基于格的密码学技术专题讲座(二)

第 3 讲　基于格的公钥密码方案

王兆丽，杨　明，韩敬利
(解放军理工大学指挥信息系统学院，江苏 南京 210007)

摘　要：文章主要介绍了几种有代表性的基于格的公钥密码方案及其改进方案，对他们的设计思想、具体方案进行了详细的阐述，并对其安全性进行了讨论，最后从效率、安全性等方面对它们进行了简单的比较。

关键词：格；公钥密码方案；有误差学习
中图分类号：TP309.2　　**文献标识码**：A　　**文章编号**：CN32-1289(2014)02-0079-08

Lattice-based Public-key Encryption Scheme

WANG Zhao-li, *YANG Ming*, *HAN Jing-li*
(College of Command Information System, PLAUST, Nanjing 210007, China)

Abstract: The main lattice-based public key encryption schemes, such as AD, GGH, NTRU, LWE-PKE and its variants were described. The design strategy and the construction of the schemes were specifically explained, and the security of the schemes was also discussed. Finally the efficiency and security of these schemes were compared.

Key words: lattice; public-key encryption scheme; LWE

在过去的十几年中，格正逐渐成为一种很有吸引力的构建密码系统的基础。基于格的密码系统优势主要体现在如下三个方面：首先，它的安全性是基于最坏情况难解性的。最坏情况难解性是计算复杂性理论中的概念，即问题只有在某些输入下是困难的而在其他情况下是容易求解的。大部分已知密码系统的安全性都是建立在一个平均情况下难解问题的基础上的(例如 RSA 是基于整数因子分解问题的，虽然整数因子分解问题是最坏情况难解的，但在实际应用中，是通过某种特殊方法将问题转化成平均情况下难解的)，但是这不是最安全的，基于最坏情况难解性的密码系统显然会更安全。由于 Ajtai 开创性的工作[1]，将格问题的平均情况难解性与最坏情况难解性联系了起来，这使人们意识到基于格的密码系统恰好具有这样的优点。其次，基于格的密码系统被认为是可以抵抗量子攻击的，目前为止还没有解决格难题的量子算法。最后，由于格的线性结构，使得基于格的密码系统相对高效且易于实现，有些基于格的公钥密码方案甚至已经成为 RSA 的有力竞争者。

1997 年，Ajtai 和 Dwork 提出了一个基于格的公钥密码方案[2]，该方案是第一个被证明解决系统任意实例的难度等价于解决系统最难实例难度的密码方案。之后几年，一些基于格的公钥密码方案被相继提出，它们大致可分为两类，一类几乎是纯理论的，追求严格的安全性证明；另一类追求实用，但不具备可证明安全性。下面详细介绍几种有代表性的基于格的公钥密码方案。

收稿日期：2013-02-20；**修回日期**：2014-04-17
作者简介：王兆丽(1980－)，女，硕士，助教.

1 基本概念和符号

poly(n)表示未定义的函数 $f(n)=O(n^c)$。用 $\tilde{O}(n)$表示 $O(n)\cdot \mathrm{poly}(\log n)$。对于正整数 n,用$[n]$表示集合$\{1, 2, \cdots, n\}$。对于实数 d,$\lceil d \rfloor$ 表示最接近 d 的整数,$\mathrm{frc}(d)=|d-\lceil d \rfloor|$,代表 d 到整数集 $\mathbf{Z}$ 的距离。$\mathbf{T}_n^p(o, r)$表示以 l_p 为范数,n 维实数向量 $\boldsymbol{O}$ 为球心,r 为半径的 n 维球,如果上下文中已经提到 n,可将其省略;如果 $p=2$,p 可以省略;如果以原点为球心,$\boldsymbol{O}$ 可以省略。$\mathbf{P}(\boldsymbol{C})$表示由 $\boldsymbol{C}$ 张成的向量空间。关于格的基本概念可参考专题第一讲[3]。

2 早期基于格的密码方案

2.1 AD 公钥密码方案

在证明了格中平均情况与最坏情况难解性之间的关系后,Ajtai 和 Dwork 提出了三个基于格的公钥密码方案,这就是著名的 AD 公钥密码方案。该方案在理论密码学方面是一个重要的突破,但在实际应用中它却不是一个有效的方案,系统的公钥长度达到了 $\tilde{O}(n^4)$,每一位加密后的长度膨胀了 $\tilde{O}(n^2)$位(其中 n 是格的维数)。

2.1.1 基本思想

AD 公钥密码方案的基本思想是:在 n 维向量空间中选择一个短向量作为私钥,以该向量为法向量的超平面构成一个集合 $\mathbf{H}$,则公钥由靠近 $\mathbf{H}$ 中超平面的 m 个向量组成。加密过程是按位进行的:0 的密文是这 m 个向量的随机和(也是 $\mathbf{H}$ 中某个超平面附近的向量);1 的密文是一个随机 n 维实数向量。解密时,计算密文向量与私钥的内积,用它来确定密文向量到离它最近的超平面之间的距离,假如距离足够小,对应的明文是 0,否则为 1。

