# 基于多项式划分的NTRU加密域可逆数据隐藏方案

刘定财，吴昊天，庄振威，何军辉

引用本文

刘定财，吴昊天，庄振威，何军辉.基于多项式划分的NTRU加密域可逆数据隐藏方案[J]．计算机科学，2023，50(8)：294-303.

LIU Dingcai，WU Haotian，ZHUANG Zhenwei，HE Junhui. Reversible Data Hiding Scheme in NTRU Encrypted Domain Based on Polynomial Partition [J]. Computer Science，2023，50(8)：294-303.

---

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220800245

# 基于多项式划分的 NTRU 加密域可逆数据隐藏方案

刘定财　吴昊天　庄振威　何军辉
华南理工大学计算机科学与工程学院　广州 510006
(ldc971022@163.com)

**摘　要**　随着云计算技术的发展和隐私保护的需要,同态加密域中的可逆数据隐藏已成为一项研究热点。加密域可逆数据隐藏方案大多利用了图像中像素点之间的相关性及冗余,适用范围受到了一定的限制。为了提高数据隐藏方案的适用性和嵌入容量,针对 NTRU(Number Theory Research Unit)加密系统,提出了一种基于多项式划分的可逆数据隐藏方案。该方案将 NTRU 加密系统中的多项式空间划分为用于表示原始载体的明文段和用于隐藏数据的数据隐藏段,可用于在多种加密的数字媒体中隐藏数据。接收者可以从密文中直接提取一部分隐藏的数据,并能从解密得到的明文中提取另一部分隐藏的数据,并无损地恢复原始明文。在实验部分,分别以灰度图像和文本为例,对所提算法的可行性进行验证。实验结果表明,对于一个以 8 比特表示的明文值,其密文中最多可以隐藏 $N$-8 比特的数据,其中 $N$ 为 NTRU 加密系统中的参数;当 $N$ 取 503 时,在一个密文中最多可以隐藏 495 比特的数据,并能无损地恢复出原始明文值。与现有的同类方案相比,该方案所提的 NTRU 域可逆数据隐藏算法具有较高的嵌入容量和较强的适用性。

**关键词**:可逆数据隐藏;NTRU 加密系统;多项式划分;无损恢复;嵌入容量
**中图法分类号**　TP391

## Reversible Data Hiding Scheme in NTRU Encrypted Domain Based on Polynomial Partition

LIU Dingcai, WU Haotian, ZHUANG Zhenwei and HE Junhui
School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China

**Abstract**　With the rapid development of cloud computing techniques and demand of privacy preservation, reversible data hiding (RDH) in homomorphic encrypted domain has become a hot research topic. Most of the existing RDH schemes in encrypted domain exploit correlations between adjacent pixels and redundancy in images, whose applications are limited. To improve applicability and embedding capacity, a new RDH scheme in NTRU encrypted domain based on polynomial partitioning is proposed. It divides the polynomial space in NTRU cryptosystem, which can be applied to multiple encrypted media content for data hiding. Part of the space is used to represent the original plaintext, while the rest space is used to hide the hidden data. The receiver can retrieve part of the hidden data directly from the ciphertext, while the rest hidden data can be extracted after decryption and the original plaintext can be correctly restored. In our experiments, grayscale images and text files are chosen to verify feasibility of the proposed scheme. Experimental results show that a maximum of $N$-8 bits can be hidden into a ciphertext for a plaintext represented with 8 bits, where $N$ is a parameter used in NTRU cryptosystem. When $N$ is set to 503, at most 495 bits can be hidden in a ciphertext while the plaintext can be exactly recovered. Compared with the existing schemes, the proposed scheme has higher embedding capacity and better applicability.

**Keywords**　Reversible data hiding, NTRU cryptosystem, Polynomial partition, Lossless recovery, Embedding capacity

## 1　引言

可逆数据隐藏技术[1]是实现数据隐私保护的一种方式,其利用载体信息的冗余性,设计算法将隐私数据嵌入到载体信息中,以实现隐私数据的嵌入和无损提取。早期可逆数据隐藏算法主要包括差值扩展(Difference Expansion)法[2]和直方图平移(Histogram Shifting)法[3]等。

随着隐私保护技术的发展,密文域可逆数据隐藏越来越受到关注。文献[4]首次将加密技术与数据隐藏技术相结合,设计了一种在加密域进行可逆数据隐藏的算法。随着研究的深入,有许多优秀方案逐渐被提出,根据加密算法类型主要可分为基于流加密[5-11]的方案和基于同态加密的方案[12-19]。

到稿日期:2022-08-26　返修日期:2022-12-11
基金项目:国家自然科学基金(61772208);广东省自然科学基金(2021A1515011798)
This work was supported by the National Natural Science Foundation of China(61772208) and Natural Science Foundation of Guangdong Province(2021A1515011798).
通信作者:吴昊天(wuht@scut.edu.cn)

随着云计算、云存储等技术的普及，利用非对称加密系统实现密文域可逆数据隐藏，并能够实现密文数据在被第三方系统处理后还能无损恢复原始数据，这是一个新的应用场景需求。于是，同态加密域的可逆数据隐藏研究课题应运而生。目前常见的同态加密算法有基于 LWE（Learning With Errors）[20] 的加密算法、Paillier 加密算法[21]、BGV 加密算法[22] 以及 NTRU 加密算法[23] 等。

由于图像具有良好的空间结构特性，学者们逐渐重点研究密文图像域可逆数据隐藏。Zhang 等[12] 针对同态加密图像提出了无损的、可逆的数据隐藏方案。在无损方案中，其采用 WPC（Wet Paper Coding）[24] 技术来替换密文像素的多个较低的位平面（Least Significant Bits），实现在图像解密前隐藏数据的提取，且不影响图像的解密。Xiang 等[13] 提出的方案首先对明文图像的像素进行分组，然后对所有分组相邻像素与参考像素像素值差的绝对值建立直方图，根据直方图平移法找到密文图像的数据嵌入位置，利用 Paillier 加密系统的同态性完成数据嵌入。该方案数据嵌入率最高为 0.5 bpp（bit per pixel），数据嵌入率较低。文献[14-15]与文献[13]的方法类似，同样对图像像素点进行分组来完成数据隐藏操作，但是都没有显著地提高密文域数据的嵌入率。其中文献[14]利用基于 R-LWE 问题的加密系统的浅同态加性进行数据隐藏，一个分组内有 3 个像素，因此最高数据嵌入率为 2/3 bpp；文献[15]利用 NTRU 加密系统的同态加性进行数据隐藏，一个分组内有 5 个像素，因此最高数据嵌入率为 4/5 bpp。Wu 等[16] 提出了两种在经过 Paillier 加密后的密文图像中进行数据隐藏的算法，第一种值扩展法利用了 Paillier 的同态加性隐藏数据；第二种算法利用 Paillier 加密系统的 self-blinding 性质隐藏数据，能实现解密前最大 14 bpp 的数据提取率，但是该算法的时间复杂度较高，整个数据隐藏过程耗时较长。Wu 等[17] 结合文献[16]提出了一种与同态处理兼容的 Paillier 密文域可逆数据隐藏算法，利用随机元素替换的方式来隐藏数据。Chen 等[18] 提出了两种利用加密系统同态性进行密文域数据隐藏的方法，利用 BGN[25] 算法的同态加性实现密文域可逆数据隐藏，利用 ELGamal[26] 的同态乘性实现密文域可逆数据隐藏，但是这两种方案只能实现在密文域嵌入和提取数据，且数据嵌入率较低。Zhou 等[19] 提出的方案首先利用差值扩展法进行预处理，之后在 NTRU 密文域将 1 比特数据隐藏到像素对中的一个像素中，因此该方案的最高数据嵌入率为 0.5 bpp，数据嵌入率较低。

综上可知，基于公钥加密系统的可逆数据隐藏已成为信息隐藏领域的研究热点，但现有的方案还存在着一些问题，如算法时间复杂度较高、方案的数据嵌入率过低和原始载体适用范围受限等问题。同时，随着量子计算机的发展，公钥加密系统的安全性也受到了严重挑战[27]，密文域可逆数据隐藏方案在未来能否抵抗量子计算攻击也是衡量方案安全性的一个重要标准。

