软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2015,26(10):2696−2719 [doi: 10.13328/j.cnki.jos.004808]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

近似理想格上的全同态加密方案*

古春生[1,2,3]

[1](江苏理工学院 计算机工程学院,江苏 常州 213001)

[2](中国科学技术大学 计算机科学与技术学院,安徽 合肥 230027)

[3](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)

通讯作者: 古春生, E-mail: chunsheng_gu@163.com

摘 要: 构造高效、安全的全同态加密方案目前仍然是一个公开问题.通过扩展近似 GCD 到近似理想格的方法,首先构造一个基于整数上部分近似理想格问题(PAILP)的有点同态加密方案,并使用 Gentry 的引导技术将其转换到全同态加密方案.归约有点同态加密方案的安全性到求解部分近似理想格问题;其次,构造基于 PAILP 的批全同态加密方案和基于近似理想格(AILP)的全同态加密方案;最后,实现基于 PAILP/AILP 的全同态加密方案,并通过计算实验,其结果表明,所提方案比已有方案性能更好.

关键词: 全同态加密;近似理想格问题;近似 GCD;整数分解;稀疏子集和

中图法分类号: TP309

中文引用格式: 古春生.近似理想格上的全同态加密方案.软件学报,2015,26(10):2696−2719. http://www.jos.org.cn/1000-9825/4808.htm

英文引用格式: Gu CS. Fully homomorphic encryption from approximate ideal lattices. Ruan Jian Xue Bao/Journal of Software, 2015,26(10):2696−2719 (in Chinese). http://www.jos.org.cn/1000-9825/4808.htm

Fully Homomorphic Encryption from Approximate Ideal Lattices

GU Chun-Sheng[1,2,3]

[1](School of Computer Engineering, Jiangsu University of Technology, Changzhou 213001, China)

[2](School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China)

[3](State Key Laboratory of Information Security, Institute of Information Engineering, The Chinese Academy of Sciences, Beijing 100093, China)

Abstract: Constructing efficient and secured fully homomorphic encryption is still an open problem. By generalizing approximate GCD to approximate ideal lattice, a somewhat homomorphic encryption scheme is first presented based on partial approximate ideal lattice problem (PAILP) over the integers. The scheme is then converted it into a fully homomorphic encryption scheme (FHE) by applying Gentry's bootstrappable techniques. Next, the security of the somewhat homomorphic encryption scheme is reduced to solving a partial approximate ideal lattice problem. Furthermore, a PAILP-based batch FHE and an AILP-based FHE are constructed. Finally, the PAILP/AILP-based FHE is implemented, and the performance of the proposed scheme is demonstrated to be better than that of previous schemes by computational experimental.

Key words: fully homomorphic encryption; approximate ideal lattice problem; approximate GCD; integer factoring; SSSP (sparse subset sum problem)

* 基金项目: 教育部人文社会科学研究规划基金(14YJAZH023); 江苏省"青蓝工程"项目(KYQ14004); 常州市应用基础研究项目(CJ20140040); 江苏省前瞻性联合研究项目(BY2014038-03); 中国科学院信息工程研究所信息安全国家重点实验室开放课题(2015-MSB-10)

收稿时间: 2013-05-01; 定稿时间: 2014-12-09

1 前 言

自从 Gentry[1]设计第一个全同态加密方案以来,全同态加密方案研究立即成立当前密码学研究热点[2−19].因为全同态加密技术允许对加密数据实行任意计算,所以全同态加密技术能够用于解决云计算中用户数据安全和隐私保护问题.然而,现有全同态加密方案存在密文膨胀率大、密文同态计算复杂性高、方案安全性假设较强等问题,结果导致现有方案在云计算环境中不具有实用性.因此,如何设计实现安全、有效的全同态加密方案,目前仍然是一个公开问题.

1.1 相关研究

Gentry[1]基于理想格问题的全同态加密方案主要组成 3 个关键步骤:Gentry 首先设计支持有限密文乘法和加法的有点同态加密方案,因为每个密文中都存在噪声,并且每次密文同态运算增加新产生密文的噪声,当密文中噪声达到一定阈值时,新产生的密文就不能正确解密,这就限制了有点同态加密方案仅能够计算有界低度的密文多项式;其次,Gentry 压扁解密算法,以便它能被表示为密文和密钥比特的低度多项式;最后,Gentry 引入自举性关键思想,不仅能在密文和密钥比特上计算这个解密多项式,而且能够在那些密钥比特的密文上实行同态计算,产生原密文中明文的一个新密文.

如果解密多项式的度数足够小,则新密文中的噪声能够变得比原密文中的噪声更小,以便这个新密文能够再次进行同态加法或乘法运算.使用这种密文刷新算法后,有点同态加密方案允许同态密文运算的次数变成无限,从而获得了全同态加密方案.

目前,设计全同态加密方案主要有 3 种类型.

(1) 基于理想格的全同态加密方案和实现

为了实现 Gentry 方案,Smart 和 Vercauteren[3]基于主理想格优化设计了全同态加密方案.这种主理想格能够使用两个整数来表示,而且该方案私钥仅为一个整数.Smart 和 Vercauteren 能够实现基本的有点同态加密方案,但他们的方案并不能支持足够大参数条件下的 Gentry 压扁解密技术.因此,他们并没有真正实现全同态加密方案.主要困难在于 Smart 和 Vercauteren 方案的密钥生成算法复杂性太高.因为他们的方案需要产生行列式为素数的主理想格,而且即使发现这种主理想格后,计算其私钥的复杂性至少也得是 $\tilde{\Theta}(n^{2.5})$, n 为格的维数.结果导致他们不能产生维数 n>2028 的密钥.而且,Smart 和 Vercauteren 估计压扁解密多项式的度数将达到几百,故为获得全同态加密方案,需要理想格的维数至少为 n=2^{27}.因此,目前 Smart 和 Vercauteren 方案在实际中无法有效产生密钥.

Gentry 和 Halevi[7]第一次真正实现了 Gentry 全同态加密方案.Gentry 和 Halevi 的方案也是基于主理想格,但仅要求主理想格的行列式为奇数,并不要求其一定是素数.在实现方案时,Gentry 和 Halevi 使用了许多优化方法,包括在文献[3,17]中建议的方法.对于其方案的最安全环境(即 72bit 安全),公钥大小约为 2.3GB,在高端工作站上一次密文刷新需要用时 30 分钟,密文膨胀率约为 n=$O(2^{23})$.

针对文献[3,7]中的方案,Gu 和 Gu[20]给出了求解小倍数私钥的格归约攻击方法.根据文献[21]中计算分析的 LLL 算法[22]的平均性能,当 n≤8192 时,他们的方案并不是安全的,而且文献[20]中也给出了直接恢复明文的格归约攻击方法.使用该攻击方法,我们在中国科技大学超级计算中心的单个处理机上用时 22 天成功恢复出维数 n=256 时密文中的明文.

针对文献[3]中的方案,Hu 和 Wang[23]通过构造修改的私钥、修改的解密算法和特定密文空间子集,给出了新的攻击方法.若密文来自这个特定子集,则该方法使用修改私钥和解密算法能够正确解密.

(2) 基于整数上近似 GCD 问题的全同态加密方案和实现

2010 年,Dijk,Gentry,Halevi 和 Vaikuntanathan(DGHV)[2]构造了基于整数上近似 GCD 问题的全同态加密方案.与 Gentry 方案相比,这个方案的优点在于概念上的简单性,因为它仅在整数上进行计算操作.方案简单性的代价是其公钥大小达到 $\tilde{O}(\lambda^{10})$ (λ为安全参数),以致于对任何实际系统来说其公钥都太大.

代替文献[2]中的方案,使用公钥整数的线性函数加密,Coron,Mandal,Naccache 和 Tibouchi[4]使用公钥整数

的二次函数进行加密,将公钥大小从 $\tilde{O}(\lambda^{10})$ 减少到 $\tilde{O}(\lambda^{7})$. 使用文献[7]中的优化技术,文献[4]中的方案获得了相似的性能,即,公钥大小为 802MB 和一次密文刷新需要用时 14 分钟.因此,文献[4]真正实现了文献[2]中的全同态加密方案,其方案安全性基于近似 GCD 问题的更强变种,即,部分近似 GCD 问题.为了进一步减少文献[2]中方案的公钥大小,Coron,Naccache 和 Tibouchi[5]使用基于 Hash 函数的压缩技术,将方案公钥大小从 $\tilde{O}(\lambda^{7})$ 减少到 $\tilde{O}(\lambda^{5})$, 此时,方案语义安全性仍然基于部分近似 GCD 问题难度,但已控制在随机神谕安全模型下.文献[5]中的方案将公钥由 802MB 减少到 10.1MB,一次密文刷新需要时间为 11 分 34 秒,密文膨胀率为 $\tilde{O}(\lambda^{5})$. 而且,Coron, Naccache 和 Tibouchi[5]还扩展文献[11]中的模切换技术到整数近似 GCD 上,从而构造了无引导的层次 DGHV 型全同态加密方案.为了降低密文膨胀率,类似于文献[3]中的批密文技术,Coron,Lepoint 和 Tibouchi[6]构造了基于中国剩余定理的批全同态加密方案.

针对文献[4]中的方案,Chen 和 Nguyen[24]设计了基于单变量多点计算方法的新 PAGCD 求解算法,其时间和空间为穷举搜索数的平方根,即 $\tilde{O}(2^{\lambda/2})$, 因而从实验上破解了文献[4]中的方案.而且,Chen 和 Nguyen 直接使用 PAGCD 算法给出了求解 AGCD 算法,其时间为 $\tilde{O}(2^{3\lambda/2})$, 空间为 $\tilde{O}(2^{\lambda/2})$. 使用与文献[22]同样的方法,Coron, Naccache 和 Tibouchi[5]给出了运行时间和空间都为 $\tilde{O}(2^{\lambda})$ 的启发式求解 AGCD 算法.

(3) 基于 LWE(Ring-LWE)问题的全同态加密方案和实现

Brakerski 和 Vaikuntanathan 在文献[9,10]中分别构造了基于 Ring-LWE 和 LWE 问题的全同态加密方案.为了降低密文膨胀率和密文同态计算复杂性,Brakerski 和 Vaikuntanathan 引入了新的维数约减技术和模切换技术.理论上,文献[9,10]中的全同态方案公钥大小仅为 $O(n^{3+2\varepsilon}\log^{O(1)}n)$比特,密文大小为 $O(n^{1+\varepsilon})$,但上述参数中隐含的常数因子非常大.实际上,Lauter,Naehrig 和 Vaikuntanathan[25]实现了文献[9]中基于 Ring-LWE 的有点同态加密方案,但并未实现全同态加密方案.根据文献[25],当 Ring-LWE 的维数 n=4096、最大密文度数 D=15 时,公钥大小约为 60MB;当 n=16384、最大密文度数 D=15 时,公钥大小约为 2 946MB.

最近,Brakerski,Gentry 和 Vaikuntanathan[11]基于文献[9,10]中的模切换技术设计了无引导的层次同态加密方案,其安全性基于求解伪多项式因子 $n^{O(\log n)}$的 Ring-LWE 问题难度.假定 Ring-LWE 的维数为 n,噪声大小为 n, 则理论上该方案的公钥大小约为 $O(n^{3}\log^{O(1)}n)$.在这种新的全同态加密方案框架中,通过模切换技术将每次密文乘法时新密文中噪声增长由原来的二次增长降为线性增长,代价是每次密文同态计算后模按比例阶梯减少.因而,这种新框架有可能真正提高实际全同态加密方案的性能.

为了改进在 Ring-LWE 上全同态加密方案的性能,Gentry,Halevi 和 Smart[12]基于特殊数模(如形为整数 2^{k}+1)设计了更好的引导技术,在某些情况下,甚至允许“私钥”加密为一个密文,从而减少公钥大小.在文献[13]中,Gentry, Halevi 和 Smart 使用批密文技术进行密文同态计算,计算 t 门的宽度$\Omega(\lambda)$的电路需要时间为 $t\log^{O(1)}\lambda$. 为了将全同态加密方案应用于实际中,文献[14]使用许多优化方法实现了在全同态加密方案上计算高级加密标准算法 AES-128 电路.使用 NTL 库[26],文献[14]中实现的一个变种方案,花费 36 小时对整个 AES 运算过程进行同态密文计算;运用 SIMD(single-instruction multiple-data)技术,使得每次同态密文计算处理 54 块明文,获得每块明文的同态密文计算补偿时间约为 40 分钟;再进一步加以改进,使得每次同态密文计算能够处理 720 块明文,获得每块明文的同态密文计算补偿时间约为 5 分钟.

在 2012 年,Brakerski[15]基于新的张量技术构造了基于 LWE 的全同态加密方案,并归约其安全性到格的近似因子为 $n^{O(\log n)}$的经典 GapSVP 问题的最坏情况求解难度,n 为格的维数.然而,该方案的密文同态计算复杂性也很大,不具有实用性.最近,Brakerski 和 Vaikuntanathan[18]证明了分层全同态加密方案的安全性在量子计算上归约到近似因子为 $O(n^{1.5+\varepsilon})$(或在经典计算上归约到近似因子为 $O(n^{2+\varepsilon})$)的 GapSVP 问题求解难度.Gentry,Sahai 和 Waters[19]基于近似特征向量法构造了全同态加密方案,其安全性归约到 LWE 假设.GSW 方案的优点在于,概念上更易于理解,理论上渐近性能更好,密文同态加法和乘法计算仅是矩阵加法和乘法.但实际上,其性能要比目前最好的基于 RLWE 的方案[11−15]还差一个对数因子$(\log n)^{O(1)}$.Lopez-Alt[16]构造了基于 NTRU 型的多密钥全同态加密方案,该方案能够用于云环境下实时多方安全计算.

Gu 和 Wu[27]使用文献[10]中的密文乘法技术设计了基于近似格问题的全同态加密方案,由于方案中私钥向

量的极大范数为较小(通常为常数或安全参数的线性函数),为使方案安全,格的维数 n 需要足够大,公钥大小约为 $O(n^3\log^{O(1)}n)$,密文膨胀率也很大.

另外,Gentry 和 Halevi[8]构造了新的全同态加密方案,它是有点同态方案和乘法同态加密方案(如 Elgamal 方案)的一种混合方案.这个方案使用判定 Diffie-Hellman 难度假设替换了 Gentry 方案中的稀疏子集和难度问题(sparse subset sum problem,简称 SSSP)假设,去除了 Gentry 方案中的压扁解密电路步骤.

国内相关研究:据我们所知,目前国内全同态加密方案的研究成果甚少[23],相关研究主要是基于(全)同态加密方案构造新的密码原语[28−31].胡予濮和王凤和[23]提出了针对基于主理想格的全同态加密方案[3]的特定攻击方法.张永、温涛、郭权和李凤坤[28]通过引入全同态加密方案提出了一种对偶密钥建立方案;陈嘉勇、王超、张卫明和祝跃飞[29]采用修正全同态加密算法设计了载密图像加密算法;孙中伟、冯登国和武传坤[30]提出了基于加同态公钥密码算法的匿名数字指纹方案;光炎等人[31]基于 LWE 构造了无证书全同态加密方案.另一方面,针对云计算环境中隐私保护和密文检索等安全问题[32,33],国内密码学者也进行了深入的研究和探索[34−36].然而,这些研究着重于解决云计算安全中的某一特定函数密文计算,并不能实行任意函数的全同态密文计算.

