软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2021,32(6):1910−1922 [doi: 10.13328/j.cnki.jos.006036]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于理性信任模型的理性委托计算协议*

冯能先[1,3], 田有亮[1,2,3]

[1](贵州大学 计算机科学与技术学院,贵州 贵阳 550025)

[2](省部共建公共大数据国家重点实验室(筹),贵州 贵阳 550025)

[3](贵州大学 密码学与数据安全研究所,贵州 贵阳 550025)

通讯作者: 田有亮, E-mail: youliangtian@163.com

摘 要: 传统的委托计算需要额外开销验证计算结果的正确性,导致委托计算效率较低、开销较大.针对此问题,结合博弈论与理性信任建模(rational trust modeling,简称 RTM)的思想,提出了基于理性信任模型的理性委托计算协议.通过设置恰当的效用函数,激励计算方诚实执行协议,以此来保证计算结果的可靠性.首先,基于理性信任建模的思想构造理性信任模型,将服务器的生存周期作为效用函数的参数,设计满足委托计算参与者利益的效用函数,并分析协议中参与者的行为策略,当参与者采取"诚实"策略时,可以得到理性委托计算的纳什均衡点;其次,利用改进的 NTRU(number theory research unit)公钥密码体制实现速度快、安全性高、具有抵抗量子计算攻击的能力的优点,结合 Pedersen 承诺方案,设计理性委托计算协议;最后,从正确性、安全性与性能这 3 个方面对协议进行分析,并通过实验证明生存周期对参与者效用的影响.结果表明,该协议可有效保证计算结果的可靠性.

关键词: 理性委托计算;理性信任模型;博弈论;NTRU;Pedersen 承诺

中图法分类号: TP181

中文引用格式: 冯能先,田有亮.基于理性信任模型的理性委托计算协议.软件学报,2021,32(6):1910−1922. http://www.jos.org.cn/1000-9825/6036.htm

英文引用格式: Feng NX, Tian YL. Rational delegation computing protocol based on rational trust model. Ruan Jian Xue Bao/ Journal of Software, 2021,32(6):1910−1922 (in Chinese). http://www.jos.org.cn/1000-9825/6036.htm

Rational Delegation Computing Protocol Based on Rational Trust Model

FENG Neng-Xian[1,3], TIAN You-Liang[1,2,3]

[1](College of Computer Science and Technology, Guizhou University, Guiyang 550025, China)

[2](State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China)

[3](Institute of Cryptography & Date Security, Guizhou University, Guiyang 550025, China)

Abstract: It needs supernumerary overhead to prove the correctness of computation results in traditional delegation computing, that cause less efficient and high overhead. This study addresses these problems, proposes a rational delegation computing protocol based on rational trust model by combining the thinking of game theory and rational trust modeling. In order to ensure the reliability of computation results, appropriate utility function incentive calculator is set up to execute protocol honestly. Firstly, construct the rational trust model based on the thinking of rational trust modeling. The lifetime of server is taken as parameter to design the utility function which contented profit of participants of delegation computing; analyze behavior strategy of participants in protocol further, when they take "honest" strategy, they can earn the point of Nash equilibrium. Secondly, design rational delegation computing protocol by

* 基金项目: 国家自然科学基金(61662009, 61772008, U1836205); 贵州省科技重大专项计划(20183001); 贵州省科技计划(黔科合基础[2019]1098, ZK[2021]331); 贵州省高层次创新型人才项目(黔科合平台人才[2020]6008)

Foundation item: National Natural Science Foundation of China (61662009, 61772008, U1836205); Science and Technology Major Support Program of Guizhou Province (20183001); Science and Technology Program of Guizhou Province ([2019]1098, ZK[2021]331); Project of High-level Innovative Talents of Guizhou Province ([2020]6008).

收稿时间: 2019-12-04; 修改时间: 2020-01-31; 采用时间: 2020-03-12

combining Pedersen commitment scheme and NTRU public key cryptosystem, with the advantage of high speed, high level security, and ability of resistant to quantum computing attacks. Finally, this protocol is analyzed from three aspects: correctness, security, and performance, and affection of the lifetime on the utility of participants is proven through experiment, the outcome shows reliability of computation results can be ensured effectively by the proposed protocol.

Key words: rational delegation computing; rational trust model; game theory; NTRU; Pedersen commitment

在大数据环境下,数据安全处理技术不断推陈出新,可验证计算、安全多方计算、外包计算等技术用于数据安全处理的不同环境中[1].委托计算(delegation computing,简称 DC)作为一种特殊的外包计算方法,它将计算任务委托给一个不被信任的“工人”来完成,是解决任务委托计算中产生的计算结果可靠性问题的重要手段.传统的委托计算[2]是委托方委托计算方完成某个计算任务,同时获得的计算结果具有可验证性,即计算方返回一个可验证计算结果正确性的证明,委托方验证结果正确性的效率要比自己计算任务的效率高得多,否则就失去了委托计算的意义.Kilia 等人[3]利用 Merkle 树构造了短承诺发送给验证者,验证者通过交互的方式打开承诺的某些比特来验证计算结果的正确性.Gennaro 等人[4]利用 Yao 的混淆电路[5]构造非交互式的委托计算方案,该方案有效解决了基于计算复杂性理论的困难性问题,但该方案在离线阶段效率较低.针对此问题,Chung 等人[6]利用全同态加密技术设计方案,消除了 Gennaro 等人[4]方案中的大型公钥,提升了离线阶段的效率.Cantti 等人[7]利用多个代理商的冗余来验证计算结果,但是要求至少存在一个代理商是诚实的.Xu 等人[8]结合承诺方案和加法同态加密,提出了高效的外包验证方案来保护计算任务和第三方的验证结果.该方案确保了结果的正确性,但需要昂贵的通信开销.传统的委托计算将任务委托给不可信的客户端,若客户端仅使用单个服务器,将会导致计算效率低、功能受限,委托方还需要花费额外的开销来验证结果,增加了计算和通信开销.

