Improved Dual Attack and Trapdoor Sampling via Quantum Rejection Sampling

Cong Ling, Hao Yan, and Nicholas Zhao

Imperial College London, United Kingdom
{c.ling,h.yan22,n.zhao22}@imperial.ac.uk

Abstract. In this work, we revisit the dual attack and GPV trapdoor sampling, focusing on the lattice Gaussian sampling term, which can be a significant bottleneck in the overall complexity. We show that this sampling step can be quantumly accelerated by combining the lower bound underlying Wang and Ling's analysis of Klein's algorithm with the quantum rejection sampling (QRS) framework proposed by Ozols et al. Specifically, this lower bound gives precisely the pointwise domination condition required for quantum rejection sampling when given coherent oracle access to a truncated Klein proposal distribution, which yields a quantum procedure for preparing the truncated dual q-ary lattice Gaussian with a quadratic reduction in the sampling complexity. The truncation radius is chosen so that the truncated distribution is negligibly close to the full lattice Gaussian in total variation distance. Substituting this sampler into the dual attack framework results in reduced overall attack-cost estimates. Compared with Pouly and Shen's modern dual attack under the same parameter choices, our estimates reduce the attack cost by 9, 4, and 13 bits for Kyber-512, Kyber-768, and Kyber-1024, respectively. We also report the corresponding estimates with modulus switching. Finally, by replacing the Markov chain Monte Carlo (MCMC) sampler with the QRS algorithm, we achieve a similar quadratic speedup in the GPV signing process.

Keywords: Lattice-based cryptography · Dual attack · Learning with Errors · Lattice Gaussian sampling · Quantum rejection sampling · GPV sampling

1 Introduction

The Learning with Errors (LWE) and Short Integer Solution problems [22,16] are foundational hardness assumptions upon which modern lattice-based cryptography rests. Their worst-case to average-case reduction—where solving average-case LWE or SIS is at least as hard as approximating certain worst-case lattice problems—established them as cornerstones of post-quantum cryptography. This is reflected in the NIST standardization of lattice-based schemes such as ML-KEM, a key-encapsulation mechanism derived from Kyber whose security relies on the hardness of Module-LWE [17], and FALCON, a digital signature scheme

whose security is associated to the SIS problem [10]. LWE also underpins primitives such as the Fully Homomorphic Encryption (FHE) [6]. Understanding the concrete cost of attacks on LWE is therefore critical for assessing the security of deployed schemes.

There are many different attack methods for solving LWE, including lattice, combinatorial, algebraic, exhaustive-search, and hybrid methods. Among lattice-based approaches, the two dominant methods are primal and dual attacks; we focus on the latter. At a high level, a dual attack uses short vectors from an appropriate dual q-ary lattice to distinguish LWE samples from uniform; combined with a guessing step over part of the secret, this yields an attack on the decision variant of LWE. Prior dual attacks [2,12,15] establish this distinguisher via statistical heuristics in which a cosine-sum over short dual vectors concentrates around a Gaussian-derived expectation for the correct guess, and concentrates around zero for incorrect ones; the validity of this argument has recently been placed under scrutiny [9]. Recent work by Pouly and Shen [20] moved away from relying on such heuristics and instead rooted the attack in rigorous geometric analysis. Their attack splits the LWE secret into a guessed part and a dual part, with the matrix $\mathbf{A}$ split correspondingly into $\mathbf{A}_{\text{guess}}$ and $\mathbf{A}_{\text{dual}}$. A basis of the dual lattice $\mathcal{L}_q^{\perp}(\mathbf{A}_{\text{dual}})$ is first reduced using BKZ [23], and a Gaussian sampler via the Markov-chain Monte Carlo (MCMC) algorithm [29] is then used to draw many short vectors from this reduced basis. For each candidate guess, these vectors are combined into a cosine-sum score that tracks how close the corresponding residual lies to $\mathcal{L}_q(\mathbf{A}_{\text{dual}})$. The correct guess yields a residual equal to the LWE error modulo the lattice, while any incorrect guess shifts this residual by a nonzero coset representative whose distance to the lattice is bounded below by the shortest nonzero vector of $\mathcal{L}_q(\mathbf{A})$, separating correct from incorrect guesses whenever the error is short enough relative to this minimum. Crucially, the entire argument is geometric and requires no statistical assumptions on the short vectors, which is the central contribution distinguishing this work from prior dual attacks. The quantum variant of their dual attack based on [3] accelerates the search stage of the dual attack using quantum mean estimation to approximate the cosine score associated with a candidate guess, after which quantum maximum finding is used to search over the guessed secret coordinates, replacing the classical guessing term with a quadratically reduced one. However, the dominant Gaussian sampling cost remains untouched.

The lattice Gaussian sampling step relies on MCMC methods, following a line of work developed by Wang and Ling [27,28,29], in which Klein's algorithm [13], a randomized nearest-plane algorithm, is used as the proposal distribution and MCMC drives convergence towards the target lattice Gaussian, with convergence rate governed by the spectral gap they derive analytically. Naturally, one might first consider the quantum-walk-based methods [25,14] to accelerate the Markov chains. In particular, Wocjan et al. [30] give an end-to-end framework for quantum sampling based on quantum walks, which can be interpreted as a quantum simulated annealing procedure. However, applying this approach directly to the Markov chain introduces additional complex structures: one must

specify a cooling schedule, control the overlaps between successive target distributions, and obtain usable lower bounds on the spectral gaps of a sequence of Markov chains at different inverse temperatures, equivalently, different widths of lattice Gaussian distributions. These requirements make the resulting approach considerably more involved, and the overhead due to the requirements can offset the quadratic speedup one would hope to obtain. Our key observation is that one does not need to accelerate the chain itself. Instead, the relevant structure is already present in the Klein proposal distribution: it pointwise lower-bounds the target lattice Gaussian distribution up to a multiplicative constant. This is precisely the domination condition required for rejection sampling, whose classical form goes back to von Neumann [18]. Under such a domination bound, the classical cost of producing one accepted target sample is proportional to the inverse of this constant; related domination-based ideas for lattice Gaussian proposals also appear in [5]. Ozols et al. [19] subsequently proposed a quantum analogue, quantum rejection sampling (QRS), which uses amplitude amplification to improve the dependence on this constant quadratically. This perspective has also been used in quantum state-generation problems, including the quantum Metropolis sampling algorithm [26]. This work targets the gap left open by these results: the cost of the Gaussian sampling step, which remains classical and now constitutes the dominant bottleneck of the attack. The urgency of this gap has only grown since [21] further reduced the cost of the guessing stage, widening the disparity between the quantum-accelerated and classical components of the pipeline. We seek to close this gap by giving a quantum algorithm for the sampling step itself, thereby reducing the overall dual attack complexity.

Contributions. The main contribution of this work is a quantum algorithm for the discrete Gaussian sampling step of the dual attack framework in [20], replacing the classical MCMC sampler with a QRS procedure and yielding a quadratic reduction in the Δ term (16). First, we establish a lower bound on the truncated variant of Klein's probability distribution, a prerequisite for embedding it as the proposal oracle in QRS. We then instantiate QRS with this truncated Klein oracle to sample from a truncated lattice Gaussian whose total variation distance from the full lattice Gaussian is made negligible by an appropriate choice of truncation radius. Finally, we integrate this quantum sampler into the dual attack pipeline and tabulate the resulting end-to-end attack costs for Kyber-512, Kyber-768, and Kyber-1024, both without and with modulus switching; see Table 1 and Table 2. This gives lower overall dual-attack cost estimates. We state our main result here informally:

Theorem 1 (Informal). *Let $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$ and $\mathcal{X}_\mathcal{R} := \{\mathbf{x} \in \mathbb{Z}^n : \|\mathbf{Bx} - \mathbf{c}\| \leq \mathcal{R}\}$. Given access to the truncated Klein proposal oracle*

$$O_{Q_\mathcal{R}} : |0\rangle \mapsto \sum_{\mathbf{x} \in \mathcal{X}_\mathcal{R}} \sqrt{Q_\mathcal{R}(\mathbf{x})}|\mathbf{x}\rangle, \tag{1}$$

its inverse $O_{Q_{\mathcal{R}}}^{\dagger}$, a rotation R_a operator. Given the Proposition 1 gives the pointwise domination bound

$$Q_{\mathcal{R}}(\mathbf{x}) \geq p_{\mathcal{R}} \pi_{\mathcal{R}}(\mathbf{x}), \qquad p_{\mathcal{R}} := \frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})} \in (0, 1]. \tag{2}$$

Then QRS prepares the truncated lattice Gaussian state $|\pi_{\mathcal{R}}\rangle = \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{\pi_{\mathcal{R}}(\mathbf{x})} |\mathbf{x}\rangle$ using

$$\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right) \tag{3}$$

oracle queries, where $\Delta_{\mathcal{R}} = \dfrac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_{\mathcal{R}})}{\prod_{i=1}^{n} \rho_{s_i}(\mathbb{Z})}$ is the pointwise ratio between the truncated Klein distribution $Q_{\mathcal{R}}$ and the lattice Gaussian distribution $\pi_{\mathcal{R}}$, as defined in Proposition 1. The total number of qubits is $\lceil \log_2 |\mathcal{X}_{\mathcal{R}}| \rceil + 1$.

Measurement in the computational basis outputs a sample $\mathbf{Bx}$ from the truncated lattice Gaussian. We then use this result to speed up the modern dual attack framework in the following theorem.

Theorem 2 (Informal). *Under the same parameter setting as Theorem 6 of [20], with a set of lattice points lying inside the Euclidean ball of radius $\mathcal{R} \geq qs\left(\sqrt{\frac{m}{2\pi}} + \sqrt{\frac{\log(N/\varepsilon_{\text{tail}})}{\pi}}\right)$, there exists an algorithm using QRS such that the end-to-end complexity for the dual attack is*

$$\text{poly}(m, n)(N + q^{n_{\text{guess}}}) + T_{\text{BKZ}}(m, \beta) + N\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right). \tag{4}$$

Since $\Delta_{\mathcal{R}} = \alpha_{\mathcal{R}} \Delta$, where $\alpha_{\mathcal{R}}$ denotes the Gaussian mass captured by the ball, Corollary 1 lets us choose $\mathcal{R}$ so that $\alpha_{\mathcal{R}} \to 1$. Here Δ denotes the main complexity term incurred by MCMC in [20] (cf. (62)). Hence $\Delta_{\mathcal{R}} \to \Delta$, and substituting this into (4) yields a lower overall dual attack cost via the quadratic improvement in the sampling complexity.