通过分析上述全同态加密方案,可以发现:

- 第 1 种全同态加密方案是基于理想格构造.为保证方案安全性,理想格的维数不能太小,目前在文献[1,3,7,17]中给出的方案维数至少为 1 024 以上;为了能够实现密文同态计算,理想格基的范数必须足够大;
- 第 2 种全同态加密方案是基于整数上近似 GCD 问题.然而为了避免格归约攻击,公钥中每个 x_i 的比特长度至少为 λ^5 以上(λ为安全参数),否则方案就不安全[2,24];
- 第 3 种全同态加密方案基于 LWE/Ring-LWE 问题[37,38],该问题本身可以看作格中 CVP 问题.为了保证方案的安全性,LWE/Ring-LWE 的维数也比较大.为了实行密文同态计算,在基于 LWE/Ring-LWE 的方案中要求模 p 足够大,密文中噪声很小.然而,方案安全性反比于模 p 与噪声的比,即:如果模 p 与密文中噪声的比越大,则越易受到格归约算法的攻击,方案安全性越低.根据当前优化实现的全同态加密方案性能来看,基于 Ring-LWE 的方案效率最高[39−41].

另一方面,近年来格归约算法的研究不断取得进展[21,22,42−51],特别是求解最小格向量的随机筛算法.所以,为了保证方案的安全性,基于格/理想格问题的全同态加密方案的维数必须足够大.

因此,在上述全同态加密方案中,每种方案各有其优缺点.问题是能否设计新的全同态加密方案,继承上述方案中的优点而摒除它们的缺点.这是本文的研究主题.

1.2 本文结果

本文研究的主要结果是设计实现基于多项式环上近似理想格问题的全同态加密方案,并归约证明方案安全性到部分近似理想格问题.DGHV 方案[2]在概念上非常简单,然而为了避免格归约攻击,方案公钥元素必须设置得非常大.本文所提方案通过扩展近似 GCD 问题到多项式环上近似理想格问题方法来避免格归约攻击.

本质上,本文方案类似于文献[52]中使用理想格时的公钥加密方案,这一点与文献[2]中的方案类似于文献[53]中的公钥加密方案一样.实际上,Ajtai 和 Dwork[52]构造了基于多维格的扰动格点的公钥加密方案,而 Regev[53]构造了基于一维格的扰动格点的公钥加密方案,它们的安全性都基于唯一最短向量问题求解难度.与文献[2]中的方案一样,本文方案公钥的格点扰动程度比文献[52]中方案公钥的格点扰动程度更小,导致文献[52]中最坏情况到平均情况的安全归约似乎也不能应用于本文方案.

本文重新分析稀疏子集和问题求解难度,以减少公钥大小并提高同态计算效率.在文献[7,17]中,稀疏子集和问题都假定敌手知道稀疏子集的和.实际上,在全同态加密方案中,这个和是隐藏的.在这种情况下,隐藏稀疏子集和问题并不适用存在生日攻击问题.

1.3 本文结构

本文第 2 节描述相关预备知识.第 3 节构造基于部分近似理想格的有点同态加密方案.第 4 节归约证明有

点同态加密方案安全性到部分近似理想格问题,并讨论已知攻击.第 5 节设计基于部分近似理想格的全同态加密方案.第 6 节构造批全同态加密方案.第 7 节设计基于近似理想格的全同态加密方案.第 8 节实现全同态加密方案,并与已有方案的性能进行比较.第 9 节总结全文,并讨论构造扩展和公开问题.

2 预备知识

2.1 符号约定

设ρ为安全参数,k为 2 的幂,$k=O(\log\rho)$,n为正整数.集合$[n]=\{1,\ldots,n\}$.$\langle x\rangle$表示最接近x的整数.同理,记$\langle v\rangle$为向量(或多项式)v中每个分量(或系数)的最接近整数的向量(或多项式).设Z为整数集,R为整数多项式环$R=Z[x]/\langle x^k+1\rangle$,$R_n=R/nR$.如果定义$R$在实数上,则记为$R_{\mathbb{R}}$.设$\delta$为正整数,$R_{\mathbb{R},\delta}=\{y\in R_{\mathbb{R}}\mid 2^{\delta}y\in R\}$,即,$R_{\mathbb{R},\delta}$为$R_{\mathbb{R}}$中系数的小数位至多$\delta$比特的元素集.对于$\forall u\in R$,设$\|u\|_\infty$为$u$系数向量极大范数.对于$R$,其膨胀因子$\gamma_{mul}$为$n$,即:

$$\|u\times v\|_\infty\leqslant k\cdot\|u\|_\infty\cdot\|v\|_\infty.$$

这里,×是R中乘法.

设$r\leftarrow_\psi S$表示根据分布ψ从集合S中选取元素r.$A\equiv_c B$表示对任意多项式时间算法分布A,B是计算上不可区分的.

2.2 理想格和近似GCD问题

给定m个线性无关向量$b_1,b_2,\ldots,b_m\in Z^n$,格是$b_i$所有整数线性组合的集合$L(b_1,b_2,\ldots,b_m)=\{\sum_{i=1}^m x_ib_i,x_i\in Z\}$.我们也使用矩阵$B\in Z^{n\times m}$表示向量$b_i$.如果$B$为$L$的一个基,则$B$支撑格$L$.对于任何格$L$的基$B$,定义:

$$P(B)=\{Bx\mid x\in\mathbb{R}^m,\forall i:-1/2\leqslant x_i<1/2\}.$$

当$m=n$时,设$\det(B)$表示B的行列式.

设$\bar{u}=(u_0,u_1,\ldots,u_{n-1})^T$为$u\in R$的系数向量,定义$rot(\bar{u})=(-u_{n-1},u_0,\ldots,u_{n-2})^T$,矩阵:

$$Rot(u)=(\bar{u},rot(\bar{u}),\ldots,rot^{n-1}(\bar{u})).$$

称$Rot(u)$为理想格(u)的基.理想$I\subseteq R$称为主理想,如果它能由一个元素生成.

因参数较多,本文假设$\lambda=\tilde{O}(\rho^2),\eta=\tilde{O}(\rho^2),\tau=\tilde{O}(\rho^2)$和$\gamma=\tilde{O}(\rho^5)$.在本文方案实现时,给出各个参数选取的具体值.

设p是比特长度为λ的奇数,定义在Z上比特长度为γ的整数分布$D_{Z,\gamma,\rho}(p)$:

$$D_{Z,\gamma,\rho}(p)=\{b=qp+r\mid q\leftarrow_U Z\cap[0,2^\gamma/p),r\leftarrow_U Z\cap(-2^\rho,2^\rho)\}.$$

定义 2.1(**近似 GCD 问题**,**AGCD**). 给定$D_{Z,\gamma,\rho}(p)$的一列随机抽样$\{b_i=q_ip+r_i,i\in[\tau]\}$,求$p$.

定义 2.2(**部分近似 GCD 问题**,**PAGCD**). 给定$D_{Z,\gamma,\rho}(p)$的一列随机抽样$\{b_i=q_ip+r_i,i\in[\tau]\}$和$b_0=q_0p$,求$p$.

设多项式$h\in R$满足$\|h\|_\infty=2^\lambda$和$h \bmod 2=1$,定义在R上多项式分布$D_{R,\rho,\gamma,\eta}(h)$:

$$D_{R,\rho,\gamma,\eta}(h)=\{f=g\times h+e\mid g,e\in R,\|g\|_\infty\in[2^\eta],\|e\|_\infty\in[2^\rho]\}.$$

定义 2.3(**近似理想格问题**,**AILP**). 给定$D_{R,\rho,\gamma,\eta}(h)$的一列随机抽样$\{f_i=g_ih+e_i,i\in[\tau]\}$,求$h$.

定义 2.4(**部分 AILP 的问题**,**PAILP**). 给定$D_{R,\rho,\lambda,\eta}(h)$的一列随机抽样$\{f_i=g_i\times h+e_i,i\in[\tau]\}$和$f_0=g_0h$,求$h$.

变种 PAILP1. 设$q_0=O(2^{k\eta})$,$p=\det(Rot(h))$为素数.给定$D_{R,\rho,\lambda,\eta}(h)$的一列随机抽样$\{f_i=g_i\times h+e_i,i\in[\tau]\}$和$f_0=q_0p$,求$h$.

变种 PAILP2. 设$p_i=\det(Rot(h_i))$,$i=1,2$为素数,$h=h_1h_2$.给定$D_{R,\rho,\lambda,\eta}(h)$的一列随机抽样$\{f_i=g_i\times h+e_i,i\in[\tau]\}$和$f_0=p_1p_2$,求$h$.

评论 2.1. 当$k=1$时,近似理想格问题变成近似 GCD 问题,PAILP 变成 PAGCD.

评论 2.2. 条件$h \bmod 2=1$仅是为了使全同态加密方案的解密算法简单,在安全归约证明时并不需要这个条件.

评论 2.3. 在 PAILP1 中,要求q_0,p为素数是为了使整数$f_0=q_0p$难以分解.因为$\gcd(h,x^k+1)=(x-\alpha) \bmod p$,即:

在模 p 下,h,x^k+1 有公因式 $x-\alpha$.如果对手已知 p(如分解 $f_0=q_0p$),则在有限域 F_p 上可以有效分解 x^k+1,并进而通过格归约算法求解 h.但对于合数 f_0,目前还没有有效算法[54]分解 x^k+1,也得不到公因式 $x-\alpha$.因此在 PAILP1 中,要求 $f_0=q_0p$ 为两个大素数之积.

评论 2.4. 在 PAILP1 中,因为 $p=p(h^{-1}\times h)=(ph^{-1})\times h$,即,$f_0=q_0p=(q_0ph^{-1})\times h$ 对应于 PAGCD 中无噪声元素 $b_0=q_0p$.这里,h^{-1} 是 h 在实数上的逆,ph^{-1} 是整数上多项式.因此,变种 PAILP1 是 PAILP 的一种特殊情况.另一方面,在 PAILP 中,敌手能够计算 $f_0'=\det(Rot(f_0))=\det(Rot(g_0))\det(Rot(h))=q_0p$, 即,转换 PAILP 到 PAILP1.从评论 3 可知:在 PAILP 中要求 f_0' 难以分解,否则易于通过分解 f_0' 来求解 h.

评论 2.5. 在 PAILP2 中,$f_0=p_1p_2=(f_0h^{-1})\times h$,$f_0h^{-1}$ 是整数上多项式,而且也要求 $f_0=p_1p_2$ 难以分解.否则,当敌手分解 f_0 后,则首先分解 $x^k+1=\prod_{j\in[k]}(x-\alpha_{i,j}) \bmod p_i$;然后,利用中国剩余定理计算 β_{j_1,j_2} 满足:

$$\begin{cases}\beta_{j_1,j_2}=\alpha_{i,j_1} \bmod p_1\\ \beta_{j_1,j_2}=\alpha_{i,j_2} \bmod p_2\end{cases},j_1,j_2\in[k];$$

最后,由 $\gcd(h,x^k+1)=(x-\alpha) \bmod f_0$ 可知:在由两个元素 (p,β_{j_1,j_2}) 生成的 k^2 个理想格中,有一个与理想格 h 相同.又因为 k 较小,LLL 算法一般能够获得理想格 (p,β_{j_1,j_2}) 的最小生成基.因此,在 k^2 个格归约基中有一个与 h 相同.

评论 2.6. 本质上,PAILP1 和 PAILP2 与 PAILP 相同,基于它们设计的全同态加密方案都依赖于大整数分解难度.然而,基于 PAILP1 方案的密文膨胀率最大(约为 $k^2(\lambda+\eta)$),基于 PAILP2 方案的密文膨胀率次之(约为 $k^2\lambda$),基于 PAILP 方案的密文膨胀率最小(约为 $k(\lambda+\eta)$).但 PAILP 除了提供 $f_0=g_0h$ 外,还给定了 $f_0'=\det(Rot(f_0))=q_0p$,即,PAILP 提供给敌手的信息最多.然而目前并不知道在不分解 f_0' 的情况下,如何利用 f_0 求解 PAILP.

2.3 对称多项式(symmetric polynomial)

根据文献[55]中的引理 4,给定比特向量 $\omega=(\omega_1,\omega_2,\ldots,\omega_t)$,其海明权重二进制表示的第 i 比特等于第 2^i 初等对称多项式 $e_{2^i}(\omega) \bmod 2$, 即, $e_{2^i}(\omega) \bmod 2=(\sum_{|S|=2^i,S\subset[t]}\prod_{j\in S}\omega_j) \bmod 2$. 而且,我们能够计算在 ω 上初等对称多项式作为多项式 $P_\omega(x)=\prod_{j=1}^{t}(x-\omega_j)$ 中形式变量 x 的系数,如 $e_{2^i}(\omega) \bmod 2$ 为 x^{t-2^i} 的系数.

计算初等对称多项式的动态规划算法[2].

输入:比特向量 $\omega=(\omega_1,\omega_2,\ldots,\omega_t)$;输出:向量 ω 的分量和 $P_{1,t},P_{2,t},P_{4,t},\ldots,P_{2^t,t}$.

(1) 初始化:设 $P_{0,k}=1,k=0,1,\ldots,t-1$ 和 $P_{j,0}=0,j=1,2,\ldots,2^i$ //$P_{j,k}$ 为在 $\omega_1,\ldots,\omega_k$ 上的第 j 个对称多项式

(2) For k=1 to t //每次循环加入 ω_k,计算产生 $\omega_1,\ldots,\omega_k$ 组成的对称多项式

(3) For $j=2^i$ downto 1

(4) $P_{j,k}=\omega_k\times P_{j-1,k-1}+P_{j,k-1}$

(5) 输出 $P_{1,t},P_{2,t},P_{4,t},\ldots,P_{2^t,t}$.

2.4 剩余Hash引理(leftover Hash lemma)

从有限集 X 到有限集 Y 的 Hash 函数族 H 是两两独立的(pairwise independent),如果对于所有不同的 x,x', $\Pr_{h\leftarrow_R H}[h(x)=h(x')]=1/|Y|$. 分布 D 是 ε-均匀的(ε-uniform),如果它与均匀分布的统计距离至多为 ε.这里,有限集 X 上分布 D_1,D_2 的统计距离是 $\frac{1}{2}\sum_{x\in X}|D_1(x)-D_2(x)|$.

定义 2.5. 从 X 到 Y 的 Hash 函数族 H 是 ε-两两独立的,如果:

$$\sum_{x\neq x'}(\Pr_{h\leftarrow_R H}[h(x)=h(x')]-1/|Y|)\leqslant\frac{|X|^2}{|Y|}\varepsilon.$$

引理 2.1(leftover Hash lemma[56]). 设 H 是从 X 到 Y 两两独立的 Hash 函数族.假定 $h\leftarrow_U H$ 和 $x\leftarrow_U X$ 是独立均匀选择的,则 $(h,h(x))$ 是 $H\times Y$ 上 $\frac{1}{2}\sqrt{|Y|/|X|}$ -均匀的.

引理 2.2(leftover Hash lemma[4]). 设 H 是从 X 到 Y 的 ε-两两独立 Hash 函数族.假定 $h\leftarrow_U H$ 和 $x\leftarrow_U X$ 是独

立均匀选择的,则$(h,h(x))$是 $H\times Y$ 上 $\frac{1}{2}\sqrt{|Y|/|X|+\varepsilon}$ -均匀的.

2.5 全同态加密

下面基于文献[57]自适应地回顾全同态加密方案及其语义安全性的定义.

定义 2.6(同态公钥加密方案 HE). HE 是由 4 个概率多项式时间算法(HE.KeyGen,HE.Enc,HE.Dec,HE.Eval)组成的.

- $HE.KeyGen(1^\rho)$是概率多项式时间算法,输入为安全参数 1^ρ,输出为公钥 pk 和私钥 sk;
- $HE.Enc(pk,m)$是概率多项式时间算法,输入为公钥 pk 和明文比特 m,输出密文 c;
- $HE.Dec(sk,c)$是多项式时间算法,输入为私钥 sk 和密文 c,输入明文比特 m;
- $HE.Eval(pk,C,c_1,c_2,...,c_n)$是多项式时间算法,输入为公钥 pk,电路 C,n 个密文 $c_1,c_2,...,c_n$,输出密文 c.这里,电路 C 输入 n 个比特,输出 1bit.

