随机分布对格 KEM 安全性与性能的影响分析——以 Kyber 和 Saber 为例

邵明尧[1),2),3)] 刘月君[4)] 周永彬[1),4)] 邵 岩[4)] 张 倩[1),2)]

[1)](中国科学院信息工程研究所 北京 100085)

[2)](网络空间安全防御全国重点实验室 北京 100085)

[3)](中国科学院大学网络空间安全学院 北京 100049)

[4)](南京理工大学网络空间安全学院 南京 210000)

摘 要 随机分布类型及参数的选取对格密钥封装机制(KEM)的理论安全性与实现性能具有根本影响。为提升采样效率并规避高斯采样的计时侧信道攻击风险,当前主流格 KEM 多采用均匀分布(UD)或中心二项分布(CBD)。然而,二者对格 KEM 安全性与性能的作用机制仍缺乏系统性分析与对比研究。本文以 Kyber 和 Saber 为例,对 CBD 与 UD 的影响差异展开理论与实证分析。安全性方面,本文系统评估格攻击抗性与解密失败攻击风险。结果表明,相同取值区间下,UD 的标准差为 CBD 的 $\sqrt{4/3}$ 倍以上,使底层困难问题实例更难求解,从而将 Kyber 和 Saber 的安全强度提升约 12 比特~25 比特;然而,其代价是解密失败概率的急剧升高。针对高解密失败概率下失败位置难定位、私钥恢复复杂度过高的问题,本文提出迭代增强的失败定位方法和多数投票的密钥恢复方法,将 UD 下 Kyber512 和 LightSaber 的私钥恢复复杂度分别降至 $O(2^{37})$ 和 $O(2^{42})$,远低于原始 CBD 方案的 $O(2^{146})$ 和 $O(2^{128})$。性能方面,本文综合考虑存储开销与运行时间。在模数和压缩参数固定的条件下,分布变化不影响密钥和密文的存储开销;但由于拒绝采样机制,使得 UD 采样时间增加约 25%~65%。综上,随机分布选取对格 KEM 在抗格攻击能力、解密失败风险与实现性能方面产生显著而不均衡的影响。为平衡上述矛盾,本文引入非对称 LWE 问题以支持分布参数的灵活调节,并据此提出基于 UD 的变体 uKyber。在保持与原始 Kyber 可比的安全强度和解密失败概率的同时,uKyber 显著缩短密文长度(将 Kyber512 的密文长度由 768 字节降至 672 字节,减少约 12.5%),并通过多值拒绝采样提升 UD 采样效率,使其密钥生成和封装/解封装速度均优于原始 Kyber。

关键词 随机分布;格 KEM;安全强度;解密失败;拒绝采样

中图法分类号 TP309 **DOI 号** 10.11897/SP.J.1016.2026.01649

On the Impact of Random Distribution in the Security and Performance of Lattice-Based KEMs: A Case Study of Kyber and Saber

SHAO Ming-Yao[1),2),3)] LIU Yue-Jun[4)] ZHOU Yong-Bin[1),4)] SHAO Yan[4)] ZHANG Qian[1),2)]

[1)](Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100085)

[2)](Key Laboratory of Cyberspace Security Defense, Beijing 100085)

[3)](School of Cyber Security, University of the Chinese Academy of Sciences, Beijing 100049)

[4)](School of Cyber Science and Engineering, Nanjing University of Science and Technology, Nanjing 210000)

Abstract The choice of random distribution type and parameters plays a fundamental role in determining both the security and implementation efficiency of lattice-based key encapsulation

收稿日期:2025-09-29;在线发布日期:2026-03-12。本课题得到国家自然科学基金(No. U2336205)资助。**邵明尧**,博士研究生,中国计算机学会(CCF)会员,主要研究领域为后量子密码学。E-mail:shaomingyao@iie.ac.cn。**刘月君(通信作者)**,博士,副教授,中国计算机学会(CCF)会员,主要研究领域为密码工程与后量子密码学。E-mail:liuyuejun@njust.edu.cn。**周永彬(通信作者)**,博士,教授,中国计算机学会(CCF)杰出会员,主要研究领域为密码学与密码工程、后量子密码学。E-mail:zhouyongbin@njust.edu.cn。**邵 岩**,博士研究生,主要研究领域为后量子密码学。**张 倩**,博士,助理研究员,中国计算机学会(CCF)会员,主要研究领域为侧信道攻击与防御、密码硬件设计与实现。

mechanisms (KEMs). With the ongoing deployment of post-quantum cryptography, lattice-based schemes such as Kyber and Saber have become leading candidates for practical use. In these schemes, the hardness of the underlying Learning With Errors (LWE) or Module-LWE problems depends heavily on the statistical properties of the random distribution. To improve sampling efficiency and avoid the timing side-channel risks of Gaussian sampling, mainstream lattice-based KEMs typically adopt either the Uniform Distribution (UD) or the Centered Binomial Distribution (CBD). However, a systematic comparison of how these two distributions affect KEM security and performance remains largely unexplored. In this work, we conduct a comprehensive comparative study of CBD and UD using Kyber and Saber as representative examples. We first analyze the statistical characteristics of the two distributions and evaluate their impact on the hardness of the corresponding lattice problems. On the security side, we quantify the trade-off between resistance to lattice reduction attacks and vulnerability to decryption failure attacks. Our analysis shows that under the same value range, the standard deviation of UD is at least $\sqrt{4/3}$ times that of CBD. This larger variance results in harder lattice problem instances and increases the security strength of Kyber and Saber by approximately 12 bits—25 bits. However, it also significantly increases the probability of decryption failure, introducing new security risks. To study the implications of higher failure probabilities, we further analyze decryption failure attacks under UD settings. Since locating failure positions and recovering the secret key becomes challenging when failure probability increases, we propose two techniques to address these issues: an iterative failure-localization method to identify likely failure positions and a majority-vote key recovery method to improve secret coefficient estimation. Using these techniques, the key recovery complexity under UD can be reduced to $O(2^{37})$ for Kyber512 and $O(2^{42})$ for LightSaber, far below the complexities $O(2^{146})$ and $O(2^{128})$ under the original CBD settings. On the implementation side, we evaluate the impact of distribution choice on storage overhead and runtime performance. When modulus and compression parameters remain fixed, the distribution type does not affect key or ciphertext sizes. However, due to rejection sampling, UD increases sampling time by approximately 25%—65%. Overall, the choice of random distribution creates a complex trade-off among lattice attack resistance, decryption failure risk, and implementation efficiency. To better balance these factors, we introduce the Asymmetric LWE problem to allow flexible tuning of distribution parameters and propose uKyber, a UD-based variant of Kyber. uKyber maintains security strength and decryption failure probabilities comparable to those of the original Kyber while reducing the ciphertext size of Kyber512 from 768 bytes to 672 bytes ($\approx$12.5%). Furthermore, by adopting a multi-value rejection sampling strategy, the sampling efficiency of UD can be significantly improved, leading to faster key generation as well as encapsulation and decapsulation operations. These results demonstrate that properly designing noise distributions can provide a practical way to simultaneously optimize security, efficiency, and implementation performance in lattice-based KEMs.

Keywords random distribution; lattice-based KEMs; security strength; decryption failure; rejection sampling

1 引 言

随机分布在密码学中始终扮演着关键角色。在经典的一次一密(One-Time Pad,OTP)中,密钥的均匀随机性是实现信息论安全的根本保障[1]。对于传统公钥密码,以 ElGamal 加密方案为例[2],其安全性不仅依赖于离散对数问题的计算困难性,还要求加密过程中使用的随机数服从均匀分布。若该分

布偏离均匀,攻击者可利用统计方法区分真实密文与理想分布,从而破坏方案的不可区分性[3-4]。因此,随机分布的选取是密码方案安全设计的核心要素。

为应对量子计算发展带来的安全威胁,美国国家标准与技术研究院(National Institute of Standards and Technology, NIST)于 2016 年启动了全球后量子密码(Post-Quantum Cryptography, PQC)标准化进程[5]。其中,格密码凭借可证明安全性与显著的效率优势成为主流选择。在格密钥封装机制(Key Encapsulation Mechanism, KEM)中,随机分布的类型和参数设置从根本上决定了方案的理论设计与安全强度,并显著影响其工程实现与技术应用。

带误差学习(Learning with Errors, LWE)问题在经典与量子计算模型下的求解困难性,构成众多格密码方案安全性的理论基础[4]。早期基于 LWE 问题的密码方案通常依高斯分布(Gaussian Distribution, GD)生成误差向量[6-7],但高斯采样的计算开销大[8]且存在计时侧信道攻击的安全隐患[9-10]。为此,研究者提出更易于实现的替代分布,特别是均匀分布(Uniform Distribution, UD)和中心二项分布(Centered Binomial Distribution, CBD)。Döttling 等人[11]与 Micciancio 等人[12]首先论证了当误差向量采样自 UD 时,LWE 问题依然保持难解性;Cabarcas 等人[13]给出了首个私钥与误差向量均取自小区间 UD 的可证明安全的公钥加密方案;Alkim 等人[14]则在 NewHope 中基于 CBD 采样私钥和误差向量,Kyber[15-16]和 Saber[17]亦延续了这一选择。如表 1 所示,当前主流基于 LWE 问题的格密码多以 CBD 或 UD 作为随机分布选择。

表 1　NIST PQC 格密码候选方案的随机分布类型

方案	功能	困难假设	分布	δ
Kyber	KEM	MLWE	CBD	2^{-174}
NewHope	KEM	RLWE	CBD	2^{-216}
LAC	KEM	RLWE	CBD	2^{-122}
Saber	KEM	MLWR	CBD	2^{-165}
uSaber	KEM	MLWR	UD	2^{-152}
KINDI	KEM	MLWE	UD	2^{-216}
FrodoKEM	KEM	LWE	GD	2^{-174}
Dilithium	SIG	MLWE	UD	—

注:δ 为解密失败概率(该表中 δ 值对应 NIST 安全等级 V 的方案变体的解密失败概率)。

然而,关于随机分布选取对格 KEM 安全性与性能的影响,现有工作仍缺乏系统性的理论分析与对比研究。尽管在 NIST PQC 第三轮评选中,Saber 设计团队提出了基于 UD 的变体方案 uSaber[17],但该工作主要展示了 UD 下的一种可行实现,并未就分布选择对格 KEM 安全性与性能的作用机制开展深入论证。此外,在密码方案的实际部署中,受效率优化、代码复用或开发便利性等因素影响,分布采样过程可能出现函数调用错误或随机性偏差等问题[18-19]。这些潜在的误用风险不仅可能削弱方案安全性,也进一步放大方案对分布选取的敏感性,从而凸显了开展随机分布选择对格密码影响的系统性评估的现实意义。

随机分布对格 KEM 的影响是多层面且系统性的。在安全性方面,需要在抗格攻击能力与解密失败风险之间审慎权衡;在性能方面,则应兼顾算法的存储开销与运行时间。本文以 Kyber 和 Saber 两类典型格 KEM 为研究示例,围绕 CBD 与 UD 开展系统性对比分析。首先,从理论层面对随机分布在格 KEM 安全性与性能中的作用机制进行分析,在保持分布区间不变的前提下,对 CBD 与 UD 的影响差异进行比较;随后,引入非对称 LWE(Asymmetric LWE, ALWE)问题以支持参数的灵活调整,从而在不同分布下实现方案安全性与性能的平衡。本文的主要贡献如下:

(1)本文首先分析随机分布对格 KEM 抗格攻击能力的影响。格 KEM 的安全强度取决于其底层困难问题实例的求解复杂度,当前最有效的求解路径为格攻击方法。其核心目标是在扩展格中寻找形如 $\boldsymbol{v}=(\boldsymbol{s}^T,\boldsymbol{e}^T,1)$ 的短向量,随着私钥 $\boldsymbol{s}$ 与误差 $\boldsymbol{e}$ 系数的分布标准差增大,目标向量 $\boldsymbol{v}$ 的范数随之增加,从而提高格攻击难度。与 CBD 相比,相同区间下的 UD 具有 $\sqrt{4/3}$ 倍以上的标准差,使得对应 LWE 问题实例具备更强的抗格攻击能力。基于 core-SVP 模型的定量评估表明,将 Kyber 和 Saber 的随机分布由原始 CBD 替换为相同区间下的 UD,可使方案的安全强度提升约 12～25 比特。

(2)抗格攻击能力提升的代价是解密失败概率的急剧上升。基于 LWE 的 KEM 解密正确性高度依赖加密噪声项(如 $\boldsymbol{s},\boldsymbol{e}$)的分布,分布标准差增大显著提高解密失败率,加剧解密失败攻击的威胁。分析表明,将 Kyber 和 Saber 的随机分布由 CBD 替换为相同取值区间下的 UD 后,解密失败概率显著上升。尤其对于 Kyber512 和 LightSaber,失败概率分别由 2^{-139} 和 2^{-120} 提高至 $2^{-25.4}$ 和 2^{-30},使得在 NIST PQC 规定的解封装查询上限(2^{64})内收集足量失败密文并实施密钥恢复成为可行路径。为应对高失败率场景中“失败位置难定位、私钥恢复复杂度高”的挑战,本文提出迭代增强的失败定位方法和多

数投票的密钥恢复方法。相较现有最优攻击方法，将密钥恢复所需失败密文数量降低 50%以上。具体而言，针对 UD 下的 Kyber512 和 LightSaber，分别以 $O(2^{37})$和 $O(2^{42})$的计算复杂度恢复完整私钥，远低于原始 CBD 方案的 $O(2^{146})$和 $O(2^{128})$。

(3)格 KEM 的存储开销主要由随机变量的存储形式和编码方式决定。由于 Kyber 和 Saber 的密钥、密文均在模整数环上压缩存储，在模数与压缩参数固定的前提下，分布类型的变更不会影响存储长度，但会显著影响分布采样效率。具体而言，多项式环元素采样需先经哈希生成伪随机字节流，再由采样算法完成映射。尽管 UD 在多数参数下所需的伪随机字节更少，可减少哈希挤压轮数，但其固有的拒绝采样机制会显著增加采样算法的平均调用次数，使得整体运行速度慢于 CBD 采样。在 Kyber 和 Saber 的实际参数下，UD 采样使多项式生成过程的运行时间增加约 25%～65%，进而延长了密钥生成、封装与解封装的运行时间。

(4)综合以上分析，随机分布选取在抗格攻击能力、解密失败风险与实现性能方面对格 KEM 产生显著而不均衡的影响。为平衡上述矛盾，本文引入 ALWE 问题以支持分布参数的灵活调节，据此首次提出基于 UD 的 Kyber 变体 uKyber，并给出严格的安全性证明。在保持与原始 Kyber 可比的安全强度和解密失败概率的同时，uKyber 显著缩短密文长度(如将 Kyber512 的密文长度由 768 字节降至 672 字节，减少约 12.5%)，此外，本文提出的多值拒绝采样方法显著降低 UD 采样的拒绝概率，从而提升采样效率，使得 uKyber 在密钥生成、封装与解封装各阶段的运行时间均优于原始 Kyber。

2 预备知识

2.1 符 号

记 $\mathbb{Z}_q=\mathbb{Z}/q\mathbb{Z}$ 为模 q 的整数环，多项式环 $R=\mathbb{Z}[x]/(x^n+1)$，$R_q=\mathbb{Z}_q[x]/(x^n+1)$。小写字母表示 R 或 R_q 中的元素，粗体小写字母表示由环元素组成的向量，粗体大写字母表示 $R_q^{k\times l}$ 中的多项式矩阵，矩阵 $\boldsymbol{A}$ 的转置记作 $\boldsymbol{A}^{\mathrm{T}}$。对多项式向量 $\boldsymbol{c}\in R_q^k$，记 $\bar{\boldsymbol{c}}\in\mathbb{Z}_q^{kn}$ 为其每个多项式的系数拼接得到的向量。设 $w(x)=w_0+w_1x+\cdots+w_{n-1}x^{n-1}\in R$，其无穷范数与 2 范数分别为：$\|w\|_\infty=\max_i|w_i|$，$\|w\|_2=\sqrt{|w_0|^2+\cdots+|w_{n-1}|^2}$。设 $f(x)=\sum_{i=0}^{n-1}a_ix^i$，$g(x)=\sum_{j=0}^{n-1}b_jx^j\in R_q$，则乘积 $h(x)=f(x)\cdot g(x)=\sum_{k=0}^{n-1}c_kx^k\in R_q$，其中 $c_k=\sum_{i+j\equiv k \bmod n}a_ib_j$。