Beyond its application to attacks, we demonstrate a constructive use case for the proposed QRS algorithm: accelerating GPV trapdoor sampling. The foundational technique of the GPV signature scheme is discrete Gaussian sampling over a trapdoor lattice [11]. To prevent private key leakage, its security critically depends on the output distribution of the discrete Gaussian sampling remaining completely oblivious to the specific basis used during the process. While the original GPV scheme relied on Klein's algorithm, any sufficiently secure Gaussian sampling algorithm can be adapted for GPV signatures. In particular, Wang and Ling [29] proposed an MCMC sampler based on the independent MHK algorithm for GPV-style trapdoor sampling.

As shown in [11], the security of GPV signing reduces to the hardness of the Inhomogeneous Short Integer Solution (ISIS) problem with an approximation factor of $\sqrt{n/(2\pi)} \cdot s$. Consequently, the parameter s is the defining characteristic of a discrete Gaussian sampler in this context. This introduces a tradeoff between

security and computational efficiency when using MCMC [29] for trapdoor sampling: a smaller parameter s enhances security but demands a longer running time. In particular, above smoothing, if $B_{\max} = \max_i \|\widehat{\mathbf{b}}_i\|$ and $s \geq \sqrt{\gamma} B_{\max}$ for $\gamma \geq 1$, their bound [29, Eq. 77] gives an inverse-gap at most

$$T_{\mathrm{MCMC}} \leq \vartheta_3(\gamma)^n (1 + 2\varepsilon),$$

and hence mixing time, where d is the lattice dimension and ϑ_3 is the Jacobi theta function.

Our QRS sampler can replace this MCMC Gaussian sampler. Under the same truncation argument as in Theorem 5, the sampling factor is quadratically reduced, giving the corresponding bound

$$T_{\mathrm{QRS}} \lesssim \vartheta_3(\gamma)^{n/2} \sqrt{1 + 2\varepsilon}.$$

For the FALCON-512 parameters considered by Wang and Ling, this changes the $\gamma = 2$ upper bound from approximately 45.49 to 6.74.

Related and concurrent work. We became aware of concurrent and independent work by Chevignard et al. [8], who also use quantum rejection sampling for discrete Gaussian sampling over lattices. Their work gives an explicit coordinate-wise quantum implementation of a Klein-type proposal sampler over a finite box-truncated support, and then applies quantum rejection sampling to transform this proposal state into a state whose measurement distribution approximates the corresponding finite lattice Gaussian. They further develop two applications: a quantum speedup for the modern dual-attack framework and a quantum speedup for the Gaussian-sampling component of the SIS algorithm proposed by Bollauf et al. [4]. In particular, their full quantum dual-attack framework avoids QRACM by replacing the precomputed list oracle with a quantum state for a discrete Gaussian, combined with quantum mean estimation based on the previous work of [3]. The QRACM-free variant naturally becomes feasible once one has a quantum algorithm to prepare a quantum state for a lattice Gaussian distribution, since this state can be used directly inside the quantum mean estimation subroutine instead of relying on QRACM to load a precomputed list.

Our work differs in scope and level of abstraction. We formulate the quantum rejection sampling step as an oracle and high-level circuit construction for performing lattice Gaussian sampling. Instead of giving a coordinate-by-coordinate implementation over the box-truncated region, we work with a spherical truncation. We then apply this construction to speed up the sampling subroutine of the modern dual attack framework of [20], including table comparisons between our attack-cost model and theirs under the same parameter constraints, both with and without modulus switching. Finally, we apply our quantum sampler to FALCON's trapdoor sampling, obtaining a quadratic speedup in signing, which we verify with numerical simulations.

Paper organization. Section 2 gives the basic preliminaries on lattices, LWE, discrete Gaussians, Klein's algorithm, the MCMC time complexity, and the

quantum primitives used in our quantum algorithm. In Section 3, we establish the pointwise domination bound needed for QRS and use it to obtain a quantum lattice Gaussian sampler with quadratically improved sampling complexity, up to negligible truncation error. In Section 4, we recall the modern dual-attack framework of [20] and incorporate our quantum sampler into the sampling component of their complexity model. We then compare the resulting dual-attack estimates with Table 1 of [20], using the same parameters, both with and without modulus switching. In Section 5, we apply the proposed QRS algorithm to accelerate trapdoor sampling. Finally, Section 6 concludes the paper.

2 Preliminaries

We provide the following definitions and results we will use in our main theorems:

2.1 Classical Preliminaries

Definition 1 (Lattice). *A lattice $\mathcal{L} \subset \mathbb{R}^n$ is a discrete additive subgroup generated by the set of integer combinations of n linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$, where $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n] \in \mathbb{R}^{n \times n}$:*

$$\mathcal{L}(\mathbf{B}) = \left\{ \sum_{i=1}^{n} x_i \mathbf{b}_i : x_i \in \mathbb{Z} \right\}. \tag{5}$$

Definition 2 (Dual Lattice). *Let $\mathcal{L} \subseteq \mathbb{R}^n$ be a full-rank lattice. The dual lattice $\mathcal{L}^*$ is defined as*

$$\mathcal{L}^* = \{ \mathbf{y} \in \mathbb{R}^n : \langle \mathbf{y}, \mathbf{x} \rangle \in \mathbb{Z} \text{ for all } \mathbf{x} \in \mathcal{L} \}, \tag{6}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product in $\mathbb{R}^n$.

Definition 3 (q-ary lattices). *For a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, the q-ary lattices associated with $\mathbf{A}$ are*

$$\begin{aligned} \mathcal{L}_q(\mathbf{A}) &= \left\{ \mathbf{z} \in \mathbb{Z}^m : \exists \mathbf{x} \in \mathbb{Z}_q^n,\ \mathbf{z} \equiv \mathbf{A}\mathbf{x} \pmod{q} \right\}, \\ \mathcal{L}_q^{\perp}(\mathbf{A}) &= \left\{ \mathbf{z} \in \mathbb{Z}^m : \mathbf{A}^{\top}\mathbf{z} \equiv 0 \pmod{q} \right\}. \end{aligned} \tag{7}$$

Remark 1. Although $\mathcal{L}_q^{\perp}(\mathbf{A})$ is standardly called the dual q-ary lattice, it is not identical to the Euclidean dual lattice $\mathcal{L}_q(\mathbf{A})^*$ from Definition 2. Rather, the two are related by

$$\mathcal{L}_q(\mathbf{A})^* = \frac{1}{q}\mathcal{L}_q^{\perp}(\mathbf{A}). \tag{8}$$

Thus $\mathcal{L}_q^{\perp}(\mathbf{A})$ denotes the integer lattice of vectors orthogonal to the columns of $\mathbf{A}$ modulo q.

Definition 4 (Learning with Errors [22]). *Let $m, n, q \in \mathbb{N}$, and let χ be a distribution over $\mathbb{Z}_q$. The decisional* $\mathrm{LWE}_{n,q,\chi}$ *problem with m samples is to distinguish $(\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e})$ from $(\mathbf{A}, \mathbf{u})$, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}$, $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, $\mathbf{e} \leftarrow \chi^m$, and $\mathbf{u} \leftarrow \mathbb{Z}_q^m$.*

Definition 5 (Discrete Gaussian Distribution). *For any $\mathcal{L} \subset \mathbb{R}^n$, $s > 0$, and $\mathbf{c} \in \mathbb{R}^n$, the discrete Gaussian distribution over $\mathcal{L}$ is*

$$\mathcal{D}_{\mathcal{L}+\mathbf{c},s}(\mathbf{x}) = \frac{\rho_{s,\mathbf{c}}(\mathbf{x})}{\rho_{s,\mathbf{c}}(\mathcal{L})} \tag{9}$$

for all $\mathbf{x} \in \mathcal{L}$, where $\rho_{s,\mathbf{c}}(\mathbf{x}) = \exp(-\pi \|\mathbf{x}-\mathbf{c}\|^2 / s^2)$ and $\rho_{s,\mathbf{c}}(\mathcal{L}) = \sum_{\mathbf{y}\in\mathcal{L}} \rho_{s,\mathbf{c}}(\mathbf{y})$.

See Figure 1 for discrete Gaussians with different width s, where s controls the sampling complexity. Most algorithms sample at or above the smoothing parameter [16].



(a) $s = 10$ (b) $s = 4$

Fig. 1: Discrete Gaussian mass on $\mathbb{Z}^2 \cap [-15, 15]^2$ with Gaussian parameter $s = 10$ (left) and $s = 4$ (right). The plotted values show the unnormalized mass underlying $\mathcal{D}_{\mathbb{Z}^2,s,\mathbf{0}}(\mathbf{x}) = \rho_{s,\mathbf{0}}(\mathbf{x})/\rho_{s,\mathbf{0}}(\mathbb{Z}^2)$.

Corollary 1 ([24]). *Let $\mathcal{L} \subset \mathbb{R}^m$ be a lattice, let $\mathbf{c} \in \mathbb{R}^m$, let $s > 0$ and let $\varepsilon_{\text{tail}} \in (0, 1)$. If*

$$\mathcal{R} \geq s\left(\sqrt{\frac{m}{2\pi}} + \sqrt{\frac{\log(1/\varepsilon_{\text{tail}})}{\pi}}\right), \tag{10}$$

then

$$\rho_s\left((\mathcal{L}-\mathbf{c}) \setminus \mathcal{B}_m(\mathcal{R})\right) \leq \varepsilon_{\text{tail}} \rho_s(\mathcal{L}). \tag{11}$$

Here $\mathcal{B}_m(\mathcal{R})$ denotes the radius-$\mathcal{R}$ Euclidean ball in $\mathbb{R}^m$.

Definition 6 (Klein Probability Distribution, [13]). *Let $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$, where $\mathbf{B} = \mathbf{QR}$ is the QR decomposition, and set $\mathbf{c}' = \mathbf{Q}^\top \mathbf{c}$. For $\mathbf{x} \in \mathbb{Z}^n$, define*

$$s_i := \frac{s}{|r_{ii}|}, \qquad \mu_i(\mathbf{x}_{>i}) := \frac{c'_i - \sum_{j>i} r_{ij} x_j}{r_{ii}}. \tag{12}$$

Then the Klein probability distribution is

$$Q(\mathbf{x}) := \prod_{i=1}^{n} \frac{\rho_{s_i,\mu_i(\mathbf{x}_{>i})}(x_i)}{\rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})} = \frac{\rho_{s,\mathbf{c}}(\mathbf{Bx})}{\prod_{i=1}^{n} \rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})}, \tag{13}$$

where $\rho_{s,\mathbf{c}}(\mathbf{Bx}) = \prod_{i=1}^{n} \rho_{s_i,\mu_i(\mathbf{x}_{>i})}(x_i)$.