实际上,任意电路计算都可以转化为由加法门和乘法门组成的电路计算.因此,本文在设计同态加密方案时仅考虑加法门和乘法门同态计算.

定义 2.7(电路保密性(circuit-privacy)和简洁性(compactness)). 电路保密性是指由同态加密中算法 HE.Eval 计算产生的密文不显示超出电路输出本身的电路任何其他内容,即使知道私钥;简洁性是指 HE.Eval 输出密文长度至多为安全参数ρ的多项式 $p(\rho)$大小.

定义 2.8(C-同态). 设 $C=\{C_n\}_{n\in N}$ 为一类布尔电路,这里,C_n 指的是输入 n 个比特,输出 1 个比特的电路集.同态方案 HE 是 C-同态,如果对于足够大安全参数ρ,每个多项式 $n(\rho)$,每个电路 $E\in C_n$ 和每个输出比特序列 m_1, $m_2,...,m_n$:

$$\Pr[(pk,sk)\leftarrow HE.KeyGen(1^\rho);$$
$$c_i\leftarrow HE.Enc(pk,m_i),i\in[n];$$
$$c\leftarrow HE.Eval(pk,E,c_1,c_2,...,c_n);$$
$$HE.Dec(sk,c)\neq E(m_1,...,m_n)]=negl(\rho).$$

这里,概率是在 HE.KeyGen,HE.Enc 随机选择之上.

定义 2.9(全同态加密 FHE). 方案 HE 是全同态加密方案,如果它对于 $GF(2)$上所有算术电路类是同态的.

定义 2.10(IND-CPA 安全). 方案 HE 是 IND-CPA 安全的,如果对任意概率多项式时间敌手 A:

$$|\Pr[(pk,sk)\leftarrow HE.KeyGen(1^\rho):A(pk,HE.Enc(pk,0))=1]-$$
$$\Pr[(pk,sk)\leftarrow HE.KeyGen(1^\rho):A(pk,HE.Enc(pk,1))=1]|=negl(\rho).$$

3 有点同态加密方案(SHE)

跟随 Gentry[1]构造全同态加密方案的框架,我们首先构造基于 PAILP 的有点同态加密方案 SHE;然后,通过使用 Gentry 引导技术转换 SHE 方案到全同态加密方案 FHE.

3.1 SHE构造

密钥生成算法. SHE.KeyGen.

(1) 随机选择多项式 $g_0,h\in R$ 满足 $q_0=\det(Rot(g_0))$,$p=\det(Rot(h))$为素数,且 $h \bmod 2=1$ 和$\|h\|_\infty=2^\lambda$,$\|g_0\|_\infty\in[2^\eta]$,计算 $f_0=g_0h\in R$;

(2) 随机选择两组多项式 $g_{t,j},e_{t,j}\in R,t\in[2],j\in[\tau]$满足$\|g_{i,j}\|_\infty\in[2^\eta]$,$\|e_{i,j}\|_\infty\in[2^\rho]$,计算 $f_{t,j}=hg_{t,j}+e_{t,j}$;

(3) 输出公钥 $pk=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]})$和私钥 $sk=(h)$.

加密算法. SHE.Enc.

给定公钥 pk 和消息比特 $m\in\{0,1\}$,随机选择 $r_{i,j}\in R,i,j\in[\tau]$和 $r\in R$ 满足$\|r_{i,j}\|_\infty\in[2^\rho]$和$\|r\|_\infty\in[2^\rho]$.输出密文:

$$c=(m+2r+2\sum r_{i,j}f_{1,i}f_{2,j})\bmod f_0.$$

密文加算法. SHE.Add.

给定公钥 pk 和密文 c_1,c_2,计算输出密文 $c=(c_1+c_2) \bmod f_0$.

密文乘算法. SHE.Mul.

给定公钥 pk 和密文 c_1,c_2,计算输出密文 $c=(c_1\times c_2) \bmod f_0$.

解密算法. SHE.Dec.

给定私钥 sk 和密文 c,计算输出 $m=[\langle c\times h^{-1}\rangle \bmod x]_2\oplus[c \bmod x]_2$.这里,$h^{-1}$ 是在 $R_{\mathbb{R}}$ 上 h 的逆.

上述算法中,$c \bmod f_0$ 是指计算 $c \bmod Rot(f_0)$,即,将 c 对应的向量 $\vec{c}$ 映射到以 $Rot(f_0)$为格基的平行四边形(parallelization)内部.

3.2 SHE正确性

因为要求 $q_0=\det(Rot(g_0))$,$p=\det(Rot(h))$为素数,故算法 SHE.KeyGen 的时间主要用于产生满足条件的 g_0,h.尽管对于很大的 k(如 $k>1024$),目前不易产生满足条件的 g_0,h,但对于实用的 k(如 $k\leqslant 1024$,见文献[3]),则易于通过随机方法产生满足条件的 g_0,h.

如果 $k=2$,则可以利用当 $p=1 \bmod 4$ 时 $p=a^2+b^2$ 的性质,在概率多项式时间内产生素数 $p_i=a_i^2+b_i^2$ 满足 $p_i=1 \bmod 4$,分别取 $h=a_1+b_1x$,$g_0=a_2+b_2x$.易于验证 $p=p_1,q_0=p_2$.

引理 3.1. 算法 SHE.Enc,SHE.Add,SHE.Mul 输出密文 c 都具有形式 $c=g\times h+2e+m$.

证明:由 SHE.Enc 易得 $c=(m+2r+2\sum r_{i,j}f_{2,i}f_{2,j}) \bmod f_0=g\times h+2e+m$.

由 SHE.Add 可知,给定密文 $c_i=g_ih+2e_i+m_i,i=1,2$,则:

$$c=(c_1+c_2) \bmod p=(g_1+g_2)h+2(e_1+e_2)+(m_1+m_2) \bmod f_0.$$

由 SHE.Mul 可知,给定密文 $c_i=g_ih+2e_i+m_i,i=1,2$,则:

$$c=(c_1\times c_2) \bmod f_0=(g_1g_2h+g_2(2e_1+m_1)+g_1(2e_2+m_2))h+2(2e_1e_2+e_1m_2+e_2m_1)+m_1m_2 \bmod f_0.$$

证毕. □

引理 3.2. 如果密文 c 中噪声多项式的极大范数小于$\lfloor 2^{\lambda}/(2k)\rfloor$,则 SHE.Dec 是正确的.

证明:给定密文 c 和私钥 sk,由引理 3.1 可知,c 具有形式 $c=g\times h+2e+m$.解密计算如下:

$$\begin{aligned}[\lfloor c\times h^{-1}+0.5I\rfloor \bmod x]_2\oplus[c \bmod x]_2&=[\lfloor (g\times h+2e+m)\times h^{-1}+0.5I\rfloor \bmod x]_2\oplus[(g\times h+2e+m) \bmod x]_2\\&=[\lfloor g+(2e+m)\times h^{-1}+0.5I\rfloor \bmod x]_2\oplus[(g+m) \bmod x]_2\\&=[g \bmod x]_2\oplus[(g+m) \bmod x]_2\\&=[g \bmod x]_2\oplus[g \bmod x]_2\oplus[m]_2\\&=m.\end{aligned}$$

因为$\|2e\|_{\infty}<\lfloor 2^{\lambda}/(2k)\rfloor$,所以$\|(2e+m)\times h^{-1}\|_{\infty}\leqslant 2^{\lambda}/(2k)\times\|h^{-1}\|_1<1/2$. □

引理 3.3. 如果密文中噪声多项式的极大范数小于 $2^{\rho'}$,则 SHE 能够正确计算由这些密文组成的电路深度为 $d\leqslant\log\dfrac{\lambda-\log 2k}{\rho'+\log k}$ 的任意加法和乘法门电路 C.

证明:假定密文 $c_i=g_ih+2e_i+m_i,i=1,2$.为了正确解密,由引理 3.2 可知,算术电路输出密文的噪声多项式的极大范数$\lfloor 2^{\lambda}/(2k)\rfloor$.加法门的噪声多项式的极大范数呈线性增长,而乘法门的极大范数呈指数增长.因此,乘法运算控制了密文算术电路深度.

设 $c=(c_1\times c_2) \bmod f_0=gh+2e+m_1m_2 \bmod f_0$,这里,$g=(g_1g_2h+g_2(2e_1+m_1)+g_1(2e_2+m_2)) \bmod f_0$,$e=(2e_1e_2+e_1m_2+e_2m_1)$,则有:

$$\|2e\|_{\infty}=\|(2e_1+m_1)\times(2e_2+m_2)\|_{\infty}\leqslant k\|2e_1+m_1\|_{\infty}\|2e_2+m_2\|_{\infty}\leqslant 2^{2\rho'+\log k}<2^{2(\rho'+\log k)}.$$

因此,d 必须满足不等式 $2^{2^d(\rho'+\log k)}<2^{\lambda}/(2k)$,即 $d\leqslant\log\dfrac{\lambda-\log(2k)}{\rho'+\log k}$. □

3.3 SHE性能

公钥 $pk=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]})$的比特大小为 $\tilde{O}(k\rho^4)$,私钥 $sk=(h)$的比特大小为 $\tilde{O}(k\rho^2)$.密文的膨胀率为 $\tilde{O}(k\rho^2)$.加密算法时间主要为 $\tilde{O}(\tau^2)$ 次乘法.为了减少加密时间在实现时一般选择 $\tilde{O}(\rho)$ 非 0 的元素 $r_{i,j}$.在这种情况下,加密时间压缩为 $\tilde{O}(\rho)$ 次乘法.解密算法时间为 1 次乘法.注意,当维数 k 越小时,为保证安全,$\eta=\tilde{O}(\rho^2)$ 中隐含的对数因子需要取得越大.

4 SHE 的安全性

我们首先在第 4.1 节给出 SHE 方案的安全性归约证明.然后,在第 4.2 节描述针对 SHE 方案已知攻击的安全分析.注意:第 4.2 节和第 5.4 节的安全分析并不是严格的安全证明,仅是分析方案能够避免目前已知攻击算法.

本节归约第 3 节 SHE 方案的安全性到求解部分近似理想格问题难度.实际上,部分近似理想格问题扩展了部分近似 GCD 问题.Dijk,Gentry,Halevi 和 Vaikuntanathan 在文献[2]的变种和优化部分已经提出设计基于部分近似 GCD 的方案,文献[4,5]使用部分近似 GCD 问题优化实现了文献[2]的全同态加密方案.

本节安全性归约证明主要自适应文献[2]中安全性归约证明方法.文献[2]的主要归约思路是通过使用攻击 SHE 的算法 A 构造求解近似 GCD 问题的算法 B,包括 4 个步骤:(1) 根据近似 GCD 问题产生 SHE 方案公钥;(2) 使用 A 构造求解 p 的近似倍数的最小比特位算法;(3) 利用步骤(2)中的算法构造求解 p 的近似倍数的 Binary GCD 算法;(4) 直接恢复近似 GCD p.

在自适应上述安全归约的过程中,困难主要来自于第 3 步,即,并不能使用 Binary GCD 算法求解近似理想格问题的近似倍数.因为文献[2]中 Binary GCD 能够使整数近似 GCD 的近似倍数不断减少(每次约减少 1/2),而在近似理想格问题上无法实现 Binary GCD 算法.然而,我们能够自适应归约证明基于部分近似理想格问题的安全性,这也是构造基于 PAILP 的 SHE 的原因之一.

本文的安全归约方法与文献[2]中方法的不同之处在于:文献[2]中的 Binary GCD 算法针对近似 GCD 中的近似倍数,而本文的 Binary GCD 算法则针对近似理想格中的噪声多项式.本文安全归约方法也适用于基于部分近似 GCD 的 SHE 方案.

4.1 安全性归约

引理 4.1. 给定 $n=\det(Rot(f))$为奇数的 $f\in R$,则在模 f 下 2 的逆元 $2^{-1}=(n+1)/2$.

证明:因为 $n=\det(Rot(f))$,故 $n=(nf^{-1})\times f$,且 nf^{-1} 是整数上多项式,所以 $2\times(n+1)/2=(n+1) \bmod f=1 \bmod f$,即:

$$2^{-1}=(n+1)/2 \bmod f.$$

证毕. □

例 4.1:设 $f=125+16x+4x^2+2x^3$,$e=2+6x-4x^2+4x^3$,则

$$\vec{e}=(2\ 6\ -4\ 4)^T,\ n=\det(Rot(f))=246202433,2^{-1}=(n+1)/2=123101217.$$

所以,

$$\vec{e}_1=2^{-1}\times\vec{e}=(246202434\ 738607302\ -492404868\ 492404868)^T,$$
$$\vec{e}_2=\langle Rot(f)^{-1}\times\vec{e}_1\rangle=(2463163\ 5656947\ -4673078\ 4316960)^T,$$
$$\vec{e}_1-Rot(f)\times\langle Rot(f)^{-1}\times\vec{e}_1\rangle=(1\ 3\ -2\ 2)^T.$$

故 $2^{-1}e \bmod f=1+3x-2x^2+2x^3$.

所以,当多项式 e 的系数都是偶数时,在模 f 下乘以 2^{-1} 等价于 e 除以 2.

定理 4.1. 任意具有优势ε的 SHE 攻击算法 A 都能够转换为求解分布为 $D_{R,\rho,\lambda,\eta}(h)$的近似理想格问题算法 B,成功概率为$\varepsilon/2$.B 的运行时间是 t_A,ρ和 $1/\varepsilon$的多项式,t_A 为 A 的运行时间.

证明:我们从算法 A 构造求解近似理想格问题算法 B.算法 B 从分布 $D_{R,\rho,\lambda,\eta}(h)$中获取需要的独立抽样数,目的是求解 h.

在证明中,我们假定每次调用算法 A,其返回形如 $c=(g\times h+r) \bmod p$ 中噪声 r 的常系数奇偶比特,无论 r 的其他系数的奇偶性.实际上,这个假设与 SHE 中解密算法的返回结果一致.

评论 4.1. 如果密文 $c=(g\times h+r)$中 r 为其他形式时,算法 A 返回随机结果(或特殊符号⊥),则本节归约证明仅对 $k=O(\log\rho)$成立.因为在这种情况下,只有 $2^k=\rho^{O(1)}$个可能值,敌手能够猜测 $r \bmod 2$ 的值,每次猜测后调用 A.如果猜测正确,则在多项式次调用中返回正确结果的概率具有绝对优势;否则,返回结果中 0 和 1 的概率基本相等(或特殊符号⊥).

步骤 1. 产生公钥.

B 从分布 $D_{R,\rho,\lambda,\eta}(h)$中抽取 2τ个抽样 $f_{t,j}, t\in[2], j\in[\tau]$和 $f_0=g_0h$,输出公钥 $pk=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]})$.

步骤 2. 计算噪声多项式系数奇偶性子程序.

给定 h 的一列近似倍数,B 使用 A 获得关于 h 的这些近似倍数中噪声多项式系数的奇偶性,即,调用子程序 *Learn-Noise-Coeff-Parity*.

Learn-Noise-Coeff-Parity(v,pk)

给定 $v=(g\times h+e) \bmod n$ 满足$\|e\|_\infty\leqslant 2^\rho$,计算 $e \bmod 2$.

(1) For t=0 to k–1 do //依次求解噪声多项式的第 t 项系数的奇偶性

(2) $w_t=(x^{k-t}v) \bmod (x^k+1)$ //将噪声多项式的第 t 项系数移到常数项位置

(3) For s=1 to $poly(\rho)/\varepsilon$ do

(4) 随机选择比特 $m_i\in\{0,1\}, r_{i,j}^s\in R, i,j\in[\tau], r_0^s\in R$,满足 $\|r_{i,j}^s\|_\infty=2^\rho, \|r_0^s\|_\infty=2^\rho$.

