Subexponential Algorithms for LWE via Rejection Sampling on Superlattices

Ligeng Fan Changpeng Shao Hongbo Li Dingkang Wang

State Key Laboratory of Mathematical Sciences, AMSS, CAS, Beijing, China

School of Mathematical Sciences, UCAS, Beijing, China

{fanligeng, changpeng.shao}@amss.ac.cn; {hli, dwang}@mmrc.iss.ac.cn

Abstract

Recent superlattice algorithms have improved the provable complexity of lattice problems. Ducas, Engelberts, and Loyer used this framework to obtain a subexponential algorithm for infinity-norm short integer solution, while also identifying a key obstacle to extending it to learning with errors (LWE). A similar difficulty arose in the analysis of Kirchner and Fouque, whose claimed subexponential running time with only linearly many LWE samples was not rigorously established. Building on the framework of Ducas, Engelberts, and Loyer, we develop two refinements based on rejection sampling that solve this open problem.

First, we use rejection sampling to cancel the shift-dependent normalization factor that arises when sampling from a shifted, scaled integer lattice. This factor is a product of one-dimensional Gaussian masses and can be evaluated efficiently. This yields a provably subexponential classical search-LWE algorithm for prime $q \in [n^2, 2n^2]$ and constant Gaussian error width. It uses only $m = 4n$ independent samples and runs in time and space $\exp(O(n/\ln\ln n))$.

Second, we use random auxiliary superlattices and apply rejection sampling to the difference of two Gaussian lattice samples, ensuring that the accepted output follows the desired discrete Gaussian distribution up to a small statistical error. For $q = n^{\kappa+o(1)}$, $m = n+h = O(n)$, and $R = h\ln q/[m\ln(2+\sigma)] \to \infty$, the resulting algorithm uses time and space $2^{(1/2+o(1))n/\ln R}$, where σ is the Gaussian error width. In particular, for $\sigma = n^{o(1)}$ and any fixed $m/n > 1$, our algorithm achieves provable subexponential classical running time, whereas the primal and dual baselines have exponential-time bounds and earlier provable subexponential BKW analyses require $\Omega(n\ln n)$ samples.

Keywords. Learning with errors, lattice algorithms, rejection sampling.

1 Introduction

Learning with errors (LWE) and the short integer solution problem (SIS) are two of the central computational problems in lattice-based cryptography. Informally, LWE asks to recover a hidden vector from noisy linear equations modulo an integer q, whereas SIS asks for a short nonzero modular relation among given public vectors. Their importance is both theoretical and practical: worst-case to average-case reductions connect these problems to fundamental lattice problems [Ajt96, Reg05, MR07], while modern post-quantum cryptographic standards rely extensively on lattice-based constructions [NIST24a]. LWE also forms the basis of fully homomorphic encryption, enabling computation directly on encrypted data [Gen09, BV11, GSW13].

Many cryptographic constructions rely on LWE or its ring and module variants with narrow error distributions, including fully homomorphic encryption [CKKS17, LMK+23] and key encapsulation [NIST24a]. In homomorphic encryption, errors grow as operations are performed on ciphertexts. Smaller initial errors leave more room for this growth and can therefore support better multiplicative depth before bootstrapping is needed [Bra12]. These applications motivate a closer study of the hardness of LWE with small errors, particularly whether substantially faster algorithms become possible in this parameter regime.

Kirchner and Fouque [KF15] developed an improved variant of the Blum-Kalai-Wasserman (BKW) algorithm [BKW03], achieving subexponential time for LWE with polynomial moduli and sufficiently narrow errors. The underlying BKW method eliminates coordinates by finding and combining samples with matching blocks. Although Kirchner and Fouque claimed that a linear number of samples suffices, Herold, Kirshanova, and May [HKM18] subsequently identified a gap in the sample-amplification argument and provided a corrected analysis using $\Theta(n \log n)$ samples. This left open whether subexponential time could be achieved with only a linear number of samples.

Recent progress on lattice algorithms using superlattices suggests another approach to this question. A *superlattice* is a lattice that contains the target lattice and has the same dimension. Working in a denser superlattice can make it easier to obtain short vectors. Subtracts two vectors that belong to the same coset of the target lattice. Their difference then lies in the target lattice. The underlying idea of combining vectors within a common lattice coset already plays a central role in the discrete Gaussian sampling algorithms of Aggarwal et al. [ADRS15, ADS15]. More recent works have extended such techniques in superlattices to solve SVP and Gaussian sampling [GFH26, Kim26, Hhan26]. These motivate our use of superlattices to solve LWE with few samples.

Ducas, Engelberts, and Loyer [DEL25] introduced a particularly useful framework for solving SIS. They successfully translate Wagner's algorithm into a process that iteratively lifts discrete Gaussian lattice vectors along a lattice chain. Their result shows a subexponential algorithm for infinity-norm SIS. They also identify a key obstacle to extending the same framework to LWE: the result distribution must be accurate enough to reveal a small Fourier bias, but the smoothing requirement on their scaled integer auxiliary lattices is then too expensive. They leave the problem of applying their framework to fix the unsolved claim by [KF15].

In this work, we study the above open question and provide affirmative answers to it.

1.1 Our results

We use m for the number of initial LWE equations throughout. Let $\mathbf{b} = A\mathbf{s} + \mathbf{e} \pmod q$, with uniform $A \in \mathbb{F}_q^{m\times n}$, arbitrary secret $\mathbf{s} \in \mathbb{F}_q^n$, and independent errors $e_i \sim D_{\mathbb{Z},\sigma}$, where $\Pr[e_i = t] \propto \exp(-\pi t^2/\sigma^2)$. In search-LWE, the goal is to recover $\mathbf{s}$ with success probability $1 - o(1)$.

Result 1 (Constant-error, theorem 3.7)**.** Let $q \in [n^2, 2n^2]$ be prime, $m = 4n$, and $0 < \sigma = O(1)$. There is a classical randomized search-LWE algorithm with time and space $\exp(O(n/\ln\ln n))$.

Result 2 (Narrow-error LWE with linear samples, theorem 4.7)**.** Fix $\kappa > 0$. Let $q = n^{\kappa+o(1)}$ be prime, let $m = n + h = O(n)$ with $h \geq 1$, and let $\sigma \geq 1$. If

$$R = \frac{h \ln q}{m \ln(2+\sigma)} \longrightarrow \infty, \tag{1.1}$$

then search-LWE can be solved in time and space

$$2^{(1/2+o(1))n/\ln R}.$$

For every fixed $c > 1$, this is subexponential when $m = (c + o(1))n$ and $\sigma = n^{o(1)}$.

The theorem permits $h = o(n)$ provided (1.1) holds. For $q = n^{\kappa+o(1)}$, fixed $\kappa > 0$, and $1 \leq \sigma = n^{o(1)}$, any fixed sample ratio $m/n \to c > 1$ therefore gives subexponential time and space. Constant or polylogarithmic error widths give $2^{(1/2+o(1))n/\ln\ln n}$. Table 1 compares explicit classical baselines.

Table 1: Classical algorithms for subpolynomial errors

LWE algorithm	Original samples	Time and space
Primal baseline: exact CVP [ADS15]	$(1 + o(1))n$	$2^{(1+o(1))n}$
Gaussian dual baseline [Kim26] and lemma 2.14	$(1 + o(1))n$	$2^{(1/2+o(1))n}$
Earlier BKW amplification [KF15, HKM18, DEL25]	$\Theta(n \ln n)$	$2^{o(n)}$ for $\kappa > 1/2$
This work, Result 2	cn, any fixed $c > 1$	$2^{o(n)}$

The earlier BKW analyses in the table use more original samples; see [DEL25, Section 4.3] for a discussion on this amplification issue. Practical primal and dual costs also depend on the chosen reduction or sampling algorithm and success model [PT26, DP26, PS24, QX25, WWZZ26].

For polynomial error widths, theorem 4.9 gives an exponential tradeoff. Regev's original parameters have $q \asymp n^2$ and $\alpha \asymp 1/(\sqrt{n}(\ln n)^2)$, hence $\sigma \asymp n^{3/2}/(\ln n)^2$ [Reg05, Section 5]. At this scale, corollary 4.10 gives $T, S \leq 2^{(1/[2\ln(4/3)]+o(1))n}$ from $5n$ original samples, with exponent approximately 1.73803. Table 2 records the comparison.