The product form in Definition 6 is exactly the distribution generated by Klein's Algorithm (see Algorithm 1), in the form later used by Gentry et al. [11] with a short basis to sample from lattice Gaussian distributions. Given $\mathbf{B} = \mathbf{QR}$ and $\mathbf{c}' = \mathbf{Q}^\top \mathbf{c}$, the sampler generates the coefficient vector $\mathbf{x} \in \mathbb{Z}^n$ sequentially in reverse order from x_n down to x_1. At each step i, after the higher-index coordinates $x_{i+1}, \ldots, x_n$ have been fixed, it samples $x_i \leftarrow \mathcal{D}_{\mathbb{Z}, s_i, \mu_i(\mathbf{x}_{>i})}$. Multiplying the conditional probabilities over $i = n, \ldots, 1$ gives precisely $Q(\mathbf{x})$. Thus Algorithm 1 samples $\mathbf{x}$ from Q in the coefficient space and outputs the corresponding lattice vector $\mathbf{Bx}$.

Algorithm 1: Klein's Algorithm [13]

Input: $\mathbf{B}, s, \mathbf{c}$
Output: $\mathbf{Bx} \in \mathcal{L}$
Let $\mathbf{B} = \mathbf{QR}$ and $\mathbf{c}' = \mathbf{Q}^\top \mathbf{c}$
for $i = n, \ldots, 1$ **do**
  Let $s_i = \frac{s}{|r_{ii}|}$ and $\mu_i(\mathbf{x}_{>i}) = \frac{c'_i - \sum_{j>i} r_{ij} x_j}{r_{ii}}$
  Sample x_i from $\mathcal{D}_{\mathbb{Z}, s_i, \mu_i(\mathbf{x}_{>i})}$
return $\mathbf{Bx}$

Definition 7 (Total Variation Distance). *Let μ and ν be two probability distributions on a countable state space Ω. The total variation distance between μ and ν is*

$$d_{\mathrm{TV}}(\mu, \nu) := \frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \nu(x)|. \tag{14}$$

Theorem 3 ([29, Theorem 1]). *Let $\mathcal{L} \subset \mathbb{R}^n$ be a lattice with basis $\mathbf{B}$, let $s > 0$, and let $\varepsilon > 0$. There exists an algorithm that outputs a sample according to a distribution $\mathcal{D}^{\varepsilon}_{\mathcal{L},s}$ satisfying $d_{\mathrm{TV}}\left(\mathcal{D}^{\varepsilon}_{\mathcal{L},s}, \mathcal{D}_{\mathcal{L},s}\right) \le \varepsilon$. The algorithm runs in time*

$$\ln\left(\frac{1}{\varepsilon}\right) \cdot \frac{1}{\Delta} \cdot \mathrm{poly}(n), \tag{15}$$

where

$$\frac{1}{\Delta} = \frac{\prod_{i=1}^{n} \rho_{s/\|\widehat{\mathbf{b}}_i\|}(\mathbb{Z})}{\rho_s(\mathcal{L})}, \tag{16}$$

and $\widehat{\mathbf{b}}_1, \ldots, \widehat{\mathbf{b}}_n$ are the Gram–Schmidt vectors of $\mathbf{B}$.

In [29], the MCMC algorithm performs lattice Gaussian sampling by embedding Klein's algorithm into a Metropolis–Hastings (MH) framework, where $\mathcal{D}_{\mathcal{L}+\mathbf{c},s}$ is set as the target distribution. Specifically, a candidate state is generated by Klein's algorithm, followed by the MH acceptance condition to either accept or reject it.

2.2 Quantum Preliminaries

For the quantum part, we require the following results:

Definition 8 (Quantum Sample, [1]). *For a probability distribution ν on a finite or countable state space Ω, its coherent quantum encoding is*

$$|\nu\rangle = \sum_{x\in\Omega} \sqrt{\nu(x)}|x\rangle. \tag{17}$$

Measuring $|\nu\rangle$ in the computational basis returns $x \in \Omega$ with probability $\nu(x)$, so $|\nu\rangle$ is the quantum analogue of a classical sample from ν.

Theorem 4 (Amplitude amplification [7]). *Let $\mathcal{A}$ be a unitary such that*

$$\mathcal{A}|0\rangle = \sqrt{p}|\psi\rangle + \sqrt{1-p}|\psi^{\perp}\rangle, \tag{18}$$

where $p \in (0,1]$, $|\psi\rangle \in \mathrm{im}(\Pi)$ and $|\psi^{\perp}\rangle \in \mathrm{im}(\mathbb{I}-\Pi)$ where Π is the target subspace. Let $S_0 := \mathbb{I} - 2|0\rangle\langle 0|$ be the reflection across input state $|0\rangle$, and $S_\Pi := \mathbb{I} - 2\Pi$ be the reflection across the target subspace. Then amplitude amplification is defined by the following operations:

$$\mathcal{U} := -\mathcal{A}\, S_0\, \mathcal{A}^{\dagger}\, S_\Pi \tag{19}$$

Ozols et al. [19] introduced quantum rejection sampling as a quantum analogue of classical rejection sampling. Quantum rejection sampling uses amplitude amplification to boost the accepted branch during the procedure. Given a unitary oracle that prepares $O_\nu : |0\rangle \mapsto |\nu\rangle$, if the target distribution π satisfies $\nu(x) \geq p\pi(x)$ for all $x \in \Omega$, where $p > 0$, and a coherent implementation of the acceptance probability $p\pi(x)/\nu(x)$ is available, then QRS prepares the target quantum sample $|\pi\rangle$ using quadratically fewer repetitions than classical rejection sampling. We use the following specialization.

Lemma 1 (Lemma 4.1, [19], restated). *Let Ω be a finite state space and let ν and π be probability distributions on Ω. Suppose that for some $p \in (0,1]$, $\nu(x) \geq p\pi(x)$ for every $x \in \Omega$. Assume access to a unitary preparation oracle $O_\nu : |0\rangle \mapsto |\nu\rangle = \sum_{x\in\Omega} \sqrt{\nu(x)}|x\rangle$, and to a controlled rotation implementing the acceptance probability*

$$a(x) := p\frac{\pi(x)}{\nu(x)}. \tag{20}$$

Then quantum rejection sampling prepares

$$|\pi\rangle = \sum_{x\in\Omega} \sqrt{\pi(x)}|x\rangle \tag{21}$$

using $\mathcal{O}(1/\sqrt{p})$ queries to O_ν, $O_\nu^{\dagger}$, and the controlled-rotation oracle.

Remark 2. A caveat is that standard amplitude amplification requires knowledge of the success probability p or the rotation angle, in order to choose the number of reflections without overshooting. If p is not known exactly but a certified lower bound $p_{\min} \leq p$ is available, one can instead use fixed-point amplitude amplification [31], which uses

$$\mathcal{O}\left(\frac{\log(1/\varepsilon)}{\sqrt{p_{\min}}}\right)$$

iterations to prepare the target state up to error ε. Thus the quadratic improvement over classical rejection sampling is retained with respect to the known lower bound $p_{\min}$. This construction avoids overshooting by using phase sequences derived from Chebyshev polynomials and results in monotonic convergence toward the target subspace.

3 Quantum Algorithm for Lattice Gaussian Sampling

We now derive the lower bound required for preparing a lattice Gaussian from the Klein probability distribution. We use truncated versions of both distributions, since embedding them into a meaningful quantum state requires finite support.

Proposition 1. *Let $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$ and let $\mathcal{X}_\mathcal{R} := \{\mathbf{x} \in \mathbb{Z}^n : \|\mathbf{B}\mathbf{x} - \mathbf{c}\| \leq \mathcal{R}\}$ be a coefficient-space truncation such that*

$$\pi_\mathcal{R}(\mathbf{x}) = \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathbf{x})}{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R})}, \qquad Q_\mathcal{R}(\mathbf{x}) = \frac{Q(\mathbf{x})}{Q(\mathcal{X}_\mathcal{R})}$$

where $Q(\mathcal{X}_\mathcal{R}), \rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R}) > 0$, with $Q(\mathcal{X}_\mathcal{R}) := \sum_{\mathbf{x}\in\mathcal{X}_\mathcal{R}} Q(\mathbf{x})$ and $\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R}) := \sum_{\mathbf{x}\in\mathcal{X}_\mathcal{R}} \rho_{s,\mathbf{c}}(\mathbf{B}\mathbf{x})$. Let $\pi_\mathcal{R}$ denote the truncated lattice Gaussian on $\mathcal{X}_\mathcal{R}$, and let $Q_\mathcal{R}$ denote the normalized restriction of the Klein probability distribution to $\mathcal{X}_\mathcal{R}$; both distributions are zero outside $\mathcal{X}_\mathcal{R}$. Then for every $\mathbf{x} \in \mathbb{Z}^n$,

$$Q_\mathcal{R}(\mathbf{x}) \geq p_\mathcal{R}\pi_\mathcal{R}(\mathbf{x}), \qquad p_\mathcal{R} := \frac{\Delta_\mathcal{R}}{Q(\mathcal{X}_\mathcal{R})} \in (0, 1], \tag{22}$$

where

$$\Delta_\mathcal{R} = \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R})}{\prod_{i=1}^n \rho_{s_i}(\mathbb{Z})}. \tag{23}$$

Proof. Let $\mathbf{x} \in \mathbb{Z}^n$. If $\mathbf{x} \notin \mathcal{X}_\mathcal{R}$, then $\pi_\mathcal{R}(\mathbf{x}) = Q_\mathcal{R}(\mathbf{x}) = 0$, proving the claim. It remains to consider $\mathbf{x} \in \mathcal{X}_\mathcal{R}$. Using the definitions of $Q_\mathcal{R}$, Q, and $\pi_\mathcal{R}$, we have

$$\begin{aligned}\frac{Q_\mathcal{R}(\mathbf{x})}{\pi_\mathcal{R}(\mathbf{x})} &= \frac{1}{Q(\mathcal{X}_\mathcal{R})} \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathbf{x})}{\prod_{i=1}^n \rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})} \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R})}{\rho_{s,\mathbf{c}}(\mathbf{B}\mathbf{x})} \\ &= \frac{1}{Q(\mathcal{X}_\mathcal{R})} \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R})}{\prod_{i=1}^n \rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})}.\end{aligned} \tag{24}$$

By the inequality $\rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z}) \leq \rho_{s_i}(\mathbb{Z})$, we get

$$\frac{Q_{\mathcal{R}}(\mathbf{x})}{\pi_{\mathcal{R}}(\mathbf{x})} \geq \frac{1}{Q(\mathcal{X}_{\mathcal{R}})} \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_{\mathcal{R}})}{\prod_{i=1}^{n} \rho_{s_i}(\mathbb{Z})} = \frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})} = p_{\mathcal{R}}. \tag{25}$$

Hence

$$Q_{\mathcal{R}}(\mathbf{x}) \geq p_{\mathcal{R}} \pi_{\mathcal{R}}(\mathbf{x}). \tag{26}$$

It remains to show that $p_{\mathcal{R}} \in (0, 1]$. By the same bound, summing over $\mathcal{X}_{\mathcal{R}}$ gives

$$Q(\mathcal{X}_{\mathcal{R}}) \geq \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_{\mathcal{R}})}{\prod_{i=1}^{n} \rho_{s_i}(\mathbb{Z})} = \Delta_{\mathcal{R}}. \tag{27}$$

Therefore, since $Q(\mathcal{X}_{\mathcal{R}}) > 0$ and $\Delta_{\mathcal{R}} > 0$,

$$0 < p_{\mathcal{R}} = \frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})} \leq 1. \tag{28}$$

□

We now instantiate Lemma 1 with a truncated Klein oracle to prepare the quantum state that encodes the truncated lattice Gaussian as the target distribution.

