密码学报 ISSN 2095-7025 CN 10-1195/TN
Journal of Cryptologic Research,2014,1(3):225–234

E-mail: jcr@cacrnet.org.cn
http://www.jcr.cacrnet.org.cn
Tel/Fax: +86-10-81033101

模背包向量问题的实际复杂度与基于格密码体制的实际安全性*

彭力强[1,2], 胡 磊[1,2], 黄章杰[1,2], 许 军[1,2]

1. 中国科学院信息工程研究所 信息安全国家重点实验室, 北京 100093
2. 中国科学院数据与通信保护研究教育中心, 北京 100093
通讯作者: 胡磊, E-mail: hu@is.ac.cn

摘 要: 背包问题常被用来构造公钥密码算法, 它是公钥密码学中的一个研究热点, 模背包向量问题是同时求解若干个在模意义下的背包问题. 本文将模背包向量问题转化为格中短向量的求解. 我们利用LLL、BKZ 等格基约化算法或它们的联合方法求解目标向量, 实际地解决了维数较小时的模背包向量问题, 讨论了关于模背包向量问题的安全标准, 并展示了由模背包向量问题引出的格的 Hermite 因子随维数的变化关系. 我们的实验结果, 一方面验证了我们的理论分析, 成功地在格维数较小时求解出了目标向量, 即模背包向量问题在维数较小时可解; 另一方面, 由目标向量在维数较大的格中未被找到可以看出, 格基约化算法在求解格中短向量问题的计算能力受维数的限制. 随着格维数的变大, 格基约化算法的运行时间指数级增长并且找到目标向量的概率减小. 另外, 我们通过具体的实验数据, 验证并说明了格基约化算法中参数选取对实验结果产生的影响. 对于 CANS 2011 会议上提出的一个基于格与背包问题混合设计的公钥加密方案, 我们将针对该方案的唯密文攻击转化为模背包向量问题的求解, 从而在唯密文攻击下实际地攻破了该方案的一个推荐参数 $m=100$.

关键词: 模背包向量问题; 格基约化算法; 唯密文攻击; 实际安全性
中图法分类号: TP309.7 **文献标识码:** A DOI: 10.13868/j.cnki.jcr.000021

中文引用格式: 彭力强, 胡磊, 黄章杰, 许军. 模背包向量问题的实际复杂度与基于格密码体制的实际安全性[J]. 密码学报, 2014, 1(3): 225–234.
英文引用格式: Peng L Q, Hu L, Huang Z J, Xu J. Actual complexity of modular knapsack vector problem and practical security of a lattice based public key cryptosystem[J]. Journal of Cryptologic Research, 2014, 1(3): 225–234.

Actual Complexity of Modular Knapsack Vector Problem and Practical Security of a Lattice Based Public Key Cryptosystem

PENG Li-Qiang[1,2], HU Lei[1,2], HUANG Zhang-Jie[1,2], XU Jun[1,2]

1. State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
2. Data Assurance and Communication Security Research Center, Chinese Academy of Sciences, Beijing 100093, China
Corresponding author: HU Lei, E-mail: hu@is.ac.cn

* 基金项目: 国家重点基础研究发展计划(973计划)(2013CB834203); 国家自然科学基金项目(61070172); 中国科学院战略性先导科技专项(XDA06010702)
收稿日期: 2014-03-08 定稿日期: 2014-05-13

Abstract: Knapsack problem is usually utilized to design asymmetric cryptographic algorithms, and it is a research hotspot of public key cryptography, the modular knapsack vector problem is the problem of solving several knapsack problems simultaneously in the sense of modulo operation. In this paper, this problem is converted into finding short vectors in some lattices. We show that the modular knapsack vector problem can be practically solved for small dimensions by using the LLL or BKZ algorithm or their combination, and discuss the security standard of the modular knapsack vector problem and show the relationship between the Hermite factor and the dimensions of the lattices which are derived from the modular knapsack vector problem. Our experiments coincide with the theoretical analysis and can solve the target vector for small dimensions, namely the modular knapsack vector problem can be solved for small dimensions. On the other hand, due to the failure to find target vectors in lattices with large dimensions, we can see that the computation capability of lattice basis reduction algorithms is restricted by the dimensions of lattices. When the dimension increases, the running time of lattice basis reduction algorithm increases exponentially and the probability of success for finding out the target vector decreases. Furthermore, based on the experiments, we verified and showed that the choices of parameters of algorithms also affect the experimental results. For a lattice-knapsack mixed public key cryptosystem proposed at the CANS 2011 conference, when converted into modular knapsack vector problem, we can practically break the cryptosystem under ciphertext-only attack for one of its recommended parameters $m=100$.