为了弥补现有方案的不足，本文提出了一种基于多项式划分的 NTRU 域可逆数据隐藏方案。该方案利用运算简单但安全性高的 NTRU 加密算法对明文数据进行加密，利用 NTRU 加密系统的特性进行数据隐藏，能够对不同类型的多媒体载体加密后得到的密文数据进行数据隐藏，同时能提供不低于现有大多数同态加密域可逆数据隐藏方案的数据嵌入能力。

## 2 NTRU 加密系统及相关工作

### 2.1 NTRU 加密系统

NTRU 公钥加密系统于 1998 年被提出，其运算过程中只运用到简单的多项式模乘与加减运算，在安全性等级相同的前提下，相比目前现有的公钥加密系统如 RSA，McEliece，Paillier 等，其加解密速度更快，时间效率更高。除此之外，该加密系统的安全性规约于格上最短向量问题（Shortest Vector Problem，SVP）[28]，而格上最短向量问题不能被 Shor[29] 量子计算算法破解，因此 NTRU 加密系统能够抵抗量子计算攻击，属于抗量子密码，安全性更高。

NTRU 加密系统定义在多项式截断环 $R=\frac{Z[X]}{X^N-1}$ 上，在环上的多项式可以表示为 $f(x)=f_0+f_1\cdot x+\cdots+f_{N-1}\cdot x^{N-1}$，多项式 $f(x)$ 通常简写为 $f$，多项式 $f$ 可使用一个向量表示，即 $f=\sum_{i=0}^{N-1}f_i\cdot x^i=[f_0,f_1,\cdots,f_{N-1}]$。

#### 2.1.1 密钥生成及加解密流程

密钥生成及加解密流程的具体步骤如下。

(1)设置加密系统参数

NTRU 加密系统的参数由 $(N,p,q)$ 决定，其中 $q$ 远大于 $p$ 且 $p,q$ 互质。NTRU 加密系统参数的解释如下。

$N$：多项式截断环的维度，即多项式的最高阶，用于限定加密系统中的多项式的阶。

$p$：非 2 素数，用于加解密过程中的模操作以及限定明文空间 $R_p$ 的范围，常用值为 3。其中 $R_p$ 内多项式的系数均在 $\left[-\frac{p}{2},\frac{p}{2}\right]$ 之间，即明文多项式系数均需在 $\left[-\frac{p}{2},\frac{p}{2}\right]$ 之间。只有对明文空间 $R_p$ 中的多项式进行加密，对应的密文才能被正确解密。

$q$：较大模数，值为 2 的幂次方，用于加解密过程中的模操作。

(2)密钥对生成

随机生成多项式 $f$ 和多项式 $g$，保证 $f$ 模 $p$ 的逆元和 $f$ 模 $q$ 的逆元都存在，否则重新生成 $f$，然后计算 $f$ 模 $p$ 的逆元 $F_p$ 和 $f$ 模 $q$ 的逆元 $F_q$，即 $f\times F_p\equiv 1 \bmod p$，$f\times F_q\equiv 1 \bmod q$，最后计算多项式 $h=p\cdot F_q\times g(\bmod q)$（其中，$\cdot$ 为整数与多项式之间的乘法运算符，$\times$ 为多项式之间的乘法运算符），从而得到公钥 $K_{\text{pub}}=h$，私钥 $K_{\text{pri}}=(f,F_p)$。

(3)加密

给定明文多媒体数据 $data$，将其转换成二进制数据，然后构造一多项式 $m$ 用于表示该二进制数据，$m$ 的系数为二进制数据流对应比特位，由于 $m$ 的系数均在 $\left[-\frac{p}{2},\frac{p}{2}\right]$ 之间，因此多项式属于明文空间 $R_p$。随机生成多项式 $r$，利用公钥 $h$ 根据式(1)对 $m$ 加密得到密文多项式 $e$。

$$e=Enc(m)=r\times h+m(\bmod q) \tag{1}$$

(4)解密和恢复原始数据

对于密文多项式 $e$,利用私钥$(f,F_p)$根据式(2)解密得到明文多项式 $m$。

$$\begin{aligned}Dec(e)&=F_p\times[f\times e(\bmod q)](\bmod p)\\&=F_p\times f\times[r\times h+m(\bmod q)](\bmod p)\\&=F_p\times f\times[r\times(p\cdot F_q\times g)+m(\bmod q)](\bmod p)\\&=p\cdot F_p\times r\times g+F_p\times f\times m(\bmod p)\\&=m\end{aligned}\tag{2}$$

2.1.2 同态性质

对于明文空间 $R_p$ 任意的两个明文多项式 $m_1$ 和 $m_2$,选择两个随机的多项式 $r_1$ 和 $r_2$,经过加密之后对应的密文为 $e_1=r_1\times h+m_1$ 和 $e_2=r_2\times h+m_2$。

同态加性的表达式如下:

$$\begin{cases}e_1+e_2=(r_1+r_2)\times h+m_1+m_2(\bmod q)\\Dec(e_1+e_2)=m_1+m_2=Dec(e_1)+Dec(e_2)(\bmod p)\end{cases}\tag{3}$$

### 2.2 NTRU 加密域可逆数据隐藏研究工作

NTRU 加密系统具有同态性,因此可以利用其同态性进行密文域可逆数据隐藏。下面介绍现有的 NTRU 同态加密域的可逆数据隐藏的相关研究工作。

文献[15]通过密文数据与待隐藏数据相加来实现数据隐藏,利用直方图平移法进行预处理与可逆数据恢复。该方案首先对图像进行预处理,将明文图像按一组 $T$ 个像素分成多个像素组,每个组内标记一个参考像素,其余为相邻像素。然后对图像加密,按与明文图像同样的分组方式对密文图像分组得到密文像素组,计算组内每个相邻密文像素与参考密文像素的差值,对差值建立直方图。最后结合直方图平移法,对分组内差值符合条件的相邻密文像素隐藏 1 比特数据(其等价于在明文像素组某个相邻像素隐藏 1 比特数据)。解密后,对得到的含有隐藏信息的明文图像继续按上文同样的方法分组,根据直方图平移法原理从分组内相邻像素点提取隐藏数据和还原原始明文像素值。该方案根据 $T$ 值的不同,数据隐藏能力也不同,当 $T$ 取 5 时,理论上组内 4 个像素点都可以用于隐藏数据,因此方案的嵌入率最高为 0.8 bpp。

文献[19]利用 NTRU 加密系统的同态加性进行数据隐藏,利用差值扩展法进行预处理与可逆数据恢复。在预处理阶段,利用差值扩展法对明文像素对进行差值扩展,然后利用 NTRU 加密系统对图像中每一个明文像素对加密得到密文像素对。在数据嵌入阶段,首先将 1 比特的待隐藏数据加密得到密文数据,利用 NTRU 加密系统的同态加性将隐藏数据对应的密文与密文像素对中的其中一个密文相加来完成数据隐藏,其效果等价于对明文像素对隐藏 1 比特数据。解密后从一个明文像素对中提取出隐藏的 1 比特数据。该方案利用了差值扩展法进行数据隐藏,加密一个明文像素对能隐藏 1 比特数据,因此方案的平均嵌入率为 0.5 bpp。该数据隐藏方案需要进行两阶段加密,分别对明文加密和对需要隐藏的数据加密。

上述方案在数据嵌入率方面与现有其他加密域可逆数据隐藏方案相比没有很大提升,而 NTRU 加密系统加密后具有一定的数据扩张,密文中存在较多的冗余空间,如果能利用多项式的冗余空间进行数据隐藏,将能大大提高 NTRU 加密域的数据嵌入率。

## 3 本文方案

本节首先提出了一种对 NTRU 加密系统中的多项式系数进行划分的方案,然后基于该划分方案提出了两种 NTRU 域可逆数据隐藏算法,最后结合这两种算法提出了本文方案——基于多项式划分的 NTRU 域可逆数据隐藏方案。

### 3.1 NTRU 加密系统多项式系数划分方案

NTRU 加密系统中的基本运算对象是多项式,明文和密文都是以多项式的形式存在,且多项式的阶相同。对于加密系统涉及的所有多项式,本节提出了一种通用的多项式系数划分方案,如图 1 所示(以多项式 $S=[s_0,s_1,\cdots,s_{N-1}]$ 系数划分为例,假设明文段长度为 $k$)。多项式的一部分低阶系数称为明文段,多项式中的剩余系数称为数据隐藏段。

