可证明安全的理性委托计算协议

田有亮[1,2]，李秋贤[1]，张铎[1,3]，王琳杰[1]

（1. 贵州大学计算机科学与技术学院，贵州 贵阳 550025；2. 贵州省公共大数据重点实验室，贵州 贵阳 550025；
3. 贵州大学数学与统计学院，贵州 贵阳 550025）

摘 要：针对理性委托计算中的安全性需求问题，提出了一种可证明安全的理性委托计算协议。首先，在委托计算中引入博弈理论并分析理性参与者的行为偏好，并且在博弈论框架下构建理性委托计算博弈模型；其次，根据博弈模型中的均衡需求及理性委托计算的安全需求，设计理性安全模型；再次，结合 Yao 的混淆电路可以随机化重用的优势及全同态加密技术，构造理性委托计算协议，且协议中参与者的策略组合可以达到纳什均衡状态；最后，根据理性安全模型证明了协议的安全性和输入输出的隐私性，且性能分析表明了协议的有效性。所提理性委托计算协议在满足传统安全性的同时，又考虑了参与者的行为偏好，更符合大数据环境下的委托计算模式。

关键词：理性委托计算；混淆电路；全同态加密；可证明安全

中图分类号：TP309

文献标识码：A

doi: 10.11959/j.issn.1000−436x.2019133

Provably secure rational delegation computation protocol

TIAN Youliang[1,2], LI Qiuxian[1], ZHANG Duo[1,3], WANG Linjie[1]

1. College of Computer Science and Technology, Guizhou University, Guiyang 550025, China
2. Guizhou Provincial Key Laboratory of Public Big Data, Guiyang 550025, China
3. College of Mathematics and Statistics, Guizhou University, Guiyang 550025, China

Abstract: A provably secure rational delegation computation scheme was proposed to solve the requirement of security issues in rational delegate computation. Firstly, game theory was introduced into delegation computation and according to rational participants behavior preferences analysis, a rational delegate computing game model was designed. Secondly, according to the equilibrium demand of game model and the security requirement of rational delegation computation, a rational security model was established. Thirdly, combining Yao's garbled circuit with its advantages of re-randomization, as well as full homomorphic encryption technology, the rational delegation computation protocol was constructed. And the combination of strategies in the protocol could reach the Nash equilibrium state. Finally, the security of the protocol and the privacy of the input and output were proved according to the rational security model, and the performance analysis shows the validity of the protocol. The proposed rational delegation computing protocol not only satisfies the traditional security, but also considers the behavioral preference of participants, which is more in line with the delegation-computing mode under the big data environment.

Key words: rational delegate computation, garble circuit, full homomorphic encryption, provable security

收稿日期：2018–10–09；**修回日期**：2019–02–15

基金项目：国家自然科学基金资助项目（No.61772008）；贵州省教育厅科技拔尖人才支持基金资助项目（No. [2016] 060）；贵州省科技重大专项计划基金资助项目（No.20183001）；贵州省科技计划基金资助项目（No.[2017]5788）；教育部−中国移动科研基金资助项目（No.MCM20170401）；贵州大学培育基金资助项目（No.[2017]5788）

Foundation Items: The National Natural Science Foundation of China (No. 61772008), Guizhou Provincial Department of Education Science and Technology Talents Support Project(No. [2016] 060), Science and Technology Major Support Program of Guizhou Province (No. 20183001), Guizhou Provincial Science and Technology Plan Project (No. [2017]5788), Ministry of Education-China Mobile Research Fund Project (No. MCM20170401), Guizhou University Cultivation Project (No. [2017]5788)

1 引言

委托计算[1]是指计算能力相对较弱或资源受限的委托方将函数 F 的计算任务委托给不信任的计算方，计算方将返回一个计算结果及计算结果的正确性证明给委托方。委托方通过执行验证协议来验证返回结果的正确性，并且委托方验证该证明的工作量比计算函数 F 的开销要小得多，否则将失去委托计算的意义。委托计算一直受到学者的广泛研究，主要有基于复杂性理论构造方案和基于密码技术构造方案。基于复杂性理论构造方案主要应用的工具是交互式证明系统[2]、PCP（probabilistic checking of proofs）定理[3]等，Chung 等[4]在随机语言模型下对非交互式委托计算进行研究，给出了有效的解决方法。基于密码技术构造方案主要应用的工具有全同态加密[5]、基于属性加密[6]、混淆电路[7]等，Gennaro 等[8]利用文献[7]的混淆电路构造了非交互式的委托计算方案，该方案有效地解决了基于计算理论方案的困难性问题。

理性委托计算属于理性密码学的研究范围，针对理性密码协议的研究领域，大多学者较多地关注利用博弈论方法来解决秘密共享、安全多方计算等问题，涉及理性委托计算的研究尚少。理性委托计算结合博弈论与委托计算的思想，协议中参与者都是理性的，而不是诚实的或是恶意的，且协议中通过效用函数来保证计算结果的正确性。传统的委托计算协议中，通常假设参与者要么是诚实的，要么是恶意的，但实际应用中，参与者大多是理性的，因此理性委托计算的研究成为当前的研究热点。Azar 等[9]根据适当的评分规则，提出了一种理性证明系统，该系统中参与者既不是诚实的，也不是恶意的，而是理性的；随后 Azar 等[10]又利用 Utility Gaps 的思想构造了一种超有效的理性证明系统；Guo 等[11]通过对理性证明系统的研究，解决了证明者计算能力受限的理性证明系统问题；Tian 等[12]从理性的角度分析了安全通信问题，并提出了贝叶斯理性秘密共享方案；随后 Chen 等[13]从复杂性理论的角度研究了当存在多个证明者时，理性证明系统的理性证明问题。

关于理性委托计算的安全性问题是研究者最为关心的，如何利用效用函数构建安全可靠的理性委托计算协议更是当前的研究需求。Kilian 等[14]提出了证明者使用 Merkle 树向验证者发送对整个证明的短承诺的有效论证，证明者可以交互式地打开验证者的请求。Micali's CS Proof[1]可以获得非交互式解决方案，该解决方案根据随机 oracle 应用承诺字符串来选择要打开的请求，消除涉及参数的交互。在最近的研究中，更多研究者较为关注非交互式协议，并且可以在标准模型中给予证明。