Definition 9 (Truncated Klein Oracle). *Let $O_{Q_{\mathcal{R}}}$ be a unitary oracle on state space $\mathcal{X}_{\mathcal{R}}$ satisfying*

$$O_{Q_{\mathcal{R}}} : |0\rangle \mapsto \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{Q_{\mathcal{R}}(\mathbf{x})} |\mathbf{x}\rangle. \tag{29}$$

Theorem 5 (QRS preparation from the truncated Klein proposal). *Let $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$, let $\mathcal{X}_{\mathcal{R}} := \{\mathbf{x} \in \mathbb{Z}^n : \|\mathbf{B}\mathbf{x} - \mathbf{c}\| \leq \mathcal{R}\}$, and let $Q_{\mathcal{R}}$ and $\pi_{\mathcal{R}}$ be normalized distributions on $\mathcal{X}_{\mathcal{R}}$ as in Proposition 1. Let $\Delta_{\mathcal{R}}$ be as in Proposition 1 and assume oracle access* (29) *and controlled rotation gate R_a that implements the acceptance probability $a(\mathbf{x})$. Then Algorithm 2 prepares the exact state $|\pi_{\mathcal{R}}\rangle$ using*

$$\mathcal{O}\left(\frac{1}{\sqrt{p_{\mathcal{R}}}}\right) \leq \mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right). \tag{30}$$

queries to $O_{Q_{\mathcal{R}}}$, $O_{Q_{\mathcal{R}}}^{\dagger}$, controlled rotations R_a, $R_a^{\dagger}$ and the reflections S_0, S_{Π}, and uses $\lceil \log_2 |\mathcal{X}_{\mathcal{R}}| \rceil + 1$ number of qubits.

Proof. Begin with the oracle

$$O_{Q_{\mathcal{R}}}|0\rangle = |Q_{\mathcal{R}}\rangle = \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{Q_{\mathcal{R}}(\mathbf{x})} |\mathbf{x}\rangle. \tag{31}$$

Using Lemma 1, for $\mathbf{x} \in \mathcal{X}_{\mathcal{R}}$, the acceptance probability can be obtained using (20):

$$
\begin{aligned}
a(\mathbf{x}) &= p_{\mathcal{R}} \frac{\pi_{\mathcal{R}}(\mathbf{x})}{Q_{\mathcal{R}}(\mathbf{x})} \\
&= \frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})} \left(\frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathbf{x})}{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_{\mathcal{R}})} \right) \left(\frac{Q(\mathcal{X}_{\mathcal{R}}) \prod_{i=1}^{n} \rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})}{\rho_{s,\mathbf{c}}(\mathbf{B}\mathbf{x})} \right) \\
&= \Delta_{\mathcal{R}} \frac{\prod_{i=1}^{n} \rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})}{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_{\mathcal{R}})} \\
&= \prod_{i=1}^{n} \frac{\rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z})}{\rho_{s_i}(\mathbb{Z})},
\end{aligned}
\tag{32}
$$

since $\rho_{s_i,\mu_i(\mathbf{x}_{>i})}(\mathbb{Z}) \leq \rho_{s_i}(\mathbb{Z})$, each factor above lies in $[0,1]$. Hence $0 \leq a(\mathbf{x}) \leq 1$, so the corresponding controlled acceptance rotation is well-defined. Now the rotation is defined as

$$
R_a : |\mathbf{x}\rangle|0\rangle \mapsto |\mathbf{x}\rangle \left(\sqrt{1 - a(\mathbf{x})}|0\rangle + \sqrt{a(\mathbf{x})}|1\rangle \right), \tag{33}
$$

where, on the subspace spanned by $\{|\mathbf{x}\rangle : \mathbf{x} \in \mathcal{X}_{\mathcal{R}}\}$, R_a is block-diagonal over $\mathbf{x}$ with

$$
R_a = \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} |\mathbf{x}\rangle\langle\mathbf{x}| \otimes \begin{pmatrix} \sqrt{1 - a(\mathbf{x})} & -\sqrt{a(\mathbf{x})} \\ \sqrt{a(\mathbf{x})} & \sqrt{1 - a(\mathbf{x})} \end{pmatrix}. \tag{34}
$$

Applying R_a to the proposal state gives

$$
R_a \left(|Q_{\mathcal{R}}\rangle|0\rangle \right) = \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{Q_{\mathcal{R}}(\mathbf{x})} \sqrt{1 - a(\mathbf{x})} |\mathbf{x}\rangle|0\rangle + \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{Q_{\mathcal{R}}(\mathbf{x})} \sqrt{a(\mathbf{x})} |\mathbf{x}\rangle|1\rangle. \tag{35}
$$

The second term is the accepted branch:

$$
\begin{aligned}
\sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{Q_{\mathcal{R}}(\mathbf{x})} \sqrt{a(\mathbf{x})} |\mathbf{x}\rangle|1\rangle &= \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{Q_{\mathcal{R}}(\mathbf{x})} \sqrt{\frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})} \frac{\pi_{\mathcal{R}}(\mathbf{x})}{Q_{\mathcal{R}}(\mathbf{x})}} |\mathbf{x}\rangle|1\rangle \\
&= \sqrt{\frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})}} \sum_{\mathbf{x} \in \mathcal{X}_{\mathcal{R}}} \sqrt{\pi_{\mathcal{R}}(\mathbf{x})} |\mathbf{x}\rangle|1\rangle \\
&= \sqrt{\frac{\Delta_{\mathcal{R}}}{Q(\mathcal{X}_{\mathcal{R}})}} |\pi_{\mathcal{R}}\rangle|1\rangle.
\end{aligned}
\tag{36}
$$

The normalized accepted state is $|\pi_{\mathcal{R}}\rangle|1\rangle$, so discarding the accept qubit gives $|\pi_{\mathcal{R}}\rangle$. Finally, taking the inverse gives $\frac{1}{\sqrt{p_{\mathcal{R}}}} \leq \sqrt{\frac{Q(\mathcal{X}_{\mathcal{R}})}{\Delta_{\mathcal{R}}}} \leq \frac{1}{\sqrt{\Delta_{\mathcal{R}}}}$, since $Q(\mathcal{X}_{\mathcal{R}}) \leq 1$. Therefore the query complexity is

$$
\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}} \right). \tag{37}
$$

Algorithm 2: Quantum Rejection Sampler for Lattice Gaussians

Input: Basis $\mathbf{B}$ of $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$, width $s > 0$, center $\mathbf{c}$, radius $\mathcal{R}$, $\mathcal{X}_\mathcal{R} \leftarrow \{\mathbf{x} \in \mathbb{Z}^n : \|\mathbf{Bx} - \mathbf{c}\| \leq \mathcal{R}\}$, oracle $O_{Q_\mathcal{R}}$, acceptance rotation R_a, $p_\mathcal{R}$, register $\mathcal{X}$ and coin register initialized in $|0\rangle_c$

Output: A lattice vector $\mathbf{Bx}$.

$p_\mathcal{R} \leftarrow \Delta_\mathcal{R}/Q(\mathcal{X}_\mathcal{R})$

Prepare $|Q_\mathcal{R}\rangle$ on register $\mathcal{X}$ via $O_{Q_\mathcal{R}}$

Apply R_a on coin register to obtain the initial state $\mathcal{A}_\mathcal{R}|0\rangle_\mathcal{X}|0\rangle_c$

for $\mathcal{O}(1/\sqrt{p_\mathcal{R}})$ times **do**

  Apply $\mathcal{U}_\mathcal{R} = \mathcal{A}_\mathcal{R} S_0 \mathcal{A}_\mathcal{R}^\dagger S_\Pi$

Measure the coin register

return Bx

Given our oracle model, we need not consider the registers required to implement $O_{Q_\mathcal{R}}$, R_a, or their inverses. The QRS circuit only uses a coefficient register and one accept coin; hence the number of qubits is

$$N_{\text{qubits}} = \lceil \log_2 |\mathcal{X}_\mathcal{R}| \rceil + 1. \tag{38}$$

□

Let $\pi(\mathbf{x}) := \mathcal{D}_{\mathcal{L}+\mathbf{c},s}(\mathbf{Bx})$. We define the retained Gaussian mass as

$$\alpha_\mathcal{R} := \sum_{\mathbf{x}\in\mathcal{X}_\mathcal{R}} \pi(\mathbf{x}) = \frac{\rho_{s,\mathbf{c}}(\mathbf{B}\mathcal{X}_\mathcal{R})}{\rho_{s,\mathbf{c}}(\mathcal{L})}. \tag{39}$$

For $\mathbf{x} \in \mathcal{X}_\mathcal{R}$, the truncated and full distributions are related by

$$\pi_\mathcal{R}(\mathbf{x}) = \frac{\pi(\mathbf{x})}{\alpha_\mathcal{R}}. \tag{40}$$

Therefore,

$$\begin{aligned} d_{\text{TV}}(\pi, \pi_\mathcal{R}) &= \frac{1}{2} \sum_{\mathbf{x}\in\mathcal{X}_\mathcal{R}} |\pi(\mathbf{x}) - \pi_\mathcal{R}(\mathbf{x})| + \frac{1}{2} \sum_{\mathbf{x}\notin\mathcal{X}_\mathcal{R}} \pi(\mathbf{x}) \\ &= \frac{1}{2} \sum_{\mathbf{x}\in\mathcal{X}_\mathcal{R}} \pi(\mathbf{x}) \left(\frac{1}{\alpha_\mathcal{R}} - 1\right) + \frac{1}{2} \sum_{\mathbf{x}\notin\mathcal{X}_\mathcal{R}} \pi(\mathbf{x}) \\ &= \frac{1}{2} \left(\frac{1}{\alpha_\mathcal{R}} - 1\right) \alpha_\mathcal{R} + \frac{1}{2}(1 - \alpha_\mathcal{R}) \\ &= 1 - \alpha_\mathcal{R}. \end{aligned} \tag{41}$$

Thus, choosing $\mathcal{R}$ according to Corollary 1 makes $\alpha_\mathcal{R} \to 1$, so $1 - \alpha_\mathcal{R}$ becomes negligible and the truncated sampler is statistically close to the full lattice Gaussian. Hence, measuring this quantum state outputs $\mathbf{Bx}$ with $\mathbf{x} \leftarrow \pi_\mathcal{R}$. Additionally, if $\alpha_\mathcal{R} \to 1$, then $1/\sqrt{\Delta_\mathcal{R}}$ approaches $1/\sqrt{\Delta}$, giving the quadratic reduction in the Δ term that dominates the MCMC sampling cost. Algorithm 2 contains details of our quantum sampler, and Figure 2 shows one instance of the QRS circuit.

