软件学报 ISSN 1000-9825, CODEN RUXUEW
2026,37(1):398−424 [doi: 10.13328/j.cnki.jos.007437] [CSTR: 32375.14.jos.007437]

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

格上困难问题量子求解算法综述*

曹金政[1], 罗向阳[1], 陈晓峰[2], 程庆丰[1]

[1](信息工程大学 网络空间安全学院, 河南 郑州 450001)

[2](西安电子科技大学 网络与信息安全学院, 陕西 西安 710071)

通信作者: 程庆丰, E-mail: qingfengc2008@sina.com

摘　要: 随着基于格的后量子密码体制快速发展, 格上困难问题求解算法已成为评估后量子密码方案安全性的关键技术. 当前, 经典计算模型下已存在枚举、筛法、格基约化等格上困难问题求解算法, 同时量子筛法、量子枚举等格上困难问题量子求解算法正逐步引起关注. 围绕后量子密码研究中涉及的格上困难问题, 对格上困难问题量子求解算法给出综述. 首先, 分类整了格上困难问题量子求解算法研究现状. 其次, 梳理各类格上困难问题量子求解算法的设计思路和应用的量子计算技术, 并总结各类格上困难问题量子求解算法的复杂度. 最后, 展望格上困难问题量子求解算法的未来发展趋势.

关键词: 格公钥密码; 格上困难问题; 量子算法

中图法分类号: TP309

中文引用格式: 曹金政, 罗向阳, 陈晓峰, 程庆丰. 格上困难问题量子求解算法综述. 软件学报, 2026, 37(1): 398–424. http://www.jos.org.cn/1000-9825/7437.htm

英文引用格式: Cao JZ, Luo XY, Chen XF, Cheng QF. Survey on Quantum Algorithms for Solving Hard Problems in Lattice. Ruan Jian Xue Bao/Journal of Software, 2026, 37(1): 398–424 (in Chinese). http://www.jos.org.cn/1000-9825/7437.htm

Survey on Quantum Algorithms for Solving Hard Problems in Lattice

CAO Jin-Zheng[1], LUO Xiang-Yang[1], CHEN Xiao-Feng[2], CHENG Qing-Feng[1]

[1](School of Cyberspace Security, Information Engineering University, Zhengzhou 450001, China)

[2](School of Cyber Engineering, Xidian University, Xi'an 710071, China)

Abstract: With the rapid development of Lattice-based post-quantum cryptography, algorithms for hard problems in Lattices have become an essential tool for evaluating the security of post-quantum cryptographic schemes. Algorithms such as enumeration, sieve, and Lattice basis reduction have been developed under the classical computing model, while quantum algorithms for solving hard problems in Lattices, such as quantum sieve and quantum enumeration, are gradually attracting attention. Although Lattice problems possess post-quantum properties, techniques such as quantum search can accelerate a range of Lattice algorithms. Given the challenges involved in solving hard problems in Lattices, this study first summarizes and analyzes the research status of quantum algorithms for such problems and organizes their design principles. Then, the quantum computing techniques applied in these algorithms are introduced, followed by an analysis and comparison of their computational complexities. Finally, potential future developments and research directions for quantum algorithms addressing Lattice-based hard problems are discussed.

Key words: Lattice-based public key cryptography; hard problem in Lattice; quantum algorithm

为应对量子计算对经典密码的安全威胁[1−5], 基于格上困难问题设计的后量子格密码凭借抵抗量子计算的特性成为研究热点[6−8]. 美国国家标准与技术研究院经过 2012–2022 年的多轮后量子密码算法征集与遴选后, 于 2023 年 8 月发布了 3 种后量子公钥密码方案, 其中 2 种方案是基于格设计的[9−11], 足以体现格密码的重要地位, 格

* 基金项目: 国家自然科学基金 (62472438, 62172433, 62172435); 国家重点研发计划 (2022YFB3102900); 河南省自然科学基金 (242300421414)
收稿时间: 2024-01-18; 修改时间: 2024-04-07; 采用时间: 2025-03-28; jos 在线出版时间: 2025-07-30
CNKI 网络首发时间: 2025-07-31

上困难问题求解算法作为评估后量子格密码安全性的手段也日益受到重视[12]. 已有的格上困难问题经典求解算法针对最短向量问题 (shortest vector problem, SVP)、错误学习 (learning with errors, LWE) 等问题的求解积累了丰富成果[13−15]. 但随着量子计算机的快速发展, 量子计算技术直接应用成为可能, 采用量子搜索等加速技术[16,17]的格上困难问题量子求解算法逐渐兴起, 受到国内外研究者广泛关注[18−30]. 格上困难问题量子求解算法中最常用的量子加速技术是 Grover 搜索、量子碰撞搜索、量子傅里叶变换等. 这些量子加速方法与格上困难问题求解的基本原理相结合, 发展出一系列量子算法, 如图 1 所示.

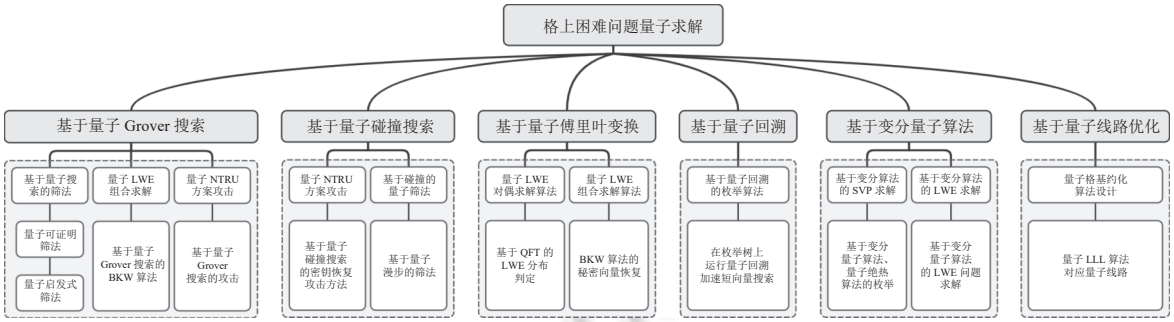

图 1　格上困难问题量子求解算法基本技术

2004 年, Regev 等人[31]最早对 SVP 的量子求解做了分析, 将其转化为二点问题, 进而归约至量子计算中二面体群隐含子群问题, 证明了其在量子计算中复杂性仍为亚指数级的. 此后, 研究者提出了一系列 SVP 量子求解算法, 可分类为量子筛法、量子枚举、量子格基约化等. 图 2 给出了格上困难问题量子求解算法主要类型的发展概况.

后量子格密码研究领域中, 格上困难问题求解思路主要分为两类: (1) 将问题归约后使用 SVP 求解算法[32−40], (2) 针对问题的特定结构设计专用算法[41−47]. 基于这两种求解思路, 可将格上困难问题量子求解算法同样分为基于 SVP 求解的通用算法和针对特定格问题的专用算法.

(1) 量子 SVP 求解算法. 大多数格上困难问题都可以归约为 SVP, 进而利用 SVP 求解算法解决, 这也是求解格上困难问题最基本的思想. 量子计算模型下 SVP 求解算法主要包括量子筛法、量子枚举、量子格基约化. 量子筛法方面, 2013 年, Laarhoven 等人[48]提出使用 Grover 量子搜索加速筛法中向量组对约减的步骤, 进而对 AKS 筛法、列表筛法等启发式筛法实现了平方级次加速[49]. 2016 年, Becker 等人提出了正轴体筛法[50]和 LD 筛法[51]. 2019 年, Kirshanova 等人[52]研究了量子计算下 k 列表问题的求解方法, 并将其应用于元组筛法, 给出了查询复杂度的分析. Albrecht 等人[53]对量子筛法中多种关键组成部分进行了量子实现和复杂度分析, 如高维球的临近向量算法、过滤搜索、向量和的重量计数等. 2021 年, Chailloux 等人[54]在量子筛法的基础上进一步扩展, 使用量子漫步替代量子筛法中的 Grover 量子搜索步骤, 并利用局部敏感过滤加速. 2023 年, Bonnetain 等人[55]基于量子漫步给出了同时求多组碰撞的算法, 并应用在量子筛法中; Chailloux 等人[56]结合量子加速与编码技术加快了元组筛法的运行. 量子枚举方面, Grover 量子搜索在枚举过程中难以适用. 为了构造量子枚举, 主要有两种设计思路. 一种是依托经典枚举的搜索树结构, 在枚举过程中应用量子回溯算法, 对 T 个节点的搜索树, 时间复杂度约为 $\sqrt{T}$, 该方法由 Aono 等人[57]提出, 并由 Bai 等人[58]给出了复杂度分析. 2023 年, Bindel 等人[59]进一步降低了该方法的复杂度上界. 另一种量子枚举方法由 Albrecht 等人[60]提出, 主要思路是将 SVP 通过编码转化为哈密顿算符的基态, 并利用变分量子算法求解. 量子格基约化方面, LLL 是最基础、应用最广的约化算法. 2019 年, Tiepelt 等人[61]首次将 LLL 算法的流程通过量子线路进行实现, 并利用该方法构造了对梅森素数密码系统的攻击.

(2) 特殊问题量子求解算法. 主要包括对于 LWE、NTRU、CVP 等特定的格上困难问题的求解算法, 以及带结构的特殊格上困难问题求解算法. 这些特殊问题是多种后量子公钥密码标准的安全性基础, 其量子求解算法具有重要研究价值. LWE 方面, 2018 年, Esser 等人[41]基于 c-求和方法提出一种基于组合的 LWE 求解算法, 并利用 Grover 量子搜索算法对其主要操作 (向量组合查找) 实现量子加速, 相比经典计算下的时间复杂度达到平方加速.

2022 年, Albrecht 等人[42]改进了 LWE 的对偶格攻击, 提出算法中可以使用量子傅里叶变换替代经典算法. Liu 等人[43]针对 LWE 问题的特性对组合算法做了进一步优化, 使算法的应用条件得到放松. 2024 年, Chen 等人[44]基于带有复高斯窗口的窗口量子傅里叶变换, 将 LWE 转换为具有纯虚高斯振幅的量子态. 进而提出针对特定参数 LWE 的多项式级量子求解算法. NTRU 方面, 2015 年, Fluhrer[45]首次提出基于 Grover 量子搜索的 NTRU 求解, 主要思想是根据密钥多项式的生成结构设计搜索算法. 2019 年, Laaji 等人[46]针对三元 NTRU 提出了两种基于 Grover 量子搜索的量子算法, 目标分别是恢复密钥与密文. 2021 年, 董经等人[47]提出利用 Claw-Finding 量子碰撞算法替代 Grover 量子搜索算法进行密钥多项式搜索, 可进一步降低 NTRU 量子密钥恢复的复杂度. CVP 方面, 2016 年, Eldar 等人[62]利用系统范式设计了特殊参数的 CVP 量子求解算法. 2018 年, Aono 等人[57]使用改进的量子枚举算法直接进行问题求解, 并探讨了一系列枚举剪枝技术的量子加速. Eldar 等人[63]、Allen 等人[64]提出并分析了基于问题归约的最差情况问题量子求解算法. 特殊结构格方面, 2017 年, Cramer 等人[65]由实际密码方案的攻击出发, 提出主理想格上 SVP 的量子求解算法, 并通过经典归约扩展至任意的理想格. 此后, Cramer 等人[66]、Ducas 等人[67]、Boer 等人[68]从算法输出结果、量子比特复杂度等方面对理想格上 SVP 量子求解算法进行了分析与提高.

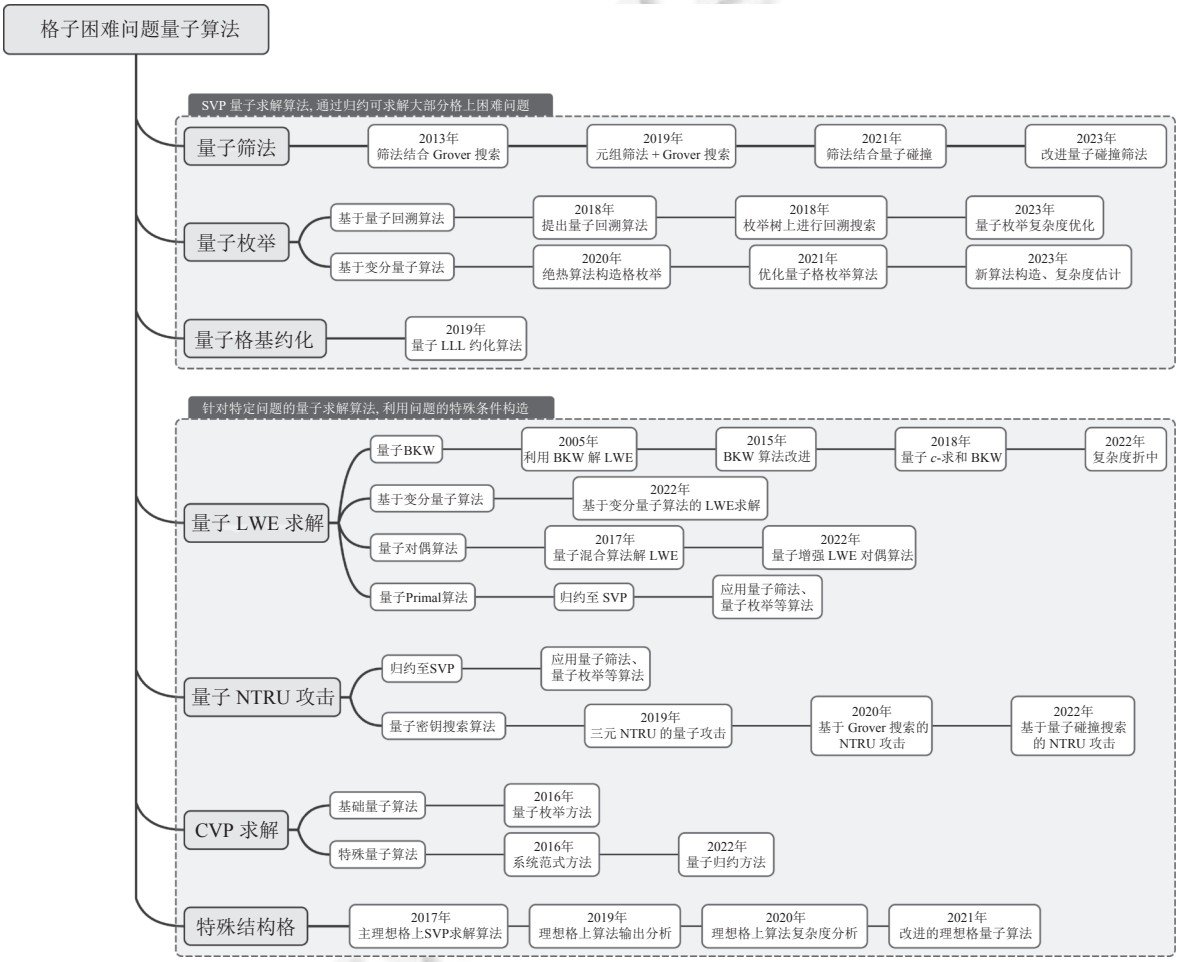

图 2 格上困难问题量子求解算法发展概况

本文立足于后量子格公钥密码的安全性评估, 对格上困难问题量子求解算法的相关研究进行回顾和梳理. 首先, 归纳格上困难问题量子求解算法分类. 其次, 介绍格上困难问题量子求解算法设计中应用到的量子加速技术,

并讨论格上困难问题量子求解算法的复杂度. 最后, 展望未来可以继续探索的问题. 本文第 1 节介绍格上困难问题量子求解算法相关的基础知识, 包括格理论和格问题相关定义、量子计算相关背景知识等. 第 2–4 节介绍量子筛法、量子枚举、量子格基约化这 3 类 SVP 的量子求解算法, 基于算法实例梳理了格上困难问题求解算法由经典计算模型向量子计算模型迁移的基本设计思想, 并对其设计方法、复杂度估计进行讨论. 其中, 第 2 节介绍量子筛法, 包括量子可证明筛法和量子启发式筛法. 第 3 节介绍量子枚举算法, 包括基于回溯的枚举和基于变分量子算法的枚举. 第 4 节介绍量子格基约化. 第 5–8 节介绍针对特殊格上困难问题的量子求解算法, 通过考察特定格问题的结构, 结合量子计算特性进行加速的算法设计方法, 探讨了多种量子计算技术对问题复杂度的影响. 其中, 第 5 节介绍 LWE 问题的量子求解算法. 第 6 节介绍对 NTRU 公钥加密体制的量子攻击方法. 第 7 节介绍最近向量问题的量子求解算法发展现状. 第 8 节介绍特殊结构格上困难问题量子求解算法. 第 9 节对全文进行总结, 并对未来可以继续探索的问题进行展望.

1 预备知识

本节介绍本文中用到的基础知识, 首先说明格的基本定义与主要的格上困难问题, 如 SVP、 CVP 等, 随后简介量子计算与量子算法的基本知识.

1.1 格和格上困难问题

本节介绍格的基本知识, 包括格的定义、主要性质和格上困难问题. 更多关于格的知识可参考文献 [6].

定义 1. 格. R^m 上 n 个线性无关向量 $\boldsymbol{b}_1,\boldsymbol{b}_2,\ldots,\boldsymbol{b}_n\in R^m$ 的整数组合得到的集合, 称为 R^m 上的格 $\mathcal{L}$, 表示为:

$$\mathcal{L}(\boldsymbol{b}_1,\boldsymbol{b}_2,\ldots,\boldsymbol{b}_n)=\left\{\sum_{i=1}^{n}x_i\boldsymbol{b}_i : x_i\in Z\right\},$$

其中, 向量 $\boldsymbol{b}_1,\boldsymbol{b}_2,\ldots,\boldsymbol{b}_n\in R^m$ 为一组格基, n 为格的秩, m 为格的维度. 当 m=n, 称格为满秩的. 将基向量为行向量排列为格基矩阵 $\boldsymbol{B}$. 对给定的 $\boldsymbol{B}$, 定义 $\mathcal{P}(\boldsymbol{B})=\left\{\sum_{i=1}^{n}\lambda_i\boldsymbol{b}_i, 0\leqslant\lambda_i<1\right\}$.