Katz[9]讨论了博弈论和密码学的联系,指出了博弈论和密码学协议均研究互不信任的参与者之间的交互问题,两个看似没有关系的领域可相互渗透.理性委托计算是在委托计算中引入博弈论的思想,从参与者自利的角度出发,通过设置满足参与者利益的效用函数来保障计算结果的可靠性.在理性委托计算中,参与者不是诚实的或恶意的,而是理性的.Halpern 等人[10]引入理性参与者来分析和设计共享方案和安全多方计算协议,他们认为:参与者在执行协议时,并不是要么诚实地遵守协议要么恶意地任意破坏协议,而是受效用驱使的.田等人[11]基于博弈论框架研究了秘密共享体制的分发机制和重构机制,引入了理性参与者.Azar 等人[12]根据适当评分规则提出一种理性证明系统,该系统的参与者既不是诚实的也不是恶意的,而是理性的.随后,Azar 等人[13]又利用 Utility Gaps 的思想构造了一种超高效的理性证明系统.Inasawa 等人[14]针对随机预言模型中的可计算函数构造了一个 three-message 委托方案,该方案中,验证者也是理性的.Hubácek 等人[15]通过研究理性证明系统,解决了计算能力受限的理性证明系统问题,但此方案健全性较弱.随后,Chen 等人[16]从复杂性理论的角度分析了多证明者理性证明系统的理性证明问题.Tian 等人[17]通过从理性的角度分析安全通信的问题,提出了贝叶斯理性秘密共享方案.

在理性委托计算中,理性参与者总是选择使自己效用最大的行动.因此,如何设置恰当的效用函数激励参与者诚实地执行协议的问题受到越来越多的学者关注.Yin 等人[18]基于博弈论研究委托计算问题,提出了基于比特币和 Micali-Rabin 随机向量表示技术的一对一型理性委托计算协议.Tian 等人[19]结合 Yao 的混淆电路与全同态加密,提出了可证明安全的理性委托计算协议.随后,Tian 等人[20]又结合信息论中的平均互信息量,提出了理性委托计算的攻防模型,并通过实验证明了委托方与计算方之间的最优攻防策略是博弈达到平衡点时的策略.Pham 等人[21]从经济学的角度对外包验证计算进行了研究,设计了确定外包合同价格最优模型,分析了一个和多个服务器的情况.该模型要求服务器完全诚实,但这并不符合实际情况.基于此,Khouani 等人[22]进一步研究,并在允许多服务器共谋的情况下设计了外包验证方案的最佳设置,但仍然未考虑用户容许有一定的错误率.Li 等人[23]结合信息论和博弈论的优点,根据博弈模型中纳什均衡和通道容量极限的组合,构造了一种理性委托计算方案.Vaidya 等人[24]设计了一个博弈论框架来改进现有的验证机制,该框架只考虑到了服务器的决策模型,没有考虑到用户的策略.Mehrdad 等人[25]通过信任管理和博弈论引入了理性信任建模的概念,从参与者的角度形式化了理性信任模型,模型的设计者可通过将适当的参数合并到信任函数中来激励参与者的可信度.

在传统的委托计算方案中,委托方需要花费额外的开销来验证计算方发送的证据,此过程虽然保障了计算结果的正确性,但是增加了计算和通信开销.针对目前的研究存在的委托计算效率较低、开销较大的问题,本文采用文献[25]中所提的 RTM 思想构造理性信任模型,结合 NTRU 公钥密码体制[26,27]与 Pedersen 承诺方案[28]构造了基于理性信任模型的理性委托计算协议.该方案不仅保证了计算结果的可靠性,还提高了委托计算的效率,同时还保障了理性参与者的最大利益,体现了委托计算的意义.本文的具体工作如下:

(1) 基于理性信任模型对理性委托计算协议进行博弈分析,在信任函数中引入服务器的生存周期作效用函数的参数.该模型可抵抗重放攻击,并且保证了协议的正确性;

(2) 结合理性信任模型提出了满足理性参与者最大化效益的效用函数,在此函数下分析了理性委托计算协议中参与者的行为策略,当参与者采取“诚实”策略时,可以得到理性委托计算的纳什均衡点;

(3) 在协议的委托计算阶段,利用改进的 NTRU 公钥密码体制实现速度快、安全性高、具有抵抗量子计算攻击能力的优点设计理性委托计算协议,保证了委托计算任务在传输过程中的安全性;

(4) 对协议的正确性、安全性与性能进行了分析与证明,并通过实验证明了服务器的生存周期对计算方效用的影响.

1 基础知识

1.1 博弈论

定义 1(博弈). 博弈[29]表达的基本式由局中人集合 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,$P=\{P_1,\ldots,P_n\}$,$u=\{u_1,\ldots,u_n\}$.效用函数 $u_i:S\to R$(R 代表实数空间),它表示第 i 位局中人在不同策略组合下所得到的收益.

定义 2(纳什均衡). 在博弈 $G=\{P,S,u\}$中,如果由每个博弈方的一个策略组成的某个策略组合 $s^*=(s_1^*,\ldots,s_n^*)$ 中,任一博弈方 P_i 的策略集 s_i^* 都是对其余博弈方策略组合 $(s_1^*,\ldots,s_{i-1}^*,s_{i+1}^*,\ldots,s_n^*)$ 的最佳策略,即:对于所有的 $s_i^*,s_j^*\in S(i=1,\ldots,n;\ j=1,\ldots,n)$,存在博弈 $u_i(s_i^*,s_{-i}^*)\geqslant u_i(s_j^*,s_{-i}^*)$,则称 $s^*=(s_1^*,\ldots,s_n^*)$ 为 G 的一个纳什均衡.

1.2 理性委托计算

理性委托计算是在委托计算中引入博弈论的思想,从参与者自利的角度出发,通过设置满足参与者最大化利益的效用函数来保障计算结果的可靠性.在理性委托计算中,参与者并不是要么诚实地遵守协议,要么恶意地任意破坏协议,而是理性的,是受效用驱使的.理性委托计算的过程如图 1 所示.

Fig.1 Rational delegation computing

图 1 理性委托计算

1.3 理性信任模型

理性信任模型[25]是结合信任管理和博弈论的思想,在信任模型中引入理性参与者,通过设置恰当的效用函数激励参与者“诚实”行动.

定义 3(信任函数). 设 $\Gamma_i^T(-1\leqslant\Gamma_i^T\leqslant+1)$ 代表参与人 S_i 在周期 T 的信任值,$\Gamma_i^T=0$ 表示新的参与人 S_i 的信任值.信任函数是 $R\times N$ 到 R 的映射:$(\Gamma_i^{T-1},\alpha_i)\mapsto\Gamma_i^T$,其中,

- Γ_i^{T-1} 表示参与人 S_i 在周期 T–1 的信任值;
- $\alpha_i \in \{0,1\}$:α_i=1 表示 S_i 在 T 周期诚实;α_i=0 表示 S_i 在 T 周期背叛.

信任模型的主要目的是:基于一组实体的行为,连续地度量他们(例如销售者、服务器、代理、节点、参与者等)的信任度Γ.在理性信任模型建模过程中,模型的设计者通过引入博弈论来分析参与人的行为策略,并将适当的参数合并到信任函数中来,激励参与人的信任度Γ.以卖方困境为例,假设两位商家(V_1,V_2)销售同一种商品,每位商家都有两种可能的行为.