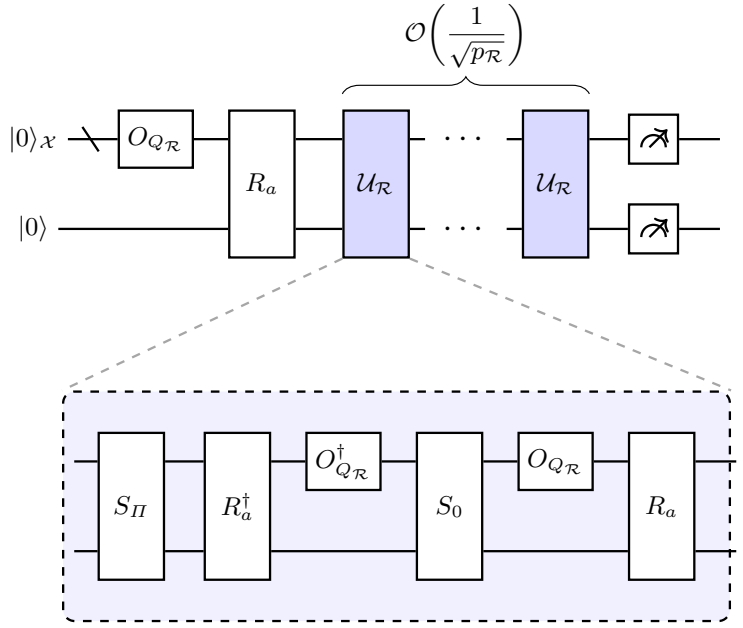

Fig. 2: Oracle-level decomposition of our quantum sampler. The top wire is the coefficient register X, and the bottom wire is the coin register. The first two gates $\mathcal{A}_{\mathcal{R}} = R_a(O_{Q_{\mathcal{R}}} \otimes \mathbb{I}_0)$ prepare the state for QRS. The operator $\mathcal{U}_{\mathcal{R}}$ is one iterate of the amplitude amplification subroutine, whose dashed box gives its full decomposition. This is repeated $\mathcal{O}\left(\frac{1}{\sqrt{p_{\mathcal{R}}}}\right)$ times in order to converge to the target state $|\pi_{\mathcal{R}}\rangle$. Measuring coin register c outputs a lattice vector $\mathbf{Bx}$ from the truncated lattice Gaussian distribution.

4 Quantum Speedup of Modern Dual Attack

We first recall the modern dual attack of [20]. Let q be a prime power and let $\mathbf{A} \in \mathbb{Z}_q^{m\times n}$, $\mathbf{s} \in \mathbb{Z}_q^n$, $\mathbf{e} \in \mathbb{Z}^m$, an LWE instance is

$$\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \pmod q. \tag{42}$$

Fix a split $n = n_{\text{guess}} + n_{\text{dual}}$. After permuting the columns of $\mathbf{A}$ so that the first n_{guess} columns correspond to the coordinates to be guessed, write

$$\mathbf{A} = \begin{pmatrix}\mathbf{A}_{\text{guess}} & \mathbf{A}_{\text{dual}}\end{pmatrix}, \qquad \mathbf{s} = \begin{pmatrix}\mathbf{s}_{\text{guess}} \\ \mathbf{s}_{\text{dual}}\end{pmatrix}, \tag{43}$$

where $\mathbf{A}_{\text{guess}} \in \mathbb{Z}_q^{m\times n_{\text{guess}}}$, $\mathbf{A}_{\text{dual}} \in \mathbb{Z}_q^{m\times n_{\text{dual}}}$, $\mathbf{s}_{\text{guess}} \in \mathbb{Z}_q^{n_{\text{guess}}}$, and $\mathbf{s}_{\text{dual}} \in \mathbb{Z}_q^{n_{\text{dual}}}$. Substituting into (42) gives

$$\mathbf{b} = \mathbf{A}_{\text{guess}}\mathbf{s}_{\text{guess}} + \mathbf{A}_{\text{dual}}\mathbf{s}_{\text{dual}} + \mathbf{e} \pmod q. \tag{44}$$

For any matrix $\mathbf{M} \in \mathbb{Z}_q^{m\times k}$, we use the associated q-ary lattice and its dual from Definition 3. The dual attack samples a list of short vectors $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ from the discrete Gaussian $\mathcal{D}_{\mathcal{L}_q^\perp(\mathbf{A}_{\text{dual}}),\, qs}$ on the dual lattice, where s is the Gaussian width parameter. Their sampling is performed in two stages: a basis of $\mathcal{L}_q^\perp(\mathbf{A}_{\text{dual}})$ is BKZ-reduced with block size β, then the MCMC algorithm of [29] is used to draw N samples. For each candidate guess $\widetilde{\mathbf{s}}_{\text{guess}} \in \mathbb{Z}_q^{n_{\text{guess}}}$, as in the proof of [20, Theorem 6],

$$\mathbf{b} - \mathbf{A}_{\text{guess}}\widetilde{\mathbf{s}}_{\text{guess}} = \mathbf{A}_{\text{dual}}\mathbf{s}_{\text{dual}} + \mathbf{A}_{\text{guess}}(\mathbf{s}_{\text{guess}} - \widetilde{\mathbf{s}}_{\text{guess}}) + \mathbf{e} \pmod q. \tag{45}$$

Since $\mathbf{w}_j \in \mathcal{L}_q^\perp(\mathbf{A}_{\text{dual}})$, the term involving $\mathbf{A}_{\text{dual}}\mathbf{s}_{\text{dual}}$ vanishes modulo q in the inner product $\langle \mathbf{w}_j,\, \mathbf{b} - \mathbf{A}_{\text{guess}}\widetilde{\mathbf{s}}_{\text{guess}}\rangle$. The resulting cosine score is precisely the empirical score used in [20, Algorithm 2], the geometric separation condition guaranteeing that the correct guess maximizes this score is therefore inherited from [20, Theorem 6]. We write this empirical score as

$$g_{\mathbf{W}}(\mathbf{y}) = \frac{1}{N}\sum_{j=1}^{N} \cos\left(\frac{2\pi\,\langle \mathbf{w}_j, \mathbf{y}\rangle}{q}\right) \tag{46}$$

For each candidate guess, the score is

$$F(\widetilde{\mathbf{s}}_{\text{guess}}) = N g_{\mathbf{W}}\left(\mathbf{A}_{\text{guess}}(\mathbf{s}_{\text{guess}} - \widetilde{\mathbf{s}}_{\text{guess}}) + \mathbf{e}\right), \tag{47}$$

hence maximizing F is equivalent to maximizing $g_{\mathbf{W}}$.

Lemma 2 (Lemma 9 of [20]). *Let $\mathbf{M} \in \mathbb{Z}_q^{m\times k}$, let $\mathcal{L} \subset \mathbb{Z}^m$ be a lattice, let $\mathbf{e} \in \mathbb{Z}^m$, let $s, \delta > 0$, and let $N \in \mathbb{N}$. Let $\tau = \frac{1}{s}\sqrt{m/(2\pi)}$ and $\eta \geq 0$, and assume that $\lambda_1(\mathcal{L} + \mathcal{L}_q(\mathbf{M})) \geq \tau + \|\mathbf{e}\|$ and*

$$\rho_{1/s}(\mathbf{e}) - \rho_{1/s}(\lambda_1(\mathcal{L} + \mathcal{L}_q(\mathbf{M})) - \|\mathbf{e}\| - \tau) > 2\delta + \eta.$$

Then, with probability at least $1 - q^m \cdot 2^{-\Omega(N\delta^2)}$ over the choice of $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ from $\mathcal{D}^N_{\mathcal{L}_q^\perp(\mathbf{M}),qs}$, we have

$$g_{\mathbf{W}}(\mathbf{e}) \geq \rho_{1/s}(\mathbf{e}) - \delta > \rho_{1/s}(\lambda_1(\mathcal{L} + \mathcal{L}_q(\mathbf{M})) - \|\mathbf{e}\| - \tau) + \delta + \eta \geq g_{\mathbf{W}}(\mathbf{e} + \mathbf{x}) + \eta$$

for all $\mathbf{x} \in \mathcal{L} \setminus \mathcal{L}_q(\mathbf{M})$, where $g_{\mathbf{W}}$ is defined in (46).

