

# 一种大宗商品交易数据共享的同态加密方法

郭金锋<sup>1,3</sup> 王兴芬<sup>2,3</sup>

GAO Jinfeng WANG Xingfen

## 摘要

为保证大宗商品交易数据共享的安全性和时效性,研究了一种基于同态加密的数据共享方法。结合 Paillier 同态算法和 RSA 签名技术对数据共享的方法、流程进行了设计,保证了数据共享中的隐私安全。通过对算法各阶段的实验分析后,利用中国剩余定理对 Paillier 算法的模指数运算进行优化,提升了算法加解密时效性。结果表明,Paillier 算法的加解密时效分别提升 33.5%、30.6%,能够有效满足密文计算下的大宗商品实际共享需求。

## 关键词

大宗商品交易;数据共享交换;同态加密;隐私保护;数据安全

doi: 10.3969/j.issn.1672-9528.2024.07.031

## 0 引言

石油、矿石、农产品等大宗商品的交易主体包括交易商、交易平台和监管机构。因大宗商品种类多、体量大,每天交易中会产生和处理大量的金融、交易和物流等数据,这些数据不仅包含交易商隐私信息,还含有交易平台的商业秘密,更是承载着重要的市场动态,对企业决策和政府监管具有重要的意义。然而,监管机构因职责存在交易平台共享数据的需求,而各大平台因数据资产安全又不愿共享,使得二者存在明显的“共享-安全”的矛盾。如何打破困局,保证数据隐私安全实现数据有效共享,成为当前亟须研究解决的问题。

## 1 数据共享交换的研究现状

目前,国内外关于数据共享的研究已涉及政务<sup>[1]</sup>、交通<sup>[2]</sup>、医疗<sup>[3]</sup>等诸多领域。学者们基于云服务、区块链、加密等技术进行了深入的研究与分析。(1)云服务数据共享:汪菲等<sup>[4]</sup>利用云存储技术设计了一种数据共享模型,以此保证共享数据的隐私安全;张格易<sup>[5]</sup>采用云服务技术优化传统技术架构,构建了一套综合性数据共享集成模型。

(2)区块链数据共享:葛琳等人<sup>[6]</sup>基于双链模式实现了数据的分布式存储和防篡改;Xia 等人<sup>[7]</sup>以联盟链为基础设计了一种实现跨部门隐私保护与安全共享的方法;Zahrani 等人<sup>[8]</sup>基于数据服务理念提出了一种订阅式的区块链数据共享模型。(3)加密式数据共享:董祥千等人<sup>[9]</sup>提出了一

种去中心化的共享模型,以数据加密存储,共识算法更新数据的方式保证共享过程中的安全;夏喆等人<sup>[10]</sup>使用代理重加密、同态加密和数字签名等技术,实现了共享数据的细粒度访问;Luo 等人<sup>[11]</sup>提出了基于 Paillier 算法和实用拜占庭容错共识算法的数据聚合方案。

在大宗场景下,上述方法均能实现数据共享的监管需求,但其实际应用要求却不同。云服务具备成本低、灵活易管理等优点,但存在易受攻击、网络要求高的不足;区块链具备去中心化、不可篡改性能充分保证数据隐私安全,但其部署成本高、速度时效低存在一定的制约。在加密方式下,可融合二者的优点,探索一种弱中心化的共享方法,从而实现安全、灵活、高效的数据共享。

同态加密是加密技术的一种,其技术十分成熟,支持密文计算,可有效防止数据泄露,保证共享安全。Paillier<sup>[12-13]</sup>是一种高效、算力要求低且证明完备的加法同态算法,被广泛应用于数据共享领域。RSA<sup>[14-15]</sup>则是具有乘法同态性的算法,常用于数字签名领域。基于上述分析,本文结合两者的应用设计了一种弱中心化的数据共享方法。

## 2 基于 Paillier 同态加密算法的大宗商品数据共享方法

### 2.1 大宗商品数据共享模型设计

为了应用同态加密保证大宗商品交易机构(bulk commodity trading institutions, BCT)与大宗商品监管机构(bulk commodity regulators, BCR)间数据的有效共享,模型须设置执行密文计算和统计的机构——大宗商品数据统计中心(commodity data statistics center, DC)。为保证安全性,DC 不能获取并掌握同态密钥等重要信息,故设置一个同态密钥管理机构——密钥管理中心(key management center,