本文结合混淆电路和全同态加密技术提出了一种可证明安全的理性委托计算方案，该方案保证了所有理性参与者都得到最优的利益，保证委托计算输入和输出的隐私性。本文的具体工作如下。

1) 通过分析参与者的行为策略及参与者选择行为策略而得到的效用，设计了理性委托计算博弈模型。

2) 根据构建的委托计算博弈模型中纳什均衡需求，以及理性委托计算的安全需求，设计了可证明安全的理性委托计算安全模型。

3) 利用随机化混淆电路可重用的优点与全同态加密技术，保证了理性参与者结果的正确性及委托计算输入和输出隐私，从而构建了安全的理性委托计算协议。

4) 对协议的安全性与性能进行分析，证明了协议的安全性与输入输出隐私性，保证了所有参与者在协议中能获得利益的最大化即达到唯一纳什均衡。

2 基础知识

2.1 博弈论

定义 1 博弈。博弈表达的基本形式[15]由局中人集合 P、策略空间 S 和效用函数 u 这 3 个要素组成，即 $G=\{P,S,u\}$，其中，$P=\{P_1,\cdots,P_n\}$，$S=\{S_1,\cdots,S_n\}$，$u=\{u_1,\cdots,u_n\}$。效用函数 $u_i:S\to \mathrm{R}$（R 代表实数空间），表示第 i (i=1,2,⋯,n)个局中人在不同组合下所得的效益。

定义 2 纳什均衡。对于博弈 $G=\{P,S,u\}$，如果由每个博弈方的一个策略所组成的策略组合 $s^*=(s_1^*,\cdots,s_n^*)$ 中，任一博弈方 P_i 的策略 s_i^*，都是应对其他博弈方策略组合 $(s_1^*,\cdots,s_n^*)$ 的最佳策略，即对于所有的 $s_i^*\in S$，存在博弈 $u_i(s_i^*)\geqslant u_i(s_j^*,s_{-i}^*)$。对于任意 $s_{ij}\in S$，则称 $s^*=(s_1^*,s_2^*,\cdots,s_n^*)$ 为博弈 G 的一个纳什均衡。

2.2 全同态加密

全同态加密方案一般由以下 4 个阶段组成[16]。

预处理阶段$(\mathrm{SK}_E,\mathrm{PK}_E)\leftarrow\mathrm{Setup}_{\mathrm{FHE}}(1^\lambda)$：输入安全参数$\lambda$，密钥生成算法随机产生公钥私钥对$(\mathrm{SK},\mathrm{PK})$。

加密阶段$c\leftarrow\mathrm{Encrypt}_{\mathrm{FHE}}(\mathrm{PK}_E,m)$：输入公钥$\mathrm{PK}_E$和需要加密的消息$m$，利用加密算法输出一个对应的密文。

解密阶段$m\leftarrow\mathrm{Decrypt}_{\mathrm{FHE}}(\mathrm{SK}_E,c)$：输入私钥$\mathrm{SK}_E$和需要解密的密文$c$，利用解密算法输出一个对应的明文。

运算函数$c_f\leftarrow\mathrm{Eval}_{\mathrm{FHE}}(\mathrm{PK}_E,c_i,f)$：输入公钥$\mathrm{PK}_E$，加密的密文组$c_i$和需要求值的函数$f$，利用运算函数求解函数值。

其中，明文$m=(m_1,\cdots,m_n)$，密文$c=(c_1,\cdots,c_n)$，密文组$c_i=(c_{i1},\cdots,c_{in})$，函数值$c_f=f(c_i)$。

2.3　混淆电路

Yao 的混淆电路[17]协议允许参与双方在不对明文做任何加密的情况下，对明文进行保密计算，该协议一般应用于半诚实参与者之间，是确保双方计算的通用方法。当应用在委托计算方案中时，委托方首先将委托的任意函数F转换为布尔电路C，然后将转换电路的混淆形式$G(C)$和委托方需要计算的函数x的混淆形式$G(x)$一起发送给计算方，这样代表该布尔电路的每条输入输出线的随机数均被加密。然后借助于准备阶段生成对应电路C的混淆表进行查表运算，通过计算布尔电路的每个门得到整个电路的输出。最后，计算方将计算结果的混淆形式$G(F(x))$发送给对应的委托方，委托方根据混淆电路将计算结果转换为实际的输出y。

图 1 为混淆电路的结构，其中X和Y为输入线，Z为输出线。这 3 根导线分别对应有 2 个值：0 和 1，即输入线的输入值和输出线的输出值。例如，当输入值a=0 与b=1 被选中后，混淆电路的任务就是需要安全地计算$g(a,b)$（g表示表 1 中的输出线）的值。混淆电路表如表 1 所示。

由表 1 可知，每个输入输出线X、Y、Z，且每根导线制定 2 个随机值，对应于 0 和 1。由表 1 可知，需要使用混淆电路表将K_Z^0、K_Z^1、K_X^0、K_X^1、K_Y^0、K_Y^1联系起来，即在混淆电路表中，K_Z^0、K_Z^1、K_X^0、K_X^1、K_Y^0、K_Y^1作为加密密钥，在合适的密钥输入下对K_Z^0、K_Z^1进行加密，从而形成混淆电路。其中，当给定 2 个输入密钥K_X^a和K_Y^b时，混淆计算表只有一行是可以正确解密的，即$E_{K_X^a}\left(E_{K_Y^b}\left(K_Z^{g(a,b)}\right)\right)$，这样可以有效地保证输入信息的隐秘性。