设 $q\in\mathbb{Z}^+$，定义 $r'=r \bmod\pm q$，满足 $-q/2<r'\leqslant q/2$ 且 $r'\equiv r \bmod q$；$r'=r \bmod+q$，满足 $0\leqslant r'<q$ 且 $r'\equiv r \bmod q$。$\lceil x\rceil$、$\lfloor x\rfloor$、$\lceil x\rfloor$分别表示向上取整、向下取整和舍入取整。对整数 $q\geqslant p\geqslant 2$，舍入函数 $\lceil\cdot\rfloor_p:\mathbb{Z}_q\to\mathbb{Z}_p$ 定义为 $\lceil x\rfloor_p=\left\lceil\frac{p}{q}\right\rfloor x \bmod p$。对集合 S，记 $s\leftarrow S$ 表示从 S 中均匀随机采样。若 S 表示概率分布，则表示依分布 S 采样。可扩展输出函数 $y\sim S:\mathrm{Sam}(x)$表示输入 x，依分布 S 生成任意长度 y。

2.2 定 义

定义 1.(中心二项分布，CBD)给定 $\eta\in\mathbb{Z}^+$，中心二项分布 β_η 定义为 $\beta_\eta:=\sum_{i=1}^{\eta}(a_i-b_i)$，其中 $(a_1,\cdots,a_\eta,b_1,\cdots,b_\eta)\leftarrow\{0,1\}^{2\eta}$，标准差为 $\sigma=\sqrt{\eta/2}$。

定义 2.(均匀分布，UD)给定 $u\in\mathbb{Z}^+$，均匀分布 $\mathcal{U}_u$ 定义为区间 $[-u,u]\cap\mathbb{Z}$ 上的离散均匀分布，标准差为 $\sigma=\sqrt{((2u+1)^2-1)/12}$。

定义 3.(LWE 问题)给定正整数 m,n,q，私钥 $\boldsymbol{s}\leftarrow\mathbb{Z}_q^n$，误差向量 $\boldsymbol{e}\leftarrow\chi^m$($\chi$ 为公开分布)，从 $\mathbb{Z}_q^{m\times n}$ 均匀随机选取矩阵 $\boldsymbol{A}$，定义 LWE 分布 $(\boldsymbol{A},\boldsymbol{b})\in\mathbb{Z}_q^{m\times n}\times\mathbb{Z}_q^m$，其中 $\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e} \bmod q$。判定 LWE 问题要求区分 $(\boldsymbol{A},\boldsymbol{b})$ 与在 $\mathbb{Z}_q^{m\times n}\times\mathbb{Z}_q^m$ 上的均匀分布。搜索 LWE 问题要求给定 $(\boldsymbol{A},\boldsymbol{b})$ 恢复 $\boldsymbol{s}$。LWE 的环版本和模版本分别称为环 LWE(Ring LWE，RLWE)和模 LWE(Module LWE，MLWE)。

定义 4.(LWR 问题)给定正整数 m,n,p,q，私钥 $\boldsymbol{s}\leftarrow\mathbb{Z}_q^n$，从 $\mathbb{Z}_q^{m\times n}$ 均匀随机选取矩阵 $\boldsymbol{A}$，定义 LWR 分布 $(\boldsymbol{A},\boldsymbol{b})\in\mathbb{Z}_q^{m\times n}\times\mathbb{Z}_q^m$，其中 $\boldsymbol{b}=\lceil\boldsymbol{A}\boldsymbol{s}\rfloor_p$。判定 LWR 问题要求区分 $(\boldsymbol{A},\boldsymbol{b})$ 与在 $\mathbb{Z}_q^{m\times n}\times\mathbb{Z}_q^m$ 上的均匀分布。搜索 LWR 问题要求给定 $(\boldsymbol{A},\boldsymbol{b})$ 恢复 $\boldsymbol{s}$。LWR 的环版本和模版本分别称为环 LWR(Ring LWR，RLWR)和模 LWR(Module LWR，MLWR)。此外，LWR 可表示为 $(\boldsymbol{A},\boldsymbol{b})=\frac{p}{q}(\boldsymbol{A}\boldsymbol{s} \bmod q)+\boldsymbol{e} \bmod p$，转化为模 q 下的 LWE 问题 $\left(\boldsymbol{A},\frac{q}{p}\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\frac{q}{p}\boldsymbol{e} \bmod q\right)$，其中 $\boldsymbol{e}$ 为舍入操作带来的误差。

2.3 Kyber

Kyber[15]基于 MLWE 问题，已被 NIST 标准化为后量子密钥封装机制标准 ML-KEM[16]。Kyber 的私钥和误差向量系数均采样自 CBD，具体参数见

表 2。为便于后续分析,本文给出 Kyber 的密钥生成、加密与解密算法,对应算法 1、算法 2 与算法 3。

表 2　Kyber 方案推荐参数

参数	Kyber512	Kyber768	Kyber1024
n	256	256	256
k	2	3	4
q	3329	3329	3329
(η_1,η_2)	(3,2)	(2,2)	(2,2)
(d_u,d_v)	(10,4)	(10,4)	(11,5)
安全等级	1 (AES-128)	3 (AES-192)	5 (AES-256)

算法 1. Kyber 密钥生成算法

输入:随机种子$(\rho,\zeta)\leftarrow\{0,1\}^{256}$

输出:公钥 pk,私钥 sk

1. $\boldsymbol{A}\sim R_q^{k\times k}:=\mathrm{Sam}(\rho)$
2. $(\boldsymbol{s},\boldsymbol{e})\sim\beta_{\eta_1}^k\times\beta_{\eta_1}^k:=\mathrm{Sam}(\zeta)$
3. $\boldsymbol{t}:=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e}$
4. 返回 $(pk:=(\boldsymbol{t},\rho),sk:=\boldsymbol{s})$

算法 2. Kyber 加密算法

输入:公钥 $pk=(\boldsymbol{t},\rho)$,消息 m,随机数 φ

输出:密文 $\boldsymbol{c}$

1. $\boldsymbol{A}\sim R_q^{k\times k}:=\mathrm{Sam}(\rho)$
2. $(\boldsymbol{r},\boldsymbol{e}_1,e_2)\sim\beta_{\eta_1}^k\times\beta_{\eta_2}^k\times\beta_{\eta_2}:=\mathrm{Sam}(\varphi)$
3. $\boldsymbol{u}:=\boldsymbol{A}^T\boldsymbol{r}+\boldsymbol{e}_1$
4. $v:=\boldsymbol{t}^T\boldsymbol{r}+e_2+\lceil q/2\rceil\cdot m$
5. $\boldsymbol{c}_1:=\mathrm{Compress}_q(\boldsymbol{u},d_u)$
6. $c_2:=\mathrm{Compress}_q(v,d_v)$
7. 返回 $\boldsymbol{c}:=(\boldsymbol{c}_1,c_2)$

算法 3. Kyber 解密算法

输入:私钥 $sk=\boldsymbol{s}$,密文 $\boldsymbol{c}=(\boldsymbol{c}_1,c_2)$

输出:解密结果 m'

1. $\boldsymbol{u}':=\mathrm{Decompress}_q(\boldsymbol{c}_1,d_u)$
2. $v':=\mathrm{Decompress}_q(c_2,d_v)$
3. 返回 $m':=\mathrm{Compress}_q(v'-\boldsymbol{s}^T\boldsymbol{u}',1)$

定义 5.(压缩与解压缩函数)Kyber 中定义压缩函数 $\mathrm{Compress}_q(x,d)=\lceil(2^d/q)\cdot x\rfloor\bmod^+ 2^d$,输入 $x\in\mathbb{Z}_q$,输出$\{0,1,\cdots,2^d-1\}$中的整数,其中 $d<\lceil\log_2 q\rceil$;定义解压缩函数 $\mathrm{Decompress}_q=\lceil(q/2^d)\cdot x\rfloor$,使得 $x'=\mathrm{Decompress}_q(\mathrm{Compress}_q(x,d),d)$满足$|x'-x\bmod^+ q|\leqslant\lceil q/(2^{d+1})\rfloor$。

当 $\mathrm{Compress}_q$ 或 $\mathrm{Decompress}_q$ 作用于 $x\in R_q$ 或 $x\in R_q^k$ 时,上述过程独立地作用于每个系数。

2.4　Saber

Saber[17] 基于 MLWR 问题,为 NIST PQC 第三轮候选 KEM 算法。Saber 的私钥向量采样自 CBD,误差向量来自舍入操作,具体参数见表 3。为便于后续分析,本文给出 Saber 的密钥生成、加密与解密算法,对应算法 4、算法 5 与算法 6。

表 3　Saber 方案推荐参数

参数	LightSaber	Saber	FireSaber
n	256	256	256
k	2	3	4
(q,ϵ_q)	$(2^{13},13)$	$(2^{13},13)$	$(2^{13},13)$
(p,ϵ_p)	$(2^{10},10)$	$(2^{10},10)$	$(2^{10},10)$
(T,ϵ_T)	$(2^3,3)$	$(2^4,4)$	$(2^6,6)$
η	5	4	3
安全等级	1 (AES-128)	3 (AES-192)	5 (AES-256)

算法 4. Saber 密钥生成算法

输入:随机种子$(\rho,\zeta)\leftarrow\{0,1\}^{256}$

输出:公钥 pk,私钥 sk

1. $\boldsymbol{A}\sim R_q^{k\times k}:=\mathrm{Sam}(\rho)$
2. $\boldsymbol{s}\sim\beta_\eta^k:=\mathrm{Sam}(\zeta)$
3. $\boldsymbol{b}:=((\boldsymbol{A}^T\boldsymbol{s}+\boldsymbol{h})\bmod q)\gg(\epsilon_q-\epsilon_p)\in R_p^{k\times 1}$
4. 返回 $(pk:=(\rho,\boldsymbol{b}),sk:=\boldsymbol{s})$

算法 5. Saber 加密算法

输入:公钥 $pk=(\rho,\boldsymbol{b})$,消息 m,随机数 φ

输出:密文 $\boldsymbol{c}$

1. $\boldsymbol{A}\sim R_q^{k\times k}:=\mathrm{Sam}(\rho)$
2. $\boldsymbol{s}'\sim\beta_\eta^k:=\mathrm{Sam}(\varphi)$
3. $\boldsymbol{b}':=((\boldsymbol{A}\boldsymbol{s}'+\boldsymbol{h})\bmod q)\gg(\epsilon_q-\epsilon_p)\in R_p^{k\times 1}$
4. $v:=\boldsymbol{b}^T\boldsymbol{s}'$
5. $c_m:=(v+h_1-2^{\epsilon_p-1}m\bmod p)\gg(\epsilon_q-\epsilon_T)\in R_T$
6. 返回 $\boldsymbol{c}=(c_m,\boldsymbol{b}')$

算法 6. Saber 解密算法

输入:私钥 $sk=\boldsymbol{s}$,密文 $\boldsymbol{c}=(c_m,\boldsymbol{b}')$

输出:解密消息 m'

1. $v':=\boldsymbol{b}'^T(\boldsymbol{s}\bmod p)\in R_p$
2. $m':=((v'-2^{\epsilon_p-\epsilon_T}c_m+h_2)\bmod p)\gg(\epsilon_p-1)$
3. 返回 m'

为提高舍入算法的效率,Saber 通过引入常数多项式 $\boldsymbol{h}$,h_1,h_2 并使用位移代替舍入运算。常数多项式 h_1 的所有系数均设为 $2^{\epsilon_q-\epsilon_p-1}$,多项式向量 $\boldsymbol{h}$ 的每个多项式分量都等于 h_1,常数多项式 h_2 的所有系数均设为 $2^{\epsilon_p-2}-2^{\epsilon_p-\epsilon_T-1}+2^{\epsilon_q-\epsilon_p-1}$。

2.5　解密失败

基于 LWE/LWR 的 KEM 方案中,解密过程需从带误差的解密结果中恢复正确消息;当加解密过程中引入的总误差超过容错阈值时即发生解密失败,从而影响方案的正确性与安全性[20-23]。

以 Kyber 为例，解密失败概率受多个变量影响，包括私钥向量 $\boldsymbol{s},\boldsymbol{r}$，误差向量 $\boldsymbol{e},\boldsymbol{e}_1,e_2$，以及压缩/解压缩引入的误差向量 $\boldsymbol{\varepsilon}_u$ 和 ε_v。具体记 $\boldsymbol{\varepsilon}_u=\boldsymbol{u}'-\boldsymbol{u}=\mathrm{Decompress}_q(\mathrm{Compress}_q(\boldsymbol{u},d_u),d_u)-\boldsymbol{u}$，$\varepsilon_v=v'-v=\mathrm{Decompress}_q(\mathrm{Compress}_q(v,d_v),d_v)-v$。

根据算法 3，Kyber 的解密过程计算如下：

$$\begin{aligned}m'&=\mathrm{Compress}_q(v'-\boldsymbol{s}^{\mathrm{T}}\boldsymbol{u}',1)\\&=\lceil 2/q\,(v'-\boldsymbol{s}^{\mathrm{T}}\boldsymbol{u}')\rfloor \bmod 2\\&=m+\left\lceil\frac{2}{q}(\boldsymbol{e}^{\mathrm{T}}\boldsymbol{r}-\boldsymbol{s}^{\mathrm{T}}(\boldsymbol{e}_1+\boldsymbol{\varepsilon}_u)+e_2+\varepsilon_v)\right\rfloor \bmod 2\end{aligned}\tag{1}$$

令 $\boldsymbol{S}=[-\boldsymbol{s},\boldsymbol{e}]$，$\boldsymbol{C}=[\boldsymbol{e}_1+\boldsymbol{\varepsilon}_u,\boldsymbol{r}]$，$G=e_2+\varepsilon_v$，则发生解密失败的充要条件为 $\|\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G\|_\infty=\|\boldsymbol{e}^{\mathrm{T}}\boldsymbol{r}-\boldsymbol{s}^{\mathrm{T}}(\boldsymbol{e}_1+\boldsymbol{\varepsilon}_u)+e_2+\varepsilon_v\|_\infty>q/4$。

定义 6.（多项式旋转）向量 $\boldsymbol{C}\in R_q^k$ 的第 r 次旋转 $\boldsymbol{C}^r$ 定义为：$\boldsymbol{C}^r(x):=x^r\cdot\boldsymbol{C}(x^{-1})\bmod(x^n+1)$。记 $\overline{\boldsymbol{C}^r}\in\mathbb{Z}_q^{kn}$ 为其系数拼接向量。对于 $\boldsymbol{S},\boldsymbol{C}\in R_q^k$，$\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}$ 可分解为标量积 $\overline{\boldsymbol{S}}^{\mathrm{T}}\,\overline{\boldsymbol{C}^r}$ 的和：$\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}:=\sum_{r=0}^{n-1}\overline{\boldsymbol{S}}^{\mathrm{T}}\,\overline{\boldsymbol{C}^r}\cdot x^r$。上述展开式在确定解密失败的具体位置中具有重要作用，更详细的计算细节见附录 A。

3 随机分布对格 KEM 抗格攻击能力的影响分析

首先，本节从格 KEM 抗格攻击能力的角度，分析随机分布对格 KEM 安全性的影响。格 KEM 的安全强度由其底层困难问题实例的求解复杂度决定，而格攻击方法是目前最有效的求解方法。现有最优格攻击方法尚未有效利用模格结构，因此本文将 Kyber 和 Saber 的底层 MLWE/MLWR 问题视作一般格上的 LWE/LWR 问题进行分析，再进一步将 LWR 问题转化为对应的 LWE 问题求解。

3.1 基于 core-SVP 模型的安全强度评估

目前已有多种 LWE 问题的求解策略[24]，包括组合方法（如 BKW 算法）、代数方法（如 Arora-Ge 算法）、穷搜方法以及格攻击方法。在格 KEM 方案的参数规模与样本数量限制下，格攻击方法被认为是目前最有效的求解手段，主要包括原始攻击（Primal attack）和对偶攻击（Dual attack）[25]。两者均将 LWE 问题归约为最短向量问题（Shortest Vector Problem，SVP）问题，再使用格约简算法（如 BKZ）进行求解。

BKZ 算法通过在低维区块中调用 SVP 谕言机实现约简，谕言机调用次数为多项式级别[26]。在安全强度评估中，通常采用忽略多项式因子的 core-SVP 模型，即专注于单个 b 维区块上求解 SVP 的复杂度。借助局部敏感哈希（Locality-Sensitive Hashing，LSH）技术[27-28]，经典计算条件下 core-SVP 求解复杂度约 $2^{0.292b+o(b)}$。结合 Grover 搜索算法，在量子计算条件下，复杂度进一步降至 $2^{0.265b+o(b)}$[29-30]。

3.2 原始攻击

原始攻击将搜索 LWE 问题视为有界距离解码（Bounded Distance Decoding，BDD）问题，并通过 Kannan 嵌入将其转化为唯一最短向量问题（unique-SVP，uSVP），继而利用格约简算法求解。