Table 2: Classical algorithms in Regev's parameter

LWE algorithm	Original samples	Time and space	Status
Coded-BKW with sieving [M19]	Polynomial supply	$2^{(1.6399+o(1))n}$	Heuristic
Same algorithm [M19]	Exponential supply	$2^{(0.8917+o(1))n}$	Heuristic
Primal exact-CVP baseline [ADS15]	$(4+o(1))n$	$2^{(4+o(1))n}$	Proved
Gaussian dual baseline [Kim26]	$(4+o(1))n$	$2^{(2+o(1))n}$	Proved
This work, corollary 4.10	$5n$	$2^{(1.7381+o(1))n}$	Proved

The complexity result of polynomial-sample BKW algorithm is smaller, but it is heuristic and does not establish the same bound from $5n$ samples. Further BKW time–memory tradeoffs are studied in [WBLW25].

1.2 Gaussian sampling through superlattices

Our algorithms follow the Gaussian-sampling framework of [DEL25]. We first explain its basic idea, before stating our technical contributions. More details are in section 2.6. Given the sample matrix A, let $\Lambda_q^\perp(A) := \{\mathbf{x} \in \mathbb{Z}^m : A^T\mathbf{x} = 0 \pmod q\}$. For a lattice point from $\Lambda_q^\perp(A)$, we have $\langle \mathbf{x}, \mathbf{b}\rangle = \langle \mathbf{x}, \mathbf{e}\rangle \pmod q$: the secret disappears from the linear combination. Gaussian samples of a suitable width on this lattice can therefore provide information about the error. The remaining task is to construct these samples efficiently.

To do so, [DEL25] builds a chain of projected lattices, starting with $\Lambda_0 = \mathbb{Z}^{m-n}$ and ending with $\Lambda_r = \Lambda_q^\perp(A)$. Sampling Gaussian on Λ_0 is simple. The algorithm works in r stages. At each stage, the Gaussian vectors from the preceding lattice Λ_{i-1} are lifted to the next lattice Λ_i along the lattice chains while preserving the Gaussian distribution. Finally, the left vectors fall in Λ_r with the desired Gaussian width. Along the lattice chain, the length of lattice points is increased by a corresponding block size; we may also understand the lifting stage as a process of adding new coordinates.

A direct lift at each stage can be too expensive in Gaussian width. The usual Gaussian lifting argument requires a width above the smoothing parameter of $\ker(\Lambda_i \twoheadrightarrow \Lambda_{i-1})$ [EWY23, Algorithm 2 and Theorem 1]. In the case of [DEL25], this width is too large for the intended LWE application.

This is where a superlattice helps. Instead of lifting directly to Λ_i, [DEL25] introduces a denser lattice $\widetilde{\Lambda}_i \supseteq \Lambda_i$ which permits lifting at a smaller width. That is, from a narrow Gaussian vector in Λ_{i-1}, we can easily lift it to $\widetilde{\Lambda}_i$ and preserve the same small Gaussian width. The superlattices picked in [DEL25] are highly motivated by Wagner's algorithm.

Now the lifted vectors belong to $\widetilde{\Lambda}_i$, and we still need them return to Λ_i. Notice that two vectors in the same $\widetilde{\Lambda}_i/\Lambda_i$ coset have a difference in Λ_i. Group them according to their cosets modulo Λ_i and subtract pairs within each group. Under a suitable smoothing condition, we can show the difference is approximately Gaussian in Λ_i, with width increased by a factor $\sqrt{2}$. Thus at each stage, one first lifts to the superlattice without changing the width and then takes differences to reach the target lattice. Only a constant fraction of the list is lost.

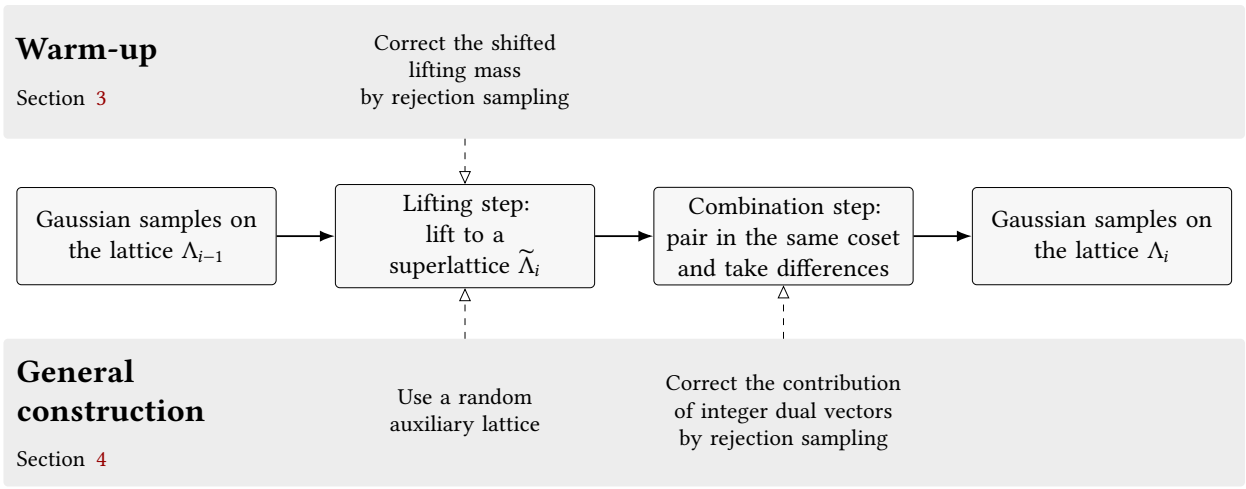

Figure 1: Comparison of the two constructions at stage i.

The choices of superlattices and the block sizes determine both the number of cosets (i.e. $[\widetilde{\Lambda}_i : \Lambda_i]$) and the Gaussian width. A smaller width makes the eventual LWE signal stronger, but it also requires a stronger smoothing requirement, which can force a denser superlattice and a larger list. The obstacle identified in [DEL25, Section 4.3] is that the accuracy needed for LWE makes the lifting requirement too costly when $m = O(n)$. Our modifications correct the relevant Gaussian factors by rejection sampling. This permits weaker smoothing requirements while retaining the output accuracy required for recovery.

1.3 Technical overview

We will briefly explain the main ideas behind these two constructions; figure 1 provides a general overview and a comparison of two methods.

Warm-up construction. Our first construction changes only the lifting step and keeps the ordinary Gaussian combination step. Its value is that the correction can be seen directly: it removes the shifted Gaussian normalization by rejecting samples.

Consider one stage of the construction in [DEL25]. Starting with Gaussian vectors on Λ_{i-1}, we add b coordinates and eventually obtain Gaussian vectors on Λ_i. The intermediate step uses a superlattice $\widetilde{\Lambda}_i \supseteq \Lambda_i$. Let $S := \ker(\widetilde{\Lambda}_i \twoheadrightarrow \Lambda_{i-1})$ which is a lattice. The lifting procedure of [EWY23], as used in [DEL25], samples $\mathbf{u} \sim D_{\Lambda_{i-1},s}$ and then $\mathbf{y} \sim D_{S+\mathbf{c}(\mathbf{u}),s}$, where $\mathbf{c}(\mathbf{u})$ is determined by $\mathbf{u}$. Its output probability is

$$\mathbb{P}[(\mathbf{u},\mathbf{y})] = \frac{\rho_s((\mathbf{u},\mathbf{y}))}{\rho_s(\Lambda_{i-1})\,\rho_s(S+\mathbf{c}(\mathbf{u}))}.$$

The numerator is the desired Gaussian weight, but the factor $\rho_s(S+\mathbf{c}(\mathbf{u}))$ in the denominator depends on $\mathbf{u}$. The analysis in [DEL25] requires $s \geq \eta_\varepsilon(S)$, making

$$\rho_s(S+\mathbf{c}) = (1 \pm \varepsilon)\frac{s^b}{\det S}$$

for any $\mathbf{c}$. Then the output is approximately Gaussian.

Our observation is that their auxiliary lattice is $S = a\mathbb{Z}^b$, where $a = q/p$. Its shifted mass factors as

$$\rho_s(a\mathbb{Z}^b + \mathbf{c}) = \prod_{j=1}^{b} \left(\sum_{k\in\mathbb{Z}} e^{-\pi(ak+c_j)^2/s^2} \right),$$

which can be evaluated efficiently to any required precision. We therefore accept the proposed pair with probability

$$a(\mathbf{u}) = \frac{\rho_s(S + \mathbf{c}(\mathbf{u}))}{\rho_s(S)} \leq 1.$$

Multiplication cancels the unwanted denominator exactly. Conditioned on acceptance, the output has the desired distribution $D_{\widetilde{\Lambda}_i,s}$.