(5) 计算 $c_i=(w_t+m_i+2r_0^s+2\sum_{i,j\in[\tau]}r_{i,j}^s f_{1,i}f_{2,j}) \bmod f_0$,

(6) 调用 A 得到 $b_i=A(pk,c_i)\oplus m_i$,

(7) 设置 u_t 等于 b_i 中出现次数最多的值.

(8) 输出 $u=\sum_{t=0}^k u_t x^t$ 作为 $e \bmod 2$.

步骤 3. 去除 h 的近似倍数中噪声多项式.

当由步骤 2 获得噪声多项式系数的奇偶性神谕(oracle)时,去除 h 的近似倍数中噪声多项式就比较容易.

给定 $v=(g\times h+e) \bmod f_0$,输出 $v'=(2^{\rho+\log\rho}g\times h) \bmod f_0=g'\times h$.

(1) For i=0 to $\rho+\log\rho$ do

(2) u=*Learn-Noise-Coeff-Parity*(v,pk)

(3) $v=v-u$

(4) $v=(2^{-1}\times v) \bmod f_0$ //注意:当 $n=\det(Rot(f_0))$为奇数时,模 f_0 下,$2^{-1}=(n+1)/2$

(5) u=*Learn-Noise-Coeff-Parity*(v,pk)

(6) $v'=v+u$ //为了去除噪声多项式中系数为‘–1’的项.如果 $v'=0$,步骤 3 重新开始.

评论 4.2. 因为 $v=(g\times h+e) \bmod f_0$ 中噪声 e 满足条件$\|e\|_\infty\leqslant 2^\rho$,但 e 的系数有正有负:如果某个系数 e_i 为正,则 $e_i–u_i$ 为正偶数,且 $2^{-1}(e_i–u_i) \bmod f_0$ 将 $e_i–u_i$ 减少 1/2,程序一直循环,直到该系数为 0;如果某个系数 e_i 为负,则 $e_i–u_i$ 为负偶数,且 $2^{-1}(e_i–u_i) \bmod f_0$ 将 $e_i–u_i$ 在绝对值上减少 1/2,但这个系数不可能减至 0,因为当 $e_i=–1$ 时,$u_i=1$,则 $e_i–u_i=–2$ 和 $2^{-1}(e_i–u_i) \bmod f_0=–1$,即,当 e_i 减少到–1 后,每次循环后仍然为–1.故步骤 3 中第(6)小步使用加 u 以去除噪声中所有系数为‘–1’的项.

步骤 4. 计算 GCD.

使用 GCD 算法求解 $p=\gcd(\det(Rot(v')),\det(Rot(f_0)))$和 $\gcd(f_0,x^k+1)=(x-\alpha) \bmod p$.

步骤 5. 求解 h.

因为两个元素(p,α)生成的理想格与 h 生成的理想格相同,故由(p,α)构造格 L,然后调用 LLL 格归约算法得到最短向量作为多项式 h 的系数向量:

$$L=\begin{pmatrix} p & 0 & 0 & \cdots & 0 \\ -\alpha & 1 & 0 & \cdots & 0 \\ -\alpha^2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ -\alpha^{k-1} & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

评论 4.3. 因为本文中维数 k 一般较小(通常为 $O(\log\rho)$或常数),故步骤 5 中通过 LLL 格归约算法一般能够直接得到 h.另一方面,若 k 较大(如为 $\rho^{O(1)}$),则可通过 LLL 格归约算法尝试求解 h 或 h 的倍数 $h'=s\times h$.若 $\|h'\|_\infty/\|h\|_\infty$ 较小且 $\det(Rot(h'))=p$,则可直接使用 h'的逆$(h')^{-1}$ 去解密密文;否则,构造格 $L'=\begin{pmatrix} Rot(v') \\ Rot(h') \end{pmatrix}=\begin{pmatrix} Rot(h)Rot(g') \\ Rot(h)Rot(s) \end{pmatrix}$,并调用 LLL 格归约算法求解 $Rot(h)$或 $Rot(h)U$,U 为单模矩阵,这里假定 g',s 互素.当然,在 k 较大时,使用 LLL 格归约算法求解 h 不一定成功.

上述步骤 4 和步骤 5 也可以由步骤 4-5′替换.

步骤 4-5′. 求解 h.

构造格 $L=\begin{pmatrix} Rot(v') \\ Rot(f_0) \end{pmatrix}=\begin{pmatrix} Rot(h)Rot(g') \\ Rot(h)Rot(g_0) \end{pmatrix}$,并调用 LLL 格归约算法求解 $Rot(h)$或 $Rot(h)U$,U 为单模矩阵.由于本文中 k 较小,故 LLL 算法一般能够得到 h.

B 的成功概率分析:

首先,由步骤 1 可知,B 产生的公钥分布与 SHE 方案中的公钥分布相同;

其次,步骤 2 中的子程序 *Learn-Noise-Coeff-Parity*(v,pk)是 $e \bmod 2$ 的可靠神谕.由引理 4.1 可知:除了公钥的可忽略部分外,对于所有公钥,程序 *Learn-Noise-Coeff-Parity* 第(5)行产生的密文 c_i 的分布是统计上接近于 SHE 方案中比特$(e \bmod x \bmod 2)\oplus m_i$ 的密文分布.使用计数方法易于证明:如果在公钥 pk 下敌手 A 具有猜测加密比特优势ε,那么对于私钥集中至少$\varepsilon/2$ 的私钥 h,敌手 A 具有优势至少$\varepsilon/2$;而且在与这部分私钥 h 对应的公钥集中,至少$\varepsilon/4$ 部分的公钥,敌手 A 具有优势至少$\varepsilon/4$.因此,*Learn-Noise-Coeff-Parity* 的第(6)行敌手 A 具有优势至少$\varepsilon/4-negl$,并且 *Learn-Noise-Coeff-Parity*(v,pk)的多数选举策略将以具压倒性优势的概率返回正确答案.所以,步骤 3 将去除 h 的近似倍数中噪声多项式,获得 h 的倍数;

最后,算法 B 的步骤 4 和步骤 5 将恢复 h.因此,对于 A 具有优势至少$\varepsilon/2$ 的 h,公钥集中至少$\varepsilon/4-negl$ 部分,A 具有优势至少$\varepsilon/4$.对于 A 具有优势至少$\varepsilon/2$ 的 h,使用新随机公钥 $4/\varepsilon\times\omega(\log\rho)$次重复调用算法 B,以压倒性优势的概率恢复 h.因此,算法 B 的成功概率至少为$\varepsilon/2$.

证毕. □

引理 4.2. 给定参数(ρ,λ,η,τ),设 $sk=h$,$pk=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]})$是 SHE.KeyGen 随机产生的.对每个满足$\|e\|_\infty\leqslant 2^\rho$ 的 $f^*=(g\times h+e) \bmod f_0$,设分布:

$$D_{pk}(f^*)=\{c^*=(f^*+2r_0+2\sum_{j\in[\tau]}r_{i,j}f_{1,i}f_{2,j}) \bmod f_0 \mid r_{i,j}\in R,\|r_{i,j}\|_\infty\leqslant 2^\rho,i,j\in[\tau]\}.$$

那么在 sk,pk 选择上以压倒性优势的概率,每个分布 $D_{pk}(f^*)$在统计上接近于分布:

$$SHE.Enc(pk,m=e \bmod x \bmod 2).$$

证明:引理 4.2 的证明与文献[4]中引理 D.1 的证明相同,仅由引理 4.3 替换文献[4]中的引理 4.2.

设 $g\in R$,$R_g=\{y \bmod g \mid y\in R\}$是以 $Rot(g)$为格基的平行四边形内部元素,$R_\rho=\{y\in R\mid\|y\|_\infty\in[2^\rho]\}$.设 H 是从 $R_\rho^{\tau\times\tau}$ 到 R_g 的 Hash 函数簇,成员 $w\in H$ 关联到 R_g 中的元素 $g_{t,j}$,$t\in[2]$,$j\in[\tau]$.对于 $r\in R_\rho^{\tau\times\tau}$,定义:

$$w(r)=\sum_{i,j\in[\tau]}r_{i,j}g_{1,i}g_{2,j} \bmod g.$$

证毕. □

引理 4.3. 设 $g\in R$,$q=\det(Rot(g))$为素数,Hash 函数簇 H 是ε两两独立的.这里,$\varepsilon\approx\frac{1}{q}+\frac{\tau^2}{2^{k\rho\tau^2-2(k\rho+1)\tau}}$.

证明:由引理 2.2,引理 4.3 的证明与文献[4]中引理 4.2 的证明相同,除了文献[4]在 Z_q 上计算,而这里在 R_g 上计算.符号对应关系:在公式ε中,这里的 q 指的是 R_g 中的元素个数,文献[4]中指的是 Z_q 中的元素个数;这里的 $k\rho$ 对应于文献[4]中的α. □

评论 4.4. 上述针对变种 PAILP 的安全归约证明可以直接应用于 PAILP1 和 PAILP2 问题,仅在算法中将 $\frac{1}{2}=(\det(Rot(f_0))+1)/2$ 替换为 $\frac{1}{2}=(f_0+1)/2$, 其他证明相同.

4.2 已知攻击

由于 PAILP 扩展自 PAGCD 问题,故针对 PAGCD 的攻击方法[2,58]都需要考虑其攻击 PAILP 的可行性.本节主要研究分析对 SHE 方案的已知攻击和如何设置方案参数以避免这些攻击.

4.2.1 因式分解攻击

因为在 SHE 的 pk 中包含 $f_0=g_0h$,敌手能够计算 $f_0'=\det(Rot(f_0))=q_0p$.f_0' 中最小素因子约为 $k\eta$比特,使用 Lenstra 的椭圆曲线因子分解算法[52]需要时间约为 $\exp(O(\sqrt{k\eta}))\approx\exp(O(\sqrt{k}\rho))$, 仍为$\rho$的指数时间.

4.2.2 连分数攻击[58]

如果直接计算 $f_i'=\det(Rot(f_i))=q_ip+e_i$, 易于验证:通常情况下,$e_i\approx(q_ip)^{(k-1)/k}>>p$.不失一般性,假定 $e_i \bmod p$ 均匀分布,则 $e_i \bmod p << \sqrt{p}$ 的概率几乎为 0.故,通过两个元素 (f_0', f_i') 的连分数计算得到 p 的概率几乎为 0.

下面首先通过例子扩展整数上连分数概念到多项式环 R 上的连分数.

例 4.2:设 $y_0=125+16x+4x^2+2x^3$,$y_1=2+6x-4x^2+4x^3$.在 R 上使用欧几里德算法计算:

$$y_0=q_0(x)y_1+y_2=(13-2x-3x^2-18x^3)y_1+(-5+2x+2x^2-4x^3);$$
$$y_1=q_1(x)y_2+y_3=(-1-2x-2x^2-2x^3)y_2+(-3+2x+0x^2-2x^3);$$
$$y_2=q_2(x)y_3=(3+0x-2x^2-2x^3)y_3.$$

在上述算法中,$q_i(x)=\langle y_iy_{i+1}^{-1}\rangle$,简记为 q_i.易于验证 $\det(Rot(y_3))=1$.

定义 y_0/y_1 连分数为 $\frac{y_0}{y_1}=q_0+\cfrac{1}{q_1+\cfrac{1}{q_2}}$.

为了使对任意一对(y_0,y_1),$y_1\neq0$,上述扩展连分数定义有意义,需证明:如果 $y_{i+1}\neq0$,则在 $R_{\mathbb{R}}$ 中存在 y_{i+1}^{-1}.因为如果 $Rot(y_{i+1})$可逆,即 $\det(Rot(y_{i+1}))\neq0$,则 y_{i+1}^{-1} 存在.又因为当 k 为 2 的幂时,$f=x^k+1$ 为 Z 上不可约多项式,故 $\gcd(f,y_{i+1})=1$.由文献[54]中的推论 6.15 可知,f,y_{i+1} 的结式(resultant)S 的行列式 $\det(S)\neq0$.通过简单矩阵变换,可得 $\det(Rot(y_{i+1}))=\det(S)$.所以,当 $y_{i+1}\neq0$ 时,$\det(Rot(y_{i+1}))\neq0$,即,y_{i+1} 在 $R_{\mathbb{R}}$ 中可逆.

现在考虑公钥元素(f_0,f_i)形成的连分数.因为 $\left\|\frac{f_i}{f_0}-\frac{g_i}{g_0}\right\|_\infty=\left\|\frac{r_i}{g_0h}\right\|_\infty$, 所以期望 $\frac{g_i}{g_0}$ 出现在多项式环的连分数中.

实际上,仅有 $\frac{f_i}{f_0}$ 接近 $\frac{g_i}{g_0}$ 并不意味着连分数的输出需要近似值 $\frac{g_i}{g_0}$. 因此,当$||r_i||_\infty$较大时,连分数法攻击不会成功.

4.2.3 格归约攻击

因为近似 GCD 问题中求解的 p 具有原子性,而在近似理想格问题中求解的 h 具有更多组合性,导致文献[2]中的方案易于受到格归约的攻击,而基于近似理想格的方案更难以通过格归约进行攻击.

(1) 通过公钥求解私钥 h 的格归约攻击

如果由$(f_0,f_{1,1})$构造格 $L_1=\begin{pmatrix}Rot(f_{1,1}) & I_k\\ Rot(f_0) & 0\end{pmatrix}$, 则 $\det(L_1)=q_0p$.由 Minkowsky 定理[59]可知,L_1 的最短长度向量 v 满足条件$||v||_\infty\leqslant(q_0p)^{1/(2k)}\approx2^{(\lambda+\eta)/2}$.另一方面,$L_1$ 包含向量 $v_1=(\overline{g_0r_1}\quad\overline{g_0})$ 满足条件$||v_1||_\infty\approx2^{\eta+\rho}$.因此,当$\lambda<<\eta$时,$L_1$ 中包含有指数多个向量 v,其长度小于$||v_1||_\infty$,即,格归约并不能产生关于向量 v_1 的有用信息.

如果由$(f_{1,i},f_{1,j})$构造格 $L_2=\begin{pmatrix} Rot(f_{1,i}) & I_k \\ Rot(f_{1,j}) & 0 \end{pmatrix}$, 则 $\det(L_2)=\det(Rot(f_{1,j}))\approx 2^{k(\lambda+\eta)}$,与格 L_1 的情况相似.

易于验证:如果由 $t(t>2)$个公钥元素构造格,其包含的最小向量长度与上述由 2 个元素构造格的情况类似.

(2) 通过公钥和密文求解明文的格归约攻击

给定公钥 $pk=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]})$和密文 c,构造格 L_3:

$$L_3=\begin{pmatrix} \vec{c} & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 \\ Rot(2f_{1,1}f_{2,1} \bmod f_0) & 0 & \cdots & 0 & 0 & \cdots & I_k & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \iddots & 0 & \vdots \\ Rot(2f_{1,1}f_{2,\tau} \bmod f_0) & 0 & \cdots & 0 & I_k & \cdots & 0 & 0 \\ Rot(2f_{1,2}f_{2,1} \bmod f_0) & 0 & \cdots & I_k & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \iddots & 0 & \vdots & \vdots & \vdots & \vdots \\ Rot(2f_{1,\tau}f_{2,\tau} \bmod f_0) & I_k & \cdots & 0 & 0 & \cdots & 0 & 0 \\ Rot(f_0) & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

因为 $\det(L_3)=q_0p$,由 Minkowsky 定理可知,L_3 的最短长度向量 v 满足条件:

$$\|v\|_\infty \leqslant (q_0p)^{1/((\tau^2+1)k+1)} \approx 2^{(\lambda+\eta)/(\tau^2+1)}=2^{O(1)}.$$

而由密文 $c=(m+2r+2\sum r_{i,j}f_{1,i}f_{2,j}) \bmod f_0$ 可知 $v_c=(\overline{2r+m},\bar{r}_{1,1},...,\bar{r}_{1,\tau},\bar{r}_{\tau,1},...,\bar{r}_{\tau,\tau},1)\in L_3$, 满足$||v_c||_\infty=||r_{i,j}||_\infty=O(2^\rho)$.因此,如果格归约算法能够求得较短长度的向量 v_1,则 v_1 与向量 v_c 之间关系能够获得明文信息.然而,为了获得 2^τ近似因子的向量,现有格归约算法需要时间约为 $O(2^\tau)$,显然,在计算上是不可行的.