对于明文多项式,仅用明文段表示明文数据,对于密文多项式,仅在数据隐藏段隐藏数据。明文段和数据隐藏段的实际长度应根据实际应用场景,由数据隐藏者和接收者共同约定。

### 3.2 多项式划分在 NTRU 域可逆数据隐藏的应用

假如有一个明文空间 $R_p$ 中的明文多项式 $m$(其多项式系数都在 $\left[-\frac{p}{2},\frac{p}{2}\right]$ 之间),对 $m$ 经式(1)加密得到密文多项式 $e$,$e=Enc(m)=r\times h+m$。对 $e$ 加上多项式 $u$,得到 $e'=e+u=r\times h+m+u$,若$(m+u)$的所有系数也在$[-\frac{p}{2},\frac{p}{2}]$之间,则$(m+u)$也在明文空间 $R_p$ 中,有 $Enc(m+u)=r\times h+m+u=e'$,$Dec(e')=m+u$。假设 $u$ 携带有隐私数据,对 $e$ 加 $u$ 完成隐私数据嵌入得到密文 $e'$,对 $e'$解密得到多项式$(m+u)$,若能从多项式$(m+u)$中分离出 $m$ 和 $u$,就得到了密文域隐藏的数据 $u$ 和明文多项式 $m$。

那么如何从多项式$(m+u)$中分离 $m$ 和 $u$ 呢?多项式划分能解决这个问题。按照图 1 所示的方案对多项式进行划分后,多项式的明文段用于表示明文数据,数据隐藏段系数均为 0。

基于图 1 所示的多项式划分方案,提出了两种 NTRU 域可逆数据隐藏算法,分别为:1)解密后提取数据的 NTRU 域可逆数据隐藏算法;2)解密前提取数据的 NTRU 域可逆数据隐藏算法。

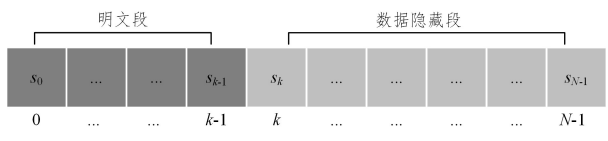



图 1 对多项式 $S$ 系数的划分示意图

Fig. 1 Schematic diagram of partitioning coefficients of polynomial $S$

3.2.1 解密后提取数据的 NTRU 域可逆数据隐藏算法

本算法能够在密文域实现嵌入,并在解密域实现提取和无损恢复原始明文。下文介绍基于图 1 所示的多项式划分

方案，在解密前提取数据的 NTRU 域可逆数据隐藏算法的执行流程。

(1)明文数据加密。假设明文段长度为 $k$，定义一个属于明文空间 $R_p$ 的明文多项式 $m=[m_0,\cdots,m_{k-1},0,\cdots,0]$（$m_i$ 为 0 或 1，在 $\left[-\frac{p}{2},\frac{p}{2}\right]$ 之间），经式(1)加密得到密文多项式 $e=r\times h+m$，密文 $e$ 用向量表示为：$e=[e_0,\cdots,e_{N-1}]$。

(2)数据隐藏。构造多项式 $u=[\overbrace{0,\cdots,0}^{k个0},a_0,\cdots,a_{N-1-k}]$，其中 $u$ 中数据隐藏段系数 $(a_0,\cdots,a_{N-1-k})B$ 为待隐藏的 $N-k$ 比特数据，最后根据式(4)，对密文 $e$ 加上 $u$，将携有隐私数据的多项式 $u$ 加到密文多项式 $e$ 中，得 $e'=r\times h+m+u=[e_0,\cdots,e_{k-1},e_k+a_0,\cdots,e_{N-1}+a_{N-1-k}]$，此时 $e'$ 中已含有隐藏的数据。

$$e'=e+u=r\times h+m+u \tag{4}$$

(3)数据提取和明文恢复。由于 $(m+u)$ 属于明文空间 $R_p$，因此对 $e'$ 解密得到 $(m+u)$，$(m+u)=[m_0,\cdots,m_k,a_0,\cdots,a_{N-1-k}]$。分离 $(m+u)$ 明文段系数作为 $m$ 的明文段系数，$m$ 的数据隐藏段系数全为 0，得到多项式 $m=[m_0,\cdots,m_{k-1},0,\cdots,0]$，即无损恢复了明文；分离 $(m+u)$ 数据隐藏段系数作为 $u$ 的数据隐藏段系数，从而得到 $u=[\overbrace{0,\cdots,0}^{k个0},a_0,\cdots,a_{t-1}]$，进而从 $u$ 中数据隐藏段系数中提取出的隐藏的 $N-k$ 比特隐私数据。

### 3.2.2 解密前提取数据的 NTRU 域隐藏算法

3.2.1 节的算法可以实现在解密后提取数据嵌入者在密文域嵌入的数据，但是在一些场景下，需要在密文数据中隐藏一些让没有解密密钥的人也能获取到的信息。

对 3.2.1 节中的算法稍作改造，可以得到解密前提取数据的算法。该算法的具体流程如下：

(1)明文数据加密。同理，对于明文空间 $R_p$ 内的明文多项式 $m=[m_0,\cdots,m_{k-1},0,\cdots,0]$，经式(1)加密得到密文多项式 $e=[e_0,\cdots,e_{N-1}]$。

(2)数据隐藏。改变密文多项式数据隐藏段中的系数得到新系数，用新的系数模 2 的结果来表示一个系数隐藏的数据。对于待隐藏数据 $(b_0b_1\cdots b_{N-1-k})B$，构造多项式 $v=[\overbrace{0,\cdots,0}^{k个0},w_0,\cdots,w_{N-1-k}]$，$v$ 的明文段系数设为 0，数据隐藏段系数根据式(5)获得，最后根据式(4)，对密文 $e$ 加上 $v$ 完成数据隐藏，得 $e'=r\times h+m+v$，其用向量表示为 $e'=[e_0,\cdots,e_{k-1},e_k+a_0,\cdots,e_{N-1}+a_{N-1-k}]$。

$$w_j=\begin{cases}0, & e_{k+j} \bmod 2=b_j\\ 1, & e_{k+j} \bmod 2\neq b_j\end{cases},j\in[0,N-k) \tag{5}$$

(3)解密前的数据提取。对于密文 $e'$，根据式(6)提取 $e'$ 中数据隐藏段系数中隐藏的数据，得到密文域中隐藏的数据 $(b_0b_1\cdots b_{N-1-k})B$。

$$b_j=e'_{k+j} \bmod 2,j\in[0,N-k) \tag{6}$$

### 3.3 基于多项式划分的可逆数据隐藏方案

本节结合 3.2 节的两种 NTRU 密文域可逆数据隐藏算法，提出基于多项式划分的 NTRU 域可逆数据隐藏方案，该方案的流程如图 2 所示。

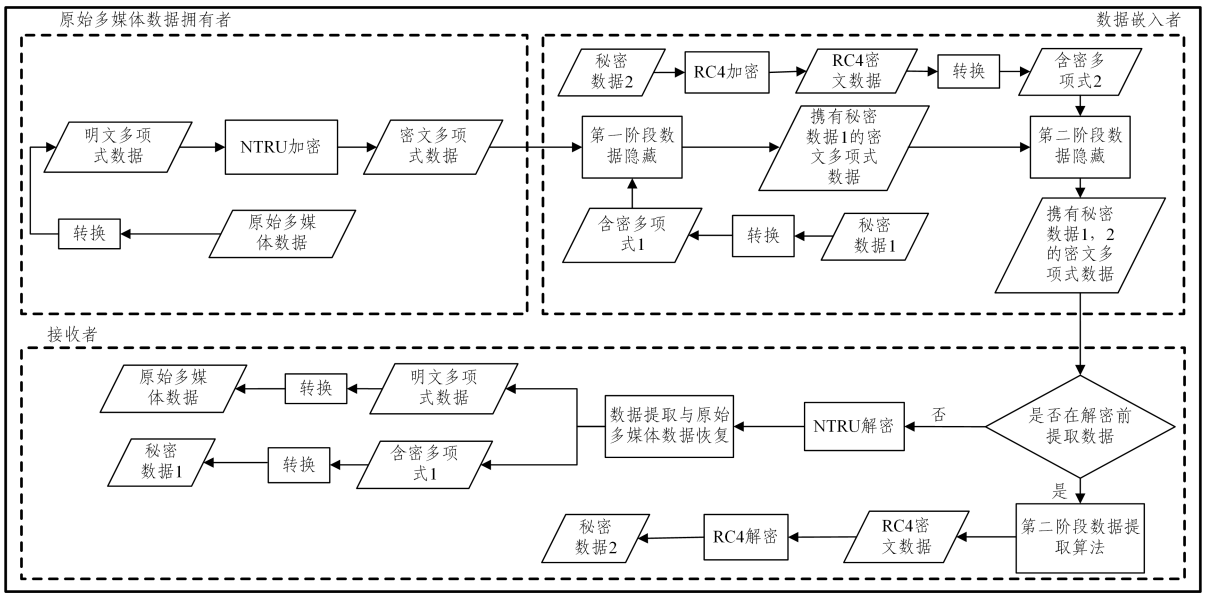



图 2 本文提出的 NTRU 加密域可逆数据隐藏方案流程图

Fig. 2 Flowchart of the proposed scheme for reversible data hiding in NTRU encryption domain