The analysis of [20] introduces a parameter $\delta > 0$ that quantifies how the score separates the correct guess from incorrect ones: under a precondition relating $\|\mathbf{e}\|$, $\lambda_1(\mathcal{L}_q(\mathbf{A}))$, and δ (see [20, Theorem 6] for the exact form), [20, Algorithm 2] returns $\mathbf{s}_{\text{guess}}$ with probability at least $1 - q^m \cdot 2^{-\Omega(N\delta^2)}$ over the choice of $\mathbf{W}$. The total cost of the attack is

$$T_{\text{dual}} = \text{poly}(m, n)\,(N + q^{n_{\text{guess}}}) + T_{\text{BKZ}}(m, \beta) + N \cdot T_{\text{MCMC}}\big(\mathcal{L}_q^\perp(\mathbf{A}_{\text{dual}}), qs\big). \tag{48}$$

Here $T_{\text{BKZ}}(m, \beta)$ is the cost of BKZ-β reduction in dimension m, and T_{MCMC} is the per-sample MCMC cost with mixing-time dependence as in Theorem 3. Our method bypasses MCMC by replacing it with QRS as given in Theorem 5, yielding the following improved complexity result:

Algorithm 3: QRS-accelerated modern dual attack

Input: m, $n_{\text{guess}}, n_{\text{dual}}$, prime power q, $N \in \mathbb{N}$, and LWE instance $(\mathbf{A}, \mathbf{b})$
Output: A guess for $\mathbf{s}_{\text{guess}}$, or $\perp$

$\mathbf{W}_{\mathcal{R}} = (\mathbf{w}_1, \ldots, \mathbf{w}_N) \leftarrow N$ samples generated by Algorithm 2
$\mathbf{s}_{\text{guess}} \leftarrow \perp$
$F_{\max} \leftarrow 0$
for $\widetilde{\mathbf{s}}_{\text{guess}} \in \mathbb{Z}_q^{n_{\text{guess}}}$ **do**
  $y_j \leftarrow \langle \mathbf{w}_j,\, \mathbf{b} - \mathbf{A}_{\text{guess}}\widetilde{\mathbf{s}}_{\text{guess}} \rangle \pmod q$, for $j = 1, \ldots, N$
  $F \leftarrow \sum_{j=1}^{N} \cos\left(\frac{2\pi y_j}{q}\right)$
  if $F \geq F_{\max}$ **then**
    $F_{\max} \leftarrow F$
    $\mathbf{s}_{\text{guess}} \leftarrow \widetilde{\mathbf{s}}_{\text{guess}}$
return $\mathbf{s}_{\text{guess}}$

Theorem 6. *Assume the conditions of [20, Theorem 6] hold, where* $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, $\mathbf{e} \in \mathbb{Z}^m$, $\mathbf{s} \in \mathbb{Z}_q^n$, $s, \delta > 0$, $N \in \mathbb{N}$, *and* $\varepsilon_{\text{tail}} \in (0,1)$, *with* $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$. *Assume the split of* $\mathbf{A}$ *and* $\mathbf{s}$ *is as in* (43). *Let* $\mathbf{B}$ *be a BKZ-*β*-reduced basis of* $\mathcal{L}_q^{\perp}(\mathbf{A}_{\text{dual}})$, *with Gram–Schmidt vectors* $\widehat{\mathbf{b}}_1, \ldots, \widehat{\mathbf{b}}_m$. *Set*

$$\mathcal{R} \geq qs\left(\sqrt{\frac{m}{2\pi}} + \sqrt{\frac{\log(N/\varepsilon_{\text{tail}})}{\pi}}\right). \tag{49}$$

Let $\mathcal{S}_{\mathcal{R}} := \mathcal{L}_q^{\perp}(\mathbf{A}_{\text{dual}}) \cap \mathcal{B}_m(\mathcal{R})$ *and* $\mathcal{X}_{\mathcal{R}} := \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{B}\mathbf{x} \in \mathcal{S}_{\mathcal{R}}\}$. *Let* $\mathbf{W}_{\mathcal{R}} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ *be sampled from* $\mathcal{D}^N_{\mathcal{S}_{\mathcal{R}}, qs}$ *using Theorem 5. Then, Algorithm 3 on input* $(m, n_{\text{guess}}, n_{\text{dual}}, q, N, (\mathbf{A}, \mathbf{b}))$ *returns* $\mathbf{s}_{\text{guess}}$ *with probability at least* $1 - q^m 2^{-\Omega(N\delta^2)} - \varepsilon_{\text{tail}}$ *and its end-to-end complexity under the QRS oracle model is*

$$\text{poly}(m,n)(N + q^{n_{\text{guess}}}) + T_{\text{BKZ}}(m,\beta) + N\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right). \tag{50}$$

Proof. Under the dual-attack setup (43), let $\mathcal{L}_q^{\perp} := \mathcal{L}_q^{\perp}(\mathbf{A}_{\text{dual}}) = \mathbf{B}\mathbb{Z}^m$. Since $\mathcal{X}_{\mathcal{R}} := \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{B}\mathbf{x} \in \mathcal{S}_{\mathcal{R}}\}$, applying Theorem 5 to $\mathcal{X}_{\mathcal{R}}$, with $\mathbf{c} = \mathbf{0}$ and Gaussian width qs, gives a coefficient vector $\mathbf{x} \in \mathcal{X}_{\mathcal{R}}$ distributed as

$$\pi_{\mathcal{R}}(\mathbf{x}) = \frac{\rho_{qs}(\mathbf{B}\mathbf{x})}{\rho_{qs}(\mathbf{B}\mathcal{X}_{\mathcal{R}})}. \tag{51}$$

Since $\mathbf{B}\mathcal{X}_{\mathcal{R}} = \mathcal{S}_{\mathcal{R}}$, setting $\mathbf{w} := \mathbf{B}\mathbf{x}$ gives

$$\mathcal{D}_{\mathcal{S}_{\mathcal{R}}, qs}(\mathbf{w}) = \frac{\rho_{qs}(\mathbf{w})}{\rho_{qs}(\mathcal{S}_{\mathcal{R}})}, \tag{52}$$

where $\mathbf{w} \in \mathcal{S}_{\mathcal{R}}$ is sampled from (52). The query complexity for one sample is $\mathcal{O}(\Delta_{\mathcal{R}}^{-1/2})$, where

$$\Delta_{\mathcal{R}} = \frac{\rho_{qs}(\mathcal{S}_{\mathcal{R}})}{\prod_{i=1}^{m} \rho_{qs/\|\widehat{\mathbf{b}}_i\|}(\mathbb{Z})}. \tag{53}$$

Therefore the list of N independent samples $\mathbf{W}_{\mathcal{R}} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ from $\mathcal{D}^N_{\mathcal{S}_{\mathcal{R}},qs}$ is generated using $N \cdot \mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right)$ queries. Including the BKZ cost $T_{\mathrm{BKZ}}(m,\beta)$, the complexity incurred from sampling is

$$T_{\mathrm{samp}} = T_{\mathrm{BKZ}}(m,\beta) + N\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right). \tag{54}$$

Define the retained full-Gaussian mass

$$\begin{aligned} \alpha_{\mathcal{R}} &:= \sum_{\mathbf{w}\in\mathcal{S}_{\mathcal{R}}} \mathcal{D}_{\mathcal{L}_q^\perp,qs}(\mathbf{w}) \\ &= \frac{\rho_{qs}(\mathcal{S}_{\mathcal{R}})}{\rho_{qs}(\mathcal{L}_q^\perp)}. \end{aligned} \tag{55}$$

Then for $\mathbf{w} \in \mathcal{S}_{\mathcal{R}}$, one can relate the truncated and the full distribution as

$$\mathcal{D}_{\mathcal{S}_{\mathcal{R}},qs}(\mathbf{w}) = \frac{\mathcal{D}_{\mathcal{L}_q^\perp,qs}(\mathbf{w})}{\alpha_{\mathcal{R}}}, \tag{56}$$

while for $\mathbf{w} \in \mathcal{L}_q^\perp \setminus \mathcal{S}_{\mathcal{R}}$ we get $\mathcal{D}_{\mathcal{S}_{\mathcal{R}},qs}(\mathbf{w}) = 0$. Thus, the total variation distance between $\mathcal{D}_{\mathcal{L}_q^\perp,qs}$ and $\mathcal{D}_{\mathcal{S}_{\mathcal{R}},qs}$ can be obtained by splitting the sum into $\mathcal{S}_{\mathcal{R}}$ and $\mathcal{L}_q^\perp \setminus \mathcal{S}_{\mathcal{R}}$:

$$\begin{aligned} d_{\mathrm{TV}}(\mathcal{D}_{\mathcal{L}_q^\perp,qs}, \mathcal{D}_{\mathcal{S}_{\mathcal{R}},qs}) &= \frac{1}{2}\sum_{\mathbf{w}\in\mathcal{S}_{\mathcal{R}}} \mathcal{D}_{\mathcal{L}_q^\perp,qs}(\mathbf{w}) \left|1 - \frac{1}{\alpha_{\mathcal{R}}}\right| + \frac{1}{2}\sum_{\mathbf{w}\in\mathcal{L}_q^\perp\setminus\mathcal{S}_{\mathcal{R}}} \mathcal{D}_{\mathcal{L}_q^\perp,qs}(\mathbf{w}) \\ &= \frac{1}{2}\left(\frac{1}{\alpha_{\mathcal{R}}} - 1\right)\sum_{\mathbf{w}\in\mathcal{S}_{\mathcal{R}}} \mathcal{D}_{\mathcal{L}_q^\perp,qs}(\mathbf{w}) + \frac{1}{2}\sum_{\mathbf{w}\in\mathcal{L}_q^\perp\setminus\mathcal{S}_{\mathcal{R}}} \mathcal{D}_{\mathcal{L}_q^\perp,qs}(\mathbf{w}) \\ &= \frac{1}{2}\left(\frac{1}{\alpha_{\mathcal{R}}} - 1\right)\frac{\rho_{qs}(\mathcal{S}_{\mathcal{R}})}{\rho_{qs}(\mathcal{L}_q^\perp)} + \frac{1}{2}\left(1 - \frac{\rho_{qs}(\mathcal{S}_{\mathcal{R}})}{\rho_{qs}(\mathcal{L}_q^\perp)}\right) \\ &= \frac{1}{2}\left(\frac{1}{\alpha_{\mathcal{R}}} - 1\right)\alpha_{\mathcal{R}} + \frac{1}{2}(1 - \alpha_{\mathcal{R}}) \\ &= 1 - \alpha_{\mathcal{R}}. \end{aligned} \tag{57}$$

For N independent samples, the union bound gives

$$d_{\mathrm{TV}}(\mathcal{D}^N_{\mathcal{L}_q^\perp,qs}, \mathcal{D}^N_{\mathcal{S}_{\mathcal{R}},qs}) \le N \cdot d_{\mathrm{TV}}(\mathcal{D}_{\mathcal{L}_q^\perp,qs}, \mathcal{D}_{\mathcal{S}_{\mathcal{R}},qs}) = N(1 - \alpha_{\mathcal{R}}). \tag{58}$$

Given our chosen value of $\mathcal{R}$, we can bound

$$1 - \alpha_{\mathcal{R}} = \frac{\rho_{qs}(\mathcal{L}_q^\perp \setminus \mathcal{S}_{\mathcal{R}})}{\rho_{qs}(\mathcal{L}_q^\perp)} \le \frac{\varepsilon_{\mathrm{tail}}}{N}, \tag{59}$$

hence $d_{\mathrm{TV}}(\mathcal{D}^N_{\mathcal{L}_q^\perp,qs}, \mathcal{D}^N_{\mathcal{S}_{\mathcal{R}},qs}) \le \varepsilon_{\mathrm{tail}}$. Given that [20, Algorithm 2] returns $\mathbf{s}_{\mathrm{guess}}$ with probability at least $1 - q^m 2^{-\Omega(N\delta^2)}$, define the event $\mathcal{E} := \{\text{Algorithm 3 returns } \mathbf{s}_{\mathrm{guess}}\}$.