Key words: modular knapsack vector problem; lattice basis reduction; ciphertext-only attack; practical security

1 引言

子集和问题定义为: 给定若干正整数 $a_1,a_2,\cdots,a_n$ 以及 s，求解 $x_1,x_2,\cdots,x_n\in\{0,1\}$，使得 $s=\sum_{i=1}^{n}x_ia_i$ 成立. 子集和问题也被称为背包问题. 背包问题是 NP 完全问题，是理论计算机科学中一个非常重要的研究课题. 由于理论计算机科学与公钥密码学紧密联系，许多研究学者尝试利用背包问题来构造安全的公钥密码算法.

1978 年, Merkle 和 Hellman 提出了第一个基于背包问题设计的公钥密码算法[1]. 在公钥密码学发展初期，由于背包算法在加密和解密运算中的高效率，基于背包问题设计的公钥密码算法吸引了众多密码学家的关注. 然而，Shamir 在 1982 年提出了一个算法，成功地在多项式时间内攻破了 Merkle-Hellman 密码体制[2]. 紧接着，利用格在密码学中的应用[3], Largarias 和 Odlyzko 利用 LLL 格基约化算法实现的 SVP 预言机，实际地计算了背包密度小于 0.6463 的背包问题，其中背包密度定义为 $n/\max\{\log_2 a_i:1\leqslant i\leqslant n\}$. 不久之后，Coster 等学者进一步将可解的背包密度上界提升到了 0.9408[4]. 另一方面，出于安全性考虑，公钥加密体制所选用的背包密度一般都大于 1，但这时密码方案解密的唯一性很难被保证.

近些年，随着量子计算机研究的不断发展，以及 Shor 提出的量子算法成功地攻击了基于数论问题所设计的 RSA、ECC 等被广泛应用的公钥密码算法[5]，因此如何设计出基于其他困难性问题的密码算法再次成了学者的研究热点. 其中，由于量子计算似乎对加速求解背包问题影响不大，背包问题又重新被考虑用来设计公钥密码算法.

2011 年，Pan 等学者提出了一个新的格与背包问题混合的公钥加密方案[6]，我们称其中的背包问题为模背包向量问题. 不久之后，Xu 等学者证明了该密码体制在广播攻击和重放攻击下是不安全的(重放攻击是广播攻击的一种特殊形式)[7]. 广播攻击即是针对同一明文采用不同的密钥对该明文进行多次加密，从所得到的多个密文中恢复出明文的攻击方式. 如何在单一密文下，恢复出相应明文，这是唯密文攻击. 针对 Pan 等学者所设计密码体制的唯密文攻击，就是对模背包向量问题的求解. 在本文中，我们将模背包向量问题的求解转化为对与背包向量有关的格中短向量的寻找. 我们通过实验验证了求解模背包向量问题的实际复杂度，根据我们的实验结果，LLL 格基约化算法或 BKZ 格基约化算法[8,9]可实际地求解维数较低的

模背包向量问题. 可解的维数中包括一个原加密体制推荐的安全参数[6], 这也就意味着对于这个参数, 该密码体制在唯密文攻击下可被实际地攻破. 此外, 我们提出了一种联合 NTL 中实现的 BKZ 算法与 Magma 中实现的 LLL 算法的方法, 可以更高效地求解出了目标向量.

本文组织如下: 在第 2 节中, 我们简要地回顾了密码体制[6]以及一些关于格的预备知识. 在第 3 节中, 我们把模背包向量问题转化为格中短向量的计算, 然后利用 LLL 算法和 BKZ 算法分别求解. 在第 4 节中, 我们通过删减掉格 $\boldsymbol{L}$ 中若干列, 来构造一个维数较小的格来求解目标向量. 最后一节为总结.

2 模背包向量问题以及格的预备知识

2.1 Pan等学者的密码体制及模背包向量问题

下面我们简要描述 Pan 等学者所设计的公钥密码体制. 对于该方案具体设计细节, 请参考文献[6]. 我们用参数 m 来描述该密码体制中向量以及矩阵的维数.