- 行为 1:诚实.商家以 5 元的价格卖优质的商品;
- 行为 2:背叛.商家以 3 元的价格卖劣质的商品.

如果都选择诚实或背叛,他们有相同的概率 0.5 被买方选中;否则,根据价格低者优先被选择的原则,选择背叛的卖方被买方选择的概率为 1,卖方 V_1 和 V_2 的收益矩阵见表 1.

Table 1 Income matrix of V_1 and V_2

表 1 V_1 和 V_2 的收益矩阵

	V_2	
V_1	诚实	背叛
诚实	0.5,0.5	0,1
背叛	1,0	0.5,0.5

由表 1 可知:无论一位卖家选择什么行为,对于另外一位玩家来说,选择“背叛”都能够使自己的利益最大化.

假设卖方 V_1 选择诚实,卖方 V_2 选择背叛,且在 T–1 周期时有 $\Gamma_{V_1}^{T-1}=\Gamma_{V_2}^{T-1}$,此时卖方 V_2 可以达到利益最大.若采用基本信任函数 f_1 来计算两位商家(V_1,V_2)在 T 周期的信任值 $\Gamma_{V_i}^T$:

$$f_1:(\Gamma_i^{T-1},\alpha_i)\mapsto\Gamma_i^T \tag{1}$$

假设买方不知道卖方 V_2 的商品是劣质的,通过对比两位卖方的信任值和价格后更容易偏向于卖方 V_2 的商品,因此,卖方 V_2 会得到一笔巨大的收益.但是这种情况容易受到重入攻击,即卖方 V_2 选择背叛得到巨大的收益后,重新以一个新的身份重回市场销售另一种商品.理性信任模型通过合并适当的参数来激励参与人的信任度,防止了这种攻击.在卖方困境中,通过合并参数 l_i 重构信任函数 f_2:

$$f_2:(\Gamma_i^{T-1},\alpha_i,l_i)\mapsto\Gamma_i^T \tag{2}$$

其中,l_i 表示商家 i 在该市场的生存周期.根据以上所述,当卖方 V_2 背叛后,他可能会选择重入市场,但是其在市场上的生存周期及其信任值会从零开始,将很难得到买方的信任,从而得不偿失.

1.4 NURU公钥密码体制

1.4.1 NTRU 公钥密码体制

NTRU[26]是由 Hoffstein,Pipher 和 Silverman 在 1996 年提出的基于格困难问题的快速公钥密码系统,系统的密钥短且容易生成,算法的运算速度快,所需的存储空间小.其密钥生成、加密和解密过程如下.

(1) 密钥生成 *KeyGen*:选取 3 个合适的整数参数(N,p,q)、4 个次数为 N–1 的整系数多项式集合 L_h,L_g,L_ϕ,L_m,其中,p 和 q 不要求是素数,但要满足 gcd(p,q)=1,且 $p<q$.随机选取 2 个多项式 $h\in L_h$,$g\in L_g$,其中,多项式 h 在 mod q 和 mod p 下均可逆,其逆元分别表示为 H_q 和 H_p.计算$\delta\equiv H_q*g(\text{mod } q)$,以$\delta$作为公钥,$h$ 作为私钥,接收方同时还需要保存 H_p;

(2) 加密 *Enc*:设发送方欲将消息 $m\in L_m$ 发送给接收方,可对 m 作如下加密:随机选取多项式$\phi\in L_\phi$用公钥δ对消息进行加密,如下所示:

$$e=p\phi*\delta+m(\text{mod } q).$$

将密文 e 发送给接收方;

(3) 解密 *Dec*:接收方接收 e 后,使用公钥进行如下解密操作.

- $a\equiv h*e(\text{mod } q)$;

- $m \equiv H_p * a \pmod{p}$.

1.4.2 改进的 NTRU 公钥密码体制

NTRU 公钥密码体制存在一个明显的缺陷,即,在参数选取不当的情况下容易造成解密失败.文献[27]提出了改进的 NTRU 公钥密码体制[27],给出了一种新的密钥生成算法,该算法可以避免有效的格攻击.其密钥生成、加密和解密过程如下.

(1) 密钥生成 $KeyGen(1^\lambda)$:λ为安全参数.选取一个足够大的标准差σ,使得 h 可以表示为 $h=p\cdot h'+1$,其中,h'是从高斯离散分布中取样的一个多项式.

1) 从高斯分布中取样 h',使得 $h=p\cdot h'+1$;

2) 从高斯分布中取样 g,如果 $g \bmod q \in R_q^*$,其中, R_q^* 是 R_q 中可逆元素的集合,$R_q=R/qR$;

3) 返回公钥 $pk=\delta=gf^{-1} \in R_q^*$,私钥 $sk=h$;

(2) 加密 Enc:用公钥δ对消息 m 进行加密如下:

$$e=\delta s+pc+m.$$

其中,s 和 c 都是多项式;

(3) 解密 Dec:接收方接收 e 后,使用公钥进行如下解密操作:

$$m=he \pmod{p}.$$

1.5 Pedersen承诺机制

Pedersen 承诺机制[28]是满足无条件秘密性的同态承诺机制,其构造包括 3 个阶段.

(1) 初始化阶段:选择拥有大素阶 q 的乘法群 G,并选择生成元 $g_1,g_2 \in G$(假设参与双方无法获知 $\log_g g_1$),公布(g,g_1,q);

(2) 承诺阶段:发送者选择随机值 $r \in Z_q$,计算 $com=g^m g_1^r \bmod q$,然后发送 com 给接收者;

(3) 打开承诺:发送者发送(m,r)给接收者,接收者验证 com 是否等于 $g^m g_1^r \bmod q$:若 $com=g^m g_1^r \bmod q$,则接受;若 $com \neq g^m g_1^r \bmod q$,则拒绝.

2 基于理性信任模型的理性委托计算博弈分析

委托计算是指计算能力受限的委托方将计算任务委托给具有强大计算能力的计算方来进行计算,并验证计算方返回的验证计算结果正确性的证据的计算方法.理性委托计算是在委托计算的基础上引入了博弈论的思想, 从参与者自利的角度出发,通过设置恰当的效用函数来保障计算结果的正确性.因此,理性委托计算在保证计算结果正确性的同时提高了计算效率,降低了通信开销.本文引入可信第三方(trusted third party,简称 TTP),并假设参与者都是理性参与者,即参与者总是选择使自己效用最大化的行动.