Smoothing is now needed only to ensure efficient acceptance: $s \geq \eta_{1/10}(S)$ gives acceptance probability at least $9/11$ (lemma 3.1). For the scaled integer lattice, this replaces the factor $\sqrt{\ln(2b/\varepsilon)}$ in the required width by $\sqrt{\ln(2b)}$. At the same width s, we can consequently choose a larger block size b. Larger blocks reduce the number of stages r. Since each stage increases the width by $\sqrt{2}$, the final width $s_r = 2^{r/2}s_0$ becomes smaller. This gives shorter lattice vectors and preserves more information about the LWE errors. Section 3 verifies the remaining sampling and smoothing requirements and obtains subexponential LWE recovery with $m = 4n$ and constant Gaussian error width.

General construction. For the stronger result, we improve both the lifting and combination step. First, we replace the scaled integer auxiliary lattice by a random lattice. At equal dimension b and determinant a^b, the comparison of the smoothing parameter is

$$\eta_\varepsilon(a\mathbb{Z}^b) = \Theta\left(a\sqrt{\ln(2b/\varepsilon)} \right),$$
$$\eta_\varepsilon(\text{random lattice } S) \leq C_2 a \sqrt{1 + \frac{\ln(1/\varepsilon)}{b}},$$

where the second bound holds with high probability under the conditions of lemma 4.1. When $\ln(1/\varepsilon) \gg b$, this saves a factor of order $\sqrt{b}$ in the width required for high-accuracy lifting.

Sampling on its shifts no longer separates by coordinates. We use the sampler of [ADS15, Theorem 4.7], whose cost is $2^{b+O(\sqrt{b}\log b)}$ up to polynomial factors. Our block sizes keep this cost relatively low.

The combination step presents another restriction. For independent $\mathbf{u}, \mathbf{v} \sim D_{\Lambda+\mathbf{c},s}$, where $\Lambda = \Lambda_i \subseteq \mathbb{Z}^d$, their difference satisfies, for every $\mathbf{w} \in \Lambda$,

$$\mathbb{P}[\mathbf{u} - \mathbf{v} = \mathbf{w}] = \frac{\rho_{\sqrt{2}s}(\mathbf{w})\, \rho_{s/\sqrt{2}}(\Lambda + \mathbf{c} - \mathbf{w}/2)}{\rho_s(\Lambda + \mathbf{c})^2}.$$

The extra shifted mass $\rho_{s/\sqrt{2}}(\Lambda + \mathbf{c} - \mathbf{w}/2)$ again distorts the desired Gaussian distribution. Luckily, we can remove this part by rejection sampling. Put $v = s/\sqrt{2}$. Because $\mathbb{Z}^d \subseteq \Lambda^*$, Poisson

summation separates this mass into integer and noninteger dual contributions:

$$\rho_v(\Lambda + \mathbf{t}) = \frac{v^d}{\det \Lambda}\big(T_v(\mathbf{t}) + E_v(\mathbf{t})\big),$$

where

$$T_v(\mathbf{t}) = v^{-d}\rho_v(\mathbb{Z}^d + \mathbf{t}) = \prod_{j=1}^{d}\left(\frac{1}{v}\sum_{k\in\mathbb{Z}} e^{-\pi(k+t_j)^2/v^2}\right), \quad E_v(\mathbf{t}) = \sum_{\mathbf{z}\in\Lambda^*\setminus\mathbb{Z}^d} e^{-\pi v^2\|\mathbf{z}\|^2} e^{2\pi i\langle \mathbf{z},\mathbf{t}\rangle}.$$

The integer contribution $T_v(\mathbf{t})$ is efficiently computable. For the remainder,

$$|E_v(\mathbf{t})| \le \rho_{1/v}(\Lambda^* \setminus \mathbb{Z}^d),$$

which is small for the random q-ary target lattices by lemma 2.11.

Next, writing $\mathbf{w} = \mathbf{u} - \mathbf{v}$, we accept the pair with probability

$$a(\mathbf{u},\mathbf{v}) = \frac{1}{2T_v((\mathbf{u}+\mathbf{v})/2)} = \frac{1}{2T_v(\mathbf{c} - \mathbf{w}/2)}.$$

We prove that the accepted $\mathbf{w}$ has probability approximately proportional to $\rho_{\sqrt{2}s}(\mathbf{w})$. Therefore, the accepted distribution is correspondingly close to $D_{\Lambda,\sqrt{2}s}$ (lemma 4.4).

In conclusion, random auxiliary lattices help improve the lifting step, while the rejection trick removes the need for high-accuracy smoothing during the combination step. Together, they yield a general LWE algorithm in section 4.

Comparison with recent results. Recent preprints also obtain subexponential algorithms with few LWE samples. Han et al. [HGH26] use random auxiliary lattices and martingale analysis for decision-LWE with narrow Gaussian errors and $n + o(n)$ samples. Chen et al. [CJWZ26] use exact shifted Gaussian sampling for decision binary LWE with sublinear samples. Lin and Vasudevan [LV26] analyze Fourier moments for bounded-error LWE. Our constructions use rejection sampling as the main contribution point.

Organization. The remainder is organized as follows. Section 2 introduces the notation, background results, and lattice-chain construction of [DEL25]. Sections 3 and 4 prove the two constructions and their parameter bounds; section 4.7 gives the extension to polynomial errors, including Regev's parameters.

2 Preliminaries

Notation. We write $\mathbb{Z}$, $\mathbb{R}$, and $\mathbb{F}_q$ for the integers, the reals, and the field with q elements, respectively; q is prime throughout the paper. For $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $\langle \mathbf{x}, \mathbf{y}\rangle, \|\mathbf{x}\| = \|\mathbf{x}\|_2, \|\mathbf{x}\|_\infty$ denote the inner product, Euclidean and infinity norm. The transpose of a matrix A is A^T, and I_d is the d-dimensional identity matrix.

For a set S, its *shift by* $\mathbf{c}$ is $S + \mathbf{c} := \{\mathbf{x} + \mathbf{c} : \mathbf{x} \in S\}$, and $\mathrm{dist}(\mathbf{c}, S) := \inf_{\mathbf{x}\in S}\|\mathbf{c} - \mathbf{x}\|$. For $U, V \subseteq \mathbb{R}^n$, $U - V := \{\mathbf{u} - \mathbf{v} : \mathbf{u} \in U, \mathbf{v} \in V\}$.

All asymptotic statements are as $n \to \infty$ unless specified. We use the usual O, Ω, Θ, and o notation; An event holds *with high probability* if its probability is $1 - o(1)$. bits$(\cdot)$ denotes the binary encoding length of rational input data.

2.1 Lattices, SIS, and LWE

A *full-rank lattice* in $\mathbb{R}^d$ with *basis* B is the set $\Lambda = \mathcal{L}(B) = \{B\mathbf{x} : \mathbf{x} \in \mathbb{Z}^d\}$, where $B \in \mathbb{R}^{d\times d}$ is invertible. The *dimension* of Λ is the dimension of the linear space spanned by any basis of Λ. The *determinant* $\det\Lambda := |\det B|$ is the volume of the fundamental parallelepiped. The *dual lattice* of Λ is

$$\Lambda^* := \{\mathbf{y} \in \mathbb{R}^d : \langle \mathbf{x}, \mathbf{y}\rangle \in \mathbb{Z} \text{ for every } \mathbf{x} \in \Lambda\} = B^{-T}\mathbb{Z}^d.$$

A *superlattice* Λ' of Λ is a lattice that contains Λ and has the same dimension. A *coset* of Λ in Λ' is a set $\Lambda + \mathbf{c}$ with $\mathbf{c} \in \Lambda'$. Two points lie in the same coset exactly when their difference lies in Λ. The *index* $[\Lambda' : \Lambda] = \det\Lambda/\det\Lambda'$ is the number of these cosets.

The following two problems underlie lattice cryptography [Ajt96, Reg05, MR07].

Definition 2.1 (SIS)**.** Given $A \in \mathbb{F}_q^{m\times n}$ and a bound $\beta > 0$, the *infinity-norm short integer solution problem*, denoted SIS$^\infty$, asks for a vector $\mathbf{x} \in \mathbb{Z}^m$ such that

$$\mathbf{x} \neq 0, \qquad A^T\mathbf{x} = 0 \pmod q, \qquad \|\mathbf{x}\|_\infty \le \beta.$$

An SIS instance is random when A is uniform. Other norm versions replace the last inequality by a bound in the specified norm.