2.1.2 AD 密码方案

方案 1 :AD 密码方案

Setup(1^n)。根据安全参数 n,计算 $m=n^3$,$g=2^{O(n\log n)}$,$r=n^{-3}$。

KeyGen( )。选择 $\boldsymbol{u}\leftarrow\mathbf{T}(1)$,$\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m\leftarrow\{\boldsymbol{x}\in\mathbf{T}(g)\mid\langle\boldsymbol{x}, \boldsymbol{u}\rangle\in\mathbf{Z}\}$。随机选择 $\boldsymbol{y}_{i,j}\leftarrow\mathbf{T}(r)$,其中 $i=1, \cdots, m$,$j=1, \cdots, n$。对 $i=1, \cdots, m$ 计算 $\boldsymbol{z}_i=\sum_{j=1}^{n}\boldsymbol{y}_{i,j}$。然后,计算 $\boldsymbol{a}_i=\boldsymbol{x}_i+\boldsymbol{z}_i$。令 i_0 等于满足下面条件的最小 i,由 $\boldsymbol{a}_{i+1}, \cdots, \boldsymbol{a}_{i+n}$张成的平行六面体的宽度至少是 n^{-2}。对 $j=1, \cdots, n$,令 $\boldsymbol{b}_j=\boldsymbol{a}_{i_0+j}$。解密密钥 $\boldsymbol{u}$,加密密钥$(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_m, i_0)$。

Enc($(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_m, i_0)$, $m=t$)。令明文 $t\in\{0, 1\}$。加密 $t=0$ 时,选择 $\boldsymbol{e}\in\{0, 1\}^m$,计算 $\boldsymbol{c}=\boldsymbol{A}\boldsymbol{e} \bmod \boldsymbol{B}$,其中 $\boldsymbol{B}=[\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n]$;加密 $t=1$ 时,随机均匀地选择 $\boldsymbol{c}\leftarrow\mathbf{P}(\boldsymbol{B})$,密文即为 $\boldsymbol{c}$。

Dec($\boldsymbol{u}$, $\boldsymbol{c}$)。对于收到的密文,计算 $d=\langle\boldsymbol{c}, \boldsymbol{u}\rangle$。假如$|\mathrm{frc}(d)|\leqslant 1/n$,输出 0;否则输出 1。

2.1.3 安全性与攻击

AD 方案的安全性依赖于 u-SVP 问题,即:找出 n 维格中的唯一非零最短向量 $\boldsymbol{v}$,使得其他长度小于它的 n^c 倍的向量都平行于 $\boldsymbol{v}$,c 是一个充分大的常数。方案提出者证明,假如方案的一个随机实例被攻破,那么存在概率多项式时间算法解决 u-SVP 问题的最坏实例。目前的研究表明,u-SVP 是 NP-hard 问题,所以 AD 密码方案是最坏情况下可证明安全的。这种最坏情况下的安全性保证并不能使系统免受攻击,但可以做到:确保针对系统的攻击只对少部分参数有效,并且不是渐进的。也就是说,它可以确保系统在设计上不存在基本缺陷,并且可以帮助系统设计者选择合适的系统参数。

Nguyen 和 Stern 提出了一种旨在恢复私钥的攻击方法[4],他们的实验表明,要达到实用的安全级别,需要非常大的密钥;他们还对 AD 方案的安全水平给出了准确的评估,证明了如果可以解决近似因子是 $cn^{1.33}$

的 app-CVP 问题，就可以以概率优势 d 恢复明文（d 是与 c 相关的常数），但目前还没有这样的算法出现。

2.2 GGH 公钥密码方案

2.2.1 基本思想

通过格的一组基可以表示出格中的任何格点 $\boldsymbol{m}$，给 $\boldsymbol{m}$ 加上一个长度为 δ 的随机向量 $\boldsymbol{e}$，产生一个与 $\boldsymbol{m}$ 距离为 δ 的随机点 $\boldsymbol{c}=\boldsymbol{m}+\boldsymbol{e}$。如果 δ 比较小，$\boldsymbol{m}$ 就是 $\boldsymbol{c}$ 在格中的最近向量。如果给定格的一组基 $\boldsymbol{B}$ 和向量 $\boldsymbol{c}$，反过来求解最靠近 $\boldsymbol{c}$ 的格向量 $\boldsymbol{m}$，就是格中的最近向量问题（CVP）。从第一讲已经介绍过，CVP 是 NP 完全问题。但这也和给定的基 $\boldsymbol{B}$ 的质量有关，如果 $\boldsymbol{B}$ 是一组“好”的规约基，有算法可以在较短时间内找到 $\boldsymbol{m}$；如果 $\boldsymbol{B}$ 是一组普通基，将很难找到 $\boldsymbol{m}$ 的近似解（与 δ 的大小有关）。利用这一特性，在改进 AD 公钥密码方案的同时，Goldreich、Goldwasser 和 Halevi 提出了一种新的密码方案 GGH[5]，他们的贡献是将密钥的大小降低到了 $O(n^3)$。