1. 北京信息科技大学计算机学院 北京 100192

2. 北京信息科技大学信息管理学院 北京 100192

3. 北京信息科技大学商务智能研究所 北京 100192

KC)。针对 DC 和所有参与共享的 BCT/BCR，模型需设置一个权威的认证机构，以保证各机构的合法性和共享有效性，该机构是共享平台认证中心（shared platform certification authority, CA）。

综上，模型中设计了 DC、KC、CA 三个模型实体，加上数据共享参与者的 BCT/BCR，共计五个实体。总体模型结构设计如图 1 所示。

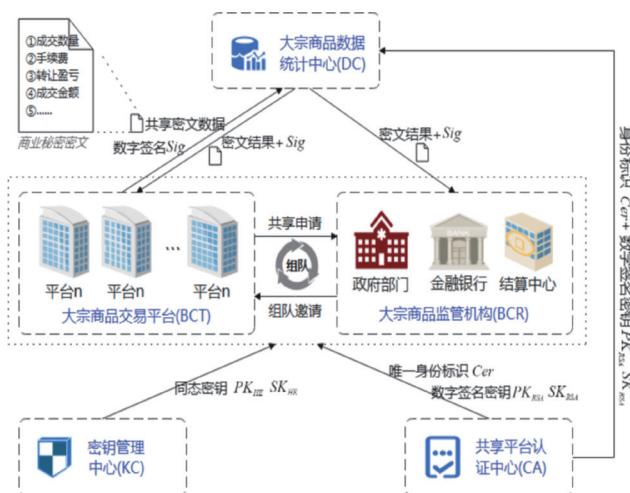


图 1 同态加密的数据共享总体模型结构

## 2.2 大宗商品数据共享交换流程

数据共享模型主要包括以下环节：环境初始化、发起共享 / 接受共享、同态密钥申请、原始数据加密、数据签名与传输、同态密文计算、密文数据解密与查看。主要算法有 Paillier 同态算法和 RSA 签名算法。模型的时序图如图 2 所示。

