



基于隐私保护的可证明安全委托计算协议

李秋贤¹,周全兴¹,王振龙¹,丁红发²,潘齐欣¹

(1.凯里学院 大数据工程学院,贵州 凯里 556011; 2.贵州财经大学 信息学院,贵阳 550025)

摘 要: 通过云计算提供的委托计算服务能够为委托方节省大量的计算时间和计算成本,但如何保证委托计算的隐私性和可证明安全性是具有挑战性的问题。结合全同态加密和多线性映射技术的优势,提出基于隐私保护的证明安全多元多项式委托计算协议。根据委托计算的输入输出隐私安全需求设计委托计算安全模型,通过多线性映射方案和全同态加密技术构造任意第三方可公开验证的委托计算协议,并在标准模型下基于多线性 Diffie-Hellman 困难性问题假设证明协议的安全性与隐私性。实验与性能分析结果表明,该协议可保证安全性,同时能够减少计算成本,满足大数据环境下委托计算模式的应用需求。

关键词: 委托计算;多线性映射;全同态加密;隐私保护;可证明安全性

开放科学(资源服务)标志码(OSID):



中文引用格式: 李秋贤,周全兴,王振龙,等. 基于隐私保护的证明安全委托计算协议[J]. 计算机工程, 2021, 47(5): 131-137.

英文引用格式: LI Qiuxian, ZHOU Quanxing, WANG Zhenlong, et al. Provable secure delegation computing protocol based on privacy protection[J]. Computer Engineering, 2021, 47(5): 131-137.

Provable Secure Delegation Computing Protocol Based on Privacy Protection

LI Qiuxian¹, ZHOU Quanxing¹, WANG Zhenlong¹, DING Hongfa², PAN Qixin¹

(1.College of Big Data Engineering, Kaili University, Kaili, Guizhou 556011, China;

2.School of Information, Guizhou University of Finance and Economics, Guiyang 550025, China)

[Abstract] Cloud-based delegation computing services can provide tremendous savings in time and computation costs for the delegate, but their privacy and provable security problems remain challenging. This paper combines fully homomorphic encryption and multi-linear mapping technology to propose a provable secure multivariate polynomial delegation computing protocol based on privacy protection. According to the input and output privacy security requirements of delegation computing, the delegation computing security model is designed. On this basis, a delegation computing protocol that is publicly verifiable to any third party is constructed by using the multi-linear mapping scheme and fully homomorphic encryption technology. Then the security and privacy of the protocol is verified based on the assumption of Multi-linear Diffie-Hellman (MDH) difficulty under the standard model. The results of the experiment and performance analysis show that this protocol ensures security and reduces the computing cost, meeting the application requirements of the delegation computing mode in big data environment.

[Key words] delegation computing; multi-linear mapping; fully homomorphic encryption; privacy protection; provable security

DOI:10.19678/j.issn.1000-3428.0058236

0 概述

在大数据快速发展的时代背景下,云计算通过强大的计算能力和存储能力^[1]为云端客户提供各种委托计算服务,能够节省委托方大量的计算时间和

计算成本,但委托计算验证结果所消耗的时间必须远少于计算函数本身,否则委托计算将毫无意义。随着计算需求量的增加,委托计算也面临很多严峻的挑战^[2],如计算协议中会存在恶意的计算方故意

基金项目: 贵州省高等学校教学内容和课程改革项目(2019170);国家自然科学基金(61772008);教育部-中国移动科研基金(MCM20170401);贵州省教育厅科技拔尖人才支持项目(黔教合KY字[2016]060);贵州省科技重大专项计划(20183001);贵州省科技计划项目(黔科合平台人才[2017]5788号);贵州省教育厅高校人文社科项目(2016FDY42)。

作者简介: 李秋贤(1992—),女,硕士研究生,主研方向为密码学、委托计算安全协议;周全兴,讲师;王振龙,教授;丁红发、潘齐欣,副教授。

收稿日期: 2020-05-03 **修回日期:** 2020-06-28 **E-mail:** 547230161@qq.com

偏离协议的执行,从而泄露用户的隐私数据或返回非正确的计算结果,或者计算方诚实地将计算结果返回给用户,而恶意的私自验证客户却声称服务器返回的结果不正确,从而拒绝支付委托费用。因此,有必要设计一种在不泄露用户隐私的前提下能够公开验证的委托计算协议。

针对数据的安全性和隐私性,公开可验证计算(PVC)方案^[3]能够提供较好的解决思路,其计算过程为:委托方对需要委托的函数进行一系列混淆加密后发送给云端服务器;服务器返回正确的计算结果和结果证明,所有的公开验证者都可以验证其结果的正确性,且验证成本远小于本地执行的成本。可验证计算一般分为两类:一类是一般函数的可验证委托计算^[4-5],适用于任何函数的计算;另一类是特殊函数的委托计算,如矩阵乘法和多项式计算^[6-8]、模指数运算^[9]等。在实际委托计算环境中,许多问题都可以转化为多元多项式的求值模型,如评估一个人的健康状况、根据水源中的COD浓度建立污染物水质分析模型等。

2010年,GENNARO等人^[10]在CRYPTO会议上首次提出可验证计算的概念,并利用全同态加密技术和混淆电路技术构造可验证计算方案,从而保证了委托计算输入与输出的隐私性。文献[11]设计选择明文攻击(CPA)安全的多项式委托计算协议来保证多项式函数的隐私性,以解决可验证计算方案只能私自验证的问题。文献[12]利用全同态加密技术,在所构造的非交互式委托计算方案中降低了委托计算的通信量。文献[13]利用多线性映射和全同态加密技术,在保证输入隐私性的情况下,设计了单变量多项式委托计算方案。文献[14]构造了多项式委托计算协议,但不能保证其可证明安全性。文献[15]通过随机化混淆电路提出的可验证委托计算方案,实现了混淆电路的可重用。文献[16-17]在博弈论的框架下设计理性委托计算协议,利用混淆电路与全同态加密技术设计可证明安全的委托计算方案,保证了可验证计算的有效性。