Definition 2.2 (LWE)**.** For $\sigma > 0$ and a secret $\mathbf{s} \in \mathbb{F}_q^n$. An instance with m independent samples consists of

$$A \in \mathbb{F}_q^{m\times n}, \qquad \mathbf{b} = A\mathbf{s} + \mathbf{e} \pmod q,$$

where A is uniform and each coordinate of $\mathbf{e}$ are independent discrete Gaussian samples from $D_{\mathbb{Z},\sigma}$, that is, $\mathbb{P}[e_i = t] \propto e^{-\pi t^2/\sigma^2}$ for $t \in \mathbb{Z}$. In *search-LWE*, the task is to recover $\mathbf{s}$. In *decision-LWE*, the task is to distinguish these pairs $(A, \mathbf{b})$, with a uniform secret, from uniform pairs in $\mathbb{F}_q^{m\times n} \times \mathbb{F}_q^m$.

Two lattices associated with A connect these problems:

$$\Lambda_q^\perp(A) := \{\mathbf{x} \in \mathbb{Z}^m : A^T\mathbf{x} = 0 \pmod q\}, \qquad \Lambda_q(A) := A\mathbb{Z}^n + q\mathbb{Z}^m.$$

An integer representative of A is used in $\Lambda_q(A)$ and the lattice does not depend on the choice. Both are *q-ary lattices*: they lie between $q\mathbb{Z}^m$ and $\mathbb{Z}^m$. If A has rank n, counting residue classes gives $\det\Lambda_q^\perp(A) = q^n$ and $\det\Lambda_q(A) = q^{m-n}$. Moreover, $\Lambda_q^\perp(A)^* = \Lambda_q(A)/q$. Indeed, each vector of $\Lambda_q(A)/q$ has integer inner product with $\Lambda_q^\perp(A)$, and the determinants of these two lattices are both q^{-n}.

For $\mathbf{x} \in \Lambda_q^\perp(A)$, the noisy equation gives

$$\langle \mathbf{x}, \mathbf{b}\rangle = \langle \mathbf{x}, A\mathbf{s} + \mathbf{e}\rangle = \langle \mathbf{x}, \mathbf{e}\rangle \pmod q.$$

Thus $\mathbf{x}$ removes the secret. A short $\mathbf{x}$ keeps the resulting error small enough to be informative. This is why sampling Gaussian vectors on $\Lambda_q^\perp(A)$ can help solve LWE. The quantitative recovery statement appears in lemma 2.14.

Gaussian errors also give two useful size bounds. For independent $e_i \sim D_{\mathbb{Z},\sigma}$ and $m = O(n)$ with $m \ge n$, there exist absolute constants C_0, C_1 such that, with probability $1 - o(1)$,

$$\|\mathbf{e}\|^2 \le C_0 m\sigma^2, \qquad \|\mathbf{e}\|_\infty \le C_1\sigma\sqrt{\ln n}. \tag{2.1}$$

We prove explicit tail bounds in lemma A.1.

2.2 Distances between distributions

We distinguish two assertions: one sample is approximately Gaussian, and a list is approximately a list of independent Gaussians. The definitions below specify the difference; a more detailed treatment is given in [DEL25, Section 2].

For distributions P, Q on a countable set, their *total variation distance*, also called *statistical distance*, is

$$\Delta_{\mathrm{TV}}(P, Q) = \frac{1}{2} \sum_{x} |P(x) - Q(x)|.$$

We call P, Q *ε-close* if $\Delta_{\mathrm{TV}}(P, Q) \le \varepsilon$; for lists, this refers to their joint distributions.

Definition 2.3 (δ-similarity, [DEL25, Definition 2.1])**.** Let $\delta \ge 0$. Distributions P, Q are *δ-similar* if

$$e^{-\delta} Q(x) \le P(x) \le e^{\delta} Q(x) \quad \text{for every } x.$$

We abbreviate these two inequalities as $P(x) = e^{\pm\delta} Q(x)$. A random vector is δ-similar to Q when its distribution is.

Following [DEL25, Definition 2.2], a list of random vectors $\mathbf{x}_1, \dots, \mathbf{x}_M$ is *conditionally δ-similar* to independent samples from Q if, for every i, fixing all the other entries leaves a conditional distribution of $\mathbf{x}_i$ that is δ-similar to Q. We write $P_1 \otimes \cdots \otimes P_M$ for the joint distribution of independent entries with distributions $P_1, \dots, P_M$.

Lemma 2.4 (Statistical distance under computation and products)**.** *For a randomized procedure $\mathcal{A}$ and discrete distributions,*

$$\Delta_{\mathrm{TV}}(\mathcal{A}(P), \mathcal{A}(Q)) \le \Delta_{\mathrm{TV}}(P, Q), \qquad \Delta_{\mathrm{TV}}\left(\bigotimes_{i=1}^{M} P_i, \bigotimes_{i=1}^{M} Q_i \right) \le \sum_{i=1}^{M} \Delta_{\mathrm{TV}}(P_i, Q_i).$$

The first one is the standard contraction property [GMPW20, Section 2]; the second follows by replacing one factor at a time and applying the triangle inequality.

2.3 Gaussian identities and sampling

Definition 2.5 (Gaussian mass and discrete Gaussian)**.** For $s > 0$ and $\mathbf{x} \in \mathbb{R}^d$, define

$$\rho_s(\mathbf{x}) = \exp(-\pi \|\mathbf{x}\|^2 / s^2), \qquad \rho_s(S) = \sum_{\mathbf{x} \in S} \rho_s(\mathbf{x})$$

for a countable set S for which the sum is finite. The quantity $\rho_s(S)$ is its *Gaussian mass*, namely the normalizing constant for these weights. For a shifted lattice $\Lambda + \mathbf{c}$, the *discrete Gaussian* $D_{\Lambda+\mathbf{c},s}$ is

$$D_{\Lambda+\mathbf{c},s}(\mathbf{x}) = \frac{\rho_s(\mathbf{x})}{\rho_s(\Lambda + \mathbf{c})} \quad (\mathbf{x} \in \Lambda + \mathbf{c}).$$

The following identities explain the corrections used in this paper. The first rewrites a shifted Gaussian mass as a sum over the dual lattice. The second identifies exactly why a difference of two discrete Gaussians need not be a discrete Gaussian.

Lemma 2.6 (Shifted Gaussian mass and Gaussian differences)**.** *For a full-rank lattice $\Lambda \subset \mathbb{R}^d$ and $\mathbf{c} \in \mathbb{R}^d$, Poisson summation gives*

$$\rho_s(\Lambda + \mathbf{c}) = \frac{s^d}{\det \Lambda} \sum_{\mathbf{y} \in \Lambda^*} e^{-\pi s^2 \|\mathbf{y}\|^2} e^{2\pi \mathrm{i} \langle \mathbf{y}, \mathbf{c} \rangle}. \tag{2.2}$$

The series converges absolutely, and $\rho_s(\Lambda + \mathbf{c}) \le \rho_s(\Lambda)$. If $\mathbf{u}, \mathbf{v}$ are independent with distribution $D_{\Lambda+\mathbf{c},s}$, then, for every $\mathbf{w} \in \Lambda$,

$$\mathbb{P}[\mathbf{u} - \mathbf{v} = \mathbf{w}] = \frac{\rho_{\sqrt{2}s}(\mathbf{w})\, \rho_{s/\sqrt{2}}(\Lambda + \mathbf{c} - \mathbf{w}/2)}{\rho_s(\Lambda + \mathbf{c})^2}. \tag{2.3}$$

Proof. Equation (2.2) is the standard Poisson summation formula; see, for example, [MR07, ADRS15]. Each phase has absolute value one, so the triangle inequality gives the shifted-mass upper bound.

For the second identity, write $\mathbf{v} = \mathbf{u} - \mathbf{w}$ and use

$$\|\mathbf{u}\|^2 + \|\mathbf{u} - \mathbf{w}\|^2 = \tfrac{1}{2}\|\mathbf{w}\|^2 + 2\|\mathbf{u} - \mathbf{w}/2\|^2.$$

Summing the resulting product of weights over $\mathbf{u} \in \Lambda + \mathbf{c}$ gives (2.3). □

In (2.3), the first factor has the desired Gaussian width $\sqrt{2}s$. The second factor is a shifted mass that can vary with $\mathbf{w}$. In [DEL25], s is chosen above the smoothing parameter so that this factor is nearly equal for every $\mathbf{w}$. However, in sections 3 and 4, our new rejection step instead repairs this difference.