定义 2. 长度. 令向量 $\boldsymbol{b}=[b_1,b_2,\ldots,b_n]\in\mathcal{L}$, 称 $\|\boldsymbol{b}\|=(b_1^2+b_2^2+\ldots+b_n^2)^{1/2}$ 为向量 $\boldsymbol{b}$ 的欧几里得范数, 又称为向量 $\boldsymbol{b}$ 的长度. 在求解格上困难问题时通常考虑向量的长度.

定义 3. 行列式. 设 $\boldsymbol{b}_1,\boldsymbol{b}_2,\ldots,\boldsymbol{b}_n\in R^m$ 为格 $\mathcal{L}$ 的一组基, $\boldsymbol{b}_1^*,\boldsymbol{b}_2^*,\ldots,\boldsymbol{b}_n^*$ 为 $\boldsymbol{b}_1,\boldsymbol{b}_2,\ldots,\boldsymbol{b}_n$ 进行施密特正交化后所得的向量, 则称 $\det(\mathcal{L})=\Pi_{i=1}^{n}\|\boldsymbol{b}_i^*\|$ 为格 $\mathcal{L}$ 的行列式或体积. 格的行列式仅与格本身有关, 与格中基的选取无关.

定义 4. 最短向量问题. 给定格 $\mathcal{L}$, 最短向量问题要求寻找最短的非零格向量, 即满足 $\|\boldsymbol{v}\|=\min\{\|\boldsymbol{b}\|:\ \boldsymbol{b}\in\mathcal{L}\backslash\{0\}\}$ 的向量 $\boldsymbol{v}$. 最短向量长度表示为 $\lambda_1(\mathcal{L})$.

定义 5. 最近向量问题. 给定格 $\mathcal{L}$ 和目标向量 $\boldsymbol{t}$, 最近向量问题要求寻找格向量 $\boldsymbol{v}$, 满足 $\forall\boldsymbol{b}\in\mathcal{L}$, $\|\boldsymbol{v}-\boldsymbol{t}\|\leqslant\|\boldsymbol{b}-\boldsymbol{t}\|$.

定义 6. 错误学习问题. 假设 n, q 是正整数, χ 是整数上的一个概率分布. $\boldsymbol{s}$ 是 $Z_q^{n\times1}$ 上的一个秘密向量. 已知给定矩阵 $\boldsymbol{A}\in Z_q^{m\times n}$ 和根据 χ 采样得到的噪音向量 $\boldsymbol{e}\in Z_q^{m\times1}$. 给定 $(\boldsymbol{A},\boldsymbol{b})=(\boldsymbol{A},<\boldsymbol{A},\boldsymbol{s}>+\boldsymbol{e})\in Z_q^n\times Z_q$, LWE 问题需要求出 $\boldsymbol{s}$, 满足 $\boldsymbol{b}_i=<\boldsymbol{A}_i,\boldsymbol{s}>+\boldsymbol{e}_i$, $i=1,\ldots,m$.

定义 7. NTRU 问题. 假设 n 是正整数, q 是素数, p 是一个小整数. 私钥多项式 $f(x)$ 和生成多项式 $g(x)$ 是未知量, f, g 的系数为 0 或±1, 公钥多项式 $h(x)=pf_q^{-1}(x)\cdot g(x)\bmod q$ 是已知量. NTRU 问题要求由给定的 h 恢复 f 和 g.

1.2 量子计算的基本知识

本节介绍量子算法的基本概念, 主要参考文献 [69]. 量子计算的基本单元是量子比特, 与经典计算的二进制比特对应, 表示为 $|0\rangle$ 和 $|1\rangle$, 其向量形式为:

$$|0\rangle=\begin{bmatrix}0\\1\end{bmatrix},\ |1\rangle=\begin{bmatrix}1\\0\end{bmatrix}.$$

两个基态 $|0\rangle$ 和 $|1\rangle$ 构成二维希尔伯特复空间的基, 其线性组合称为叠加态, 表示为 $|\psi\rangle=\alpha|0\rangle+\beta|1\rangle$, 其中复数 α、β 满足 $|\alpha|^2+|\beta|^2=1$. 使用量子系统的张量积 $|\psi_1\rangle\otimes|\psi_2\rangle$ 作为其复合系统 $|\psi_1\psi_2\rangle$. 量子计算通过各种量子门电路构造各种功能. 一个量子门对应一个酉矩阵 U, 实现特定的计算功能. 常见的单量子比特门包括 Hadamard 门、

Pauli-X 门、Pauli-Y 门、Pauli-Z 门、Phase 门等. 另一类特殊的门当一个量子比特满足特定条件时, 才对另一个量子比特执行操作, 称为受控量子门. 对于受控 U 门, 控制位记为 c, 目标位记为 t. 量子门执行 $|c\rangle|t\rangle \mapsto |c\rangle U|t\rangle$, 即当 c 为 $|1\rangle$ 时, 对 t 执行 U 操作, 对其他情况不执行操作. 格上困难问题量子求解算法中最常见的组成部分是 Grover 量子搜索算法. 给定大小为 N 的无序列表 L, 其中目标元素个数为 M, 引入函数 $f(x):\{0,1\}^n \to \{0,1\}$, 表示为从 n 比特的元素地址到 0 和 1 上的映射. 若地址 x 上的数据 d_x 是一个目标元素, 有 $f(x)$=1, 否则 $f(x)$=0. Grover 量子搜索算法以 $O(\sqrt{N/M})$ 的复杂度找到使 $f(x)$=1 的 x, 即找到 L 中一个目标元素. Grover 量子搜索算法中 $f(x)$ 的计算通过量子预言实现, 包括两部分: 一部分是将数据 d_x 通过索引地址 x 读出来, 这一部分常称为 LOAD; 另一部分是通过判断数据 d_x 是否为目标, 若是目标则将目标量子态翻转, 否则量子态保持不变. Grover 量子搜索算法首先准备量子态, 之后进行多次目标态翻转和关于平均值翻转的迭代. 经过一次迭代, 目标态的幅值增大. 使用 Grover 量子搜索迭代足够次数, 目标态的幅值可以比较接近 1. 随后进行测量, 算法可以接近 1 的概率得到目标态, 即搜索到了目标.

量子傅里叶变换是另一种重要的量子算法, 可用在多种组合类算法中. 经典的离散傅里叶变换将时域中 N 维向量 $(x_1,\ldots,x_N)$ 映射到频域中的 N 维向量 $(y_1,\ldots,y_N)$, 其中 $y_k = \frac{1}{\sqrt{N}}\sum_{j=0}^{N-1} x_j e^{\frac{2\pi i jk}{N}}$, $i^2=-1$. 类似地, 量子傅里叶变换作用在量子态上可实现类似映射. 任意 N 维向量 $(x_1,\ldots,x_N)$ 在量子环境下表示为量子态 $\sum_{j=0}^{N-1} x_j|j\rangle$. 经过量子傅里叶变换后, 得到新的量子态为 $\sum_{j=0}^{N-1} x_j|j\rangle \mapsto \sum_{k=0}^{N-1} y_k|k\rangle$. 用计算基表示, 有:

$$|j\rangle \mapsto \frac{1}{\sqrt{N}}\sum_{j=0}^{N-1} e^{\frac{2\pi i jk}{N}}|k\rangle = \frac{(|k\rangle + e^{2\pi i 0.j_n}|k\rangle)(|k\rangle + e^{2\pi i 0.j_{n-1}j_n}|k\rangle)\ldots(|k\rangle + e^{2\pi i 0.j_1\ldots j_n}|k\rangle)}{\sqrt{N}},$$

其中, $j_1 j_2 \ldots j_n$ 是量子态 j 的二进制表示, $0.j_1\ldots j_n = j_1/2 + j_2/2^2 + \ldots + j_n/2^n$ 是小数的二进制形式. 经典傅里叶变换的时间复杂度为 $O(N2^N)$, 而量子傅里叶变换的时间复杂度 $O(N^2)$.

2 量子筛法

本节从算法分类、量子加速原理、算法复杂度估计等方面对量子筛法进行介绍. 量子筛法的算法主要结构与已有经典筛法相近, 其主要的量子加速手段是用量子搜索替代经典筛法中的格向量筛取、短向量搜索等关键步骤[70]. 最早的可证明筛法是 Ajtai 等人[34]提出的 AKS 筛法, 算法的主要思想可以总结为: 1) 生成指数级的格向量; 2) 进行多轮筛选, 直到获得足够多的格上短向量; 3) 将上述向量两两相减, 得到格上最短向量. 研究者后续在 AKS 筛法基础上提出了 NV 筛法[71,72]、层次筛法[73,74]等启发式筛法. 另一类可证明筛法是 Micciancio 等人[35]提出的列表筛法, 该算法定义一个空列表作为初始列表, 随后不断抽取格向量并使用已经在列表中的向量对其进行约化, 再添加至列表中, 重复此过程直至找到格中最短向量. 后续的研究者参考列表筛法提出了元组筛法[75,76]、BDGL 筛法[75]等启发式筛法.

量子计算模型下, 筛法的研究整体表现出从直观到深入, 从交叉融合到结构创新的发展态势. 相比枚举、格基约化等格上困难问题求解算法, 筛法的量子化较早受到研究者关注, 产生了较多的量子筛法研究成果, 如图 3 所示. 从筛法算法基本思想与量子加速相结合的方式观察, 量子筛法的整体发展可分 3 个阶段.

第 1 阶段, 量子筛法开始由经典算法向量子算法迁移. 主要设计思路遵循已有经典筛法结构, 将向量搜索步骤替换为 Grover 量子搜索算法, 其设计方法较为直观. 2013 年, Laarhoven 等人[48]首先提出使用 Grover 量子搜索加速筛法, 在指数层面降低了时间复杂度. 该阶段量子筛法的研究主要聚焦于已有经典筛法向量子计算模型的迁移, 通过原有经典算法结构与量子搜索的交叉产生一系列基于 Grover 量子搜索的量子筛法.

第 2 阶段, 在基于 Grover 的量子算法基础上, 开始对筛法本身的设计思路与算法结构进行调整, 设计更加适合量子计算模型的量子筛法. 2019 年, Kirshanova 等人[52]提出了新的筛法模型 k-结构问题, 最终利用 Grover 搜索算法求解, 并据此设计对应的量子筛法, 得到了优化的时间存储效率. 该阶段量子筛法突破了已有筛法的常用结构, 开始针对量子搜索的性质对筛法进行改造, 更好发挥量子加速的作用.

第 3 阶段, 随着量子算法的发展, 研究者开始探索多种途径设计量子筛法. 2021 年, Chailloux 等人[54]将 k-结构问题进一步分解, 将其转化为碰撞搜索问题, 进而使用基于量子漫步的碰撞搜索算法给出了新的量子筛法复杂度. Bonnetain 等人[55]进一步提出了量子多碰撞搜索技术, 通过一次量子漫步同时求出多个碰撞. 该阶段量子筛法的加速原理已不局限于 Grover 量子搜索, 而开始探索更多样化的量子算法设计方式.

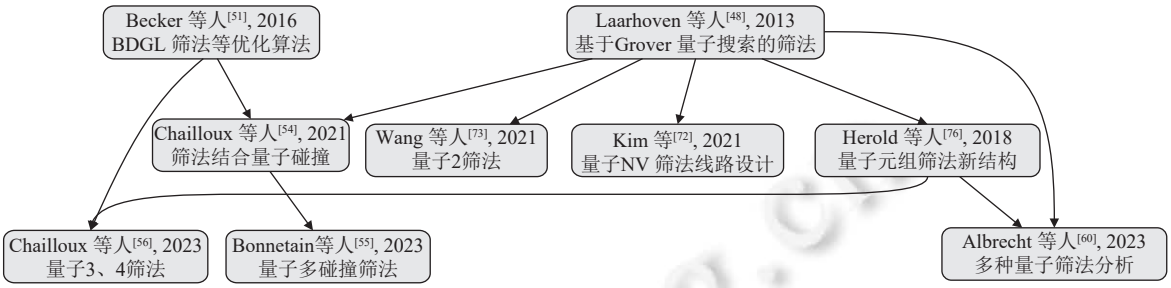

图 3　量子筛法主要研究成果发展过程

从量子筛法与经典算法结构对比观察, 量子筛法与经典筛法具有较清晰的对应关系, 可参照经典筛法将量子筛法分类为量子可证明筛法与量子启发式筛法, 如图 4 所示. 本节按照对量子筛法的分类, 依次介绍量子可证明筛法与量子启发式筛法的对应量子算法.

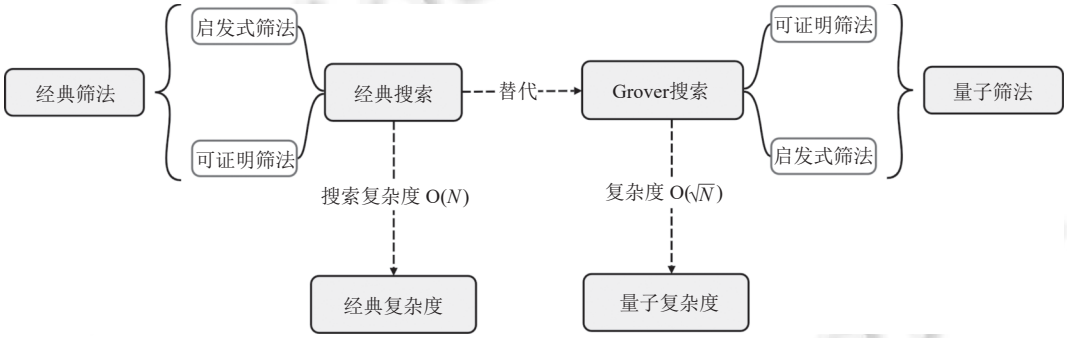

图 4　量子筛法设计与分析模型

2.1　量子可证明筛法

对量子可证明筛法的研究起始于 Laarhoven 等人提出的多种量子筛法[48,49], 包括列表筛法和 AKS 筛法等可证明筛法的量子版本, 这两种筛法提出时间最早, 结构简洁, 其设计思路也为后续一系列启发式筛法提供了基本的设计模式. 本节主要介绍以量子列表筛法为代表的量子可证明筛法设计思路. 列表筛法的主要过程如算法 1 所示, 其经典复杂度如下: 生成列表 L 需要时间 $\mathrm{O}(N_1\cdot|L|)=2^{(c_g+2c_t)n+\mathrm{o}(n)}$, 生成列表 S 需要时间 $\mathrm{O}(N_2\cdot|L|)=2^{(c_g+c_b/2+c_t)n+\mathrm{o}(n)}$, 在 S 中搜索向量对需要时间 $\mathrm{O}(|S|^2)=2^{(2c_g+c_b)n+\mathrm{o}(n)}$, 其中 c_g, c_b, c_t 是算法参数, 算法整体的时间复杂度为 $2^{2.465n+\mathrm{o}(n)}$. 在文献 [48,49] 中, $\mathrm{o}(n)$ 表示上界小于 n, $\mathrm{O}(n)$ 表示上界小于等于 n. 为了实现对列表筛法的量子加速, Laarhoven 等人[48,49]在算法的 6、14、19 步使用 Grover 量子搜索替代经典搜索, 达到了与经典算法中相同的效果, 即穷举搜索得到满足一定条件的短向量.

算法 1. 列表筛法.

输入: 基矩阵 $\boldsymbol{B}$, N_1, N_2;
输出: 格 $\mathcal{L}(\boldsymbol{B})$ 的一个短向量.

1. 随机采样整数 $N_1' \in [0, N_1]$
2. 初始化空列表 L, S
3. FOR $i \leftarrow 1, \ldots, N_1'$
4. 采样扰动向量 $\boldsymbol{e} \leftarrow \boldsymbol{B}_n(0, \xi\mu)$
5. 转化向量 $\boldsymbol{v}' \leftarrow \boldsymbol{e} \bmod \mathcal{P}(\boldsymbol{B})$
6. WHILE $\boldsymbol{w} \leftarrow \text{search}\{\boldsymbol{w} \in L : \|\boldsymbol{v}' \pm \boldsymbol{w}\| < \gamma \|\boldsymbol{v}'\|\}$
7. 用 $\boldsymbol{w}$ 约化 $\boldsymbol{v}'$
8. 减去扰动向量 $\boldsymbol{e}$: $\boldsymbol{v} \leftarrow \boldsymbol{v}' - \boldsymbol{e}$
9. IF $\|\boldsymbol{v}\| > R\mu$
10. 将格向量 $\boldsymbol{v}$ 加入列表 L
11. FOR $i \leftarrow 1, \ldots, N_2'$
12. 采样扰动向量 $\boldsymbol{e} \leftarrow \boldsymbol{B}_n(0, \xi\mu)$
13. 转化向量 $\boldsymbol{v}' \leftarrow \boldsymbol{e} \bmod \mathcal{P}(\boldsymbol{B})$
14. WHILE $\boldsymbol{w} \leftarrow \text{search}\{\boldsymbol{w} \in L : \|\boldsymbol{v}' \pm \boldsymbol{w}\| < \gamma \|\boldsymbol{v}'\|\}$
15. 用 $\boldsymbol{w}$ 约化 $\boldsymbol{v}'$
16. 减去扰动向量 $\boldsymbol{e}$: $\boldsymbol{v} \leftarrow \boldsymbol{v}' - \boldsymbol{e}$
17. IF $\|\boldsymbol{v}\| > R\mu$
18. 将格向量 $\boldsymbol{v}$ 加入列表 S
19. $(\boldsymbol{s}_1, \boldsymbol{s}_2) \leftarrow \text{search}\{(\boldsymbol{s}_1, \boldsymbol{s}_2) \in S^2 : \|\boldsymbol{s}_1 - \boldsymbol{s}_2\| \leqslant \mu,\ \boldsymbol{s}_1 \neq \boldsymbol{s}_2\}$
20. 输出 $\boldsymbol{s}_1 - \boldsymbol{s}_2$

下面以列表筛法第 19 行搜索近似向量对 $(\boldsymbol{s}_1, \boldsymbol{s}_2) \in S^2$ 的操作为例, 讨论如何利用量子计算实现向量搜索. 假设 S 中已存储足够多的短向量, 计算列表中所有向量对的叠加:

$$\frac{1}{\sqrt{N_2}}\sum_{\boldsymbol{s}_1 \in S}|\boldsymbol{s}_1\rangle \otimes \frac{1}{\sqrt{N_2}}\sum_{\boldsymbol{s}_2 \in S}|\boldsymbol{s}_2\rangle \mapsto \frac{1}{N_2}\sum_{\boldsymbol{s}_1, \boldsymbol{s}_2 \in S}|\boldsymbol{s}_1, \boldsymbol{s}_2\rangle.$$