设参与者为委托方 P 和计算方 C,委托计算任务为 D,计算函数为 $F(\cdot)$,计算任务 D 的价值为 $V(D)$,委托方 P 支付金额 $W_1(D)$,计算方 C 正确计算任务的成本 $W_2(D)$,计算方 C 返回错误结果的成本 $W_2'(D)$,且:

$$V(D)>W_1(D)>W_2(D)>W_2'(D).$$

2.1 理性信任模型

假设在理性信任模型下,信任管理机制对 n 个服务器 $S_i(i=1,2,\ldots,n)$有详细的信息记录,包括服务器的编号 i $(i=1,2,\ldots,n)$、服务器 $S_i(i=1,2,\ldots,n)$在周期 $T(T=0,1,\ldots)$的信任值 $\Gamma_i^T(i=1,2,...n;T=0,1,...)$、服务器 $S_i(i=1,2,\ldots,n)$的生存周期 $l_i(i=1,2,\ldots,n)$等.

在本文理性信任模型设置中,服务器 $S_i(i=1,2,\ldots,n)$在周期 $T(T=0,1,\ldots)$的信任函数为

$$f:\Gamma_i^T=\Gamma_i^{T-1}+\alpha_i\mu+\rho l_i \tag{3}$$

其中,

- $-1 \leqslant \Gamma_i^T \leqslant +1$;

- $\alpha_i\in\{0,1\}$:当$\alpha_i=1$ 时,表示服务器 $S_i(i=1,2,\ldots,n)$在周期 $T(T=0,1,\ldots)$诚实;当$\alpha_i=0$ 时,表示服务器 $S_i(i=1,2,\ldots,n)$在周期 $T(T=0,1,\ldots)$背叛;
- l_i 表示服务器 $S_i(i=1,2,\ldots,n)$的生存周期,若服务器 $S_i(i=1,2,\ldots,n)$在某一次工作中选择的行为是背叛,记此服务器 $S_i(i=1,2,\ldots,n)$的生存周期 $l_i=0$,l_i 从下一次选择诚实行动时开始增长;
- μ为常数且 $0\leqslant\mu<0.1$;
- ρ为常数且 $0\leqslant\rho l_i<1$.

服务器 $S_i(i=1,2,\ldots,n)$的效用函数取决于他自身选择的行为以及他的信任值.为了更好地激励计算方 C 正确地完成计算任务并返回正确的结果,设委托方 P 给予计算方 C 的效用函数为

$$u=W_1(D)+re(f) \tag{4}$$

$$re(f)=\begin{cases}re_1(f)=\rho l_i, & \alpha_i=1\\ re_2(f)=-\rho l_i, & \alpha_i=0\end{cases} \tag{5}$$

其中,$W_1(D)$表示计算方 C 完成计算任务 D 后委托方 P 向计算方 C 支付的金额.l_i 表示计算方 C 的生存周期,设 $0\leqslant\rho l_i<1$,当生存周期达到设定的阈值时,令$\rho l_i=1$.这为诚实的计算方 C 保证了最大利益,也保障了委托方 P 的最大利益.因为只有计算方 C 获得最大的利益后会诚实遵守协议规则,委托方 P 才能获得最大利益.$\alpha=1$ 时,表示计算方 C 诚实,此时 $re(f)=re_1(f)$,表示委托方 P 对计算方 C 正确计算任务 D 并返回正确结果的额外奖励;$\alpha=0$ 时,表示计算方 C 背叛,此时 $re(f)=re_2(f)$,表示委托方 P 对计算方 C 背叛的惩罚.

2.2 理性委托计算博弈分析

下面基于理性信任模型对理性委托计算进行博弈分析.在本文中,假设计算能力受限的委托方 P 想要将一个计算任务 D 委托给计算方 C.首先,委托方 P 将计算任务 D、计算函数 $F(\cdot)$和支付金额 $W_1(D)$经过处理后通过可信第三方 TTP 发布计算任务 D、计算函数 $F(\cdot)$和支付金额 $W_1(D)$并等待 t'时间;在 t'时间内,想要接受此计算任务的服务器 $S_i(i=1,2,\ldots,n)$向可信第三方 TTP 响应,可信第三方 TTP 接受响应并做记录;在 t'时间后,可信第三方 TTP 向信任管理机制查询已响应的服务器 $S_i(i=1,2,\ldots,n)$的信息并返回给委托方 P,委托方 P 根据自己的需求选择一个服务器 $S_i(i=1,2,\ldots,n)$作为计算方 C 来委托其完成计算任务.

委托方 P 将计算任务 D 和计算函数 $F(\cdot)$发送给计算方 C,计算方 C 接收到计算任务 D 和计算函数 $F(\cdot)$后进行计算,计算方 C 完成计算后,将计算结果 $F(D)$返回给委托方 P.委托方 P 接收到计算结果 $F(D)$后,根据计算结果对计算方 C 进行奖惩.

在博弈中,理性的参与者总会根据自己的类型选择使自身利益达到最大的行为策略.由于参与者都是理性参与者,对于计算方 C,为了使自己的效用最大,他可能选择诚实,即返回正确的计算结果;也可能选择背叛,返回错误的计算结果或在规定时间内不返回结果,因此,计算方 C 的行动策略集为{诚实,背叛}.若计算方 C 返回错误的结果或不返回结果,委托方 P 可选择“惩罚”或“不惩罚”背叛的计算方 C,因此,委托方 P 的行动策略集为{惩罚,不惩罚}.当计算方 C 按时完成计算并返回正确的计算结果,而委托方 P 没有向计算方 C 支付相应的金额,计算方 C 可向可信第三方 TTP 申诉,对委托方 P 进行索赔.设索赔金额为 ω,且$\omega>W_1(D)+|re(f)|$.委托方 P 与计算方 C 的博弈树如图 2 所示.

由图可知,该理性委托计算分两个阶段进行,参与者先后采取行动.

(1) 计算方 C 接收到任务后,选择行动策略“诚实”或“背叛”;

(2) 委托方 P 根据计算方 C 的行动策略选择自己的行动策略“惩罚”或“不惩罚”计算方 C.