GGH 密码方案的粗略描述如下：私钥是一个随机格 $\mathbf{L}$ 的“好”基 $\boldsymbol{G}$，公钥是这个随机格的“坏”基 $\boldsymbol{B}=\boldsymbol{G}\boldsymbol{U}^{-1}$，其中 $\boldsymbol{U}$ 是幺模矩阵。明文信息被编码成系数向量 $\boldsymbol{s}$，密文 $\boldsymbol{p}=\boldsymbol{B}\boldsymbol{s}+\boldsymbol{x}$，这里 $\boldsymbol{x}$ 是一个小的随机误差向量。为了解密 $\boldsymbol{p}$，首先计算 $\boldsymbol{d}\leftarrow\lceil \boldsymbol{G}^{-1}\boldsymbol{p} \rfloor$，由于误差 $\boldsymbol{x}$ 很小并且 $\boldsymbol{G}$ 是好基，$\boldsymbol{d}$ 即为 $\boldsymbol{G}^{-1}\boldsymbol{B}\boldsymbol{s}$。用 $\boldsymbol{B}^{-1}\boldsymbol{G}$ 乘以 $\boldsymbol{d}$，即可得到明文 $\boldsymbol{s}$。

2.2.2 GGH 密码方案

作者提出了两种不同的密钥生成算法：①KeyGen1：选择一个“随机”格，生成 $\boldsymbol{G}\leftarrow\{-l,-(l-1),\cdots,l-1,l\}^{n\times n}$，这里 l 是一个小整数，例如 4；②KeyGen2：选择一个“正交”格，生成噪声矩阵 $\boldsymbol{G}'\leftarrow\{-l,\cdots,l\}^{n\times n}$，这里 l 是一个小整数，例如 4。计算 $\boldsymbol{G}\leftarrow\boldsymbol{G}'+k\boldsymbol{I}_n$，这里 k 是一个大整数，例如 $\sqrt{n}$。

KeyGen2 生成的私有基 $\boldsymbol{G}$ 可以从较长的扰动向量中恢复明文，但是，也给敌手从私有基恢复出公共基提供了更大的可能。

方案 2：GGH 密码方案

KeyGen(1^n)。使用 KeyGen1 或者 KeyGen2 生成基 $\boldsymbol{G}$ 作为私钥，然后根据概率分布 χ 在 $\mathbf{L}(\boldsymbol{B})$ 所有可能的基中选择一个基 $\boldsymbol{B}$ 作为公钥。

Enc($\boldsymbol{B}$, $\boldsymbol{s}$)。首先选择 $\boldsymbol{x}\leftarrow D_n$，其中 $D_n\subseteq\mathbf{Z}^n$，并且 D_n 中每个元素都是短的。然后，计算密文 $\boldsymbol{p}=\boldsymbol{B}\boldsymbol{s}+\boldsymbol{x}$。

Dec($\boldsymbol{G}$, $\boldsymbol{p}$)。计算并输出 $\boldsymbol{s}=\boldsymbol{B}^{-1}\boldsymbol{G}\lceil \boldsymbol{G}^{-1}\boldsymbol{p} \rfloor$。

2.2.3 安全性与攻击

GGH 方案没有安全性证明，自从方案提出后，一直是密码分析攻击的对象。它的安全性依赖于用非正交基解决 CVP 问题的困难性，而 LLL 格基归约算法可以在多项式时间内找到接近正交的基，在实践中，假如格的维数 $n<100$，LLL 算法很容易找到一个足够好的基攻破 GGH 方案，甚至当 $n<200$ 时，一些 LLL 的优化算法也可以攻破 GGH。随着维数 n 的增加，攻破 GGH 的难度也在增加，然而方案提出两年后，Nguyen 解决了 GGH 作者提出的参数 $n=350$ 的挑战，并且给出了 $n=400$ 时的部分解决方案[6]。几年后 Lee 和 Hahn 使用 Nguyen 的方法完全解决了 $n=400$ 的 GGH 挑战[7]。但是，这种攻击方法并不是渐进的，他们只是证明对于特定的安全参数系统可以被攻破，因此可以通过增大安全参数来避免攻击。但是，考虑到运行效率和存储空间，从实用角度看，GGH 方案是不安全的。

2.3 NTRU 公钥密码方案

NTRU 是基于多项式环的公钥密码方案[8]，它也与一类特殊格密切相关。NTRU 是迄今为止最实用的基于格的公钥密码方案，它以算法简洁、计算速度快、占用存贮空间小（公钥的长度是 $O(n\log n)$）等优点，成为最有可能替代如今大量使用的 RSA 的公钥密码方案。

2.3.1 基本思想

下面先从多项式环的角度介绍 NTRU 的基本思想：令 $\mathbf{R}=\mathbf{Z}[x]/(x^n-1)$，由于 $\mathbf{R}$ 中的多项式 $f=\sum_{i=0}^{n-1}f_ix^i$ 可以用 $\boldsymbol{f}=(f_0,\cdots,f_{n-1})\in\mathbf{Z}^n$ 表示，所以很自然地环 $\mathbf{R}$ 可以用 $\mathbf{Z}^n$ 表示。定义 $\mathbf{R}$ 上的加法运算为

普通的多项式加法，如下定义 $\mathbf{R}$ 上的乘法运算$\otimes$：设 $f, g\in\mathbf{R}$，$f(x)=f_0+f_1x+\cdots+f_{n-1}x^{n-1}$，$g(x)=g_0+g_1x+\cdots+g_{n-1}x^{n-1}$，$f\otimes g=h_0+h_1x+\cdots+h_{n-1}x^{n-1}$，其中 $h_k=\sum_{i+j\equiv k(\bmod n)} f_i\cdot g_j$。