密钥生成: 随机选取素数 p, 满足 $p \approx 2m$. 根据文献[6]中的特定方法, 生成一个 $m \times m$ 阶矩阵 $\boldsymbol{H}$. $\boldsymbol{H}$ 在有限域 F_p 上是可逆的, 所包含分量均为 $[0, p)$ 间的整数(在我们的分析中, 我们没有利用 $\boldsymbol{H}$ 的特性). 该密码体制的公钥为 $(p, \boldsymbol{H})$.

加密: 对于给定明文 $t \in \{0,1\}^m$, 随机选择向量 $r \in \{0,1\}^m$. 密文 c 计算如下:

$$c = \boldsymbol{H}t + r \pmod p$$

给定 $\boldsymbol{H}$ 和 p, 从密文 c 中恢复出 t 和 r 即为对 Pan 密码体制的唯密文攻击. 该问题是由 m 个在模 p 意义下的背包问题组成:

$$c = \boldsymbol{H}_{i,1}t_1 + \boldsymbol{H}_{i,2}t_2 + \cdots + \boldsymbol{H}_{i,m}t_m + r_i \pmod p, \quad (i = 1,2,\cdots,m) \tag{1}$$

这个可以被认为是一个模背包向量问题, 其中, $\boldsymbol{H}_{i,j}, r_i, t_i$ 和 c_i 表示 $\boldsymbol{H}, r, t$ 和 c 中的分量.

上述问题可以被看成对一个线性码的错误译码, 校验矩阵为 $(\boldsymbol{H}, \boldsymbol{I})$, 其中错误向量的分量被限定为 0 或 1[7]. 针对该线性码的解码, 我们不限制错误向量所包含的错误数(即错误向量的汉明距离), 只要求错误向量中的分量为 0 或 1. 文献[7]的作者将此问题考虑为对一个含有 m 个变量、 $2m$ 个方程的二次方程组求解, 方程组如下所示:

$$\begin{cases} x_1^2 - x_1 = 0 \\ \cdots \\ x_m^2 - x_m = 0 \\ (c_1 - h_{11}x_1 - \cdots - h_{1m}x_m)^2 - (c_1 - h_{11}x_1 - \cdots - h_{1m}x_m) = 0 \\ \cdots \\ (c_m - h_{m1}x_1 - \cdots - h_{mm}x_m)^2 - (c_m - h_{m1}x_1 - \cdots - h_{mm}x_m) = 0 \end{cases}$$

他们尝试利用如 XL、Fix-XL 以及 ElimLin 等线性化方法对该方程组求解, 但没有得到有效的方法. 我们在第 3 节、第 4 节中介绍如何利用格的方法对此问题进行求解.

2.2 格与Hermite因子

令 $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_n$ 为 R^n 中 n 个线性无关的向量. n 维格 $\boldsymbol{L}$ 由 $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_n$ 张成, 是所有整系数线性组合的集合, 即 $c_1\boldsymbol{\omega}_1 + c_2\boldsymbol{\omega}_2 + \cdots + c_n\boldsymbol{\omega}_n$, 其中 $c_1, c_2, \cdots, c_n \in Z$. 我们称 $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_n$ 为格 L 的一组基. 任意维数大于 1、非平凡的格都有无数组格基.

在随机归约下，计算格中的最短向量是 NP 困难问题. 因此，如何设计出能计算较短且近似正交的格基约化算法取代了计算最短向量问题，成为了研究热点. 两个广泛应用的算法，LLL 算法和 BKZ 算法被发明，其中 LLL 格基约化算法由 Lenstra 等学者提出[8]，该算法是第一个成功地输出指数因子近似最短向量的多项式算法，利用分组技术实现的 BKZ 格基约化算法[9]在实际应用中效果更好. 这两种算法的实际比较可以在文献[10,11]中找到.

在此我们简要回顾 Gama 和 Nguyen[10]以及 Schneider 和 Buchmann[11]关于格基约化算法实际运行效率的分析工作. 他们利用 Hermite 因子来评估格基约化算法的质量，其中 Hermite 因子定义为 $\|b_1\|/\det(\boldsymbol{L})^{1/n}$，$b_1$ 为格基约化算法所输出的第一个短向量.