若计算方 C 背离协议返回错误的计算结果或不返回结果,信任管理机制会将其生存时长 l_i 清空,即 $l_i=0$.当 $l_i=0$ 时,该服务器在之后的工作中将会遭到很大的损失.假设计算方 C 背叛后的生存时长 l_i 的增长过程为 0,$\frac{1}{5}l_i,\frac{2}{5}l_i,\frac{3}{5}l_i,\frac{4}{5}l_i,l_i$,那么他将会遭受未来的损失为

$$\begin{aligned}\gamma &= \rho\left((l_i-0)+\left(l_i-\frac{1}{5}l_i\right)+\left(l_i-\frac{2}{5}l_i\right)+\left(l_i-\frac{3}{5}l_i\right)+\left(l_i-\frac{4}{5}l_i\right)+(l_i-l_i)\right)\\ &= \rho\left(l_i+\frac{4}{5}l_i+\frac{3}{5}l_i+\frac{2}{5}l_i+\frac{1}{5}l_i+0\right)\\ &= 3\rho l_i\end{aligned} \tag{6}$$

其中,(l_i-0)表示计算方 C 选择背叛时的生存周期为 l_i,背叛后第 1 次计算的生存周期为 0.此时,他的损失为ρl_i.以此类推,当背叛的服务器的生存周期再增长到 l_i 时,除了被惩罚的金额,他还会受到额外的损失 $3\rho l_i$,参与者的效用矩阵见表 2.

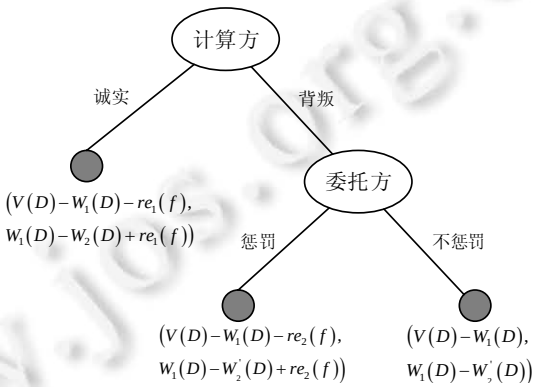

Fig.2 Game tree of participants

图 2 参与者博弈树

Table 2 Utility matrix of participants

表 2 参与者效用矩阵

		计算方	
		诚实	背叛
委托方	诚实	$V(D)-W_1(D)-re_1(f),W_1(D)-W_2(D)+re_1(f)$	$V(D)-W_1(D)-re_2(f),W_1(D)-W_2(D)+re_2(f)-\gamma$
	背叛	$V(D)-\omega,\omega-W_2(D)$	$V(D)-\omega,-W_2'(D)-re_2(f)-\gamma$

显然,只要 $W_2(D)<W_2'(D)+\gamma$,即 $W_2(D)-W_2'(D)<3\rho l_i$,“诚实”就总是该博弈的纳什均衡点.换句话说,只要计算方 C 未来的损失大于诚实计算任务成本 $W_2(D)$与背叛成本 $W_2'(D)$ 之差,参与者只有选择“诚实”行动才能使自己的效用最大.

推论 1. 若 $W_2(D)-W_2'(D)<3\rho l_i$,当计算方 C 选择“背叛”协议时将会遭受巨大的损失,因此,“诚实”总是参与者的最优选择.

3 理性委托计算协议构造

根据理性委托计算博弈分析,参与者选择“诚实”策略是本文理性委托计算协议的纳什均衡点,参与者只有选择诚实的行动,才能保证其能获得最大效用.本节结合改进的 NTRU 公钥加密算法与 Pedersen 承诺方案,构造基于理性信任模型的理性委托计算协议.

本文引入可信第三方 TTP,假设可信第三方 TTP 可以为委托方发布计算任务并记录在时间 t'内响应的服务器,响应时间 t'后,将记录发送给委托方.假设 TTP 可以帮助诚实的一方惩罚背叛的一方,在这里,我们不考虑 TTP 的酬金问题.设顺利执行一次本文理性委托计算协议花费时间为 t,因网络问题等客观原因最多容忍延迟时间为 t'',将时间 t 划分为 3 段:$(0,t_1)$为预处理阶段花费的时间,(t_1,t_2)为计算方 C 计算时间,(t_2,t_3)时间为验证和支付时间.协议执行过程如图 3 所示.

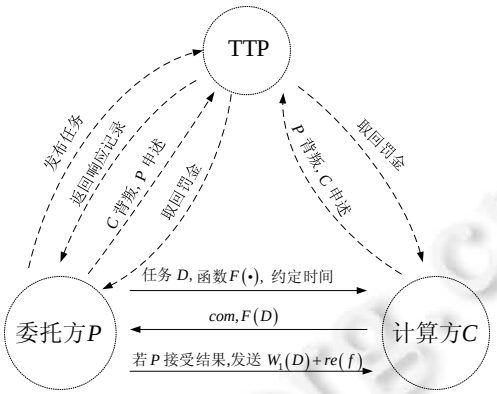

Fig.3 Rational delegation computing protocol

图 3 理性委托计算协议

3.1 预处理阶段

首先,委托方 P 通过可信第三方 TTP 发布计算任务 D 和计算函数 $F(\cdot)$以及相应的报酬 $W(D)$,并等待 t''时间.在时间 t''内,希望取得此次计算任务的服务器 $S_i(i=1,2,\ldots,n)$向 TTP 进行响应.在时间 t''后,TTP 将已响应的服务器信息记录列表发送给委托方 P,委托方 P 根据已响应的服务器的信任值选择计算方 C.

然后,根据改进的 NTRU 公钥密码体制,委托方 P 与计算方 C 选取公私钥对(pk,sk),公钥 $pk=\delta=pgf^{-1}$、私钥 $sk=h\in L_h$,用于协议中对计算任务的加密(如图 4 所示).

1. $KeyGen(N,p,q,L_h,L_g,L_\phi)\rightarrow(pk,sk)$.该过程为预处理阶段,算法由委托方执行以下步骤:
   Step 1:委托方 P 通过 TTP 发布计算任务,并选择计算方 C;
   Step 2:委托方 P 与计算方 C 选取密钥对(pk,sk).

Fig.4 Formalization of the preprocessing stage

图 4 预处理阶段形式化

3.2 委托计算阶段

委托方 P 将计算任务 D 和计算函数 $F(\cdot)$通过改进的 NTRU 公钥加密体制加密后发送给计算方 C,计算方 C 接收到计算任务 D 和计算函数 $F(\cdot)$后,在 t_2 时间内利用自己的计算资源进行计算获得计算结果 $F(D)$,并采用 Pedersen 承诺方案对计算结果 $F(D)$进行承诺.计算方选择随机数 $r\in Z_q$,计算承诺值 $com=g^{F(D)}g_1^r \bmod q$,然后将计算结果 $F(D)$和承诺值 com 发送给委托方 P(如图 5 所示).

2. $DelC(D,F)\rightarrow(F(D))$.该过程为委托计算阶段,由以下 3 个步骤组成:
   Step 1:$Enc(pk,D,F)\rightarrow(c(D),c(F))$.加密算法由委托方 P 执行,输入公钥 pk、计算任务 D 和函数 F,输出 D 和 F 对应的密文 $c(D)$和 $c(F)$;
   Step 2:$Compute(D,F,M)\rightarrow(F(D))$.算法由计算方执行,计算方输入 D 和 F 以及自己的计算资源 M 进行计算,得到计算结果 $F(D)$.
   Step 3:$Commit(r,F(D))\rightarrow(com)$.输入随机数 r 和函数值 $F(D)$,输出承诺值 com.