令 p、q 是两个小的、互素的整数，例如 $p=3$，$q=128$（一般要求 q 远大于 p）。定义系统的私钥是一对系数很小（取自集合$\{0, 1, -1\}$）的多项式 $f, g\in\mathbf{R}$，要求 f, g 的系数足够小是为了让系数模 p 与模 q 的顺序可以颠倒，即先模 p 后模 q 的结果与先模 q 后模 p 的结果相等（一般来说这样运算的结果是不等的，例如(32 mod 3)mod 8≠(32 mod 8)mod 3)。公钥是 $h=pf^{-1}\otimes g \bmod q$，这里逆和乘都是环 $\mathbf{Z}_q^n[x]/(x^n-1)$ 中的运算。对于要加密的消息 m，随机选择多项式 r，加密算法输出 $t=m+h\otimes r \bmod q$ 作为密文。解密算法计算 $a=f\otimes t \bmod q$，然后计算 $m'=f^{-1}\otimes a \bmod p$。从而使解密算法以很高的概率恢复出明文，即 $m'=m$。解密算法的工作原理为：

$$a=f\otimes t \bmod q=f\otimes h\otimes r+f\otimes m(\bmod q)=pf\otimes f^{-1}\otimes g\otimes r+f\otimes m(\bmod q)=$$
$$pg\otimes r+f\otimes m(\bmod q)$$

通过选择合适的参数，将 $f\otimes t$ 的系数控制在$[-q/2, q/2]$区间内，这样可以保证其在模 q 运算时参数不会改变，将 a 模 p 约化后得到 $f\otimes m(\bmod p)$，再乘以 f^{-1} 即可恢复出加密消息 $m(\bmod p)$。

2.3.2 NTRU 密码方案

NTRU 方案也可以用一类特殊格来描述，即 $2n$ 维的 q 模 bi 循环格，也称循环模格（convolutional modular lattices）。q 模意味着它首先是 q-元格[3]。一个格是 bi 循环的必须满足如下条件：对任意向量 $x=[x_1, \cdots, x_n]^T$，定义循环函数 $\mathrm{rot}(x)=[x_n, x_1, \cdots, x_{n-1}]^T$ 和 x 的循环矩阵 $M_x=[x, \mathrm{rot}(x), \cdots, \mathrm{rot}^{n-1}(x)]$，对于任意的 $2n$ 维向量 $z=[x^T, y^T]^T$（其中 $x, y\in\mathbf{Z}^n$），定义双重循环函数 $\mathrm{rot}_2(z)=[\mathrm{rot}(x)^T, \mathrm{rot}(y)^T]^T$。假如对任意的格向量 z，$\mathrm{rot}_2(z)$ 也在格中，称 $2n$ 维的格是 bi 循环的。下面给出基于格的 NTRU 方案。

方案 3：NTRU 密码方案

Setup(1^n)。输入安全参数 n，输出参数 p，q。

KeyGen( )。私钥是一个 $2n$ 维的短向量 $v=(pg^T, f^T)^T$。令 $h=pf^{-1}\otimes g \bmod q$，公钥是由 v 产生的 q 模 bi 循环格的基 $H=\begin{bmatrix} qI & M_h \\ 0 & I \end{bmatrix}$（$I$ 为单位矩阵）。

Enc(H, m)。随机生成 n 维扰动向量 r，计算密文

$$\begin{bmatrix} m \\ -r \end{bmatrix} \bmod H=\begin{bmatrix} m \\ -r \end{bmatrix} \bmod \begin{bmatrix} qI & M_h \\ 0 & I \end{bmatrix}=\begin{bmatrix} 0 \\ m+h\otimes r \bmod q \end{bmatrix}=\begin{bmatrix} 0 \\ t \end{bmatrix}$$

Dec(v, t)。首先计算 $a=f\otimes t \bmod q$，再计算 $m=f^{-1}\otimes a \bmod p$。

通过分析不难发现，上面介绍的方案具有可区分性和可延展性，所以在实际应用中，需要通过增加随机填充的方式使 NTRU 具有不可区分性和不可延展性。

2.3.3 安全性与攻击

NTRU 密码方案的显著缺点在于它的安全性，它的安全性依赖于 NTRU 所使用的特殊格上的最近向量问题，然而，与这类格相关的问题还没有被证明，例如：这类特殊格上的 SVP、CVP 等问题同一般格上的问题一样是 NP-hard 问题吗？能证明最坏实例同一般实例之间的联系吗？目前，针对 NTRU 的攻击有很多种，Jaulmes 和 Joux 提出了选择密文攻击[9]，并且指出通过选择合理的填充方案可以防止这种攻击。Graham 提出了根据解密失败来恢复密钥的攻击[10]，并在之后综合利用格基归约和中间相遇策略降低了恢复密钥所需要的循环次数。2009 年，Hirschhorn[11]推荐了一组参数选择和填充方案，使 NTRU 可以抵抗绝大部分攻击。

3 基于 LWE 的公钥密码方案

2005 年，Regev 提出了一个基于有误差学习问题 LWE（Learning With Errors）的公钥密码方案（简称

其为 LWE-PKE)[12]，并且证明了格问题 GapSVP 和 SIVP 在量子条件下可归约为 LWE 问题，也就是说，解决 LWE 问题意味着可以找到解决格问题 GapSVP 和 SIVP 的量子算法。