3.2.1 攻击思路

给定 LWE 问题实例 $(\boldsymbol{A},\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e})\in\mathbb{Z}_q^{m\times n}\times\mathbb{Z}_q^m$，先将其转化为非齐次小整数解问题 $\boldsymbol{b}=(\boldsymbol{A}\mid\boldsymbol{I}_m)(\boldsymbol{s}^{\boldsymbol{T}}\mid\boldsymbol{e}^{\boldsymbol{T}})^{\boldsymbol{T}}\bmod q$。然后构造嵌入格 $\Lambda=\{x\in Z^{n+m+1}:(\boldsymbol{A}\boldsymbol{I}_m-\boldsymbol{b})x=0\bmod q\}$。格 Λ 的基由如下矩阵的列向量给出：

$$\boldsymbol{B}=\begin{pmatrix}\boldsymbol{I}_n & \boldsymbol{0} & \boldsymbol{0}\\ -\boldsymbol{A} & q\boldsymbol{I}_m & \boldsymbol{b}\\ \boldsymbol{0} & \boldsymbol{0} & 1\end{pmatrix}\in Z^{(n+m+1)\times(n+m+1)}\tag{2}$$

因此嵌入格 Λ 的维数为 $d=n+m+1$，体积为 $\det(\Lambda)=q^m$，且包含短向量 $\boldsymbol{v}=(\boldsymbol{s}^{\boldsymbol{T}},\boldsymbol{e}^{\boldsymbol{T}},1)$。为估计该向量范数，引入如下引理。

引理 1. 设 $\boldsymbol{v}=(v_1,v_2,\cdots,v_l)$ 由均值为 0、标准差为 σ 的独立同分布随机变量构成的向量，则其 2 范数近似于 $\|\boldsymbol{v}\|_2\approx\sigma\sqrt{l}$[15]。

引理 2. 设 $\boldsymbol{v}=(s_1,s_2,\cdots,s_n,e_1,e_2,\cdots,e_m)$，其中 s_i 和 e_j 分别服从均值为 0、标准差为 σ_s 和 σ_e 的分布，则 $\boldsymbol{v}$ 的 2 范数近似于 $\|\boldsymbol{v}\|_2\approx\sqrt{n\sigma_s^2+m\sigma_e^2}$，$\boldsymbol{v}$ 的标准差近似为 $\sigma\approx\sqrt{(n\sigma_s^2+m\sigma_e^2)/(n+m)}$[16]。

由此可知，目标向量 $\boldsymbol{v}=(\boldsymbol{s}^{\mathrm{T}},\boldsymbol{e}^{\mathrm{T}},1)$ 的范数近似为 $\sqrt{n\sigma_s^2+m\sigma_e^2+1}$。在 Kyber 和 Saber 的实际参数下，该范数显著小于高斯启发式[23]对格中最短向量的预测值，故目标向量 $\boldsymbol{v}$ 极有可能是格 Λ 中的最短非零向量。

3.2.2 安全强度评估

假设攻击者使用 BKZ 约简算法寻找格 Λ 中的最短非零向量，设 Λ 的维度 d，BKZ 的分组长度为 b，根据几何级数假设（Geometric Series Assumption，GSA）[31]，BKZ-b 输出的正交基满足 $\|\boldsymbol{b}_1^*\|_2=\delta_0{}^d\det(\Lambda)^{1/d}$，$\|\boldsymbol{b}_i^*\|_2=\delta_0{}^{-2d(i-1)/(d-1)}\|\boldsymbol{b}_1^*\|_2$，其中

Hermite 因子 δ_0 近似为 $\delta_0 \approx ((\pi b)^{1/b} \cdot b/2\pi e)^{1/(2(b-1))}$[23,32]。原始攻击中,通常关注最后 b 个正交向量张成的子空间[26]。令 $i=d-b+1$,则第 $d-b+1$ 个正交向量的长度为 $\|\boldsymbol{b}_{d-b+1}^*\|_2 = \delta_0^{-2d(d-b)/(d-1)}\|\boldsymbol{b}_1^*\|_2 = \delta_0^{(-d^2+2db-d)/(d-1)}\det(\Lambda)^{1/d}$。

根据引理 2,目标向量 $\boldsymbol{v}=(\boldsymbol{s}^{\mathrm{T}},\boldsymbol{e}^{\mathrm{T}},1)$ 的 2 范数近似为 $\ell \approx \sqrt{n\sigma_s^2+m\sigma_e^2+1}$。在 GSA 下,BKZ 输出基可视为各向同性,即格中短向量在各个正交方向上的能量近似均匀分布,因此目标向量 $\boldsymbol{v}$ 在 BKZ 最后 b 个正交向量张成的子空间的投影长度近似为 $\|\pi_b(\boldsymbol{v})\|_2 \approx \ell\sqrt{b/d}$。在格 Λ 中,若目标向量 $\boldsymbol{v}$ 在末 b 维子空间上的投影长度不超过该子空间中最短基向量长度,则 BKZ-b 在枚举该区块时有很大概率输出该向量,即 $\|\pi_b(\boldsymbol{v})\|_2 \leqslant \|\boldsymbol{b}_{d-b+1}^*\|_2$。代入上式得

$$\ell\sqrt{b/d} \leqslant \delta_0^{(-d_2+2db-d)/(d-1)}\det(\Lambda)^{1/d} \tag{3}$$

原始攻击的计算复杂度由满足该成功条件的最小区块维度 b 决定。随着私钥 $\boldsymbol{s}$ 与误差向量 $\boldsymbol{e}$ 系数的标准差 σ_s 和 σ_e 增加,目标向量 $\boldsymbol{v}$ 的范数 ℓ 随之增大,从而需要更大的 b 才能满足该成功条件,因此提高了 LWE 问题实例在原始攻击下的求解难度。

3.3　对偶攻击

对偶攻击将判定 LWE 问题转化为小整数解(Small Integer Solution, SIS)问题,构造对偶格并寻找短向量,用以区分 LWE 样本与均匀随机样本。

3.3.1　攻击思路

给定 LWE 实例 $(\boldsymbol{A},\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e}) \in \mathbb{Z}_q^{m\times n}\times\mathbb{Z}_q^m$,记 $\boldsymbol{A}^{\mathrm{T}}=(\boldsymbol{A}_1|\boldsymbol{A}_2)\in\mathbb{Z}_q^{n\times m}$,其中 $\boldsymbol{A}_1\in\mathbb{Z}_q^{n\times n}$ 为 $\boldsymbol{A}^{\mathrm{T}}$ 前 n 列组成的方阵。假设 $\boldsymbol{A}_1$ 模 q 下可逆,定义 $\boldsymbol{A}'=(\boldsymbol{I}_n|\boldsymbol{A}_1^{-1}\boldsymbol{A}_2)$。则对偶格 $\Lambda=\{x\in\mathbb{Z}^m|\boldsymbol{A}^{\mathrm{T}}x=0 \bmod q\}$可表示为 $\Lambda=\{x\in\mathbb{Z}^m|\boldsymbol{A}'x=0 \bmod q\}$。格 Λ 的基由如下矩阵的列向量给出:

$$\boldsymbol{B}=\begin{pmatrix} q\boldsymbol{I}_n & -\boldsymbol{A}_1^{-1}\boldsymbol{A}_2 \\ 0 & \boldsymbol{I}_{m-n}\end{pmatrix}\in\mathbb{Z}^{m\times m} \tag{4}$$

攻击者在格 Λ 中使用格约简算法寻找短向量 $\boldsymbol{v}$,并计算内积值 $u=\langle\boldsymbol{v},\boldsymbol{b}\rangle$。若 $\boldsymbol{b}$ 为 LWE 样本,则 $u=\boldsymbol{v}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e})=\langle\boldsymbol{v},\boldsymbol{e}\rangle$。由于误差向量 $\boldsymbol{e}$ 的标准差为 σ_e,$\boldsymbol{v}$ 的范数为 ℓ,则内积 u 的分布近似为标准差为 $\ell\sigma_e$ 的亚高斯分布。反之,若 $\boldsymbol{b}$ 为均匀随机向量,则 $\langle\boldsymbol{v},\boldsymbol{b}\rangle$ 近似服从 $\mathbb{Z}_q$ 上的均匀分布。

3.3.2　安全强度评估

区分标准差为 $\ell\sigma_e$ 的亚高斯分布与均匀分布的区分优势为 $\epsilon=4\exp(-2\pi^2\tau^2)$,其中 $\tau=\ell\sigma_e/q$。为实现大于 1/2 的成功率,需要至少 $1/\epsilon^2$ 个短向量[25]。

由于 BKZ-b 一次生成约 $2^{0.2075b}$ 个短向量,因此需重复 SIS 求解过程 $T=\max(1,1/(2^{0.2075b}\epsilon^2))$ 次。于是对偶攻击的总体复杂度估计为 $T\cdot\mathrm{cost}_b = \max(1,1/(2^{0.2075b}\cdot 16\exp(-4\pi^2\tau^2)))\cdot\mathrm{cost}_b$。其中,$\mathrm{cost}_b$ 表示在区块维度为 b 的格上使用经典或量子算法求解 SVP 的计算复杂度。

综上,随着误差向量的标准差 σ_e 增大,$\tau=\ell\sigma_e/q$ 增大,区分优势 ϵ 减小。为维持同等区分优势,攻击者需采用更大的 BKZ 区块维度 b,以获得更小范数 ℓ 的格向量 $\boldsymbol{v}$。因此,标准差 σ_e 的增大提高了 LWE 问题实例在对偶攻击下的求解难度。

3.4　实验验证

基于前述理论分析,本文针对 Kyber 和 Saber,在相同取值区间下对 CBD 与 UD 对比分析,评估其底层 MLWE/MLWR 实例在格攻击下的安全强度。为量化两类分布引入的标准差差异,给出如下定理:

定理 1. 中心二项分布 β_η 的标准差为 $\sigma_{CBD}=\sqrt{\eta/2}$,均匀分布 $\mathcal{U}_u$ 的标准差为 $\sigma_{UD}=\sqrt{((2u+1)^2-1)/12}$。当 $u=\eta$ 时,有 $\sigma_{UD}/\sigma_{CBD}=\sqrt{((2u+1)^2-1)/(6u)}=\sqrt{(2u+2)/3}>\sqrt{4/3}$。因此,在等取值区间下,直接用 UD 替代 CBD 使得标准差至少增加$\sqrt{4/3}$倍。

由定理 1 可知,UD 引入更大的标准差将直接导致原始攻击中目标向量 $\boldsymbol{v}=(\boldsymbol{s}^{\mathrm{T}},\boldsymbol{e}^{\mathrm{T}},1)$ 的范数增加,同时降低对偶攻击中的区分优势,迫使 BKZ 算法使用更大的区块大小 b 求解相应的 SVP,从而提升格攻击的整体复杂度。

本文基于 Léo Ducas 等人[15]提出的安全强度评估脚本①,系统评估了 Kyber 和 Saber 在 CBD 和 UD 下的安全强度。具体地,在 MLWE 参数设置中分别选择分布参数为“binomial”或“uniform“。实验结果如表 4 所示,对于 Kyber 和 Saber,当把随机分布从原始 CBD 替换为相同区间的 UD 时,原始攻击与对偶攻击所需的最小 BKZ 区块大小提升约 46 比特~86 比特;在 core-SVP 评估模型下,方案的安全强度提升约 12 比特~25 比特。

标准差的增加虽然提升了格 KEM 抗格攻击的能力,但同时引入新的安全隐患。随机分布标准差的增加将不可避免地提高解密失败概率,进而加剧解密失败攻击的风险。下节将围绕随机分布对格 KEM 解密失败攻击的影响展开系统分析。

① https://github.com/pq-crystals/security-estimates。

表 4 Kyber 和 Saber 在相同取值区间的 CBD 和 UD 下，原始攻击与对偶攻击的经典安全强度与量子安全强度评估

	Kyber512		Kyber768		Kyber1024		LightSaber		Saber		FireSaber	
私钥分布区间	[−3,3]		[−2,2]		[−2,2]		[−5,5]		[−4,4]		[−3,3]	
私钥分布类型	CBD	UD	CBD	UD	CBD	UD	CBD	UD	CBD	UD	CBD	UD
私钥分布标准差 σ	1.225	2.0	1.0	1.414	1.0	1.414	1.581	3.175	1.414	2.598	1.225	2.02
core-SVP 模型，原始攻击												
BKZ-区块大小	406	474	624	688	874	960	405	451	649	706	891	964
经典安全强度($\log_2$)	118	138	182	200	255	280	118	131	189	206	260	281
量子安全强度($\log_2$)	107	125	165	182	231	254	107	119	172	187	236	255
core-SVP 模型，对偶攻击												
BKZ-区块大小	408	476	625	688	879	952	411	454	659	715	902	962
经典安全强度($\log_2$)	119	139	182	200	256	278	120	132	192	208	263	281
量子安全强度($\log_2$)	108	126	165	182	233	252	109	120	174	189	239	255

4 随机分布对格 KEM 解密失败攻击的影响分析

本节从解密失败攻击的角度，分析随机分布对格 KEM 安全性的影响。由于误差向量的存在，基于 LWE 的格 KEM 通常不具备完美正确性，即存在合法密文在对应私钥下无法正确解密。这不仅导致错误的解密结果，还可能泄漏私钥的相关信息。攻击者可利用密码系统对解密失败的响应发起针对性的密钥恢复攻击。解密失败攻击通常包括收集解密失败密文、确定解密失败符号和位置、利用解密失败信息恢复私钥三个阶段。

4.1 收集解密失败密文

首先评估当将 Kyber 和 Saber 的随机分布从 CBD 替换为相同区间的 UD 后，解密失败概率的变化。由 2.5 节可知，解密失败发生当且仅当 $\|\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G\|_\infty=\|\boldsymbol{e}^{\mathrm{T}}\boldsymbol{r}-\boldsymbol{s}^{\mathrm{T}}(\boldsymbol{e}_1+\boldsymbol{\varepsilon}_u)+e_2+\varepsilon_v\|_\infty>q/4$。为估算解密失败发生的概率，本文采用离散卷积技术，结合各变量 $\boldsymbol{s},\boldsymbol{r},\boldsymbol{e},\boldsymbol{e}_1,e_2,\boldsymbol{\varepsilon}_u,\varepsilon_v$ 分布，先计算单个系数触发解密失败的概率，记为 δ'。

假设 1. 对于未使用纠错码的格 KEM 方案，假设解密结果 m' 各系数之间的解密失败事件是相互独立的[33]。

对于不使用纠错码的格 KEM 方案（如 Kyber、Saber），单系数的解密失败不会扩散到其他系数，并且由于秘密向量 $\boldsymbol{S}$ 及密文向量 $\boldsymbol{C}$ 各系数间均为独立采样，因此各系数解密失败事件之间无显著依赖关系[33]。根据独立性假设 1，长度为 n 的消息中至少有一个系数触发解密失败的概率近似为 $\delta=1-(1-\delta')^n\approx n\delta'$。本文使用 python 实现上述解密失败概率评估算法，统计 Kyber 和 Saber 在 CBD 和 UD 下的解密失败概率 δ。实验结果见表 5，当把随机分布由 CBD 替换为相同区间的 UD 后，Kyber 和 Saber 的解密失败概率显著提升。尤其对 Kyber512（和 LightSaber），解密失败概率由 $2^{-139.1}$（$2^{-120.3}$）上升至 $2^{-25.4}$（$2^{-30.1}$），大幅提升了解密失败样本的可获得性。NIST PQC 单目标安全模型允许的解封装查询上限通常为 2^{64} 次[23]，UD 下的高失败概率（$\gg 2^{-64}$）使攻击者在允许的查询次数限制内可收集到足够多的失败密文，以开展后续分析及密钥恢复。

为验证针对采用 UD 的 Kyber512 和 LightSaber 实际解密失败攻击的可行性，本文在配备 2.7GHz Intel Xeon CPU、1TB 内存的 56 核 Linux 服务器上，在官方 C 参考实现基础上实现并部署了采用 UD 的 Kyber512 和 LightSaber。通过随机生成密文并请求解密的方式，两个月内成功收集到约 200000 条 UD 下 Kyber512 的解密失败密文和 10000 条 UD 下 LightSaber 的解密失败密文，为后续分析提供了充足数据支持。