The lifting step needs samples on shifted copies of a scaled integer lattice. This case is simple because the coordinates can be sampled independently. We cite the following sampling result.

Lemma 2.7 (Sampling on shifted integer lattices, [BLPRS13, Lemma 2.3])**.** *Let $a > 0$, $\mathbf{c} \in \mathbb{R}^b$, and $s \ge a$. There is an exact sampler for $D_{a\mathbb{Z}^b+\mathbf{c},s}$ in expected polynomial time. For the rational inputs and polynomial parameter ranges used in this paper, an implementation with total variation error at most δ takes time polynomial in the input size and $\ln(1/\delta)$.*

Proof. In dimension one, the width condition of [BLPRS13, Lemma 2.3] is $s \ge a\sqrt{\ln 6/\pi}$, which follows from $s \ge a$. The Gaussian weight and its normalizing constant on $a\mathbb{Z}^b + \mathbf{c}$ both factor across coordinates. Hence sampling the coordinates independently and putting them together gives exactly the desired joint distribution. For a finite implementation in our parameter ranges, truncate and evaluate each one-dimensional Gaussian with error at most δ/b; lemma 2.4 then bounds the total error by δ. The required truncation and precision estimates are given in section A.5. □

2.4 Smoothing bounds

The dependence of $\rho_s(\Lambda + \mathbf{c})$ on $\mathbf{c}$ is the main source of distributional error. The smoothing parameter gives a convenient sufficient condition for all these masses to be nearly equal.

Definition 2.8 (Smoothing parameter, [MR07])**.** For $0 < \varepsilon < 1$, the *smoothing parameter* $\eta_\varepsilon(\Lambda)$ is the least $s > 0$ such that

$$\rho_{1/s}(\Lambda^* \setminus \{0\}) = \sum_{\mathbf{y} \in \Lambda^* \setminus \{0\}} e^{-\pi s^2 \|\mathbf{y}\|^2} \le \varepsilon.$$

The zero dual vector contributes exactly $s^d / \det \Lambda$ in (2.2). When $s \ge \eta_\epsilon(\Lambda)$, all nonzero dual vectors in (2.2) contributes at most $\varepsilon s^d / \det \Lambda$. Smaller ε asks for greater accuracy and generally requires a larger width.

Lemma 2.9 (Masses above smoothing)**.** *If $s \ge \eta_\varepsilon(\Lambda)$, then for every shift $\mathbf{c}$,*

$$(1 - \varepsilon)\frac{s^d}{\det \Lambda} \le \rho_s(\Lambda + \mathbf{c}) \le (1 + \varepsilon)\frac{s^d}{\det \Lambda}.$$

Also, $\eta_\varepsilon(a\Lambda) = a\eta_\varepsilon(\Lambda)$ for $a > 0$.

Proof. Subtract the zero term in (2.2) and bound all other terms in absolute value using definition 2.8. The scaling identity follows from $(a\Lambda)^* = a^{-1}\Lambda^*$. □

The integer lattice is especially simple because its Gaussian sum is a product of one-dimensional Gaussian sums. The next bound will help to set the Gaussian width of the auxiliary lattice in section 3.

Lemma 2.10 (Smoothing the integer lattice)**.** *For every integer $d \ge 1$ and $t > 0$,*

$$\rho_{1/t}(\mathbb{Z}^d) - 1 \le \exp\left(\frac{2de^{-\pi t^2}}{1 - e^{-\pi t^2}}\right) - 1. \tag{2.4}$$

In particular, for $0 < \varepsilon < 1$,

$$\eta_\varepsilon(\mathbb{Z}^d) \le \sqrt{\frac{\ln(1 + 4d/\varepsilon)}{\pi}} = O\left(\sqrt{\ln(2d/\varepsilon)}\right).$$

The bound for $a\mathbb{Z}^d$ is obtained by multiplying by a.

Proof. Put $r = e^{-\pi t^2}$. Since $k^2 \ge k$ for $k \ge 1$,

$$\rho_{1/t}(\mathbb{Z}) = 1 + 2\sum_{k \ge 1} r^{k^2} \le 1 + \frac{2r}{1 - r}.$$

Taking the dth power and using $1 + x \le e^x$ gives (2.4). At $t^2 = \ln(1 + 4d/\varepsilon)/\pi$, its exponent equals $\varepsilon/2$. The inequality $e^{\varepsilon/2} - 1 \le \varepsilon$ for $0 \le \varepsilon \le 1$ proves the claimed smoothing bound. □

This is a standard integer-lattice smoothing estimate; related bounds are used in [DEL25, Lemma 2.8].

2.5 Random q-ary lattices

For a random q-ary lattice, we can separate the same integer contribution from an additional random contribution. This separation will serve two purposes. In section 3 both parts must be small. In section 4 we will compute and correct the integer part, so only the rest dual part needs to be extremely small.

Lemma 2.11 (The extra dual mass of a random q-ary lattice). *Let $B \in \mathbb{F}_q^{n\times h}$ be uniform, with $h \geq 1$, and define the d-dimensional lattice*

$$\Lambda = \{(\mathbf{x}, \mathbf{z}) \in \mathbb{Z}^h \times \mathbb{Z}^n \,:\, \mathbf{z} + B\mathbf{x} = 0 \pmod q\}, \qquad d = h + n.$$

Its dual contains $\mathbb{Z}^d$. For every $v > 0$, the expected Gaussian mass of the dual vectors outside $\mathbb{Z}^d$ satisfies

$$\mathbb{E}_B[\rho_{1/v}(\Lambda^* \setminus \mathbb{Z}^d)] \leq q^{-h}(1 + q/v)^d. \tag{2.5}$$

Consequently, for every $0 < \tau < 1$, with probability at least $1 - \tau$ over B,

$$\rho_{1/v}(\Lambda^* \setminus \{0\}) \leq \big(\rho_{1/v}(\mathbb{Z}^d) - 1\big) + \frac{q^{-h}(1 + q/v)^d}{\tau}.$$

If this last upper bound is at most ε, then $\eta_\varepsilon(\Lambda) \leq v$.

Proof. Since $\Lambda = \Lambda_q^\perp([B \mid I_n]^T)$, we first compute its dual:

$$\Lambda^* = \frac{1}{q}\Lambda_q([B \mid I_n]^T) = \{(B^T\mathbf{u}/q, \mathbf{u}/q) \,:\, \mathbf{u} \in \mathbb{Z}^n\} + \mathbb{Z}^d.$$

Reducing the dual vectors modulo $\mathbb{Z}^d$ gives the q^n distinct lattice cosets

$$\Lambda^*/\mathbb{Z}^d = \{(B^T\mathbf{u}/q, \mathbf{u}/q) + \mathbb{Z}^d \,:\, \mathbf{u} \in \mathbb{F}_q^n\}.$$

Fix $\mathbf{u} \neq 0$. Each coordinate of $B^T\mathbf{u}$ is uniform in $\mathbb{F}_q$, and these coordinates are independent. Averaging over their q^h possible values therefore gives

$$\begin{aligned}\mathbb{E}_B[\rho_{1/v}(\mathbb{Z}^h + B^T\mathbf{u}/q)] &= q^{-h}\sum_{\mathbf{a}\in\mathbb{F}_q^h} \rho_{1/v}(\mathbb{Z}^h + \mathbf{a}/q) \\ &= q^{-h}\rho_{1/v}(q^{-1}\mathbb{Z}^h) = q^{-h}\rho_{q/v}(\mathbb{Z}^h).\end{aligned}$$

Multiply by the remaining factor $\rho_{1/v}(\mathbb{Z}^n + \mathbf{u}/q)$ and sum over $\mathbf{u} \neq 0$. Including $\mathbf{u} = 0$ only increases this upper bound, so

$$\begin{aligned}\mathbb{E}_B[\rho_{1/v}(\Lambda^* \setminus \mathbb{Z}^d)] &= \mathbb{E}_B\Big[\sum_{\mathbf{u}\in\mathbb{F}_q^n\setminus\{0\}} \rho_{1/v}\Big(\frac{B^T\mathbf{u}}{q} + \mathbb{Z}^h\Big)\rho_{1/v}\Big(\frac{\mathbf{u}}{q} + \mathbb{Z}^n\Big)\Big] \\ &= q^{-h}\rho_{q/v}(\mathbb{Z}^h)\sum_{\mathbf{u}\in\mathbb{F}_q^n\setminus\{0\}} \rho_{1/v}(\mathbb{Z}^n + \mathbf{u}/q) \\ &\leq q^{-h}\rho_{1/v}(\frac{1}{q}\mathbb{Z}^n)\rho_{q/v}(\mathbb{Z}^h) = q^{-h}\rho_{q/v}(\mathbb{Z}^d).\end{aligned}$$

For every $t > 0$, comparison with the integral gives

$$\rho_t(\mathbb{Z}) = 1 + 2\sum_{k\geq 1} e^{-\pi k^2/t^2} \leq 1 + 2\int_0^\infty e^{-\pi x^2/t^2}\, dx = 1 + t.$$

This proves (2.5). Finally, using Markov's inequality and adding $\rho_{1/v}(\mathbb{Z}^d) - 1$ gives the displayed high probability bound. This proves the smoothing consequence as well. □

2.6 The lattice chain, lifting, and pairing

We now give the construction of superlattices described in the introduction. Our algorithms follow the framework of [DEL25]; the changes to its two sampling steps are presented in sections 3 and 4.