有误差学习问题可以简单的概括为：从给定的一组关于向量 s 的“近似”等式中恢复出 s。例如，给定如下“近似”等式

$$\begin{aligned}
&14s_1+15s_2+5s_3+2s_4\approx 8(\bmod 17)\\
&13s_1+14s_2+14s_3+6s_4\approx 16(\bmod 17)\\
&6s_1+10s_2+13s_3+s_4\approx 3(\bmod 17)\\
&10s_1+4s_2+12s_3+16s_4\approx 12(\bmod 17)\\
&\vdots\\
&6s_1+7s_2+16s_3+2s_4\approx 3(\bmod 17)
\end{aligned}$$

目标是从中恢复出 s(答案是 $s=(0, 13, 9, 11)$)。如果没有误差，使用高斯消元法很容易解出 s。引入误差后，求解 s 的困难性将与格上的 GapSVP 和 SIVP 问题的最坏情况难解性联系起来。

3.1 单比特 LWE-PKE 方案

Regev 提出的基于 LWE 的公钥密码方案是逐位加密的，它不仅具有可证明安全性，在效率方面较 AD 和 GGH 方案也有优势：公钥的长度是 $\tilde{O}(n^2)$，信息加密后的密文长度膨胀了 $\tilde{O}(n)$，而且在各方共享一个长度为 $\tilde{O}(n^2)$ 的随机位串的情况下，公钥的长度可以进一步降为 $\tilde{O}(n)$。

3.1.1 基本原理

Regev 的方案依赖于求解 s 的困难性，它将一组“近似”等式设置成公钥，等式对应的解向量 s 设为私钥，加密时，随机选择若干个“近似”等式相加。如果明文是 0，就将得到的新“近似”等式作为密文输出；加密 1 时，在得到的新“近似”等式右边加上一个合适的值后输出。解密时，很容易利用 s 判断作为密文的“近似”等式是否真的“近似”相等并以此恢复明文。

3.1.2 单比特 LWE 公钥密码方案

令 n 为系统的安全参数，p 是 n^2 到 $2n^2$ 之间的一个素数，$m=(1+\varepsilon)(n+1)\log p$(任意常数 $\varepsilon>0$)，χ 是 $\mathbf{Z}_p$ 上的概率分布。下面的加法运算都是在 $\mathbf{Z}_p$ 中进行，即全部是模 p 运算。

方案 4：基于 LWE 的单比特密码方案(LWE-PKE)

Setup(1^n)。输入安全参数 n，选择 n^2 到 $2n^2$ 之间的一个素数 p，对任意常数 $\varepsilon>0$，计算 $m=(1+\varepsilon)(n+1)\log p$。$\mathbf{Z}_p$ 上的概率分布 $\chi=\overline{\Psi}_{\alpha(n)}$，其中 $\alpha(n)=O(1/(\sqrt{n}\log n))$，即 $\alpha(n)$ 满足 $\lim\limits_{n\to\infty}\alpha(n)\cdot\sqrt{n}\log n=0$。

KeyGen( )。随机均匀选择 $s\in\mathbf{Z}_p^n$ 作为私钥；对 $i=1, \cdots, m$，从均匀分布中选择 m 个线性独立的向量 $\boldsymbol{a}_1, \cdots, \boldsymbol{a}_m\in\mathbf{Z}_p^n$，再根据 χ 选择 m 个元素 $e_1, \cdots, e_m\in\mathbf{Z}_p$，公钥为 $(\boldsymbol{a}_i, b_i)_{i=1}^m$，这里 $b_i=\langle\boldsymbol{a}_i, s\rangle+e_i$。

Enc($(\boldsymbol{a}_i, b_i)_{i=1}^m$, t)。随机在 $[m]$ 的 2^m 个子集中选择一个集合 $\mathbf{S}$，用 $\left(\sum_{i\in\mathbf{S}}\boldsymbol{a}_i, \sum_{i\in\mathbf{S}}b_i\right)$ 加密 0；用 $\left(\sum_{i\in\mathbf{S}}\boldsymbol{a}_i, \left\lfloor\frac{p}{2}\right\rfloor+\sum_{i\in\mathbf{S}}b_i\right)$ 加密 1。

Dec(s, ($\boldsymbol{a}$, b))。如果 $b-\langle\boldsymbol{a}, s\rangle$ 更接近 0，输出 0；如果 $b-\langle\boldsymbol{a}, s\rangle$ 更接近 $\left\lfloor\frac{p}{2}\right\rfloor$，输出 1。

方案中，如果没有误差量 e，$b-\langle\boldsymbol{a}, s\rangle$ 的值要么是 0，要么是 $\left\lfloor\frac{p}{2}\right\rfloor$，解密将永远正确。因此只有当误差量的和大于 $p/4$ 时，才可能发生解密错误。而加密过程中最多只有 m 个误差量相加，每个误差量的标准偏差是 $\alpha(n)\cdot p$，则标准偏差的和小于 $p/\log n$，因此，误差量的和大于 $p/4$ 的概率是可以忽略的。

3.1.3 安全性与攻击

文献[12]中，作者将最坏情况下的格问题 GapSVP 和 SIVP 在量子条件下归约为 LWE 问题，这表明 LWE-PKE 的安全性是建立在量子条件下最坏情况 GapSVP 问题的困难性上的。这里有必要强调安全性是建立在量子条件下的，这意味着直到出现量子算法解决 GapSVP 问题，LWE-PKE 都是安全的，因此，从这个角度看 LWE-PKE 的安全性要弱于 AD 的安全性。虽然存在从复杂格问题到 LWE 问题的规约，但关于