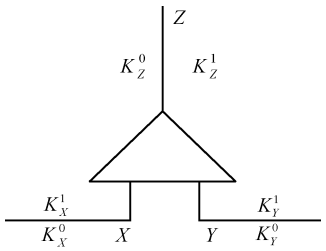

图 1　混淆电路

表 1　　混淆电路表

输入线 X	输入线 Y	输出线 Z	混淆电路
K_X^0	K_Y^0	K_Z^0	$E_{K_X^0}\left(E_{K_Y^0}\left(K_Z^0\right)\right)$
K_X^0	K_Y^1	K_Z^0	$E_{K_X^0}\left(E_{K_Y^1}\left(K_Z^0\right)\right)$
K_X^1	K_Y^0	K_Z^0	$E_{K_X^1}\left(E_{K_Y^0}\left(K_Z^0\right)\right)$
K_X^1	K_Y^1	K_Z^1	$E_{K_X^1}\left(E_{K_Y^1}\left(K_Z^1\right)\right)$

在委托计算方案中，首先，委托方为每根导线选择 2 个随机密钥，因此委托方共具有 6 个密钥；其次，委托方对表 1 的每一行进行加密，形成混淆电路表；接着，委托方将混淆电路表进行重新随机排列，这样密钥的位置就不会显示与其有关联的值，保证了密钥的安全性。混淆电路表形成后，委托方将混淆电路表发送给计算方，以及输入计算值x和输入密钥K_X^x，由于计算方收到混淆电路表后，使用自己的密钥来解密混淆电路表，所以，委托方必须将密钥发给计算方。在此过程中，如果计算方将 2 个计算任务K_Y^0、K_Y^1发送给委托方，那么计算方就可以解密更多信息，如果计算方要求委托方提供与其输入相对应的密钥，那么委托方就可以学习计算方的输入；为了避免信息的泄露，委托方与计算方使用不经意传输协议，使委托方获得与其在所有导线上的输入相关联的密钥，以保证输入输出的隐私性；最后，计算方计算电路，并将输出结果返回给委托方。

虽然混淆电路在委托计算方案中保护了客户端的输入输出隐私，但多次使用混淆电路进行计算时，恶意参与方可能将之前计算的标签输出作为本次的输出。结合全同态加密技术的混淆电路方法能够解决混淆电路的重用问题，委托方利用全同态加

密技术为每次委托计算的输入选择一个新的密钥，以保证门电路信息不被泄露，从而解决了委托计算中混淆电路的重用问题。

2.4 可重用混淆电路

为解决混淆电路的重用问题，理性委托计算协议采用了随机化混淆电路技术，该技术主要利用全同态加密同态性质。可重用混淆电路方案中，密钥串 $s\in\{0,1\}^l$，需要使用的公钥是基于素数阶群 q 的元素向量；明文串 $x\in\{0,1\}^n$，其中，$n=2l$，l 和 n 分别表示密钥串长度和明文串的长度，密文也是基于素数阶群 q 的多个元素向量。利用全同态加密的同态性质，通过群 Z_p 上的 2 个已知映射将 0-1 向量映射为同样长度的 0-1 向量。

将需要使用的混淆电路进行随机化处理，即在图 1 所示的布尔电路中，假设门电路 g 的第一根输入导线的 2 个标签为 A_0 和 A_1，第二根输入导线的 2 个标签为 B_0 和 B_1，输出导线的 2 个标签为 C_0 和 C_1。为了实现随机化混淆，将每根导线随机化选择比特置换。假设将门电路 g 的第一根输入导线的 2 个标签 A_0 和 A_1 进行比特置换，其比特置换为 θ 和 θ'，则输入导线新标签为 $\theta(A_0)$ 和 $\theta(A_1)$。根据全同态加密的密钥与明文的同态性质，随机选择 h、$h'\in(0,1)^l$，则密文 $E_{A_a}(h)$ 变换为 $E_{\theta(A_a)}(\theta'(h))$。继续选择随机数 $\beta\in\{0,1\}^l$，则形成的密文对为 $\left(E_{\theta(A_a)}(\beta\oplus h),E_{\theta(B_b)}(\beta\oplus h')\right)$。

由分析可知，通过为混淆电路的每根导线进行比特置换，实现重新随机化电路导线的标签和电路导线对应的密文对，从而实现随机化重复使用混淆电路的方法，且保证输入输出的隐私性。

2.5 理性委托计算

根据传统的委托计算定义，结合 Yao 的混淆电路与全同态加密技术，引出理性委托计算定义。假设理性委托方将计算函数 F 委托给理性计算方，理性计算方根据其博弈模型及效用返回计算结果。理性委托计算方案 RD 的形式化描述由以下 4 个算法构成。

1) $\text{KeyGen}(F,1^\lambda)\to(\text{PK},\text{SK})$：将计算函数 F 用电路 C 来表示。根据 Yao 的混淆电路为每条导线 w_i 随机选择 2 个值，w_i^0、$w_i^1\leftarrow\{0,1\}^\lambda$。对于每个门电路 g，计算其 4 个密文 $\left(\gamma_{00}^g,\gamma_{01}^g,\gamma_{10}^g,\gamma_{11}^g\right)$。其中，公钥 PK 为全部的密文集，即 $\text{PK}\leftarrow\cup_g(\gamma_{00}^g,\gamma_{01}^g,\gamma_{10}^g,\gamma_{11}^g)$；私钥 SK 是其选择的导线值，即 $\text{SK}\leftarrow\cup_i(w_i^0,w_i^1)$。

2) $\text{ProGen}_{\text{SK}}(x)\to\sigma_x$：运行全同态加密算法，产生一个新的密钥对，$(\text{SK}_E,\text{PK}_E)\leftarrow\text{Setup}_{\text{FHE}}(1^\lambda)$。令 $w_i\subset\text{SK}$ 表示为输入 x 的二进制线值，且公共值为 $\sigma_x\leftarrow\left(\text{PK}_E,\text{Encrypt}_E(\text{PK}_E,w_i)\right)$，私有值为 $\tau_x\leftarrow\text{SK}_E$。

3) $\text{Compute}_{\text{PK}}(\sigma_x)\to\sigma_y$：计算 Yao 的混淆电路协议中的解密算法 $\text{Decrypt}_E(\text{PK}_E,\gamma_i)$，以获得正确输出线的标签，令 σ_y 为输出线的标签。