Lattice chain Write $h = m - n$ and $A^T = [A_1 \mid A_2]$, where A_2 has n columns. Abort if A_2 is singular; for uniform A, this event has probability at most $1/(q-1)$. Multiplying $A^T\mathbf{u} = 0$ by A_2^{-1} puts the constraint matrix in the form $[B \mid I_n]$, where $B = A_2^{-1}A_1 \in \mathbb{F}_q^{n\times h}$ remains uniform. Only the equations change: the vector $\mathbf{u}$ and its Euclidean norm stay the same.

Choose positive block sizes $b_1, \ldots, b_r$ with $\sum_{i=1}^r b_i = n$, and put $n_0 = 0$ and $n_i = \sum_{k=1}^i b_k$. Let $B^{(i)}$ be the first n_i rows of B. For $i = 0, \ldots, r$, define

$$\Lambda_i = \Lambda_q^\perp([B^{(i)} \mid I_{n_i}]^T) = \{(\mathbf{x}, \mathbf{z}) \in \mathbb{Z}^h \times \mathbb{Z}^{n_i} \,:\, \mathbf{z} + B^{(i)}\mathbf{x} = 0 \pmod q\}.$$

In particular, $\Lambda_0 = \mathbb{Z}^h$ and $\Lambda_r = \Lambda_q^\perp(A)$. Each Λ_i has the form in lemma 2.11, with n_i equations. The integer i counts stages, whereas n_i counts equations already imposed. Our parameter choices will give $r = O(\log n)$; the block sizes need not all be equal.

For $i = 1, \ldots, r$, define the projection

$$\pi_i \,:\, \Lambda_i \longrightarrow \Lambda_{i-1}, \qquad \pi_i(\mathbf{x}, \mathbf{z}, \mathbf{y}) = (\mathbf{x}, \mathbf{z}).$$

It forgets the last b_i coordinates, so it is an *orthogonal projection* onto the retained coordinate space. It is onto: every $(\mathbf{x}, \mathbf{z}) \in \Lambda_{i-1}$ can be extended by choosing an integer $\mathbf{y}$ with $\mathbf{y} = -B_{\text{new}}\mathbf{x} \pmod q$, where B_{new} contains rows $n_{i-1} + 1, \ldots, n_i$ of B. Thus projecting Λ_r onto its first $h + n_i$ coordinates gives exactly Λ_i. Consecutive lattices have different dimensions and are related by projection. The sampling algorithm travels in the reverse direction, from fewer coordinates to more coordinates.

The algorithm of [DEL25] starts with sufficiently many Gaussian samples on Λ_0, using lemma 2.7. At stage i, it converts samples on Λ_{i-1} at width s_{i-1} into approximate Gaussian samples on Λ_i at width $s_i = \sqrt{2}s_{i-1}$, while retaining a constant fraction of the list.

Introducing the superlattice Temporarily write $b = b_i$. For an old vector $(\mathbf{x}, \mathbf{z}) \in \Lambda_{i-1}$, the new coordinates allowed by Λ_i form $q\mathbb{Z}^b - B_{\text{new}}\mathbf{x}$. Choose a full-rank rational lattice instead $S \supseteq q\mathbb{Z}^b$ and allow them to lie in the denser set $S - B_{\text{new}}\mathbf{x}$. Following [DEL25], define

$$\widetilde{\Lambda}_i = \{(\mathbf{x}, \mathbf{z}, \mathbf{y}) \,:\, (\mathbf{x}, \mathbf{z}) \in \Lambda_{i-1},\ \mathbf{y} + B_{\text{new}}\mathbf{x} \in S\}.$$

The projection extends to $\widetilde{\pi}_i : \widetilde{\Lambda}_i \twoheadrightarrow \Lambda_{i-1}$. Its *kernel*, the vectors it maps to zero, is $\{0\}^{h+n_{i-1}} \times S$. Thus the smoothing condition for lifting is controlled by S in the new coordinates. Moreover,

$$\Lambda_i \subseteq \widetilde{\Lambda}_i, \qquad [\widetilde{\Lambda}_i : \Lambda_i] = [S : q\mathbb{Z}^b].$$

Indeed, the map sending $(\mathbf{x}, \mathbf{z}, \mathbf{y})$ to $\mathbf{y} + B_{\text{new}}\mathbf{x} \bmod q\mathbb{Z}^b$ is onto $S/q\mathbb{Z}^b$, and its kernel is Λ_i. Its value is the *coset label* used for pairing. For $S = (q/p)\mathbb{Z}^b$, with positive integer p, there are p^b labels. These labels are computable from a lattice basis.

Gaussian lifting Given $\mathbf{u} = (\mathbf{x}, \mathbf{z}) \in \Lambda_{i-1}$, sample $\mathbf{y} \sim D_{S-B_{\text{new}}\mathbf{x},s}$ and append it to obtain $(\mathbf{u}, \mathbf{y}) \in \widetilde{\Lambda}_i$. This is the coordinate form of [DEL25, Algorithm 2], a special case of [EWY23, Algorithm 2].

Lemma 2.12 (Gaussian lifting, [DEL25, Lemma 3.1])**.** *If $0 < \varepsilon \le 1/2$ and $s \ge \eta_\varepsilon(S)$, lifting an independent sample from $D_{\Lambda_{i-1},s}$ by the preceding procedure gives a distribution 3ε-similar to $D_{\widetilde{\Lambda}_i,s}$. Each lift uses one shifted Gaussian sample on S and polynomial-time additional work.*

To see why this condition appears, the probability of an output $(\mathbf{u}, \mathbf{y})$ is

$$\frac{\rho_s(\mathbf{u})\rho_s(\mathbf{y})}{\rho_s(\Lambda_{i-1})\rho_s(S - B_{\text{new}}\mathbf{x})}.$$

The numerator is already the desired Gaussian weight, since $\rho_s(\mathbf{u})\rho_s(\mathbf{y}) = \rho_s((\mathbf{u}, \mathbf{y}))$. The extra dependence comes from $\rho_s(S - B_{\text{new}}\mathbf{x})$. By lemma 2.9, the ratio of any two such shifted masses is at most $(1+\varepsilon)/(1-\varepsilon) \le e^{3\varepsilon}$. After normalization, this gives the claimed 3ε-similarity. The correction in section 3 will cancel this denominator instead of requiring it to be almost constant.

Pairing returns to the target lattice Two vectors in $\widetilde{\Lambda}_i$ with the same coset label have their difference in Λ_i. We therefore group the input list by label and pair entries within each group, using each entry at most once. Consequently, a list of M entries in I cosets gives at least $(M-I)/2$ pairs, and hence at least $\lfloor M/3 \rfloor$ when $M \ge 3I$ [DEL25, Lemma 3.2].

Lemma 2.13 (Gaussian differences within cosets)**.** *Let $\Lambda \subseteq \Lambda'$ be full-rank lattices, $0 < \varepsilon \le 1/2$, and $s/\sqrt{2} \ge \eta_\varepsilon(\Lambda)$. For $M \ge 3[\Lambda' : \Lambda]$ independent samples from $D_{\Lambda',s}$, the preceding pairing procedure returns $M_0 = \lfloor M/3 \rfloor$ differences whose joint distribution has distance at most $3M_0\varepsilon$ from $D_{\Lambda,\sqrt{2}s}^{\otimes M_0}$.*

The proof is in section A.2. The full conditional-similarity analysis of this construction appears in [DEL25, Algorithms 3–4 and Lemmas 3.2–3.4].