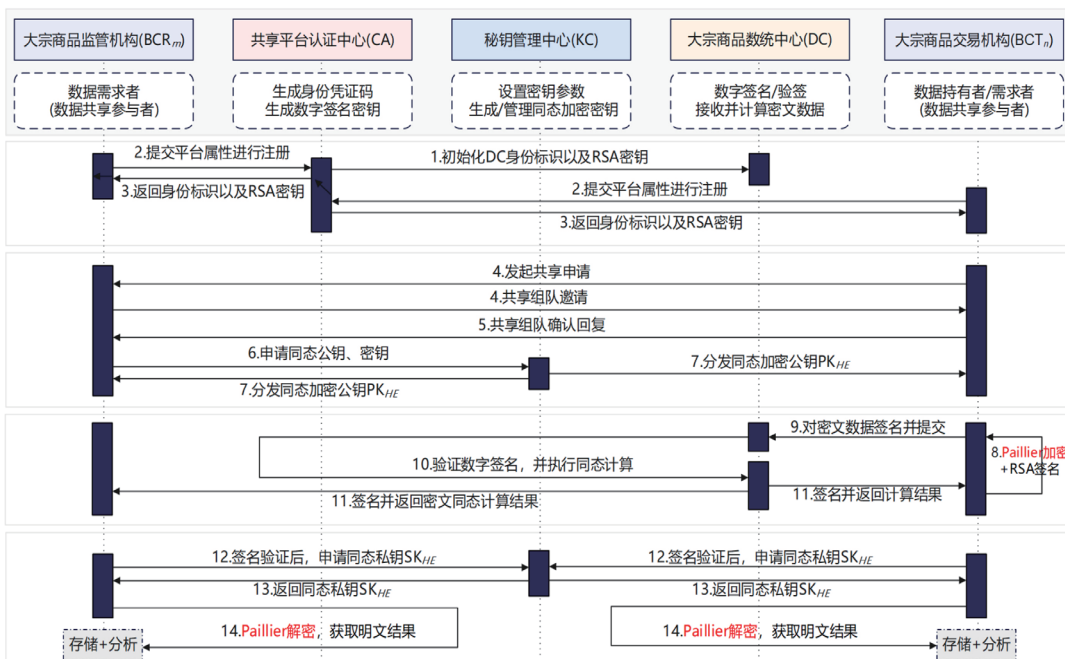


图 2 数据共享模型时序图

具体执行流程表述如下。

(1) 共享环境初始化。①内部认证：CA 为 DC 生成并分发唯一身份标识  $Cer_{DC}$ 、数字签名密钥 ( $PK_{DC}^{RSA}$  和  $SK_{DC}^{RSA}$ )。机构 KC 和 DC 同步初始化同态密钥参数和计算环境  $Env_{HE}$ 。②机构注册：当大宗交易机构 BCT 和监管机构 BCR 首次接入系统时，需通过 CA 注册生成唯一身份标识  $Cer$ ，同时生成并分发对应的数字签名密钥对 ( $pk, sk$ )。

(2) 发起 / 接受共享。有数据共享需求的 BCT/BCR 机构，通过唯一身份标识登录系统即可参与共享。共享需求者向其他 BCT 机构发出共享组队申请 / 邀请，被邀请的一方可根据自身需求进行确认是否接受。

(3) 同态密钥申请。组队成功后，由 BCT 主管的 BCR 机构向 KC 申请同态密钥，KC 生成该队伍的同态密钥对 ( $PK_{HE}$ ,  $SK_{HE}$ )，并将公钥  $PK_{HE}$  先分发给所有参与者，私钥  $SK_{HE}$  在解密时再返回，以保证数据安全。

(4) 原始数据加密。参与共享的 BCT 机构对即将共享的历史交易数据  $M_i = \{P_i, S_i\}$  进行同态加密 (含  $m$  个字段的  $n$  条历史交易信息)，数据  $M_i$  的第 1 个字段是指大宗商品的统一商品 ID (假定不同机构中的大宗商品 ID 是统一的)，后续  $k$  个字段为商品的历史交易数据  $S_i$ 。保持商品 ID 不变，对每个交易数据  $s_{pq}$  用同态公钥  $PK_{HE}$  逐个加密得到密文数据  $c_{spq} = Enc_{HE}(s_{pq}, PK_{HE})$  ( $Enc_{HE}$  为加密算法)，则交易数据集  $M_i \rightarrow C_i = \{P_i, C_{si}\}$ 。图 3 表示某共享机构的原始数据加密过程。

(5) 签名与传输：BCT 原始数据加密后，使用 MD5 生成密文  $c_{mi}$  的摘要信息  $InfoSum_{c_{mi}} = H_{MD5}(C_{mi})$ ，经私钥  $SK_{BCT}^{RSA}$

签名后获取签名信息

$Sig(InfoSum_{c_{mi}}, SK_{BCT}^{RSA})$

( $Sig$  为签名算法)，

将签名密文信息组

$\{C_{mi}, InfoSum_{c_{mi}}, Sig_{c_{mi}}\}$

提交至大宗商品数据统计中心 DC。

(6) 同态加密计算。DC 接到各 BCT 机构的数据后，首先进行验签，确认信息的合法性，然后对具有相同 ID 的各记录中  $k$  个字段进行同态计算，得到交易密文数据集  $C_{mi_{HE}} = \{C_{mi_{ALL}}, P_{ALL}\}$ 。

最后用私钥  $SK_{DC}^{RSA}$  进行签名, 并将签名密文信息组  $\{Cmi_{HE}, InfoSum_{Cmi_{HE}}, Sig_{DC}\}$  分发给各参与者。

(7) 密文数据解密。机构  $BCT_i$ 、 $BCR_i$  对签名密文信息组  $Cmi_{HE}$  验签通过后向 KC 申请同态私钥  $SK_{HE}$ , 解密获取结果  $M_{ALL}$ 。