4.2.4 枚举攻击

最简单的枚举攻击猜测公钥中噪声多项式.因为 $f_{t,j}=hg_{t,j}+e_{t,j}$,则可以猜测 $e_{t,j}$.这种蛮力攻击的时间复杂性为 $O(2^{k\rho})$.

最近,Chen 和 Nguyen[24]给出了求解 PAGCD 的算法.如果 $x_0=pq_0$ 和 $x_i=pq_i+r_i$,这里,$0\leqslant r_i<2^\rho$,$1\leqslant i\leqslant\tau$,则 $p=\gcd\left(x_0,\prod_{j=0}^{2^{\rho'}-1}(x_1-j) \bmod x_0\right)$. 因为 $f_t(x)=\prod_{i=0}^{t-1}(x_1-(x+i)) \bmod x_0$, 所以 $\prod_{j=0}^{2^{\rho'}-1}(x_1-j)=\prod_{j=0}^{2^{\rho'+(\rho \bmod 2)}-1}f_{2^{\rho'}}(2^{\rho'}j) \bmod x_0$, 记 $\rho'=\lfloor\rho/2\rfloor$.因此,求解 PAGCD 需要使用一次 GCD、$2^{\rho'+(\rho \bmod 2)}-1$ 次模乘法和计算 $2^{\rho'}$度的多项式的 $2^{\rho'+(\rho \bmod 2)}$点值,这个计算费用至多为 $\tilde{O}(2^{\rho'})=\tilde{O}(2^{\rho/2})$.

然而在自适应 PAGCD 算法求解近似理想格问题时,并不能获得同样的性能.

尽管 $h=\gcd(\prod_{\|r\|_\infty\leqslant 2^\rho}(f_{1,1}-r) \bmod f_0, f_0)$, 但在模 f_0 下,定义多项式系数为关于变量 $r_0,r_1,\ldots,r_{k/2-1}$ 的 $d=(2^{\rho+1}+1)^{k/2}$ 度的多元多项式:

$$b_d(r_0,r_1,...,r_{k/2-1})=\prod_{r_{k-1}=-2^\rho}^{2^\rho}...\prod_{r_{k/2}=-2^\rho}^{2^\rho}(f_{1,1}-r_{k-1}x^{k-1}-...-r_{k/2}x^{k/2}-r_{k/2-1}x^{k/2-1}-r_0) \bmod f_0.$$

易于看出,多元多项式 $b_{2^{k\rho/2}}(r_0,r_1,...,r_{k/2-1})$ 的展开式中项数至少为 $2^{k\rho}$.故,通过这种方法并不能减少计算 $\prod_{\|r\|_\infty\leqslant 2^\rho}(f_{1,1}-r) \bmod f_0$ 的时间.

易于看出:当 $1<i<k$ 时,蛮力猜测 r 的 $k-i$ 个系数会产生度为 $d=(2^{\rho+1}+1)^{(k-i)}$的 i 元多项式需要计算,其计算时间复杂性并不比穷举算法时间 $O(2^{k\rho})$更少.

然而,当蛮力猜测 r 的 $k-1$ 个系数,产生度为 $d=(2^{\rho+1}+1)^{(k-1)}$的单变量多项式:

$$b_d(r_0)=\prod_{r_{k-1}=-2^\rho}^{2^\rho}...\prod_{r_1=-2^\rho}^{2^\rho}(f_{1,1}-r_{k-1}x^{k-1}-...-r_{k/2}x^{k/2}-r_1x^1-r_0) \bmod f_0$$

时,则这种蛮力攻击计算时间复杂性为 $O(2^{(k-1)\rho})$,比完全穷举攻击稍好一点,但代价是需要空间为 $O(2^{(k-1)\rho})$.

5 全同态加密方案(FHE)

因为 SHE 的解密算法所需要的布尔电路深度比有点同态加密方案能够处理的电路深度更深,因此我们使用 Gentry 压扁解密算法从 SHE 构造全同态加密方案.Gentry 压扁解密算法的目的是使 SHE 的解密算法能够表示成由 SHE 同态运算支持的低度多项式,并且最终应用引导变换实现全同态加密方案.关键是构造 SHE 方案,使其能同态计算的多项式度数超过解密多项式度数的两倍.这种 SHE 方案称为可引导的,并且能够被转化为全同态加密方案.

5.1 压扁解密电路

因为解密时仅需计算$\langle c\times h^{-1}\rangle$,所以为正确解密 h^{-1} 中的每个系数,保留精度至$\delta=\lambda+\eta+O(\log\rho)$比特位.

使用 Gentry 引导技术,在公钥中添加一个稀疏子集 $S=\{y_i\in R_{\mathbb{R},\delta}:i\in[\Theta]\}$, 私钥为对应于集合 S 的比特特征向量$\omega=(\omega_1,\ldots,\omega_\Theta)$,满足$\omega$的 Hamming 权重$\theta<<\Theta$,并且$\|(\sum_{i=1}^{\Theta}\omega_i y_i \bmod 2)-h^{-1}\|_\infty\leqslant 2^{-\delta-1}$.

给定一个密文 $c\in R$,解密算法可以修改为

$$\begin{aligned}Dec(c,S,\omega)&=[\langle c\times h^{-1}\rangle \bmod x]_2\oplus[c \bmod x]_2\\&=[\langle c\times\sum_{i=1}^{\Theta}\omega_i y_i\rangle \bmod x]_2\oplus[c \bmod x]_2\\&=[\langle\sum_{i=1}^{\Theta}\omega_i(c\times y_i)\rangle \bmod x]_2\oplus[c \bmod x]_2\\&=[\langle\sum_{i=1}^{\Theta}\omega_i u_i\rangle \bmod x]_2\oplus[c \bmod x]_2\\&=[\langle\sum_{i=1}^{\Theta}\omega_i u_i \bmod x\rangle]_2\oplus[c \bmod x]_2.\end{aligned}$$

这里,$u_i=c\times y_i$.

为提高效率,与文献[7,17]一样,我们将Θ个元素分成θ个小块,每块中包含Θ/θ个元素,满足在比特特征向量$\omega=(\omega_1,\ldots,\omega_\Theta)$中对应于每个小块中有且仅有一个$\omega_i=1$,其余都为 0.

5.2 FHE构造

在密文同态计算过程中,密文中的噪声一直在增长.当密文中的噪声达到某一阈值时,就不能再实行密文同态计算,否则,新产生的密文就不是明文比特的正确密文.在这种情况下,需要使用密文刷新算法(recrypt). Recrypt 能够将高噪声密文 c 转化为低噪声密文 c_{new},并且密文 c_{new} 与 c 加密的明文比特相同.为了方案可引导,在 FHE 方案公钥中,需要提供私钥比特向量ω的密文.因此,与现有全同态加密方案一样,我们也假定 FHE 是 KDM[60]安全的.

FHE 密钥生成算法. FHE.KeyGen.

(1) 使用 SHE.KeyGen 产生 $pk=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]})$和 $sk=(h)$;

(2) 计算产生 $h^{-1}\in R_{\mathbb{R},\delta}$;

(3) 随机选择θ个子集 $S_i=\{y_{i,j}\in R_{\mathbb{R},\delta},j\in[\Theta/\theta]\}$, 子集 S_i 的比特特征向量$\omega_i=(\omega_{i,1},\ldots,\omega_{i,\Theta/\theta})$满足有且仅有一个$\omega_{i,j}=1$,其余都为 0,并且 $\sum_{i=1}^{\theta}\sum_{j=1}^{\Theta/\theta}\omega_{i,j}y_{i,j}=h^{-1} \bmod 2$;

(4) 加密$\omega_{i,j}$作为 $\bar{\omega}_{i,j}=a_{i,j}\times h+2e_{i,j}+\omega_{i,j}$ 满足$\|a_{i,j}\|_\infty\in[2^\eta],\|2e_{i,j}\|_\infty\in[2^\rho]$;

(5) 输出公钥 $PK=(k,f_0,\{f_{t,j}\}_{t\in[2],j\in[\tau]},\{\bar{\omega}_{i,j},y_{i,j}\}_{i\in[\theta],j\in[\Theta/\theta]})$ 和私钥 $SK=(\omega_1,\ldots,\omega_\theta)$.

加密算法. FHE.Enc.

与 SHE.Enc 相同.

密文加算法. FHE.Add.

与 SHE.Add 相同.

密文乘算法. FHE.Mul.

与 SHE.Mul 相同.

解密算法. FHE.Dec.

给定 SK 和密文 c,计算输出消息比特 m.

(1) 计算 $u_{i,j}=[\langle(2^{\sigma}c\times y_{i,j}) \bmod x\rangle/2^{\sigma}]_2$,这里,$u_{i,j}$ 保留 $\sigma=\lceil\log(\theta+1)\rceil$ 位比特小数,且 $u_{i,j}$ 最接近数 $(c\times y_{i,j}) \bmod x$;

(2) 计算 $\overline{u}_{i,j}=u_{i,j}\times\omega_{i,j}$;

(3) 计算每个小块和 $\overline{u}_i=\sum_{j=1}^{\Theta/\theta}\overline{u}_{i,j}$;

(4) 计算和 $\overline{u}=[\sum_{i=1}^{\theta}\overline{u}_i]_2$;

(5) 输出明文 $m=[c \bmod x]_2\oplus\overline{u}_0\oplus\overline{u}_{-1}$,这里,数 $\overline{u}=\overline{u}_0.\overline{u}_{-1}.....\overline{u}_{-\sigma}$.

密文刷新算法. *FHE.Recrypt*(*PK*,*c*).

(1) 计算 $u_{i,j}=[\langle(2^{\sigma}c\times y_{i,j}) \bmod x\rangle/2^{\sigma}]_2$;

(2) 转换 $u_{i,j}$ 为密文形式的数 $\vec{u}_{i,j}=u_{i,j}\times\vec{\omega}_{i,j}$;

(3) 计算小块密文形式的和数 $\vec{u}_i=\sum_{j=1}^{\Theta/\theta}\vec{u}_{i,j}$;

(4) 使用对称多项式方法[2,7]计算密文形式的和数 $\vec{u}=[\sum_{i=1}^{\theta}\vec{u}_i]_2$;

(5) 输出新密文 $c_{new}=[c \bmod x]_2\oplus\vec{u}_0\oplus\vec{u}_{-1}$, 这里设密文形式的数 $\vec{u}=\vec{u}_0.\vec{u}_{-1}.....\vec{u}_{-\sigma}$.

评论 5.1. 在解密或密文刷新时,计算 $u_{i,j}$ 时保留到最接近 σ 位比特小数,因此,其每个误差至多为 $\frac{1}{2^{\sigma+1}}$. 而在这些 $u_{i,j}$ 中,仅有 θ 个 $\omega_{i,j}$ 为 1,故,舍入产生的误差至多为 $\frac{\theta}{2^{\sigma+1}}$.

为了保证解密或密文刷新的正确,即 $[\langle\sum_{i=1}^{\theta}\sum_{j=1}^{\Theta/\theta}\omega_{i,j}u_{i,j}\rangle]_2=[\langle c\times h^{-1}\rangle \bmod x]_2$, 故要求参数满足条件:

$$|\langle c\times h^{-1}\rangle \bmod x-(c\times h^{-1}) \bmod x|<\frac{1}{2\theta}.$$

5.3 FHE正确性

算法 FHE.KeyGen,FHE.Enc,FHE.Add,FHE.Mul 的正确性易于由 SHE.KeyGen,SHE.Enc,SHE.Add,SHE.Mul 的正确性得到.

解密算法 FHE.Dec 的正确性可由第 5.1 节的压扁解密电路得到,FHE.Recrypt 的正确性由解密算法 FHE.Dec 得到.这两种算法之间的差异在于:FHE.Dec 直接使用私钥 *SK*,而 FHE.Recrypt 使用 *SK* 的比特密文形式.结果是,FHE.Dec 得到密文中的明文比特,而 FHE.Recrypt 得到密文中明文比特的新密文.

定理 5.1. 设 $\theta(\rho+\log k+\log\theta+1)\leqslant\lambda$,则新密文 $b_i=FHE.Recrypt(PK,c_i)$,i=1,2 与密文 c_i 具有相同的明文比特,并且 b_1,b_2 能够进行一次同态密文乘法运算.

证明:比较两种算法 FHE.Recrypt,FHE.Dec 可以看出,FHE.Recrypt 是 FHE.Dec 在密文形式下解密.故定理前半部分仅需证明 FHE.Recrypt 能够在密文形式下正确实现 FHE.Dec.

步骤(1)

该步与 FHE.Dec 的步骤(1)相同.

步骤(2)

该步将 $\omega_{i,j}$ 的密文 $\vec{\omega}_{i,j}$ 与 $u_{i,j}$ 中每个比特相乘,从而得到 $\vec{u}_{i,j}$.如果 $\omega_{i,j}$ 为 0,则 $\vec{u}_{i,j}$ 是数 0 的密文形式;如果 $\omega_{i,j}$ 为 1,则 $\vec{u}_{i,j}$ 就是 $u_{i,j}$ 的密文形式.因此,步骤(2)的结果就是 FHE.Dec 的步骤(2)中结果的密文形式.

步骤(3)

该步计算小块密文形式的和数,这里,每个小块中有且仅有一个 $\omega_{i,j}$ 为 1,故在密文形式的数 $\vec{u}_{i,j}$ 中有且仅有一个非零数,即,密文数 $\vec{u}_i=\sum_{j=1}^{\Theta/\theta}\vec{u}_{i,j}$ 等于某个 $\vec{u}_{i,j}$. 所以,密文数可直接相加,无需考虑进位问题.而 FHE.Dec(3)和 $\overline{u}_i=\sum_{j=1}^{\Theta/\theta}\overline{u}_{i,j}$ 中有且仅有一个非零数,与 $\vec{u}_i=\sum_{j=1}^{\Theta/\theta}\vec{u}_{i,j}$ 中非零密文数相对应.所以,步骤(3)的结果也是 FHE.Dec(3)中结果的密文形式.

步骤(4)

计算密文形式的和数 $\vec{u}=[\sum_{i=1}^{\theta}\vec{u}_i]_2$.我们采用文献[7]中的一般加法(grade-school addition,如图 1 所示)求θ个密文数和:首先,将θ个密文数安排成θ行、σ+1 列,这些列从左到右依次对应于第 0,−1,…,−σ位;然后,使用第 2.3 节中计算对称多项式的动态规划算法依次计算从右(第−σ位)到左(第 0 位)的比特和,第−j 列进位到第−j+t 列的比特值是第−j 列比特值的 2^t 度的初等对称多项式.因为$\sigma=\lceil\log(\theta+1)\rceil$,故在按列序(即−$\sigma$列、−$\sigma$+1 列,直到−1 列)求和时,需要计算该列向左最大进位数 t 分别为σ−1,σ−2,…,1.因此,在计算相应列时,这些列上的比特数分别为$\theta,\theta+1,\ldots,\theta+\sigma-1$.