如何使委托计算能在公开可验证情况下保证计算结果的隐私性以及协议的安全性,一直是众多研究学者所关心的问题。文献[18]针对身份验证过程,提出一种新的方案来解决因用户损坏和服务器受损而引起的各种问题,该方案被证明是安全的,且避免了隐私性与实用性的冲突。文献[19]针对协议中会存在恶意攻击者的问题构造了一种更强大的C2C PAKE协议来应对恶意攻击,以保证协议中通信的安全。文献[20]提出了一种鲁棒的多因素身份验证方案,利用RSA密码系统的不平衡计算特性保证方案的安全性。文献[21]通过使用基于身份的签名方案构造一种针对MCC服务的PAA方案,该方案不仅能够保证参与者的通信安全,而且可以满足

MCC服务的安全要求。

本文利用全同态加密与多线性映射技术构造可公开验证的多元多项式委托计算安全协议,并在标准模型下,基于多线性 Diffie-Hellman (Multi-linear Diffie-Hellman, MDH) 困难性数学问题假设证明协议的有效性和安全隐私性,存在任意第三方利用多线性性质都可以满足委托计算结果的正确性。

1 预备知识

1.1 全同态加密

全同态加密通常包括4个阶段^[22],即预处理阶段 $(SK, PK) \leftarrow \text{Setup}_{\text{FHE}}(1^k)$ 、加密阶段 $c \leftarrow \text{Encrypt}_{\text{FHE}}(PK, m)$ 、解密阶段 $m \leftarrow \text{Decrypt}_{\text{FHE}}(SK, c)$ 和运算函数阶段 $c_f \leftarrow \text{Eval}_{\text{FHE}}(PK, c_{\text{in}}, f)$ 。

对于任意的 $k \in \mathbb{Z}_N$ 且 $k \geq 2$, BGN 同态加密算法 $\text{BGN}_k = (\text{Gen}, \text{Enc}, \text{Dec})$ 描述如下:

1) 密钥生成算法。 $\Gamma_k = (N, G_1, G_2, \dots, G_k, e, g_1, g_2, \dots, g_k) \leftarrow G(1^k, k)$ 表示为随机的 k 多线性映射实例,其中, $N = pq$, p 和 q 均为 λ bit大素数, e 是任意自然常数, g_i 是循环群 G_i 的生成元,公钥 $PK = (g_1, h)$ 和私钥 $SK = p, h = u^q, u \leftarrow G_1, h = u^q, u \leftarrow G_1$ 表示散列函数。

2) 加密算法。密文 $c = \text{Enc}(PK, m)$,输入明文 m 和公钥 PK ,输出密文 $c = g_1^m h^r \in G_1$,其中, $r \in \mathbb{Z}_N$ 。设 $\text{Enc}(m_l) = g_1^{m_l} h^{r_l} (1 \leq l \leq k)$, $h = g_1^{q\delta}, \delta \in \mathbb{Z}_N$ 且 $r_l \in \mathbb{Z}_N, h_k = (h, g_1, g_1, \dots, g_1) = g_k^{q\delta}$ 。

3) 解密算法。明文 $m = \text{Dec}(SK, c)$,此算法是以 SK 和密文 c 为输入的解密算法,输出消息 m ,使得 $c^p = (g_1^p)^m h^{rp} = (g_1^p)^m$,并求解离散对数问题得到明文 m 。

1.2 多线性映射

设 G_1 和 G_2 分别为 q 阶(q 为大素数)加法循环群 $(G_1, +)$ 和乘法循环群 $(G_2, +)$,多线性映射 $e: G_1^n \rightarrow G_2$ 具有以下性质:

1) 多线性:对于任意 $g_1, g_2, \dots, g_n \in G_1$ 和 $a_1, a_2, \dots, a_n \in \mathbb{Z}_q^*$,满足等式 $e_n(a_1 g_1, a_2 g_2, \dots, a_n g_n) = e_n(g_1, g_2, \dots, g_n)^{a_1 a_2 \dots a_n}$ 。

2) 非退化性:如果任意 g 是 G_1 的生成元,则 $e_n(g, g, \dots, g)$ 是 G_2 的生成元。

3) 可计算性:对所有的 $g_1, g_2, \dots, g_n \in G_1$,存在有效的计算法则 $e_n(g, g, \dots, g)$,则称 $e_n(g, g, \dots, g)$ 为多线性映射。

1.3 困难性问题假设

MDH 困难性问题描述如下:在 (G_1, G_2, e) 中, G_1, G_2 均是 N 循环群,随机选取 $x_1, x_2, \dots, x_n \in \mathbb{Z}_N$,对于给定 G_1 的生成元 $g_1, g_1^{a_1}, g_1^{a_2}, \dots, g_1^{a_n}$,计算

$e(g_1, g_1, \dots, g_1)^{x_1 x_2 \dots x_n} \in G_i (i=1, 2, \dots, n)$ 是困难的。

多线性离散对数问题(MDL)描述如下:设 G_1 为 N 阶循环群,对于所有的 $k > 1 (1 \leq i \leq k)$ 以及 g_i 是循环群 G_i 的一个生成元,给定 (i, g_i, g_i^x) , 对于任意 $x \in \mathbb{Z}_N^*$, 求解 x 是困难的。

2 安全模型

本文提出基于隐私保护可证明安全的委托计算协议及其安全模型,并从安全性和隐私性角度对协议进行验证。

2.1 协议安全性

在安全模型试验中,假设需要委托多元多项式函数 F , 将函数的输入定义为 x 。本文协议运用全同态加密算法对输入 x 进行加密,其中公共值为 $\sigma_x \leftarrow (PK, \text{Encrypt})$, 私有值 $\tau_x \leftarrow (SK)$ 。定义敌手在此安全模型中的优势为 $\text{ADV}_A(\text{PVMPC}, F, \lambda) = \Pr[\text{Exp}_A(\text{PVMPC}, F, \lambda) = 1]$, 即存在任意多项式概率的敌手 A , 如果概率 $\text{ADV}_A(\text{PVMPC}, F, \lambda) - \frac{1}{2}$ 是可以忽略的, 则协议是安全的。形式化分析如下:

Experiment $\text{Exp}_A[\text{PVMPC}, F, \lambda]$

$(PK, SK) \leftarrow \text{KeyGen}(1, F)$

$(x_1, x_2) \leftarrow A^{\text{ProbGen}}(PK)$

$(\sigma_1, \tau_1) \leftarrow \text{ProbGen}(PK, SK, x_1)$

$(\sigma_2, \tau_2) \leftarrow \text{ProbGen}(PK, SK, x_2)$

$b \leftarrow \{0, 1\}$

$b' \leftarrow A^{\text{ProbGen}}(PK, x_1, x_2, \sigma_b)$

if $b' = b$

else

output 0

end if

公开可验证多元多项式委托计算协议的安全性验证过程如下:

假设该试验中存在敌手 A 和挑战者 C 两个主要参与者。

1) 初始化阶段: 挑战者 C 首先需要执行 KeyGen 算法, 然后把生成的公钥 PK 发送给敌手 A 。

2) 询问阶段: 敌手 A 对 C 进行有界多项式次加密询问后发送函数 (x_1, x_2, \dots, x_n) 给 C , C 加密后将密文 $\sigma = (\sigma_{x_1}, \sigma_{x_2}, \dots, \sigma_{x_n})$ 发送给 A 。

3) 挑战阶段: 询问结束后, C 将敌手 A 攻击的目标输入 $(x_1^*, x_2^*, \dots, x_n^*)$ 和加密结果 $\sigma^* = (\sigma_{x_1^*}, \sigma_{x_2^*}, \dots, \sigma_{x_n^*})$ 发送给 A , A 返回输入为 σ^* 的计算及证明结果 $(\bar{\rho}, \bar{\pi}_1, \bar{\pi}_2, \dots, \bar{\pi}_n)$ 给公开验证者, 且 $\bar{\rho} \in \{\rho^*, \perp\}$, ρ^* 解密为 $y^* = f(a_1^*, a_2^*, \dots, a_n^*)$ 。如果 $\bar{\rho} \in \{\rho^*, \perp\}$, 则输出 1, 否则输出 0。 A 成功的优势定义为 $\text{Adv}_A(\text{PVMVC}, F, \lambda) =$

$\Pr(\text{Exp}_A^{\text{ver}}(\text{PVMVC}, F, \lambda) = 1)$ 。如果 $\text{Adv}_A(\text{PVMVC}, F, \lambda) = \Pr(\text{Exp}_A^{\text{ver}}(\text{PVMVC}, F, \lambda) = 1) < \text{neg}(\lambda)$ 成立, 其中 $\text{neg}(\lambda)$ 是关于 λ 的可忽略函数, 则表明本文协议是安全的。

2.2 协议隐私性

该协议的隐私性验证过程如下:

假设游戏中存在敌手 A 、挑战者 C 和模拟器 S 3 个主要参与者。

1) 初始化阶段: S 与 C 执行 Setup 算法, 并将公钥 PK 发送给 A 。

2) 询问阶段: A 发送 (x_1, x_2, \dots, x_n) 给 S 进行加密询问, S 返回 $\sigma = (\sigma_{x_1}, \sigma_{x_2}, \dots, \sigma_{x_n})$ 给 A , 在此过程中的询问可以是重复多次的。

3) 挑战阶段: 询问结束后, A 选择两个输入 $x_1 = (x_{11}, x_{12}, \dots, x_{1n})$ 和 $x_2 = (x_{21}, x_{22}, \dots, x_{2n})$ 发送给 S , S 选择 $i \in \{1, 2, \dots, k\}$, 计算 $\beta_1 = x_1^{2^i}$ 和 $\beta_2 = x_2^{2^i}$, 并发送给 C , C 选择 $b \in \{0, 1\}$, 并发送 $\text{Enc}(\beta_b)$ 给 S , S 计算出 $T = (\text{Enc}(x_1), \text{Enc}(x_2), \dots, \text{Enc}(x_1^{2^{i-1}}), \text{Enc}(\beta_b), \text{Enc}(x_1^{2^{i+1}}), \dots, \text{Enc}(x_1^{2^{k-1}}))$, 并将 T 发送给 A , A 返回 $b' \in \{0, 1\}$ 给 S 。

4) 猜测阶段: 如果 $b' = 1$, 则 S 输出 $\hat{b} = 1$, 否则输出 $\hat{b} = 0$; 如果 $b' = \hat{b}$, 则敌手 A 赢得这个游戏。

在上述游戏中, 本文定义敌手 A 的优势为 $|\Pr[b' = \hat{b}] - 1/2|$, 即如果敌手 A 在多项式时间内经过多次重复询问后, 都不能以大于 ϵ 的概率赢得游戏, 则表明本文协议满足输入输出隐私性。

3 协议构造

在上节构建的安全模型中引入全同态加密技术和双线性映射方案, 设计隐私保护下可公开验证的多元多项式委托计算协议, 如图 1 所示。

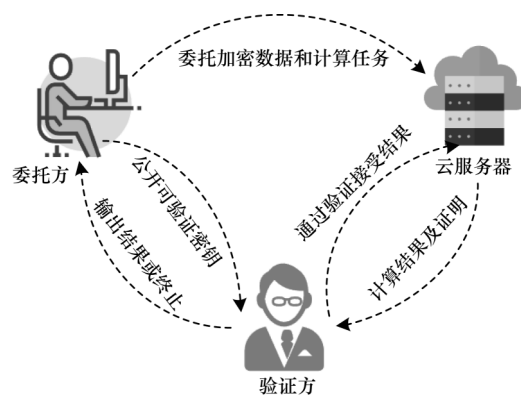


图1 可公开验证委托计算协议示意图

Fig.1 Schematic diagram of publicly verifiable delegation computing protocol

假设委托方需要将 n 元 d 次多项式函数 $F(x_1, x_2, \dots, x_n) = \sum_{i_1 + i_2 + \dots + i_n \leq d} F_{i_1 i_2 \dots i_n} x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}$ 以及输入 (x_1, x_2, \dots, x_n) 委托给云服务器进行计算, 协议算法描

述如下:

1) 初始化阶段

(1) KeyGen(l^1, F)。首先选取一个随机的 k 多线性映射实例 $\Gamma = (N, G_1, G_2, \dots, G_{k(n+1)}, e, g_1, g_2, \dots, g_{k(n+1)}) \leftarrow G(l^1, k(n+1))$, $k = \lceil \lg(d+1) \rceil$, 其中群的阶数为 $N = pq$, 且 p, q 均为 λ bit 素数, 然后选择 $s = (s_1, s_2, \dots, s_n) \in \mathbb{Z}_N^n$, 并使用 $\text{BGN}_{k(n+1)}$ 加密算法对 s 进行加密, 密文为 $\sigma_s = (\sigma_{s_1}, \sigma_{s_2}, \dots, \sigma_{s_n})$, $\sigma_{s_i} = (\sigma_{s_{i1}}, \sigma_{s_{i2}}, \dots, \sigma_{s_{ik}}) = (g_1^{s_i}, g_1^{s_i^2}, \dots, g_1^{s_i^{k-1}})$, $i \in \{1, 2, \dots, n\}$ 。为简化计算, 先令 $M = (M_1, M_2, \dots, M_n) = (\sigma_{s_1}, \sigma_{s_2}, \dots, \sigma_{s_n})$, 再计算 $F = g_1^{\text{Enc}(f(s_1, s_2, \dots, s_n))}$ 。令 $u \in G_1$, 则 $\text{BGN}_{k(n+1)}$ 加密算法的公钥为 $PK = \{\Gamma, g_1, h\}$, 私钥为 $SK = \{p, q\}$ 。

(2) ProbGen(SK, x_1, x_2, \dots, x_n)。设函数的输入为 (x_1, x_2, \dots, x_n) , 使用 $\text{BGN}_{k(n+1)}$ 加密算法对输入进行加密操作, 密文为 $\sigma_s = (\sigma_{s_1}, \sigma_{s_2}, \dots, \sigma_{s_n})$, 其中, $\sigma_{x_i} = (\sigma_{x_{i1}}, \sigma_{x_{i2}}, \dots, \sigma_{x_{ik}})$, $i \in \{1, 2, \dots, n\}$, 选取 $r_{ij} \in \mathbb{Z}_N$, 则存在 $\sigma_{x_{ij}} = g_1^{x_{ij}^{r_{ij}}}$, $i \in \{1, 2, \dots, k\}, j \in \{1, 2, \dots, k\}$, 同时输出公开验证密钥 $VK_x = (F, M)$ 和私自验证密钥 $SK_x = p$ 。

2) 委托计算阶段

(1) Compute(PK, σ)。云计算方收到 $\sigma = (\sigma_{s_1}, \sigma_{s_2}, \dots, \sigma_{s_n})$ 后, 对接收的函数进行计算 ρ 且验证证明 π , 并返回给公开验证者。

(2) 计算加密函数值。先将 $i_b (b \in \{1, 2, \dots, n\})$ 用二进制形式表示, 记为 $i_b = \sum_{l=1}^k i_{bl} 2^{l-1}$, 则 $x_i^{i_b} = x_i^{i_{b1}} \cdot (x_i^{i_{b2}})^2 \cdot \dots \cdot (x_i^{i_{bk}})^{2^{k-1}}$, 用户使用 $\text{BGN}_{k(n+1)}$ 加密算法对 $x_b, x_b^2, \dots, x_b^{2^{k-1}}$ 进行加密, 再对于任意的 $j \in \{1, 2, \dots, k\}$, 如果 $i_{bj} = 1$, 令 $\varphi_{bj} = \sigma_{x_{bj}}$, 否则令 $\varphi_{bj} = g_1$ 。因此 $\sigma_{x_b^{i_b}} = e_k(\varphi_{b1}, \varphi_{b2}, \dots, \varphi_{bk}) = g_k^{\mu_{x_b}} = g_k^m h_k^r$ 是明文 $m = x_b^{i_b}$ 的密文, 其中, $\mu_{i_b} = \prod_{j=1}^k (x_b^{2^{j-1}} + q\delta r_{bj})^{i_{bj}}, r = (\mu_{x_b} - m)/q\delta, b \in \{1, 2, \dots, n\}$, 因此, $x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}$ 在协议中所产生的密文为 $\rho_{x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}} = e_n(\rho_{x_1^{i_1}}, \rho_{x_2^{i_2}}, \dots, \rho_{x_n^{i_n}}) = g_{kn}^{\mu_{x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}}}$, 多项式函数值 $f(x_1, x_2, \dots, x_n) = \sum_{i_1 + i_2 + \dots + i_n \leq d} f_{i_1 i_2 \dots i_n} x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}$ 对应的密文为 $\sigma_{f(x_1, x_2, \dots, x_n)} = \prod_{i_1 + i_2 + \dots + i_n \leq d} \sigma_{x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}}^{f_{i_1 i_2 \dots i_n}}$ 。

3) 验证阶段

(1) 计算验证证明 π 。对于 n 元 d 次多项式 $F(x_1, x_2, \dots, x_n)$, 存在 $c_1(x_2, x_3, \dots, x_n), c_2(x_3, x_4, \dots, x_n), \dots, c_n(x_n)$, 因此, 用 $\text{BGN}_{k(n+1)}$ 加密算法对 $c_1(x_2, x_3, \dots, x_n), c_2(x_3, x_4, \dots, x_n), \dots, c_n(x_n)$ 进行加密操作, 得到密文如下:

$$\begin{aligned} \pi_1 &= \prod_{j_{1,1} + j_{1,2} + \dots + j_{1,n+1} < d} g_{k(n+1)}^{\lambda_{1,1}, \lambda_{1,2}, \dots, \lambda_{1,n} c_{j_{1,1}}, c_{j_{1,2}}, \dots, c_{j_{1,n+1}}} \\ \pi_2 &= \prod_{j_{1,1} + j_{1,2} + \dots + j_{1,n+1} < d} g_{k(n+1)}^{\lambda_{2,1}, \lambda_{2,2}, \dots, \lambda_{2,n} c_{j_{2,1}}, c_{j_{2,2}}, \dots, c_{j_{2,n+1}}} \\ &\vdots \\ \pi_n &= \prod_{j_{1,1} + j_{1,2} + \dots + j_{1,n+1} < d} g_{k(n+1)}^{\lambda_{n,1}, \lambda_{n,2}, \dots, \lambda_{n,n} c_{j_{n,1}}, c_{j_{n,2}}, \dots, c_{j_{n,n+1}}} \end{aligned} \quad (1)$$

因此, 验证证明为 $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ 。

(2) PubVer(VK_x, ρ, π)。验证方收到 $VK_x = (F, M)$, ρ, π 验证下列等式组是否成立:

$$\begin{aligned} e(F/g_1^p, g_{k(n+1)}) &= e(g_1^{M_1}/g_1^{\sigma_{x_1}}, g_{k(n+1)}^{\pi_1}) \cdot \\ &e(g_1^{M_2}/g_1^{\sigma_{x_2}}, g_{k(n+1)}^{\pi_2}) \cdot \dots \cdot e(g_1^{M_n}/g_1^{\sigma_{x_n}}, g_{k(n+1)}^{\pi_n}) \end{aligned} \quad (2)$$