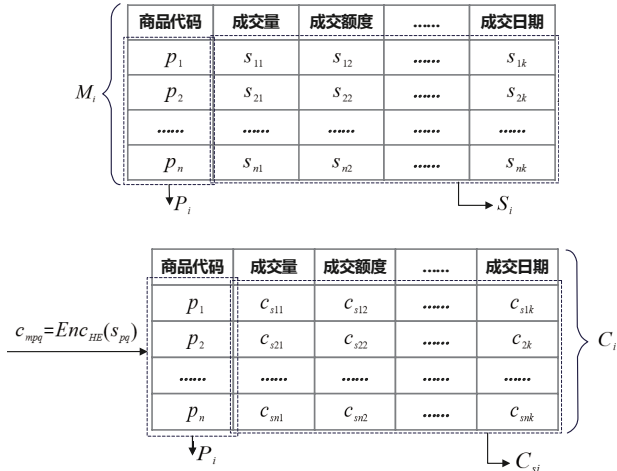


图3 数据共享机构原始数据加密过程示意图

### 2.3 基于 CRT 优化的 Paillier 加密算法

在空间意义上, 中国剩余定理 CRT<sup>[13]</sup> 是将一个代数空间分解为若干相互正交的子空间, 并与原代数空间保持同构映射关系。特别是, 当  $n=p \cdot q$ ,  $p$ 、 $q$  互质时, 存在代数空间同构性:  $Z_n = Z_p \times Z_q$ 。为提高 Paillier 算法的加解密速度, 引入 CRT 将算法模指数运算从  $Z_{n^2}$  转化到  $Z_{p^2}$  和  $Z_{q^2}$  上进行操作, 从而提升 Paillier 的加解密运算性能。

传统加解密公式如下。其中,  $E(m)$  指明文  $m$  的加密函数,  $D(c)$  指密文  $c$  的解密函数。

$$c = E(m) = g^m R^n \bmod n^2$$

$$m = D(c) = L(c^\lambda \bmod n^2) \mu \bmod n$$

针对传统 Paillier 算法, 给定两个互质整数  $p$ 、 $q$ , 且  $n=p \cdot q$ , 则存在  $Z_n$  空间上的模指数运算公式  $x = a^b \bmod n$ 。

利用 CRT, 可对 Paillier

算法模指数运算进行优化, 以提高其运算效率。

优化原理如下。

(1) 将  $a^b$  映射到  $Z_p$ 、 $Z_q$  空间。首先计算出  $a^b$  在  $Z_p$ 、 $Z_q$  上的映射:

$$x_p = a_p^{b_p}, \quad a_p = a \bmod p;$$

$x_q = a_q^{b_q}, \quad a_q = a \bmod q$ 。根据欧拉定理,  $b_p = b \bmod \phi(p)$ 、 $b_q = b \bmod \phi(q)$ , 因为  $p$  是质数, 所以  $\phi(p) = p-1$ 、 $\phi(q) = q-1$  ( $\phi(y)$  是  $y$  的欧拉函数)。代入上述映射, 有:

$$x_p = a_p^{b_p} = (a \bmod p)^{b \bmod (p-1)}$$

$$x_q = a_q^{b_q} = (a \bmod q)^{b \bmod (q-1)}$$

(2) 将  $Z_p$ 、 $Z_q$  聚合到  $Z_n$ 。根据 CRT 的同余方程组通解特性, 当  $M = m_1 m_2 \cdots m_k$  存在唯一解  $x \equiv (a_1 M_1 M_1^{-1} + a_2 M_2 M_2^{-1} + \cdots + a_k M_k M_k^{-1}) \bmod M$ , 其中  $M_i = M/m_i$ ,  $M_i^{-1}$  是  $m_i$  的逆元。

使用 CRT 通项公式计算, 得:

$$x = x_p q^{-1}(\bmod p) q + x_q p^{-1}(\bmod q) p$$

根据裴蜀定理,  $p$ 、 $q$  互质存在:

$$q^{-1}(\bmod p) q + p^{-1}(\bmod q) p = 1$$

则有:

$$x = x_p q^{-1}(\bmod p) q + x_q p^{-1}(\bmod q) p$$

$$= x_p (1 - p^{-1}(\bmod q) p) + x_q p^{-1}(\bmod q) p$$

$$= x_p + (x_q - x_p) p^{-1}(\bmod q) p$$

根据上述原理过程, 传统加解密公式优化如下:

$$c = E(m) = g^m R^n \bmod n^2 = (g^m \bmod n^2) \times (R^n \bmod n^2)$$

$$= \left\{ (g \bmod p^2)^{m \bmod p(p-1)} + \left[ (g \bmod q^2)^{m \bmod q(q-1)} \right. \right.$$

$$\left. - (g \bmod p^2)^{m \bmod p(p-1)} \right] p^{-1}(\bmod q) p \}$$

$$\times \left\{ (R \bmod p^2)^{m \bmod p(p-1)} + \left[ (R \bmod q^2)^{m \bmod q(q-1)} \right. \right.$$

$$\left. - (R \bmod p^2)^{m \bmod p(p-1)} \right] p^{-1}(\bmod q) p \}$$

$$m = D(c) = L(c^\lambda \bmod n^2) \mu \bmod n$$

$$= [L(c^\lambda \bmod n^2) / L(g^\lambda \bmod n^2)] \bmod n$$

## 3 实验与结果分析

### 3.1 实验环境

Windows 10 家庭版, 内存 16 GB DDR4, 处理器 Intel(R) Core(TM) i7-9750H, 显卡 NVIDIA GeForce GTX 1650 8 GB, 语言 Java 8.0。