4) $\text{Recover}_{\text{SK}}(\sigma_y)\to y\cup\perp$：使用公钥 SK 将输出线标签 σ_y 中的导线值映射到输出结果 y 的二进制表示形式上。如果映射失败，将输出 $\perp$，并令计算方接受相应的惩罚。

定义 3 正确性。如果问题生成算法生成的值使理性计算方输出正确的值，则理性委托计算协议 RD 是正确的，其形式化表示如下。

对于任意 $x\in\text{Domain}(F)$，如果 $\text{KeyGen}(F,1^\lambda)\to(\text{PK},\text{SK})$，$\text{ProGen}_{\text{SK}}(x)\to\sigma_x$ 和 $\text{Compute}_{\text{PK}}(\sigma_x)\to\sigma_y$ 都成立，且以不可忽略的概率使 $\text{Recover}_{\text{SK}}(\sigma_y)\to(y=F(x),1)$ 成立，则理性委托计算协议 RD 是正确的。

在协议中，若对于所有的概率多项式时间，敌手 $\mathcal{A}$ 不能使委托方接受一个不正确的输出，则理性委托计算协议 RD 是安全的。

定义 4 隐私性。理性委托计算协议 RD 的输入输出是隐私的，为理性委托计算协议 RD 定义敌手 $\mathcal{A}$，在协议中的优势为

$$\text{ADV}_{\text{RD}}^{\text{Priv}}(\text{RD},F,\lambda)=\Pr\left[\text{Exp}_{\mathcal{A}}^{\text{Priv}}[\text{RD},F,\lambda]=1\right]$$

在协议中，若对于任意的函数 F 和所有的概率多项式时间敌手 $\mathcal{A}$，概率 $\text{ADV}_{\text{RD}}^{\text{Priv}}(\text{RD},F,\lambda)-\frac{1}{2}$ 是可以忽略不计的，则理性委托计算协议 RD 是隐私的，其形式化分析如下

$$\begin{aligned}&\text{Experiment}\text{Exp}_{\mathcal{A}}^{\text{Priv}}[\text{RD},F,\lambda]\\&(\text{PK},\text{SK})\leftarrow\text{KeyGen}(1^\lambda,F)\\&(x_0,x_1)\leftarrow\mathcal{A}^{\text{ProbGen}}(\text{PK})\\&(\sigma_0,\tau_0)\leftarrow\text{ProbGen}(\text{PK},\text{SK},x_0)\\&(\sigma_1,\tau_1)\leftarrow\text{ProbGen}(\text{PK},\text{SK},x_1)\\&b\leftarrow\{0,1\}\\&b'\leftarrow\mathcal{A}^{\text{ProbGen}}(\text{PK},x_0,x_1,\sigma_b)\\&\text{if}\ \ b'=b\end{aligned}$$

else
　output 0
end if

即如果存在 2 个不相同的输入，产生的 2 个输出结果以可忽略的概率区分，则理性委托计算协议 RD 是隐私的。

3　博弈分析及其安全模型

3.1　博弈模型分析

理性委托计算是将博弈论与委托计算结合的新型的委托计算方案，通过引入理性参与者，使用效用函数来保证计算结果的正确性。一般来说，在委托计算方案中，存在 3 种类型的计算方：诚实的计算方，诚实的计算方会完全按照委托方的要求进行计算，并返回正确的结果；理性的计算方，理性计算方正确执行计算任务的效用必须大于执行其他任务的效用，如果在计算过程中懒惰计算的效用大于诚实计算，则理性计算方会选择懒惰计算；恶意的计算方，恶意计算方将试图破坏委托计算协议，并返回一个不正确的结果。实际上，由于协议中的参与者大多都是理性的，无论参与者是诚实或恶意的，在现实的协议中都是不合理的，因此，本文对理性的参与者进行分析。

假设存在理性委托方 P_1 和理性计算方 P_2，理性委托方有计算任务 F，其计算任务的本身价值为 Re。此时，理性计算方有“诚实”地返回正确结果和“恶意”地返回错误结果这 2 种策略，即理性计算方 P_2 的策略集为{诚实,恶意}。当理性计算方诚实地计算委托任务，其计算任务的成本为 $c(1)$，效用为 $u(1)$，返回正确答案后得到奖励为 r，且奖励大于计算成本，即 $r>c(1)$；若理性计算方存在恶意行为，则计算成本为 $c(q)$，效用为 $u(q)$，其中 q 为理性计算方作弊的概率，且 $u(1)>u(q)$。

由于参与者都是理性的，此时理性委托方将根据理性计算方返回的计算结果选择“奖励”诚实的理性计算方或者“惩罚”恶意的理性计算方，即理性委托方 P_1 的策略集为{惩罚, 不惩罚}。但当理性委托方未按照约定对理性计算方进行奖励，理性计算方可向可信第三方提出申诉，并对理性委托方进行罚款，记该罚款为 Q_1；同理，理性计算方存在恶意行为返回不正确的答案，理性委托方将对其进行惩罚，记该罚款为 Q_2。

基于对博弈模型的分析，可以得到理性参与者的效用矩阵。根据理性计算方的行为策略和理性委托方的行为策略，可以得到相应的效用矩阵，如表 2 所示。

表 2　理性委托计算参与者的效用矩阵

委托方	诚实的计算方	恶意的计算方
诚实的委托方	R_e-r , $r-c(1)$	$R_e-(rq-Q_2(1-q)-c(q))$, $rq-Q_2(1-q)-c(q)$
恶意的委托方	R_e-Q_1 , $r-c(1)+Q_1$	R_e-Q_1 , $rq-Q_2(1-q)-c(q)$