特定 LWE 实例的具体难解性还知之甚少(例如给定具体 p、m 和 n,使用已知的一些算法恢复密钥的计算代价的多少?)。正因为如此,很多基于 LWE 的系统没有建议具体参数和评估安全性,这也成为将基于 LWE 问题的密码系统应用于实际的障碍。近来,有一些学者已经开始从事这样的工作,文献[13,14]使用格基规约、BKW 等方法评估了具体的安全参数所能达到的安全级别、算法效率和计算所需的存储空间等。

3.2 多比特 LWE-PKE 方案

LWE-PKE 的原始方案被提出后,出现了一些对其效率进行改进的方案,其中一个改进方向是将加密过程由单比特加密改进为多比特加密。文献[15]提出了一种多比特方案,但在效率上只对原始方案有微小的改进,即密文膨胀了 $O(n)$,去掉了原始方案的 log 因数。文献[16]提出的多比特方案对原方案的效率有重大改进,通过观察不难发现,原始 LWE-PKE 方案中使用了一个由 m 个线性独立的向量 $\boldsymbol{a}_1, \cdots, \boldsymbol{a}_m \in \mathbf{Z}_p^n$ 构成的 $n \times m$ 矩阵 $\boldsymbol{A}$,正是由于它的使用,使得原方案公钥的长度是 $\tilde{O}(n^2)$,密文长度膨胀了 $\tilde{O}(n)$。基于此文献[16]做了如下改进:在加密明文空间 $\mathbf{Z}_p^l$ 中的向量 $\boldsymbol{v}$ 的每个分量 v_i 时,将 $\boldsymbol{A}$ 安全地重用 l 次(每个 v_i 对应各自的私钥 $\boldsymbol{s}_i$ 和误差向量 $\boldsymbol{e}_i$),这样可在不改变安全性的同时将系统在多个方面的效率提高 $O(n)$。

3.3 对偶 LWE-PKE 方案

自从 Ajtai 创造性地将格问题的平均情况难解性和最坏情况难解性联系起来之后,出现了很多基于格的哈希函数和公钥加密方案,但一直没有安全有效的 IBE 方案。直到 2008 年,Gentry 等人构造了一个基于 LWE 问题的 IBE 方案[17],方案构建之初,Gentry 等人考虑过以 LWE-PKE 方案为基础来构建这个 IBE 方案,但是,LWE-PKE 方案的公钥在空间中不是均匀分布的,而是集中分布于格附近,因此,无法将任意身份信息映射成有效的公钥。为了解决这一问题,Gentry 等人在文献[17]中提出了一个 LWE-PKE 的"对偶"方案,该方案将 LWE-PKE 中的密钥产生过程和加密算法进行了互换,在解决公钥分布问题的同时,也将公钥长度降低到 $\tilde{O}(n)$。

3.3.1 对偶 LWE-PKE 方案

下面的加法运算都是在 $\mathbf{Z}_p$ 中进行,即全部是模 p 运算。

Setup(1^n)。输入安全参数 n,输出随机矩阵 $\boldsymbol{A} \in \mathbf{Z}_q^{n \times m}$ 作为参数。

KeyGen($\boldsymbol{A}$)。根据离散高斯分布 $D_{\mathbf{Z}^m,r}$ 选择误差向量 $\boldsymbol{e}$ 作为私钥;公钥为 $\boldsymbol{u}=\boldsymbol{A}\boldsymbol{e} \bmod q$。

Enc($\boldsymbol{u}$, b)。为了加密 $b \in \{0, 1\}$,随机选择 $\boldsymbol{s} \in \mathbf{Z}_q^n$,独立地根据概率分布 χ 选择向量 $\boldsymbol{x} \in \mathbf{Z}_q^m$ 的每个分量,计算 $\boldsymbol{p}=\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}+\boldsymbol{x} \in \mathbf{Z}_q^m$。根据概率分布 χ 选择标量 x,输出($\boldsymbol{p}$, $c=\boldsymbol{u}^{\mathrm{T}}\boldsymbol{s}+x+b\left\lfloor\frac{q}{2}\right\rfloor$)作为密文。

Dec($\boldsymbol{e}$, ($\boldsymbol{p}$, c))。计算 $b'=c-\boldsymbol{e}^{\mathrm{T}}\boldsymbol{p} \in \mathbf{Z}_q$,如果 b' 比 $\left\lfloor\frac{q}{2}\right\rfloor$ 更接近 0,输出 0;否则输出 1。

3.3.2 正确性与安全性

容易计算 $c-\boldsymbol{e}^{\mathrm{T}}\boldsymbol{p}=x-\boldsymbol{x}^{\mathrm{T}}\boldsymbol{e}+b\left\lfloor\frac{q}{2}\right\rfloor$。文献[15]中证明当 $q \geqslant 5rm$,$\alpha \leqslant 1/(r\sqrt{m} \cdot \omega(\sqrt{\log n}))$,$\chi=\overline{\Psi}_\alpha$,$m \geqslant 2n\lg q$ 时,$x-\boldsymbol{x}^{\mathrm{T}}\boldsymbol{e}$ 的取值以压倒性的概率优势小于 $\frac{q}{5}$,以此保证 b 被正确解密。同时,作者证明,如上选择参数时,对偶 LWE-PKE 方案是 IND-CPA 安全的。