### 3.2 实验数据来源

实验数据集是某企业脱敏公开的历史交易数据集, 包括 38 200 条历史交易数据。数据样本见图 4 所示。

| 市场ID | 成交单号          | 委托单号          | 成交时间     | 大宗商品ID代码 | 买卖标志 | 开平仓 | 成交价格 | 成交数量 | 转让盈亏 | 手续费 | 成交金额   |
|------|---------------|---------------|----------|----------|------|-----|------|------|------|-----|--------|
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | MZ2111   | S    | O   | 2781 | 2    | 0    | 0   | 5562   |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2110   | B    | C   | 6355 | 20   | 100  | 40  | 127100 |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2108   | B    | C   | 5769 | 1    | 25   | 2   | 5769   |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2108   | B    | C   | 5769 | 1    | 32   | 2   | 5769   |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2110   | S    | C   | 6355 | 3    | 81   | 6   | 19065  |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2108   | S    | C   | 5769 | 1    | 129  | 2   | 5769   |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2110   | S    | C   | 6355 | 4    | 108  | 8   | 25420  |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2108   | S    | C   | 5769 | 1    | 123  | 2   | 5769   |
| 3016 | 211000000**** | 211000000**** | 8/6/2021 | DS2110   | S    | C   | 6355 | 5    | 130  | 10  | 31775  |

图4 实验数据样本(历史交易数据)

### 3.3 实验结果分析

#### 3.3.1 Paillier 算法各阶段运算效率对比实验

Paillier 算法执行过程包括密钥生成、加密、加法同态运算与解密四个步骤。为了验证 Paillier 同态加密算法效率的影

响因素（算法密钥位数  $n$ 、加密数据量），通过控制变量进行了对比实验分析。

（1）密钥位数  $n$  对算法效率的影响

验证算法密钥位数  $n$  对效率的影响时，保持数据加密操作时的数据量和数字位不变，分别执行算法各阶段运算，对比算法运行时效差异。实验中，由于 Java 语言计时函数精确度的局限性，为了放大结果区分度，这里固定加密数据量 100 条 / 次、数字位数长度选择为 32 bit。实验数据统计情况如表 1 和图 5 所示。

表 1 密钥位数  $n$  效率影响数据表

| 密钥位数 /bit | 密钥生成 /ms | 加密 /ms | 运算 /ms | 解密 /ms |
|-----------|----------|--------|--------|--------|
| 64        | 2        | 17     | 0      | 9      |
| 128       | 3        | 23     | 0      | 15     |
| 256       | 5        | 40     | 1      | 46     |
| 512       | 12       | 123    | 2      | 210    |
| 1024      | 40       | 700    | 3      | 1305   |

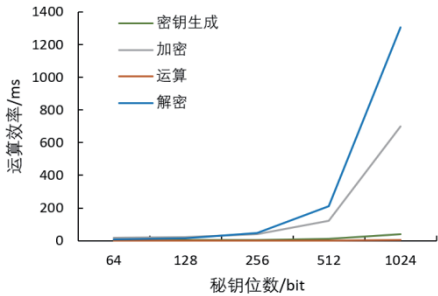


图 5 密钥位数  $n$  效率影响折线图

从实验数据可以看出，算法密钥位数  $n$  越大，时间消耗也会越大，即算法越安全，时间损耗越大；密钥生成时效相对稳定；加密和解密运算时效变化幅度最大；而 Paillier 加法同态运算时效变化最小。由此可知，在进行实际加密操作时，需根据具体需求选择合适的密钥位数。

（2）数据量对算法效率的影响

一般来说，算法加解密是  $O(n)$  复杂度的  $n$  次循环执行的过程。实验中，可以通过固定密钥位数  $n$ ，改变数据量对其进行分析。实验设置密钥位数 1024 bit、数字位数 32，Paillier 算法各阶段运行时效在不同数据量的实验测试结果如表 2 和图 6 所示。