该步骤起到了经典算法中遍历所有向量对的作用. 下一步, 计算向量差 $\boldsymbol{s}_1 - \boldsymbol{s}_2$ 的长度并存储在辅助寄存器中:

$$\frac{1}{N_2}\sum_{\boldsymbol{s}_1, \boldsymbol{s}_2 \in S}|\boldsymbol{s}_1, \boldsymbol{s}_2\rangle|0\rangle_{\text{aux}} \mapsto \frac{1}{N_2}\sum_{\boldsymbol{s}_1, \boldsymbol{s}_2 \in S}|\boldsymbol{s}_1, \boldsymbol{s}_2, \|\boldsymbol{s}_1 - \boldsymbol{s}_2\|\rangle.$$

接下来, 使用 Grover 量子搜索算法找到满足 $\|\boldsymbol{s}_1 - \boldsymbol{s}_2\| \leqslant \mu$ 的向量对. 该算法实际上对满足 $\|\boldsymbol{s}_1 - \boldsymbol{s}_2\| \leqslant \mu$ 的量子态 $|\boldsymbol{s}_1, \boldsymbol{s}_2, \|\boldsymbol{s}_1 - \boldsymbol{s}_2\|\rangle$ 进行了幅值放大, 能以高概率得到如下叠加态:

$$\frac{1}{N_2}\sum_{\boldsymbol{s}_1, \boldsymbol{s}_2 \in S}|\boldsymbol{s}_1, \boldsymbol{s}_2, \|\boldsymbol{s}_1 - \boldsymbol{s}_2\|\rangle \mapsto \frac{1}{\sqrt{N_2}}\sum_{\boldsymbol{s}_1, \boldsymbol{s}_2 \in S, \|\boldsymbol{s}_1 - \boldsymbol{s}_2\| \leqslant \mu}|\boldsymbol{s}_1, \boldsymbol{s}_2, \|\boldsymbol{s}_1 - \boldsymbol{v}_2\|\rangle.$$

最后, 通过测量得到一对 $(\boldsymbol{s}_1, \boldsymbol{s}_2)$ 满足 $\|\boldsymbol{s}_1 - \boldsymbol{s}_2\| \leqslant \mu$. 在不改变筛法基本运行流程的前提下, 量子可证明筛法将第 6、14、19 行中向量搜索的时间复杂度分别降低至 $\mathrm{O}(N_1 \cdot \sqrt{|L|}) = 2^{(c_g + 3/(2c_t))n + \mathrm{o}(n)}$、$\mathrm{O}(N_2 \cdot \sqrt{|L|}) = 2^{(c_g + c_b/2 + c_t/2)n + \mathrm{o}(n)}$、$\mathrm{O}(|S|) = 2^{(c_g + c_b/2)n + \mathrm{o}(n)}$. 综合以上结果, 可得量子列表筛法的整体时间复杂度为 $2^{1.799n + \mathrm{o}(n)}$.

AKS 筛法[35,77]也可使用类似设计思路进行量子加速. AKS 筛法的主要步骤与列表筛法有所区别, 其筛选操作主要通过 sieving 子过程实现, 但该子过程同样需要在向量列表中搜索符合条件的格向量, 故可以使用 Grover 量子搜索算法替代经典的穷举搜索. 经典模型下, 一轮 sieving 子过程的时间复杂度为 $2^{(c_0 + c_s)n + \mathrm{o}(n)}$, 其中 c_0 与 c_s 是算法参数, 而 Grover 量子搜索使其复杂度降低至 $2^{(c_0 + c_s/2)n + \mathrm{o}(n)}$. 与列表筛法类似, 在 AKS 筛法最后也需要搜索向量对 $(\boldsymbol{s}_1, \boldsymbol{s}_2)$ 使其向量差最短, 该步骤复杂度可以由经典算法的 $2^{(2c_R + c_u)n + \mathrm{o}(n)}$ 降低至量子算法的 $2^{(3c_R/2 + c_u)n + \mathrm{o}(n)}$. 其中, c_R 与 c_u 是算法参数. 经典 AKS 筛法的时间复杂度为 $2^{3.398n + \mathrm{o}(n)}$, 存储复杂度为 $2^{1.985n + \mathrm{o}(n)}$. 综合上述两个部分的改进, 算法的整体复杂度可由 $2^{3.398n + \mathrm{o}(n)}$ 降低至 $2^{2.672n + \mathrm{o}(n)}$.

总体而言, 量子可证明筛法作为量子加速算法在格上困难问题求解中的典型应用, 是最早被关注和研究的格上困难问题量子求解算法的类别. 在该类算法提出之后, 其设计思路可以广泛地启发其他类别的格算法. 在此基础上, 研究者设计了量子启发式筛法等结构类似的量子筛法.

2.2 量子启发式筛法

AKS 筛法、列表筛法等可证明筛法使用了随机扰动技术, 增大了算法的实现难度. 实际的格问题求解中常使用由可证明筛法发展而来的各种启发式筛法, 如 NV 筛法 (NV sieve)、高斯筛法 (Gauss sieve)、元组筛法 (tuple sieve) 等. 启发式筛法在可证明筛法的基础上取消扰动向量并引入启发式假设, 虽然正确性的理论证明弱于可证明筛法, 但是时间复杂度显著降低. 在多种量子启发式筛法中, 元组筛法应用了多种量子加速技术, 产生了较多研究成果. 本节主要以元组筛法为例介绍量子启发式筛法的设计思路与发展过程.

2.2.1 基于 Grover 搜索的量子启发式筛法

元组筛法的核心思想是将 k 个向量进行组合, 得到一个新的短向量, 其特殊形式是 2 筛法, 即组合两个向量得到新的短向量. 假设 2 筛法过程中 S 集合储存了足够多长度不超过 R 的格向量, 并要求查找符合条件 $\|\boldsymbol{u}-\boldsymbol{v}\|\leqslant\gamma R$ 的 $\boldsymbol{u},\boldsymbol{v}$ 向量, 这可以通过量子搜索算法实现. 首先计算列表中所有向量对的叠加:

$$\frac{1}{\sqrt{N}}\sum_{\boldsymbol{u}\in S,\|\boldsymbol{u}\|\leqslant R}|\boldsymbol{u}\rangle\otimes\frac{1}{\sqrt{N}}\sum_{\boldsymbol{v}\in S,\|\boldsymbol{v}\|\leqslant R}|\boldsymbol{v}\rangle=\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S}|\boldsymbol{u},\boldsymbol{v}\rangle.$$

对应经典算法中遍历所有向量对的操作. 接下来计算 $\|\boldsymbol{u}-\boldsymbol{v}\|$:

$$\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S}|\boldsymbol{u},\boldsymbol{v}\rangle|0\rangle_{\text{aux}}\mapsto\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S}|\boldsymbol{u},\boldsymbol{v},\|\boldsymbol{u}-\boldsymbol{v}\|\rangle.$$

之后, 使用 Grover 量子搜索, 可获得一个满足向量差长度小于 γR 的向量对:

$$\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S}|\boldsymbol{u},\boldsymbol{v},\|\boldsymbol{u}-\boldsymbol{v}\|\rangle\mapsto\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S,\|\boldsymbol{u}-\boldsymbol{v}\|\leqslant\gamma R}|\boldsymbol{u},\boldsymbol{v},\|\boldsymbol{u}_1-\boldsymbol{v}\|\rangle+\left|\phi_{\text{junk}}\right\rangle.$$

通过测量可以去除 $\left|\phi_{\text{junk}}\right\rangle$, 下一步使用酉变换 U_{diff} 计算 $\boldsymbol{u}-\boldsymbol{v}$ 并存储在第 1 个寄存器中, 将新向量命名为 v_1:

$$\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S,\|\boldsymbol{u}-\boldsymbol{v}\|\leqslant\gamma R}U_{\text{diff}}|\boldsymbol{u},\boldsymbol{v}\rangle\mapsto\frac{1}{N}\sum_{\boldsymbol{u},\boldsymbol{v}\in S,\|\boldsymbol{u}-\boldsymbol{v}\|\leqslant\gamma R}|\boldsymbol{u}-\boldsymbol{v},\boldsymbol{v}\rangle=\frac{1}{N}\sum_{\boldsymbol{v}_1=\boldsymbol{u}-\boldsymbol{v}}U_{\text{diff}}|\boldsymbol{v}_1,\boldsymbol{v}\rangle.$$

此时, 量子 2 筛法已经将向量的长度上界由 R 降至 γR. 递归运行量子 2 筛法可以进一步缩短向量, 形成一个二叉树结构. 完整的量子 2 筛法运行 $t=poly(n)$ 次迭代需要使用 2^t 个叠加态. 该方法的设计思路与量子可证明筛法类似, 都是将经典算法中的搜索步骤替换为 Grover 搜索. 这种设计思路的优点是算法原理较为直观, 也可以充分利用量子搜索的平方加速性质.

2.2.2 基于 k 结构问题的量子启发式筛法

2 筛法要求找到满足 $\|\boldsymbol{u}\pm\boldsymbol{v}\|\leqslant\gamma R$ 的向量对 $\boldsymbol{u},\boldsymbol{v}$, 而一般形式的元组筛法需要对确定的整数 $k\geqslant 2$ 找到满足 $\|\boldsymbol{u}_1\pm\boldsymbol{u}_2\pm\ldots\pm\boldsymbol{u}_k\|\leqslant\gamma R$ 的向量组 $\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_k$. 除了直接利用 Grover 搜索算法设计量子元组筛法, Kirshanova 等人[52]提出了另一种量子元组筛法的设计思路, 将 k 个向量的组合抽象为近似 k 列表问题, 即给定目标长度 t 和相同指数级别大小的 k 个列表 $L_1,L_2,\ldots,L_k$, 找到 k 元组 $(\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_k)\in L_1\times L_2\times\ldots\times L_k$ 满足 $\|\boldsymbol{u}_1+\boldsymbol{u}_2+\ldots+\boldsymbol{u}_k\|\leqslant t$. Kirshanova 等人进一步将近似 k 列表问题转化为 k 结构问题: 给定 k 个列表 $L_1,L_2,\ldots,L_k$, Gram 矩阵 C (称为结构) 和 $\epsilon>0$, 求 $(\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_k)\in L_1\times L_2\times\ldots\times L_k$ 满足 $<\boldsymbol{u}_i,\boldsymbol{u}_j-C_{i,j}><\epsilon$. 量子元组筛法使用以下方法求解 k 结构问题. 首先, 抽取一个 $\boldsymbol{u}_1\in L_1$, 并使用 Grover 量子搜索在剩余 $k-1$ 个列表中找到满足 $<\boldsymbol{u}_1,\boldsymbol{u}_i-C_{1,i}><\epsilon$, $i\in[2,k]$ 的元素 $\boldsymbol{u}_i$. 其次, 抽取 $\boldsymbol{u}_2$, 用类似方法搜索得到满足 $<\boldsymbol{u}_2,\boldsymbol{u}_i-C_{1,i}><\epsilon$, $i\in[3,k]$ 的元素 $\boldsymbol{u}_i$. 最后, 迭代以上步骤, 直至 i=k. 2023 年, Chailloux 等人[56]在文献 [52] 基础上结合编码技术的改进, 简化了归约所得的 k 结构问题, 进一步提高了量子元组筛法效率.

上述基于直接使用 Grover 搜索的方法和转化至 k 列表问题的方法, 区别在于: (1) 设计产生的算法结构不同; (2) 解决的优化方向不同: 基于 Grover 搜索的方法通过直接替换穷搜索步骤, 意图达到最快的速度, 而基于 k 列表问题的方法通过调整归约过程的参数, 可以实现算法时间与存储的折中. 考虑到影响筛法效率的一大因素是消耗

存储资源过大, 因此需要考虑具体求解的问题规模和掌握的计算资源选取转化至 k 列表问题的量子筛法或直接使用 Grover 搜索的量子筛法.

2.2.3 基于碰撞搜索的量子启发式筛法

以上介绍的多种量子筛法, 虽然结构不同, 但其基本的设计思路都是将经典搜索转化至量子 Grover 搜索. 事实上, 该方法应用至元组筛法等算法结构仍有一些不足: (1) 对于元组筛法中的 k 列表问题, 需要求出 k 个系数 $(\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_k)$, 而 Grover 搜索每次运行只能取得一个系数 $\boldsymbol{u}_i$, 降低了问题求解的效率; (2) 筛法中的某些步骤要求得到满足相同条件的大量短向量, 而 Grover 搜索在列表中只能搜索得到一个满足条件的元素. 为解决这些问题, 应考虑采用新的量子加速技术应用至量子筛法.

2021 年, Chailloux 等人[54]在筛法中使用量子碰撞问题替代 Grover 搜索. 令 $f:\{0,1\}^n\to\{0,1\}^m$, $n\leqslant m\leqslant 2n$ 是随机映射, 量子碰撞问题要求找到此函数的一个碰撞. 该问题的构造相比穷举搜索文献可以将搜索位数折半, 更适合在元组筛法等筛法版本中应用. 文献 [54] 通过量子漫步构造了量子碰撞求解算法, 复杂度约为 $2^{m/3}$. 2023 年, Bonnetain 等人[55]改进了量子漫步流程, 通过链式漫步算法使量子元组筛法能在同一次漫步过程中求多组碰撞而不需要反复进行碰撞搜索, 该算法的性质结合元组筛法的结构, 降低了筛法时间复杂度. 除了以上介绍的 2 筛法和元组筛法, 一系列启发式筛法均可利用 Grover 量子搜索算法进行加速. 由于量子筛法的主要结构与经典算法一致, 故主要的经典筛法都有对应的量子版本. 一些量子启发式筛法的复杂度如表 1 所示.

表 1 部分量子启发式筛法的复杂度指数

筛法类型	空间	经典时间	量子时间
高斯筛法[35]	0.208	0.415	0.311
哈希筛法[48]	0.208	0.337	0.286
球筛法[49]	0.208	0.292	0.268
正轴体筛法[50]	0.208	0.292	0.268
LD筛法[51,54]	0.208	0.292	0.257
元组3筛法[52,56]	0.189	0.566	0.311
NV筛法[71]	0.208	0.415	0.311
两层筛法[73]	0.208	0.415	0.311
三层筛法[74]	0.208	0.415	0.311
元组2筛法[75]	0.208	0.415	0.311

总结以上量子筛法研究成果, 可将其基本思路和设计模型分为 3 类. (1) 在经典筛法的算法结构基础上, 直接将向量搜索环节替换为 Grover 量子搜索, 这也是最初始的量子筛法设计方法; (2) 为了针对性加速 k 元组筛法, 将其中 k 个向量组合得短向量的环节转化为 k 结构问题, 进而得到新的时间存储折中; (3) 将向量组合转化为函数碰撞的搜索, 以此将搜索空间维数折半, 更进一步使用量子多碰撞算法快速得到一系列碰撞. 上述设计模型的分析比较见表 2. 量子筛法的下一步发展方向, 一是继续针对算法特殊结构采用创新的量子加速技术; 二是将已构造的量子加速技术应用至更多的筛法结构或其他算法结构中.

表 2 量子筛法设计模型

文献	算法设计类型	应用的量子加速技术	加速环节	加速效果	模型优势	模型劣势
Laarhoven等人[48,49]	直接替换经典穷搜索构造筛法	每步单次运行Grover量子搜索算法	单个短向量搜索	相比经典搜索平方级加速	通用性较强, 可适用于各种筛法类型	加速效果不充分
Kirshanova等人[52] Albrecht等人[53]	转化至k结构问题构造筛法	反复运行Grover量子搜索算法	元组筛法中k列表问题	调整k改变时间/空间折中	结合筛法特殊结构, 针对性加速	需要特殊的筛法结构, 通用性不强
Chailloux等人[54] Bonnetain等人[55]	基于量子碰撞构造筛法	量子漫步算法	元组筛法中k列表问题	相比经典搜索立方级加速	采用新的量子加速技术, 降低搜索复杂度	

3 量子枚举算法

本节介绍量子枚举算法的设计思想、算法结构与复杂度估计. 量子枚举的发展脉络如图 5 所示. 枚举是与筛法并列的 SVP 精确求解算法, 也是求解多种上困难格问题的重要方法. 经典模型下的枚举算法考虑以原点为中心的特定球中包含的向量. 由于任意的格中向量 $\boldsymbol{u}\in\mathcal{L}$ 均可分解为格基向量 $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ 的组合 $\boldsymbol{u}=\sum_i u_i\boldsymbol{b}_i$, 故枚举算法通过系数 $(u_1,u_2,\ldots,u_n)$ 表示格向量并构造枚举树.

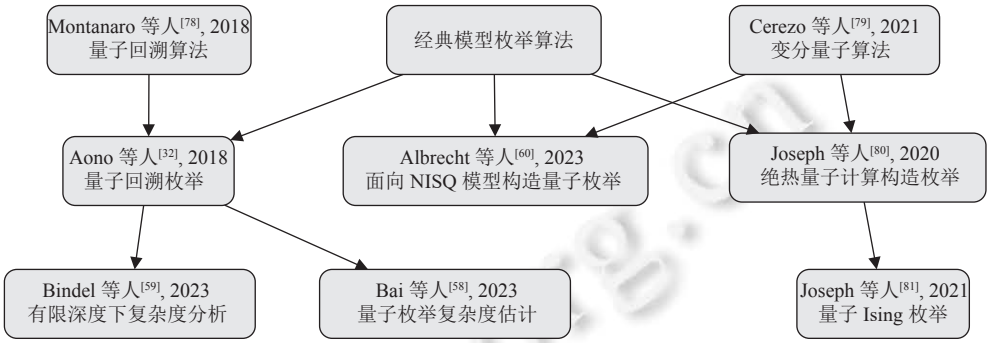

图 5 量子枚举算法发展概况

相比筛法多样化的量子版本设计模式, 枚举的算法结构决定了其与量子计算的结合更加困难. 枚举算法在球中所有的 $(u_1,u_2,\ldots,u_n)$ 组合中从右至左进行深度优先搜索, 查找所有满足存在 $(u'_1,u'_2,\ldots,u'_{n-1},u_n)$ 的参数 u_n. 找到所有可能的 u_n 之后, 再针对每个 u_n 搜索对应的 u_{n-1}, 进而以此类推搜索至 u_{n-1}. 可以发现枚举算法对系数进行顺序搜索[49], 前一个 u_i 的值引出多个可能的 u_{i-1} 值, 搜索空间随算法运行不断变化; 然而 Grover 量子搜索要求搜索集合是提前已知且可编号的, 故枚举算法中无法直接调用量子搜索进行优化. 虽然 Grover 量子搜索无法直接应用于枚举算法, 但研究者通过考察枚举算法结构和量子物理和量子计算模型设计了多种量子枚举. 总体而言, 量子枚举的设计思路有: (1) 沿用经典算法的枚举树结构, 利用树结构上量子回溯算法加速目标节点搜索; (2) 将短向量搜索过程转化为量子物理模型中 Hamilton 算子求基态, 并利用对应量子算法进行模拟. 下面分别对两种量子枚举类型进行介绍.