表 5 Kyber 和 Saber 在相同取值区间的 CBD 和 UD 下的解密失败攻击

	Kyber512		Kyber768		Kyber1024		LightSaber		Saber		FireSaber	
私钥分布类型	CBD	UD	CBD	UD	CBD	UD	CBD	UD	CBD	UD	CBD	UD
私钥分布标准差 σ	1.225	2.0	1.0	1.414	1.0	1.414	1.581	3.175	1.414	2.598	1.225	2.02
解密失败概率 δ	$2^{-139.1}$	$2^{-25.4}$	$2^{-165.2}$	$2^{-50.3}$	$2^{-175.2}$	$2^{-47.5}$	$2^{-120.3}$	$2^{-30.1}$	$2^{-136.1}$	$2^{-40.1}$	$2^{-165.2}$	2^{-61}
攻击密文数量(SOTA)	300	13000	220	8500	480	18600	420	16500	350	14800	710	26300
攻击密文数量(Our)	120	3000	100	2600	210	5000	180	4500	150	4200	320	8700
攻击复杂度	$O(2^{146})$	$O(2^{37})$	$O(2^{172})$	$O(2^{62})$	$O(2^{183})$	$O(2^{60})$	$O(2^{128})$	$O(2^{42})$	$O(2^{143})$	$O(2^{52})$	$O(2^{175})$	$O(2^{74})$

4.2　确定解密失败的符号和位置

在成功收集到充足的解密失败密文后,需进一步确定触发解密失败的符号与具体系数位置。

4.2.1　确定解密失败符号

记密文长度为 n,则解密过程中仅有一个系数触发解密失败的概率为 $\delta''=n\delta'(1-\delta')^{n-1}$。当 $\delta'\ll 1/n$,有 $\delta''\approx n\delta'\approx\delta$,表明解密失败事件大概率仅由单个系数触发。设触发该解密失败的系数索引为 r,即 $|(\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G)[r]|>q/4$。若 $(\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G)[r]$ 值为正,记为正失败;否则,记为负失败。

本文利用误差项 $G[r]$ 与 $(\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G)[r]$ 的符号高度一致性确定解密失败的符号。对于 Kyber512 和 LightSaber,根据压缩和舍入函数,$G[r]$ 的分布近似 $\mathcal{U}_{104}$ 和 $\mathcal{U}_{512}$,对解密失败事件的发生影响显著。实验统计表明,对于 UD 下的 Kyber512 和 LightSaber,$\Pr[\mathrm{sign}(G[r])=\mathrm{sign}((\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G)[r])]>99.8\%$。因此,可直接根据 $G[r]$ 的符号高置信度确定失败符号。

4.2.2　确定解密失败位置

当解密失败发生时,直接定位触发失败的具体系数并不容易。D'Anvers 等人[21-22]提出了基于失败向量间相关性的方法确定解密失败位置,但当失败概率较高时,该方法的准确率会明显下降。例如,对于 UD 下 Kyber512,解密失败概率高达 $\delta\approx 2^{-25.4}$,该定位方法的准确率仅为 0.04。

但如图 1 所示,本文观察到即使在高解密失败概率下,解密失败向量与私钥之间仍保留较强相关性。基于此,本文提出一种迭代增强的失败位置确定方法(算法 7)。其主要思想为:首先猜测少量失败位置,利用其初始化私钥方向估计,随后在更多失败密文上贪心扩充并迭代校正该方向估计,逐步确定剩余解密失败位置。

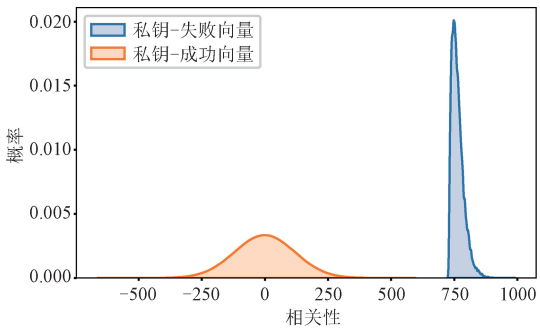

图 1　UD 下 Kyber512 的私钥与解密成功向量和解密失败向量之间的相关性

算法 7. 迭代增强的解密失败位置确定方法

输入:m 条解密失败密文 $\boldsymbol{T}=\{\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_m\}$

输出:每条解密失败密文的失败位置索引 r_i

初始化阶段

1. $\boldsymbol{F}=\{\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_k\}$,$\boldsymbol{P}=\{r_1,r_2,\cdots,r_k\}$ //猜测 k 条解密失败密文对应的解密失败位置
2. $\bar{\boldsymbol{E}}=\sum_{i\in[1,k]}\overline{\boldsymbol{C}_i^{r_i}}/\|\overline{\boldsymbol{C}_i^{r_i}}\|_2$ //初始化私钥方向估计

增强阶段

3. FOR $t\in[k+1,m]$ DO
4. $\quad\boldsymbol{C},r:=\mathrm{argmax}_{\boldsymbol{C}_i\in\boldsymbol{T}/\boldsymbol{F},r_i\in[0,n)}|\bar{\boldsymbol{E}}\cdot\overline{\boldsymbol{C}_i^{r_i}}|/\|\overline{\boldsymbol{C}_i^{r_i}}\|_2$
5. $\quad\boldsymbol{F}=\boldsymbol{F}\cup\{\boldsymbol{C}\}$,$\boldsymbol{P}=\boldsymbol{P}\cup\{r\}$
6. $\quad\bar{\boldsymbol{E}}=\bar{\boldsymbol{E}}+\overline{\boldsymbol{C}^r}/\|\overline{\boldsymbol{C}^r}\|_2$
7. END FOR

迭代阶段

8. FOR $d\in[1,ite]$ DO
9. $\quad$FOR $i\in[1,m]$ DO
10. $\quad\quad r:=\mathrm{argmax}_{r\in[0,n)}|\bar{\boldsymbol{E}}\cdot\overline{\boldsymbol{C}_i^r}|/\|\overline{\boldsymbol{C}_i^r}\|_2$
11. $\quad\quad$IF $r\neq r_i$ THEN
12. $\quad\quad\quad\bar{\boldsymbol{E}}=\bar{\boldsymbol{E}}-\overline{\boldsymbol{C}_i^{r_i}}/\|\overline{\boldsymbol{C}_i^{r_i}}\|_2$
13. $\quad\quad\quad\bar{\boldsymbol{E}}=\bar{\boldsymbol{E}}+\overline{\boldsymbol{C}_i^r}/\|\overline{\boldsymbol{C}_i^r}\|_2$
14. $\quad\quad$END IF
15. $\quad$END FOR
16. END FOR
17. 返回 $r_i,i\in[1,m]$

初始化阶段,首先估计私钥方向 $\bar{\boldsymbol{E}}$,$\bar{\boldsymbol{E}}$ 应当是一个与秘密向量 $\bar{\boldsymbol{S}}$ 的方向尽可能一致的向量。设 $\bar{\boldsymbol{E}}$ 与 $\bar{\boldsymbol{S}}$ 的夹角为 $\theta_{\boldsymbol{ES}}$,$|\cos(\theta_{\boldsymbol{ES}})|$ 越大,说明估计越接近真实私钥方向。对于失败密文 $\boldsymbol{C}$,设其失败位置为 r,则有 $|\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^r}+G[r]|>q/4$。取 $\bar{\boldsymbol{E}}=\overline{\boldsymbol{C}^r}/\|\overline{\boldsymbol{C}^r}\|_2$,计算 $|\cos(\theta_{\boldsymbol{ES}})|=\dfrac{|\bar{\boldsymbol{S}}^{\mathrm{T}}\bar{\boldsymbol{E}}|}{\|\bar{\boldsymbol{S}}\|_2\|\bar{\boldsymbol{E}}\|_2}=\dfrac{|\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^r}|}{\|\bar{\boldsymbol{S}}\|_2\|\overline{\boldsymbol{C}^r}\|_2}>\dfrac{q/4-G[r]}{\|\bar{\boldsymbol{S}}\|_2\|\overline{\boldsymbol{C}^r}\|_2}$。高维空间中,随机向量几乎总是彼此接近正交[34],随机向量之间的夹角余弦值强烈趋向于 0,因此即使 $|\cos(\theta_{\boldsymbol{ES}})|$ 稍微大于 0,$\bar{\boldsymbol{E}}$ 也是私钥方向的有效估计值。假设已知 k 条解密失败密文 $F=\{\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_k\}$ 对应的失败位置 $P=\{r_1,r_2,\cdots,r_k\}$,利用失败向量求和 $\bar{\boldsymbol{E}}=\sum_{i\in[1,k]}\overline{\boldsymbol{C}_i^{r_i}}/\|\overline{\boldsymbol{C}_i^{r_i}}\|_2$ 初始化私钥方向估计[26]。增强阶段,遍历剩余解密失败密文的所有可能旋转。对于每个旋转 $\overline{\boldsymbol{C}_i^{r_i}}$,计算旋转向量与私钥方向估计 $\bar{\boldsymbol{E}}$ 的内积。选择产生最大内积值的解密失败密文 $\boldsymbol{C}_i$ 及其对应失败位置 r_i,并将其添加到已确定的解密失败密文集 $\boldsymbol{F}$ 和失败位置集 $\boldsymbol{P}$ 中,然后

更新私钥估计值 $\bar{\boldsymbol{E}}$。随着已确定失败位置的解密失败密文数量增加,私钥方向估计的准确性逐渐提高。迭代进行该过程,直到无法进一步更新私钥方向或达到预设的最大迭代次数。实验中,最大迭代次数通常设为 $ite=5$ 以实现效率和准确度的平衡。

4.2.3 复杂度分析

初始化阶段,猜测 k 条解密失败密文的失败位置。对于解密失败密文 $\boldsymbol{C}$,有 $\|\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}+G\|_{\infty}>q/4$,存在系数 r,使得 $|\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{r}}+G[r]|>q/4$。根据 4.2.1 节,可根据 $G[r]$ 的符号确定解密失败的符号。假设 $\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{r}}+G[r]>q/4$,$G[r]$ 值越大,$\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{r}}>q/4-G[r]$ 成立的概率越大。因此,可按 $G[r]$ 值降序枚举所有可能的失败位置 r。记成功猜测一条解密失败密文失败位置的平均尝试次数为 Q,则成功猜中 k 条密文失败位置的平均尝试次数为 Q^{k}。增强阶段,假设 k 条密文的失败位置已知,记成功确定剩余全部失败位置的概率为 $\beta=P[r_i=\tau_i,i\in[k,m]]$,其中 τ_i 表示失败密文 $\boldsymbol{C}_i$ 真实的解密失败位置。则恢复所有解密失败密文位置的平均尝试次数为 $V=Q^{k}\times(1/\beta)$。随着已知失败密文的失败位置数量 k 的增加,枚举复杂度 Q^{k} 增大,概率值 β 增加,$1/\beta$ 减小。确定合适 k 值,使得整体平均尝试次数 V 最少。

4.2.4 实验验证

对于 UD 下的 Kyber512 和 LightSaber,本文分别使用 3000 条解密失败密文,应用上述迭代增强方法确定解密失败位置。实验结果见表 6,随着已知失败位置个数 k 的增加,恢复全部失败位置的成功率显著提升,但初始化阶段所需的尝试次数 Q^{k} 亦随之指数增长。对于 UD 下的 Kyber512 和 LightSaber,实测 Q 的经验值分别为 48 和 32。为使恢复全部解密失败位置所需总尝试次数最少,取 $k=1$,平均总尝试次数为 V 分别为 $2^{8.9}$ 和 $2^{5.3}$。

表 6 UD 下 Kyber512 和 LightSaber,恢复全部解密失败密文失败位置的成功率及平均尝试次数

	已知解密失败位置个数 k	1	2	3	4
Kyber512	恢复全部位置的成功率	10%	64%	94%	100%
	猜测复杂度 Q^{k}	48	48^{2}	48^{3}	48^{4}
	平均总尝试次数 V	$2^{8.9}$	$2^{11.8}$	$2^{16.8}$	$2^{22.8}$
LightSaber	恢复全部位置的成功率	79%	100%	100%	100%
	猜测复杂度 Q^{k}	32	32^{2}	32^{3}	32^{4}
	平均总尝试次数 V	$2^{5.3}$	2^{10}	2^{15}	2^{20}

4.3 利用解密失败密文恢复私钥

在确定解密失败符号与位置之后,可利用这些解密失败信息恢复私钥。下文分别利用 D'Anvers 等人提出的概率方法[21]和几何方法[22]进行密钥恢复。

4.3.1 概率方法恢复私钥

概率方法的主要思路是:先基于贝叶斯定理计算解密失败条件下私钥的后验概率,然后使用极大似然估计给出最可能的密钥估计值。

给定一条解密失败密文 $(\boldsymbol{C},G)$,设失败位置 $r=0$ 且为正失败,即 $\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{0}}+G>q_t$。在该解密失败条件下,私钥系数 $\bar{\boldsymbol{S}}_i$ 取某候选值的概率为:

$$P(\bar{\boldsymbol{S}}_i\mid\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{0}}>q_t-G)=\frac{P(\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{0}}>q_t-G\mid\bar{\boldsymbol{S}}_i)P(\bar{\boldsymbol{S}}_i)}{P(\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{0}}>q_t-G)}=\frac{P\left(\sum_{j\neq i}\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_j^{0}}>q_t-G-\bar{\boldsymbol{S}}_i\overline{\boldsymbol{C}_i^{0}}\right)P(\bar{\boldsymbol{S}}_i)}{P(\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}^{0}}>q_t-G)}\tag{5}$$

高维空间中的向量呈现出显著的正交趋势[34],因此发生线性相关的概率较低。给定多条解密失败密文 $\{(\boldsymbol{C}_1,G_1),\cdots,(\boldsymbol{C}_m,G_m)\}$,设解密失败位置 $r_i=0$ 且均为正失败。

假设 2. 假设解密失败向量 $\{\overline{\boldsymbol{C}_1^{r_1}},\overline{\boldsymbol{C}_2^{r_2}},\cdots,\overline{\boldsymbol{C}_m^{r_m}}\}$ 之间相互独立[21-22]。

根据独立性假设 2,私钥系数 $\bar{\boldsymbol{S}}_i$ 取某值的概率为:

$$\begin{aligned}&P(\bar{\boldsymbol{S}}_i\mid\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_k^{0}}>q_t-G_k,k\in[1,m])\\&=\frac{P(\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_k^{0}}>q_t-G_k,k\in[1,m]\mid\bar{\boldsymbol{S}}_i)P(\bar{\boldsymbol{S}}_i)}{P(\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_k^{0}}>q_t-G_k,k\in[1,m])}\\&=\frac{P(\bar{\boldsymbol{S}}_i)\prod_{k=1}^{m}P\left(\sum_{j\neq i}\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_{k,j}^{0}}>q_t-G-\bar{\boldsymbol{S}}_i\overline{\boldsymbol{C}_{k,i}^{0}}\mid\bar{\boldsymbol{S}}_i\right)}{\prod_{k=1}^{m}P(\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_k^{0}}>q_t-G_k)}\end{aligned}\tag{6}$$

记 $P_{\mathrm{fail}}[i]=P\left(\sum_{j\neq i}\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_j^{0}}>q_t-G-\bar{\boldsymbol{S}}_i\overline{\boldsymbol{C}_i^{0}}\right)$,项 $\sum_{j\neq i}\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_j^{0}}$ 的分布计算涉及 $n-1$ 个变量 $\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_j^{0}}$ 的卷积,直接计算复杂度高。为提高效率,本文应用中心极限定理(定理 3),将和式近似为正态分布,则

$$\begin{aligned}P_{\mathrm{fail}}[i]&=P\left(\frac{\sum_{j\neq i}\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_j^{0}}-\sum_{j\neq i}\overline{\boldsymbol{C}_j^{0}}\,\mathbb{E}[\bar{\boldsymbol{S}}_j]}{\sqrt{\sum_{j\neq i}\overline{\boldsymbol{C}_j^{0}}^{2}\cdot\mathrm{Var}[\bar{\boldsymbol{S}}_j]}}>\frac{q_t-G-\bar{\boldsymbol{S}}_i\overline{\boldsymbol{C}_i^{0}}-\sum_{j\neq i}\overline{\boldsymbol{C}_j^{0}}\,\mathbb{E}[\bar{\boldsymbol{S}}_j]}{\sqrt{\sum_{j\neq i}\overline{\boldsymbol{C}_j^{0}}^{2}\cdot\mathrm{Var}[\bar{\boldsymbol{S}}_j]}}\right)\\&\approx F_{\mathrm{norm}}\left(\frac{\bar{\boldsymbol{S}}_j\overline{\boldsymbol{C}_j^{0}}+\sum_{j\neq i}\overline{\boldsymbol{C}_j^{0}}\,\mathbb{E}[\bar{\boldsymbol{S}}_j]-q_t+G}{\sqrt{\sum_{j\neq i}\overline{\boldsymbol{C}_j^{0}}^{2}\cdot\mathrm{Var}[\bar{\boldsymbol{S}}_j]}}\right)\end{aligned}\tag{7}$$