Fig.5 Formalization of delegating computing stage

图 5 委托计算阶段形式化

3.3 验证和支付阶段

当委托方 P 在正常时间段接收到计算方 C 发送的计算结果和承诺时,委托方 P 在 t 时间内完成验证和支付.若承诺值 com 与 $g^{F(D)}g_1^r \bmod q$ 相等,输出 1,即 $com=g^{F(D)}g_1^r \bmod q$,则委托方 P 接受该计算结果,并将相应的酬金和奖励支付给计算方 C;若承诺值 com 与 $g^{F(D)}g_1^r \bmod q$ 不相等,输出 0,即 $com\neq g^{F(D)}g_1^r \bmod q$,则委托方 P 拒绝接

受该计算结果(如图 6 所示).当输出为 0 时,委托方向可信第三方 TTP 提出申诉,TTP 对计算方 C 进行惩罚,取回罚金交给委托方 P.

当委托方 P 在 t_2 时间没有收到计算方 C 的计算结果 $F(D)$和承诺 com,委托方 P 考虑可能是因为网络问题等客观因素导致没有及时收到,委托方 P 在自己能容忍的范围内继续等待 t''时间.若委托方 P 在 t''时间内收到计算方发送的计算结果 $F(D)$和承诺 com,委托方 P 在 $t+t''$时间内按上述步骤进行验证和支付;若委托方 P 在 t''时间仍没有收到计算结果 $F(D)$和承诺 com,则判定计算方 C 背叛协议,委托方 P 向 TTP 提出申诉,可信第三方 TTP 对计算方 C 进行惩罚,取回罚金交给委托方 P.

若计算方 C 选择“诚实”行动,如约提交了正确的计算结果 $F(D)$和承诺 com,并且通过了委托方 P 的验证,但在响应时间没有收到相应的酬金和奖励,此时,计算方 C 可向 TTP 提出申诉,TTP 对委托方进行惩罚,取回罚金 ω 交给计算方 C.

3. $VerCom(r,com,F(D))\rightarrow\{0,1\}$.验证承诺算法由委托方 P 执行.输入随机数 r、函数值 $F(D)$和承诺值 com,若 $com = g^{F(D)}g_1^r \bmod q$,输出 1;否则,输出 0.

Fig.6 Formalization of verifying commitment stage

图 6 验证承诺阶段形式化

4 协议分析

本节首先通过证明定理 1 与定理 2 的形式,对基于理性信任模型的理性委托计算协议进行正确性分析与安全性分析;其次,采用表格的形式列出 l_i 和 α_i 与效用 u_i 的关系,并通过实验证明引入参数 l_i(服务器 $S_i(i=1,2,\ldots,n)$的生存周期)作为计算方 C 诚实执行协议时,得到奖励的参数对计算方的影响;然后,对本文所提基于理性信任模型的理性委托计算协议的复杂性与文献[6,30,31]的方案进行性能对比.

4.1 正确性分析

定理 1. 本文基于理性信任模型的理性委托计算协议具有正确性.

证明:根据上述博弈分析可知,“诚实”是该理性委托计算协议的纳什均衡点,即:只有委托方 P 和计算方 C 都选择诚实执行协议时,委托方 P 和计算方 C 才能获得最大效益.假设计算方 C 背叛协议,发送错误的计算结果给委托方 P,则计算方 C 将会受到严重的处罚,罚金为 $|re_2(f)|=\rho l_i$.此外,该计算方的生存时长将会被清零,即 $l_i=0$,从而导致产生未来的损失 $\gamma=3\rho l_i$.因此,若计算方 C 想要使自己获得的效益最大,将会选择“诚实”行动,正确完成计算任务,并向委托方 P 返回正确的计算结果,保证计算结果的正确性. □

4.2 安全性分析

定理 2. 假设格中最短向量问题(shortest vector problem,简称 SVP)是困难的,本文基于理性信任模型的理性委托计算协议满足语义安全性.

证明:定理的证明采用 IND 游戏的方法,假设游戏中存在多项式时间敌手 A,用 $Adv_{IND}(A)$表示敌手 A 在游戏中的优势.

初始化阶段:挑战者产生系统 Π,调用密钥生成算法得到公钥私钥对(pk,sk),并将公钥 pk 发给敌手.

挑战阶段:敌手 A 在得到公钥 pk 后选择两个长度相同的明文消息 m_0 和 m_1 提交给系统.

Phase0:挑战者选择一个随机比特 $b\in\{0,1\}$,调用加密算法将 m 加密成目标密文 $e^*=\delta s+pc+m$,同时将 e^*发送给 A.A 收到 e^*后输出一个比特 b'作为对 b 的猜测,敌手 A 的优势为:

$$Adv_{IND}(A)=\left|\Pr[b'=b]-\frac{1}{2}\right|.$$

Phase1:改变 Phase0 中公钥的生成方式,Phase1 中的公钥直接从 R_q^* 中随机均匀选取.文献[32]中指出,根据高斯离散分布输出的样本与 R_q^* 上的均匀分布是概率不可区分的.因此,敌手 A 无法区分 Phase0 和 Phase1,即:

$$|Adv_{Game0}(A)-Adv_{Game1}(A)|=0.$$

Phase2:Phase2 在 Phase1 的基础上改变其加密方法,Phase2 中加密算法不再使用改进的 NTRU 加密算法进行加密,直接从$\{0,1\}^{m\times m}$中随机均匀选取.NTRU 加密体制的困难性是基于格中最短向量问题(shortest vector problem,简称 SVP)的,因此在 Phase2 中,敌手 A 的优势与 Phase1 中敌手 A 的优势之差就是解决 SVP 问题的优势,即:

$$|Adv_{Game2}(A)-Adv_{Game1}(A)|=SVPAdv(A).$$

Phase3:在 Phase3 中,挑战者给出的目标密文 e^*不再由加密算法生成,而是从$\{0,1\}^{m\times m}$中随机均匀地选取.因此,Phase3 与 Phase2 相同,即:

$$|Adv_{Game3}(A)-Adv_{Game2}(A)|=SVPAdv(A).$$

因为 Phase3 中挑战者给出的目标密文是随机的,与明文 $m_b,b\in\{0,1\}$没有关系,所以敌手 A 在 Phase3 中的优势为:

$$Adv_{Game3}(A)=0.$$

综上可得:

$$Adv_{IND}(A)=SVPAdv(A)+SVPAdv(A).$$

在 SVP 困难性假设下,$Adv_{IND}(A)$是可以忽略的,因此,本文基于理性信任模型的理性委托计算协议是语义安全的. □

4.3 实验与性能分析

根据第 3 节基于理性信任模型的博弈分析,本文引入参数 l_i(服务器 $S_i(i=1,2,\ldots,n)$的生存周期)作为计算方 C 诚实执行协议时得到奖励的参数,服务器 $S_i(i=1,2,\ldots,n)$的生存周期 l_i 越大,得到的奖励就会越多.同时,l_i 也作为惩罚计算方背叛的参数,当计算方 C 背叛协议时,计算方 C 除了会被处罚罚金$|re_2(f)|=\rho l_i$外,其生存周期 l_i 将会清零.当计算方 C 的生存周期再次增长到 l_i 时,要遭受 $3\rho l_i$ 的损失.因此,背叛的服务器 $S_i(i=1,2,\ldots,n)$的生存周期 l_i 越大,遭受的损失也会越多.

本文设生命周期的阈值为 10,委托方 P 支付的金额 $W_1(D)=1$,$\rho=0.1$,计算方背叛协议时的计算成本 $W_2'(D)=0$,由公式(4)~公式(6)可得生存周期 l_i 和α_i与效用 u_i 的关系见表 3,生存周期 l_i 对效用 u_i 的影响如图 7 所示.

Table 3 Relationship of l_i and α_i to u_i

表 3 l_i 和α_i与 u_i 的关系

l_i		0	1	2	3	4	5	6	7	8	9	10
α_i	1	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
	0	0	−0.4	−0.8	−1.2	−1.6	−2.0	−2.4	−2.8	−3.2	−3.6	−4.0

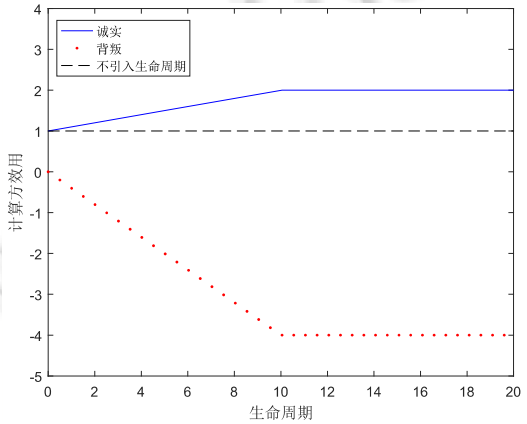

Fig.7 Effect of l_i on utility

图 7 l_i 对效用的影响

将本文提出的理性委托计算协议与文献[6,30,31]的方案进行比较,从通信复杂性、计算复杂性和是否抵抗重放攻击这 3 个方面进行对比,见表 4.

Table 4 Performance comparison of protocol

表 4 协议性能对比

	通信复杂性	计算复杂性	是否抵抗重放攻击
文献[30]	≥4	$O((2n+8)\cdot Exp+1\cdot Paring)$	否
文献[31]	$poly(S(n))$	$O(n\cdot poly\log(T(n))+poly(S(n)))$	否
文献[6]	≥2	$O(2poly(k,n,\log T))$	否
本文协议	2	$O(1)$	是

文献[30]提出了完全委托的可验证外包计算(verifiable outsourced computation with full delegation,简称 FD-VC)方案,通信复杂性大于等于 4,计算复杂性为 $O((2n+8)\cdot Exp+1\cdot Paring)$,它不能抵抗重放攻击.该方案将预处理阶段和函数计算委托给云服务器,并支持计算函数的动态更新,方案的正确性和安全性基于双线性对和双线 Diffie-Hellman 指数问题的困难性假设.由于双线性对在计算上开销比较大,因此该方案效率比较低.

文献[31]提出了一个用于非确定性计算的非交互私自验证委托计算方案,通信复杂性为 $poly(S(n))$,计算复杂性为 $O(n\cdot poly\log(T(n))+poly(S(n)))$,其中,$n$ 表示输入长度,$T(n)$和 $S(n)$分别表示非确定性时间和空间.该方案基于亚指数 LWE(learning with errors)假设,在此方案之前的工作都是基于随机预言模型(the random oracle model,简称 ROM)或知识假设,但该方案性能较低且不能抵抗重放攻击.

文献[6]基于全同态加密技术,提出了对 Gennaro 等学者[4]工作的改进协议,文献中的第 1 个协议消除了 Gennaro 等学者方案中的大公钥,委托方计算复杂性仍然是 $poly(T,k)$,但密钥长度为 $poly(n,k,\log T)$;第 2 个协议将委托方在离线阶段的计算复杂性减少到 $poly(n,k,\log T)$.该协议不能抵抗重放攻击.

本文采用 NTRU 公钥加密体制和 Pedersen 承诺方案,设计基于理性信任模型的理性委托计算协议.其通信复杂性为 2,计算复杂性为 $O(1)$,该协议可抵抗重放攻击.协议引入了可信第三方 TTP,用来帮助委托方 P 发布任务和选择计算方 C,以及帮助诚实的参与者惩罚背叛的另一方.与文献[6,30,31]相比,在本文中,委托方 P 不需要花费大量时间去验证计算方 C 返回的证明计算结果正确性的证据,而是通过验证计算方 C 返回的对计算结果的承诺:若 $com=g^{F(D)}g_1^r \bmod q$,则委托方 P 接受该计算结果并支付和奖励计算方 C;若 $com\neq g^{F(D)}g_1^r \bmod q$,则委托方 P 拒绝接受该计算结果.

5 结束语

本文基于理性信任模型研究了理性委托计算问题,并在理性信任模型下,详细分析了理性参与者在博弈论环境中的策略及效用问题,提出了基于理性信任模型的理性委托计算协议.该协议结合改进的 NTRU 公钥密码体制与 Pedersen 承诺机制,将计算任务委托给理性的计算方.为了更好地激励理性计算方“诚实”地完成计算任务并返回正确的计算结果,本文引入了计算方的生存周期作为奖励或惩罚计算方的参数,防止了参与者重入攻击的问题.本文采用 NTRU 公钥加密体制做加密运算,NTRU 加密算法虽然运算速度快、具有抵抗量子攻击的能力,但其参数多的问题将会给协议的执行增加一定的工作量.因此,选取比 NTRU 公钥密码体制更加有效的加密算法将是下一步工作的方向.

References:

[1] Hamlin A, Schear N, Shen E, Varia M, Yakoubov S, Yerukhimovich A. Cryptography for big data security. In: Proc. of the Big Data: Storage, Sharing, and Security (3S). Boca Raton: CRC Press, 2016. 241−288.