3.1 基于量子回溯的枚举算法

Aono 等人[57]提出了第 1 种设计量子枚举算法的新思路, 其主要应用的量子加速技术是量子回溯算法. 枚举算法构造的枚举树 $\mathcal{T}$ 中上每个节点代表部分候选值 $u_i,u_{i+1},\ldots,u_n$, 其子节点代表扩展信息 $u_1,u_2,\ldots,u_{i-1}$. 回溯算法可以判定树 $\mathcal{T}$ 中是否包含目标向量即标记点, 并输出这样的目标点. 经典回溯算法需要访问预言机 $\#\mathcal{V}(\mathcal{T})$ 次, 即搜索树上节点的个数, 而 Montanaro[78]提出了量子版本的回溯算法: 给定 $\epsilon>0$, 度为 $d(\mathcal{T})=\mathrm{O}(1)$ 的树 $\mathcal{T}$, 预言机 $\mathcal{P}$, 树深度的上界 n, 存在量子回溯算法 $\mathit{FindSolution}(\mathcal{T},\mathcal{P},n,\epsilon)$ 输出 x 使得 $\mathcal{P}(x)$ 为真, 或输出“不存在”. 量子回溯算法进行 $\mathrm{O}(\sqrt{\#\mathcal{V}(\mathcal{T})}n^{3/2}\log(n)\log(1/\epsilon))$ 次对 $\mathcal{T}$ 和 $\mathcal{P}$ 查询, 正确概率至少 $1-\epsilon$, 每次查询使用 $\mathrm{O}(1)$ 次辅助操作, 使用 $poly(n)$ 个量子比特. 基于量子回溯构造枚举的主要步骤总结如下.

(1) 按照经典枚举的方法进行格向量投影和枚举树构造.

(2) 将经典枚举树转化为等价二叉树.

(3) 以新的二叉树为输入, 运行量子回溯算法找到目标节点.

(4) 由该节点恢复目标短向量.

Aono 等人[57]给出了量子枚举算法复杂度的理论估计. Bai 等人[58]给出了枚举量子线路设计, 并对量子枚举的复杂度做了更详细的估计. Bindel 等人[59]进一步讨论了量子计算深度受限情况下量子枚举算法的复杂度. 给定 $\epsilon>0$, 枚举树 $\mathcal{T}$ 的度上界 d, 预言机 $\mathcal{P}$, 向量长度的函数为 g, 此时量子枚举算法可以输出使 g 取最小值的节点 $\mathcal{N}$, 概率为 $1-\epsilon$. 量子枚举算法需要 $\mathrm{O}(\sqrt{T}(n\log d)^{3/2}\log(n\log d)\log(\lceil\log_2 R\rceil/\epsilon)\lceil\log_2 R\rceil)$ 次对 $\mathcal{T}$ 和 $\mathcal{P}$ 的访问, 其中 $T=\#\mathcal{V}(\mathcal{T})$.

每个对 $\mathcal{T}$ 的访问需要 $O(\log d)$ 次辅助操作. 算法需要 $poly(n\log d, \log R)$ 个量子比特. 利用基础量子枚举算法的复杂度可估计各种枚举算法版本的量子复杂度. 例如, 使用量子加速的 Kannan 算法需要 $O(n^{n/(4e)})\cdot poly(\log(n), \log(1/\epsilon), \beta)$ 次预言机访问和 $poly(n,\beta)$ 个量子比特, 其中 β 是格向量元素的比特长度.

经典模型下的枚举算法通常使用剪枝方法对枚举树结构进行修改, 以一定的成功率为代价提高求解速度. 类似地, 基于量子回溯的量子枚举算法也可使用剪枝方法实现加速. 量子圆筒剪枝的时间复杂度是 $O(\sqrt{T}n^3)\cdot poly(\log(n), \log(1/\epsilon))$, 其中 T 是剪枝枚举所搜索的节点个数. 量子离散剪枝的时间复杂度 $O(n^2\sqrt{M})\cdot poly(\log(n), \log(M), \log(1/\epsilon), \beta)$ $poly(n,\beta)$, M 是剪枝过程中使用的参数, 代表最优标签的个数. 量子极限剪枝的时间复杂度 $O(\sqrt{T}n^3\beta)\cdot poly(\log(n), \log(1/\epsilon), \log(\beta), \log(m))$ 和 $O(\sqrt{M})\cdot poly(n, \log(M), \log(1/\epsilon), \beta)$. 具体复杂度如表 3 所示.

表 3 基于量子回溯的量子枚举算法复杂度 (省略对数部分)

枚举	经典时间	量子时间
无剪枝	$O(n^{n/(2e)})$	$O(n^{n/(4e)})$
离散剪枝	$O(n^4M)$	$O(n^2\sqrt{M})$
圆筒剪枝	$O(Tn^6)$	$O(\sqrt{T}n^3)$
极限圆筒剪枝	$O(Tn^6\beta^2)$	$O(\sqrt{T}n^3\beta)$
极限离散剪枝	$O(M)$	$O(\sqrt{M})$

3.2 基于变分量子算法的枚举算法

以上介绍了典型的通过枚举树上量子回溯算法实现的量子枚举算法. 除此之外, 还存在一类基于量子物理模型实现的量子枚举算法, 基本思路是通过编码将 SVP 归约为 Hamiltonian 算子的基态, 并采用变分量子算法 (variational quantum algorithm, VQA)[79]等方法进行搜索. Joseph 等人[80,81]考虑使用绝热算法求解 SVP 中的量子态能量差, 在理论上给出了算法的成功条件. Albrecht 等人[60]考虑了在受限的量子计算 (noisy intermediate scale quantum, NISQ) 模型下实现枚举的量子加速. 基于量子绝热算法的量子枚举算法一般情况下使用的量子比特数为 $O(n\log n)$, 而在特定的格结构上所需的量子比特个数为 $O(3n\log(n/2)+n+\log\det(\mathcal{L}))$. 基于变分量子算法的量子枚举算法的主要步骤如下.

(1) 将最短向量问题编码至 $\mathcal{H}$ 算子的基态;

(2) 对参数 θ, 取一个拟设态 $|\phi(\theta)\rangle$: 一个简单量子线路输出的一系列量子态;

(3) 制备其中一个量子态并测量. 估计 $\langle\phi(\theta)|\mathcal{H}|\phi(\theta)\rangle$ 的期望值, 用以计算新的参数值 θ;

(4) 使用新的 θ 重复以上步骤, 直到得到符合条件的量子态.

考察量子枚举算法相关研究成果, 可以将量子枚举按照图 6 所示分类. 相比更适合利用量子搜索技术的筛法, 枚举的树搜索结构限制了应用 Grover 量子搜索加速的效果. 为了更好地将枚举与量子计算技术进行结合, 研究者探索了在枚举算法中使用量子回溯算法和变分量子算法, 实现了量子加速技术与格算法结构的进一步融合, 为格上困难问题的量子求解算法的后续发展提供了新思路.

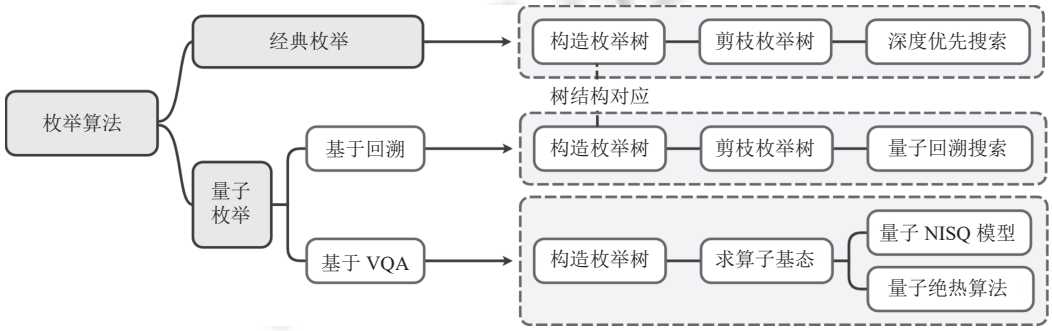

图 6 量子枚举算法分类

4 量子格基约化算法

本节介绍基于量子线路的量子格基约化算法, 包括算法设计原理及算法量子线路结构. 经典格基约化算法不仅是求解格上近似 SVP 的重要手段, 而且在求解多种格上困难问题的过程中发挥关键作用. 实际上, 大多数格问题都可以归约为求近似最短向量, 进而利用格基约化算法求解. 实际求解格问题过程中, 最常用的算法是 LLL 约化算法[39]、BKZ 约化算法[40]及其变形 BKZ 2.0[82], 其中 LLL 是第 1 种实用的格基约化算法, 也是后续各种格基约化算法的结构基础和重要组成部分. LLL 约化算法的一般步骤如算法 2 所示.

算法 2. LLL 算法.

输入: 格基矩阵 $\boldsymbol{B}$, 参数 δ;
输出: LLL 约化基.

1. $i \leftarrow 2$
2. WHILE $i \leqslant n$
3. 计算施密特正交系数 $u_{i,j}$ 和 $u_{i,j}\|\boldsymbol{b}_j^*\|^2$, $j = 1,\ldots,i-1$
4. $\boldsymbol{b}_i = \boldsymbol{b}_i - [u_{i,i-1}]\boldsymbol{b}_{i-1}$, 更新施密特正交化系数
5. IF $\delta\|\boldsymbol{b}_{i-1}^*\|^2 \leqslant \|\boldsymbol{b}_i^*\|^2 + u_{i,i-1}^2\|\boldsymbol{b}_{i-1}^*\|^2$
6. FOR $j \leftarrow i-2$ to 0
7. $\boldsymbol{b}_i = \boldsymbol{b}_i - [u_{i,j}]\boldsymbol{b}_j$, 更新正交化系数
8. $\boldsymbol{v} \leftarrow x$, $N \leftarrow \lceil (N+N')/2 \rceil$
9. ELSE
10. 交换 $\boldsymbol{b}_{i-1}$ 和 $\boldsymbol{b}_i$, 更新正交化系数
11. $i \leftarrow i+1$
12. 输出约化基 $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$

文献 [39] 证明, 设给定的 n 维格基中最大长度为 $\tilde{B}$, 则通过有理数运算实现的 LLL 约化算法复杂度为 $\mathrm{O}(n^6\log^3\tilde{B})$. Schnorr[83]给出了一个可证明的浮点数近似版本 LLL 约化算法, 将正交化系数用 $\mathrm{O}(n+\log\tilde{B})$ 精度表示, 整体复杂度为 $\mathrm{O}(n^4\log\tilde{B}(n+\log\tilde{B})^2)$. Nguên 等人[84]提出了 L^2 约化算法, 复杂度为 $\mathrm{O}(n^5\log\tilde{B}(n+\log\tilde{B}))$. Ryan 等人[85]在 LLL 运行过程中进行浮点数精度调整, 得到新的启发式 LLL 版本, 复杂度为 $\mathrm{O}(n^w(n+\log\tilde{B})^{1+\epsilon})$, $w \in (2,3]$. 近年来, 量子格基约化方面的研究成果集中在量子 LLL 约化算法的量子线路设计. Tiepelt 等人[61]首次提出 LLL 约化算法的量子版本, 并利用其攻击梅森素数密码体制. 量子 LLL 将主要算法操作通过量子线路进行实现, 主要使用的量子寄存器种类如表 4 所示.

表 4 量子 LLL 使用量子寄存器种类

寄存器符号	对应变量
$\|\boldsymbol{B}\rangle$	格基
$\|\boldsymbol{B}^*\rangle$	正交基
$\|L\rangle$	格基中向量长度
$\left\|\boldsymbol{U}^{(i)}\right\rangle$	第i轮的Gram-Schmidt矩阵M
$\|\mathcal{L}\rangle$	单个量子比特, 判定格向量是否符合Lovász条件
$\|I\rangle$	主循环计数
$\|ctl\rangle$	控制比特

量子 LLL 中应用的基本量子线路包括加法 $\mathcal{A}$: $|x,y,0\rangle \mapsto |x,y,x+y\rangle$、乘法 $\mathcal{M}$: $|x,y,0\rangle \mapsto |x,y,xy\rangle$ 和除法 $\mathcal{D}$: $|x,y,0\rangle \mapsto |x,y,x/y\rangle$. 在基本线路之上, Tiepelt 等人[61]为经典 LLL 的循环结构、存储复位、正交化与系数取整、取大值等

步骤设计了对应的量子线路, 如图 7 所示.

量子 LLL 算法中 Size-reduce 过程代表格基向量满足 Lovász 条件时运行的约减步骤, Swap 过程代表不满足时进行的向量交换步骤. 量子 LLL 使用 $r^4 d\log^2\tilde{B}+r^2\log\tilde{B}+\max(d,r)\cdot m$ 个量子比特, 其中 $\tilde{B}$ 是初始基向量长度上界, m 是算法所计算数值的比特长度, r 是格的秩, d 是维度. 文献 [61] 进一步设计了可逆的 LLL 线路, 减小了量子存储的消耗量. 总体上, 格基约化较为依赖经典算法结构, 其对应量子算法的设计也主要集中于各个步骤的量子线路实现. 在 LLL 算法的量子化的基础上, 如何应用各种优化的格基约化算法和量子线路设计技巧[85–88]改进现有的量子格基约化算法, 是未来的潜在研究课题之一.

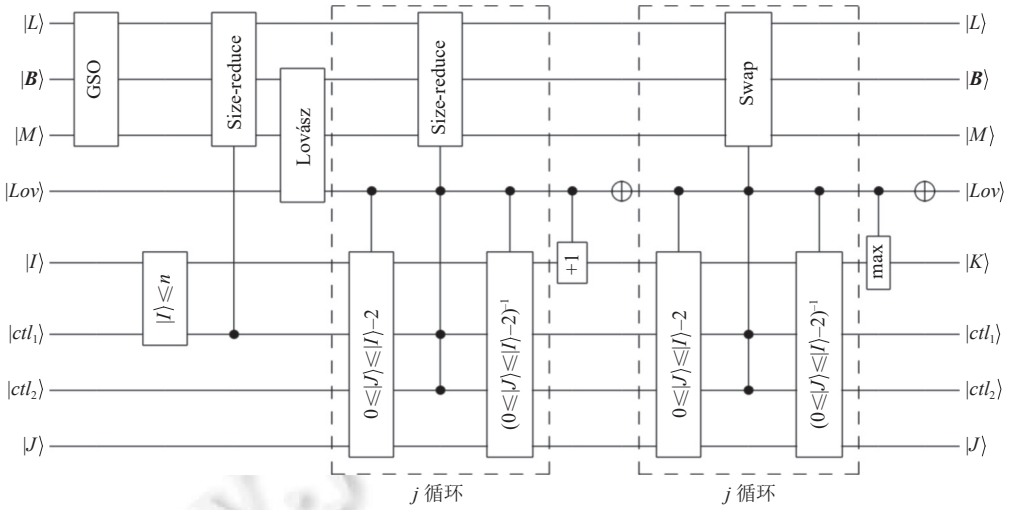

图 7 量子 LLL 算法线路图

格密码研究中的另一种重要格基约化算法是 BKZ 算法, 其流程结合了 LLL 约化算法与局部枚举、筛法等 SVP 求解算法, 在约化效果与时间消耗等方面达到较好平衡. 理论上, BKZ 格基约化算法在量子计算模型下一方面可受益于本节介绍的 LLL 量子化改进, 另一方面可受益于前两节介绍的量子筛法、量子枚举改进. 当前对 SVP 量子求解算法的研究多数关注各环节本身的进一步加速与分析, 对 BKZ 综合多方面量子改进的整体优化与分析尚待深入, 这也是 SVP 量子求解算法领域未来的潜在研究课题之一.

5 LWE 问题的量子算法

使用以上介绍的量子筛法、量子枚举、量子格基约化这 3 类算法可以解决精确或近似版本的 SVP, 进而通过格问题的归约解决大部分格上困难问题, 这也是格上困难问题的一种通用解法. 除此之外, 针对特定格上困难问题专门设计的量子求解算法也是格上困难问题量子求解算法的重要组成部分. 目前, 越来越多的后量子公钥密码方案选择 LWE 问题或其变形作为安全性基础, 对该问题的量子求解算法作为重要的密码分析手段也成为研究热点. 设 k、q 是整数, $\boldsymbol{s}\in Z_q^k$ 是未知秘密向量, $\boldsymbol{e}\in Z_q^k$ 是未知噪音向量, 给定已知矩阵 $\boldsymbol{A}=\{\boldsymbol{a}_i\}\in Z_q^{m\times k}$ 和已知向量 $\boldsymbol{b}\equiv\boldsymbol{A}\cdot\boldsymbol{s}+\boldsymbol{e}\bmod q$, LWE 问题要求计算秘密向量 $\boldsymbol{s}$ 和噪音向量 $\boldsymbol{e}$. 该问题有多种求解方法, 除了归约至格上 SVP 的通用求解方法 (称为 Primal 方法), 还存在 BKW 算法、对偶算法等专门的求解方法, 其具体关系如图 8 所示, 对这些算法的量子加速是格上困难问题量子求解算法的重要研究内容. 近年来基于变分量子算法的 LWE 求解研究也得到了学术界的关注.