为了解决这一问题，他们对随机格分别应用不同的格基约化算法进行实验. 根据实验结果，随机格的 Hermite 因子可以粗略地由格基约化算法确定. 对于随机的格，Hermite 因子似乎是关于格维数的指数函数，即 Hermite 因子可表示为 c^{an+b}，其中 n 为格的维数，a、b、c 为常数[10]. 底数 c 会随着格基约化算法的改变而变化. 例如，LLL 算法的 Hermite 因子大于 BKZ 算法的 Hermite 因子. 同时，BKZ 算法中参数 blocksize 的选择也会影响 Hermite 因子的值，blocksize 越大，Hermite 因子的值越小. 另外，一些结构特殊的格的 Hermite 因子可能会更小.

3 模背包向量问题的实际复杂度分析：基本方法

在本节中，我们把模背包向量问题转化为格中短向量的计算，然后利用 LLL 算法和 BKZ 算法分别求解. 在本节最后，我们展示了求解该问题的实际复杂度.

3.1 理论分析

我们构造一个 $2m+1$ 维的格 $\boldsymbol{L}$，它由下面矩阵的行向量生成:

$$\begin{pmatrix} \boldsymbol{I}_{m\times m} & 0_{m\times 1} & \boldsymbol{H}^{\mathrm{T}}_{m\times m} \\ 0_{1\times m} & 1 & c \\ 0_{m\times m} & 0_{m\times 1} & p\boldsymbol{I}_{m\times m} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 & \boldsymbol{H}_{1,1} & \boldsymbol{H}_{2,1} & \cdots & \boldsymbol{H}_{m,1} \\ 0 & 1 & \cdots & 0 & 0 & \boldsymbol{H}_{1,2} & \boldsymbol{H}_{2,2} & \cdots & \boldsymbol{H}_{m,2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \boldsymbol{H}_{1,m} & \boldsymbol{H}_{2,m} & \cdots & \boldsymbol{H}_{m,m} \\ 0 & 0 & \cdots & 0 & 1 & c_1 & c_2 & \cdots & c_m \\ 0 & 0 & \cdots & 0 & 0 & p & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & p & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & p \end{pmatrix}$$

显然地由公式(1)我们能得到

$$-r_i = \boldsymbol{H}_{i,1}t_1 + \boldsymbol{H}_{i,2}t_2 + \cdots + \boldsymbol{H}_{i,m}t_m - c_i + k_i p \quad (i = 1,2,\cdots,m)$$

其中 k_i 为未知的整数，这也就意味着，向量

$$\boldsymbol{\omega} = (t_1,\cdots,t_m,-1,-r_1\cdots,-r_m)$$

属于格 $\boldsymbol{L}$. 由于 $t_i, r_i \in \{0,1\}$ $(i = 1,2,\cdots,m)$，向量 $\boldsymbol{\omega}$ 长度的上界可以被定为 $\sqrt{2m+1}$.

另一方面，$2m+1$ 维格 $\boldsymbol{L}$ 的体积为 $\mathrm{vol}(\boldsymbol{L}) = p^m$. 明显地我们有

$$\sqrt{2m+1} \leqslant \sqrt{2m+1}\left(p^m\right)^{1/(2m+1)} = \sqrt{2m+1}\,\mathrm{vol}(\boldsymbol{L})^{1/(2m+1)}$$

根据 Gaussian 启发式，向量 $\boldsymbol{\omega}$ 有很大概率为格 $\boldsymbol{L}$ 的最短向量，我们利用 LLL 算法或 BKZ 算法分别求解格

$\boldsymbol{L}$ 中的短向量.

3.2 实验结果

我们在一台配置为 Intel(R) Core(TM) Quad CPU(2.67GHz, 2.00GB RAM, Windows 7)的个人计算机上分别运行 2.11-版本的 Magma[12]以及 5.5.2-版本的 NTL[13]进行了上述实验.

Magma 中 LLL 算法的实验结果:

在实验中我们发现, 当格的维数相对较小时, Magma 中的 LLL 算法输出的约化基能够包含目标向量, 但随着格维数的增加, LLL 算法的输出往往不再包含想要的结果. 然而, 如果我们对前一次 LLL 算法输出的结果再次运用 LLL 算法, 新得到的约化基更接近我们想要的结果. 当 $m\leqslant 80$ 时, 我们能够通过这种迭代的方法, 成功地计算出目标向量; 当 $m>80$ 时, 运算量似乎超出了我们个人计算机的运算能力. 参数 δ 在 Magma 中的默认值为 0.75, 它可以被设定为 $(0.25,1)$ 中的任意值. 众所周知, δ 的选取会明显地影响到 LLL 算法的运行时间以及输出向量的长度. 根据我们的实验结果, 当 δ 越大时, 尽管增加了每次 LLL 算法的运行时间, 但降低了得到目标向量所需要的迭代次数, 减小了总体运行时间.