根据理性参与者的行为策略分析，理性委托计算可以分为 3 个阶段进行。

1) 理性委托方 P_1 对于计算任务 F，可以选择自己计算，或者选择委托给计算能力强大的服务器，即理性计算方 P_2 进行计算。

2) 理性计算方 P_2 对于计算任务 F，从策略集{诚实, 恶意}中选择一种策略进行反馈。

3) 理性委托方 P_1 根据理性计算方 P_2 反馈的结果，从策略集{惩罚, 不惩罚}中选择一种策略进行反馈。

将博弈论引入本协议中，利用子博弈精炼纳什均衡来分析理性委托计算。在每个阶段中，对应的参与方都有行为策略进行对应，例如，理性计算方 P_2“恶意”地返回错误结果，则理性委托方 P_1 会选择惩罚理性计算方 P_2，即该策略组合可以达到纳什均衡状态，其理性委托方与理性计算方的策略与效用用博弈树来表示，如图 2 和图 3 所示。

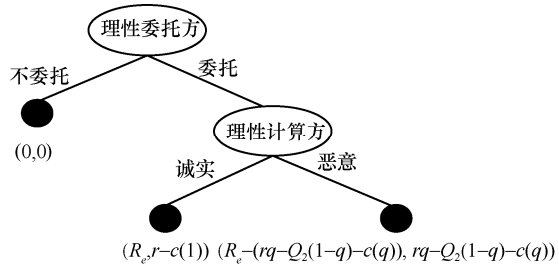

图 2　理性计算方的效用博弈树

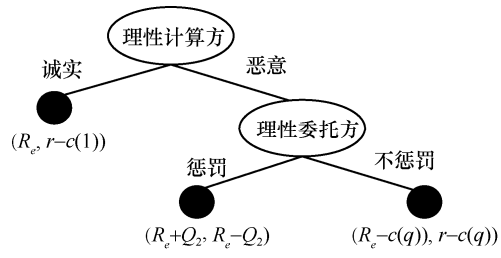

图 3　理性委托方的效用博弈树

3.2 安全模型

根据上述的博弈分析，给出了基于全同态加密与随机化混淆电路相结合的理性委托计算安全模型。对真实实验中与理想实验中输出结果进行分析，如果任意概率多项式时间的区分器不能区分这 2 个实验结果，则本实验是安全的，满足语义安全。此安全模型用于抵御恶意敌手的多项式次查询，保证委托计算输入输出的隐私性及委托计算结果的正确性。

在实验中，定义敌手在此安全模型中的优势为 $\mathrm{ADV}_{\mathcal{A}}=\Pr\left[b'=b\right]-\frac{1}{2}$，理性委托计算理想实验和真实实验下基于随机化混淆电路与全同态加密技术的安全模型如下。

1) 理性委托计算理想实验下的安全模型

理想实验 挑战者与仿真器 S。

初始化阶段 挑战者将安全参数 1^{λ} 发送给仿真器。

挑战阶段 敌手向挑战者发送有效的明文概率分布 w_i，挑战者根据概率分布 w_i 随机选择 2 个明文 w_{i0} 和 w_{i1}。

解密阶段 仿真器随机选择 $b\in\{0,1\}$，将其发送给挑战者。挑战者打开对应的明文分量得到 $\left(w_i\right)_{i\in b}$，然后将明文分量发送给仿真器。

输出阶段 仿真器调用解密算法 $\sigma_y\leftarrow\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\overline{w}_i\right)$ 得到解密结果，并输出解密结果 σ_y。

2) 理性委托计算真实实验下的安全模型

真实实验 挑战者与敌手。

初始化阶段 挑战者调用密钥生成算法 $\mathrm{KeyGen}\left(1^{\lambda},F\right)\rightarrow\left(\mathrm{PK},\mathrm{SK}\right)$ 得到公钥私钥对，并将公钥 PK 发送给敌手。

解密阶段 1 敌手查询密文 γ_i，挑战者调用解密算法 $\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\gamma_i\right)$ 进行回复。

挑战阶段 敌手向挑战者提交 2 个长度相同的明文消息 w_0 和 w_1，挑战者随机选择一个比特 b，其中 $b\in\{0,1\}$，调用加密算法 $\sigma_x^*\leftarrow\left(\mathrm{PK}_E,\mathrm{Encrypt}_E\left(\mathrm{PK}_E,w_b\right)\right)$ 加密 w_b，从而得到挑战密文 σ_x^*。挑战者将得到的挑战密文 σ_x^* 发送给敌手。

解密阶段 2 敌手查询密文 σ_x，其中 $\sigma_x\neq\sigma_x^*$。挑战者继续调用解密算法 $\sigma_y\leftarrow\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\overline{w}_i\right)$ 得到解密结果 σ_y。

输出阶段 挑战者将结果 σ_y 发送给敌手，敌手猜测 b 的值 b'。

若敌手猜出 $b=b'$，则敌手赢得本次比赛。

4 理性委托计算方案构造

本节结合混淆电路与全同态加密技术，设计可重用的理性委托计算协议。在协议中，假设理性委托方 P_1 将需要计算的函数 F 秘密地发送给理性计算方 P_2，只有当理性参与者发送正确的结果，才能使自己得到的效用最大；若理性参与者存在欺骗行为，则会受到远大于计算成本的惩罚。方案中 λ 为安全参数，执行计算函数 F 任务所需时间为 t，具体如下。

初始化阶段

首先，理性委托方 P_1 将计算函数 F 转换为布尔电路 C，并生成混淆电路 $G(C)$。根据 Yao 的混淆电路的构造，为每个电路导线 w_i 随机选择 2 个值，$w_i^0,w_i^1\leftarrow\{0,1\}^{\lambda}$。对于每个门电路 g，计算其 4 个密文 $(\gamma_{00}^g,\gamma_{01}^g,\gamma_{10}^g,\gamma_{11}^g)$。每个门电路的公钥 PK 为密文组集合，即 $\mathrm{PK}\leftarrow\cup_g(\gamma_{00}^g,\gamma_{01}^g,\gamma_{10}^g,\gamma_{11}^g)$，私钥 SK 是其选择的导线值，即 $\mathrm{SK}\leftarrow\cup_i(w_i^0,w_i^1)$。