3.4 Ring-LWE-PKE 方案

在多比特 LWE-PKE 方案中已经分析过,矩阵 $\boldsymbol{A}$ 的使用造成了 LWE-PKE 方案效率过低的问题(如图1),因此改进效率的另一个方向就是采用更紧凑简洁的代数结构(如图2)。基于这一思路,2010 年,Vadim Lyubashevsky、Chris Peikert 和 Oded Regev 提出了 LWE 问题的一个变体 Ring-LWE 问题[18],并

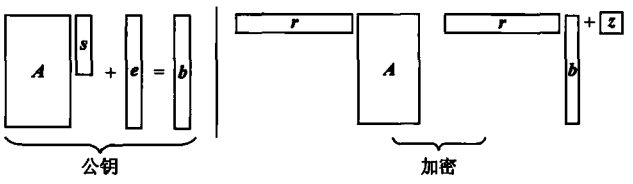

图 1 LWE-PKE 方案图

且证明 ring-LWE 问题同样具有很强的难解性。

3.4.1　基本原理

下面给出 Ring-LWE 问题的非形式化描述，为了更具体，下面的介绍只涉及一个特殊的“好”环。令 $f(x)=x^n+1\in\mathbf{Z}[x]$，这里的安全参数 n 取 2 的幂，这使得 $f(x)$ 在有理数域不可约。令 $\mathbf{R}=\mathbf{Z}[x]/\langle f(x)\rangle$ 是模 $f(x)$ 的整数多项式环，则 $\mathbf{R}$ 中的元素全部是次数小于 n 的整数多项式。令 $q=1 \bmod 2n$ 是一个足够大的素数，$\mathbf{R}_q=\mathbf{R}/\langle q\rangle=\mathbf{Z}_q[x]/\langle f(x)\rangle$ 是模 $f(x)$ 和 q 的整数多项式环，则 $\mathbf{R}_q$ 中的元素全部是次数小于 n 且系数取自集合 $\{0, \cdots, q-1\}$ 的多项式。搜索 Ring-LWE 问题可以描述为：秘密 s 是 $\mathbf{R}_q$ 中的元素，g_i 随机取自 $\mathbf{R}_q$，e_i 很小，目标是从若干个 $(g_i, g_i\cdot s+e_i)$ 中恢复出 s；判定 Ring-LWE 问题可以描述为：给定若干个 (g_i, t_i)，判定是否存在 s 和一组小的 e_i，使得 $t_i=g_i\cdot s+e_i$，或者 t_i 在 $\mathbf{R}_q$ 中是随机分布的。

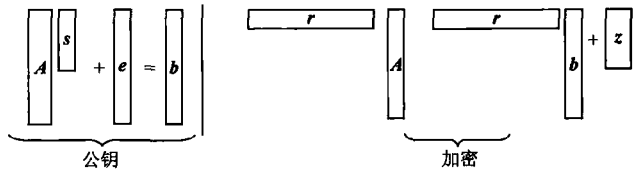

图 2　Ring-LWE-PKE 方案图

3.4.2　Ring-LWE-PKE 方案

Setup(1^n)。输入安全参数 n，选择 $q=1 \bmod 2n$，$m\approx \lg q=O(\log n)$。

KeyGen(q, m)。随机独立地选择 m 个多项式 $a_i\in\mathbf{R}_q$ 和 m 个“小”多项式 $r_i\in\mathbf{R}$。令 $a_{m+1}=\sum_{i\in[m]} r_i a_i$，输出私钥 $(r_1, \cdots, r_m, r_{m+1}=-1)\in\mathbf{R}^{m+1}$，公钥 $(a_1, \cdots, a_{m+1})\in\mathbf{R}_q^{m+1}$。

Enc($(a_1, \cdots, a_{m+1})$, z)。为了加密 n 位信息 $z\in\{0, 1\}^n$，随机选择 $s\in\mathbf{R}_q$，对于每一个 $i\in[m+1]$，$e_i\in\mathbf{R}$ 是取自某一分布的小系数多项式，计算 $b_i=a_i\cdot s+e_i\in\mathbf{R}_q$，最后从 b_{m+1} 中减去 $z\cdot\lceil q/2\rfloor$。输出密文 $(b_1, \cdots, b_{m+1})\in\mathbf{R}_q^{m+1}$。

Dec($(r_1, \cdots, r_m, r_{m+1})$, $(b_1, \cdots, b_{m+1})$)。计算 $\sum r_i\cdot b_i\approx z\cdot\lceil q/2\rfloor+(\sum r_i\cdot a_i)\cdot s=z\cdot\lceil q/2\rfloor+0\cdot s\in\mathbf{R}_q$，如果 z 的系数 z_i 比 $\lceil q/2\rfloor$ 更接近 0，输出 0；否则输出 1。