Since the probability of any event changes by at most the total variation distance between the input distributions, we get

$$\begin{aligned}\Pr_{\mathbf{w}_{\mathcal{R}}\sim\mathcal{D}^N_{\mathcal{S}_{\mathcal{R}},qs}}[\mathcal{E}] &\geq \Pr_{\mathbf{w}\sim\mathcal{D}^N_{\mathcal{L}_q^\perp,qs}}[\mathcal{E}] - d_{\mathrm{TV}}(\mathcal{D}^N_{\mathcal{L}_q^\perp,qs},\mathcal{D}^N_{\mathcal{S}_{\mathcal{R}},qs}) \\ &\geq 1 - q^m 2^{-\Omega(N\delta^2)} - d_{\mathrm{TV}}(\mathcal{D}^N_{\mathcal{L}_q^\perp,qs},\mathcal{D}^N_{\mathcal{S}_{\mathcal{R}},qs}) \\ &\geq 1 - q^m 2^{-\Omega(N\delta^2)} - \varepsilon_{\mathrm{tail}}.\end{aligned} \tag{60}$$

Finally, since their algorithm contributes a complexity of $\mathrm{poly}(m,n)(N+q^{n_{\mathrm{guess}}})$, the total complexity of the dual attack amounts to

$$\mathrm{poly}(m,n)(N+q^{n_{\mathrm{guess}}}) + T_{\mathrm{BKZ}}(m,\beta) + N\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right). \tag{61}$$

□

Following [20], Δ is defined as

$$\Delta := \frac{\rho_{qs}(\mathcal{L}_q^\perp(\mathbf{A}_{\mathrm{dual}}))}{\prod_{i=1}^m \rho_{qs/\|\widehat{\mathbf{b}}_i\|}(\mathbb{Z})}. \tag{62}$$

By the definitions of $\alpha_{\mathcal{R}}$ and $\Delta_{\mathcal{R}}$ in (55) and (53), respectively, we have the identity

$$\Delta_{\mathcal{R}} = \alpha_{\mathcal{R}}\Delta. \tag{63}$$

Therefore the Gaussian mass outside of the truncation ball can be bounded above as

$$1-\alpha_{\mathcal{R}} \leq \frac{\varepsilon_{\mathrm{tail}}}{N} \implies \alpha_{\mathcal{R}} \geq 1 - \frac{\varepsilon_{\mathrm{tail}}}{N}, \tag{64}$$

Hence

$$\frac{1}{\sqrt{\Delta_{\mathcal{R}}}} \leq \frac{1}{\sqrt{\left(1-\frac{\varepsilon_{\mathrm{tail}}}{N}\right)\Delta}}. \tag{65}$$

Given that we choose the truncation radius according to Corollary 1, most of the Gaussian mass is covered. Since the Gaussian mass decays exponentially fast outside the chosen radius, we get $\alpha_{\mathcal{R}} \to 1$, hence $\frac{1}{\sqrt{\Delta_{\mathcal{R}}}} \to \frac{1}{\sqrt{\Delta}}$. Finally, under the above conditions, this gives a quadratic speedup in the sampling term of the dual attack framework:

$$T_{\mathrm{dual}} = \mathrm{poly}(m,n)(N+q^{n_{\mathrm{guess}}}) + T_{\mathrm{BKZ}}(m,\beta) + N\mathcal{O}\left(\frac{1}{\sqrt{\Delta}}\right). \tag{66}$$

We use the reduced complexity in (61) to estimate fixed-parameter costs for Kyber. Table 1 gives the no-modulus-switching comparison, using the same parameter choices $(m, n_{\mathrm{guess}}, n_{\mathrm{dual}}, \beta, s)$ as in [20] and replacing only the Gaussian-sampling term. The estimated costs decrease from 185 to 176, from 273 to 269,

and from 376 to 363 bits for Kyber512, Kyber768, and Kyber1024, respectively. Since Theorem 6 is stated for the no-modulus-switching Pouly–Shen framework, the modulus-switching figures in Table 2 should be read only as an indicative fixed-parameter comparison. In this case, the sampling improvement is largely absorbed by the remaining cost terms for Kyber512 and Kyber768, while Kyber1024 still drops from 279 to 261 bits.

Without Modulus Switching

	Attack Cost		Parameters				
Scheme	[20]	This Work	m	n_{guess}	n_{dual}	β	s
Kyber512	185	176	1013	15	497	550	0.200
Kyber768	273	269	1469	23	745	870	0.260
Kyber1024	376	363	2025	31	993	1230	0.270

Table 1: Comparison with the dual attack costs without modulus switching reported in [20]. All costs are base-2 logarithms. The "This Work" column keeps the same parameters $(m, n_{\text{guess}}, n_{\text{dual}}, \beta, s)$ and substitutes only the QRS Gaussian-sampling term.

Separately, since Theorem 5 prepares a quantum state encoding a dual-lattice Gaussian, our quantum sampler can be combined with the quantum dual attack framework of Albrecht and Shen [3]. In this composition, our QRS state-preparation oracle would replace the precomputed Gaussian sample list and then be used in the quantum mean estimation procedure of [3, Theorem 5] to estimate the score for each guess, enabling a fully quantum end-to-end dual attack that is QRACM-free. A more detailed treatment of this composition appears in the recent concurrent work of [8], to which we refer the reader for the full analysis.

With Modulus Switching

	Attack Cost		Parameters				
Scheme	[20]	This Work	m	n_{guess}	n_{dual}	β	s
Kyber512	141	141	763	141	371	390	0.170
Kyber768	202	201	1169	201	567	610	0.240
Kyber1024	279	261	1575	261	763	890	0.260

Table 2: Comparison with the dual attack costs with modulus switching reported in [20]. All costs are base-2 logarithms. The "This Work" column keeps the same parameters $(m, n_{\text{guess}}, n_{\text{dual}}, \beta, s)$ and substitutes only the QRS Gaussian-sampling term.

5 Quantum Speedup of Trapdoor Sampling

We now describe how the same QRS idea can be used as a replacement for the Gaussian sampling subroutine in GPV/FALCON-type trapdoor sampling. This section is not meant to give a new implementation of FALCON's signing algorithm, but rather it identifies the precise sampling step in which the MCMC sampler in [29] can be replaced by our QRS sampler.

5.1 GPV Trapdoor Sampling

In [11], key generation produces two bases of the same lattice: a hard public basis and a short private basis. The public basis is used for verification, while the short private basis is the trapdoor used for signing. Given a message, the signer obtains a coset representative by hashing, denoted here by $\mathbf{c}$, and samples a short vector from a discrete Gaussian over the corresponding coset $\mathcal{D}_{\mathcal{L}+\mathbf{c},s}$. The signature is the resulting short lattice-coset vector, and the verifier checks both the coset relation and the norm bound using the public key. The purpose of the trapdoor sampler is therefore to output a vector whose distribution is close to the target lattice Gaussian while hiding the private basis used to sample it.

FALCON [10] instantiates this GPV paradigm over NTRU lattices. Following the notation in [29], let $R = \mathbb{Z}[x]/(x^d + 1)$, where d is a power of two, and let the secret polynomials $f, g \in R$ satisfy that f is invertible. One finds $F, G \in R$ such that

$$fG - gF = q \bmod (x^d + 1).$$

The private NTRU basis is

$$\mathbf{B} = \begin{pmatrix} C(g) & -C(f) \\ C(G) & -C(F) \end{pmatrix}^T,$$

where $C(\cdot)$ denotes the negacyclic matrix associated with a polynomial. The corresponding public basis is

$$\mathbf{A} = \begin{pmatrix} -C(h) & I_d \\ qI_d & \mathbf{0}_d \end{pmatrix}^T, \qquad h = g/f \bmod q.$$

Both bases generate the same NTRU lattice $\mathcal{L} = \{(u, v) \in R^2 : u + vh = 0 \bmod q\}$, whose dimension is $n = 2d$. In the signing step, the short basis $\mathbf{B}$ is used to sample from a coset Gaussian distribution associated with the message digest.

5.2 Quantum-Accelerated GPV Sampling

We replace only the trapdoor Gaussian sampler in the GPV/FALCON signing pipeline. The key generation, hash-to-coset map, public verification, and trapdoor basis are unchanged; only the sampler for $\mathcal{D}_{\mathcal{L}+\mathbf{c},s}$ is replaced. Wang and

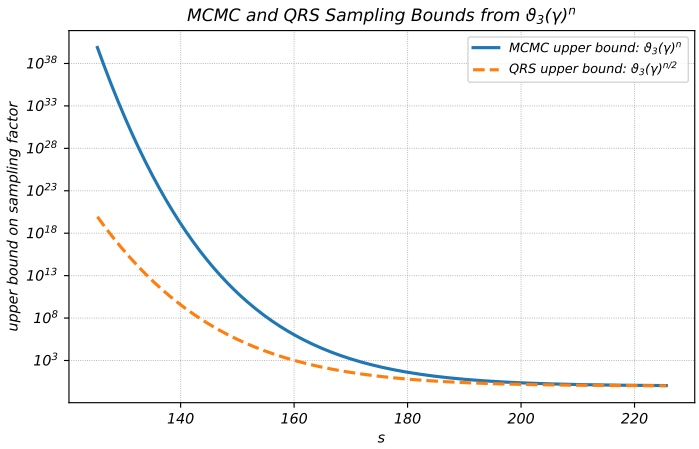

Fig. 3: MCMC and QRS upper bounds derived from [29]'s $\vartheta_3(\gamma)^n$ estimate for FALCON-512. The horizontal axis uses the Gaussian parameter s, with $\gamma = (s/B_{\max})^2$.

Ling instantiate this step with the independent MHK sampler [29]. Here we use the truncated Klein proposal as the coherent proposal oracle for QRS.

Let $\widehat{\mathbf{b}}_1, \ldots, \widehat{\mathbf{b}}_n$ be the Gram–Schmidt vectors of the trapdoor basis $\mathbf{B}$ and set $s_i = s/\|\widehat{\mathbf{b}}_i\|$, $B_{\max} = \max_i \|\widehat{\mathbf{b}}_i\|$. By Theorem 3, for the independent MHK sampler, the convergence rate is controlled by a Δ. The classical MCMC sampling cost is given by

$$T_{\mathrm{MCMC}} \asymp \frac{1}{\Delta} = \frac{\prod_{i=1}^{n} \rho_{s_i}(\mathbb{Z})}{\rho_{s,\mathbf{c}}(\mathcal{L})}. \tag{67}$$

This expression depends on the full Gram–Schmidt profile of the trapdoor basis and on the coset center $\mathbf{c}$. Wang and Ling further derive a basis-profile-independent upper bound above smoothing [29, Eq. 77]: if $s \geq \sqrt{\gamma} B_{\max}$ for some $\gamma \geq 1$, then

$$T_{\mathrm{MCMC}} \lesssim \vartheta_3(\gamma)^n (1 + 2\varepsilon), \ \ \vartheta_3(\tau) = \sum_{k \in \mathbb{Z}} e^{-\pi \tau k^2}. \tag{68}$$

In the numerical comparison below we use this latter upper bound, rather than (67).