然后，协议将执行全同态加密算法，首先由密钥生成算法生成一个新的密钥对 $\left(\mathrm{SK}_E,\mathrm{PK}_E\right)$。在此过程中将随机选择的导线 w_i 表示为输入 x 的二进制线值。利用全同态加密的密钥对将输入线值进行编码，其公有编码值为 $\sigma_x\leftarrow(\mathrm{PK}_E,\mathrm{Encrypt}_E(\mathrm{PK}_E,w_i))$。

最后，理性委托方 P_1 把混淆电路 $G(C)$ 和输入 x 的编码一起发送给接受计算任务的理性计算方 P_2，以便理性计算方在没有理性委托方存在的情况下获得 $G(x)$ 关于 x 的任何信息，从而保证输入的安全性。

委托计算阶段

理性计算方 P_2 接收到计算任务后，根据输入导线 w、w'、γ 和输出导线 $D_w(D_{w'}(\gamma))$ 构建电路 Δ，其中 D_w 为 Yao 的混淆电路中加密算法 E 对应的解密算法。根据 Yao 的混淆电路，计算方解析收到的输入编码 σ_x。由解密算法 $\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\gamma_i\right)$ 得到布尔电路正确的输出线的标签 σ_y。其中 $\sigma_y\leftarrow\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\overline{w}_i\right)$，$\overline{w}_i$ 为二进制中表示 $y=F(x)$ 的线值。理性计算方将得到的计算结果 $\sigma_y\leftarrow\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\overline{w}_i\right)$ 作为输出返还给理性委托方 P_1。

支付效用阶段

理性委托方 P_1 接收到计算结果 σ_y 后，首先利用同态加密算法的私钥 SK_E 解密 $\sigma_y \leftarrow \mathrm{Encrypt}_E\left(\mathrm{PK}_E, \overline{w}_i\right)$ 来获得 $\overline{w}_i$。然后使用公钥SK将输出线标签 σ_y 中的导线值映射到输出结果 y 的二进制表示形式上。

如果 $y=F(x)$，理性委托方 P_1 需根据约定在时间 t 内将奖励金 r 支付给理性计算方 P_2，此时理性委托方的效用为 R_e-r，理性计算方的效用为 $r-c(1)$。

如果映射失败，即 $y\neq F(x)$，理性委托方 P_1 将会对理性计算方 P_2 进行惩罚，罚金为 Q_2。此时理性委托方的效用为 $R_e-(rq-Q_2(1-q)-c(q))$，理性计算方的效用为 $rq-Q_2(1-q)-c(q)$。

由博弈分析可知，只有当理性委托方和理性计算方都选择诚实的行为策略时，他们的效益才最大，此时该策略组合也可以达到纳什均衡。

5　安全性分析

定理 1　在 DDH（decision Diffie-Hellman）假设下，本文所提的理性委托计算协议是语义安全的。

证明　本文所提理性委托计算协议是在 DDH 假设下，以全同态加密与随机化混淆电路技术为基础的。在分析其安全性时，如果 2 次输入执行的猜测结果是以不可忽略的概率分辨的，则定理的结果成立。

假设存在概率多项式时间（PPT, probabilistic polynomial time）敌手 $\mathcal{A}$，其安全参数为 λ，有不可忽略的 δ，且

$$\mathrm{ADV}_{\mathrm{RD}}^{\mathrm{Verif}}(\mathrm{RD},F,\lambda)\geqslant\delta(\lambda)$$

在协议中，定义 L 为敌手 $\mathcal{A}$ 执行查询的上限。且在理性委托计算的过程中，随机化混淆电路的门电路会随机生成，因此敌手 $\mathcal{A}$ 不能因为多次执行查询而学习标签的相关情况，如果敌手 $\mathcal{A}$ 在游戏中获得胜利，则必须是一次性查询就获得成功。

假设敌手 $\mathcal{A}$ 在第 i 次执行时获得成功，其中，$1\leqslant i\leqslant L$，则 $H_{\mathcal{A}}^{i}(\mathrm{RD},F,\lambda)=1$；如果执行失败，则 $H_{\mathcal{A}}^{i}(\mathrm{RD},F,\lambda)=0$。表示为

$$\mathrm{ADV}_{\mathcal{A}}^{i}(\mathrm{RD},F,\lambda)=\mathrm{Prob}\left[H_{\mathcal{A}}^{i}(\mathrm{RD},F,\lambda)=1\right]$$

定义敌手 $\mathcal{A}$ 的游戏为 $H_{\mathcal{A}}^{L}(\mathrm{RD},F,\lambda)$，敌手的第 i 次查询为 $H_{\mathcal{A}}^{i}(\mathrm{RD},F,\lambda)$。协议执行过程中，为第一根输入导线的标签 A 随机化选择比特置换 (θ,θ')，返回重新随机化的标签和门电路的密文对，通过解密算法可得 $\sigma_y^i=\mathrm{Eval}\left(G^i(C),A_i\right)$。同理，敌手 $\mathcal{A}$ 在第 $i+1$ 次查询为 $H_{\mathcal{A}}^{i+1}(\mathrm{RD},F,\lambda)$，协议执行过程中，为第一根输入导线的标签 A 随机化选择比特置换 (π,π')，返回重新随机化的标签和门电路的密文对，通过解密算法可得 $\sigma_y^{i+1}=\mathrm{Eval}\left(G^{i+1}(C),A_{i+1}\right)$。

在协议执行过程中，如果敌手 $\mathcal{A}$ 在第 $i+1$ 次执行时获得成功，则敌手 $\mathcal{A}$ 猜测的计算输入为 A_{i+1}；如果执行失败，敌手 $\mathcal{A}$ 猜测的计算输入为 A_i。因此敌手 $\mathcal{A}$ 在 2 次实验过程中有可忽略的概率区分，具体如下。

$$\left|H_{\mathcal{A}}^{i+1}(\mathrm{RD},F,\lambda)-H_{\mathcal{A}}^{i}(\mathrm{RD},F,\lambda)\right|\leqslant\frac{1}{p(\lambda)}$$