如果以上验证成立, 则输出计算加密函数值 ρ , 否则输出 \perp 。

(3) PrivRet(VK_x, PK_x, ρ)。当公开验证者返回计算加密函数值 ρ 给用户时, 用户用私人检索密钥验证计算的真实值 $y = F(x_1, x_2, \dots, x_n)$, 且 y 满足 $\rho^p = (g_{kn}^p)^y$, 否则公开验证者输出 \perp 。

4 协议分析

对本文提出的基于隐私保护的证明安全委托计算协议进行有效性和安全性分析。

4.1 有效性分析

如果协议中服务器返回的计算加密函数值 ρ 及验证证明 π 能够通过公开验证者的验证, 即表明本文协议是正确且有效的。

引理 如果云端服务器是诚实的, 则有式(2)成立且 $y = f(x_1, x_2, \dots, x_n)$ 成立。

证明 因为公开验证密钥 $VK_x = (F, M)$, $M = (\sigma_{s_1}, \sigma_{s_2}, \dots, \sigma_{s_n})$, $F = g_1^{\text{Enc}(f(s_1, s_2, \dots, s_n))}$, $\pi_1 = g_{k(n+1)}^{\lambda_{1,1}, \lambda_{1,2}, \dots, \lambda_{1,n} c_{j_{1,1}}, c_{j_{1,2}}, \dots, c_{j_{1,n+1}}} = g_{k(n+1)}^{c_1(s_1, s_2, \dots, s_n)}$, 同理可得 $\pi_2 = g_{k(n+1)}^{c_2(s_2, s_3, \dots, s_n)}$, $\pi_n = g_{k(n+1)}^{c_n(s_n)}$, 所以验证式(2)是否成立等同于验证以下等式左右两边是否相等。令 $C_1 = g_{k(n+1)}^{c_1(s_1, s_2, \dots, s_n)}$, $C_2 = g_{k(n+1)}^{c_2(s_2, s_3, \dots, s_n)}$, \dots , $C_n = g_{k(n+1)}^{c_n(s_n)}$, $c_1(s_1, s_2, \dots, s_n), c_2(s_2, s_3, \dots, s_n), \dots, c_n(s_n)$ 密文为 $\sigma_{c_1(s_1, s_2, \dots, s_n)}, \sigma_{c_2(s_2, s_3, \dots, s_n)}, \dots, \sigma_{c_n(s_n)}$, 则验证过程如下:

$$\begin{aligned}
& e(g_1^{M_1/g_1^{\sigma_{s_1}}}, g_{k(n+1)}^{\pi_1}), e(g_1^{M_2/g_1^{\sigma_{s_2}}}, g_{k(n+1)}^{\pi_2}), \dots, e(g_1^{M_n/g_1^{\sigma_{s_n}}}, g_{k(n+1)}^{\pi_n}) = \\
& e(g_1^{\sigma_{s_1}}/g_1^{\sigma_{s_1}}, g_{k(n+1)}^{C_1}), e(g_1^{\sigma_{s_2}}/g_1^{\sigma_{s_2}}, g_{k(n+1)}^{C_2}), \dots, e(g_1^{\sigma_{s_n}}/g_1^{\sigma_{s_n}}, g_{k(n+1)}^{C_n}) = \\
& e(g_1, g_{k(n+1)})^{C_1(\sigma_{s_1}-\sigma_{s_1})}, e(g_1, g_{k(n+1)})^{C_2(\sigma_{s_2}-\sigma_{s_2})}, \dots, e(g_1, g_{k(n+1)})^{C_n(\sigma_{s_n}-\sigma_{s_n})} = \\
& e(g_1, g_{k(n+1)})^{\sigma_{s_1}(\sigma_{s_1}-\sigma_{s_1})}, e(g_1, g_{k(n+1)})^{\sigma_{s_2}(\sigma_{s_2}-\sigma_{s_2})}, \dots, e(g_1, g_{k(n+1)})^{\sigma_{s_n}(\sigma_{s_n}-\sigma_{s_n})} = \\
& e(g_1, g_{k(n+1)})^{\sigma_{[f(s_1, s_2, \dots, s_n)-f(x_1, x_2, \dots, x_n)]}} = e(g_1^{\sigma_{[f(s_1, s_2, \dots, s_n)-f(x_1, x_2, \dots, x_n)]}}, g_{k(n+1)}) = e(F/g_1^{\rho}, g_{k(n+1)})
\end{aligned}$$

因此,如果云端服务器是诚实的,则式(2)成立。
又因为有如下等式成立:

$$\begin{aligned}
\mu_{x_b} &= \prod_{j=1}^k (x_b^{2^{j-1}} + q\delta r_{b_j})^{i_{b_j}}, r = \frac{1}{q\delta} (\mu_{x_b} - m) \\
b &\in \{1, 2, \dots, m\} \\
p\mu_{x_1}\mu_{x_2}\dots\mu_{x_n} &\equiv px_1^{i_1} px_2^{i_2} \dots px_n^{i_n} \pmod{N} \\
\rho^p &= \prod_{i_1+i_2+\dots+i_n \leq d} \rho_{x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}}^{pf_{i_1, i_2, \dots, i_n}} = \prod_{i_1+i_2+\dots+i_n \leq d} g_{k(n+1)}^{pf_{i_1, i_2, \dots, i_n} \mu_{x_1}^{i_1} \mu_{x_2}^{i_2} \dots \mu_{x_n}^{i_n}} = \\
&\prod_{i_1+i_2+\dots+i_n \leq d} g_{k(n+1)}^{pf_{i_1, i_2, \dots, i_n} x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}} = (g_{k(n+1)}^{\rho})^{f(x_1, x_2, \dots, x_n)}
\end{aligned}$$

因此,若式(2)成立,则 $y=f(x_1, x_2, \dots, x_n)$ 也成立,即证明本文提出的基于隐私保护的证明安全委托计算协议是正确且有效的。如果存在多元多项式函数,本文协议能够在可公开验证条件下达到隐私保护和可证明安全的目标。