因为如果第−j 列有 m 个比特,则计算这些比特上所有直到 2^t 度的初等对称多项式需要使用至多 $m2^t$ 次乘法,因此,这一步计算需要的乘法总数至多为 $\theta 2^{\sigma-1}+\sum_{t=1}^{\sigma-1}(\theta+t)2^{\sigma-t}=O(\theta^2)$.

为简单起见,我们取θ=15,$\sigma=\lceil\log(\theta+1)\rceil$=4,图 1 中,方块中数字为计算相应比特密文需要的多项式度数.

尽管存在需要更少乘法次数的其他加法,然而上述一般加法使用更小度数的多项式.因此,下面实现全同态方案时,我们仍然使用这种一般加法.

步骤(5)

计算最接近密文形式的数 $\vec{u}=\vec{u}_0.\vec{u}_{-1}.....\vec{u}_{-\sigma}$ 模 2 整数的密文 $\vec{u}_0\oplus\vec{u}_{-1}$,即:等价于密文形式计算$\langle c\times h^{-1}\rangle \bmod x$,相当于 FHE.Dec 步骤(5)中的明文计算 $\vec{u}_0\oplus\vec{u}_{-1}$.最后,将比特$[c \bmod x]_2$直接加到密文 $\vec{u}_0\oplus\vec{u}_{-1}$ 的常系数项上.

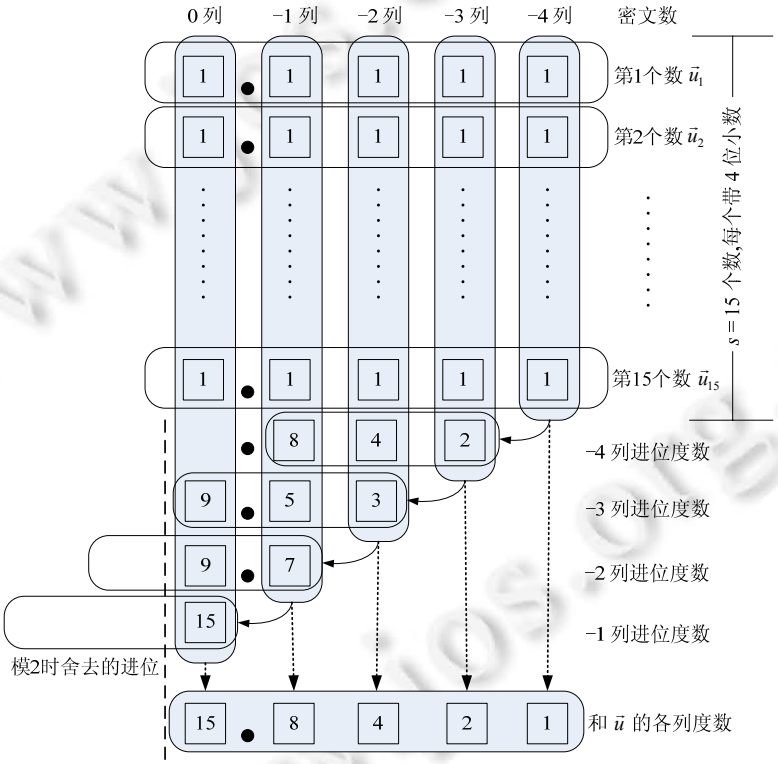

Fig.1 Grade-School addition for encrypted data

图 1 密文数一般加法示意图

现在分析新密文中噪声多项式的大小,进而确定新密文能够再进行至少一次同态密文计算.

根据文献[7]中的对称多项式计算方法可知(如图 1 所示),计算θ个密文数和至多需要计算θ度对称多项式.

易于验证:表示密文加法的度为θ的单项式数目至多为$\binom{\theta}{\lceil\theta/2\rceil}\times\binom{\theta}{\lceil\theta/4\rceil}\times\ldots\times\binom{\theta}{1}$,它小于$2^{\theta\log\theta}$.而在度为$\theta$的单项式产生的密文中,噪声多项式的极大范数至多为$k^{\theta-1}2^{\theta\rho}=2^{\theta\rho+(\theta-1)\log k}$.因此,在密文刷新算法输出密文的噪声多项式极大范数至多为$2^{\theta\rho+(\theta-1)\log k+\theta\log\theta}$.为实现全同态加密,还需要密文刷新后新产生密文必须能够进行一次同态乘法运算,故,密文噪声极大范数至多为$2^{2\theta\rho+(2\theta-1)\log k+2\theta\log\theta}$.

因此,为了正确解密,由引理 3.2 和条件$|\langle c\times h^{-1}\rangle \bmod x-(c\times h^{-1}) \bmod x|<\frac{1}{2\theta}$,需要满足$2^{2\theta\rho+(2\theta-1)\log k+2\theta\log\theta}\leqslant 2^{\lambda}/(2k\theta)$,即,$2\theta(\rho+\log k+\log\theta+1)\leqslant\lambda$. □

5.4 FHE的安全性

5.4.1 稀疏子集和的格攻击

在上述 FHE 方案中,除了 KDM 假设和因式分解难度假设,还引入了稀疏子集和问题 SSSP 假设.

在 SSSP 问题中,攻击者需求解方程$\sum_{j=1}^{\Theta}\omega_j y_j=h^{-1} \bmod 2$. 这里假定攻击者知道 h^{-1} 和集合 $y_j\in R_{\mathbb{R},\delta}$, $j\in[\Theta]$,并且私钥$\omega=(\omega_1,\ldots,\omega_\Theta)$具有小 Hamming 权重$\theta$.对于 SSSP 问题,易于构造行向量的格 L_i:

$$L_i=\begin{pmatrix} 2^{\delta+1} & 0 & 0 & \cdots & 0\\ -(2^{\delta}h^{-1})_i & 1 & 0 & \cdots & 0\\ (2^{\delta}y_1)_i & 0 & 1 & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ (2^{\delta}y_\Theta)_i & 0 & 0 & \cdots & 1 \end{pmatrix},i\in[k].$$

这里,$(h^{-1})_i,(y_j)_i,j\in[\Theta]$分别为 h^{-1},y_j 的第 i 项.

在格 L_i 中,$v=(0,1,\omega_1,\ldots,\omega_\Theta)$一般为最短长度向量,其范数为$\sqrt{\theta+1}$. 因为格 L_i 的行列式为 $2^{\delta+1}$,则由 Minkowsky 定理可知,格 L_i 的第二短长度向量的范数大约为$(2^{\delta+1})^{1/\Theta}$.因此,当$\Theta$足够大时,目前格归约算法无法得到向量 $v=(0,1,\omega_1,\ldots,\omega_\Theta)$.

实际上,在文献[2,7,17]的 FHE 方案中,SSSP 是隐藏的 SSSP(HSSSP),即,敌手并不知道需要求解的子集和值.因此,即使存在有效求解 SSSP 算法,该算法也不一定能够求解 HSSSP.本文考虑 HSSSP 的原因在于:一方面,在文献[2,7,17]的 FHE 方案中,SSSP 问题易于受到格归约攻击,因为 SSSP 构造的格中最短长度向量通常为 SSSP 的解;另一方面,使用 HSSSP 假设能够减少公钥大小.

5.4.2 稀疏子集和的生日攻击

在文献[4,7]中,当针对 FHE 方案安全进行分析时,作者们认为稀疏子集和问题存在生日攻击问题.生日攻击原指一群人中发生两人为同一天生日的概率,这里,同一天生日是指一年 365 天中任意一天发生两人生日相同均可[61].

给定 $y=(y_1,\ldots,y_\Theta),y_j\in R_{\mathbb{R},\delta},j\in[\Theta]$ 和 h^{-1},我们可以定义 Hash 函数$w_y(x)=\sum_{j=1}^{\Theta}x_j y_j \bmod 2,x\in\{0,1\}^{\Theta}$.针对$w_y(x)$的生日攻击,是指随机产生多少个 Hash 值会以高概率发生一次碰撞.易于计算,需要随机产生$(2^{\delta+1})^{k/2}$ 个 Hash 值发生一次碰撞的概率大约为 0.328(等价于生日攻击中的$\sqrt{365}$).

在稀疏子集和问题中并不存在生日攻击问题,因为方程$\sum_{j=1}^{\Theta}x_j y_j=h^{-1} \bmod 2$ 的解为 $x\in\{0,1\}^{\Theta}$,所以稀疏子集和问题是该方程有多少解.根据剩余 Hash 引理 2.1,如果$y=(y_1,\ldots,y_\Theta)$和 x 都是随机的,则$\sum_{j=1}^{\Theta}x_j y_j \bmod 2$ 在 $R_{\mathbb{R},\delta}$ 几乎也是均匀的.尽管稀疏子集和问题中存在一个 Hamming 权重θ的解$\omega=(\omega_1,\ldots,\omega_\Theta)$满足$\sum_{j=1}^{\Theta}x_j y_j=h^{-1} \bmod 2$, 但对于不等于$\omega$的任意 x,$\sum_{j=1}^{\Theta}x_j y_j \bmod 2$ 在 $R_{\mathbb{R},\delta}$ 上是几乎均匀的.因此,集合$\{0,1\}^{\Theta}$中不等于ω的任意 x 满足方程$\sum_{j=1}^{\Theta}x_j y_j=h^{-1} \bmod 2$ 的概率几乎为 0,即,以概率为 1 仅有ω一个解.所以,通过蛮力攻击必须猜测到解ω才可行,即:在 FHE 中,猜测隐藏 SSSP 问题的复杂性为$O((\Theta/\theta)^{\theta})$,而不是文献[4,7]中给出的$O((\Theta/\theta)^{\theta/2})$.

6 批全同态加密方案

在FHE方案中,我们能够扩展明文消息空间从$\{0,1\}$到$\{0,1\}^k$,以减少密文膨胀率从$O(k\rho^2)$到$O(\rho^2)$.给定消息$m\in\{0,1\}^k$,我们映射它到多项式$m(x)=\sum_{i=0}^{k-1}m_ix^i$.加密算法FHE.Enc变为$c=(m(x)+2r+2\sum r_{i,j}f_{1,i}f_{2,j})\bmod f_0$.解密算法(*FHE.Dec*(*SK*,*c*))变为:

(1) 计算$u_{i,j}=[\langle 2^\sigma c\times y_{i,j}\rangle/2^\sigma]_2$,即$u_{i,j}$最接近多项式$c\times y_{i,j}$;

(2) 计算$\overline{u}_{i,j}=u_{i,j}\times\omega_{i,j}$;

(3) 计算每个小块和$\overline{u}_i=\sum_{j=1}^{\Theta/\theta}\overline{u}_{i,j}$;

(4) 计算多项式和$\overline{u}=[\sum_{i=1}^{\theta}\overline{u}_i]_2$;

(5) 输出明文$m(x)=[c]_2\oplus\overline{u}_0\oplus\overline{u}_{-1}$,这里,记$\overline{u}=\overline{u}_0.\overline{u}_{-1}.....\overline{u}_{-\sigma}$,$\overline{u}_j$为多项式$\overline{u}$中每个系数按比特展开时,第$j$比特位组成的多项式.

如果需要进行比特密文计算,则在同态密文计算之前,首先将多比特密文解包成单个比特密文,然后再按照FHE算法进行正常密文同态计算.

密文解包算法. *FHE.Unpack*(*PK*,*c*).

(1) 计算$u_{i,j}=[\langle 2^\sigma c\times y_{i,j}\rangle/2^\sigma]_2$;

(2) 转换$u_{i,j}$为密文形式多项式$\vec{u}_{i,j}=u_{i,j}\times\vec{\omega}_{i,j}$;

(3) 计算小块密文形式多项式和$\vec{u}_i=\sum_{j=1}^{\Theta/\theta}\vec{u}_{i,j}$;

(4) 使用对称多项式方法[2,7]计算密文形式多项式和$\vec{u}=[\sum_{i=1}^{\theta}\vec{u}_i]_2$;

(5) 输出新密文多项式$c_{new}(x)=[c]_2\oplus\vec{u}_0\oplus\vec{u}_{-1}$,$c_{new}(x)$中的$k$个比特密文系数可以单独参加后续比特密文计算.这里,记密文形式多项式$\vec{u}=\vec{u}_0.\vec{u}_{-1}.....\vec{u}_{-\sigma}$,$\vec{u}_j$为多项式$\vec{u}$中每个系数按比特展开时,第$j$比特位组成的密文多项式.

在密文同态计算结束后,需要将k个比特密文打包成一个新密文.假定$c_i,i\in[k]$为比特密文,则打包密文为

$$c(x)=(\sum_{i=1}^{k}c_i\times x^{i-1})\bmod(x^k+1)\bmod f_0.$$

7 基于AILP的FHE

在基于PAILP的FHE中,公钥中包括无噪声的h倍数多项式$f_0=g_0h$,使得方案安全性依赖于大整数分解难度假设.然而,目前存在量子多项式时间算法分解大整数[62].实际上,我们可以构造基于近似理想格的全同态加密方案.然而,目前我们并不能归约基于AILP的SHE1安全性到近似理想格问题.但SHE1安全性能够归约到部分近似理想格问题,即,基于AILP的SHE1安全性不比SHE的安全性低.因为从SHE1构造全同态加密方案FHE1的方法与构造FHE的方法相同,所以下面仅给出基于AILP的SHE1方案.

密钥生成算法. SHE1.KeyGen.

(1) 随机选择多项式$h\in R$满足$h \bmod 2=1$和$\|h\|_\infty=2^\lambda$;

(2) 随机选择$g_j,e_j\in R,j\in[\tau^2]$满足$\|g_j\|_\infty\in[2^\eta]$,$\|e_j\|_\infty\in[2^\rho]$,计算输出$f_j=hg_j+e_j$.设$f_0=hg_0+2e_0$满足$\|g_0\|_\infty\in[2^\eta]$,$\|e_0\|_\infty\in[2^\rho]$,$\|f_0\|_\infty\geqslant\|f_j\|_\infty$和$f_0 \bmod 2\neq 0$;

(3) 随机选择$\beta=O((\lambda+\eta)/\rho)$对$a_i,e_i\in R,i\in[\beta]$满足$\|a_i\|_\infty\approx 2^{\eta+\rho i}$和$\|e_i\|_\infty\in[2^\rho]$,计算$\{d_i=a_ih+2e_i\}_{i=1}^{\beta}$.计算验证$\|d_i^{-1}\|_\infty\approx 2^{-(\lambda+\eta+\rho i)}$,如果不满足,则重新选择$a_i,e_i\in R$;

(4) 输出公钥$pk=(k,f_0,\{f_j\}_{j\in[\tau^2]},\{d_i\}_{i\in[\beta]})$和私钥$sk=(h)$.

加密算法. SHE1.Enc.

给定公钥pk和比特$m\in\{0,1\}$,随机选择$r_j\in R,j\in[\tau^2]$和$r\in R$满足$\|r_j\|_\infty\in[2^\rho]$和$\|r\|_\infty\in[2^\rho]$.计算输出密文:

$$c=(m+2r+\sum 2r_jf_j)\bmod f_0.$$

密文加算法. SHE1.Add.

给定公钥 pk 和密文 c_1,c_2,计算输出密文 $c=(c_1+c_2) \bmod f_0$.

密文乘算法. SHE1.Mul.

给定公钥 pk 和密文 c_1,c_2,计算输出密文 $c=Opt(c_1\times c_2)$,这里,Opt 是类似于文献[2]中的优化方法,即:

$$c=(c_1\times c_2) \bmod d_t \bmod d_{t-1} \cdots \bmod d_1 \bmod f_0.$$

解密算法. SHE1.Dec.

给定 sk 和密文 c,计算输出 $m=[\langle c\times h^{-1}\rangle \bmod x]_2 \oplus [c \bmod x]_2$.

评论 7.1.