该方案的整体流程可分为 3 个部分：多媒体数据加密、数据隐藏、隐藏数据提取与多媒体数据恢复。

为了实现两个阶段的数据隐藏，本文在 3.1 节提出的多项式系数划分方案的基础上，将数据隐藏段进一步划分为数据隐藏段 1 和数据隐藏段 2(长度分别为 $t$ 和 $s$)，分别用于第一阶段和第二阶段数据隐藏，其中第一阶段数据隐藏使用 3.2.1 节中的算法进行数据隐藏，第二阶段数据隐藏使用 3.2.2 节中的算法进行数据隐藏。

为了方便介绍本文可逆数据隐藏方案的详细流程，本节以灰度图 $I$ 作为原始载体数据进行密文可逆数据隐藏。由于灰度图一个像素值仅需 8 比特表示，因此在多项式划分中，明文段长度设置为 8。本文所提方案的多项式划分示意图如

图 3 所示(以多项式 $G=[g_0,\cdots,g_{7+s+t}]$为例)。

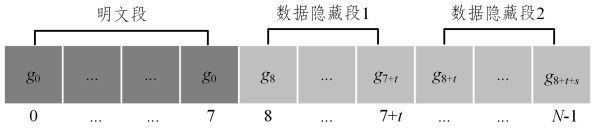



图 3 对多项式 $G$ 系数划分示意图

Fig. 3 Schematic diagram of partitioning coefficients of polynomial $G$

对于本文数据隐藏方案执行流程中涉及的多项式都按图 3 所示的划分方案进行划分,但是不同类型多项式中的不同系数段有不同作用。对于明文多项式,仅用明文段表示原始明文多媒体数据,数据隐藏段 1 和数据隐藏段 2 系数均为 0;对于密文多项式,在数据隐藏段 1 嵌入第一阶段待隐藏的数据,在数据隐藏段 2 隐藏第二阶段待隐藏的数据,明文段不做处理。

下文将对数据隐藏方案流程中的每部分具体实现进行详细介绍,其中对某个明文像素 $P$ 对应的明文多项式 $m$ 进行密文域数据隐藏和数据提取的流程示意图如图 4 所示。

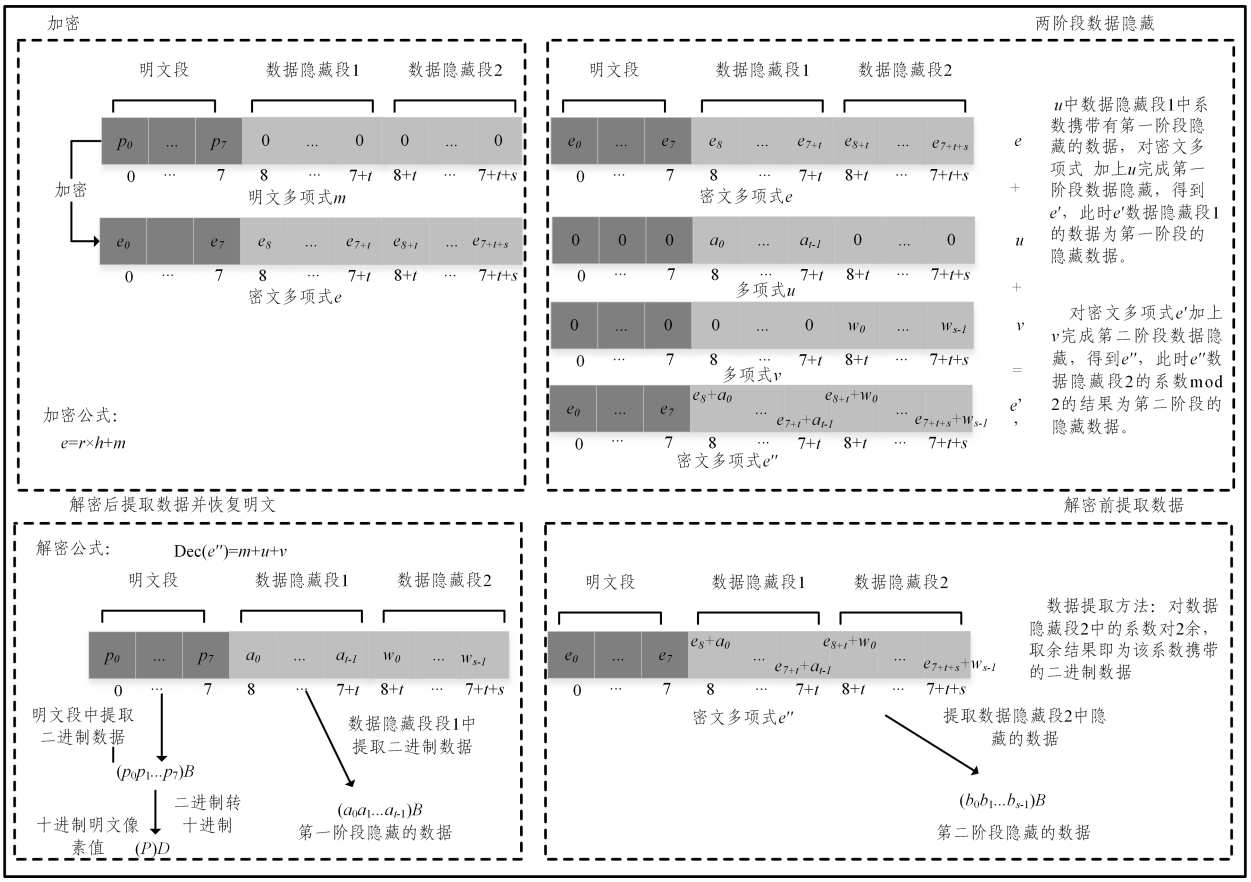



图 4 对明文多项式 $m$ 密文域进行数据隐藏和数据提取的示意图

Fig. 4 Schematic diagram of data hiding and data extraction in cipher domain of plain-text polynomial

### 3.3.1 多媒体数据加密

首先设置 NTRU 加密系统的参数$(N,p,q)$,使用密钥生成技术生成公钥 $h$ 和私钥$(f,F_p)$。设置参数$(N,p,q)=(503,3,256)$,多媒体数据所有者执行下列步骤:

(1)将图像 $I$ 的每一个十进制像素值(假设为 $P$)都转换成二进制的形式,有$(P)D=(p_0p_1\cdots p_7)B(P\in[0,255],p_j$ 是二进制 0 或 $1,j\in[0,8))$。构造对应的明文多项式 $m=[p_0,\cdots,p_7,0,\cdots,0]$(此时 $p_j$ 是十进制 0 或 $1,j\in[0,8))$,$m$ 系数值均为 0 或 1,因此多项式 $m$ 在明文空间 $R_p$ 中。

(2)根据式(1)对明文多项式 $m$ 进行加密,得到对应的密文多项式 $e=r\times h+m(\mathrm{mod}\ q)$。将每个明文像素加密完后,此时得到的密文数据为 $E(I)$,将密文数据上传至云服务器等第三方平台。

### 3.3.2 数据隐藏

数据隐藏者执行下列步骤:

(1)设置数据 $t$ 和 $s$,保证$(t+s)\leqslant(N-8)$。设第一阶段需要隐藏的二进制数据为 $D_A$,其数据长度为 $L_1$,第二阶段需要隐藏的二进制数据为 $D_B$,其数据长度为 $L_2$。