4.2 安全性分析

定理 1 在MDH困难性数学问题假设下,本文提出的隐私保护下可公开验证的委托计算协议是可证明安全的。

证明 假设PVMPC方案是不安全的,那么就存在一个PPT敌手A能以不可忽略的概率 ε_1 欺骗公开验证者接受一个错误的加密函数值。因此,就可以调用子程序A构建一个PPT模拟器S,以不可忽略的概率 ε_1 来解决 $(k(n+1))$ -MDH困难问题,即A的目标是在收到 $(p, q, g_1^s, g_1^{\pi_1}, g_1^{\pi_2}, \dots, g_1^{\pi_n})$ 之后, S能够计算出 $e(g_1, g_{k(n+1)}^{\rho-\rho^*})$ 。游戏开始之前,发送一个向量 $(p, q, g_1^s, g_1^{s^2}, \dots, g_1^{s^d})$ 。S模拟A过程如下:

1) 模拟器S生成系统参数。随机选择一个 n 元的 d 次多项式 $f(x_1, x_2, \dots, x_n) \in \mathbb{Z}_N[x]$, 运行群生成器得:

$$e(F/g_1^{\bar{\rho}}, g_{k(n+1)}) = e(g_1^{M_1/g_1^{\sigma_{s_1}^*}}, g_{k(n+1)}^{\pi_1^*}) e(g_1^{M_2/g_1^{\sigma_{s_2}^*}}, g_{k(n+1)}^{\pi_2^*}) \dots e(g_1^{M_n/g_1^{\sigma_{s_n}^*}}, g_{k(n+1)}^{\pi_n^*}) \quad (4)$$

$$e(F/g_1^{\rho^*}, g_{k(n+1)}) = e(g_1^{M_1/g_1^{\sigma_{s_1}^*}}, g_{k(n+1)}^{\pi_1^*}) e(g_1^{M_2/g_1^{\sigma_{s_2}^*}}, g_{k(n+1)}^{\pi_2^*}) \dots e(g_1^{M_n/g_1^{\sigma_{s_n}^*}}, g_{k(n+1)}^{\pi_n^*}) \quad (5)$$

$$e(g_1^{\rho^* - \bar{\rho}}, g_{k(n+1)}) = e(g_1^{M_1 - \sigma_{s_1}^*}, g_{k(n+1)}^{\pi_1^* - \pi_1^*}) e(g_1^{M_2 - \sigma_{s_2}^*}, g_{k(n+1)}^{\pi_2^* - \pi_2^*}) \dots e(g_1^{M_n - \sigma_{s_n}^*}, g_{k(n+1)}^{\pi_n^* - \pi_n^*}) \quad (6)$$

$$e(g_1^{\rho^* - \bar{\rho}}, g_{k(n+1)}) = e(g_1^t, g_{k(n+1)}^{\pi_1^* - \pi_1^*}) e(g_1^t, g_{k(n+1)}^{\pi_2^* - \pi_2^*}) \dots e(g_1^t, g_{k(n+1)}^{\pi_n^* - \pi_n^*}) \quad (7)$$

$\Gamma = (N, G_1, G_2, \dots, G_{k(n+1)}, e, g_1, g_2, \dots, g_{k(n+1)}) \leftarrow G(1^k, k(n+1))$,
 $k = \lceil \lg(d+1) \rceil$, $g_1 \leftarrow G_1$ 。令 $s_i = s + x_i^*$, $i \in \{1, 2, \dots, n\}$ 。由于S知道 $(p, q, g_1^s, g_1^{s^2}, \dots, g_1^{s^d})$ 和 $(x_1^*, x_2^*, \dots, x_n^*)$, 因此S可以计算以下结果:

$$\begin{aligned}
\sigma_{s_1} &= (\sigma_{s_{11}}, \sigma_{s_{12}}, \dots, \sigma_{s_{1k}}) = (g_1^{s_1}, g_1^{s_1^2}, \dots, g_1^{s_1^{2^{k-1}}}) \\
\sigma_{s_2} &= (\sigma_{s_{21}}, \sigma_{s_{22}}, \dots, \sigma_{s_{2k}}) = (g_1^{s_2}, g_1^{s_2^2}, \dots, g_1^{s_2^{2^{k-1}}}) \\
&\vdots \\
\sigma_{s_n} &= (\sigma_{s_{n1}}, \sigma_{s_{n2}}, \dots, \sigma_{s_{nk}}) = (g_1^{s_n}, g_1^{s_n^2}, \dots, g_1^{s_n^{2^{k-1}}}) \quad (3)
\end{aligned}$$

随机获得 $u \leftarrow G_1$, 通过计算 $h = u^q$, 将 $PK = (\Gamma, g_1, h; \sigma_{s_1}, \sigma_{s_2}, \dots, \sigma_{s_n}; f)$ 发送给敌手A。

2) 询问阶段。敌手A对S进行多次重复询问, A发送函数的输入 (x_1, x_2, \dots, x_n) 给S, S选择一个 $r_j \leftarrow \mathbb{Z}_N$, 然后计算 $\sigma_{x_{b_j}} = g_1^{x_{b_j}^{2^{j-1}}} h^{r_j}$, $j \in \{1, 2, \dots, k\}$, 其中 $k = \lceil \lg(d+1) \rceil$ 。输出 $\sigma_{x_b} = (\sigma_{x_{b1}}, \sigma_{x_{b2}}, \dots, \sigma_{x_{bk}})$, $b \in \{1, 2, \dots, n\}$, 最终S输出 $\sigma = (\sigma_{x_1}, \sigma_{x_2}, \dots, \sigma_{x_n})$ 给敌手A。