其中，$F_{\text{norm}}()$为标准正态分布函数。将式(7)代入式(6)，即得私钥系数 $\bar{\boldsymbol{S}}_i$ 的后验概率分布，对每个系数概率分布取极大似然估计即得密钥估计值。

对于 UD 下 Kyber512，采取上述“概率方法”进行密钥恢复。对每个密钥系数 $\bar{\boldsymbol{S}}_i$，计算其在 m 条解密失败密文下的条件概率，并选取最大概率对应的取值作为其估计值。实验结果如图 2 所示，“概率方法”大约需要 13000 条解密失败密文，以 100%成功率恢复私钥向量的全部系数。

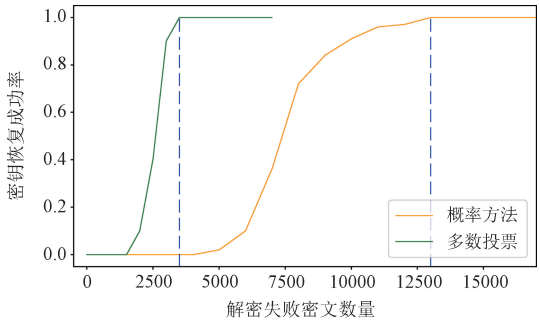

图 2　UD 下 Kyber512，密钥恢复成功率与解密失败密文数量的关系

4.3.2　几何方法恢复私钥

几何方法利用高维空间中内积值较大的两个向量的方向近乎一致的性质[34]，将解密失败向量的均值作为密钥方向的估计值。再结合解密失败的阈值进行密钥长度估计，从而恢复完整私钥。

设给定 m 条解密失败密文 $\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_m$ 及其失败位置 $r_1,r_2,\cdots,r_m$，则有 $\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_i^{r_i}}>q_t-G$。在高维空间中，内积值较大的两个向量的方向往往接近[34]。由于解密失败阈值 q_t 足够大，因此可用失败向量的均值来估计私钥方向，即 $\bar{\boldsymbol{E}}=\overline{\boldsymbol{C}_{rav}}/\|\overline{\boldsymbol{C}_{rav}}\|_2$，其中 $\overline{\boldsymbol{C}_{rav}}=\sum_{i\in[1,m]}\overline{\boldsymbol{C}_i^{r_i}}/\|\overline{\boldsymbol{C}_i^{r_i}}\|_2$。由于 $\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_i^{r_i}}$ 分布近似正态分布，尾部的概率密度呈指数平方衰减，则 $\bar{\boldsymbol{S}}^{\mathrm{T}}\overline{\boldsymbol{C}_i^{r_i}}\approx q_t-G$。于是 $\overline{\boldsymbol{E}'}\cdot\sum_{i\in[1,m]}\overline{\boldsymbol{C}_i^{r_i}}/m\approx q_t-G'$，其中 $\overline{\boldsymbol{E}'}$ 为私钥估计值；则有 $\|\overline{\boldsymbol{E}'}\|_2\approx m\cdot(q_t-G')/\left\|\sum_{i\in[1,m]}\overline{\boldsymbol{C}_i^{r_i}}\right\|_2$，其中 $G'=\sum_{i\in[1,m]}G_i/m$，则私钥向量的估计值为：$\overline{\boldsymbol{E}'}=\bar{\boldsymbol{E}}\cdot m\cdot(q_t-G')/\left\|\sum_{i\in[1,m]}\overline{\boldsymbol{C}_i^{r_i}}\right\|_2$。

对于 UD 下 LightSaber，采取上述“几何方法”进行密钥恢复。实验结果如图 3 所示，“几何方法”大约需要 9500 条解密失败密文，以 100%成功率恢复私钥向量的全部系数。

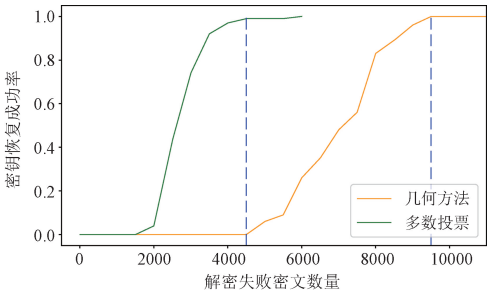

图 3　UD 下 LightSaber，密钥恢复成功率与解密失败密文数量的关系

4.4　多数投票方法优化密钥恢复

密钥恢复所需的解密失败密文数量是评估解密失败攻击复杂度的关键指标。实验发现，少量解密失败密文即可恢复大部分密钥系数。以 UD 下的 Kyber512 为例，约 2000 条失败密文即可恢复密钥向量 $\boldsymbol{S}=[-\boldsymbol{s},\boldsymbol{e}]$ 的 1024 个系数中的 939 个，但无法确定这些系数的位置。若要恢复完整密钥，所需解密失败数量急剧增加，达到 13000 条。

4.4.1　分组与投票策略

为了降低恢复完整密钥所需的解密失败密文数量，本文提出了一种基于多数投票的密钥恢复框架。首先对失败密文分组，组内独立恢复并输出候选结果；然后对所有组的输出做众数投票得到密钥近似估计值；再结合 LWE 约束 $\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e}\bmod q$，当准确恢复超过一半的系数及其位置后，代入该约束关系即可解出其余系数，重构完整密钥。

具体地，将收集到的 m 条解密失败密文随机分为 h 个子集，每个子集包含 m' 条失败密文。各子集彼此独立地执行密钥恢复。对每个系数索引 j，统计其在 h 个子集中的预测值并取众数作为全局估计 $E(\bar{\boldsymbol{S}}_j)$。随后按票数从高到低选出前一半系数估计值，代入 $\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e}\bmod q$ 求解其余系数，得到最终密钥 $\boldsymbol{S}=[-\boldsymbol{s},\boldsymbol{e}]$。该方法具体流程如算法 8 所示。

算法 8. 基于多数投票的密钥恢复方法

输入：m 条解密失败密文 $\boldsymbol{T}=\{\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_m\}$

输出：密钥估计值

分组恢复并多数投票

1. $\boldsymbol{B}_1=\{\boldsymbol{C}_{1,1},\boldsymbol{C}_{1,2},\cdots,\boldsymbol{C}_{1,m'}\},\cdots,\boldsymbol{B}_h=\{\boldsymbol{C}_{h,1},\boldsymbol{C}_{h,2},\cdots,\boldsymbol{C}_{h,m'}\}$
2. FOR $i\in[1,h]$ DO
3.　　FOR $j\in[0,2kn-1]$ DO
4.　　FOR $l\in[1,m']$ DO
5.　　$P_{\text{fail}}[l,j]\leftarrow\boldsymbol{C}_{i,l}$ using Eq. (7)
6.　　END FOR
7.　　$P_{\text{fail}}[j]=P(\bar{\boldsymbol{S}}_j)\cdot\prod_{i=1}^{m'}P_{\text{fail}}[l,j]$
8.　　END FOR

9. $\bar{\boldsymbol{S}}_{j,i}=\text{argmax}(P_{\text{fail}}[j])$ for $j\in[0,2kn-1]$
10. END FOR
11. FOR $j\in[0,2kn-1]$ DO
12. $E(\bar{\boldsymbol{S}}_j)=\text{mode}(\bar{\boldsymbol{S}}_{j,i})$ FOR $i\in[1,h]$
13. END FOR
代入置信度最高的一半系数至 $\boldsymbol{b}=\boldsymbol{As}+\boldsymbol{e}$
14. Sorted_counts = argsort($E(\bar{\boldsymbol{S}}_j)$.count)
15. index = sorted_counts[:$2kn/2-1$]
16. $E_{\text{half}}=E(\bar{\boldsymbol{S}})_{\{\text{index}\}}$
17. 代入 E_{half} 至 $\boldsymbol{b}=\boldsymbol{As}+\boldsymbol{e}$ 恢复剩余密钥系数
18. 返回密钥 $\boldsymbol{S}=[-\boldsymbol{s},\boldsymbol{e}]$

4.4.2 密钥恢复攻击

本节对 UD 下的 Kyber512 和 LightSaber 进行实际的解密失败攻击验证实验。针对 4.1 节中收集到的解密失败密文，利用 4.2 节的方法确定解密失败的符号和位置，然后分别使用 4.3 节的“概率方法”和“几何方法”进行密钥恢复，并应用本节提出的多数投票框架降低密钥恢复所需的密文数量。

实验设置取参数 $h=23$，$m'=m/2$，选择全部分组中众数频率最高的一半密钥 $\boldsymbol{S}=[-\boldsymbol{s},\boldsymbol{e}]$ 系数代入 $\boldsymbol{b}=\boldsymbol{As}+\boldsymbol{e}$ 重构完整密钥。实验结果见图 2 和图 3 中“多数投票”曲线，对于 UD 下 Kyber512 和 LightSaber，多数投票方法所需的解密失败密文数量分别为 3000 和 4500。相较于已有的“概率方法”和“几何方法”，密钥恢复所需的解密失败数量分别减少了约 77%和 53%。

对于 CBD 和 UD 下的 Kyber768、Kyber1024、Saber 和 FireSaber 的解密失败攻击，同样采取上述多数投票框架进行密钥恢复。实验结果如表 5 所示，攻击所需失败密文数量降低 50%以上。

4.5 解密失败攻击复杂度分析

记解密失败密文收集阶段的计算复杂度为 cost_1，解密失败概率为 δ，密钥恢复所需的失败密文数量为 W。D’Anvers 等人提出“失败增强”技术[21-22]，可在量子计算模型下借助 Grover 搜索加速失败密文收集，但本文在经典计算模型下进行实际数据收集，因此 $\text{cost}_1=W\cdot(1/\delta)=W/\delta$。随后，应用迭代增强方法确定解密失败符号和位置，设平均需要 V 次尝试可成功定位。最后，使用多数投票方法恢复密钥，记每次密钥恢复的复杂度为 cost_2。于是，解密失败攻击的总复杂度近似为 $\text{cost}_1+V\cdot\text{cost}_2$。在实际攻击中，$V\cdot\text{cost}_2$ 远小于 cost_1，因此解密失败攻击的总复杂度 $\text{cost}_{\text{total}}\approx\text{cost}_1=W/\delta$。

对于 UD 下的 Kyber512，$W\approx3000$，$\delta\approx2^{-25.4}$，$V=2^{8.9}$，因此解密失败攻击的总复杂度约为 $W/\delta\approx3000\times2^{25.4}\approx O(2^{37})$。

对于 UD 下的 LightSaber，$W\approx4500$，$\delta\approx2^{-30.1}$，$V=2^{5.3}$，因此解密失败攻击的总复杂度约为 $W/\delta\approx4500\times2^{30.1}\approx O(2^{42.2})$。

对于 CBD 下的 Kyber512 和 LightSaber，由于解密失败概率极低，难以实际获取解密失败密文。本文采用 D’Anvers 等人[23]提出的仿真方法模拟解密失败密文，然后使用上述多数投票方法进行密钥恢复。对于 CBD 下的 Kyber512，需要约 120 条解密失败密文；对于 CBD 下的 LightSaber，需要约 180 条解密失败密文。因此针对原始 Kyber512 和 LightSaber 的解密失败攻击的总复杂度分别约为 $120\times2^{139}\approx O(2^{146})$ 和 $180\times2^{120.3}\approx O(2^{127.8})$。

针对 CBD 和 UD 下的 Kyber768、Kyber1024、Saber 与 FireSaber，采取同样的仿真方法模拟生成解密失败密文，然后应用本文提出的多数投票方法进行密钥恢复，进而评估解密失败攻击的复杂度。实验结果如表 5 所示，将 Kyber 和 Saber 的随机分布从原始 CBD 替换为相同取值范围的 UD 后，其解密失败攻击复杂度显著下降，降低约 $2^{86}\sim2^{123}$ 倍。

需要注意的是，在 NIST PQC 单目标安全模型下，攻击者最多可对解封装谕言机发起 2^{64} 次查询。因此，在该攻击模型下，CBD 分布下的 Kyber 与 Saber 所需的解封装查询次数远超上限，解密失败攻击几乎不可实施，对实际安全性构成影响极小。相比之下，UD 分布下解密失败密文收集所需解封装查询复杂度显著降低，使得攻击者在可允许的查询次数内收集到足量失败密文成为可能，显著提升解密失败攻击成功率。

综上，直接将 Kyber 和 Saber 的随机分布从原始 CBD 替换为相同取值区间的 UD 并不可取。虽然 UD 增强了底层困难问题实例的抗格攻击能力，但也导致了过高的解密失败概率。在 NIST 解封装查询次数上限内即可收集足量失败密文，大幅降低了解密失败攻击复杂度，使得 Kyber 和 Saber 更易遭受解密失败攻击的威胁。

5 随机分布对格 KEM 实现性能的影响分析

随机分布的选择不仅对密码方案的理论安全性产生根本影响，也直接关系其实现性能。在对性能的影响分析方面，本文综合考虑格 KEM 的存储开销与算法运行时间。

5.1　随机分布对算法存储开销的影响

5.1.1　密钥长度

对于 Kyber,模数 $q=3329<2^{12}$,故$\mathbb{Z}_q$中每个系数需要 12 个比特表示。公钥 $pk:=\text{Encode}_{12}(\hat{\boldsymbol{t}} \bmod^{+} q \| \rho)$,其中 $\hat{\boldsymbol{t}}$ 含 kn 个$\mathbb{Z}_q$系数,ρ 为 32 字节随机数。私钥 $sk:=(sk' \| pk \| \text{H}(pk) \| z)$,其中 $sk':=\text{Encode}_{12}(\hat{\boldsymbol{s}} \bmod^{+} q)$,$\text{H}(pk)$和 z 各为 32 字节。于是,在不改变模数和编码方式的条件下,Kyber 的公钥长度恒为 $12 \cdot k \cdot n/8+32$ 字节,私钥长度恒为 $24 \cdot k \cdot n/8+96$ 字节。

对于 Saber,模数 $q=2^{13}$,$p=2^{10}$,故$\mathbb{Z}_q$中每个系数需要 13 个比特表示,$\mathbb{Z}_p$中每个系数需要 10 个比特表示。公钥 $pk:=(\rho \| \boldsymbol{b})$,其中 $\boldsymbol{b}$ 含 kn 个$\mathbb{Z}_p$上的系数,ρ 为 32 字节随机数。私钥 $sk:=(z \| \text{H}(pk) \| pk \| \boldsymbol{s})$,其中 $\boldsymbol{s}$ 的系数在$\mathbb{Z}_q$上存储,$\text{H}(pk)$和 z 均为 32 字节。在不改变模数的条件下,Saber 的公钥长度恒为 $10 \cdot k \cdot n/8+32$ 字节,私钥长度恒为 $(10+13) \cdot k \cdot n/8+96$ 字节。

Kyber 和 Saber 在 CBD 和 UD 下的密钥长度如表 7 所示,在模数和编码方式固定的条件下,分布变化并不会对公钥和私钥长度造成影响。

5.1.2　密文长度

对于 Kyber,密文 $\boldsymbol{c}:=(\boldsymbol{c}_1,c_2)$,根据压缩函数定义(定义 5),$\boldsymbol{c}_1$ 系数小于 2^{d_u},需 d_u 个比特存储;c_2 系数小于 2^{d_v},需要 d_v 个比特存储。因此 Kyber 密文长度为 $\text{length}(\boldsymbol{c})=(k \cdot d_u+d_v) \cdot n/8$,在压缩参数$(d_u,d_v)$不变时,密文长度保持不变。

对于 Saber,密文 $\boldsymbol{c}:=(c_m,\boldsymbol{b}')$,其中 c_m 的系数小于 T,需要 $\log_2 T$ 个比特表示;$\boldsymbol{b}' \in R_p^{k \times 1}$,每个系数需要 $\log_2 p$ 个比特表示。因此 Saber 密文长度为:$\text{length}(\boldsymbol{c})=(k \cdot \log_2 p+\log_2 T) \cdot n/8$,当模数 p 与 T 不变时,密文长度不随分布改变而变化。