By Theorem 5, after truncating the coefficient space so that the tail mass is negligible, QRS prepares the target coset Gaussian using $\mathcal{O}(1/\sqrt{\Delta_{\mathcal{R}}})$ oracle queries. Since $\Delta_{\mathcal{R}} \approx \Delta$ for such a truncation, the same bound gives

$$T_{\mathrm{QRS}} \lesssim \vartheta_3(\gamma)^{n/2} \sqrt{1 + 2\varepsilon}. \tag{69}$$

Thus the QRS replacement turns the MCMC sampling cost into its square root.

γ	$s/B_{\max}$	$\sigma/B_{\max}$	$\vartheta_3(\gamma)^n$	$\vartheta_3(\gamma)^{n/2}$
$\pi/2$	1.253	0.500	2.24×10^6	1.50×10^3
2	1.414	0.564	45.49	6.74
3	1.732	0.691	1.18	1.09

Table 3: Upper-bound evaluation of the MCMC and QRS sampling factors for FALCON-512. The fourth column upper-bounds T_{MCMC}, while the fifth column upper-bounds T_{QRS}. We use $n = 1024$, $s = \sqrt{2\pi}\sigma$, $B_{\max} = 127$, and $\varepsilon \approx 0$.

For FALCON-512, we take parameters $d = 512$, $q = 12289$, lattice dimension $n = 2d = 1024$, and report $B_{\max} = 127$ for one generated NTRU trapdoor basis [29, Example 1]. Taking $\varepsilon \approx 0$, the corresponding profile-free upper bounds are shown in Table 3.

Figure 3 and Table 3 should be interpreted as worst-case upper-bound comparisons depending only on $B_{\max}$. They do not reproduce the instance-dependent curve in [29, Fig. 4], which uses the full Gram–Schmidt profile (67) of a sampled FALCON key and can be much smaller than the profile-free bound in (68).

6 Conclusion

In this work, we showed that the lower-bound relation behind Klein's proposal distribution can be used as the domination condition required for quantum rejection sampling. After truncating the coefficient space, QRS prepares the truncated lattice Gaussian with query complexity

$$\mathcal{O}\left(\frac{1}{\sqrt{\Delta_{\mathcal{R}}}}\right),$$

while the truncation error can be made negligible by choosing a sufficiently large radius. We applied this sampler to the modern dual attack framework of Pouly and Shen [20]. Replacing the classical IMHK sampler by QRS changes the Gaussian-sampling contribution from a $1/\Delta$ dependence to a $1/\sqrt{\Delta_{\mathcal{R}}}$ dependence, yielding lower fixed-parameter attack-cost estimates for Kyber. We also showed that the same idea applies to GPV/FALCON trapdoor sampling using the upper bound $\vartheta_3(\gamma)^n$ from [29], for which the corresponding QRS bound becomes $\vartheta_3(\gamma)^{n/2}$.

References

1. Aharonov, D., Ta-Shma, A.: Adiabatic quantum state generation and statistical zero knowledge. In: Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing. pp. 20–29. Association for Computing Machinery (2003). https://doi.org/10.1145/780542.780546

2. Albrecht, M.R.: On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. In: Coron, J.S., Nielsen, J.B. (eds.) Advances in Cryptology – EUROCRYPT 2017, Part II. Lecture Notes in Computer Science, vol. 10211, pp. 103–129. Springer (2017). https://doi.org/10.1007/978-3-319-56614-6_4
3. Albrecht, M.R., Shen, Y.: Quantum augmented dual attack. Cryptology ePrint Archive, Paper 2022/656 (2022). https://doi.org/10.48550/ARXIV.2205.13983, https://eprint.iacr.org/2022/656
4. Bollauf, M.F., Pouly, A., Shen, Y.: Solving SIS in any norm via gaussian sampling. Cryptology ePrint Archive, Paper 2026/225 (2026), https://eprint.iacr.org/2026/225
5. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC '13). pp. 575–584. ACM (2013). https://doi.org/10.1145/2488608.2488680
6. Brakerski, Z., Vaikuntanathan, V.: Fully homomorphic encryption from Ring-LWE and security for key dependent messages. In: Advances in Cryptology – CRYPTO 2011. Lecture Notes in Computer Science, vol. 6841, pp. 505–524. Springer (2011). https://doi.org/10.1007/978-3-642-22792-9_29, https://www.iacr.org/archive/crypto2011/68410501/68410501.pdf
7. Brassard, G., Høyer, P., Mosca, M., Tapp, A.: Quantum amplitude amplification and estimation. Contemporary Mathematics **305**, 53–74 (2002). https://doi.org/10.1090/conm/305/05215
8. Chevignard, C., Schrottenloher, A., Shen, Y.: Quantum algorithm for Discrete Gaussian Sampling. Cryptology ePrint Archive, Paper 2026/984 (2026), https://eprint.iacr.org/2026/984
9. Ducas, L., Pulles, L.N.: Does the dual-sieve attack on learning with errors even work? In: Advances in Cryptology – CRYPTO 2023. Lecture Notes in Computer Science, vol. 14083, pp. 37–69. Springer (2023). https://doi.org/10.1007/978-3-031-38548-3_2
10. Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: Falcon: Fast-Fourier lattice-based compact signatures over NTRU. Specification v1.2, National Institute of Standards and Technology (Jan 2020), https://falcon-sign.info/falcon.pdf, submission to the NIST Post-Quantum Cryptography Standardization Project
11. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC). pp. 197–206. ACM (2008). https://doi.org/10.1145/1374376.1374407
12. Guo, Q., Johansson, T.: Faster dual lattice attacks for solving LWE with applications to CRYSTALS. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology – ASIACRYPT 2021, Part IV. Lecture Notes in Computer Science, vol. 13093, pp. 33–62. Springer (2021). https://doi.org/10.1007/978-3-030-92068-5_2
13. Klein, P.N.: Finding the closest lattice vector when it's unusually close. In: Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). pp. 937–941. Society for Industrial and Applied Mathematics (2000)
14. Magniez, F., Nayak, A., Roland, J., Santha, M.: Search via quantum walk. SIAM Journal on Computing **40**(1), 142–164 (2011). https://doi.org/10.1137/090745854

15. MATZOV: Report on the security of LWE: Improved dual lattice attack. Tech. rep., Zenodo (April 2022). https://doi.org/10.5281/zenodo.6493704, https://doi.org/10.5281/zenodo.6493704
16. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on gaussian measures. SIAM Journal on Computing **37**(1), 267–302 (2007). https://doi.org/10.1137/S0097539705447360
17. National Institute of Standards and Technology: Module-Lattice-Based Key-Encapsulation Mechanism Standard. Federal Information Processing Standards Publication 203, National Institute of Standards and Technology (2024). https://doi.org/10.6028/NIST.FIPS.203, https://doi.org/10.6028/NIST.FIPS.203
18. von Neumann, J.: Various techniques used in connection with random digits. In: Monte Carlo Method, National Bureau of Standards Applied Mathematics Series, vol. 12, pp. 36–38. U.S. Government Printing Office, Washington, DC (1951), notes by G. E. Forsythe
19. Ozols, M., Roetteler, M., Roland, J.: Quantum rejection sampling. ACM Transactions on Computation Theory **5**(3), 11:1–11:33 (2013). https://doi.org/10.1145/2493252.2493256
20. Pouly, A., Shen, Y.: Provable dual attacks on learning with errors. In: Advances in Cryptology – EUROCRYPT 2024. Lecture Notes in Computer Science, vol. 14657, pp. 256–285. Springer (2024). https://doi.org/10.1007/978-3-031-58754-2_10
21. Qu, H., Xu, G.: On the provable dual attack for LWE by modulus switching. In: Advances in Cryptology – ASIACRYPT 2025. Lecture Notes in Computer Science, vol. 16247, pp. 34–64. Springer (2026). https://doi.org/10.1007/978-981-95-5099-9_2
22. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM **56**(6), 34:1–34:40 (2009). https://doi.org/10.1145/1568318.1568324
23. Schnorr, C.P.: A hierarchy of polynomial time lattice basis reduction algorithms. Theoretical Computer Science **53**(2–3), 201–224 (1987). https://doi.org/10.1016/0304-3975(87)90064-8
24. Stephens-Davidowitz, N.: On the Gaussian Measure Over Lattices. Ph.D. thesis, New York University (2017), https://cs.nyu.edu/media/publications/stephens-davidowitz_noah.pdf
25. Szegedy, M.: Quantum speed-up of Markov chain based algorithms. In: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science. pp. 32–41. IEEE Computer Society (2004). https://doi.org/10.1109/FOCS.2004.53
26. Temme, K., Osborne, T.J., Vollbrecht, K.G.H., Poulin, D., Verstraete, F.: Quantum Metropolis sampling. Nature **471**(7336), 87–90 (2011). https://doi.org/10.1038/nature09770
27. Wang, Z., Ling, C.: Independent Metropolis–Hastings–Klein algorithm for lattice Gaussian sampling. In: 2015 IEEE International Symposium on Information Theory (ISIT). pp. 2470–2474. IEEE (2015). https://doi.org/10.1109/ISIT.2015.7282900
28. Wang, Z., Ling, C.: On the geometric ergodicity of Metropolis–Hastings algorithms for lattice Gaussian sampling. IEEE Transactions on Information Theory **64**(2), 738–751 (2018). https://doi.org/10.1109/TIT.2017.2742509
29. Wang, Z., Ling, C.: Lattice Gaussian sampling by Markov chain Monte Carlo: Bounded distance decoding and trapdoor sampling. IEEE Transactions on Information Theory **65**(6), 3630–3645 (2019). https://doi.org/10.1109/TIT.2019.2901497

30. Wocjan, P., Abeyesinghe, A.: Speedup via quantum sampling. Physical Review A **78**(4), 042336 (2008). https://doi.org/10.1103/PhysRevA.78.042336
31. Yoder, T.J., Low, G.H., Chuang, I.L.: Fixed-point quantum search with an optimal number of queries. Physical Review Letters **113**(21), 210501 (2014). https://doi.org/10.1103/PhysRevLett.113.210501