From Gaussian vectors to the LWE secret For $\mathbf{x} \in \Lambda_q^\perp(A)$, we have $\langle \mathbf{x}, \mathbf{b} \rangle = \langle \mathbf{x}, \mathbf{e} \rangle \pmod q$. Thus the phase $2\pi\langle \mathbf{x}, \mathbf{b} \rangle/q$ depends only on the error. Its Gaussian Fourier bias can be used to recover the error, and hence the secret [AR05, Reg05]. The next lemma states sufficient conditions directly in terms of the error width σ and the sampling width s.

Lemma 2.14 (LWE recovery from Gaussian samples)**.** *Let q be prime, $m > n$, and let the LWE errors be independent $D_{\mathbb{Z},\sigma}$ samples. Suppose $\sigma\sqrt{\ln n} = o(q)$, $s \le q$, and $s^2/\ln n \to \infty$. If $\ln N \gg \ln n$ and*

$$\frac{m\sigma^2 s^2}{q^2} \le \frac{\ln N}{100}, \qquad q^{n-m}(1+q/s)^m \le N^{-1}.$$

Then N independent samples from $D_{\Lambda_q^\perp(A),s}$ allow recovery of the secret with probability $1-o(1)$, with time $N\,\mathrm{poly}(n, \ln N)$. For a list generated whose distribution is jointly δ-close to these samples, the success probability is at least $1 - o(1) - \delta$.

The first inequality preserves enough Fourier bias; the second makes incorrect candidates have small total Gaussian weight. The proof and recovery procedure are in section A.3.

3 A warm-up: correcting the lifting step

We now give our first construction. The key observation is that the lifting step of [DEL25] only needs constant-accuracy smoothing if we correct its outputs by rejection sampling. The rejection probability is easy to compute: a shifted Gaussian mass on a scaled integer lattice is a product of one-dimensional sums. This lets us use larger blocks at each stage and eventually solve constant-error LWE with $m = 4n$ samples.

3.1 An efficiently computable rejection probability

Consider stage i of the chain in section 2.6. Sample $\mathbf{u} = (\mathbf{x}, \mathbf{z})$ from $D_{\Lambda_{i-1},s}$ and then sample $\mathbf{y}$ from $D_{S-B_{\mathrm{new}}\mathbf{x},s}$. The resulting vector $(\mathbf{u}, \mathbf{y})$ belongs to $\widetilde{\Lambda}_i$. Its probability is

$$\mathbb{P}[\text{proposal} = (\mathbf{u}, \mathbf{y})] = \frac{\rho_s(\mathbf{u})\rho_s(\mathbf{y})}{\rho_s(\Lambda_{i-1})\rho_s(S - B_{\mathrm{new}}\mathbf{x})}. \tag{3.1}$$

The numerator is already the Gaussian weight we want. The denominator contains $\rho_s(S - B_{\mathrm{new}}\mathbf{x})$, which depends on the old coordinates. The following rejection step cancels this factor.

Lemma 3.1 (Exact lifting correction)**.** *Accept the proposed vector $(\mathbf{u}, \mathbf{y})$ with probability*

$$a(\mathbf{u}) = \frac{\rho_s(S - B_{\mathrm{new}}\mathbf{x})}{\rho_s(S)}. \tag{3.2}$$

This probability is valid, and the accepted vector has distribution $D_{\widetilde{\Lambda}_i,s}$. If $s \ge \eta_\varepsilon(S)$, the acceptance probability is at least $(1-\varepsilon)/(1+\varepsilon)$.

Proof. By lemma 2.6, $0 < a(\mathbf{u}) \le 1$. Multiplying (3.1) by the acceptance probability gives

$$\mathbb{P}[(\mathbf{u}, \mathbf{y}) \text{ is proposed and accepted}] = \frac{\rho_s(\mathbf{u})\rho_s(\mathbf{y})}{\rho_s(\Lambda_{i-1})\rho_s(S)}.$$

The denominator is independent of the proposed vector, and $\rho_s(\mathbf{u})\rho_s(\mathbf{y}) = \rho_s((\mathbf{u}, \mathbf{y}))$. Dividing by the total acceptance probability therefore gives exactly $D_{\widetilde{\Lambda}_i,s}$. Finally, lemma 2.9 bounds every ratio in (3.2) below by $(1-\varepsilon)/(1+\varepsilon)$. □

Remark 3.2. A rejected vector must be discarded in its entirety, including its old coordinates. Related corrections of Gaussian normalization factors appear in [BLPRS13, Section 5.2].

We choose $S = a\mathbb{Z}^b$, so the required mass separates into coordinates:

$$\rho_s(a\mathbb{Z}^b + \mathbf{c}) = \prod_{j=1}^{b} \rho_{s/a}(\mathbb{Z} + c_j/a).$$

Each factor is a sum of positive real numbers. Thus the correction can be computed without enumerating points in a high-dimensional lattice.

Lemma 3.3 (Computing the lifting probability)**.** *For rational input data with $s/a \geq 1$, the ratio*

$$\frac{\rho_s(a\mathbb{Z}^b + \mathbf{c})}{\rho_s(a\mathbb{Z}^b)}$$

can be evaluated to additive error ξ in time polynomial in b, s/a, the input encoding length, and $\ln(1/\xi)$.

Its proof, including the truncation error for the one-dimensional sums, is in section A.5.

3.2 The algorithm and its correctness

We use the chain of [DEL25], insert the preceding correction into the lifting step, and then pair vectors within each coset of Λ_i. Initialization and the new coordinates are based on lemma 2.7.

Both constructions retain a fixed number of vectors at each stage: start with $24^r N$ vectors and retain $24^{r-i}N$ after stage i. The following bound measures the joint output distribution, treating Fail as one extra possible output. We give a more general version so that it can apply directly in section 4. Its proof is in section A.4.

Lemma 3.4 (Fixed-size lists)**.** *Suppose each stage has at most N cosets. On ideal independent inputs, assume accepted lifts and accepted differences are each $O(\varepsilon)$-close to their target Gaussians, with success probabilities at least $4/5$ and $2/5$, respectively. With error at most ξ per sampling or rejection call, the fixed-size procedure has joint output distance at most*

$$O(24^r N(\varepsilon + \xi)) + r\exp(-\Omega(N)) \tag{3.3}$$

from N independent samples from the final Gaussian.

Write $N = \lceil e^H \rceil$ and $r = O(\ln n)$, then the first term in (3.3) is $e^{H+O(\ln n)}O(\varepsilon + \xi)$. Since the ideal target Gaussian distribution never fails, the same bound also controls the failure probability.

Algorithm 1 Corrected lifting with scaled integer auxiliaries

Require: Systematic matrix $[B \mid I_n]$; block sizes $b_1, \dots, b_r$ summing to n; width s_0; positive integers $p_1, \dots, p_r$; final count N.

Independently sample $24^r N$ vectors from $D_{\mathbb{Z}^h, s_0}$.
for $i = 1, \dots, r$ **do**
  Set $s = s_{i-1} = s_0 2^{(i-1)/2}$ and $S_i = (q/p_i)\mathbb{Z}^{b_i}$; let M be the current list size.
  for each current vector $\mathbf{u} = (\mathbf{x}, \mathbf{z})$ **do**
    Sample $\mathbf{y} \leftarrow D_{S_i - B_{\text{new}}\mathbf{x}, s}$.
    Keep $(\mathbf{u}, \mathbf{y})$ with probability $\rho_s(S_i - B_{\text{new}}\mathbf{x})/\rho_s(S_i)$; otherwise discard the whole vector.
  end for
  Retain the first $M/2$ accepted vectors; return FAIL if there are too few.
  Group them by coset modulo Λ_i and form disjoint pairs.
  Form their differences and retain the first $24^{r-i}N$; return FAIL if there are too few.
end for
return the final list on $\Lambda_r = \Lambda_q^{\perp}(A)$.

Theorem 3.5 (Correctness and cost of corrected lifting)**.** *Let $N = \lceil e^H \rceil$, $H \gg \ln n$, $r = O(\ln n)$, and $0 < \varepsilon \le 1/10$. Suppose $s_0 \ge 1$, and at every stage*

$$p_i^{b_i} \le e^H, \qquad s_{i-1} \ge \max\{\eta_{1/10}(S_i), q/p_i\}, \qquad s_{i-1}/\sqrt{2} \ge \eta_\varepsilon(\Lambda_i).$$

With per-operation error at most ξ, the output of algorithm 1 has distance at most (3.3) from $D_{\Lambda_q^{\perp}(A), s_r}^{\otimes N}$, where $s_r = s_0 2^{r/2}$. For polynomially bounded dimensions, scaled widths, and input encoding lengths, its time and space are $e^{H+O(\ln n)}\,\mathrm{poly}(\ln(1/\xi))$.

