\delta})$ .

事实上，如果  $f$  可以被一个图灵机在时间  $S$  内计算，那么这个协议可以被实现，以至于 Alice 和 Bob 都有图灵机算法来执行这个协议在  $O(S \log(\frac{1}{\epsilon\delta}))$ .

**相互怀疑的参与者 (Mutually-Suspecting Participants)**

### 3.4 应用

## 4 概率计算

## 5 $m$ 方情况的一般化描述

## 6 什么不能做