Kyber 和 Saber 在 CBD 和 UD 下的密文长度如表 7 所示,在模数和压缩参数固定的条件下,分布变化并不会对密文长度造成影响。

表 7　Kyber 和 Saber 在相同取值区间的 CBD 和 UD 下的密钥、密文长度及运行时间(CPU 周期)比较

	Kyber512		Kyber768		Kyber1024		LightSaber		Saber		FireSaber	
私钥分布区间	[−3,3]		[−2,2]		[−2,2]		[−5,5]		[−4,4]		[−3,3]	
私钥分布类型	CBD	UD	CBD	UD	CBD	UD	CBD	UD	CBD	UD	CBD	UD
私钥长度(字节)	1632	1632	2400	2400	3168	3168	1568	1568	2304	2304	3040	3040
公钥长度(字节)	800	800	1184	1184	1568	1568	672	672	992	992	1312	1312
密文长度(字节)	768	768	1088	1088	1568	1568	736	736	1088	1088	1472	1472
$Q_{\text{poly\_PRBs}}$	192	110	128	154	128	154	320	186	256	228	192	110
$Q_{\text{poly\_sam}}$	256	410	256	292	256	292	256	372	256	456	256	410
$T_{\text{poly\_sam}}$	2178	2838	1614	2386	1631	2412	2828	3556	2562	3725	2156	2762
密钥生成时间	90503	93678	153714	160314	246225	253367	60625	64378	114863	117396	188320	192253
封装时间	120314	129738	185361	191273	285712	293562	86690	91280	149045	153289	232354	238156
解封装时间	147688	156237	216446	224879	329308	337257	88416	92834	159308	162782	248367	253148

注:$Q_{\text{poly\_PRBs}}$、$Q_{\text{poly\_sam}}$、$T_{\text{poly\_sam}}$ 分别代表多项式环元素生成所需的 PRBs 数量、采样算法调用次数以及运行时间。

5.2　随机分布对算法运行时间的影响

随机分布变化对算法的运行时间产生显著影响。格 KEM 中,运行时间受分布影响的模块主要包括私钥、随机数及误差等多项式向量的生成过程。多项式生成过程包含两阶段,先由哈希函数挤压得到采样所需的伪随机比特串(Pseudo-Random Bits,PRBs),然后利用这些 PRBs 依分布采样。当分布变化时,哈希函数的挤压轮数以及采样算法的运行时间也随之变化,进而影响算法的整体运行时间。

5.2.1　CBD 采样

对于 CBD 采样(见算法 9),在区间$[-u,u]$上依分布采样一个值需要的 PRBs 数量为 $Q_{\text{PRBs}}=2u$,可保证 100% 成功率生成目标值。以多项式环元素的生成为例,采样自 CBD 的多项式 $f \in R_q$ 生成所需采样算法的调用次数为 $Q_{\text{poly\_sam}}=n$,所需 PRBs 数量 $Q_{\text{poly\_PRBs}}=n \cdot 2u$。

算法 9. 中心二项分布 β_u 采样

输入:伪随机比特串 $\bar{b}=(b_0,b_1,\cdots,b_{2n-1}) \in \{0,1\}^{2u}$

输出:服从 β_u 分布的一个样本

1. $a=\sum_{j=0}^{u-1} b_j$
2. $b=\sum_{j=0}^{u-1} b_{u+j}$
3. 返回 $a-b$

5.2.2　UD 采样

对于 UD 采样,若区间元素个数 M 为 2 的幂次,仅需 $\log_2 M$ 个 PRBs 即可成功采样。然而区间$[-u,u]$中元素个数 $M=2u+1$ 并非 2 的幂次,需

使用拒绝采样算法(见算法 10)。在$[-u,u]$上依 UD 采样一个值至少需要$\lceil\log_2(2u+1)\rceil$个 PRBs,单次采样成功率为$p_{\text{suc}}=(2u+1)/2^{\lceil\log_2(2u+1)\rceil}$。因此采样一个值平均所需执行采样算法$Q_{\text{sam}}=1/p_{\text{suc}}$次,平均所需 PRBs 数量为$Q_{\text{PRBs}}=\lceil\log_2(2u+1)\rceil/p_{\text{suc}}$。以多项式环元素生成为例,采样自 UD 的多项式$g\in R_q$生成所需调用采样算法$Q_{\text{poly_sam}}=n/p_{\text{suc}}$次,所需 PRBs 数量为$Q_{\text{poly_PRBs}}=n\cdot\lceil\log_2(2u+1)\rceil/p_{\text{suc}}$。

算法 10. 均匀分布$\mathcal{U}_u$拒绝采样

输入:伪随机比特串$\bar{b}=(b_0,b_1,\cdots)\in\{0,1\}^{\lceil\log_2(2u+1)\rceil}$

输出:服从$\mathcal{U}_u$分布的一个样本或者$\perp$

1. $d=$binary_to_decimal($\bar{b}$) // 转为十进制数
2. IF $d<2u+1$ THEN
3. 返回 $d-u$
4. ELSE
5. 返回$\perp$

5.2.3 运行时间分析

在 Kyber 和 Saber 中,分布参数$u\in\{2,3,4,5\}$。以多项式环R_q的元素生成为例,设哈希函数的运行时间为$T_{\text{poly_hash}}$,单次采样的运行时间为T_{sam},则生成n个系数所需时间为$T_{\text{poly_sam}}=n\cdot T_{\text{sam}}/p_{\text{suc}}$。因此,多项式环$R_q$的元素生成总时间为$T_{\text{poly}}=T_{\text{poly_hash}}+T_{\text{poly_sam}}$。

参考实现上,Kyber 使用 SHAKE256 算法,每次挤压输出 136 字节,因此生成一个多项式所需的挤压轮数为$N_{\text{poly_squ}}=\lceil n\cdot Q_{\text{PRBs}}/(8\times136)\rceil$;Saber 使用 SHAKE128 算法,每次挤压输出 168 字节,因此生成一个多项式所需的挤压轮数为$N_{\text{poly_squ}}=\lceil n\cdot Q_{\text{PRBs}}/(8\times168)\rceil$。由此可见,多项式生成所需 PRBs 数量差异将直接影响挤压轮数,从而影响采样时间。

5.2.4 实验对比

对于 Kyber 与 Saber,分别计算在 CBD 和 UD 采样下,多项式环R_q元素生成所需 PRBs 个数、哈希函数挤压轮数以及采样算法调用次数。同时记录多项式环元素生成时间,以及 KEM 方案的密钥生成、封装与解封装阶段运行时间。实验环境配备 2.7GHz Intel Xeon CPU、1TB 内存的 56 核 Linux 服务器,记录 1000000 次执行的平均 CPU 周期。

如表 7 所示,当$u=2$时,UD 采样所需 PRBs 更多,哈希操作耗时更长;当$u\in\{3,4,5\}$时,UD 采样所需 PRBs 更少,哈希操作耗时更短。然而,由于 UD 采样存在拒绝概率,采样函数的平均调用次数相较于 CBD 增加约 14%~78%,导致 UD 采样生成多项式的速度慢于 CBD 采样约 25%~65%。由于 Kyber 和 Saber 的密钥生成、封装以及解封装过程均包含多项式环元素的生成,因此如表 7 所示:在当前的参数设置下,采用 UD 采样会延长 Kyber 和 Saber 的密钥生成、封装以及解封装的运行时间。

6 格 KEM 安全性与性能的平衡

综上分析,随机分布变化对格 KEM 的抗格攻击能力、解密失败风险及实现性能产生显著而不均衡的影响。在 NIST PQC 第三轮提交的文档中,Saber 设计者提出了采用 UD 的变体 uSaber,实现更短的私钥和更快的运行速度[16];但其使用均值非零的采样区间$[-u,u-1]$的安全性有待进一步论证。为平衡上述矛盾,本文引入 Zhang 等人[35]提出的非对称 LWE 问题以支持分布参数的灵活选择与安全保障,并据此提出采用 UD 的 Kyber 变体——uKyber。

6.1 灵活的参数选择

6.1.1 正规形 LWE 问题

正规形 LWE 问题中私钥与误差向量的分布类型及参数大小相同[36],广泛用于格密码的构造。然而,当采用 UD 的 Kyber 变体沿用正规形 LWE 问题时($u_s=u_e$),难以选择合适的参数实现方案安全强度和解密失败概率的平衡。如图 4 所示,当$u_s=u_e=1$时,方案的安全强度整体低于原始 Kyber,无法满足 NIST PQC 的安全等级要求;当$u_s=u_e=2$或 3 时,安全强度有所提升,但解密失败概率显著增大,因此难以实现二者的有效平衡。

6.1.2 非对称 LWE 问题

为了平衡 UD 变体方案的安全强度和解密失败概率,本文采用 Zhang 等人[35]提出的非对称 MLWE(Asymmetric MLWE,AMLWE)问题。在 AMLWE 问题中,私钥和误差向量的分布参数大小无需相同,从而实现更灵活的参数调整。

令n,q,k,l,u_s,u_e为正整数,采样自 UD 的 AMLWE 问题记为$\text{AMLWE}_{n,q,k,l,u_s,u_e}$,对应实例$(\boldsymbol{A},\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e})$,$\boldsymbol{A}\leftarrow R_q^{k\times l}$,$\boldsymbol{s}\leftarrow(\mathcal{U}_{u_s}^n)^l$,$\boldsymbol{e}\leftarrow(\mathcal{U}_{u_e}^n)^k$。关于 ALWE 问题的安全性论证,见文献[35]附录 C。

如表 8 所示,设置 uKyber 的分布参数$u_s=1$,$u_e=2$,即私钥和误差向量的系数分别从区间$[-1,1]$和$[-2,2]$上均匀采样。为匹配 Kyber 的安全强度和解密失败概率,调整封装阶段的随机数$\boldsymbol{r}$和误

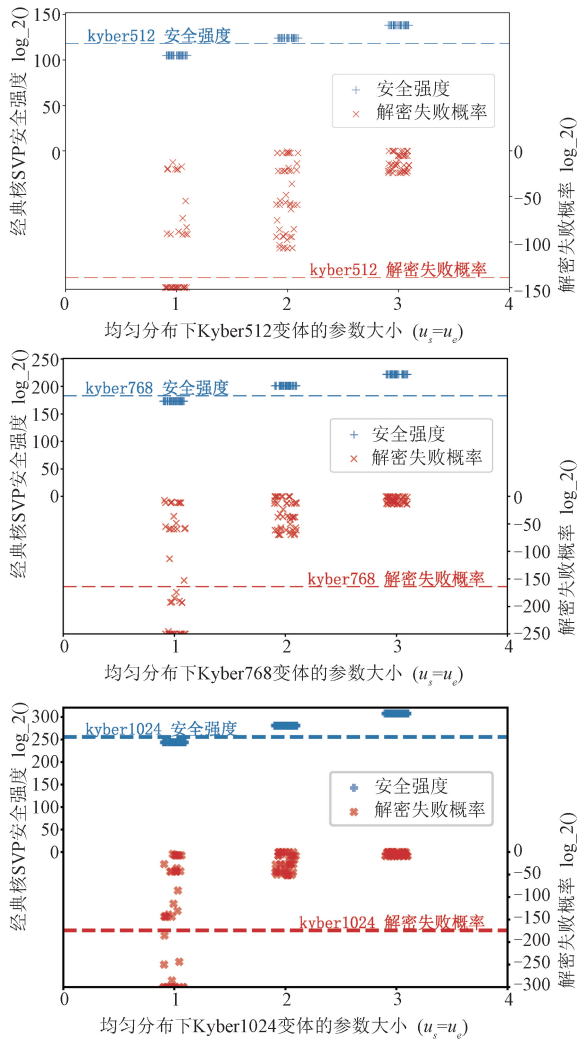

图 4　UD 下基于正规形 LWE 的 Kyber 变体的安全强度与解密失败概率(横轴表示 UD 取值区间参数值,“+”表示变体方案的安全强度,“×”表示变体方案的解密失败概率。同一列中的不同点值对应不同压缩参数(d_u,d_v)下的计算值)

差向量$\boldsymbol{e}_1,\boldsymbol{e}_2$的取值范围,以及压缩参数$(d_u,d_v)$的大小,并保持 n,q,k 等核心参数不变。uKyber 完整的结构设计与安全性证明见附录 C 和附录 D。

表 8　uKyber 方案推荐参数

	uKyber512	uKyber768	uKyber1024
n	256	256	256
k	2	3	4
q	3329	3329	3329
(u_s,u_e)	(1,2)	(1,2)	(1,2)
(u_r,u_{e_1},u_{e_2})	(1,1,1)	(1,1,1)	(1,1,1)
(d_u,d_v)	(9,3)	(10,3)	(10,5)
安全等级	1 (AES-128)	3 (AES-192)	5 (AES-256)

6.2　安全性分析

与 MLWE 问题的安全强度评估类似,本文不考虑 AMLWE 问题的模格结构,将其视为一般格上的 ALWE 问题求解。然后再利用 Zhang 等人[35]提出的缩放技术将针对 LWE 问题的格攻击方法推广到求解 ALWE 问题。Zhang 等人利用缩放技术平衡目标向量每个分量的大小,并建立了格攻击下 ALWE 问题和 LWE 问题安全强度之间的近似关系:$\mathrm{ALWE}_{n,q,m,\alpha_1,\alpha_2}\approx\mathrm{LWE}_{n,q,m,\sqrt{\alpha_1\alpha_2}}$,其中 α_1 和 α_2 分别为私钥和误差向量系数分布的标准差。据此,本文评估了 uKyber 在格攻击下的安全强度。

对于解密失败概率,采取 4.1 节的解密失败概率计算方法评估 uKyber 的解密失败概率。如表 9 所示,在相同安全等级下,uKyber 实现了与原始 Kyber 相近的安全强度和解密失败概率。

表 9　Kyber 与 uKyber 的安全强度与解密失败概率对比

方案	随机分布标准差 σ	解密失败概率 δ	经典安全强度	量子安全强度
Kyber512	1.225	2^{-139}	118	107
uKyber512	1.075	2^{-134}	114	103
Kyber768	1	2^{-164}	183	166
uKyber768	1.075	2^{-156}	186	169
Kyber1024	1	2^{-174}	256	232
uKyber1024	1.075	2^{-181}	261	237

6.3　性能分析

相较 Kyber,uKyber 呈现多方面性能优势:密文更短、伪随机比特消耗更少,且整体运行速度更快。

6.3.1　更小的密文尺寸

Kyber 的密文长度计算为:$\mathrm{length}(\boldsymbol{c})=(k\cdot d_u+d_v)\cdot n/8$。相较原始 Kyber,uKyber 的压缩参数$(d_u,d_v)$减小,从而实现更短的密文长度。如表 10 所示,uKyber512/uKyber768/uKyber1024 的密文长度分别为 672/1056/1440 字节,均小于 Kyber512/Kyber768/Kyber1024 的 768/1088/1568 字节。相较原始 Kyber,平均缩短约 100 字节。

6.3.2　更少的伪随机比特消耗

5.2 节指出,区间$[-u,u]$上的 UD 采样存在拒绝概率,会带来 PRBs 过度消耗。以 $u=2$ 为例,$\mathcal{U}_2=[-2,2]$含 5 个元素,至少需$\lceil\log_2 5=3\rceil$个 PRBs 才能编码一个候选值。3 个 PRBs 可表示 $2^3=8$ 个状态,若采样值 $d>4$ 则需要拒绝并重采,拒绝概率为 3/8。因此,平均需要 3/(5/8)=4.8 个 PRBs 才能成功从$\mathcal{U}_2$采样一个值。为减少 PRBs 消耗,本文提出多值拒绝采样的方法,以降低拒绝概率。

算法 11. 多值均匀分布拒绝采样

输入:伪随机比特串 $\bar{b}=(b_0,b_1,\cdots)\in\{0,1\}^{\lceil\log_2(2u+1)^m\rceil}$

输出：服从$\mathcal{U}_u$的m个样本或者⊥

1. d=binary_to_decimal($\bar{b}$) //二进制转十进制
2. IF $d<(2u+1)^m$ THEN
3. FOR $j\leftarrow 0$ TO $m-1$ DO
4. $c\leftarrow d \bmod (2u+1)$
5. $d\leftarrow \lfloor d/(2u+1) \rfloor$
6. $V_j\leftarrow c$
7. END FOR
8. 返回V_j-u FOR $j\in[0,m]$
9. ELSE
10. 返回⊥
11. END IF

如算法11所示，多值均匀分布拒绝采样将$[0,k-1]$中均匀随机采样m个值，转化为$[0,k^m-1]$中均匀随机采样1个值c。然后将c转换为k进制形式$(u_1u_2\cdots u_m)_{(k)}$，得到的$u_j,j\in[1,m]$均服从$[0,k-1]$上的均匀分布。采用均匀分布拒绝采样方法获取c，需要$\lceil\log_2(k^m)\rceil$个PRBs，拒绝概率为$1-(k^m/2^{\lceil\log_2(k^m)\rceil})$。