(2)第一阶段数据隐藏。对于待隐藏二进制数据 $D_A$,将其分割成长度为 $t$ 的多段数据,最后一段数据长度可小于 $t$,再按序将这多段数据隐藏在不同密文多项式中。以对一个密文多项式 $e$ 隐藏 $D_A$ 中 $t$ 比特的二进制数据为例,首先需将该段二进制数据用多项式表示,设该段数据为$(a_0a_1\cdots a_{t-1})B$,参照 3.2.1 节中的算法,构造一个用于数据隐藏的多项式 $u$,其向量表示为 $u=[0,\cdots,0,a_0,\cdots,a_{t-1},0,\cdots,0]$,其中 $u$ 的明文段的系数和数据隐藏段 2 的系数均设为 0,数据隐藏段 1 的系数为第一阶段隐藏的数据。对每个密文 $e$ 加上 $u$ 以隐藏 $D_A$ 中一段长度为 $t$ 的数据 $a_0a_1\cdots a_{t-1}$,得到新的密文多项式 $e'=e+u$。对 $D_A$ 中每一段数据重复上述操作,直到二进制数据 $D_A$ 隐藏完毕,完成第一阶段数据隐藏。

(3)第二阶段数据隐藏。对于待隐藏数据 $D_B$，使用 RC4 加密算法(数据嵌入者与数据提取者拥有相同密钥，密钥如何共享在此不做过多赘述)对 $D_B$ 加密得到同样长度的密文数据 $D_B^e$，然后 $D_B^e$ 首先将其分割成二进制位长度为 $s$ 的 $n$ 段数据，最后一段数据长度可小于 $s$。参照 3.2.2 节的数据隐藏算法，将每一段流加密密文数据隐藏到 $e'$ 的数据隐藏段 2 中。

以对一个密文多项式隐藏 $D_B^e$ 中某段 $s$ 比特的二进制数据为例，设该段数据为 $(b_0 b_1 \cdots b_{s-1})B$，构造一个多项式 $v$，$v=[\overbrace{0,\cdots,0}^{(8+t)\text{个}0},w_0,\cdots,w_{s-1}]$，多项式 $v$ 中明文段和数据隐藏段 1 的系数全为 0，数据隐藏段 2 的系数 $w_0,\cdots,w_{s-1}$ 根据式(5)得出；对完成第一阶段数据隐藏得到的密文多项式 $e'$ 加上 $v$ 得到新的密文多项式 $e''$，使 $e''$ 的数据隐藏段 2 的系数满足 $e''_{8+t+j} \bmod 2=b_j(j\in[0,s))$，以此在数据隐藏段 2 中隐藏 $D_B^e$ 中一段 $s$ 比特的二进制数据 $(b_0 b_1 \cdots b_{s-1})B$。对 $D_B^e$ 中每一段数据重复上述操作，直到 $D_B^e$ 隐藏完毕，完成第二阶段数据隐藏，这阶段隐藏的数据可以在密文解密前根据式(6)提取隐藏数据。

(4)最后隐藏 $t$ 和 $s$ 以及隐藏数据的相关信息。对于最后一个像素的密文多项式，在数据隐藏段的前 80 个系数隐藏 $t$ 和 $s$、两阶段隐藏数据总数据长度 $L_1$ 和 $L_2$ 及两阶段隐藏数据数据类型 $T_1$ 和 $T_2$。数据隐藏者和接收方共同约定将 $t$ 与 $s$ 分别用 8 比特的二进制数来表示，$L_1$ 和 $L_2$ 分别用 20 比特二进制数来表示，两阶段隐藏数据的多媒体数据类型相关信息 $T_1$ 和 $T_2$ 分别用 12 比特二进制数表示。为了保证这部分信息不被泄露，同样使用 RC4 流加密算法对 $t,s,L_1,L_2,T_1,T_2$ 进行加密，分别得到 $t^e,s^e,L_1^e,L_2^e,T_1^e,T_2^e$。最后参考步骤(3)，按照第二阶段数据隐藏方案将这 80 比特数据隐藏到最后一个像素的密文多项式数据隐藏段的前 80 个系数中，其中约定数据隐藏段的第 1－8 个系数隐藏 $t^e$，第 9－16 个系数隐藏 $s^e$，第 17－36 个系数隐藏 $L_1^e$，第 37－56 个系数隐藏 $L_2^e$，第 57－68 个系数隐藏 $T_1^e$，第 69－80 个系数隐藏 $T_2^e$；这部分隐藏的数据可以在密文数据解密前进行提取。经过上述 4 个步骤，得到含有隐藏数据的密文数据 $E'(I)$。

### 3.3.3 隐藏数据提取与多媒体数据恢复

接收方获取到隐藏有秘密数据的密文数据 $E'(I)$，从 $E'(I)$ 中得到携有两阶段隐藏数据的所有密文多项式 $e''=r\times h+m_i+u+v$。从 $e''$ 中提取两阶段隐藏的数据执行步骤如下：

(1)对于最后一个密文多项式数据隐藏段的前 80 个系数，提取出隐藏的相关数据对应的流加密数据，将这些数据经 RC4 算法解密。具体的提取过程为：对最后一个密文多项式的数据隐藏段前 80 个系数模 2，得到 80 比特二进制数据，提取第 1－8 个、第 9－16 个、第 17－36 个、第 37－56 个，第 57－68 个以及第 69－80 个比特，分别得到 $t^e,s^e,L_1^e,L_2^e,T_1^e,T_2^e$，经 *RC*4 解密分别得到 $t,s,L_1,L_2,T_1,T_2$。

(2)对于其余密文多项式 $e''$，在解密之前，按照密文多项式顺序，对每个密文多项式的数据隐藏段 2 的 $s$ 个系数通过式(6)提取出一段长度为 $s$ 的二进制数据，按先后顺序拼接提取出的数据，当提取数据量达到 $L_2$ 时可得到在第二阶段所有隐藏的数据 $D_B^e$，当数据提取完毕，对剩余密文多项式不进行数据提取操作。对 $D_B^e$ 进行 RC4 解密得到第二阶段隐藏的数据 $D_B$。

(3)若有私钥，利用解密式(2)对收到的密文多项式 $e''$ 解密。由于多项式 $(m+u+v)$ 的系数均为 0 或 1，都在 $\left[-\frac{p}{2},\frac{p}{2}\right]$ 之间，即 $(m+u+v)$ 在明文空间 $R_p$ 中，因此密文多项式 $e''$ 等同于对明文多项式 $(m+u+v)$ 加密后的结果，对 $e''$ 解密得到 $(m+u+v)$。

(4)从解密后得到的多项式 $(m+u+v)$ 中提取隐藏数据和恢复原始明文。提取该多项式明文段系数得到二进制数据 $(p_0 p_1 \cdots p_7)B$，进而恢复原始明文十进制像素值 $P$，将每一个像素还原后，即可以恢复原始的明文图像；按序提取该多项式数据隐藏段 1 共 $t$ 个系数得到一段长度为 $t$ 的二进制数据 $(a_0 a_1 \cdots a_{t-1})B$，按先后顺序拼接提取出的数据，当提取数据量达到 $L_1$ 时即可恢复出第一阶段所有隐藏的数据 $D_A$，之后对剩余密文多项式不进行数据提取操作。

## 4 实验结果与分析

实验选取如图 5 所示的 5 幅尺寸为 512 * 512 的灰度图和文本作为测试数据，首先对方案的可行性进行了验证，然后对方案的综合性能进行了分析与对比。

(a) Lena (b) Man (c) Boat

(d) Baboon (e) Barbara

图 5 5 幅测试图像

Fig. 5 Five test images

在实验中，按照文献[23]推荐的最高安全等级参数对 NTRU 加密系统的参数进行设置：$N=503$，$q=256$，$p=3$。

### 4.1 方案可行性验证

图 6 给出了以 Lena 图像为载体，以 Man 图像和 Boat 图像为两阶段隐藏数据的实验结果。其中，图 6(a)是原始载体图像，图 6(f)是接收方对收到的密文解密后得到的图像，两者之间的 $PSNR=\infty$，即解密后恢复的图像与原始图像是完全一致的；图 6(b)是第一阶段隐藏的图像数据，图 6(e)是对收到的密文解密后提取的图像，两者之间的 $PSNR=\infty$；图 6(c)是第二阶段隐藏的明文图像数据，实验中嵌入的是采用 RC4 算法对