这个方案的安全性证明可以直接从对偶 LWE-PKE 方案的安全性证明中转化得到，这里不多赘述。

4　基于格的公钥密码方案比较

本文介绍的几种基于格的公钥密码方案，它们各有优缺点。其中 AD 方案是首个基于格的公钥密码方案，它具备可证明安全性，但由于效率方面的原因，它并不实用。GGH 方案在效率方面较 AD 略有改进，但它在安全方面存在缺陷，尽管它并不实用，但由于方案的简单和直观，经常作为讨论基于格的公钥密码方案的出发点。NTRU 是目前为止最为高效、实用的基于格的公钥密码方案，甚至是最快的公钥密码方案，尽管缺乏严格的安全性证明，但它对系统要求极低，不需要较高的计算能力以及较复杂的硬件设备的特点，使它成为对 RSA 的公钥密码方案的有益补充。LWE-PKE 方案兼顾了效率与安全性，具有严格的安全性证明，效率略低于 NTRU。而且由于 LWE 问题的代数特性，基于 LWE 的密码原语已成为构建其他密码系统的工具。表 1 是几种方案在效率方面的对比。

表 1　基于格的公钥密码方案效率对比

	AD	GGH	NTRU	LWE-PKE	Ring-LWE-PKE
密钥长度	$\tilde{O}(n^4)$	$O(n^3)$	$O(n\log n)$	$\tilde{O}(n^2)$	$\tilde{O}(n)$
密文膨胀率	$\tilde{O}(n^2)$	$\tilde{O}(n^2)$	$O(1)$	$\tilde{O}(n)$	$\tilde{O}(1)$

5　结束语

本文按照时间顺序介绍了几种主要的基于格的公钥密码方案，从最早提出的 AD 方案开始，到稍晚提出

的 GGH 和 NTRU 以及近些年非常流行的基于 LWE 的公钥密码方案，对他们的设计思想、具体方案进行了详细的阐述，并对其安全性进行了讨论，最后从效率、安全性等方面对它们进行了比较。

参考文献：

[1] Ajtai M. Generating hard instances of lattice problems[C]// ACM Symposium on Theory Of Computing(STOC). New York: ACM Press, 1996: 99-108.

[2] Ajtai M, Dwork C. A public-key cryptosystem with worst-case/average-case equivalence[C]// ACM Symposium on Theory of Computing(STOC). EI Paso: ACM Press, 1997: 284-293.

[3] 杨 明，王兆丽，韩敬利. 基于格的密码学概述[J]. 军事通信技术，2014，35(1)：67-74.

[4] Nguyen P, Stern J. Cryptanalysis of the ajtai-dwork cryptosystem[C]// Crypto'98. Santa Barbara, USA: Springer, 1998: 223-242.

[5] Goldreich O, Goldwasser S, Halevi S. Public key cryptosystems from lattice reduction problems[C]// Crypto'97. Santa Barbara, USA: Springer, 1997: 112-131.

[6] Phong Q, Nguyen P. Cryptanalysis of the goldreich-goldwasser-halevi cryptosystem from Crypto'97[C]// Crypto'99. Santa Barbara, USA: Springer, 1999: 288-304.

[7] Lee M S, Hahn S G. Analysis of the GGH cryptosystem[C]// First International Conference on Symbolic Computation and Cryptography(SCC 2008). Beijing, China: Springer, 2008: 102-217.

[8] Hoffstein J, Pipher J, Silverman J. NTRU: a ring-based public key cryptosystem[C]// Algorithmic Number Theory Symposium(ANTS). Portland: Springer, 1998: 267-288.

[9] Jaulmes E, Joux A. A chosen ciphertext attack on NTRU[C]// Crypto'00. Santa Barbara, USA: Springer, 2000: 23-26.

[10] Howgrave-Graham N A, Nguyen P Q, Pointcheval D, et al. The impact of decryption failures on the security of NTRU encryption[C]// Crypto'03. Santa Barbara, USA: Springer, 2003: 226-246.

[11] Hoffstein J, Howgrave-Graham N A, Pipher J, et al. Hybrid lattice reduction and meet in the middle resistant parameter selection for NTRU Encrypt[EB/OL]. (2012-06-10)[2013-11-23]. http: // www. grouper. ieee. org/groups/1363/lattPK/Submissions. html.

[12] Regev O. On lattices, learning with errors, random linear codes, and cryptography[C]//ACM Symposium on Theory of Computing(STOC). Baltimore: ACM Press, 2005: 84-93.

[13] Lindner R, Peikert C. Better key sizes(and attacks) for LWE-based encryption[C]// Cryptology CT-RSA 2011. Berlin: Springer, 2011: 319-339.

[14] Albrecht M R, Darlos C. On the complexity of the BKW algorithm on LWE[C]// Science Business Media. New York: Springer, 2013: 109-120.

[15] Kawachi A, Tanaka K, Xagawa K. Multi-bit cryptosystems based on lattice problems[C]// Public Key Cryptography (PKC). Berlin: Springer, 2007: 315-329.

[16] Peikert C, Vaikuntanathan V, Waters B. A framework for efficient and composable oblivious transfer[C]// Crypto'08, Berlin: Springer, 2008: 554-571.

[17] Gentry C, Peikert C, Vaikuntanathan V. Trapdoors for hard lattices and new cryptographic constructions[C]// 40th Annual ACM Symposium on Theory Of Computing(STOC 2008). Victoria BC, Canada: ACM Press, 2008: 197-206.

[18] Lyubashevsky V, Peikert C, Regev O. On ideal lattices and learning with errors over rings[C]// Crypto'10. French Riviera: Springer, 2010: 1-23.