表 1 和图 1 显示了迭代次数、运行时间与 m 的关系. 对每一个 m, 我们做了 5 次随机实验.

表 1 平均迭代次数与平均运行时间($\delta=0.99$)

Table 1 Number of iterations and running time ($\delta=0.99$, averagely)

m	p	迭代次数	时间(秒)	m	p	迭代次数	时间(秒)
30	61	1.0	0.068	60	127	47.6	13.900
35	71	1.2	0.128	65	131	503.4	169.617
40	79	2.0	0.247	70	149	2552.2	1385.420
45	89	2.4	0.406	75	157	38157.3	28767.216
50	103	8.0	1.401	80	167	380866.5	431501.067
55	113	26.2	4.683				

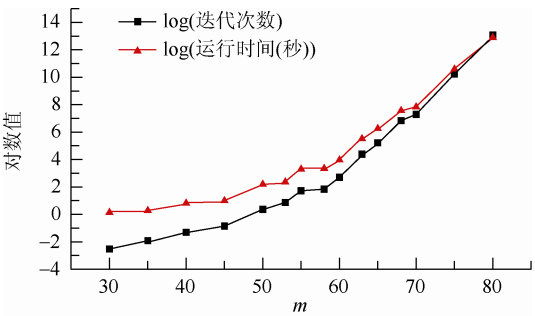

图 1 迭代次数与运行时间取对数后, 关于 m 的函数

Figure 1 Number of iterations and running time in logarithmic forms as functions of m

NTL 中 BKZ 算法的实验结果:

我们同样用 BKZ 算法对格 $\boldsymbol{L}$ 进行实验. 利用 NTL 中实现 BKZ_QP 算法, 我们发现当格维数相对较大时, BKZ_QP 算法与 Magma 中 LLL 算法相比, 能更有效地求解出目标向量. BKZ 算法可选参数 blocksize 指定了约化算法中分块的大小. blocksize 越大, BKZ 算法所输出的短向量长度越小, 但算法的运行时间会由 blocksize 指数增长. 我们设置 blocksize = 20. 根据我们结果, 此时输出向量的长度以及算法的运行时间均可接受. 在实验中, 我们观察到参数 δ 值越大, BKZ 算法输出结果中包含目标向量的概率越高, 但算法的运行时间也会相应变长. 在实际实验中, 我们选取 $\delta=0.99$ 和 $\delta=0.999999$ 分别进行了实验. BKZ_QP 算

法的另一个参数 prune 对减小算法的运行时间似乎作用不大，因此我们采用默认设置 prune = 0. 表 2 以及图 2 展示了我们的实验结果.

表 2 平均运行时间(blocksize = 20, prune = 0)

Table 2 Average running time (blocksize = 20, prune = 0)

m	p	$\delta = 0.99$		$\delta = 0.999999$	
		时间(秒)	成功率(%)	时间(秒)	成功率(%)
70	149	1188.55	100	2654.29	100
75	157	1531.84	100	3476.15	100
80	167	2396.04	100	3785.95	100
85	173	3902.16	80	5972.62	100
90	191	3935.79	40	8966.89	60
95	191	8132.15	20	16427.05	80

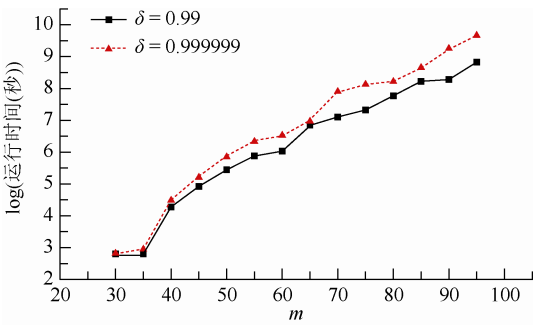

图 2 运行时间取对数后，关于 m 的函数

Figure 2 Running time in logarithmic forms as functions of m

在实验中我们发现存在两组例外的参数 $(m, p) = (55,113)$ 和 $(60,127)$，针对这两组参数，我们总是没有找到目标向量. 然而，我们稍微地改变 p 的取值，目标向量就能够被成功地求出. 我们不知道如何解释这一现象. 另外，p 的值越大，算法输出目标向量的概率越高，这与 NTRU 的安全性分析类似[14].