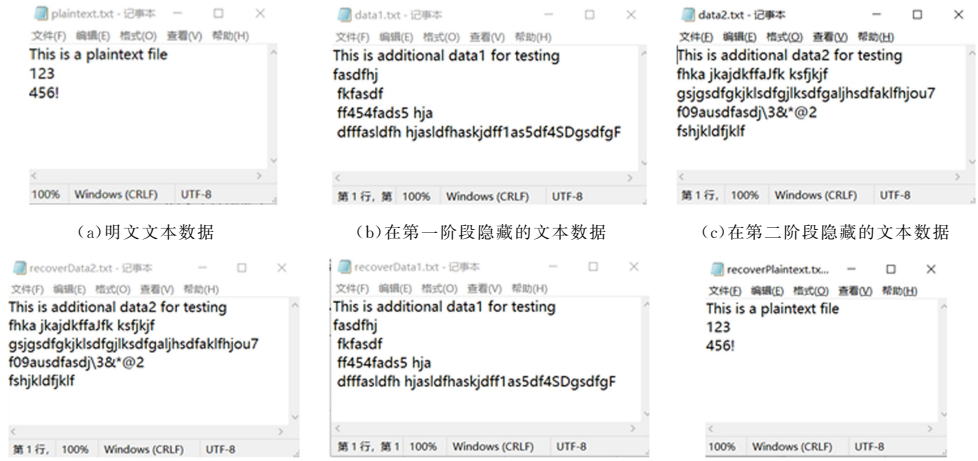
该数据加密得到的密文数据，图 6(d)是对密文解密前提取并用 RC4 解密后还原得到的图像，两者之间的 $PSNR=\infty$。

(a)原始图像 (b)在第一阶段隐藏的图像 (c)在第二阶段隐藏的图像 (d)在解密前提取的图像 (e)在解密后提取的图像 (f)解密后恢复的图像

图 6 以 Lena 图像为载体得到的实验结果

Fig. 6 Experimental results obtained by using the "Lena" image as carrier

图 7 给出了以文本数据为原始载体的实验结果，验证了本文方案能适用于不同类型的多媒体数据。上述实验结果表明，本文提出的密文域数据隐藏方案能够在不同类型的加密载体数据中隐藏数据，并能无损提取隐藏的数据和恢复原始明文载体数据，证明了本文所提数据隐藏方案的可行性和对不同类型载体数据的适用性。




(a)明文文本数据 (b)在第一阶段隐藏的文本数据 (c)在第二阶段隐藏的文本数据

(d)解密前提取的隐藏数据 (e)解密后提取的隐藏数据 (f)解密后恢复的明文数据

Fig. 7 Experimental results obtained by using a text file as carrier

图 7 以文本数据为原始载体得到的实验结果

### 4.2 数据隐藏能力与数据扩张率分析

在之前的可逆数据隐藏工作中，使用 $ER$ 来衡量方案的数据隐藏能力，其中 $ER$ 的计算式如式(7)所示：

$$ER=\frac{\text{嵌入的总比特数(bit)}}{\text{明文像素个数}}(bpp) \tag{7}$$

在同态加密域中进行可逆数据隐藏时，由于不同同态加密算法加密过程造成的数据扩张程度不一样，扩张后的密文空间大小也不一样，密文空间越大就会有越多的冗余空间，能够有更多的空间隐藏数据。为了更好地衡量同态加密域中密文的实际数据隐藏率，本文使用式(8)定义的密文域平均嵌入率 $ER_{\text{avg}}$ 来度量加密域数据隐藏方案在密文域中的数据隐藏能力。

$$ER_{\text{avg}}=\frac{\text{嵌入的总比特数}}{\text{密文总比特数}} \tag{8}$$

在灰度图中，1 像素需要 8 比特表示，因此在数值上 $ER$ 与 $ER_{\text{avg}}$ 之间的关系如式(9)所示，其中数据扩张率＝密文数据大小/明文数据大小。

$$\begin{cases} ER=\dfrac{\text{嵌入的总比特数}}{\text{明文像素个数}}=\dfrac{\text{嵌入的总比特数}}{\text{明文字节数}} \\ ER_{\text{avg}}=\dfrac{\text{嵌入的总比特数}}{8\times\text{明文字节数}\times\text{数据扩张率}}=\dfrac{ER}{8\times\text{数据扩张率}} \end{cases} \tag{9}$$

#### 4.2.1 数据扩张率分析

NTRU 加密系统多项式系数的个数为 $N$，当选择灰度图像作为载体数据时，灰度图中的一个像素值由 1 字节(8 比特)表示。用多项式表示像素值时，设 $a_0a_1\cdots a_7$ 是该像素值的二进制表示，其中 $a_i$ 是二进制 0 或 1，则需要构建一个明文多项式 $m$，$m=[a_0,\cdots a_7,0,\cdots,0]$，此时系数 $a_i$ 为十进制值且 $a_i\in[-p/2,p/2]$。对明文多项式加密得到密文多项式 $e=[e_0,\cdots e_i,\cdots,e_{N-1}]$，其中 $e_i\in[-q/2,q/2]$。方案中参数 $q$ 取 256，因此密文多项式每一个系数大小为 1 字节，密文多项式大小为 $N$ 字节，而明文灰度图一个像素值大小为 1 字节，因此 NTRU 同态加密系统的数据扩张率为 $N$。

#### 4.2.2 数据隐藏能力分析

由于明文像素值大小为 1 字节(8 比特)，因此多项式的明文段长度为 8，数据隐藏段长度为$(N-8)$，一个明文对应的密文多项式最多可隐藏$(N-8)$比特数据，即 $ER=N-8$，密文域平均数据嵌入率 $ER_{\text{avg}}=\dfrac{ER}{8\times\text{数据扩张率}}=\dfrac{N-8}{8N}$。随着 $N$ 越大，$ER_{\text{avg}}$ 越接近 $\dfrac{1}{8}$，即在 1 字节(8 比特)密文数据中就可以接近隐藏 1 比特数据。

在实际应用时，如何选取参数 $N$ 呢？考虑密文域平均

数据嵌入率的情况下，选取 $N$ 可以取决于用户想要的 NTRU 加密系统的安全程度。当 $N$ 取 107 时，加密系统的安全性为中等，密文域平均数据嵌入率 $ER_{avg}$ 为(107－8)/(8＊107)≈0.116；当 $N$ 取 167 时，加密系统为高安全性，$ER_{avg}$ 为(167－8)/(8＊167)≈0.119；当 $N$ 取 503 时，加密系统具有最高安全性，$ER_{avg}$ 为(503－8)/(8＊503)≈0.123。可见，选取不同 $N$ 值虽然具有不同的数据扩张程度，但是密文域平均数据嵌入率均大于 0.9/8，因此衡量 $N$ 的选取首先应考虑的是加密系统的安全性。本文建议 $N$ 取 167，当 $N$ 取 167 时 NTRU 加密系统具有较高的安全性，同时加密系统数据的扩张程度也在可接受的范围内。

### 4.3 与同类方案的综合性能对比

#### 4.3.1 平均嵌入率对比

表 1 列出了本文方案与文献[12，15-16，18-19]中的方案进行数据率 $ER$ 对比的结果；表 2 列出了本文方案与上述方案进行综合性能对比的结果。根据实验结果可知，本文方案最高可以隐藏 495 比特的数据，比现有的大部分同类方案的数据隐藏能力强，表明了本文方案在数据隐藏能力方面相比对比方法的优越性。

表 1 本文方案与部分现有的方案数据嵌入率($ER$)的对比

Table 1 Data embedding rate comparison between the proposed scheme and previous schemes

| 方案 | $ER$/bpp | | 加密系统 |
|---|---|---|---|
| | 加密域提取 | 解密域提取 | |
| 文献[12] | 接近 1 | 0.5 | Paillier |
| 文献[15] | 0.8 | 0.8 | NTRU |
| 文献[16] | 14.0 | 1014 | Paillier |
| 共(14＋1014) | | | |
| BGN[18] | 1 | 0 | BGN |
| ELGamal[18] | 0.5 | 0 | ELGamal |
| [19] | 0.5 | 0.5 | NTRU |
| 本文方案 | $t$ | $s$ | NTRU |