其中，p 是一个多项式，且对任意多项式时间敌手 $\mathcal{A}$ 有

$$\left|\mathrm{ADV}_{\mathcal{A}}^{n}(\mathrm{RD},F,\lambda)-\mathrm{ADV}_{\mathcal{A}}^{n-1}(\mathrm{RD},F,\lambda)\right|\leqslant\frac{1}{p(\lambda)}$$

在上述的实验中的优势为

$$\mathrm{ADV}_{\mathrm{RD}}^{\mathrm{Verif}}(\mathrm{RD},F,\lambda)=\Pr\left[\mathrm{Exp}_{\mathcal{A}}^{\mathrm{Verif}}[\mathrm{RD},F,\lambda]=1\right]$$

在协议中，若对于所有的概率多项式时间敌手 $\mathcal{A}$，概率 $\mathrm{ADV}_{\mathrm{RD}}^{\mathrm{Verif}}(\mathrm{RD},F,\lambda)$ 是可以忽略不计的，则理性委托计算协议RD是安全的，即敌手 $\mathcal{A}$ 在 2 次实验中以可以忽略的概率区分 2 次猜测结果，因此，上述理性委托计算协议是语义安全的。在存在恶意参与方的情况下，理性计算方无法获得关于输入 w 和输出 σ_y 的任何信息，则可以保证委托方的输入输出隐私。

证毕。

定理 2　根据设计的理性委托计算协议，当理性参与者都选择诚实策略时，协议可以满足纳什均衡状态，即全局可以达到效益最优。

证明　在协议的初始化阶段，如果理性委托方 P_1 和理性计算方 P_2 都遵守协议规则，双方将会选择最有利的行为策略。理性计算方 P_2 在其计算能力范围内接受理性委托方 P_1 发送的计算任务 $G(C)$ 和输入 x。

在委托计算阶段，理性计算方 P_2 在时间 t 内将计算结果 $\sigma_y\leftarrow\mathrm{Decrypt}_E\left(\mathrm{PK}_E,\overline{w}_i\right)$ 作为计算输出发送给理性委托方 P_1。此时需要考虑理性参与者选择的行为策略，若理性委托方与理性计算方都采取诚实策略，理性委托方就可得到 $R-r$ 的效用，理性计算方也可得到 $r-c(1)$ 的效用；若理性委托方

选择诚实策略，按时将奖励金返回给理性计算方，而理性计算方选择恶意策略，理性委托方就可得到 $R_e-\left(rq-Q_2\left(1-q\right)-c\left(q\right)\right)$ 的效用，理性计算方也可得到 $rq-Q_2\left(1-q\right)-c\left(q\right)$ 的效用；若理性委托方选择恶意策略，没有将奖励金返回给计算方，而理性计算方选择诚实策略，理性委托方就可得到 R_e-Q_1 的效用，理性计算方也可得到 $r-c\left(1\right)+Q_1$ 的效用；如果理性委托方和计算方都选择恶意策略欺骗对方，理性委托方就可得到 R_e-Q_1 的效用，理性计算方也可得到 $rq-Q_2\left(1-q\right)-c\left(q\right)$ 的效用。

在支付效用阶段，由于在博弈模型中奖励金大于计算成本，即 $r>c\left(1\right)$，且其效用有 $u\left(1\right)>u\left(q\right)$，罚金 Q_1 与 Q_2 也远大于计算成本。所以只有当理性参与方都选择诚实的策略时，理性委托方 P_1 和理性计算方 P_2 才能得到最大的效用，此时全局状态也达到最优。

证毕。

根据协议的分析可知，用子博弈精炼纳什均衡来分析理性委托计算，只有当理性参与方都选择诚实策略时，全局才可以达到最优状态，执行协议结束，即该协议具有正确性，且满足纳什均衡状态。

6　仿真实验与性能分析

6.1　仿真实验

针对本文所提的可证明安全的理性委托计算协议，将不同数量的模指数运算委托出去后引入本模型中，使用本文所提理性委托计算协议，理性委托方不需要对返回结果进行验证，委托计算所需时间对比如图 4 所示。由图 4 可知，用户通过理性委托计算所消耗时间比直接计算和委托计算消耗时间都要少，并且当委托数量增大时，3 种方式所需时间的差距也在增大，而理性委托计算的计算效率也更高。

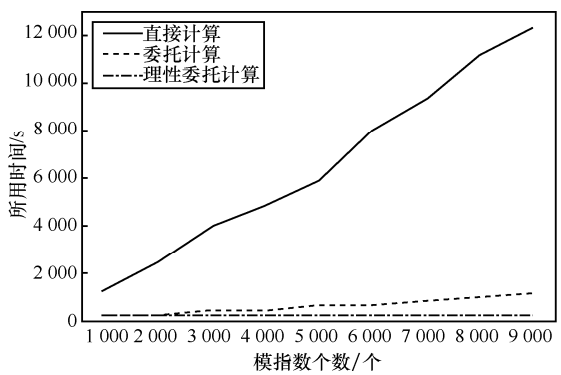

图 4　委托计算不同模型的时间开销

6.2　性能分析

本文提出的理性委托计算协议与现有的委托计算协议进行比较，具体如表 3 所示。从委托计算的计算复杂度、通信复杂度、可证明安全等方面进行对比。其中，“√”表示满足该性能，“×”表示不能满足性能。

Kupcu 等[18]提出了一种理性的委托计算协议，该协议激励所有的理性计算方正确地执行委托任务，但不关心计算任务或数据的隐藏，只关心计算结果的正确性，其计算复杂度为 $O(1)$，通信复杂度为 1，未能满足可证明安全的性能。

表 3　　本文协议与其他协议性能对比

协议	计算复杂度	通信复杂度	可证明安全
Kupcu 等协议	$O(1)$	1	×
Chen 等协议	$O(1)$	1	×
Gennaro 等协议	$O\left(\lvert C\rvert \mathrm{poly}\left(\lambda\right)\right)$	≥2	√
本文协议	$O(1)$	1	√