[2] Xue R, Wu Y, Liu MH, Zhang LF, Zhang R. Progress in verifiable computation. Scientia Sinica (Informationis), 2015,45(11): 1370−1388 (in Chinese with English abstract).

[3] Kilian J. Improved efficient arguments. In: Proc. of the CRYPTO. Berlin: Springer-Verlag, 1995. 311−324.

[4] Gennaro R, Gentry C, Parno B. Non-Interactive Verifiable Computing: Outsourcing Computation to Untrusted Workers. Berlin: Springer-Verlag, 2010. 465−482.

[5] Yao AC. Protocols for secure computations. In: Proc. of the Annual Symp. on Foundations of Computer Science. IEEE Computer Society, 1982. 160−164.

[6] Chung KM, Yael TK, Salil P. Improved delegation of computation using fully homomorphic encryption. In: Proc. of the CRYPTO. 2010. 483−501.

[7] Cantti R, Riva B, Rothblum GN. Practical delegation of computation using multiple servers. In: Proc. of the 18th ACM Conf. on Computer and Communications Security. ACM, 2011. 445−454.

[8] Xu G, Amariucai GT, Guan Y. Delegation of computation with verification outsourcing: curious verifiers. IEEE Trans. on Parallel Distributed Systems, 2017,28(3):717−730.

[9] Katz J. Bridging game theory and cryptography: Recent results and future directions. In: Proc. of the TCC. 2008. 251−272.

[10] Halpem J, Teague V. Rational secret sharing and multiparty computation: Extend abstract. In: Proc. of the STOC. 2004. 623−632.

[11] Tian YL, Ma JF, Peng CG, Ji WJ. Game-Theoretic analysis for the secret sharing scheme. Acta Electronica Sinica, 2011,39(12): 2790−2795 (in Chinese with English abstract).

[12] Azar PD, Micali S. Rational proofs. In: Proc. of the Annual ACM Symp. on Theory of Computing. ACM, 2012. 1017−1028.

[13] Azar PD, Micali S. Super-Efficient rational proofs. In: Proc. of the 14th ACM Conf. on Electronic Commerce. ACM, 2013. 29−30.

[14] Inasawa K, Yasunaga K. Rational proofs against rational verifiers. IACR Cryptology ePrint Archive, 2017,2017:270.

[15] Guo S, Hubácek P, Rosen A, Vald M. Rational arguments: Single round delegation with sublinear verification. In: Proc. of the ITCS. 2014. 523−540.

[16] Chen J, McCauley S, Singh S. Rational proofs with multiple provers. In: Proc. of the ITCS. 2016. 237−248.

[17] Tian YL, Peng CG, Lin DD. Bayesian mechanism for rational secret sharing scheme. Science China Information Sciences, 2015, 58(5):1−13.

[18] Yin X, Tian YL, Wang HL. Fair and rational delegation computation protocol. Ruan Jian Xue Bao/Journal of Software, 2018,29(7): 1953−1962 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5362.htm [doi: 10.13328/j.cnki.jos.005362]

[19] Tian YL, Li QX, Zhang D, Wang LJ. Provably secure rational delegation computation protocol. Journal on Communications, 2019, 40(7):35−143 (in Chinese with English abstract).

[20] Tian YL, Guo J, Wu YL, Lin H. Towards attack and defense views of rational delegation of computation. IEEE Access, 2019,7: 44037−44049.

[21] Pham V, Khouzani MHR, Cid C. Optimal contracts for outsourced computation. In: Proc. of the Decision and Game Theory for Security (GameSec 2014). Cham: Springer-Verlag, 2014. 79−98.

[22] Khouzani MHR, Pham V, Cid C. Incentive engineering for outsourced computation in the face of collusion. In: Proc. of the 13th Annual Workshop on the Economics of Information Security (WEIS 2014). 2014.

[23] Li QX, Tian YL. Rational delegation computing using information theory and game theory approach. In: Proc. of the 26th Int'l Conf. of Multi Media Modeling. 2020. 669−680.

[24] Vaidya J, Yakut I, Basu A. Efficient integrity verification for outsourced collaborative filtering. In: Proc. of the 2014 IEEE Int'l Conf. on Data Mining (ICDM). IEEE, 2014. 560−569.

[25] Mehrdad N. Rational trust modeling. In: Proc. of the GameSec 2018. 2018. 418−431.

[26] Hoffstein J, Pipher J, Silver MJH. NTRU: A ring-based public key cryptosystem. In: Proc. of the 3rd Int'l Symp. on Algorithmic Number Theory. Berlin: Springer-Verlag, 1998. 267−288.

[27] Stehle D, Steinfeld R. Making NTRU as secure as worst-case problem over idea lattices. In: Proc. of the Eurocrypt 2011. LNCS 6632. Springer-Verlag, 2011. 27−47.

[28] Pedersen TP. Non-Interactive and information-theoretic secure verifiable secret sharing. In: Proc. of the CRYPTO'91. 1992. 129−140.

[29] Roger BM. An Introduction to Game Theory. Oxford: Oxford University Press, 1985.

[30] Wang Q, Zhou FC, Peng S, Xu ZF. Verifiable outsourced computation with full delegation. In: Proc. of the ICA3PP. 2018. 270−287.

[31] Saikrishna B, Yael TK, Dakshita K, Amit S, Daniel W. Succinct delegation for low-space non-deterministic computation. In: Proc. of the STOC. 2018. 709−721.

[32] Gentry C, Peikert C, Vaikuntana V. Trapdoors for hard lattices and new cryptographic constructions. In: Proc. of the STOC. 2008. 197−206.

附中文参考文献:

[2] 薛锐,吴迎,刘牧华,张良峰,章睿.可验证计算研究进展.中国科学:信息科学,2015,45(11):1370−1388.

[11] 田有亮,马建峰,彭长根,姬文江.秘密共享体制的博弈论分析.电子学报,2011,39(12):2790−2795.

[18] 尹鑫,田有亮,王海龙.公平理性委托计算协议.软件学报,2018,29(7):1953−1962. http://www.jos.org.cn/1000-9825/5362.htm [doi: 10.13328/j.cnki.jos.005362]

[19] 田有亮,李秋贤,张铎,王琳杰.可证明安全的理性委托计算协议.通信学报,2019,40(7):135−143.

冯能先(1994－),女,硕士,主要研究领域为密码学,理性密码协议.

田有亮(1982－),男,博士,教授,博士生导师,CCF 专业会员,主要研究领域为博弈论,密码学与安全协议,大数据隐私保护.

www.jos.org.cn