表 2 本文方案与部分现有方案的综合性能对比

Table 2 Comprehensive performance comparison between the proposed scheme and previous schemes

| 方案 | 加密系统及参数 | 数据扩张率 | $ER_{avg}$ | | 是否仅限于图像域 |
|---|---|---|---|---|---|
| | | | 加密域提取 | 解密域提取 | |
| 文献[12] | Paillier：参数 $p$，$q$ 为大素数 | $\frac{\lceil \log_2^{p*q} \rceil}{4}$ | $\frac{1}{2*\lceil \log_2^{p*q} \rceil}$ | $\frac{1}{4*\lceil \log_2^{p*q} \rceil}$ | 是 |
| 文献[15] | NTRU：参数 $N$，可取 107，167，503 等素数 | $N$ | $\frac{1}{10N}$ | $\frac{1}{10N}$ | 是 |
| 文献[16] | Paillier：参数 $p$，$q$ 为大素数，$N$ 为表示 $p*q$ 需要的二进制比特数，方案中 $N$ 为 2048 | 256 | $\frac{7}{1024}$ | $\frac{507}{1024}$ | 否 |
| BGN[18] | BGN：参数 $q_1$，$q_2$ 是大整数 | $\frac{\lceil \log_2^{q_1*q_2} \rceil}{8}$ | $\frac{1}{\lceil \log_2^{q_1*q_2} \rceil}$ | 0 | 否 |
| ELGamal[18] | ELGamal：参数 $p$ 为大素数 | $\frac{\lceil \log_2^{p} \rceil}{4}$ | $\frac{1}{4*\lceil \log_2^{p} \rceil}$ | 0 | 否 |
| 文献[19] | NTRU：参数 $N$，可取 107，167，503 等 | $N$ | $\frac{1}{16N}$ | $\frac{1}{16N}$ | 是 |
| 本文方案 | NTRU：参数 $N$，可取 107，167，503 等 | $N$ | $\frac{t}{8N}$ | $\frac{s}{8N}$ | 否 |
| | | | 最大 $\frac{N-8}{8N}(t+s\leqslant N-8)$ | | |

由表 2 可知，本文方案平均在 1 比特密文数据中可以隐藏将近 1/8 比特数据，密文域平均嵌入率约为 1/8，除了文献[16]的方案，现有的其他方案平均嵌入率都远远小于 1/8。虽然 NTRU 加密系统数据扩张严重，但是本文方案可以充分利用密文数据冗余空间进行高容量的数据隐藏；虽然本文方案在解密域的数据提取率比文献[16]的方案低，但是加密域的数据嵌入率远比文献[16]的方案高，且文献[16]的数据隐藏方案时间复杂度高，算法执行时间较长。

#### 4.3.2 方案的适用性分析与对比

文献[12，15，19]的数据隐藏方案都利用了图像的空间结构特性，只能适用于原始载体数据为图像的场景，文献[16，18]和本文方案利用了加密系统的特性进行数据隐藏，数据隐藏方法与原始载体类型无关，因此可以对不同类型的多媒体数据进行密文域可逆数据隐藏。

在本文方案中，使用 NTRU 加密系统进行加解密，NTRU 加密系统的运算对象是多项式，明文和密文都需要用多项式表示。对于不同类型的多媒体数据，其二进制数据可用统一的数据格式(多项式)表示，因此 NTRU 加密系统可以加密不同类型的数据，提高了隐藏方案的适用性。

图 7 给出了以文本数据为载体数据的实验结果，其中图 7(a)是明文文本数据，图 7(b)是第一阶段隐藏的文本数据，图 7(c)是需要在第二阶段隐藏的明文文本数据，实验中采用 RC4 算法对该文本数据加密，然后将加密数据嵌入到加密载体。图 7(d)是对密文解密前提取并用 RC4 解密后还原得到的文本数据；图 7(e)是对密文解密后提取的文本数据；图 7(f)是解密后恢复的明文数据，与原始的明文文本数据完全相同。实验结果显示，在两个阶段隐藏的文本数据均能被正确提取。

#### 4.3.3 数据隐藏时间分析对比

NTRU 加解密算法只使用到简单的模乘、模逆与加减运算，运算量较小。表 3 列出了 NTRU 与常用的公钥加密算法的时间复杂度。在本文方案中，经过实验测试，在对一个密文多项式进行两阶段的数据隐藏时耗时均不超过 1ms。由于单个密文多项式在数据隐藏阶段处理的时间过短，本节中数据隐藏的执行时间定义为对一幅 512＊512 明文图像的密文数据进行数据隐藏的耗时。

表 3 NTRU 与其他一些公钥加密算法的时间复杂度对比

Table 3 Comparison of time complexity between NTRU and some other public-key cryptosystem algorithms

| | NTRU | RSA | McEliece | GGH | Paillier |
|---|---|---|---|---|---|
| 公钥长度 | $N$ | $N$ | $N$ | $N^2$ | $2N$ |
| 加密时间复杂度 | $O(N^2)$ | $O(N^2)$ | $O(N^2)$ | $O(N^2)$ | $O(2^N)$ |
| 解密时间复杂度 | $O(N^2)$ | $O(N^3)$ | $O(N^2)$ | $O(N^2)$ | $O(2^N)$ |

以将 Man 图像和 Boat 图像作为两阶段隐藏数据为例，图 8 给出了在不同参数下以 Lena 图像为原始载体图像进行数据隐藏的耗时，在嵌入数据量相同的情况下，基于不同参数进行数据嵌入的耗时差距较小，差距在 2%以内。图 9 给出了相同参数下对不同图像进行数据隐藏的耗时，在嵌入数据量相同的情况下，在不同载体图像上进行数据隐藏的耗时差距也较小。图 10 给出了以 Lena 图像为原始载体图像，对其密文按参数 $t=200, s=200$ 隐藏不同数据量的耗时，数据隐藏时间与数据量的大小成线性关系，进一步体现了本文所提方案在数据隐藏时间方面的稳定性。

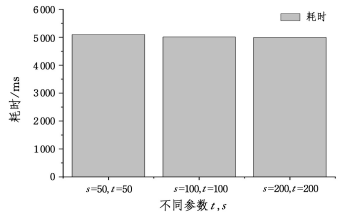



图 8 不同参数下隐藏数据的耗时

Fig. 8 Time cost to hide data with different parameters

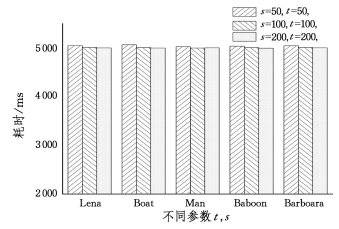



图 9 相同参数下对不同图像隐藏数据的耗时

Fig. 9 Time cost for different images to hide data with the same parameters

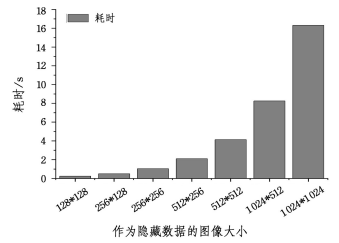



图 10 对 Lena 图像作为载体隐藏不同长度数据的耗时

Fig. 10 Time cost using Lena image as the carrier to hide data

### 4.4 安全性分析

首先分析数据隐藏过程的不可感知性，然后分析两阶段数据隐藏过程中隐藏数据的安全性。

#### 4.4.1 数据隐藏不可感知性分析

在第一阶段数据隐藏的过程中，将密文多项式 $e$ 加上多项式 $u$，将密文多项式 $e$ 转换成了密文多项式 $e'$，以实现第一阶段数据隐藏(此时 $e'$ 是明文$(m+u)$对应密文)。对于攻击者而言，无法确定 $e'$ 是明文$(m+u)$加密后得到的结果还是由密文 $e$ 加上 $u$ 实现了数据隐藏的结果。

在第二阶段数据隐藏的过程中，由于是对密文多项式 $e'$ 加上多项式 $v$，改变密文多项式数据隐藏段 2 系数实现数据隐藏，如果直接隐藏明文数据，这样可能会使新的密文数据隐藏段 2 的系数间存在一定逻辑关系，攻击者如果知道数据提取方法，则能察觉到这些系数间的一些逻辑关系，从而判断出隐藏的数据。因此，本文方案在第二阶段隐藏经 RC4 算法加密后的数据，这样就消除了密文数据隐藏段 2 的系数间的逻辑关系，即使有第三方攻击者正确获取了解密前提取隐藏数据的方法，其也只能获取到经 RC4 算法加密得到的一堆杂乱无章的 0,1 二进制数据，无法察觉多项式中隐藏了数据的系数之间的逻辑关系，从而无法感知数据的隐藏。