按照求解方法和问题归约路径, 可将常用的量子 LWE 求解方法分为以下几种: (1) 基于量子 SVP 求解的攻击, 遵循 Primal 攻击的基本思想, 实质上是量子筛法等量子 SVP 算法在 LWE 中的应用; (2) 量子 BKW 算法, 该算法的量子加速有两个方面, 其一是格向量组合的量子加速, 其二是量子傅里叶变换的加速; (3) 量子对偶攻击, 该方

法结合了格方法与傅里叶分析, 可以从这两个方面进行量子加速; (4) 变分量子算法等其他量子算法. 本节从算法结构、主要技术、复杂度分析等方面介绍 LWE 的量子求解算法, 重点梳理量子 BKW 算法的量子采样组合过程和量子对偶算法的量子傅里叶变换过程, 并介绍基于变分量子算法的 LWE 求解发展情况.

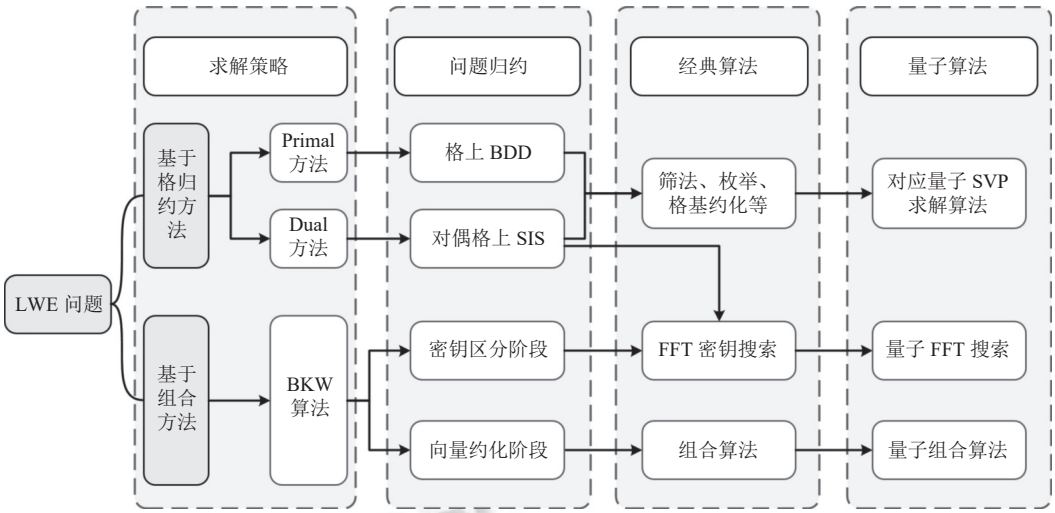

图 8 LWE 问题求解主要算法分类

5.1 量子 BKW 算法

BKW 算法是一种基于组合的求解算法, 可用于求解错误学习问题 LWE 和含噪奇偶校验问题 LPN. 该算法分为约化和区分两个阶段. BKW 算法的约化阶段将输入的 LWE 采样 $(\boldsymbol{a}_i, b_i)$ 进行加减组合, 最终得到满足 $\boldsymbol{a}_i'$ 的前若干个元素均为 0 的 LWE 采样 $(\boldsymbol{a}_i', b_i')$. 迭代进行以上描述的采样组合过程之后, BKW 算法的约化阶段输出大量的 LWE 采样, 其非零部分构成一个新的 LWE 实例. 新 LWE 实例的维度相对原 LWE 降低, 且新的秘密向量 $\boldsymbol{s}'$ 就是原秘密向量 $\boldsymbol{s}$ 在对应位置的分量. BKW 算法的区分阶段利用多数投票或傅里叶变换方法得到约化后新 LWE 实例的解 $\boldsymbol{s}'$, 即原 LWE 问题的部分秘密向量. 由 BKW 的基本结构可知, 算法的主要计算消耗来自第 1 阶段中向量组合的操作. Albrecht 等人[89]首先提出了针对 LWE 的 BKW 求解算法, 主要工作是推广求 LPN 的 BKW 算法版本, 其约化阶段只涉及 2 个向量的组合. Esser 等人[41]为优化 BKW 算法约化阶段的时间/空间复杂度折中, 讨论了以 c 个向量为一组进行组合, 将此过程描述为 c-求和问题 (c-sum problem, c-SP), 基于此思路提出了 c-求和 BKW 算法, 并分析了此算法的经典复杂度和量子复杂度. Wei 等人[90]对 BKW 求解多种参数集的复杂度进行了总结. Guo 等人[91]讨论了 BKW 算法与量子筛法的结合运用.

本节介绍基于 Grover 量子搜索的量子 c-求和算法和量子 c-求和 BKW 算法, 如算法 3 和 4 所示, 主要聚焦约化阶段, 实际上初始版本的 BKW 也可以视为 c-求和问题取 c=2 的特例. 对 c-求和 BKW 算法实现量子加速的关键是优化第 1 阶段的量子组合方法, 即使用量子加速技术求解 c-求和问题. 设自然数 $c \geqslant 2$ 和素数 q, 给定由 b 维随机向量 $\boldsymbol{l}_i$ 组成的列表 $L = (\boldsymbol{l}_1, \ldots, \boldsymbol{l}_N)$, 且定义 $\boldsymbol{l}_{-i} = -\boldsymbol{l}_i$. 对于 b 维目标向量 $\boldsymbol{t}$, c-求和问题 c-$\mathrm{SP}_b(L, \boldsymbol{t})$ 要求找到集合 $[\pm N] = \{\pm 1, \pm 2, \ldots, \pm N\}$ 的大小为 c 的子集 $\mathcal{L}$, 满足 $\sum_{j\in\mathcal{L}} \boldsymbol{l}_j \equiv \boldsymbol{t} \bmod q$. 一个指数组合 $\mathcal{L}$ 称为 c-求和问题的一个单解, 而完整的 c-求和问题解是 N 个不同 $\mathcal{L}$ 的集合. 要将 c-求和问题应用于 BKW 算法, 可以设定目标向量 $\boldsymbol{t}$ 是全零向量. 理论分析证明当输入的 LWE 采样足够多, c-求和问题可以有解. 经典 c-求和问题求解算法通过搜索所有可能的 $\{i_1, i_2, \ldots, i_c\}$ 系数组合来找到 c-求和问题的解. c-求和问题可以使用 Grover 量子搜索算法进行加速, 但首先需要解决的一个问题是对经典内存的大量访问. 为实现量子算法访问经典内存, 文献 [41] 使用量子随机访问存储 (QRAM) 模型. 在此量子计算模型下, 量子 c-求和算法构造了如下定义的目标函数 f:

$$f_t:\binom{[\pm N]}{c-1}\to\{0,1\},\ v\mapsto\begin{cases}1, & \exists i_c\in[\pm N]\backslash v:\sum_{j=1}^{c-1}\boldsymbol{l}_{i_j}=-\boldsymbol{l}_{i_c}+\boldsymbol{t}\\ 0, & \text{else}\end{cases}.$$

Grover 量子搜索算法能在定义域 $\binom{[\pm N]}{c-1}$ 中搜索使 $f_t(i_1,i_2,\ldots,i_{c-1})=1$ 的系数 $i_1,i_2,\ldots,i_{c-1}$, 进而得到 c-求和问题的一个单解. 基于 Grover 量子搜索算法构造的量子 c-求和步骤如算法 3 所示.

算法 3. 量子 c-求和算法.

输入: $L=(\boldsymbol{l}_1,\ldots,\boldsymbol{l}_N)$;
输出: 集合 S 或 $\perp$.

1. 初始化 QRAM 存储 O_W
2. 对 L 中每个 $\boldsymbol{l}_j$, 加入 O_W 第 j 位
3. 定义预言 $\hat{O}(i_1,i_2,\ldots,i_{c-1})$:
4. 从 O_W 中取出 $\boldsymbol{l}_{i_1}$, $\boldsymbol{l}_{i_2}$,…, $\boldsymbol{l}_{i_{c-1}}$
5. IF 存在 $\boldsymbol{l}_c$ 使得 $\sum_{j=1}^{c-1}\boldsymbol{l}_{i_j}=-\boldsymbol{l}_{i_c}+\boldsymbol{t}$
6. 输出 1
7. 以下步骤重复 O(N) 次:
8. Grover 搜索找到 $i_1,\ldots,i_{c-1}$, 使 $\hat{O}(i_1,\ldots,i_{c-1})=1$
9. 根据 $(i_1,i_2,\ldots,i_{c-1})$ 恢复 i_c 满足 $\sum_{j=1}^{c-1}\boldsymbol{l}_{i_j}=-\boldsymbol{l}_{i_c}+\boldsymbol{t}$
10. $S\leftarrow S\cup\{\{i_1,\ldots,i_c\}\}$
11. IF $|S|=N$
12. 输出 S
13. 输出 $\perp$

由于 BKW 中 c-求和的目标向量为零向量, 故系数 $(i_1,i_2,\ldots,i_c)$ 与 $(-i_1,-i_2,\ldots,-i_c)$ 不应重复出现. 所以, 函数 $f_t(i_1,i_2,\ldots,i_{c-1})$ 的定义域大小可折半为 $|D|=\left|\binom{[\pm N]}{c-1}\right|\cdot\frac{1}{2}=\binom{N}{c-1}2^{c-2}\leqslant N^{c-1}2^{c-2}$. Grover 量子搜索算法找到单解的复杂度可估计为 $\mathrm{O}(\sqrt{|D|/|f^{-1}(1)|})=\mathrm{O}(\sqrt{N^{c-1}2^{c-2}/N})=\mathrm{O}(N^{c/2-1}2^{c/2-1})$, 量子 c-求和算法的总时间复杂度为 $\mathrm{O}(N\cdot N^{c/2-1}2^{c/2-1})=\mathrm{O}(N^{c/2}2^{c/2-1})$. 相比经典 c-求和算法的时间复杂度 $\mathrm{O}((2N)^{c-1})$, 采用量子 Grover 搜索的 c-求和算法实现了平方级加速. 基于此 c-求和算法, 给定 m_{BKW} 个 k 维 LWE 采样, 设计量子 BKW 如算法 4 所示.

算法 4. 量子 c-求和 BKW 算法.

输入: 分块个数 $a>0$, 分块长度 $b=\frac{k(1+\epsilon_a)}{a}$, $\epsilon_a>0$;
输出: LWE 秘密向量 $\boldsymbol{s}$.

(约化阶段)
1. FOR $i\leftarrow 1,2,\ldots,m_{\mathrm{BKW}}$
2. FOR $j\leftarrow 1,2,\ldots,a-1$
3. $L\leftarrow c\text{-}SP(L,0^b)$
4. IF $L=\varnothing$

5. 输出 $\perp$

6. 在 L 中选取新生成的 k' 维 LWE 采样 $(\boldsymbol{a}_i, b_i)$

(区分阶段)

7. 使用傅里叶变换以 m_{BKW} 个新 LWE 采样作为输入并求解, 作为原 LWE 解的 k' 个分量

8. 递归进行前两阶段, 恢复完整的秘密向量 $\boldsymbol{s}$

9. 输出 $\boldsymbol{s}$

上述量子 BKW 算法在经典算法的基础上, 通过反复调用量子 Grover 搜索实现了格向量组合环节的平方级加速. 更进一步, 相比原始 BKW 算法每次只组合一对向量, c-求和 BKW 每次进行 c 个向量的组合, 通过调节 c 值可以改变算法的时间复杂度存储折中. 给定 n 维 LWE 问题, 其中模数 $q=n^{\nu_q}$, 噪音标准差 $\delta=n^{\nu_\delta}$, 则 BKW 算法的时间复杂度 $T=2^{\theta(1+\epsilon)}$ 和空间复杂度 $M=2^{\mu(1+\epsilon)}$ 可以表示为:

$$\theta=t\cdot\frac{1}{2}\cdot\frac{\nu_q}{\nu_q-\nu_\delta+\frac{1}{2}}\cdot n,\ \mu=m\cdot\frac{1}{2}\cdot\frac{\nu_q}{\nu_q-\nu_\delta+\frac{1}{2}}\cdot n,$$

其中, 不同 BKW 版本的复杂度参数 t、m 如表 5 所示. 应用量子搜索方法后, BKW 算法的时间复杂度得到了平方级的下降. 下一步量子 BKW 算法在向量组合阶段的发展方向有: (1) 给出更精确的量子 BKW 复杂度估计; (2) 参考量子筛法等, 探索 BKW 算法结构与更多量子加速技术的融合应用.

表 5 c-求和 BKW 算法复杂度参数

类型	算法	t	m	条件
经典算法	原始BKW	1	1	—
	c-求和BKW	$\log c$	$\frac{\log c}{c-1}$	$c\geqslant 2$
量子算法	原始BKW	$\frac{1}{2}$	1	—
	c-求和BKW	$\frac{c\log c}{2(c-1)}$	$\frac{\log c}{c-1}$	$c\geqslant 2$

5.2 基于量子傅里叶变换的 LWE 求解

第 5.1 节介绍了 BKW 第 1 阶段中利用 Grover 量子搜索算法加速向量组合的算法原理和复杂度. 第 1 阶段之后, BKW 第 2 阶段需要使用快速傅里叶变换恢复秘密向量, 此过程也可以使用量子加速技术, 即量子傅里叶变换 (quantum Fourier transform, QFT)[92–94]. 量子傅里叶变换不仅应用于求解 LWE 的量子 BKW 算法, 同时也是求解 LWE 的对偶攻击 (dual attack)[42,95]的重要组成部分. 对偶攻击经过 Guo 等人[96]的改进, 已经在求解 LWE 的效率上超过 Primal 方法; Albrecht 等人[42]指出, 对偶攻击能够量子傅里叶变换结合; Pouly 等人[95]给出了对偶攻击的理论证明. 下面以对偶攻击为例介绍量子傅里叶变换在求解 LWE 问题的应用.

对偶攻击的基本思路是将 LWE 转化为格上小整数解问题, 并用傅里叶变换找到秘密向量. 首先, 将 $\boldsymbol{A}$ 矩阵分块表示为 $\boldsymbol{A}=[\boldsymbol{A}_0|\boldsymbol{A}_1]$, 并将 $\boldsymbol{s}$ 相应地分段表示为 $\boldsymbol{s}=[\boldsymbol{s}_0|\boldsymbol{s}_1]$, 则有 $\boldsymbol{b}\equiv\boldsymbol{A}_0\cdot\boldsymbol{s}_0+\boldsymbol{A}_1\cdot\boldsymbol{s}_1+\boldsymbol{e} \bmod q$. 随后, 要求找到向量 $\boldsymbol{x}$, $\boldsymbol{y}$ 使得 $\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{A}_0\equiv\boldsymbol{y} \bmod q$, 这一操作相当于求小整数解问题, 可以通过对偶格上的格基约化实现. 此时有等式 $<\boldsymbol{x},\boldsymbol{b}>\equiv\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{A}_0\cdot\boldsymbol{s}_0+\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{A}_1\cdot\boldsymbol{s}_1+<\boldsymbol{x},\boldsymbol{e}>=<\boldsymbol{y},\boldsymbol{s}_0>+\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{A}_1\cdot\boldsymbol{s}_1+<\boldsymbol{x},\boldsymbol{e}>$. 接下来, 对任意可能的 $\boldsymbol{s}_1$ 计算 $<\boldsymbol{x},\boldsymbol{b}>-\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{A}_1\cdot\boldsymbol{s}_1$. 对随机的 $\boldsymbol{s}_1$ 值, 得到的值均匀随机分布; 而当 $\boldsymbol{s}_1$ 猜测恰好是秘密向量 $\boldsymbol{s}$ 的对应分量时, 得到的 $<\boldsymbol{x},\boldsymbol{b}>-\boldsymbol{x}^{\mathrm{T}}\cdot\boldsymbol{A}_1\cdot\boldsymbol{s}_1$ 是一个小值. 这个猜测 $\boldsymbol{s}_1$ 的过程可以通过傅里叶变换实现, 进而使用量子傅里叶变换达到量子加速效果.

要将量子傅里叶变换应用于对偶攻击, 需解决的关键问题是计算余弦值 $\cos(2\pi(<\boldsymbol{w}_i,\boldsymbol{b}>)/q)$. 定义 $f_W(\boldsymbol{b})=\frac{1}{N}\sum_{i=0}^{N-1}\cos(2\pi(<\boldsymbol{w}_i,\boldsymbol{b}>)/q)$. 对任意 ϵ, $\delta>0$, 存在量子算法可以实现预言 $\hat{O}$, 使得给定 $\boldsymbol{b}\in Z_q^n$ 后得到输出 $\hat{O}(\boldsymbol{b})$, 以 $1-\delta$ 的概率满足 $|\hat{O}(\boldsymbol{b})-f_W(\boldsymbol{b})|\leqslant\epsilon$, 此结果保证了量子傅里叶变换可以适用于对偶攻击. 在进行傅里叶变换后, 将

结果存储在列表中, 最后使用 Grover 量子搜索算法寻找符合条件的解. 对于长度为 $N = 2^n$ 的输入, 量子傅里叶变换的时间复杂度为 $O((\log_2 N)^2)$, 相比经典的快速傅里叶变换复杂度 $O(N\log_2 N)$ 有指数级加速.

相比将格上困难问题归约为 SVP 的通用解法, 专用的 LWE 量子求解算法立足于 LWE 问题特殊结构, 通过量子搜索、量子傅里叶变换等技术降低问题复杂度. 除了以上 LWE 量子求解方法, 研究者还基于量子计算特性设计了更多特殊的求解算法. 如 Lv 等人[97]结合编码方法和变分量子算法提出了一种 LWE 量子求解方法, 使用两个量子比特编码格基组合系数的随机浮动, 将向量扰动过程进行指数级加速. 作者还利用这种扰动加速了对 LWE 的 Primal 求解方法. 未来 LWE 量子求解发展可以探索的问题包括根据具体密码方案中 LWE 的特殊结构调整求解算法[64,98]、采用最新的量子加速技术[99]提高量子求解算法效率等, 本节总结了量子模型下 LWE 求解研究的发展概况, 如图 9 所示.