表 2 数据量效率影响数据表

| 数据量 / 条 | 加密 /ms | 运算 /ms | 解密 /ms  |
|---------|--------|--------|---------|
| 100     | 712    | 1      | 1310    |
| 500     | 3497   | 9      | 6780    |
| 1000    | 7215   | 19     | 14 065  |
| 2500    | 17 623 | 48     | 35 009  |
| 5000    | 38 629 | 127    | 78 548  |
| 10 000  | 71 584 | 218    | 136 053 |

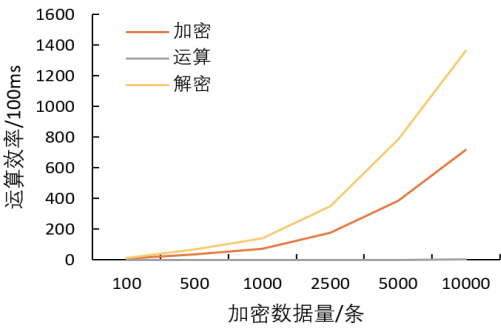


图 6 密钥位数  $n$  效率影响折线图

对比数据发现，算法加解密的运行时效与计算结果相差不大。此外，加法同态运算的时效随着数据量的增加而发生线性增长，可见二者存在一定的线性关系。数据显示，10 000 条数据执行加法同态运算有 200 ms 左右的时间花销，可知 Paillier 算法在密钥位数较长时，依然具备短时间运算处理的能力。

3.3.2 Paillier 和 DGK 算法运算效率对比实验

DGK 算法<sup>[13]</sup>和 Paillier 算法都属于概率同态加密算法。实验中，使用 Java 编程语言对 DGK 算法代码进行实验，在基于 3.3.1 小节数据量 100 条 / 次、数字位 32 bit 的条件下，分别执行 Paillier 算法和 DGK 算法的不同运算阶段，统计并记录实验数据进行对比分析，实验数据统计表 3 所示。

表 3 Paillier 和 DGK 算法密钥位数时效对比数据表

| 算法分类     | Paillier  |          |        |        | DGK       |          |        |        |
|----------|-----------|----------|--------|--------|-----------|----------|--------|--------|
|          | 密钥位数 /bit | 密钥生成 /ms | 加密 /ms | 解密 /ms | 密钥位数 /bit | 密钥生成 /ms | 加密 /ms | 解密 /ms |
| Paillier | 64        | 2        | 17     | 9      | DGK       | 26       | 2      | 3      |
|          | 128       | 3        | 23     | 15     |           | 38       | 3      | 6      |
|          | 256       | 5        | 40     | 46     |           | 106      | 3      | 23     |
|          | 512       | 12       | 123    | 210    |           | 541      | 10     | 125    |
|          | 1024      | 40       | 700    | 1305   |           | 3160     | 150    | 761    |

实验结果表明，Paillier 算法和 DGK 算法在密钥位数增加的情况下，运算时效损耗也会有所增加。在密钥生成阶段，二者运算效率比较接近；加密阶段，Paillier 算法时效性明显优于 DGK 算法；解密时，随着密钥位数变长，平均时效比在 2 倍以内，DGK 算法时效优势略显不足；在同态运算时，当密钥位数为 1024 bit 时，DGK 远高于 Paillier 的运算效率。综上可知，虽然 Paillier 的解密运算效率不高，但整体来说 Paillier 优势明显。

3.3.3 基于 CRT 优化的 Paillier 算法对比实验

在 3.3.1 节次（2）的相同实验环境下，对比分析 Paillier

算法优化前后加解密运算的时效性变化，并进行数据统计。  
实验数据见表 4。

表 4 Paillier 算法优化前后时效对比表

| 算法过程 | 加密运算            |                     |          | 解密运算            |                     |          |
|------|-----------------|---------------------|----------|-----------------|---------------------|----------|
|      | Paillier<br>/ms | CRT+<br>Paillier/ms | 效率<br>变化 | Paillier<br>/ms | CRT+<br>Paillier/ms | 效率<br>变化 |
| 100  | 712             | 463                 | +34.97%  | 1310            | 898                 | +31.45%  |
| 1000 | 7215            | 4779                | +33.76%  | 14 065          | 9243                | +30.73%  |
| 5000 | 38 629          | 26 308              | +31.90%  | 78 548          | 55 256              | +29.65%  |