综上所述，本文提出的 NTRU 加密域可逆数据隐藏方案的数据隐藏过程具有不可感知性。

#### 4.4.2 被隐藏数据的安全性分析

本文所提方案是在 NTRU 加密域完成数据隐藏的，文献[23]从数值几何和代数数论的角度对 NTRU 加密算法的正确性和安全性进行了详细证明，其安全性可规约到格上最短向量问题。

下文分别分析在第一、二阶段数据隐藏过程中的被隐藏数据的安全性。

当对密文多项式 $e(e=r\times h+m)$进行第一阶段数据隐藏时，对密文多项式加上多项式 $u$ 得到密文多项式 $e'$以隐藏多项式 $u$ 中携带的秘密数据，即 $e'=e+u=r\times h+m+u=Enc(m+u)$，此时 $e'$即为明文多项式$(m+u)$对应的密文多项式。携带了秘密的数据的多项式 $u$ 从密文中难以被正确提取，只能从密文 $e'$解密后得到的明文多项式$(m+u)$中分离提取，无法在未解密的情况下提取隐藏的数据，因此本文方案中在第一阶段隐藏的数据的安全性等价于 NTRU 加密系统的安全性。

对于第二阶段隐藏的数据，由于可以在解密前提取该部分数据，因此，该阶段的数据隐藏不建议直接隐藏明文数据。在本文方案中，待隐藏的明文数据使用 RC4 加密处理后再进行数据隐藏的操作，这样隐藏的数据的安全性等价于 RC4 算法的安全性，即使有第三方在解密前获取了第二阶段隐藏的数据，没有 RC4 解密密钥也无法获得隐藏的数据。

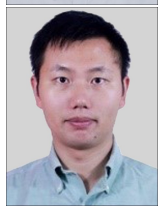
**结束语** 本文提出了一种基于多项式划分的 NTRU 加密域可逆数据隐藏方案，能在两阶段进行数据隐藏，分别在解密前提取和解密后提取。该方案利用 NTRU 加密系统中多项式的冗余特性，将多项式空间划分为明文段和数据隐藏段，隐藏数据的嵌入和提取与原始明文的恢复相互独立，并可以无损恢复原始明文载体。与同类型的其他方案相比，本文方案的数据嵌入率较高，在两个阶段可以根据应用场景的变化灵活调整。同时，本文方案使用了抗量子密码 NTRU 对原始载体数据加密，整体方案兼备了安全性高、运算速度快的优点。除此之外，本文方案可以对不同类型的载体进行加密并进行数据隐藏，应用场景广泛。本文方案的不足之处在于无法在加密域进行同态处理(如嵌入水印等)，因此与同态处理兼容的可逆数据隐藏方案将是今后的研究重点。

## 参 考 文 献

[1] SHI Y Q, LI X L, ZHANG X P, et al. Reversible data hiding: Advances in the past two decades[J]. IEEE Access, 2016, 4:

3210-3237.
[2] TIAN J. Reversible data embedding using a difference expansion [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2003, 13(8): 890-896.
[3] NI Z C, SHI Y Q, ANSARI N, et al. Reversible data hiding[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2006, 16(3): 354-362.
[4] PUECH W, CHAUMONT M, STRAUSS O. A reversible data hiding method for encrypted images[C]// Security, Forensics, Steganography, and Watermarking of Multimedia Contents X. SPIE, 2008: 534-542.
[5] HUANG F J, HUANG J W, SHI Y Q. New framework for reversible data hiding in encrypted domain[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(12): 2777-2789.
[6] PUTEAUX P, PUECH W. An efficient msb prediction-based method for high-capacity reversible data hiding in encrypted images[J]. IEEE Transactions on Information Forensics and Security, 2018, 13(7): 1670-1681.
[7] YI S, ZHOU Y C. Separable and reversible data hiding in encrypted images using parametric binary tree labeling[J]. IEEE Transactions on Multimedia, 2018, 21(1): 51-64.
[8] YIN Z X, PENG Y Y, XIANG Y Z. Reversible data hiding in encrypted images based on pixel prediction and bit-plane compression[J]. IEEE Transactions on Dependable and Secure Computing, 2020, 19(2): 992-1002.
[9] WANG Y M, CAI Z C, HE W G. High capacity reversible data hiding in encrypted image based on intra-block lossless compression[J]. IEEE Transactions on Multimedia, 2020, 23: 1466-1473.
[10] WANG Y M, HE W G. High capacity reversible data hiding in encrypted image based on adaptive msb prediction[J]. IEEE Transactions on Multimedia, 2021, 24: 1288-1298.
[11] YANG Y L, HE H J, CHEN F, et al. Reversible data hiding of image encrypted based on prediction error adaptive coding[J]. Journal of Computer Research and Development, 2021, 58(6): 1340-1350.
[12] ZHANG X P, LONG J, WANG Z C, et al. Lossless and reversible data hiding in encrypted images with public-key cryptography[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2015, 26(9): 1622-1631.
[13] XIANG S J, LUO X R. Efficient reversible data hiding in encrypted image with public key cryptosystem[J]. EURASIP Journal on Advances in Signal Processing, 2017, 2017(1): 1-13.
[14] XIONG L Z, DONG D P, XIA Z H, et al. High-capacity reversible data hiding for encrypted multimedia data with somewhat homomorphic encryption[J]. IEEE Access, 2018, 6: 60635-60644.
[15] ZHOU N, ZHANG M Q, WANG H, et al. Separable reversible data hiding scheme in homomorphic encrypted domain based on NTRU[J]. IEEE Access, 2020, 8: 81412-81424.
[16] WU H T, CHEUNG Y M, ZHUANG Z W, et al. Reversible data hiding in homomorphic encrypted images without preprocessing[C]// International Workshop on Information Security Application. Springer, 2019: 141-154.
[17] WU H T, CHEUNG Y M, ZHUANG Z W, et al. Lossless data hiding in encrypted images compatible with homomorphic proces-sing[J]. IEEE Transactions on Cybernetic, 2022, 53(6): 3688-3701.
[18] CHEN B, WU X T, LU W, et al. Reversible data hiding in encrypted images with additive and multiplicative public-key homomorphism[J]. Signal Processing, 2019, 164: 48-57.
[19] ZHOU N, ZHANG M Q, TANG H Q, et al. Reversible data hiding algorithm in encrypted domain based on NTRU[J]. Science Technology and Engineering. 2020, 20(32): 10.
[20] REGEV O. On lattices, learning with errors, random linear codes, and cryptography[J]. Journal of the ACM(JACM), 2009, 56(6): 1-40.
[21] PAILLIER P. Public-key cryptosystems based on composite degree residuosity classes[C]// International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 1999: 223-238.
[22] BRAKERSKI Z, GENTRY C, VAIKUNTANATHAN V. (Leveled) Fully homomorphic encryption without bootstrapping [J]. ACM Transactions on Computation Theory(TOCT), 2014, 6(3): 1-36.
[23] HOFFSTEIN J, PIPHER J, SILVERMAN J H. NTRU: A ring-based public key cryptosystem[C]// International Algorithmic Number Theory Symposium. Springer, 1998: 267-288.
[24] FRIDRICH J, GOLJAN M, LISONEK, et al. Writing on wet paper[J]. IEEE Transactions on Signal Processing, 2005, 53(10): 3923-3935.
[25] BONEH D, GOH E J, NISSIM K. Evaluating 2-dnf formulas on ciphertexts[C]// Theory of Cryptography Conference. Springer, 2005: 325-341.
[26] ELGAMAL T. A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE Transactions on Information Theory, 1985, 31(4): 469-472.
[27] WANG C, YAO H N, WANG B N, et al. Process in quantum computing cryptography attacks[J]. Chinese Journal of Computers. 2020, 43(9): 1691-1707.
[28] AJTAI M. Generating hard instances of lattice problems[C]// Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing. 1996: 99-108.
[29] SHORP W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer[J]. SIAM Review, 1999, 41(2): 303-332.

**LIU Dingcai**, born in 1997, postgraduate. His main research interest is reversible data hiding in homomorphic encryption domain.



**WU Haotian**, born in 1980, Ph.D, associate professor. His main research interests include reversible information hiding, privacy preservation, password guessing and blockchain.

（责任编辑：喻藜）