当$u=2$时，$k=2u+1=5$，取$m=3$，即从$[0,4]$一次性采样3个数。将$\lceil\log_2(5^3)\rceil=7$个PRBs转为十进制数$c$。若$c<5^3$，则将$c$转为5进制数$(b_1b_2b_3)_{(5)}$；否则拒绝。成功采样的概率为$5^3/2^7=125/128$，故在$[0,k-1]$上采样一个值所需PRBs个数为$(7/(125/128))/3\approx 2.4$。与直接在$\mathcal{U}_2$上采样一个值需要4.8个PRBs相比，PRBs数量消耗减少50%。

类似地，对于$u=1,k=2u+1=3$，设$m=5$。选择随机$\lceil\log_2(3^5)\rceil=8$个PRBs并将其转换为十进制数$c$。如果$c<3^5$，将$c$转换为3进制数$(b_1b_2b_3b_4b_5)_{(3)}$；否则拒绝$c$。成功采样的概率为$3^5/2^8=243/256$，平均采样一个值所需的PRBs个数为$(8/(243/256))/5\approx 1.7$。

表10展示了Kyber和uKyber在密钥生成、封装与解封装阶段私钥和误差向量采样所需的PRBs个数。结果表明，uKyber采样所需的PRBs个数不足Kyber所需的一半。

6.3.3 更快的运行速度

Kyber的私钥和误差向量采样所需的PRBs由哈希函数SHAKE-256生成，更少的PRBs消耗减少了哈希函数的挤压轮数，降低了哈希函数的运行时间。进一步，uKyber使用的“多值拒绝采样算法”有效降低了拒绝概率与重复采样次数，从而提升了采样效率并缩短密钥生成与封装/解封装阶段的总体运行时间。

本文在Kyber官方C参考实现基础上实现了uKyber算法，使用GCC编译及O3优化，测试其密钥生成、封装和解封装的运行时间。实验环境配备2.7GHz Intel Xeon CPU、1TB内存的56核Linux服务器，每个算法均执行1000000次，统计平均CPU周期。实验结果如表10所示，uKyber在运行时间方面具有一定优势。例如，相较于Kyber512，uKyber512在密钥生成、封装与解封装阶段速度分别提高约7%、4%和3%。

表10 uKyber与Kyber的实现性能对比

	大小(字节)			伪随机比特PRBs个数			运行时间(CPU周期)		
	私钥	公钥	密文	密钥生成	封装	解封装	密钥生成	封装	解封装
Kyber512	1632	800	768	6144	6144	6144	90503	120314	147688
uKyber512	1632	800	672	2126	2203	2203	84169	115509	142887
Kyber768	2400	1184	1088	6144	7168	7168	153714	185361	216446
uKyber768	2400	1184	1056	3189	3048	3048	149700	178861	209720
Kyber1024	3168	1568	1568	8192	9216	9216	246225	285712	329308
uKyber1024	3168	1568	1440	4252	3944	3944	242429	280667	317967

6.3.4 与Dilithium共享采样模块

在后量子密码的迁移与部署中(如TLS协议)，常需要协同实现用于密钥交换的KEM与用于认证的数字签名算法。资源受限平台对实现面积与功耗的要求较为严格，因此设计可复用模块来降低组合实现的开销具有重要意义。

Kyber与Dilithium同属于CRYSTALS密码套件，并分别被NIST选为KEM和数字签名标准，常在同一处理器上协同实现[37-39]。由于uKyber与Dilithium均采用UD采样，可共享统一的采样模块，从而减少硬件/软件实现中采样逻辑的重复设计，提高系统集成度与实现紧凑性。由此可见，在与Dilithium组合部署的场景下，uKyber相比传统Kyber更具工程优势。

7 讨 论

本节进一步讨论“多值均匀分布拒绝采样”对

Dilithium 采样过程的加速效果。Dilithium 的私钥向量 $\boldsymbol{s}_1,\boldsymbol{s}_2$ 需在区间$[-\eta,\eta]$上进行均匀随机采样，$\eta\in\{2,4\}$。为提高采样效率，本文在现有拒绝采样方法的基础上引入“多值均匀分布拒绝采样”对原算法进行优化。

对于 Dilithium-2 和 Dilithium-5($\eta=2$)，根据第 6.3 节的描述，每次选择 7 个 PRBs 即可以 125/128 的成功率生成 3 个服从$[-2,2]$上均匀分布的值。对于 Dilithium-3($\eta=4$)，取 $m=5$，即随机选取$\lceil \log_2(9^5)\rceil=16$ 个 PRBs 以 $9^5/2^{16}$ 的成功率生成 5 个在$[-4,4]$上均匀分布的值，则每个值的平均 PRBs 消耗为$(16/0.9)/5\approx 3.56$。

本文在 Dilithium 官方 C 参考实现基础上实现该优化算法，使用 GCC 编译及 O3 优化，测试其多项式生成、私钥采样以及密钥生成阶段的运行时间。实验环境配备 2.7GHz Intel Xeon CPU、1TB 内存的 56 核 Linux 服务器，每个算法均执行 1000000 次，统计平均 CPU 周期。实验结果如表 11 所示，优化后的 Dilithium 在多项式生成、私钥采样以及密钥生成阶段的运行时间显著低于原始方案，其中私钥采样过程的运行时间减少了约 40%。

表 11　多值均匀分布拒绝采样算法对 Dilithium 运行时间(CPU 周期)的优化效果

方案		多项式生成	私钥采样	密钥生成
Dilithium-2	原始	1548	11232	106034
	优化	929	7413	99870
Dilithium-3	原始	1604	17893	200767
	优化	920	9970	170684
Dilithium-5	原始	1559	20033	297608
	优化	961	13912	285973

8　结　论

本文以 Kyber 和 Saber 为研究示例，系统评估了随机分布选择对格 KEM 安全性与性能的影响。与 CBD 相比，在相同取值区间下采用 UD 可将 Kyber 和 Saber 的安全强度提升约 12 比特～25 比特，但同时显著增大解密失败概率，并使多项式生成过程的运行时间增加约 25%～50%。为平衡上述矛盾，本文引入 ALWE 问题以支持参数的灵活调整，并据此提出 uKyber 变体，实现安全强度与解密失败概率的平衡。进一步，本文通过参数优化并提出多值均匀分布拒绝采样算法，在缩短了密文长度的同时提升运行速度，为后量子密码的分布优化设计提供了新的思路与范例。未来工作考虑将本文研究方法推广至更多格 KEM 方案(如 FrodoKEM)，并围绕更丰富的随机分布类型开展系统性对比分析。

参　考　文　献

[1] Shannon C E. Communication theory of secrecy systems. The Bell System Technical Journal, 1949,28(4):656-715

[2] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms//Proceedings of the International Cryptology Conference. California, USA, 1984:10-18

[3] Boneh D. The decision Diffie-Hellman problem//Proceedings of the International Algorithmic Number Theory Symposium. Berlin, Germany, 1998:48-63

[4] Katz J, Lindell Y. Introduction to modern cryptography: Principles and protocols. Boca Raton, USA: Chapman and Hall/CRC, 2007

[5] Alagic G, Apon D, Cooper D, et al. Status report on the third round of the NIST post-quantum cryptography standardization process. USA Department: National Institute of Standards and Technology, Technical Report: NISTIR 8413, 2022

[6] Regev O. On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM, 2009,56(6): 1-40

[7] Brakerski Z, Langlois A, Peikert C, et al. Classical hardness of learning with errors//Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing. San Diego, USA, 2013:575-584

[8] Bos J W, Costello C, Naehrig M, et al. Post-quantum key exchange for the TLS protocol from the ring learning with errors problem//Proceedings of the 2015 IEEE Symposium on Security and Privacy. San Jose, USA, 2015:553-570

[9] Groot Bruinderink L, Hülsing A, Lange T, et al. Flush, gauss, and reload: A cache attack on the BLISS lattice-based signature scheme//Proceedings of the International Conference on Cryptographic Hardware and Embedded Systems. Berlin, Germany, 2016:323-345

[10] Espitau T, Fouque P A, Gérard B, et al. Side-channel attacks on BLISS lattice-based signatures: Exploiting branch tracing against strongswan and electromagnetic emanations in microcontrollers//Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. Dallas, USA, 2017:1857-1874

[11] Döttling N, Müller-Quade J. Lossy codes and a new variant of the learning with errors problem//Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Germany, 2013:18-34

[12] Micciancio D, Peikert C. Hardness of SIS and LWE with small parameters//Proceedings of the Annual Cryptology Conference. Berlin, Germany, 2013:21-39

[13] Cabarcas D, Göpfert F, Weiden P. Provably secure LWE en-

cryption with smallish uniform noise and secret//Proceedings of the 2nd ACM Workshop on Public-Key Cryptography. Kyoto, Japan, 2014:33-42

[14] Alkim E, Ducas L, Pöppelmann T, et al. Post-quantum key exchange-a new hope//Proceedings of the 25th USENIX Security Symposium. Austin, USA, 2016:327-343

[15] Avanzi R, Bos J, Ducas L, et al. Crystals-Kyber algorithm specifications and supporting documentation (version 3.02). USA Department: National Institute of Standards and Technology, Technical Report: NIST PQC Round 3, 2022

[16] NIST. Fips 203: Module-lattice-based key-encapsulation mechanism standard. U.S. Department: National Institute of Standards and Technology, Technical Report: FIPS 203, 2024

[17] D'Anvers J P, Karmakar A, Roy S S, et al. Saber: Mod-LWR based KEM (round 3 submission). U.S. Department: National Institute of Standards and Technology, Technical Report: NIST PQC Round 3, 2022

[18] Qin Y, Cheng C, Zhang X, et al. A systematic approach and analysis of key mismatch attacks on lattice-based NIST candidate KEMs//Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security. Singapore, 2021:92-121

[19] Okada S, Wang Y. Recovery attack on bob's reused randomness in Crystals-Kyber and Saber//Proceedings of the International Conference on Provable Security. Daejeon, Republic of Korea, 2021:155-173

[20] Hofheinz D, Hövelmanns K, Kiltz E. A modular analysis of the Fujisaki-Okamoto transformation//Proceedings of the Theory of Cryptography Conference. Baltimore, USA, 2017:341-371

[21] D'Anvers J P, Guo Q, Johansson T, et al. Decryption failure attacks on IND-CCA secure lattice-based schemes//Proceedings of the IACR International Workshop on Public Key Cryptography. Cham, Switzerland, 2019:565-598

[22] D'Anvers J P, Rossi M, Virdia F. One failure is not an option: Bootstrapping the search for failures in lattice-based encryption schemes//Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques. Cham, Switzerland, 2020:3-33

[23] D'Anvers J P, Batsleer S. Multitarget decryption failure attacks and their application to Saber and Kyber//Proceedings of the IACR International Conference on Public-Key Cryptography. Cham, Switzerland, 2022:3-33

[24] Albrecht M R, Player R, Scott S. On the concrete hardness of learning with errors. Journal of Mathematical Cryptology, 2015,9(3):169-203

[25] Chen Y, Nguyen P Q. BKZ 2.0: Better lattice security estimates//Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security. Berlin, Germany, 2011:1-20

[26] Hanrot G, Pujol X, Stehlé D. Terminating BKZ. Cryptology ePrint Archive, 2011

[27] Laarhoven T. Sieving for shortest vectors in lattices using angular locality-sensitive hashing//Proceedings of the Annual Cryptology Conference. Berlin, Germany, 2015:3-22

[28] Becker A, Ducas L, Gama N, et al. New directions in nearest neighbor searching with applications to lattice sieving//Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms. Arlington, USA, 2016:10-24

[29] Laarhoven T, Mosca M, Van De Pol J. Finding shortest lattice vectors faster using quantum search. Designs, Codes and Cryptography, 2015,77(2):375-400

[30] Laarhoven T. Search problems in cryptography: From fingerprinting to lattice sieving. Cryptology ePrint Archive, 2016

[31] Schnorr C. P. Lattice reduction by random sampling and birthday methods//Proceedings of the Annual Symposium on Theoretical Aspects of Computer Science. Berlin, Germany, 2003:145-156

[32] Chen Y. Lattice reduction and concrete security of fully homomorphic encryption. École Normale Supérieure, Paris, France, 2013

[33] D'Anvers J P, Vercauteren F, Verbauwhede I. The impact of error dependencies on Ring/Mod-LWE/LWR based schemes//Proceedings of the International Conference on Post-Quantum Cryptography. Chongqing, China, 2019:103-115

[34] Cai T, Fan J, Jiang T. Distributions of angles in random packing on spheres. The Journal of Machine Learning Research, 2013,14(1):1837-1864

[35] Zhang J, Yu Y, Fan S, et al. Tweaking the asymmetry of asymmetric-key cryptography on lattices: KEMs and signatures of smaller sizes//Proceedings of the IACR International Conference on Public-Key Cryptography. Cham, Switzerland, 2020:37-65

[36] Applebaum B, Cash D, Peikert C, et al. Fast cryptographic primitives and circular-secure encryption based on hard learning problems//Proceedings of the Annual International Cryptology Conference. Berlin, Germany, 2009:595-618

[37] Beckwith L, Abdulgadir A, Azarderakhsh R. A flexible shared hardware accelerator for NIST-recommended algorithms Crystals-Kyber and Crystals-Dilithium with SCA protection//Proceedings of Cryptographers' Track at the RSA Conference. Cham, Switzerland, 2023:469-490

[38] Miteloudi K, Bos J W, Bronchain O, et al. PQ.V.ALU.E: Post-quantum RISC-V custom ALU extensions on Dilithium and Kyber//Proceedings of the International Conference on Smart Card Research and Advanced Applications. Cham, Switzerland, 2023:190-209

[39] Mandal S, Roy D B. KID: A hardware design framework targeting unified NTT multiplication for Crystals-Kyber and Crystals-Dilithium on FPGA//Proceedings of the 2024 37th International Conference on VLSI Design and 2024 23rd In-

ternational Conference on Embedded Systems (VLSID). Kolkata, India, 2024:455-460

[40] Billingsley P. Probability and measure. Hoboken, USA: John Wiley & Sons, 2017

[41] Jiang H, Zhang Z, Chen L, et al. IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited//Proceedings of the Annual International Cryptology Conference. Cham, Switzerland, 2018:96-125

SHAO Ming-Yao, Ph. D. candidate. His main research interests focus on lattice-based post quantum cryptography.

LIU Yue-Jun, Ph. D., associate professor. Her main research interests include cryptographic engineering, and post-quantum cryptography.

ZHOU Yong-Bin, Ph. D., professor. His main research interests include cryptography, cryptographic engineering, and post-quantum cryptography.

SHAO Yan, Ph. D. candidate. His main research interests focus on lattice-based post quantum cryptography

ZHANG Qian, Ph. D., assistant research fellow. Her main research interests include side-channel attacks and countermeasures, as well as cryptographic hardware design and implementation.

Background

Random distribution plays a central role in cryptography. In the LWE setting, security and performance hinge on the distribution of secrets and errors. Early schemes mostly adopted Gaussian noise, but its sampling cost and side-channel surface motivated alternatives—especially the Centered Binomial dDistribution (CBD) and the Uniform Distribution (UD). Prior works proved that, when errors are uniformly drawn from a small interval, LWE remains hard; practical KEMs such as KINDI, Kyber and Saber therefore employ CBD or UD.

However, the systematic impact of distribution choice on KEM security and performance is still underexplored. The uSaber variant showcased only one UD instantiation and lacked a full security-performance analysis. In deployed systems, sampling bias, implementation reuse and optimization may introduce distributional deviations that widen the attack surface, underscoring the need for a structured study.

This work examines how secret/error distributions affect lattice-based KEMs through Kyber and Saber. On security, we quantify the trade-off between lattice-attack resistance and decryption-failure risk. For the same value range, the uniform distribution (UD) has standard deviation $>\sqrt{4/3}$ that of the centered binomial distribution (CBD), making instances harder and raising security by 12-25 bits, but also markedly increasing failure probability. To operate in high-failure regimes, we design an iterative failure localization procedure and a majority vote key recovery algorithm, cutting key-recovery complexity under UD to $O(2^{37})$ for Kyber512 and $O(2^{42})$ for LightSaber, far below $O(2^{146})$ and $O(2^{128})$ with CBD. On performance, with modulus and compression fixed, the distribution choice leaves key/ciphertext sizes unchanged; however, UD requires rejection sampling, increasing sampling time by 25%—65%. Overall, distribution choice imposes a significant yet imbalanced impact across attack resistance, failure risk, and runtime.

To rebalance these factors, we introduce the Asymmetric LWE (ALWE) framework for flexible parameter tuning, and propose uKyber, a UD-based variant. uKyber maintains Kyber's security and failure rates while shortening the Kyber512 ciphertext from 768 to 672 bytes (~12.5%). Using multi value rejection sampling, it also accelerates key generation and encapsulation/decapsulation relative to the baseline.

This work is supported in part by National Natural Science Foundation of China (No. U2336205).