与此同时，我们也用 NTL 中实现的 LLL 算法进行了实验，但算法输出目标向量的概率太低. 因此我们只选用 NTL 中实现的 BKZ 算法对模背包向量问题进行求解.

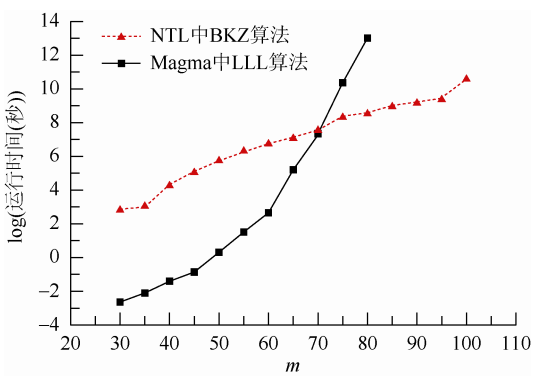

图 3 运行时间取对数后，关于 m 的函数

Figure 3 Running time in logarithmic forms as functions of m

通过与 Magma 中 LLL 算法的实验结果对比, 我们发现当维数较小时, Magma 中的 LLL 算法能够更快地计算出目标向量, 反之, 当 m 较大时, NTL 中的 BKZ 算法更有效. 上面的图 3 展示了这一点.

格 $\boldsymbol{L}$ 的 Hermite 因子的实际估计:

根据我们的实验结果, 如图 4 所示, 我们用 NTL 中的 BKZ 算法对格 $\boldsymbol{L}$ 的 Hermite 因子进行估计, 其中 BKZ 算法的参数 blocksize = 20. Hermite 因子的平均值在 m 取 85 时有大幅增加.

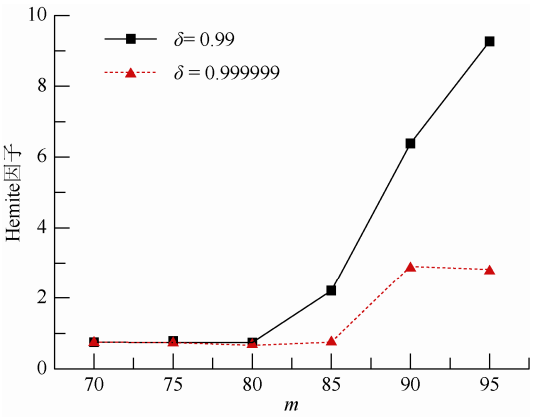

图 4 Hermite 因子关于 m 的函数

Figure 4 The Hermite factor as functions of m

4 模背包向量问题的实际复杂度分析: 扩展方法

根据 May 对 NTRU 方案的攻击思路[15], 我们可以通过删减掉格 $\boldsymbol{L}$ 中若干列, 来构造一个维数较小的格来求解目标向量.

4.1 理论分析

我们构造一个 $2m+1-l$ 维的格 $\boldsymbol{L}_1$, 它由下面矩阵的行向量生成:

$$\begin{pmatrix} 1 & 0 & \cdots & 0 & 0 & \boldsymbol{H}_{1,1} & \boldsymbol{H}_{2,1} & \cdots & \boldsymbol{H}_{m-l,1} \\ 0 & 1 & \cdots & 0 & 0 & \boldsymbol{H}_{1,2} & \boldsymbol{H}_{2,2} & \cdots & \boldsymbol{H}_{m-l,2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \boldsymbol{H}_{1,m} & \boldsymbol{H}_{2,m} & \cdots & \boldsymbol{H}_{m-l,m} \\ 0 & 0 & \cdots & 0 & 1 & c_1 & c_2 & \cdots & c_{m-l} \\ 0 & 0 & \cdots & 0 & 0 & p & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & p & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & p \end{pmatrix}_{(2m+1-l)\times(2m+1-l)}$$

其中, $1 \leqslant l < m$.

类似地, 向量 $\boldsymbol{\omega}_1 = (t_1, \cdots, t_m, -1, -r_1, \cdots, -r_{m-l})$ 属于 $\boldsymbol{L}_1$ 的. 同 3.1 节所讨论的, LLL 算法或 BKZ 算法输出的约化基中有很大概率包括向量 $\boldsymbol{\omega}_1$.

4.2 实验结果

与第 3 节的实验环境相同，我们进行了上述实验.