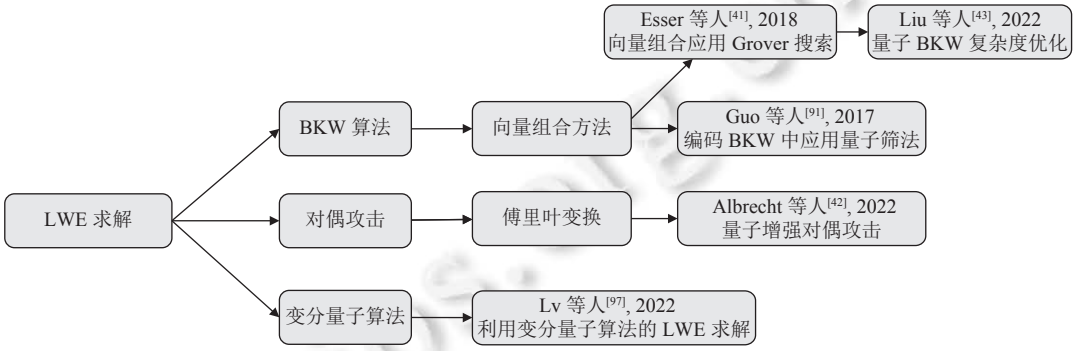

图 9 量子 LWE 求解算法发展概况

6 NTRU 量子密钥恢复

本节介绍针对 NTRU 公钥密码体制[29]的量子密钥恢复算法, 包括利用 Grover 量子搜索的量子密钥恢复和利用量子碰撞的量子密钥恢复. NTRU 提出于格理论发展早期, 至今仍是重要的格密码之一, 并且是 NIST 于 2022 年公布的后量子公钥密码标准之一. 同样入选 NIST 标准算法的 Falcon 数字签名也是构造在 NTRU 格上的. 虽然 NTRU 的安全性基础是格上 SVP, 但除了通用的归约至 SVP 求解还存在基于量子搜索算法的密钥恢复方法, 如图 10 所示.

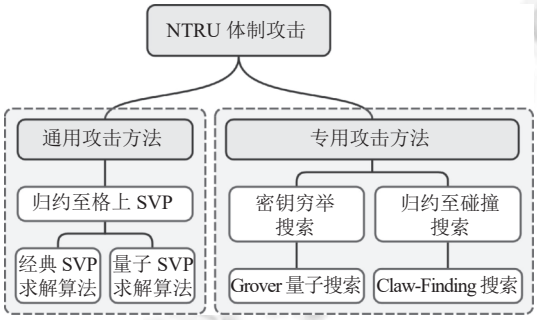

图 10 NTRU 量子密钥恢复分类

对 NTRU 进行密钥恢复的基本操作是在 N 维模多项式环上进行的[100]. 定义 $\mathcal{L}(d_1,d_2)$ 为含有 d_1 个+1 系数、d_2 个–1 系数, 其余全为 0 的三元多项式集合. 定义素数 q 和小整数 p. NTRU 中, 接收方 Bob 选取多项式, $g(x) \in \mathcal{L}_g(d_g,d_g)$, 其中 d_f、d_g 是给定参数, 多项式对 f, g 是私钥. Bob 分别求 f 模 p、q 的逆多项式 $f_p(x)$、$f_q(x)$, 并在 N 维模 q 多项式环上做卷积计算, 得到 $h(x) \equiv pf_q(x)g(x) \bmod (x^N - 1)$ 作为公钥. 为了加速私钥 f 的生成, NTRU 在实现中通常采用乘积多项式模式: 选择 3 个小系数多项式 $f_1(x)$、$f_2(x)$、$f_3(x)$ 组合得 $f(x) = f_1(x)f_2(x) + f_3(x)$. 针对这种

密钥结构, 研究者构造了对应的量子密钥恢复攻击, 包括基于 Grover 量子搜索的 NTRU 量子密钥恢复和基于 Claw-Finding 量子碰撞的量子密钥恢复, 如图 11 所示.

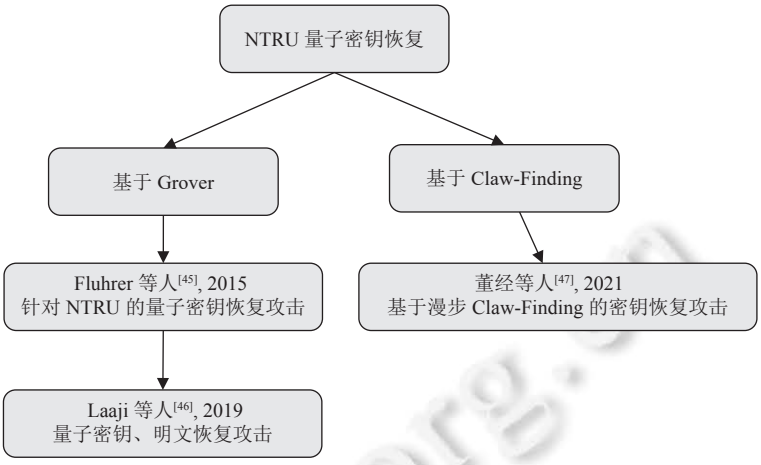

图 11 NTRU 量子攻击发展概况

6.1 基于 Grover 搜索的 NTRU 量子密钥恢复

Fluhrer 等人[45]基于 Grover 量子搜索提出了对 NTRU 的量子密钥恢复, 该方法之后由 Laaji 等人[46]扩展为针对密钥或明文的恢复攻击. 本节总结基于 Grover 搜索的 NTRU 量子密钥恢复原理. 首先将 NTRU 的公钥生成过程用向量乘积表示, 根据公钥 $h(x)=h_0+h_1x+\ldots+h_{N-1}x^{N-1}$ 的系数定义矩阵.

$$\boldsymbol{H}=\begin{bmatrix} h_0 & h_1 & \cdots & h_{N-1} \\ h_N & h_0 & \cdots & h_{N-2} \\ \vdots & \vdots & \ddots & \vdots \\ h_2 & h_1 & \cdots & h_0 \end{bmatrix}.$$

将 (f,g) 用系数向量表示, 则得到等式 $f\cdot\boldsymbol{H}=pg \bmod q$. 模 p 可得 $(f\cdot\boldsymbol{H}=0 \bmod q) \bmod p$. 在乘积多项式模式下, 等式可表示为 $((f_1f_2)\cdot\boldsymbol{H}=-f_3\cdot\boldsymbol{H} \bmod q) \bmod p$.

基于 Grover 量子搜索算法的 NTRU 量子密钥恢复分两步. 首先, 将所有可能的二元组 $(f_3,(-f_3\cdot\boldsymbol{H} \bmod q) \bmod p)$ 存储在列表 L. 随后, 搜索 f_1f_2, 使得对应的 $((f_1f_2)\cdot\boldsymbol{H} \bmod q) \bmod p$ 存在于列表 L, 即可找到一个满足 $((f_1f_2)\cdot\boldsymbol{H}=-f_3\cdot\boldsymbol{H} \bmod q) \bmod p$ 的碰撞. 该过程的搜索空间大小为 $|f_1f_2|$. 根据 NTRU 的循环移位特性, 搜索空间可还继续降至 $|f_1f_2|/N$: 多项式环 $F[x]/(x^N-1)$ 中的任意两个多项式 f_1, f_2 存在关系 $(x^mf_1)f_2=(x^m)f_1f_2$. 更进一步, $(x^mf_1)(x^{-m}f_2)=f_1f_2$, 故 NTRU 密码方案中有 $x^{-m}f\cdot\boldsymbol{H}=px^{-m}g \bmod q$. 该结果表示环 $F[x]/(x^N-1)$ 中多项式乘 x^m 表示多项式向右或向左循环移动 m 位, 循环移动前后的多项式的系数只有位置发生了变化, 而数值保持不变. 因此, 对于公钥 h, $x^{-m}f$, $x^{-m}g$ 也是一组可用的私钥, 对密文进行解密之后, 仅需再次循环位移即可得到明文信息, 且私钥对 $(x^{-m}f,x^{-m}g)$ 的恢复方法与 (f,g) 的恢复方法相同. 在乘积多项式模式下, 同样有 $((x^mf_1f_2)\cdot\boldsymbol{H}=-x^mf_3\cdot\boldsymbol{H} \bmod q) \bmod p$, 即只要有满足 $((f_1f_2)\cdot\boldsymbol{H}=-f_3\cdot\boldsymbol{H} \bmod q) \bmod p$ 的多项式 f_1'、f_2'、f_3' 就可以构造 NTRU 的私钥或其循环移位多项式. 综上所述, 基于 Grover 量子搜索的 NTRU 量子密钥搜索的查询复杂度为 $O(\sqrt{|f_1f_2|/N})$.

文献 [45] 要求使用复杂度 O(1) 的预言机 $\hat{O}$ 来判定列表 L 中是否存在 $(-f_3\cdot\boldsymbol{H} \bmod q) \bmod p$ 与特定的 $((f_1f_2)\cdot\boldsymbol{H} \bmod q) \bmod p$ 值相等, 从而恢复私钥 f. 对存储于 L 中的 $(f_3,(-f_3\cdot\boldsymbol{H} \bmod q) \bmod p)$, 预言定义如下:

$$\hat{O}(f_1f_2)=\begin{cases} 1, & (f_1f_2 \bmod q) \bmod p\text{在列表}L\text{中有对应值} \\ 0, & \text{其他} \end{cases}.$$

在此预言机基础上, 文献分析了部分 NTRU 参数下密钥恢复的复杂度, 如表 6 所示.

表 6 部分 NTRU 参数集的量子复杂度

参数集	N	df_2	df_1	df_3	k	量子复杂度	L列表大小
EES593EP1	593	10	10	8	192	2^{136}	$2^{107.3}$
EES587EP1	587	10	10	8	192	2^{132}	$2^{107.1}$
EES401EP2	401	8	8	6	112	2^{103}	$2^{75.9}$
EES439EP1	439	9	8	5	128	2^{111}	$2^{65.9}$
EES743EP1	743	11	11	15	256	2^{196}	$2^{195.2}$

6.2 基于 Claw-Finding 量子碰撞的 NTRU 量子密钥恢复

基于 Grover 量子搜索的 NTRU 量子密钥恢复需要使用判断矩阵乘积匹配的预言机, 且假定其容易通过量子实现, 但这种预言机在实际的量子算法设计中较为复杂. 此外, 基于 Grover 量子搜索的 NTRU 量子密钥恢复需要预先计算大量的 $(f_3,(-f_3\cdot \boldsymbol{H} \bmod q) \bmod p)$ 值并储存在列表中, 这进一步增加了实现难度. 为避免以上问题, 需要考虑新的 NTRU 量子攻击算法. 除了使用 Grover 量子搜索算法, 还存在使用 Claw-Finding 量子碰撞算法[17]的 NTRU 量子密钥恢复.

Claw-Finding 量子碰撞算法的思路类似中间相遇攻击. 给定有限集合上的两个函数 $f: X\to Z$ 和 $g: Y\to Z$, Claw-Finding 量子碰撞算法可以找到一对 (x,y) 称为 claw, 使得 $f(x)=g(y)$. 要将该算法应用于 NTRU 量子密钥恢复中, 可将 $((f_1f_2)\cdot \boldsymbol{H} \bmod q) \bmod p$ 与 $(-f_3\cdot \boldsymbol{H} \bmod q) \bmod p$ 定义为两个函数, 之后用 Claw-Finding 量子碰撞算法求出满足两个函数相等的 (f_1f_2) 和 f_3, 进而得到私钥 f.

为了构造 Claw-Finding 量子算法, Tani 等人[17]依靠 Johnson 图来描述 Claw-Finding 问题. 一个 Johnson 图 $J(n,k)$ 是一个连通正则图, 图上每个顶点都用 $\{1,2,\ldots,n\}$ 的大小为 k 的子集表示. 当且仅当两个点对应的子集仅有一个元素不同时, 两点之间有边. 如图 12 给出了点集合 (a,b,c,d,e) 组成的 Johnson 图 $J(5,3)$ 示例. 为了将 NTRU 的密钥恢复攻击转化为 Claw-Finding 问题, 首先定义两个 Johnson 图 $J_1(|f_1f_2|,l)$, $J_2(|-f_3|,m)$. 由此两个图定义图类积 $G=J_1\times J_2$, 将 J_1, J_2 上的点分别表示为 F, G, 则 G 上点为 (F,G). G 上两个点 (F,G) 和 (F',G') 之间有边当且仅当 F, F' 和 G, G' 之间分别有边.

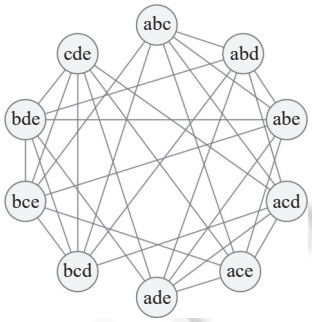

图 12 Johnson 图 $J(5, 3)$ 示例

实现 Claw-Finding 量子碰撞算法的基本思路是在 (f, g) 的定义域中选取一定数量的输入值构造 Johnson 图上的点, 进而按照定义生成图类积 G, 如果当前的点中含有一对输入可构成碰撞, 则该点为目标点. 为了利用量子漫步找到目标点, 需要利用自循环将原始的 Johnson 图由无向图变为部分有向图, 令目标点只会漫步到自身, 否则依据转移概率继续进行散射. 通过此方法构造的 Claw-Finding 量子碰撞算法分为检测算法和搜索算法两部分, 其中检测算法是搜索算法的子过程. Claw-Finding 量子碰撞算法使用的预言机仅需读取 $(f_1f_2\cdot \boldsymbol{H})$ 和 $(-f_3\cdot \boldsymbol{H})$ 的函数值, 且不需要存储大规模列表, 相比基于 Grover 搜索的 NTRU 量子密钥中的预言机更为简单. 令 $Q_2(claw_{\text{finding}}(M_1,M_2))$

表示找到函数 $f:X=[M_1]\to Z$, $g:Y=[M_2]\to Z$, 的碰撞需要查询标准预言机的次数, 则其复杂度分析如下. 搜索算法调用常数次检测算法, 而检测算法的查询复杂度为 $\mathrm{O}(\sqrt{M_1M_2/l})$ 次插入/删除操作和 $\mathrm{O}(l+m)$ 次预言机查询; 当 $M_1\leqslant M_2<M_1^2$ 时, 令 $l=m=\Theta\left((M_1M_2)^{1/3}\right)$; 当 $M_2\geqslant M_1^2$ 时, 令 $l=m=M_1$; 综合以上分析, Claw-Finding 量子碰撞算法总体的时间复杂度如下:

$$Q_2(claw_{\text{finding}}(M_1,M_2))=\begin{cases}\mathrm{O}((M_1M_2)^{1/3}\log(M_1)), & M_1\leqslant M_2<M_1^2\\ \mathrm{O}(M_2^{1/2}\log(M_1)), & M_2\leqslant M_1^2\end{cases}.$$

董经等人[47]针对 NTRU 的求解修改了 Claw-Finding 量子碰撞算法, 使其适用于 NTRU 的多比特输出形式. 由于算法中所搜索函数的自变量分别为 f_3 和 f_1f_2, 根据分析可得算法复杂度为 $\mathrm{O}(\sqrt{|f_1f_2|/N})$. 除了理论上的复杂度下降, 基于 Claw-Finding 量子碰撞的 NTRU 量子密钥恢复相比基于 Grover 搜索的量子密钥恢复不需要维护大规模的指数列表, 同时不需要强量子预言假设, 实现难度更低. 综合对 NTRU 的量子密钥恢复, 基本的设计思路是在密钥搜索的过程中应用 Grover 搜索、量子碰撞搜索等技术, 结合量子算法的特殊性质对算法加速. 在具体的攻击方法中, 还针对 NTRU 的密钥的特殊结构与具体量子加速方法的实现方式, 进一步提高密钥搜索攻击效率. 下一步可以考虑优化问题归约方法, 将原问题转化为新的量子计算问题, 并利用量子碰撞的最新成果[101–103]进一步提高算法效率. 此外, 可以结合侧信道信息[104], 改进对 NTRU 的分析方法.

7 CVP 的量子算法

最近向量问题 (closest vector problem, CVP) 是格理论中重要的基础问题, 也是多种格公钥密码设计与分析的必要工具. 格密码研究中较受关注的是 CVP 的一种变体, 即有界距离解码问题 (bounded distance decoding, BDD). 该问题中, 需要寻找与目标向量间距离小于给定值的格向量. CVP 与多数格问题相似, 存在两类量子求解方法: 第一, 与 SVP 类似, 使用量子枚举等通用量子求解算法进行处理; 第二, 根据问题特定结构设计直接求解算法. 本节据此分类方法, 对 CVP、BDD 的量子求解算法进行介绍, 基本脉络如图 13 所示.

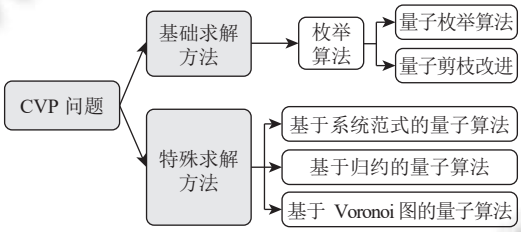

图 13 CVP 量子求解算法分类

CVP 的基础求解算法方面, 较有代表性的是量子枚举算法. Aono 等人[57]将针对 CVP 的枚举算法进行量子化, 并进一步推广至 BDD 问题的求解. 设目标向量为 $\boldsymbol{u}$, 量子枚举算法可输出集合 $\mathcal{L}\cap S$ 中的向量, 其中 $S=Ball_n(\boldsymbol{u},R)$. 接下来通过比较所有向量与 $\boldsymbol{u}$ 的距离, 获得满足条件的向量. 枚举的具体过程与 SVP 求解过程类似, 可以视为对枚举树的深度优先搜索: 深度为 $n-k$ 的节点代表 $\pi_{k+1}(\mathcal{L})\cap S$ 中的格点, 当算法遍历 $\pi_{k+1}(\mathcal{L})\cap S$ 中每个点时, 通过以该点为中心遍历一个小球体可得到 $\pi_k(\mathcal{L})\cap S$ 中所有格点. 除了基本的枚举算法, Aono 等人还针对圆柱体剪枝等多种剪枝技术进行量子算法设计, 实现枚举的多方面加速. 以 BDD 的求解为基础, 作者探讨了该量子算法在 LWE 求解中的应用.