3) 挑战阶段。S将 $(x_1^*, x_2^*, \dots, x_n^*)$ 的加密结果 $\sigma^* = (\sigma_{x_1^*}, \sigma_{x_2^*}, \dots, \sigma_{x_n^*})$ 发送给A, A返回输入值 $(x_1^*, x_2^*, \dots, x_n^*)$ 的计算结果及验证证明 $(\bar{\rho}, \bar{\pi}_1, \bar{\pi}_2, \dots, \bar{\pi}_n)$, 且解密加密函数值 $\bar{\rho}$ 后有 $\bar{y} \neq f(x_1^*, x_2^*, \dots, x_n^*)$, 由 $\rho^* = \text{Enc}(f(x_1^*, x_2^*, \dots, x_n^*))$ 且解密 ρ^* , 有 $y^* = f(x_1^*, x_2^*, \dots, x_n^*)$ 。由假设可知 $\bar{\rho} \in \{\rho^*, \perp\}$ 以不可忽略的概率 ε 发生, 那么有 $(\bar{\rho}, \bar{\pi}_1, \bar{\pi}_2, \dots, \bar{\pi}_n)$ 和 $(\rho^*, \pi_1^*, \pi_2^*, \dots, \pi_n^*)$ 满足式(2), 即式(4)成立, 将式(4)和式(5)相除可得式(6), 由于 $s_i = s + x_i^*$, $i \in \{1, 2, \dots, n\}$, 由 $\text{BGN}_{k(n+1)}$ 支持无限次的加法同态性质可得 $\sigma_{s_i} - \sigma_{x_i^*} = \sigma_s = t$, 因此式(6)变形为式(7), 等价于式(8)成立。

$$e(g_1^{\rho^* - \bar{\rho}}, g_{k(n+1)}) = e(g_1^t, g_{k(n+1)}^{\sum_{i=1}^n \pi_i - \pi_i^*}) = g_{k(n+1)+1}^{\sum_{i=1}^n \pi_i - \pi_i^*} \quad (8)$$

由此说明模拟器 S 能够以不可忽略的概率 ε 解决数学难题 $(k(n+1))$ -MDH, 这与假设 $(k(n+1))$ -MDH 是困难问题相矛盾, 因此, 假设不成立, 则表明构建的 PVMPC 方案是安全且隐私的。

5 性能分析与实验仿真

5.1 性能分析

$$f(x_1, x_2, \dots, x_n) = \sum_{i_1 + i_2 + \dots + i_n \leq d} f_{i_1 i_2 \dots i_n} x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \text{ 这个 } n \text{ 元}$$

d 次多项式最多有 $(d+1)^n$ 个单项式, 如果客户端直接计算, 则每一项需要进行 $O(n)$ 次指数运算, 共需要进行 $O(d^n n)$ 次指数运算。当计算任务委托给服务器时, 为了保护输入信息的隐私性, 用户需要用 $BGN_{k(n+1)}$ 加密方案计算出 $2n \lg[d+1]$ 多个密文 σ 。综上分析可得, 本文提出的基于隐私保护的公开可验证委托计算协议中各阶段的效率, 如表 1 所示。本文协议与已有委托计算协议的通信复杂度和可证明安全性对比如表 2 所示。

表 1 本文协议各阶段的计算复杂度

Table 1 Computational complexity of each stage in the proposed protocol

阶段	运算	用户	服务器	验证者
KeyGen	指数运算	$O(1)$	0	0
ProbGen	指数运算	$O(nk)$	0	0
Compute	指数运算	0	$O(n^2 d) O(n^2 d)$	0
	对数运算	0		0
PubVer	指数运算	0	0	$O(n)$
PrivRet	对数运算	$O(n)$	0	0

表 2 本文协议与其他协议性能对比

Table 2 Performance comparison between the proposed protocol and other protocols

协议	通信复杂度	是否满足可证明安全性
文献[23]协议	1	不满足
文献[24]协议	1	不满足
文献[25]协议	≥ 2	满足
本文协议	1	不满足

5.2 实验仿真

为更清晰地体现本文协议的效率优势, 对本文协议进行仿真实验。用户和云服务器的运行环境分别为 Intel® Core™ i3 Processor (2.4 GHz, 4 GB 内存) 和 Intel i5 Processor (3.0 GHz, 8 GB 内存)。对 n 元 d 阶多项式进行实验模拟时, 实验的安全参数设置为安全参数 $| \lambda | = 30 \text{ bit}$, $n = 4$, 多项式的次数设为 $d = 43$ 和 $d = 127$ 。

分别对四元 43 次多项式和四元 127 次多项式进

行模拟实验, 仿真委托计算和直接计算分别与群数阶 N 和 $|f|$ 的关系。设置外包任务输入 x_i 的长度 $l = 30 \text{ bit}$, 多项式的项数 $|f| = 1\,000, 5\,000, 10\,000, 15\,000, 20\,000, 25\,000$, 实验结果如图 2 所示。

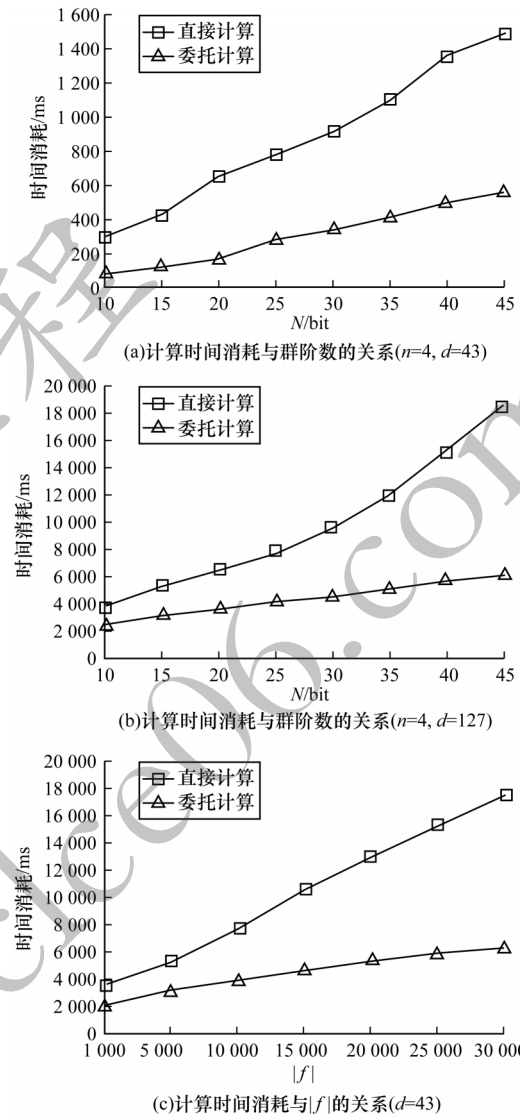


图 2 委托计算与直接计算的时间消耗

Fig.2 Time consumption of delegation calculation and direct calculation

由仿真结果可知, 本文协议中用户的计算消耗量远小于用户直接计算函数的计算量, 由此表明本文方案是有效的。

6 结束语

为实现云计算中对计算结果的公开可验证性、敏感数据的隐私性并抵抗对恶意用户偏离协议的行为, 本文基于全同态加密技术和多线性映射方案的优势, 设计一个适用于多元多项式函数的委托计算协议, 其在 MDH 困难性问题假设下满足可证明安全