Magma 中 LLL 算法的实验结果:

成功恢复出目标向量所需要运行 Magma 中 LLL 算法的迭代次数如表 3 所示，我们分别选取了 $l \approx m/3$ 以及 $l \approx m/6$. 需要注意的是，l 的取值未必是最优的.

表 3 平均迭代次数与运行时间($\delta = 0.99$)

Table 3 Average number of iterations and running time ($\delta = 0.99$)

m	p	l	迭代次数	时间(秒)	m	p	l	迭代次数	时间(秒)
60	127	0	47.6	13.90	80	167	0	380866.5	431501.07
		10	59.2	8.90			14	128024.6	58169.23
		20	49.8	4.47			27	35936.8	7546.70
70	149	0	2552.2	1385.42					
		12	837.8	213.77					
		23	937.4	132.34					

从表 3 可以看出，利用约减后的格 $\boldsymbol{L}_1$ 计算目标向量，有效地减少了所需的迭代次数，减少了算法运行所需的总时间.

NTL 中 BKZ 算法的实验结果:

在这里，表 4 列举了我们对格 $\boldsymbol{L}_1$ 运用 NTL 中 BKZ 算法的实验结果. 与 3.2 节中未约减格 $\boldsymbol{L}$ 的实验结果相比较，BKZ 算法的运行时间减少了，但成功概率，即约化基包含目标向量的概率降低了，尤其当 m 较大时. 在实验中，我们选取尽可能大的 l_1 和 l_2，同时保证成功概率是可以接受的.

表 4 平均运行时间与成功概率(blocksize = 20, prune = 0)

Table 4 Average running time and success rates(blocksize = 20, prune = 0)

m	p	l_1	时间(秒)	成功率(%)	l_2	时间(秒)	成功率(%)
80	167	13	2538.92	100	26	1648.11	60
85	174	14	4269.15	100	28	2547.82	40
90	191	15	6581.60	100	30	4518.93	40
95	191	13	11429.29	40	15	13927.88	50
100	199	10	13792.02	10			

NTL 中 BKZ 算法与 Magma 中 LLL 算法相联合的实验结果:

对于 BKZ 算法失败的实例，即算法输出的约化基不包含目标向量，我们对 BKZ 算法的输出再次运用 Magma 中 LLL 算法. 同样地，我们采用迭代 LLL 算法的方法，直到所输出的结果中包含目标向量.

当 $m = 100$ 时，我们选取 $l = m/3 \approx 33$. 表 5 展示了我们的实验结果:

表 5 联合方法的实验结果

Table 5 Result of the combination method

m	p	l	NTL 中 BKZ 算法运行时间(秒)	Magma 中 LLL 算法迭代次数	迭代部分运行时间	总时间(秒)
100	199	33	1041.04	10295.0	1057.37	2098.41

在我们的实验中, 对于 $m=100$, 我们能够在约 35 分钟内成功恢复出明文 t. 这种联合方法在效率上明显地优于单一地使用 NTL 中 BKZ 算法或 Magma 中 LLL 算法. 尽管在 $m=100$ 时, 联合方法似乎在运行时间上有很大的提升, 但我们不知道 m 取值更大时, 如何选取适当的 l, 并且也不知道该方法能否成功得到想要的结果. 例如在 $m=102$ 时, 我们没有找到恰当的l使得联合方法能够成功输出目标向量.

就我们所知, Magma 中 LLL 算法的实现与原始的算法不一样[8], 因此很难估计这种联合方法的计算能力. 但由于在部分参数上的高效率, 联合方法值得我们去进一步的实验研究.

5 结论

在本文中, 我们以模背包向量问题讨论了一个基于格与背包问题混合设计的公钥加密方案在唯密文攻击下的安全性, 我们将模背包向量问题的求解转化为了格中短向量的计算. 我们的实验结果显示, 尽管模背包向量问题在时间复杂度上似乎是渐进困难的, 但对于维数较小的情况, 例如 $m=100$, 模背包向量问题仍是可解的, 即对于推荐参数 $m=100$ 时, 该密码体制在唯密文攻击下可破.

References

[1] Merkle R, Hellman M. Hiding information and signatures in trapdoor knapsacks[J]. IEEE Transactions on Information Theory, 1978, 24(5): 525–530.

[2] Shamir A. A polynomial-time algorithm for breaking the basic Merkle-Hellman cryptosystem[J]. IEEE Transactions on Information Theory, 1984, 30(5): 699–704.