(1) SHE1 计算 $c \bmod d_i$ 实际上是计算 $c \bmod Rot(d_i)$,即,将 $\vec{c}$ 映射到格基 $Rot(d_i)$的平行四边形内部;

(2) 因为 SHE1.Enc 中的 $\| m+2r+\sum 2r_j f_j \|_\infty / \| f_0 \|_\infty$ 较小,即,mod f_0 时引入噪声范数较小,所以能够直接计算 $c=(m+2r+\sum 2r_j f_j) \bmod f_0$. 但在密文乘法时,$\|c_1\times c_2\|_\infty=O(2^{2(\lambda+\eta)})$远大于$\|f_0\|_\infty$,如果在这种情况下直接计算$(c_1\times c_2) \bmod f_0$,则引入的噪声太大,使得明文不能正确恢复.因此,我们使用 Opt 优化算法将 $c_1\times c_2$ 按阶梯分步映射,并最终映射到格基 $Rot(f_0)$的平行四边形内部.注意:这也是 SHE1.Enc 使用公钥元素线性函数的原因;而且为了避免格攻击,公钥元素 f_j 的数目需要 $\tau^2=O(\rho^4)$;

(3) $h\in R$ 仅需满足 $h \bmod 2=1$,不要求 $\det(Rot(h))$为大素数或难分解合数,因为对手不知道 $\det(Rot(h))$;另一方面,为了避免中国剩余定理的攻击,需要 $\det(Rot(h))$没有小因子(如最小因子大于 2^ρ).因为若 $\det(Rot(h))$中存在小因子积大于噪声多项式的极大范数,则敌手可针对每个因子猜测其噪声多项式,然后利用中国剩余定理求解出真正噪声多项式,并进而去除它.

引理 7.1. SHE1.KeyGen 算法运行在概率多项式时间.

证明:易于看出,SHE1.KeyGen 的步骤(1)、步骤(2)运行在多项式时间,SHE1.KeyGen 的步骤(3)实际上是要求 d_i 的逆 d_i^{-1} 的长度满足条件 $\| d_i^{-1} \|_\infty \approx (\| d_i \|_\infty)^{-1}$. 因为$\|d_i\|_\infty \approx 2^{\lambda+\eta+\rho i}$,即,要求 $\| d_i^{-1} \|_\infty \approx 2^{-(\lambda+\eta+\rho i)}$. 设 $p=res(d_i,x^k+1)$为多项式 d_i,x^k+1 的结式(resultant),易于验证 $p=\det(Rot(d_i))$.由文献[3]中的引理 1 可知:$\| pd_i^{-1} \|_\infty \leqslant \| d_i \|_2^{k-1} \| x^k+1 \|_2^{k-1}$,并且以高概率 $p \approx \| d_i \|_2^k \| x^k+1 \|_2^{k-1}$,所以 $\| d_i^{-1} \|_\infty \leqslant \| d_i \|_2^{-1} \approx 1/(\sqrt{k}\, 2^{(\lambda+\eta+\rho i)}) \approx 2^{-(\lambda+\eta+\rho i)}$. 因此,SHE1.KeyGen 能够在概率多项式时间内产生公钥和私钥. □

引理 7.2. 如果$\|d^{-1}\|_\infty \approx (\|d\|_\infty)^{-1}$ 和$\|c\|_\infty/\|d\|_\infty \leqslant 2^\rho$,则 $c \bmod d=c-qd$ 中的 q 满足$\|q\|_\infty \approx k2^\rho$.

证明:因为 $c \bmod d=c-qd$,则 $q=d^{-1}(c-c \bmod d)$.所以:

$$\begin{aligned}\| q \|_\infty &= \| d^{-1}(c-c \bmod d) \|_\infty \\ &\leqslant k \| d^{-1} \|_\infty \| (c-c \bmod d) \|_\infty \\ &\leqslant k \| d^{-1} \|_\infty (\| c \|_\infty + \| c \bmod d \|_\infty) \\ &\leqslant k \| d^{-1} \|_\infty (\| c \|_\infty + \| d \|_\infty) \\ &\approx k2^\rho.\end{aligned}$$

证毕. □

从引理 7.2 可以看出,Opt 算法每次模 d_i 在密文中增加的噪声多项式的范数较小.因此易于验证,密文 $c=Opt(c_1\times c_2)$中的噪声多项式大小由密文 c_1,c_2 中噪声多项式积的大小控制.

所以,只要选定适当参数,方案 SHE1 支持有点同态密文计算.然后,使用 Gentry 引导技术,易于将 SHE1 转换为全同态加密方案 FHE1.

8 实现 FHE

8.1 实现FHE

我们使用 NTL 库[26]实现基于 AILP/PAILP 的 FHE.在实现 FHE 时,主要考虑因素是在一定安全级别上(如攻击难度为 2^{72}),尽可能地提高方案实用性,包括密文膨胀率、同态计算复杂性、公钥大小等.

由第 5 节的 FHE 可知,方案的安全性依赖于如下 3 个方面:

(1) 大整数分解难度.目前,在经典计算机上分解方案中使用的大整数仍然不够现实;

(2) 隐藏稀疏子集和攻击难度.使用文献[7]中的优化技术,将稀疏子集分成小块,每个小块中仅一个元素属于私钥的稀疏子集和,本文所有实验都取Θ=960,θ=15,$B=\Theta/\theta$=64,$\sigma=\lceil\log(\theta+1)\rceil$=4.根据本文稀疏子集和的生日攻击分析,蛮力猜测攻击复杂性估计为 $64^{15}=2^{90}$;

(3) 蛮力攻击 PAILP 的噪声多项式,该攻击复杂性约为 $2^{k\rho}$,$\rho=\log\|e\|_\infty$或为时间和空间复杂性都为 $2^{(k-1)\rho}$(根据第 4.2.4 节分析).蛮力攻击 AILP 的复杂性至少为 $2^{k\rho}$.

下面描述当 k=4 时,实现 FHE 方案中各个参数值大小.

首先估计 $\log\|h\|_\infty$的大小.当加密私钥比特的密文中噪声多项式满足$\rho=\log\|2e\|_\infty$时,则 FHE.Recrypt 算法计算小块 B 个密文和数 $\vec{u}_i=\sum_{j=1}^{B}\vec{u}_{i,j}$ 中的噪声大小变为$\rho'=\rho+6$.计算θ=15 个密文数和 $\vec{u}=[\sum_{i=1}^{\theta}\vec{u}_i]_2$ 需要使用度为 15 的对称多项式.易于计算,一个度为 15 的多项式密文的噪声大小为 $15\rho'+28$.根据文献[7]的分析,度为 15 的多项式的密文数目至多为 2^{34}.故密文数和 $\vec{u}$ 中的噪声大小至多为 $15\rho'+28+34=15\rho'+62$.因为密文刷新后需要能支持至少一次密文乘法运算,故$\rho''=2(15\rho'+62)+2=30\rho'+126$.又因为要求多项式$(c\times h^{-1}) \bmod x$ 的每个系数与整数的最小距离在 $1/(2\theta)$内,故 $\log\|h^{-1}\|_\infty\leqslant-(30\rho'+131)$.

所以,$\log\|h\|_\infty$应略大于 $30\rho'+131$,以满足条件 $\log\|h^{-1}\|_\infty\leqslant-(30\rho'+131)$.

如果设ρ=24,则$\lambda=\log\|h\|_\infty=30\rho'+126\approx1036$,$\eta\approx51800>\lambda$,$\tau$=512.

为了提高效率,每次加密 $c=(m+2r+2\sum r_{i,j}f_{1,i}f_{2,j}) \bmod f_0$ 时,从 τ^2 对中随机选择 20 个 $r_{i,j}\in\{-1,1\}$,其余都为 0,并且$\|2r\|_\infty\in[2^\rho]$.

类似地,对于其他 k,ρ不同取值的组合,不难计算其他相应参数值的大小.

8.2 FHE性能比较

我们通过表 1 和表 2 给出本文与以前 FHE 的性能比较.表 1 给出实现 FHE 方案的具体运行环境,表 2 中给出实现的 FHE 方案性能情况.从表 2 可以看出,基于近似理想格的 FHE 方案比已有方案性能更好.

Table 1 Concrete running setting of implementing FHE schemes

表 1 实现 FHE 方案的具体运行环境

方案	机器环境	软件库
GH[7]	3GHz 64-bit quad-core Intel Xeon E5450处理器,12MB L2 cache, 24GB RAM	NTL 5.5.2
CMNT[4]	3.12GHz Intel core2 Duo E8500 CPU桌面机	Sage 4.5.3,GMP 4.3.2
CNT[5]	3GHz Intel core2 Duo E8400桌面机	Sage 4.7.2
本文	Intel Xeon E5620 4核CPU,主频2.4GHz	NTL 5.5.2

Table 2 Performance comparison of implementing FHE schemes

表 2 FHE 方案性能比较

方案/困难问题	维数k	KeyGen	Encrypt	Decrypt	Recrypt	PK大小	密文膨胀率	期望安全级别
GH/理想格[7]	32 768	2.2h	3m	0.66s	31m	2.25GB	1.263×10^7	≥72
GH/理想格[7]	2 048	41s	1.8s	0.02s	32s	69MB	7.850×10^5	≥72
CMNT/PAGCD[4]	1	43m	3m	0.05s	14.5m	802MB	1.900×10^7	≥67[24]
CNT/PAGCD[5]	1	10m	7.25m	0.05s	11.5m	10.3MB	1.935×10^7	≥72
本文/PAILP	64	9h	1.62s	0.082s	93.7s	170MB	1.152×10^5	≥72
本文/PAILP	32	25m	1.38s	0.062s	74.4s	173MB	1.232×10^5	≥72
本文/PAILP	16	16m	0.23s	0.041s	58.6s	212MB	1.512×10^5	≥72
本文/PAILP	4	4.0h	0.82s	0.034s	9.6s	290MB	2.114×10^5	≥72
本文/AILP	4	4.5h	0.18s	0.014s	978.4s	760MB	2.114×10^5	≥72

注:(1) h,m,s 分别为时间单位小时、分钟、秒;(2) 如果使用批 FHE 方案,上述基于 AILP(PAILP)的密文膨胀率会减少 k 倍;(3) 本文实现FHE方案时选取参数τ=512;(4) 基于AILP的FHE密文刷新时间较大的主要原因是:每次密文乘法时需要调用优化算法Opt,以逐步减少乘法后密文到正常密文的大小;(5) 期望安全级别栏中数字是指攻击比特长度,该值仅为实现方案的安全性估计,并不表示实现方案的实际安全性.实际上,η取值越大,方案安全性越高.

9 结论和公开问题

本文首先构造了基于 PAILP 的 SHE,并归约证明 SHE 安全性到求解 PAILP;其次,使用 Gentry 引导技术转换 SHE 到 FHE;然后,分别构造了基于 PAILP 的批同态加密方案和基于 AILP 的全同态加密方案;最后,实现了基于近似理想格的全同态加密方案,并与现有全同态加密方案的性能进行了比较.

我们注意到,Coron,Naccache 和 Tibouchi[5]使用模切换技术[11]构造了 DGHV 型的无引导层次全同态加密方案.方案构造的关键是文献[5]中的引理 4 和引理 5,这两个引理给出了在密文状态下如何实现近似 GCD 问题从私钥 p 到私钥 p'的正确切换,并且保证切换后近似 GCD 问题的噪声从 r 减少到约为 $r\cdot(p'/p)\approx r\cdot 2^{-\rho}$.因为本文的近似理想格问题是近似 GCD 问题的扩展,易于验证文献[5]中的引理 4 和引理 5 在近似理想格上同样成立.在自适应文献[5]中整数近似 GCD 上的引理证明到近似理想格上时,需将整数近似 GCD 替换成近似理想格,参数大小分析时使用的绝对值替换成近似理想格上的极大范数,而且与整数上条件 $p'/p\approx 2^{-\rho}$类似,需要保证进行切换的近似理想格 h',h 满足条件$\|h'/h\|_\infty\approx 2^{-\rho}$,证明过程完全相同.因此,使用文献[5]中同样的方法,我们能够自适应地构造出基于近似理想格的无引导的层次全同态加密方案.当然,基于近似理想格的无引导的层次全同态加密方案效率会更高,因为可以使用$\|h\|_\infty$更小的私钥 h.

本文需进一步研究的问题:证明基于 AILP 的 SHE 安全性;研究分析问题 AILP/PAILP 的求解难度,并建立 AILP 与理想格问题之间的关系;研究分析基于 AILP/PAILP 的 FHE 方案的实际安全性.

致谢 在此,我们感谢本文的匿名审稿专家所提供的宝贵修改意见.本文的数值计算得到了中国科学技术大学超级计算中心的计算支持和帮助,作者在此一并表示感谢.

References:

[1] Gentry C. Fully homomorphic encryption using ideal lattices. In: Mitzenmacher M, ed. Proc. of the 41st Annual ACM Symp. on Theory of Computing (STOC 2009). New York: Association for Computing Machinery, 2009. 169−178. [doi: 10.1145/1536414.1536440]

[2] van Dijk M, Gentry C, Halevi S, Vaikuntanathan V. Fully homomorphic encryption over the integers. In: Gilbert H, ed. Proc. of the EUROCRYPT 2010. LNCS 6110, Heidelberg: Springer-Verlag, 2010. 24−43. [doi: 10.1007/978-3-642-13190-5_2]

[3] Smart NP, Vercauteren F. Fully homomorphic encryption with relatively small key and ciphertext sizes. In: Nguyen PQ, Pointcheval D, eds. Proc. of the Public Key Cryptography (PKC 2010). LNCS 6056, Heidelberg: Springer-Verlag, 2010. 420−443. [doi: 10.1007/978-3-642-13013-7_25]

[4] Coron JS, Mandal A, Naccache D, Tibouchi M. Fully homomorphic encryption over the integers with shorter public keys. In: Rogaway P, ed. Proc. of the CRYPTO 2011. LNCS 6841, Heidelberg: Springer-Verlag, 2011. 487−504. [doi: 10.1007/978-3-642-22792-9_28]

[5] Coron JS, Naccache D, Tibouchi M. Public key compression and modulus switching for fully homomorphic encryption over the integers. In: Pointcheval D, Johansson T, eds. Proc. of the EUROCRYPT 2012. LNCS 7237, Heidelberg: Springer-Verlag, 2012. 446−464. [doi: 10.1007/978-3-642-29011-4_27]

[6] Cheon JH, Coron JS, Kim J, Lee MS, Lepoint T, Tibouchi M, Yun A. Batch fully homomorphic encryption over the integers. In: Johansson T, Nguyen P, eds. Proc. of the EUROCRYPT 2013. LNCS 7881, Heidelberg: Springer-Verlag, 2013. 315−335. [doi: 10.1007/978-3-642-38348-9_20]

[7] Gentry C, Halevi S. Implementing Gentry's fully-homomorphic encryption scheme. In: Paterson KG, ed. Proc. of the EUROCRYPT 2011. LNCS 6632, Heidelberg: Springer-Verlag, 2011. 129−148. [doi: 10.1007/978-3-642-20465-4_9]

[8] Gentry C, Halevi S. Fully homomorphic encryption without squashing using depth-3 arithmetic circuits. In: Proc. of the 2011 IEEE 52nd Annual Symp. on Foundations of Computer Science (FOCS 2011). Washington: IEEE Computer Society, 2011. 107−116. [doi: 10.1109/FOCS.2011.94]

[9] Brakerski Z, Vaikuntanathan V. Fully homomorphic encryption from ring-LWE and security for key dependent messages. In: Rogaway P, ed. Proc. of the CRYPTO 2011. LNCS 6841, Heidelberg: Springer-Verlag, 2011. 505−524. [doi: 10.1007/978-3-642-22792-9_29]