性。性能分析和实验仿真结果验证了该协议的有效性和隐私保护性。本文工作是在双线性映射方案中保证委托计算协议的安全隐私性,下一步将利用其他加密技术研究更高效的委托计算方案。

参考文献

- [1] ABO-ALIAN A, BADR N L, TOLBA M F. Data storage security service in cloud computing: challenges and solutions [M]//HASSANIEN A E, MOSTAFA F, MANAF M, et al. Multimedia forensics and security. Berlin, Germany: Springer, 2017.
- [2] ZHANG Yuqing, WANG Xiaofei, LIU Xuefeng, et al. Survey on cloud computing security[J]. Journal of Software, 2016, 27(6): 1328-1348. (in Chinese)
张玉清, 王晓菲, 刘雪峰, 等. 云计算环境安全综述[J]. 软件学报, 2016, 27(6): 1328-1348.
- [3] PARNO B, RAYKOVA M, et al. How to delegate and verify in public: verifiable computation from attribute-based encryption[C]//Proceedings of TCC'12. New York, USA: ACM Press, 2012: 422-439.
- [4] APPLEBAUM B, ISHAI Y, KUSHILEVITZ E. From secrecy to soundness: efficient verification via secure computation[C]//Proceedings of International Colloquium Conference on Automata, Languages and Programming. Berlin, Germany: Springer, 2010: 152-163.
- [5] CHOI S, KATZ J, KUMARESAN R, et al. Multi-client non-interactive verifiable computation[C]//Proceedings of TCC'13. New York, USA: ACM Press, 2013: 499-518.
- [6] ZHANG Y, BLANTON M. Efficient secure and verifiable outsourcing of matrix multiplications[C]//Proceedings of International Conference on Information Security. Berlin, Germany: Springer, 2014: 158-178.
- [7] SUN Jiameng, ZHU Binrui, QIN Jing, et al. Confidentiality-preserving publicly verifiable computation[J]. International Journal of Foundations of Computer Science, 2018, 28(6): 799-818.
- [8] HOHENBERGER S, LYSYANSKAYA A. How to securely outsource cryptographic computations [C]//Proceedings of TCC'05. New York, USA: ACM Press, 2005: 264-282.
- [9] CHEN Xiaofeng, LI Jin, MA Jianfeng, et al. New algorithms for secure outsourcing of modular exponentiations [C]//Proceedings of ESORICS'12. Berlin, Germany: Springer, 2012: 541-556.
- [10] GENNARO R, GENTRY C, PARNO B. Non-interactive verifiable computing: outsourcing computation to untrusted workers [C]//Proceedings of CRYPTO'10. Berlin, Germany: Springer, 2010: 465-482.
- [11] BENABBAS S, GENNARO R, VAHLIS Y. Verifiable delegation of computation over large datasets [C]//Proceedings of CRYPTO'11. Berlin, Germany: Springer, 2011: 111-131.
- [12] JIN Fangyuan, ZHU Yanqin, LUO Xizhao. Delegation of computation scheme based on verifiable fully homomorphic encryption[J]. Computer Engineering, 2012, 38(23): 150-153. (in Chinese)
- [13] HALPEN J, TEAGUE V. Rational secret sharing and multiparty computation: extended abstract[C]//Proceedings of the 36th Annual ACM Symposium on Theory of Computing. New York, USA: ACM Press, 2004: 623-632.
- [14] TIAN Youliang, PENG Changgen, LIN Dongdai, et al. Bayesian mechanism for rational secret sharing scheme[J]. Science China Information Sciences, 2015, 58(5): 1-13.
- [15] ZHAO Qingsong, XU Huanliang. Delegation computation based on re-randomizable garbled circuit[J]. Computer Engineering, 2013, 39(12): 136-140. (in Chinese)
赵青松, 徐焕良. 基于随机化混淆电路的委托计算[J]. 计算机工程, 2013, 39(12): 136-140.
- [16] LI Qiuxian, TIAN Youliang, WANG Zuan. Rational delegation computation protocol based on fully homomorphic encryption[J]. Acta Electronica Sinica, 2019, 47(2): 216-220. (in Chinese)
李秋贤, 田有亮, 王纘. 基于全同态加密的理性委托计算协议[J]. 电子学报, 2019, 47(2): 216-220.
- [17] TIAN Youliang, LI Qiuxian, ZHANG Duo, et al. Provably secure rational delegation computation protocol[J]. Journal on Communications, 2019, 40(7): 135-143. (in Chinese)
田有亮, 李秋贤, 张铎, 等. 可证明安全的理性委托计算协议[J]. 通信学报, 2019, 40(7): 135-143.
- [18] WANG Ding, WANG Peng. Two birds with one stone: two-factor authentication with security beyond conventional bound[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 15(4): 708-722.
- [19] CHUANG P J, LIAO Y P. Efficient and secure cross-realm client-to-client password-authenticated key exchange[C]//Proceedings of the 26th International Conference on Advanced Information Networking and Applications. Washington D. C., USA: IEEE Press, 2012: 1-5.
- [20] WANG Ding, WANG Ping, WANG Chengyu. Efficient multi-factor user authentication protocol with forward secrecy for real-time data access in WSNs [J]. ACM Transactions on Cyber-Physical Systems, 2019, 4(3): 1-5.
- [21] HE D B, KUMAR N, KHAN M K. Efficient privacy-aware authentication scheme for mobile cloud computing services[J]. IEEE Systems Journal, 2016, 12(2): 1621-1631.
- [22] GENTRY C. Fully homomorphic encryption using ideal lattice[C]//Proceedings of STOC'09. New York, ACM Press, 2009: 169-178.
- [23] KUPCU A. Incentivized outsourced computation resistant to malicious contractors [J]. IEEE Transactions on Dependable & Secure Computing, 2017, 14(6): 633-649.
- [24] 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.
- [25] GENNARO R, GENTRY C, PARNO B. Non-interactive verifiable computing: outsourcing computation to untrusted workers[C]//Proceedings of CRYPTO'10. Berlin, Germany: Springer, 2010: 465-482.