CVP 的特殊求解算法方面, Eldar 等人[62]利用系统范式 (systematic normal form) 进行格的平滑分析, 并依此设计量子算法, 可以在特定参数下有效求解 BDD 问题. Eldar 等人[63]更进一步提出一种量子求解算法, 利用量子傅里叶变换、量子相位估计等手段, 可将最差情况的 BDD 实例归约为随机格上的 BDD, 并提出了一种多项式级的量子求解算法. Aggarwal 等人[105]对 Eldar 等人提出的算法进行了详细讨论, 给出了量子算法与对应经典算法的原理分析. 在此基础上, Aggarwal 等人将 BDD 的研究成果与格向量采样等技术结合, 提出了新的量子筛法, 以 $2^{0.8345n}$ 的时间复杂度求解 SVP. 另一种重要的 CVP 求解算法是基于 Voronoi 图的算法[106], 主要原理是将空间分

割, 使得每一部分内的向量之间互相接近. 随着量子机器学习等量子信息技术的发展, 基于 Voronoi 图的量子算法逐渐受到关注[107]. 因此, CVP、BDD 的量子 Voronoi 图算法也是未来的重要研究课题.

8 特殊结构格上的量子算法

后量子密码学中, 特殊结构的格具有重要的研究意义. 尽管格问题已成为多种密码方案的理论基础, 然而实际的密码方案运行过程中, 随机格上加解密操作需要大量计算. 为追求更高效率, 很多基于格的公钥密码方案选择使用带特殊结构的格作为底层安全性基础. 在此背景下, 针对特定的格结构设计适合的算法成为后量子格密码研究的关键课题, 其中理想格上 SVP 的求解算法尤其受到研究者关注: 当前格密码方案大量使用环 SIS、环 LWE, 而这些特殊结构问题的复杂度可归约至最差情况理想格上的 SVP. 本节重点介绍理想格上 SVP 量子求解算法. 一个理想格的构造如下: 设 m 阶分圆域 K, 度 $n=\varphi(m)$, 其上整环表示为 $\mathcal{R}_K=[\omega_m]$, 在环上取一个整理想 $\mathfrak{h}\subset\mathcal{R}_K$; 在域 K 上通过 Minkowski 嵌入定义典型的 Hermite 向量空间, 即令 $\varsigma_m=\exp(2i\pi/m)\subset C$, $\psi_i: K\to C$ 是由 ω_m 至 ς_m^i 的域同态, i 与 m 互素. 基于以上定义, 任一理想 $\mathfrak{h}\subset\mathcal{R}_K$ 均可通过映射转化为一个格结构. 随着越来越多格密码方案采用特殊结构格, 理想格上的 SVP 求解算法持续吸引研究者的关注, 产生了一系列算法设计与分析成果, 如图 14 所示.

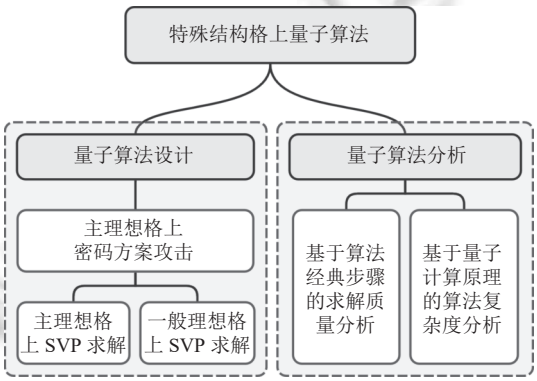

图 14 特殊结构格上量子求解算法发展脉络

理想格量子算法设计方面, 理想格上的 SVP 量子求解算法源于主理想格上全同态加密、多线性映射等密码方案的量子攻击, 其基本原理如下. 对已有的私钥, 选择失真较小的整元素 $g\in\mathcal{R}_K$, 则对应的公钥为理想 $\mathcal{J}=(g)$, 可由一组约化程度较低的格基描述. 攻击方法分为两个阶段: 首先, 利用量子计算求解主理想问题, 即恢复 $\mathcal{J}$ 的一个生成元 h; 随后, 使用 CVP 算法在对数单位格中恢复私钥 g, 此两个步骤的时间复杂度都是多项式级的. Cramer 等人[108]总结此类攻击, 针对主理想格提出了 SVP 求解算法: 假定已知主理想的任一生成元, 则算法在主理想格上能以多项式级别的时间复杂度得到近似因子为 $2^{O(\sqrt{n})}$ 的短向量. 该算法相比随机格上的 SVP 算法, 考虑了主理想格的特殊性质, 达到明显的加速效果, 具体对比见图 15. 虽然该方法可以成功实现对理想格的攻击, 但其明显短板是只讨论了主理想, 而实际格密码中常用的环 SIS、环 LWE 等结构的安全性依赖于一般理想上最差情况的 SVP 复杂度. Cramer 等人[65]总结了已有的针对主理想格密码方案的攻击, 并扩展至一般的理想格, 提出了最差情况的理想格 SVP 求解方法. 给定分圆域 K 上任意理想 $\mathcal{R}_K$, 其量子算法可在多项式时间内求解近似因子 $\gamma=\exp(O(\sqrt{n}))$ 的理想格 SVP 问题. 总体而言, 该算法主要步骤包括: (1) 以量子多项式时间求解近似主乘积问题, 从而将一般理想格上的 SVP 归约至主理想格; (2) 应用已有结果, 求主理想格上的短向量, 并由求解结果恢复理想格上短向量. Cramer 等人[66]进一步将近似因子从 $\exp(O(\sqrt{n}))$ 放宽至 $\exp(O(\sqrt{m}))$, 提出了多项式时间内求较短理想格向量的量子算法.

理想格量子算法分析方面, Ducas 等人[67]在启发式假设的基础上定量估计了上述量子算法的近似因子; 通过理论估计, 20000 阶以上的分圆域上, 基于理想格性质设计的专用量子算法输出结果优于 BKZ-300. Ducas 等人进而提出多种启发式的优化技巧, 如利用已知的大量短向量构造 Voronoi 结构, 进一步降低近似因子的具体取值.

Boer 等人[68]从量子计算基本原理考察理想格的量子算法, 系统分析了求解所需的量子比特数量. 总结已有的多种理想格上问题求解算法, 其突出特点是量子化的步骤集中于主理想格上的 SVP 求解, 而大量的准备步骤仍然是在经典计算模型下进行的, 这为下一步的工作指出了两个可能的重点方向: (1) 结合算法的经典部分与量子部分, 建立系统的复杂度分析模型; (2) 在已有算法框架之上设计新的理想格 SVP 求解算法, 更充分发挥量子计算在各环节的加速作用.

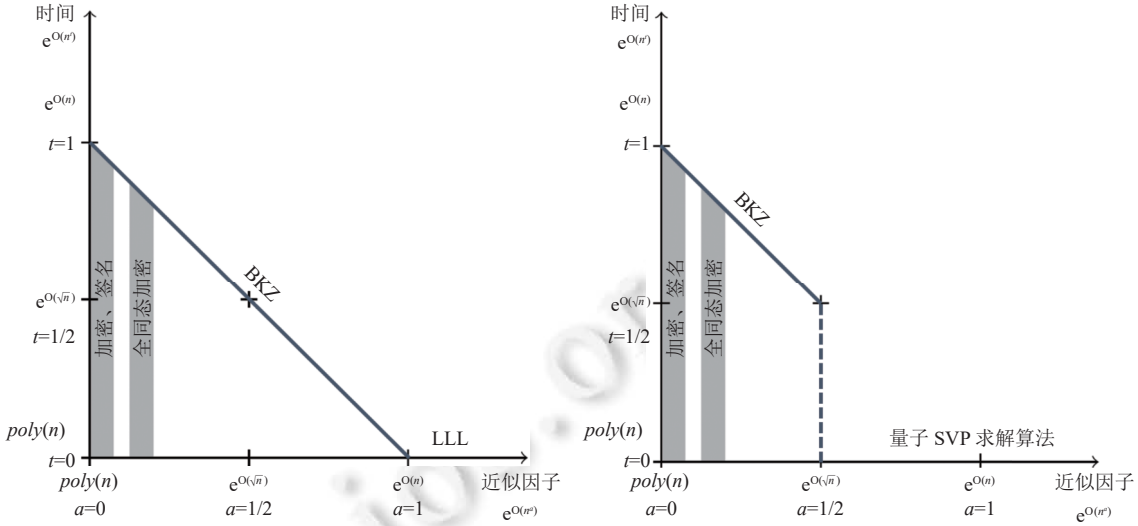

图 15 随机格与主理想格上算法复杂度对比

9 总 结

量子计算机的快速发展使得量子计算技术直接应用成为可能, 亦对公钥密码体制的量子安全性提出了越来越高的要求. 本文围绕格上困难问题量子求解算法设计原理、格上困难问题量子求解算法中应用的量子加速技术、格上困难问题量子求解算法复杂度估计等方面进行综述, 分类梳理了已有的研究成果, 分析了格上困难问题量子求解算法的计算原理, 评估了各类算法的复杂度.

结合当前格上困难问题量子求解算法的研究进展, 可进一步研究的问题包括如下.

(1) 格上困难问题量子算法设计方法创新, 根据量子加速技术的特性设计更适合量子计算模型的新算法;

(2) 新型量子加速技术的应用, 在格上困难问题中应用量子搜索等方面的最新研究成果, 并在可行的量子计算模型下优化量子线路设计;

(3) 格上困难问题量子求解算法的复杂度折中与效率优化, 考虑量子加速对各个步骤时间复杂度和存储复杂度的影响, 针对具体问题寻找最优的效率平衡.

References

[1] Shor PW. Algorithms for quantum computation: Discrete logarithms and factoring. In: Proc. of the 35th Annual Symp. on Foundations of Computer Science. Santa Fe: IEEE, 1994. 124–134. [doi: 10.1109/SFCS.1994.365700]

[2] Regev O. An efficient quantum factoring algorithm. Journal of the ACM, 2023, 72(1): 1–13. [doi: 10.1145/3708471]

[3] Ragavan S, Vaikuntanathan V. Space-efficient and noise-robust quantum factoring. In: Proc. of the 2024 Annual Int'l Cryptology Conf. Cham: Springer, 2024. 107–140. [doi: 10.1007/978-3-031-68391-6_4]

[4] Bonnetain X. Tight bounds for Simon's algorithm. In: Proc. of the 7th Int'l Conf. on Cryptology and Information Security in Latin America. Bogotá: Springer, 2021. 3–23. [doi: 10.1007/978-3-030-88238-9_1]

[5] Simon DR. On the power of quantum computation. In: Proc. of the 35th Annual Symp. on Foundations of Computer Science. Santa Fe:

IEEE, 1994. 116–123. [doi: 10.1109/SFCS.1994.365701]

[6] Wang XY, Liu MJ. Survey of Lattice-based cryptography. Journal of Cryptologic Research, 2014, 1(1): 13–27 (in Chinese with English abstract). [doi: 10.13868/j.cnki.jcr.000002]

[7] Wang C, Yao HN, Wang BN, Hu F, Zhang HG, Ji XM. Progress in quantum computing cryptography attacks. Chinese Journal of Computers, 2020, 43(9): 1691–1707 (in Chinese with English abstract). [doi: 10.11897/SP.J.1016.2020.01691]

[8] Chardouvelis O, Goyal V, Jain A, Liu JH. Quantum key leasing for PKE and FHE with a classical lessor. In: Proc. of the 2025 Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Cham: Springer, 2025: 248–277. [doi: 10.1007/978-3-031-91131-6_9]

[9] Bos J, Costello C, Ducas L, Mironov I, Naehrig M, Nikolaenko V, Raghunathan A, Stebila D. Frodo: Take off the ring! Practical, quantum-secure key exchange from LWE. In: Proc. of the 2016 ACM SIGSAC Conf. on Computer and Communications Security. Vienna: ACM, 2016. 1006–1018. [doi: 10.1145/2976749.2978425]

[10] Ducas L, Kiltz E, Lepoint T, Lyubashevsky V, Schwabe P, Seiler G, Stehlé D. CRYSTALS-Dilithium: A lattice-based digital signature scheme. IACR Trans. on Cryptographic Hardware and Embedded Systems, 2018, 2018(1): 238–268. [doi: 10.13154/tches.v2018.i1.238-268]

[11] Bos J, Ducas L, Kiltz E, Lepoint T, Lyubashevsky V, Schanck JM, Schwabe P, Seiler G, Stehle D. CRYSTALS-Kyber: A CCA-secure module-lattice-based KEM. In: Proc. of the 2018 IEEE European Symp. on Security and Privacy. London: IEEE, 2018. 353–367. [doi: 10.1109/EuroSP.2018.00032]

[12] Wang YR. Research on quantum security for lattice cryptography [Ph.D. Thesis]. Zhengzhou: PLA Strategic Support Force Information Engineering University, 2023 (in Chinese with English abstract). [doi: 10.27188/d.cnki.gzjxu.2023.000022]

[13] Pan YB, Xu J, Wadleigh N, Cheng Q. On the ideal shortest vector problem over random rational primes. In: Proc. of the 40th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Zagreb: Springer, 2021. 559–583. [doi: 10.1007/978-3-030-77870-5_20]

[14] Regev O. On lattices, learning with errors, random linear codes, and cryptography. In: Proc. of the 37th Annual ACM Symp. on Theory of Computing. Baltimore: ACM, 2005. 84–93. [doi: 10.1145/1060590.1060603]

[15] van Tilborg HCA, Jajodia S. Encyclopedia of Cryptography and Security. 2nd ed., New York: Springer, 2011. [doi: 10.1007/978-1-4419-5906-5]

[16] Grover LK. Quantum mechanics helps in searching for a needle in a haystack. Physical Review Letters, 1997, 79(2): 325–328. [doi: 10.1103/PhysRevLett.79.325]

[17] Tani S. Claw finding algorithms using quantum walk. Theoretical Computer Science, 2009, 410(50): 5285–5297. [doi: 10.1016/j.tcs.2009.08.030]

[18] Nemec M, Sys M, Svenda P, Klinec D, Matyas V. The return of coppersmith's attack: Practical factorization of widely used RSA moduli. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: ACM, 2017. 1631–1648. [doi: 10.1145/3133956.3133969]

[19] Aranha DF, Novaes FR, Takahashi A, Tibouchi M, Yarom Y. LadderLeak: Breaking ECDSA with less than one bit of nonce leakage. In: Proc. of the 2020 ACM SIGSAC Conf. on Computer and Communications Security. ACM, 2020. 225–242. [doi: 10.1145/3372297.3417268]

[20] Feng TF. Analysis of a quantum attack on the Blum-Micali pseudorandom number generator. IACR Cryptology ePrint Archive, 2023.1639.

[21] Liu WJ, Gao JT. Quantum security of Grain-128/Grain-128a stream cipher against HHL algorithm. Quantum Information Processing, 2021, 20(10): 343. [doi: 10.1007/s11128-021-03275-x]

[22] Wroński M, Burek E, Leśniak M. (In)security of stream ciphers against quantum annealing attacks on the example of the Grain 128 and Grain 128a ciphers. IEEE Trans. on Emerging Topics in Computing. [doi: 10.1109/TETC.2024.3474856]

[23] Bonnetain X, Schrottenloher A. Single-query quantum hidden shift attacks. IACR Trans. on Symmetric Cryptology, 2024, 2024(3): 266–297. [doi: 10.46586/tosc.v2024.i3.266-297]

[24] Ni BY, Ito G, Dong XY, Iwata T. Quantum attacks against type-1 generalized Feistel ciphers and applications to CAST-256. In: Proc. of the 20th Int'l Conf. on Cryptology in India. Hyderabad: Springer, 2019. 433–455. [doi: 10.1007/978-3-030-35423-7_22]

[25] Zhang ZY, Sun SW, Wang CB, Hu L. Classical and quantum meet-in-the-middle Nostradamus attacks on AES-like hashing. IACR Trans. on Symmetric Cryptology, 2023, 2023(2): 224–252. [doi: 10.46586/tosc.v2023.i2.224-252]

[26] Fehr S, Huang YH. On the quantum security of HAWK. In: Proc. of the 14th Int'l Conf. on Post-quantum Cryptography. College Park: Springer, 2023. 405–416. [doi: 10.1007/978-3-031-40003-2_15]

[27] Ramos-Calderer S, Bellini E, Latorre JI, Manzano M, Mateu V. Quantum search for scaled hash function preimages. Quantum Information Processing, 2021, 20(5): 180. [doi: 10.1007/s11128-021-03118-9]

[28] Baek S, Kim J. Quantum rebound attacks on reduced-round aria-based hash functions. ETRI Journal, 2023, 45(3): 365–378. [doi: 10.4218/etrij.2022-0032]

[29] Hoffstein J, Pipher J, Silverman JH. NTRU: A ring-based public key cryptosystem. In: Proc. of the 3rd Int'l Algorithmic Number Theory Symp. Portland: Springer, 1998. 267–288. [doi: 10.1007/BFb0054868]

[30] Gentry C, Sahai A, Waters B. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In: Proc. of the 33rd Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2013. 75–92. [doi: 10.1007/978-3-642-40041-4_5]

[31] Regev O. Quantum computation and lattice problems. SIAM Journal on Computing, 2004, 33(3): 738–760. [doi: 10.1137/S0097539703440678]

[32] Aono Y, Nguyen PQ, Seito T, Shikata J. Lower bounds on lattice enumeration with extreme pruning. In: Proc. of the 38th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2018. 608–637. [doi: 10.1007/978-3-319-96881-0_21]

[33] May A, Nowakowski J. Too many hints—When LLL breaks LWE. In: Proc. of the 29th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Guangzhou: Springer, 2023. 106–137. [doi: 10.1007/978-981-99-8730-6_4]

[34] 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. Hersonissos: ACM, 2001. 601–610. [doi: 10.1145/380752.380857]

[35] 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. Austin: SIAM, 2010. 1468–1480. [doi: 10.1137/1.9781611973075.119]

[36] Bi L, Lu XH, Wang KP. Research status and development trend of lattice sieving. Journal of Cryptologic Research, 2021, 8(5): 735–757 (in Chinese with English abstract). [doi: 10.13868/j.cnki.jcr.000474]

[37] Pohst M. On the computation of lattice vectors of minimal length, successive minima and reduced bases with applications. ACM SIGSAM Bulletin, 1981, 15(1): 37–44. [doi: 10.1145/1089242.1089247]

[38] Nguyen PQ, Vallée B. The LLL Algorithm: Survey and Applications. Berlin: Springer, 2010. [doi: 10.1007/978-3-642-02295-1]

[39] Lenstra AK, Lenstra Jr HW, Lovász L. Factoring polynomials with rational coefficients. Mathematische Annalen, 1982, 261(4): 515–534. [doi: 10.1007/BF01457454]

[40] Schnorr CP, Euchner M. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. Mathematical Programming, 1994, 66(1–3): 181–199. [doi: 10.1007/BF01581144]

[41] Esser A, Heuer F, Kübler R, May A, Sohler C. Dissection-BKW. In: Proc. of the 38th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2018. 638–666. [doi: 10.1007/978-3-319-96881-0_22]

[42] Albrecht MR, Shen YX. Quantum augmented dual attack. IACR Cryptology ePrint Archive, 2022.656.