Chen 等[19]提出了在分布式环境中将计算任务委托给不受信任的计算方，利用新的公平有条件支付方案解决委托方与不诚实的计算方之间的信任问题。该协议的计算复杂度为 $O(1)$，通信复杂度为 1。但该方案未能对安全性进行有效的证明，未能满足可证明安全的性能。

Gennaro 等[20]提出了基于 Yao 的混淆电路与全同态加密技术构造可验证的委托计算协议，该协议虽然将计算任务委托给不受信任的计算方，但能保证参与者输入和输出的隐私。该协议虽然满足了可证明安全的性能，但由于方案中需要验证计算结果的正确性，所需的计算复杂度为 $O\left(\lvert C\rvert \mathrm{poly}\left(\lambda\right)\right)$，通信复杂度至少为 2，因此性能较低。

本文协议是基于 Yao 的混淆电路技术和全同态加密技术构造的理性委托计算协议。在协议中构造委托计算博弈模型，取消了委托方对结果的验证过程，而是通过参与者的效用函数保证计算结果的正确性。只要参与者遵守协议，最终他们都能获得最大的收益，并能达到最终的纳什均衡状态。本文协议的计算复杂度为 $O(1)$，通信复杂度为 1，且满足可证明安全的性能。

7　结束语

本文引入了理性委托计算的概念，将计算任务

委托给不受信任的服务器，并详细分析了在博弈论框架下委托计算中各参与方的效用及策略，设计了可证明安全的理性委托计算协议，该协议将 Yao 的混淆电路与全同态加密方案相结合，即使协议中存在恶意的敌手也能保证高效的委托。该协议也保证了委托方的输入输出隐私安全，以及输出结果的正确性。本文是基于 Yao 的混淆电路与全同态加密方案设计了可证明安全的理性委托计算协议，而将计算任务委托给比全同态加密更有效的委托计算方案中，这将是下一步要研究的工作。

参考文献：

[1] GOLDWASSER S, KALAI Y T, ROTHBLUM G N. Delegating computation: interactive proofs for muggles[C]// ACM Symposium on Theory of Computing. ACM, 2008:113-122.

[2] GOLDWASSER S, MICALI S, RACKOFF C. The knowledge complexity of interactive proof systems[J]. SIAM Journal on Computing, 1989, 18(1): 186-208.

[3] ARORA S, SAFRA S. Probabilistic checking of proofs: a new characterization of NP[J]. Journal of the ACM, 1998, 45(1): 70-122.

[4] CHUNG K M, KALAI Y, VADHAN S. Advances in cryptology – CRYPTO 2010: improved delegation of computation using fully homomorphic encryption [M]. Berlin: Springer, 2010:483-501.

[5] GENTRY C. A fully homomorphic encryption scheme[M]. Palo Alto: Stanford University Press, 2009.

[6] GREEN M, HOHENBERGER S, WATERS B. Outsourcing the decryption of ABE ciphertexts[C]//USENIX Conference on Security. USENIX Association, 2011: 34-34.

[7] YAO A C. Protocols for secure computations[C]//The Annual Symposium on Foundations of Computer Science. IEEE Computer Society, 1982: 160-164

[8] GENNARO R, GENTRY C, PARNO B. Advances in cryptology – CRYPTO 2010: non-interactive verifiable computing: outsourcing computation to untrusted workers[M]. Berlin: Springer, 2010: 465-482.

[9] AZAR P D, MICALI S. Rational proofs[C]//The Annual ACM Symposium on Theory of Computing. ACM, 2012: 1017-1028.

[10] AZAR P D, MICALI S. Super-efficient rational proofs[C]// Fourteenth ACM Conference on Electronic Commerce. ACM, 2013:29-30.

[11] GUO S, HUBÁČEK P, ROSEN A, et al. Rational arguments: single round delegation with sublinear verification[C]//Conference on Innovations in Theoretical Computer Science. ACM, 2014:523-540.

[12] TIAN Y L, PENG C G, LIN D D. Bayesian mechanism for rational secret sharing scheme[J]. Science China Information Sciences, 2015, 58(5):1-13.

[13] CHEN J, MCCAULEY S, SINGH S. Rational proofs with multiple provers[J]. Information Processing Letters, 2015, 114(11):237-248.

[14] KILIAN J. A note on efficient zero-knowledge proofs and arguments[C]//ACM Symposium on Theory of Computing. ACM, 1992: 723-732.

[15] Quang Duy Lã, CHEW Y H, SOONG B H . An Introduction to Game Theory[M]. Oxford: Oxford University Press, 2005.

[16] GENTRY C. Fully homomorphic encryption using ideal lattices[C]//The ACM Symposium on the Theory of Computing. ACM, 2009:169-178.

[17] GENTRY C, HALEVI S, VAIKUNTANATHAN V. I-hop homomorphic encryption and rerandomizable Yao circuits[C]//The Annual Conference on Advances in Cryptology. IEEE Press, 2010: 155-172.

[18] KUPCU, ALPTEKIN. Incentivized outsourced computation resistant to malicious contractors[J]. IEEE Transactions on Dependable and Secure Computing, 2017, 14(6):633-649.

[19] CHEN X F, LI J, SUSILO W. Efficient fair conditional payments for outsourcing computations[J]. IEEE Transactions on Information Forensics & Security, 2012, 7(6):1687-1694.

[20] GENNARO R, GENTRY C, PARNO B. Non-interactive verifiable computing: outsourcing computation to untrusted workers[C]// Conference on Advances in Cryptology. Springer-Verlag, 2010:465-482.

[作者简介]

田有亮（1982– ），男，贵州盘县人，博士，贵州大学教授，主要研究方向为博弈论、密码学与安全协议。

李秋贤（1992– ），女，河南温县人，贵州大学硕士生，主要研究方向为密码学与理性密码协议。

张铎（1987– ），男，陕西汉中人，贵州大学博士生、讲师，主要研究方向为密码学与安全协议。

王琳杰（1981– ），男，山东平度人，贵州大学博士生，主要研究方向为博弈论、密码学与安全协议。