附录 A. 多项式向量的旋转

定义 7.（旋转）对于 $r\in\mathbb{Z}$，定义 R_q^k 中向量 $\boldsymbol{C}$ 的旋转为：$\boldsymbol{C}^r(x):=x^r\cdot\boldsymbol{C}(x^{-1})\bmod x^n+1$。相应地，$\overline{\boldsymbol{C}^r}\in\mathbb{Z}_q^{kn}$ 表示其系数向量。

$\boldsymbol{C}^r$ 的构造确保对于 $r\in[0,1,\cdots,n-1]$，$\boldsymbol{S}^T\boldsymbol{C}$ 的第 r 个坐标由标量积 $\bar{\boldsymbol{S}}^T\overline{\boldsymbol{C}^r}$ 给出。换句话说，可以将 $\boldsymbol{S}^T\boldsymbol{C}$ 分解为标量积之和：$\boldsymbol{S}^T\boldsymbol{C}:=\sum_{r=0}^{n-1}\bar{\boldsymbol{S}}^T\overline{\boldsymbol{C}^r}\cdot x^r$。具体在示例 1 中来说明这一点。

示例 1. 对于 $\mathbb{Z}_q^{2\times1}[x]/(x^3+1)$ 中的多项式向量 $\boldsymbol{S}$ 和 $\boldsymbol{C}$：

$$\boldsymbol{S}=\begin{bmatrix}s_{0,0}+s_{0,1}x+s_{0,2}x^2\\ s_{1,0}+s_{1,1}x+s_{1,2}x^2\end{bmatrix},\boldsymbol{C}=\begin{bmatrix}c_{0,0}+c_{0,1}x+c_{0,2}x^2\\ c_{1,0}+c_{1,1}x+c_{1,2}x^2\end{bmatrix},$$

得到以下系数向量：

$$\bar{\boldsymbol{S}}=\begin{bmatrix}s_{0,0}\\ s_{0,1}\\ s_{0,2}\\ s_{1,0}\\ s_{1,1}\\ s_{1,2}\end{bmatrix},\overline{\boldsymbol{C}^0}=\begin{bmatrix}c_{0,0}\\ -c_{0,2}\\ -c_{0,1}\\ c_{1,0}\\ -c_{1,2}\\ -c_{1,1}\end{bmatrix},\overline{\boldsymbol{C}^1}=\begin{bmatrix}c_{0,1}\\ c_{0,0}\\ -c_{0,2}\\ c_{1,1}\\ c_{1,0}\\ -c_{1,2}\end{bmatrix},\overline{\boldsymbol{C}^2}=\begin{bmatrix}c_{0,2}\\ c_{0,1}\\ c_{0,0}\\ c_{1,2}\\ c_{1,1}\\ c_{1,0}\end{bmatrix}。$$

因此，两个多项式向量的乘积为 $\boldsymbol{S}^T\boldsymbol{C}:=\sum_{r=0}^{2}\bar{\boldsymbol{S}}^T\overline{\boldsymbol{C}^r}\cdot x^r$。

附录 B. 概率论

定理 2.（贝叶斯定理）贝叶斯定理在数学上表述为以下方程：

$$P(A\mid B)=\frac{P(B\mid A)P(A)}{P(B)}\tag{8}$$

定理 3.（中心极限定理，CLT）假设 $\{X_1,X_2,\cdots,X_n\}$ 是一列独立同分布的随机变量，满足 $\mathbb{E}[X_i]=\mu$ 和 $\mathrm{Var}[X_i]=\sigma^2$，设 $\overline{X_n}=(X_1+X_2+\cdots+X_n)/n$。当 n 趋于无穷大时，随机变量 $\sqrt{n}(\overline{X_n}-\mu)$ 的分布收敛为正态分布 $\mathcal{N}(0,\sigma^2)$：$\sqrt{n}(\overline{X_n}-\mu)\to\mathcal{N}(0,\sigma^2)$。

更一般地，X_i 不必须同分布。假设 $X_1,X_2,\cdots,X_n$ 是一列独立的随机变量，每个变量具有有限的期望值 μ_i 和方差 σ_i^2。定义 $s_n^2=\sum_{i=1}^{n}\sigma_i^2$。如果满足 Lyapunov 条件[40]，那么 $(X_i-\mu_i)/s_n$ 的和在 n 趋于无穷大时在分布上收敛到标准正态分布：

$$1/S_n\sum_{i=1}^{n}(X_i-\mu_i)\to N(0,1)。\tag{9}$$

附录 C. uKyber 的方案构造

本文首先提出了 CPA 安全的 PKE 方案，然后通过 FO 变换转化为 CCA 安全的 KEM 方案。具体方案构造如下：

（1）uKyber.CPAPKE：设 $n,q,k,u_s,u_e,u_r,u_{e_1},u_{e_2},d_u,d_v$ 为正整数，可扩展输出函数 Sam 作为随机谕言机。uKyber.CPAPKE 包含以下三个算法：

①KeyGen()：随机选取种子 $\rho,\zeta\leftarrow\{0,1\}^n$，并采样 $(\boldsymbol{s},\boldsymbol{e})\sim\mathcal{U}_{u_s}^k\times\mathcal{U}_{u_e}^k=\mathrm{Sam}(\zeta)$。生成矩阵 $\boldsymbol{A}=\mathrm{Sam}(\rho)\in R_q^{k\times k}$，计算 $\boldsymbol{t}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e}$。输出公钥 $pk=(\boldsymbol{t},\rho)$ 和私钥 $sk=\boldsymbol{s}$。

②Enc(pk,m)：给定公钥 pk 和明文 m，随机选择 $\varphi\leftarrow\{0,1\}^n$，并采样 $(\boldsymbol{r},\boldsymbol{e}_1,e_2)\sim\mathcal{U}_{u_r}^k\times\mathcal{U}_{u_{e1}}^k\times\mathcal{U}_{u_{e2}}=\mathrm{Sam}(\varphi)$。计算 $\boldsymbol{u}=\boldsymbol{A}^T\boldsymbol{r}+\boldsymbol{e}_1$，$v=\boldsymbol{t}^T\boldsymbol{r}+e_2+\lceil q/2\cdot m\rceil$，密文 $\boldsymbol{c}=(\boldsymbol{c}_1,c_2)$，其中 $\boldsymbol{c}_1=\mathrm{Compress}_q(\boldsymbol{u},d_u)$，$c_2=\mathrm{Compress}_q(v,d_v)$）。

③Dec($sk,\boldsymbol{c}$)：给定私钥 $sk=\boldsymbol{s}$ 和密文 $\boldsymbol{c}=(\boldsymbol{c}_1,c_2)$，解压缩得 $\boldsymbol{u}'=\mathrm{Decompress}_q(\boldsymbol{c}_1,d_u)$，$v'=\mathrm{Decompress}_q(c_2,d_v)$，并计算 $m'=\mathrm{Compress}_q(v'-\boldsymbol{s}^T\boldsymbol{u}',1)$。

（2）uKyber.CCAKEM：令 $\mathrm{H}:\mathcal{B}^*\to\mathcal{B}^{32}$ 和 $\mathrm{G}:\mathcal{B}^*\to\mathcal{B}^{32}\times\mathcal{B}^{32}$ 为两个作为随机谕言机的哈希函数。应用 FO 变换，CPA 安全的 uKyber.CPAPKE 可转变为 CCA 安全的 uKyber.CCAKEM：

①uKyber.CCAKEM.KeyGen()：随机选择 $z\leftarrow\mathcal{B}^{32}$，并计算公私钥对 $(pk,sk')=$ uKyber.CPAPKE.KeyGen()。返回公钥 pk 和密钥 $sk=(sk'\mid pk\mid\mathrm{H}(pk)\mid z)$。

②uKyber.CCAKEM.Encap(pk)：给定公钥 pk，消息 $m\leftarrow\{0,1\}^{256}$，计算 $(\bar{K},\varphi)=\mathrm{G}(m\mid\mathrm{H}(pk))$，$\boldsymbol{c}=$ uKyber.CPAPKE.Enc(pk,m,φ)。最后，生成封装密钥 $K=\mathrm{KDF}(\bar{K}\mid\mathrm{H}(\boldsymbol{c}))$，并返回密文 $\boldsymbol{c}$ 和封装密钥 K。

③uKyber.CCAKEM.Decap($\boldsymbol{c},sk$)：给定私钥 $sk=(sk'\mid pk\mid\mathrm{H}(pk)\mid z)$ 密文 $\boldsymbol{c}$，计算 $m'=$ uKyber.CPAPKE.Dec(sk',c)，$(\overline{K'},\varphi')=\mathrm{G}(m'\mid h)$，重加密计算 $c':=$ uKyber.CPAPKE.Enc(pk,m',φ')。如果 $c'=c$，返回 $K:=\mathrm{KDF}(\overline{K'}\mid\mathrm{H}(c))$，否则返回 $K:=\mathrm{KDF}(z\mid\mathrm{H}(c))$。

附录 D. uKyber 的安全性证明

定义 8.（AMLWE 问题）令 n,q,k,l 为正整数，α_1,α_2 为正实数，$\mathrm{AMLWE}_{n,q,k,l,\alpha_1,\alpha_2}$ 要求从样本 $(\boldsymbol{A},\boldsymbol{b}=\boldsymbol{A}\boldsymbol{s}+\boldsymbol{e})\in R_q^{k\times l}\times R_q^k$ 中恢复 $\boldsymbol{s}$，其中 $\boldsymbol{A}\leftarrow R_q^{k\times l}$，$\boldsymbol{s}\leftarrow(\chi_{\alpha_1}^n)^l$，$\boldsymbol{e}\leftarrow(\chi_{\alpha_2}^m)^k$。

定义 9.（AMLWE 假设）对于适当选取的参数 $n,q,k,l,\alpha_1,\alpha_2$，不存在（量子）多项式时间的敌手

能够解决 $\mathrm{AMLWE}_{n,q,k,l,a_1,a_2}$ 问题。

在 AMLWE 假设下,uKyber.CPAPKE 是可证明 CPA 安全的,uKyber.CCAKEM 是可证明 CCA 安全的。

(1)uKyber.CPAPKE 的 CPA 安全性

定理 4. 假设 Sam 是随机谕言机。如果 $\mathrm{AMLWE}_{n,q,k,l,u_1,u_2}$ 是困难的,那么 uKyber.CPAPKE 方案是 CPA 安全的。接下来,我们将通过实验 $\boldsymbol{G}_0$,$\boldsymbol{G}_1$,$\boldsymbol{G}_2$ 来证明$\epsilon_A<negl()$,从而完成定理 4 的证明。

实验 $\boldsymbol{G}_0$:实验 $\boldsymbol{G}_0$ 是真实的 CPA 安全实验。该实验中,敌手$\mathcal{A}$可以访问随机谕言机 Sam,并获得公钥 $pk=(\boldsymbol{t},\rho)$。对手 $\mathcal{A}$ 选择两个明文 m_0,m_1,然后获得关于消息 m_b的挑战密文 $\boldsymbol{c}=(\boldsymbol{c}_1,c_2)$,其中 b 是一个随机位。最后,$\mathcal{A}$ 输出一个位 $b'\in\{0,1\}$ 作为对 $b\in\{0,1\}$ 的猜测。

实验 $\boldsymbol{G}_1$:该实验与 $\boldsymbol{G}_0$相同,除了随机选择 pk。如果存在一个 PPT 对手 $\mathcal{A}$ 能够区分 $\boldsymbol{G}_1$和 $\boldsymbol{G}_0$,那么我们可以构造一个 PPT 算法 $\mathcal{B}$ 来解决 $\mathrm{AMLWE}_{n,q,k,l,u_1,u_2}$问题。具体来说,给定 AMLWE 问题的一个实例 $(\boldsymbol{A},\boldsymbol{t})$,$\mathcal{B}$旨在判断 $(\boldsymbol{A},\boldsymbol{t})$是否从 $R_q^{k\times k}\times R_q^k$的均匀分布中采样。形式上,$\mathcal{B}$ 的行为与实验 $\boldsymbol{G}_0$完全相同,除了它选择一个随机 $\rho\leftarrow\{0,1\}^n$,随机谕言机 Sam 使得 $\mathrm{Sam}(\rho)=\boldsymbol{A}$,并返回 $pk=(t,\rho)$给$\mathcal{A}$。由于 Sam 是随机谕言机,$\mathrm{Sam}(\rho)$已定义的概率是可忽略的。如果 $(\boldsymbol{A},\boldsymbol{t})$ 是均匀随机的,那么 $\mathcal{B}$ 的行为如实验 $\boldsymbol{G}_1$。否则,$\mathcal{B}$ 的行为如实验 $\boldsymbol{G}_0$。换句话说,如果 $\mathcal{A}$ 能够以非可忽略的概率区分 实验 $\boldsymbol{G}_0$和 实验 $\boldsymbol{G}_1$,那么$\mathcal{B}$能够以非可忽略的概率解决 AMLWE 问题。

实验 $\boldsymbol{G}_2$:该实验与实验 $\boldsymbol{G}_1$相同,除了使用均匀随机值替换挑战密文生成中使用的 $\boldsymbol{u}=\boldsymbol{A}^T\boldsymbol{r}+\boldsymbol{e}_1$ 和 $v=\boldsymbol{t}^T\boldsymbol{r}+e_2$。如果存在 PPT 对手 $\mathcal{A}$能够区分实验 $\boldsymbol{G}_2$和实验 $\boldsymbol{G}_1$,那么可以构造 PPT 算法 $\mathcal{B}$ 来解决 $\mathrm{AMLWE}_{n,q,k+1,l,u_1,u_2}$问题。具体来说,给定 AMLWE 问题的一个实例 $(\boldsymbol{A},\boldsymbol{t},\boldsymbol{u},v)$,$\mathcal{B}$ 旨在判断 $(\boldsymbol{A},\boldsymbol{t},\boldsymbol{u},v)$是否从 $R_q^{k\times k}\times R_q^k\times R_q^k\times R_q$的均匀分布中采样。形式上,$\mathcal{B}$ 选择一个随机 $\rho\leftarrow\{0,1\}^n$,随机谕言机 Sam 使得 $\mathrm{Sam}(\rho)=\boldsymbol{A}$,并返回 $pk=(t,\rho)$给 $\mathcal{A}$。在接收到 $\mathcal{A}$提供的两个明文 m_0,m_1后,$\mathcal{B}$ 随机选择 $b\leftarrow\{0,1\}$,计算 $\boldsymbol{c}_1=\mathrm{Compress}_q(\boldsymbol{u},d_u)$,$c_2=\mathrm{Compress}_q(v+\lceil q/2\rceil\cdot m_b,d_v)$,然后将密文 $\boldsymbol{c}=(\boldsymbol{c}_1,c_2)$ 提供给 $\mathcal{A}$。如果 $(\boldsymbol{A},\boldsymbol{t},\boldsymbol{u},v)$ 在 $R_q^{k\times k}\times R_q^k\times R_q^k\times R_q$中是均匀随机的,那么 $\mathcal{B}$ 模拟如实验 $\boldsymbol{G}_2$。否则,$\mathcal{B}$ 模拟如实验 $\boldsymbol{G}_1$。因此,如果 $\mathcal{A}$能够以非可忽略的概率区分 $\boldsymbol{G}_2$和 $\boldsymbol{G}_1$,那么 $\mathcal{B}$能够解决 AMLWE 问题。

在 $\boldsymbol{G}_2$中,挑战密文中的 m_b被均匀随机的 v 完美隐藏。因此,$\mathcal{A}$ 在 $\boldsymbol{G}_2$中的优势为 0,这完成了定理 4 的证明。

(2)uKyber.CCAKEM 的 CCA 安全性

uKyber.CCAKEM 是通过对 PKE 方案 uKyber.CPAPKE 应用 FO 变换[29-30]获得,根据 FO 变换构造和定理 4 的结果,我们有以下定理。

定理 5. 假设 Sam、H 和 G 是随机谕言机。如果 $\mathrm{AMLWE}_{(n,q,k,l,u_1,u_2)}$是困难的,那么 uKyber.CCAKEM 方案是 CCA 安全的。

注意,解封装算法即使在检查失败时(即隐式拒绝)也会返回一个随机解封装密钥。此外,文献[41]表明,如果底层 PKE 是 CPA 安全的,那么使用 FO 变换获得的 KEM 在量子随机谕言机模型(QROM)中也是 CCA 安全的。根据 文献[41] 和定理 5 的结果,我们有以下定理。

定理 6. 假设 Sam、H 和 G 是随机谕言机。如果 $\mathrm{AMLWE}_{n,q,k,l,u_1,u_2}$ 是困难的,那么 uKyber.CCAKEM 方案在量子随机谕言机模型中是 CCA 安全的。