[43] Liu HL, Yu Y. A non-heuristic approach to time-space tradeoffs and optimizations for BKW. In: Proc. of the 28th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Taipei: Springer, 2022. 741–770. [doi: 10.1007/978-3-031-22969-5_25]

[44] Chen YL. Quantum algorithms for lattice problems. IACR Cryptology ePrint Archive, 2024.555.

[45] Fluhrer SR. Quantum cryptanalysis of NTRU. IACR Cryptology ePrint Archive, 2015.676.

[46] Laaji EH, Azizi A, Ezzouak S. Two quantum attack algorithms against NTRU when the private key and plaintext are codified in ternary polynomials. In: Serrhini M, Silva C, Aljahdali S, eds. Innovation in Information Systems and Technologies to Support Learning Research. Cham: Springer, 2019. 551–562. [doi: 10.1007/978-3-030-36778-7_61]

[47] Dong J. The quantum algorithm analysis of NTRU cryptography and the research of quantum lattice sieving algorithm [MS. Thesis]. Beijing: Beijing University of Posts and Telecommunications, 2021 (in Chinese with English abstract). [doi: 10.26969/d.cnki.gbydu.2021.001999]

[48] Laarhoven T, Mosca M, van de Pol J. Solving the shortest vector problem in lattices faster using quantum search. In: Proc. of the 5th Int'l Conf. on Post-quantum Cryptography. Limoges: Springer, 2013. 83–101. [doi: 10.1007/978-3-642-38616-9_6]

[49] 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. [doi: 10.1007/s10623-015-0067-5]

[50] Becker A, Laarhoven T. Efficient (ideal) lattice sieving using cross-polytope LSH. In: Proc. of the 8th Int'l Conf. on Cryptology in Africa. Fes: Springer, 2016. 3–23. [doi: 10.1007/978-3-319-31517-1_1]

[51] Becker A, Ducas L, Gama N, Laarhoven T. New directions in nearest neighbor searching with applications to lattice sieving. In: Proc. of the 27th Annual ACM-SIAM Symp. on Discrete Algorithms. Arlington: Society for Industrial and Applied Mathematics, 2016. 10–24. [doi: 10.5555/2884435.2884437]

[52] Kirshanova E, Mårtensson E, Postlethwaite EW, Moulik SR. Quantum algorithms for the approximate *k*-list problem and their application to lattice sieving. In: Proc. of the 25th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Kobe: Springer, 2019. 521–551. [doi: 10.1007/978-3-030-34578-5_19]

[53] Albrecht MR, Gheorghiu V, Postlethwaite EW, Schanck JM. Estimating quantum speedups for lattice sieves. In: Proc. of the 26th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Daejeon: Springer, 2020. 583–613. [doi: 10.1007/978-3-030-64834-3_20]

[54] Chailloux A, Loyer J. Lattice sieving via quantum random walks. In: Proc. of the 27th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Singapore: Springer, 2021. 63–91. [doi: 10.1007/978-3-030-92068-5_3]

[55] Bonnetain X, Chailloux A, Schrottenloher A, Shen YX. Finding many collisions via reusable quantum walks. In: Proc. of the 42nd Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Lyon: Springer, 2023. 221–251. [doi: 10.1007/978-3-031-30589-4_8]

[56] Chailloux A, Loyer J. Classical and quantum 3 and 4-sieves to solve SVP with low memory. In: Proc. of the 14th Int'l Conf. on Post-quantum Cryptography. College Park: Springer, 2023. 225–255. [doi: 10.1007/978-3-031-40003-2_9]

[57] Aono Y, Nguyen PQ, Shen YX. Quantum lattice enumeration and tweaking discrete pruning. In: Proc. of the 24th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Australia: Springer, 2018. 405–434. [doi: 10.1007/978-3-030-03326-2_14]

[58] Bai S, van Hoof MI, Johnson FB, Lange T, Ngo T. Concrete analysis of quantum lattice enumeration. In: Proc. of the 29th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Guangzhou: Springer, 2023. 131–166. [doi: 10.1007/978-981-99-8727-6_5]

[59] Bindel N, Bonnetain X, Tiepelt M, Virdia F. Quantum lattice enumeration in limited depth. In: Proc. of the 2024 Annual Int'l Cryptology Conf. Cham: Springer, 2024: 72–106. [doi: 10.1007/978-3-031-68391-6_3]

[60] Albrecht MR, Prokop M, Shen YX, Wallden P. Variational quantum solutions to the shortest vector problem. Quantum, 2023, 7: 933–949. [doi: 10.22331/q-2023-03-02-933]

[61] Tiepelt M, Szepieniec A. Quantum LLL with an application to Mersenne number cryptosystems. In: Proc. of the 6th Int'l Conf. on Cryptology and Information Security in Latin America. Santiago de Chile: Springer, 2019. 3–23. [doi: 10.1007/978-3-030-30530-7_1]

[62] Eldar L, Shor PW. An efficient quantum algorithm for a variant of the closest lattice-vector problem. arXiv:1611.06999, 2016.

[63] Eldar L, Hallgren S. An efficient quantum algorithm for lattice problems achieving subexponential approximation factor. arXiv:2201.13450, 2022.

[64] Allen R, Berker RE, Casacuberta S, Gul M. Quantum and classical algorithms for bounded distance decoding. IACR Cryptology ePrint Archive, 2022.195.

[65] Cramer R, Ducas L, Wesolowski B. Short stickelberger class relations and application to ideal-SVP. In: Proc. of the 36th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Paris: Springer, 2017. 324–348. [doi: 10.1007/978-3-319-56620-7_12]

[66] Cramer R, Ducas L, Wesolowski B. Mildly short vectors in cyclotomic ideal lattices in quantum polynomial time. Journal of the ACM, 2021, 68(2): 8. [doi: 10.1145/3431725]

[67] Ducas L, Plancon M, Wesolowski B. On the shortness of vectors to be found by the ideal-SVP quantum algorithm. In: Proc. of the 39th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2019. 322–351. [doi: 10.1007/978-3-030-26948-7_12]

[68] de Boer K, Ducas L, Fehr S. On the quantum complexity of the continuous hidden subgroup problem. In: Proc. of the 39th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Zagreb: Springer, 2020. 341–370. [doi: 10.1007/978-3-030-45724-2_12]

[69] Nielsen MA, Chuang IL. Quantum Computation and Quantum Information. Cambridge: Cambridge University Press, 2010. [doi: 10.1017/CBO9780511976667]

[70] Dunkelman O, Keller N, Shamir A. Quantum time/memory/data tradeoff attacks. Designs, Codes and Cryptography, 2024, 92(1): 159–177. [doi: 10.1007/s10623-023-01300-x]

[71] Nguyen PQ, Vidick T. Sieve algorithms for the shortest vector problem are practical. Journal of Mathematical Cryptology, 2008, 2(2): 181–207. [doi: 10.1515/JMC.2008.009]

[72] Kim H, Jang K, Oh Y, Seok W, Lee W, Bae K, Sohn I, Seo H. Finding shortest vector using quantum NV sieve on Grover. In: Proc. of the 26th Int'l Conf. on Information Security and Cryptology. Seoul: Springer, 2023. 97–118. [doi: 10.1007/978-981-97-1235-9_6]

[73] 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. Hong Kong: ACM, 2021. 1–9. [doi: 10.1145/1966913.1966915]

[74] Zhang F, Pan YB, Hu GR. A three-level sieve algorithm for the shortest vector problem. In: Proc. of the 20th Int'l Conf. on Selected Areas in Cryptography. Burnaby: Springer, 2014. 29–47. [doi: 10.1007/978-3-662-43414-7_2]

[75] Bai S, Laarhoven T, Stehlè D. Tuple lattice sieving. LMS Journal of Computation and Mathematics, 2016, 19(Special Issue A): 146–162. [doi: 10.1112/S1461157016000292]

[76] Herold G, Kirshanova E, Laarhoven T. Speed-ups and time-memory trade-offs for tuple lattice sieving. In: Proc. of the 21st IACR Int'l Conf. on Public-key Cryptography. Rio de Janeiro: Springer, 2018. 407–436. [doi: 10.1007/978-3-319-76578-5_14]

[77] Doulgerakis E, Laarhoven T, de Weger B. Sieve, enumerate, slice, and lift: Hybrid lattice algorithms for SVP via CVPP. In: Proc. of the 12th Int'l Conf. on Cryptology in Africa. Cairo: Springer, 2020. 301–320. [doi: 10.1007/978-3-030-51938-4_15]

[78] Montanaro A. Quantum-walk speedup of backtracking algorithms. Theory of Computing, 2018, 14: 15. [doi: 10.4086/toc.2018.v014a015]

[79] Cerezo M, Arrasmith A, Babbush R, Benjamin SC, Endo S, Fujii K, McClean JR, Mitarai K, Yuan X, Cincio L, Coles PJ. Variational quantum algorithms. Nature Reviews Physics, 2021, 3(9): 625–644. [doi: 10.1038/s42254-021-00348-9]

[80] Joseph D, Ghionis A, Ling C, Mintert F. Not-so-adiabatic quantum computation for the shortest vector problem. Physical Review Research, 2020, 2(1): 013361. [doi: 10.1103/PhysRevResearch.2.013361]

[81] Joseph D, Callison A, Ling C, Mintert F. Two quantum ising algorithms for the shortest-vector problem. Physical Review A, 2021, 103(3): 032433. [doi: 10.1103/PhysRevA.103.032433]

[82] Chen YM, Nguyen PQ. BKZ 2.0: Better lattice security estimates. In: Proc. of the 17th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Seoul: Springer, 2011. 1–20. [doi: 10.1007/978-3-642-25385-0_1]

[83] Schnorr CP. A more efficient algorithm for lattice basis reduction. Journal of Algorithms, 1988, 9(1): 47–62. [doi: 10.1016/0196-6774(88)90004-1]

[84] Nguên PQ, Stehlè D. Floating-point LLL revisited. In: Proc. of the 24th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Aarhus: Springer, 2005. 215–233. [doi: 10.1007/11426639_13]

[85] Ryan K, Heninger N. Fast practical lattice reduction through iterated compression. In: Proc. of the 43rd Annual Int'l Cryptology Conf. Santa Bar: Springer, 2023. 3–36. [doi: 10.1007/978-3-031-38548-3_1]

[86] Lin D, Xiang ZJ, Xu RQ, Zhang SS, Zeng XY. Optimized quantum implementation of AES. Quantum Information Processing, 2023, 22(9): 352. [doi: 10.1007/s11128-023-04043-9]

[87] Bhattacherjee S, Hernandez-Castro JC, Moyler J. A greedy global framework for lattice reduction using deep insertions. IACR Communications in Cryptology, 2025, 2(1): 1–46. [doi: 10.62056/aevuommol]

[88] Albrecht M, Bai S, Fouque PA, Kirchner P, Stehlé D, Wen W. Faster enumeration-based lattice reduction: Root Hermite factor $k^{1/(2k)}$ time $k^{k/8+o(k)}$. In: Proc. of the 40th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2023. 186–212. [doi: 10.1007/978-3-030-56880-1_7]

[89] Albrecht MR, Cid C, Faugère JC, Fitzpatrick R, Perret L. On the complexity of the BKW algorithm on LWE. Designs, Codes and Cryptography, 2015, 74(2): 325–354. [doi: 10.1007/s10623-013-9864-x]

[90] Wei Y, Bi L, Lu XH, Wang KP. Security estimation of LWE via BKW algorithms. Cybersecurity, 2023, 6(1): 24. [doi: 10.1186/s42400-023-00158-9]

[91] Guo Q, Johansson T, Mårtensson E, *et al*. Coded-BKW with sieving. In: Proc. of the 23rd Int'l Conf. on the Theory and Applications of Cryptology and Information Security (ASIACRYPT 2017). Hong Kong: Springer, 2017. 323–346. [doi: 10.1007/978-3-319-70694-8_12]

[92] Chang WL, Vasilakos AV. Fundamentals of Quantum Programming in IBM's Quantum Computers. Cham: Springer, 2021. [doi: 10.1007/978-3-030-63583-1]

[93] Schrottenloher A. Quantum linear key-recovery attacks using the QFT. In: Proc. of the 43rd Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2023. 258–291. [doi: 10.1007/978-3-031-38554-4_9]

[94] Dachman-Soled D, Gong HJ, Kippen H, Shahverdi A. BKW meets Fourier new algorithms for LPN with sparse parities. In: Proc. of the 19th Theory of Cryptography Conf. Raleigh: Springer, 2021. 658–688. [doi: 10.1007/978-3-030-90453-1_23]

[95] Pouly A, Shen YX. Provable dual attacks on learning with errors. In: Proc. of the 43rd Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Zurich: Springer, 2024. 256–285. [doi: 10.1007/978-3-031-58754-2_10]

[96] Guo Q, Johansson T. Faster dual lattice attacks for solving LWE with applications to CRYSTALS. In: Proc. of the 27th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Singapore: Springer, 2021. 33–62. [doi: 10.1007/978-3-030-92068-5_2]

[97] Lv LH, Yan B, Wang H, Ma Z, Fei YY, Meng XD, Duan QH. Using variational quantum algorithm to solve the LWE problem. Entropy, 2022, 24(10): 1428. [doi: 10.3390/e24101428]

[98] van Hoof I, Kirshanova E, May A. Quantum key search for ternary LWE. In: Proc. of the 12th Int'l Conf. on Post-quantum Cryptography. Daejeon: Springer, 2021. 117–132. [doi: 10.1007/978-3-030-81293-5_7]

[99] Chen YL, Liu QP, Zhandry M. Quantum algorithms for variants of average-case lattice problems via filtering. In: Proc. of the 41st Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Trondheim: Springer, 2022. 372–401. [doi: 10.1007/978-3-031-07082-2_14]

[100] Chen HT, Chung YH, Hwang V, Liu CT, Yang BY. Algorithmic views of vectorized polynomial multipliers—NTRU prime. In: Proc. of the 24th Int'l Conf. on Applied Cryptography and Network Security. Abu Dhabi: Springer, 2024. 24–46. [doi: 10.1007/978-3-031-54773-7_2]

[101] Peng TC, Cao SJ, Xue R. On quantum query complexities of collision-finding in non-uniform random functions. IACR Cryptology ePrint Archive, 2021.1578.

[102] Dong XY, Sun SW, Shi DP, Gao F, Wang XY, Hu L. Quantum collision attacks on AES-like hashing with low quantum random access memories. In: Proc. of the 26th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Daejeon: Springer, 2020. 727–757. [doi: 10.1007/978-3-030-64834-3_25]

[103] Dinur I. Tight time-space lower bounds for finding multiple collision pairs and their applications. In: Proc. of the 39th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Zagreb: Springer, 2020. 405–434. [doi: 10.1007/978-3-030-45721-1_15]

[104] Zhang SD, Lin XH, Yu Y, Wang WJ. Improved power analysis attacks on Falcon. In: Proc. of the 42nd Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Lyon: Springer, 2023. 565–595. [doi: 10.1007/978-3-031-30634-1_19]

[105] Aggarwal D, Chen YL, Kumar R, Shen YX. Improved classical and quantum algorithms for the shortest vector problem via bounded distance decoding. arXiv:2002.07955, 2020.

[106] Doulgerakis E, Laarhoven T, de Weger B. Finding closest lattice vectors using approximate Voronoi cells. In: Proc. of the 10th Int'l Conf. on Post-quantum Cryptography. Chongqing: Springer, 2019. 3–22. [doi: 10.1007/978-3-030-25510-7_1]

[107] Leporini R, Pastorello D. Quantum-inspired classification based on Voronoi tessellation and pretty-good measurements. Quantum Reports, 2022, 4(4): 434–441. [doi: 10.3390/quantum4040031]

[108] Cramer R, Ducas L, Peikert C, Regev O. Recovering short generators of principal ideals in cyclotomic rings. In: Proc. of the 35th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Vienna: Springer, 2016. 559–585. [doi: 10.1007/978-3-662-49896-5_20]

附中文参考文献

[6] 王小云, 刘明洁. 格密码学研究. 密码学报, 2014, 1(1): 13–27. [doi: 10.13868/j.cnki.jcr.000002]

[7] 王潮, 姚皓南, 王宝楠, 胡风, 张焕国, 纪祥敏. 量子计算密码攻击进展. 计算机学报, 2020, 43(9): 1691–1707. [doi: 10.11897/SP.J.1016.2020.01691]

[12] 王娅如. 格密码的量子安全性研究 [博士学位论文]. 郑州: 战略支援部队信息工程大学, 2023. [doi: 10.27188/d.cnki.gzjxu.2023.000022]

[36] 毕蕾, 路献辉, 王鲲鹏. 格上筛法研究现状与发展趋势. 密码学报, 2021, 8(5): 735–757. [doi: 10.13868/j.cnki.jcr.000474]

[47] 董经. NTRU 公钥密码的量子算法分析和量子格筛算法的研究 [硕士学位论文]. 北京: 北京邮电大学, 2021. [doi: 10.26969/d.cnki.gbydu.2021.001999]

作者简介

曹金政, 博士生, 主要研究领域为后量子密码.

罗向阳, 博士, 教授, 博士生导师, 主要研究领域为多媒体安全, 网络安全.

陈晓峰, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为公钥密码, 云计算安全, 数据安全.

程庆丰, 博士, 教授, 博士生导师, 主要研究领域为公钥密码, 密码协议.