[10] Brakerski Z, Vaikuntanathan V. Efficient fully homomorphic encryption from (Standard) LWE. In: Proc. of the 2011 IEEE 52nd Annual Symp. on Foundations of Computer Science (FOCS 2011). Washington: IEEE Computer Society, 2011. 97−106. [doi: 10.1109/FOCS.2011.12]

[11] Brakerski Z, Gentry C, Vaikuntanathan V. (Leveled) Fully homomorphic encryption without bootstrapping. In: Proc. of the 3rd Innovations in Theoretical Computer Science Conf. (ITCS 2012). New York: Association for Computing Machinery, 2012. 309−325. [doi: 10.1145/2090236.2090262]

[12] Gentry C, Halevi S, Smart NP. Better bootstrapping in fully homomorphic encryption. In: Fischlin M, Buchmann J, Manulis M, eds. Proc. of the Public Key Cryptography (PKC 2012). LNCS 7293, Heidelberg: Springer-Verlag, 2012. 1−16. [doi: 10.1007/978-3-642-30057-8_1]

[13] Gentry C, Halevi S, Smart NP. Fully homomorphic encryption with polylog overhead. In: Pointcheval D, Johansson T, eds. Proc. of the EUROCRYPT 2012. LNCS 7237, Heidelberg: Springer-Verlag, 2012. 465−482. [doi: 10.1007/978-3-642-29011-4_28]

[14] Gentry C, Halevi S, Smart NP. Homomorphic evaluation of the AES circuit. In: Safavi-Naini R, Canetti R, eds. Proc. of the CRYPTO 2012. LNCS 7417, Heidelberg: Springer-Verlag, 2012. 850−867. [doi: 10.1007/978-3-642-32009-5_49]

[15] Brakerski Z. Fully homomorphic encryption without modulus switching from classical GapSVP. In: Safavi-Naini R, Canetti R, eds. Proc. of the CRYPTO 2012. LNCS 7417, Heidelberg: Springer-Verlag, 2012. 868−886. [doi: 10.1007/978-3-642-32009-5_50]

[16] Lopez-Alt A, Tromer E, Vaikuntanathan V. On-the-Fly multiparty computation on the cloud via multikey fully homomorphic encryption. In: Proc. of the 44th Annual ACM Symp. on Theory of Computing (STOC 2012). New York: Association for Computing Machinery, 2012. 1219−1234. [doi: 10.1145/2213977.2214086]

[17] Stehle D, Steinfeld R. Faster fully homomorphic encryption. In: Abe M, ed. Proc. of the ASIACRYPT 2010. LNCS 6477, Heidelberg: Springer-Verlag, 2010. 377−394. [doi: 10.1007/978-3-642-17373-8_22]

[18] Gentry C, Sahai A, Waters B. Homomorphic encryption from learning with errors: Conceptually-Simpler, asymptotically-faster, attribute-based. In: Canetti R, Garay JA, eds. Proc. of the CRYPTO 2013. LNCS 8042, Heidelberg: Springer-Verlag, 2013. 75−92. [doi: 10.1007/978-3-642-40041-4_5]

[19] Brakerski Z, Vaikuntanathan V. Lattice-Based FHE as secure as PKE. In: Proc. of the 5th Conf. on Innovations in Theoretical Computer Science (ITCS 2014). New York: Association for Computing Machinery, 2014. 1−12. [doi: 10.1145/2554797.2554799]

[20] Gu CS, Gu JX. Cryptanalysis of the smart-vercauteren and Gentry-Halevi's fully homomorphic encryption. Int'l Journal of Security and Its Applications, 2012,6(2):103−108.

[21] Nguyen P, Stehle D. LLL on the average. In: Hess F, Pauli S, Pohst M, eds. Proc. of the ANTS 2006. LNCS 4076, Heidelberg: Springer-Verlag, 2006. 238−256. [doi: 10.1007/11792086_18]

[22] Lenstra HW, Lenstra AK, Lovasz L. Factoring polynomials with rational coefficients. Mathematische Annalen, 1982,261(4): 515−534. [doi: 10.1007/BF01457454]

[23] Hu YP, Wang FH. An attack on a fully homomorphic encryption scheme, ePrint archive. Technical Report, 2012/561, 2012. http://eprint.iacr.org/2012/561

[24] Chen Y, Nguyen PQ. Faster algorithms for approximate common divisors: Breaking fully homomorphic encryption challenges over the integers. In: Pointcheval D, Johansson T, eds. Proc. of the EUROCRYPT 2012. LNCS 7237, Heidelberg: Springer-Verlag, 2012. 502−519. [doi: 10.1007/978-3-642-29011-4_30]

[25] Lauter K, Naehrig M, Vaikuntanathan V. Can homomorphic encryption be practical. In: Proc. of the 3rd ACM Workshop on Cloud Computing Security Workshop (CCSW 2011). New York: Association for Computing Machinery, 2011. 113−124. [doi: 10.1145/2046660.2046682]

[26] Shoup V. NTL: A library for doing number theory. Version 7.0.2, 2009. http://shoup.net/ntl/

[27] Gu CS, Wu FS. On fully homomorphic encryption, approximate lattice problem and LWE. Int'l Journal of Cloud Computing and Services Science, 2013,2(1):1−15. [doi: 10.11591/closer.v2i1.1339]

[28] Zhang Y, Wen T, Guo Q, LI FK. Pair-Wise key establishment for wireless sensor networks based on fully homomorphic encryption. Journal on Communications, 2012,33(10):101−109 (in Chinese with English abstract).

[29] Chen JY, Wang C, Zhang WM, Zhu YF. A secure image steganographic method in encrypted domain. Journal of Electronics & Information Technology, 2012,34(7):1721−1726 (in Chinese with English abstract). [doi: 10.3724/SP.J.1146.2011.01240]

[30] Sun ZW, Feng DG, Wu CK. An anonymous fingerprinting scheme based on additively homomorphic public key cryptosystem. Ruan Jian Xue Bao/Journal of Software, 2005,16(10):1816−1821 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/16/1816.htm

[31] Guang Y, Gu CX, Zhu YF, Zheng YH, Fei JL. Certificateless fully homomorphic encryption based on LWE problem. Journal of Electronics & Information Technology, 2013,35(4):988−993 (in Chinese with English abstract). [doi: 10.3724/SP.J.1146.2012.01102]

[32] Feng DG, Zhang M, Zhang Y, Xu Z. Study on cloud computing security. Ruan Jian Xue Bao/Journal of Software, 2011,22(1):71−83 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3958.htm [doi: 10.3724/SP.J.1001.2011.03958]

[33] Yu NH, Hao Z, Xu JJ, Zhang WM, Zhang C. Review of cloud computing security. Acta Electronica Sinica, 2013,41(2):371−381 (in Chinese with English abstract). [doi: 10.3969/j.issn.0372-2112.2013.02.026]

[34] Cheng FQ, Peng ZY, Song W, Wang SL, Cui YH. An efficient privacy-preserving rank query over encrypted data in cloud computing. Chinese Journal of Computers, 2012,35(11):2215−2227 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2012.02215]

[35] Cai K, Zhang M, Feng DG. Secure range query with single assertion on encrypted data. Chinese Journal of Computers, 2011,34(11):2093−2103 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2011.02093]

[36] Zhu Q, Zhao T, Wang S. Privacy preservation algorithm for service-oriented information search. Chinese Journal of Computers, 2010,33(8):1315−1323 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2010.01315]

[37] Regev O. On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM (JACM), 2009,56(6):1−40. [doi: 10.1145/1060590.1060603]

[38] Lyubashevsky V, Peikert C, Regev O. On ideal lattices and learning with errors over rings. In: Gilbert H, ed. Proc. of the EUROCRYPT 2010. LNCS 6110, Heidelberg: Springer-Verlag, 2010. 1−23. [doi: 10.1007/978-3-642-13190-5_1]

[39] Alperin-Sheriff J, Peikert C. Practical bootstrapping in quasilinear time. In: Canetti R, Garay JA, eds. Proc. of the CRYPTO 2013. LNCS 8042, Heidelberg: Springer-Verlag, 2013. 1−20. [doi: 10.1007/978-3-642-40041-4_1]

[40] Alperin-Sheriff J, Peikert C. Faster bootstrapping with polynomial error. In: Garay JA, Gennaro R, eds. Proc. of the CRYPTO 2014. LNCS 8616, Heidelberg: Springer-Verlag, 2014. 297−314. [doi: 10.1007/978-3-662-44371-2_17]

[41] Halevi S, Shoup V. Algorithms in HElib. In: Garay JA, Gennaro R, eds. Proc. of the CRYPTO 2014. LNCS 8616, Heidelberg: Springer-Verlag, 2014. 554−571. [doi: 10.1007/978-3-662-44371-2_31]

[42] Kannan R. Improved algorithms for integer programming and related lattice problems. In: Proc. of the 15th Annual ACM Symp. on Theory of Computing (STOC'83). New York: Association for Computing Machinery, 1983. 193−206. [doi: 10.1145/800061.808749]

[43] Schnorr CP. A hierarchy of polynomial time lattice basis reduction algorithms. Theoretical Computer Science, 1987,53(2-3):201−224. [doi: 10.1016/0304-3975(87)90064-8]

[44] Ajtai M, Kumar R, Sivakumar D. A sieve algorithm for the shortest lattice vector problem. In: Proc. of the 33rd Annual ACM Symp. on Theory of Computing (STOC 2001). New York: Association for Computing Machinery, 2001. 601−610. [doi: 10.1145/380752.380857]

[45] Gama N, Howgrave-Graham N, Koy H, Nguyen PQ. Rankin's constant and blockwise lattice reduction. In: Dwork C, ed. Proc. of the CRYPTO 2006. LNCS 4117, Heidelberg: Springer-Verlag, 2006. 112−130. [doi: 10.1007/11818175_7]

[46] Gama N, Nguyen P. Finding short lattice vectors within Mordell's inequality. In: Proc. of the 40th Annual ACM Symp. on Theory of Computing (STOC 2008). New York: Association for Computing Machinery, 2008. 208−216. [doi: 10.1145/1374376.1374408]

[47] Gama N, Nguyen P. Predicting lattice reduction. In: Smart N, ed. Proc. of the EUROCRYPT 2008. LNCS 4965, Heidelberg: Springer-Verlag, 2008. 31−51. [doi: 10.1007/978-3-540-78967-3_3]

[48] Nguyen P, Vidick T. Sieve algorithms for the shortest vector problem are practical. Journal of Mathematical Cryptology, 2008,2(2):181−207.

[49] Becker A, Gama N, Joux A. A sieve algorithm based on overlattices. LMS Journal of Computation and Mathematics, 2014,17(Special Issue A):49−70. [doi: 10.1112/S1461157014000229]

[50] Micciancio D, Voulgaris P. Faster exponential time algorithms for the shortest vector problem. In: Proc. of the 21st Annual ACM-SIAM Symp. on Discrete Algorithms (SODA 2010). Society for Industrial and Applied Mathematics, 2010. 1468−1480. http://dl.acm.org/citation.cfm?id=1873601.1873720&coll=DL&dl=ACM&CFID=715343813&CFTOKEN=29694473

[51] Wang XY, Liu MJ, Tian CL, Bi JG. Improved Nguyen-Vidick heuristic sieve algorithm for shortest vector problem. In: Proc. of the 6th ACM Symp. on Information, Computer and Communications Security (ASIACCS 2011). New York: Association for Computing Machinery, 2011. 1−9. [doi: 10.1145/1966913.1966915]

[52] Ajtai M, Dwork C. A public-key cryptosystem with worst-case/average-case equivalence. In: Proc. of the 29th Annual ACM Symp. on Theory of Computing (STOC'97). New York: Association for Computing Machinery, 1997. 284−293. [doi: 10.1145/258533.258604]

[53] Regev O. New lattice-based cryptographic constructions. Journal of the ACM, 2004,51(6):899−942. [doi: 10.1145/1039488.1039490]

[54] von zur Gathen J, Gerhard J. Modern Computer Algebra. 3rd ed., Cambridge: Cambridge University Press, 2013.

[55] Boyar J, Peralta R, Pochuev D. On the multiplicative complexity of Boolean functions over the basis (∧,⊕,1). Theoretical Computer Science, 2000,235(1):43−57. [doi: 10.1016/S0304-3975(99)00182-6]

[56] Hastad J, Impagliazzo R, Levin LA, Luby M. A pseudorandom generator from any one-way function. SIAM Journal on Computing, 1999,28(4):1364−1396. [doi: 10.1137/S0097539793244708]

[57] Vaikuntanathan V. Computing blindfolded: New developments in fully homomorphic encryption. In: Proc. of the 2011 IEEE 52nd Annual Symp. on Foundations of Computer Science (FOCS 2011). Washington: IEEE Computer Society, 2011. 5−16. [doi: 10.1109/FOCS.2011.98]

[58] Howgrave-Graham N. Approximate integer common divisors. In: Silverman JH, ed. Proc. of the CaLC 2001. LNCS 2146, Heidelberg: Springer-Verlag, 2001. 51−66. [doi: 10.1007/3-540-44670-2_6]

[59] Nguyen PQ, Vallée B. The LLL Algorithm: Survey and Applications. Heidelberg: Springer-Verlag, 2009. 33−35. [doi: 10.1007/978-3-642-02295-1]

[60] Barak B, Haitner I, Hofheinz D, Ishai Y. Bounded key-dependent message security. In: Gilbert H, ed. Proc. of the EUROCRYPT 2010. LNCS 6110, Heidelberg: Springer-Verlag, 2010. 423−444. [doi: 10.1007/978-3-642-13190-5_22]

[61] Goldwasser S, Bellare M. Lecture notes on cryptography. 2008. 249−250. http://cseweb.ucsd.edu/~mihir/papers/gb.html

[62] Shor PW. Polynomial-Time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal Computing, 1997,26(5):1484−1509. [doi: 10.1137/S0097539795293172]

附中文参考文献:

[28] 张永,温涛,郭权,李凤坤.WSN 中基于全同态加密的对偶密钥建立方案.通信学报,2012,33(10):100−109.

[29] 陈嘉勇,王超,张卫明,祝跃飞.安全的密文域图像隐写术.电子与信息学报,2012,34(7):1721−1726. [doi: 10.3724/SP.J.1146.2011.01240]

[30] 孙中伟,冯登国,武传坤.基于加同态公钥密码体制的匿名数字指纹方案.软件学报,2005,16(10):1816−1821. http://www.jos.org.cn/1000-9825/16/1816.htm

[31] 光炎,顾纯祥,祝跃飞,郑永辉,费金龙.一种基于 LWE 问题的无证书全同态加密体制.电子与信息学报,2013,35(4):988−993. [doi: 10.3724/SP.J.1146.2012.01102]

[32] 冯登国,张敏,张妍,徐震.云计算安全研究.软件学报,2011,22(1):71−83. http://www.jos.org.cn/1000-9825/3958.htm [doi: 10.3724/SP.J.1001.2011.03958]

[33] 俞能海,郝卓,徐甲甲,张卫明,张驰.云安全研究进展综述.电子学报,2013,41(2):371−381. [doi: 10.3969/j.issn.0372-2112.2013.02.026]

[34] 程芳权,彭智勇,宋伟,王书林,崔一辉.云环境下一种隐私保护的高效密文排序查询方法.计算机学报,2012,35(11):2215−2227. [doi: 10.3724/SP.J.1016.2012.02215]

[35] 蔡克,张敏,冯登国.基于单断言的安全的密文区间检索.计算机学报,2011,34(11):2093−2103. [doi: 10.3724/SP.J.1016.2011.02093]

[36] 朱青,赵桐,王珊.面向查询服务的数据隐私保护算法.计算机学报,2010,33(8):1315−1323. [doi: 10.3724/SP.J.1016.2010.01315]

古春生(1971－),男,安徽芜湖人,博士,副教授,CCF 会员,主要研究领域为公钥密码学.