从实验数据可以看出，改进后的 Paillier 同态加密算法的时效性得到了明显提升，虽然数据量增大，算法的加解密运算效率有所降低，但整体优于改进前。通过计算可知，基于 CRT 优化的 Paillier 算法在加解密的运算效率上较传统 Paillier 算法分别提高了 33.5%(加密运算)、30.6%(解密运算)，算法的优势更为明显。

4 结论与展望

为提高大宗商品交易数据共享程度，降低数据所有者对于数据隐私安全问题的顾虑，本文设计了一种大宗商品交易数据共享交换的同态加密方法。模型中，设计了大宗商品数据统计中心（DC）、密钥管理中心（KC）、共享平台认证中心（CA）三个实体对数据共享流程进行了管理，并结合 Paillier 算法和 RSA 签名强化了共享过程中的安全性。实验阶段，分别对 Paillier 算法效率、Paillier 与 DGK 算法效率以及 Paillier 算法优化前后效率进行了对比分析。实验结果表明，相同条件下，Paillier 算法的运算效率会随着密钥长度变大、数据量增长，呈现时效性降低的变化。对比 Paillier 和 DGK 算法发现，Paillier 算法整体优于 DGK 算法，更适用于当前场景。最后，实验基于 CRT 对 Paillier 算法加解密过程进行优化，使得 Paillier 算法在加解密运算过程中分别节省了 33.5%、30.6% 的时间损耗。

参考文献：

[1] 李文锋, 李林勇. 基于区块链的政务数据共享交换平台设计与应用 [J]. 电脑与信息技术, 2022, 30(5): 64-68.

[2] SHAN J, JIAN N, HAN Q, et al. Privacy-preserving and efficient data sharing for blockchain-based intelligent transportation systems [J]. Information sciences, 2023, 635: 72-85.

[3] THUSHARA G A, BHANU M S. A new hybrid encryption in fog-cloud environment for secure medical data-sharing [J]. Iran

journal of computer science, 2022, 6(2): 169-183.

[4] 汪菲, 沈苏彬. 一种基于区块链的可信数据共享解决方案 [J]. 计算机技术与发展, 2020, 30(9): 115-121.

[5] 张恪易. 基于云服务技术的数据共享交换集成应用探究 [J]. 网络安全技术与应用, 2023(9): 67-69.

[6] 葛琳, 季新生, 江涛, 等. 基于区块链技术的物联网信息共享安全机制 [J]. 计算机应用, 2019, 39(2): 458-463.

[7] QI X, EMMANUEL B S, KWAME O A, et al. MeDShare: trustless medical cta sharing among cloud service proviers via blockchain [J]. IEEE access, 2017, 5: 14757-14767.

[8] ZAHRAANI A A. Subscription-based data-sharing model using blockchain and data as a service [J]. IEEE access, 2020, 8: 115966-115981.

[9] 董祥千, 郭兵, 沈艳, 等. 一种高效安全的去中心化数据共享模型 [J]. 计算机学报, 2018, 41(5): 1021-1036.

[10] 夏喆, 罗宾, 徐桂彬, 等. 智能电网中支持细粒度访问控制的隐私保护数据聚合方案 [J]. 信息网络安全, 2021, 21(11): 28-39.

[11] LUO X, XUE K, XU J, et al. Blockchain based secure data aggregation and distributed power dispatching for microgrids [J]. IEEE transactions on smart grid, 2021, 12(6): 5268-5279.

[12] 王婧琳. 基于同态加密的金融数据安全共享方案研究及实现 [D]. 哈尔滨: 哈尔滨工业大学, 2021.

[13] 马鑫堃, 李英娜, 李申章. 基于区块链技术的电网数据隐私保护与共享方法 [J]. 电力科学与工程, 2023, 39(5): 1-9.

[14] 程朝辉. 数字签名技术概览 [J]. 信息安全与通信保密, 2020(7): 48-62.

[15] MOHAMMED S J, TAHA D B. Performance evaluation of RSA, ElGamal, and paillier partial homomorphiccryption algorithms [C]// 2022 International Conference on Computer Science and Software Engineering. Piscataway: IEEE, 2022: 89-94.

【作者简介】

郜金锋（1994—），男，河南周口人，硕士研究生，研究方向：数据安全与隐私保护。

（收稿日期：2024-05-08）