[3] Lagarias J C, Odlyzko A M. Solving low-density subset sum problems[J]. Journal of the ACM, 1985, 32(1): 229–246.

[4] Coster M J, Joux A, LaMacchia B A, et al. Improved low-density subset sum algorithms[J]. Computational Complexity, 1992, 2(2): 111–128.

[5] Shor P W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer[J]. SIAM Journal on Computing, 1997, 26(5): 1484–1509.

[6] Pan Y B, Deng Y P, Jiang Y P, et al. A new lattice-based public key cryptosystem mixed with a knapsack[C]. In: Proceeding of the 10th International Conference on Cryptology and Network Security—CANS. Springer Berlin Heidelberg, 2011. 126–137.

[7] Xu J, Hu L, Sun S W, et al. Cryptanalysis of a lattice-knapsack mixed public key cryptosystem[C]. In: Proceeding of the 11th International Conference on Cryptology and Network Security—CANS. Springer, 2012. 32–42.

[8] Lenstra A K, Lenstra H W, Lovász L. Factoring polynomials with rational coefficients[J]. Mathematische Annalen, 1982, 261(4): 515–534.

[9] Schnorr C P, Euchner M. Lattice basis reduction: improved practical algorithms and solving subset sum problems[J]. Mathematical Programming, 1994, 66(1/3): 181–199.

[10] Gama N, Nguyen P Q. Predicting lattice reduction[C]. In: Advances in Cryptology—EUROCRYPT 2008. Springer Berlin Heidelberg, 2008: 31–51.

[11] Schneider M, Buchmann J. Extended lattice reduction experiments using the BKZ algorithm[J]. Sicherheit 2010, 170: 241–252.

[12] Bosma W, Cannon J, Playoust C. The Magma algebra system I: the user language[J]. Journal of Symbolic Computation, 1997, 24(3/4): 235–265.

[13] Shoup V. NTL: a library for doing number theory, version 5.4[EB/OL]. http: //www.shoup.net/ntl/.

[14] Hoffstein J, Pipher J, Silverman J. NTRU: a ring based public key cryptosystem[C]. In: Proceedings of ANTS 1998. Springer Berlin Heidelberg, 1998. 267–288.

[15] May A. Cryptanalysis of NTRU[EB/OL]. http: //www.informatik.uni-frankfurt.de/~alex/crypto.html.

作者信息

彭力强(1988–), 河北邯郸人, 博士研究生. 主要研究领域为公钥密码学.
E-mail: lqpeng@is.ac.cn

胡磊(1967–), 博士, 研究员. 主要研究领域为密码学与信息安全.
E-mail: hu@is.ac.cn

黄章杰(1988–), 福建莆田人, 博士研究生. 主要研究领域为公钥密码学.
E-mail: zjhuang@is.ac.cn

许军(1982–), 安徽六安人, 博士研究生. 主要研究领域为公钥密码学.
E-mail: jxu@is.ac.cn

中国密码学会2014年会通知

中国密码学会2014年会(ChinaCrypt 2014)将于2014年8月27–30日在河南省郑州市召开. 本届年会由中国密码学会主办, 解放军信息工程大学承办. 会议旨在汇聚国内密码领域专家学者、业界精英、在校研究生, 共同探讨密码学最新研究成果、学术动态及发展趋势, 促进密码学各领域间交流与合作, 提高密码人才培养质量, 推动密码学研究与发展. 按照会议安排, 大会届时将邀请国外2名、国内4名密码学界知名专家作特邀报告. 现将会议有关事项通知如下:

主办单位: 中国密码学会
承办单位: 解放军信息工程大学
会议时间: 2014年8月27–30日
会议地点: 河南省郑州市黄河迎宾馆
会议注册: 通过年会网站在线注册
注 册 费:
8月10日前交费: 密码学会会员1300元/1100元(学生)
非会员1500元/1300元(学生)
8月10日后交费: 密码学会会员1500元/1300元(学生)
非会员1700元/1500元(学生)
住 宿: 会议提供郑州市黄河迎宾馆和附近酒店住宿, 费用自理.
联 系 人: 丁文博 13838093103(会议注册)
孙宝忠 13938433299(宾馆预订)
郑群雄 15324992017, crypt2014@163.com(会议论文、自由论坛报名)
会议网址: http://cacr2014.cacrnet.org.cn

中国密码学会
2014 年 6 月 6 日