Proof. The conditions $s_0 \ge 1$ and $s_{i-1} \ge q/p_i$ permit all uses of lemma 2.7. By lemma 3.1, an ideal accepted lift is exactly Gaussian, and its acceptance probability is at least $9/11 > 4/5$. There are $p_i^{b_i}$ cosets. Conditional on their labels, the proof of lemma 2.13 in section A.2 gives independent differences, each $O(\varepsilon)$-close to $D_{\Lambda_i, \sqrt{2}s_{i-1}}$. There is no rejection in this pairing step. Thus lemma 3.4 gives the distributional conclusion, including the shortage events.

There are $e^{H+O(\ln n)}$ vector operations. Sampling the integer coordinates, evaluating the rejection probabilities by lemma 3.3, and computing and sorting coset labels all take polynomial time per operation. The fixed list sizes give the same bound on space. □

3.3 Parameter choice and smoothing requirements

We now choose parameters that meet the preceding theorem with $m = 4n$. Set $h = 3n$, let $q \in [n^2, 2n^2]$ be prime, and, for sufficiently large n, put

$$H = \frac{4n}{\ln \ln n}, \qquad s_0 = C\sqrt{H}, \qquad N = \lceil e^H \rceil. \tag{3.4}$$

The constant C will make the integer contribution to target smoothing smaller than e^{-100H}. At stage i, with $n_{i-1} = b_1 + \cdots + b_{i-1}$ rows already imposed, choose

$$\begin{aligned} b_i &= \min\left\{ n - n_{i-1}, \left\lfloor \frac{H}{2(\ln(q/s_{i-1}) + \ln\ln n)} \right\rfloor \right\}, \quad s_{i-1} = s_0 2^{(i-1)/2}, \\ p_i &= \left\lceil C_1 \frac{q}{s_{i-1}} \sqrt{\ln(b_i + 2)} \right\rceil. \end{aligned} \tag{3.5}$$

Choose C_1 large enough that lemma 2.10 gives $s_{i-1} \geq \eta_{1/10}((q/p_i)\mathbb{Z}^{b_i})$. Also $s_{i-1}p_i/q \geq C_1\sqrt{\ln(b_i+2)} \geq 1$, so the shifted integer sampler satisfies its width requirement. Both conditions are therefore explicit consequences of the choice of p_i.

Lemma 3.6 (Warm-up schedule)**.** *For sufficiently large constants C, C_1, the schedule (3.5) completes in $r = O(\ln n)$ stages and satisfies $p_i^{b_i} \leq e^{H/2}$ and*

$$n \leq s_r \leq q/\ln n. \tag{3.6}$$

With probability $1 - e^{-\Omega(n)}$ over B, every stage also satisfies $s_{i-1}/\sqrt{2} \geq \eta_{e^{-30H}}(\Lambda_i)$.

The proof is in section B.1. The reason why our algorithm can solve the stated LWE problem while [DEL25] fails is the following. By rejection sampling, we can lose the choice of smoothing parameter $\eta_\varepsilon(\frac{q}{p}\mathbb{Z}^b)$. For the same Gaussian width s_i, we can then pick a bigger block (i.e. b_i) at each stage. (Remember the choice of s_i restricts p_i and b_i.) Therefore, we can use less number of stages, which leads to a smaller Gaussian width for the final samples in $\Lambda_q^\perp(A)$.

Set $\varepsilon = e^{-30H}$ and implement each sampling or rejection operation with error at most $\xi = e^{-40H}$. By theorem 3.5 and lemma 3.6, the final list is jointly $e^{-\Omega(H)}$-close to independent samples from $D_{\Lambda_q^\perp(A),s_r}$, except for the stated matrix failure. The time and space are $e^{H+O(\ln n)} = e^{H+o(H)}$.

3.4 The constant-error LWE theorem

It remains to use the sampled vectors to recover the LWE secret. A constant-width error leaves a detectable Fourier signal for every endpoint in (3.6).

Theorem 3.7 (Search-LWE with constant-width errors)**.** *For prime $q \in [n^2, 2n^2]$, $m = 4n$, and error distribution $D_{\mathbb{Z},\sigma}$ with $\sigma = O(1)$, search-LWE can be solved with probability $1 - o(1)$ in time and space $\exp(O(n/\ln\ln n))$.*

Proof. Apply algorithm 1 with (3.4) and (3.5). Its output is jointly $o(1)$-close to $N = \lceil e^H \rceil$ independent Gaussian samples, where $H = 4n/\ln\ln n$. By (3.6), $\ln n \leq q/s_r \leq 2n$ and $s_r \geq n$. Consequently,

$$\frac{m\sigma^2 s_r^2}{q^2} = O\left(\frac{n}{(\ln n)^2}\right) = o(H), \qquad q^{-3n}(1 + q/s_r)^{4n} \leq e^{-2n\ln n + O(n)} \leq e^{-H}.$$

The remaining hypotheses of lemma 2.14 hold because $\sigma = O(1)$ and $s_r \geq n$. Recovery succeeds with probability $1 - o(1)$ and adds polynomial work per sample. Combining this with the sampling cost proves the claim. □

4 Random superlattices and corrected Gaussian differences

The algorithm in section 3 corrects the lifting step, but its pairing step still requires very accurate smoothing of the target lattice. We now correct the pairing step as well. The Gaussian mass that occurs in a difference can be split into two parts. One part comes from integer dual vectors and can be computed efficiently. The other part comes from noninteger dual vectors and is small for the random q-ary lattices used here. Rejection sampling removes the first part, leaving only the small error from the second. Based on this, we can replace the lifting superlattice by a random lattice with a better smoothing bound.

4.1 Random auxiliary lifting lattice

We first construct the auxiliary lattice used to lift a block of b coordinates. Choose a prime ℓ, a uniformly random k-dimensional linear subspace $C \subseteq \mathbb{F}_\ell^b$, and set

$$S = \frac{q}{\ell}(C + \ell\mathbb{Z}^b). \tag{4.1}$$

Here $C + \ell\mathbb{Z}^b$ denotes the integer vectors whose residues modulo ℓ belong to C. Directly from the definition,

$$q\mathbb{Z}^b \subseteq S, \qquad I := [S : q\mathbb{Z}^b] = \ell^k, \qquad a := \det(S)^{1/b} = \frac{q}{\ell^{k/b}},$$

and

$$S^* = \frac{1}{q}\{\mathbf{z} \in \mathbb{Z}^b : \mathbf{z} \bmod \ell \in C^\perp\}.$$

The scalar a describes the determinant per coordinate.

The smoothing proof below uses the volume of a Euclidean ball. Write $B_d(R) = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\| \le R\}$. Then

$$\mathrm{vol}(B_d(R)) = \frac{\pi^{d/2}}{\Gamma(d/2+1)}R^d, \qquad \mathrm{vol}(B_d(R)) \le \left(\sqrt{\frac{2\pi e}{d}}\,R\right)^d.$$

Lemma 4.1 (Smoothing a random auxiliary lattice)**.** *There are absolute constants $c_0, C_2, A_0 > 0$ such that, if $A_* = \ell^{k/b} \ge A_0$ and $\ell > 2c_0\sqrt{b}A_*$, then with probability at least $1 - 2^{-b}$ the lattice in (4.1) satisfies, for every $0 < \varepsilon < 1/2$,*

$$\eta_\varepsilon(S) \le C_2 a\sqrt{1 + \frac{\ln(1/\varepsilon)}{b}}. \tag{4.2}$$

Proof. We first show that the dual lattice has no very short nonzero vector. For a nonzero residue $\mathbf{z} \bmod \ell$, the probability that it belongs to $C^\perp$ is $(\ell^{b-k} - 1)/(\ell^b - 1) \le \ell^{-k}$. Let $r_* = c_0\sqrt{b}A_*$. The number of integer points of norm at most r_* is bounded by the volume of the ball of radius $r_* + \sqrt{b}/2$: put a disjoint unit cube around each point. The ball-volume bound then gives

$$\#(\mathbb{Z}^b \cap B(r_*)) \le \left[\sqrt{2\pi e}\,(c_0A_* + 1/2)\right]^b \le (A_*/2)^b.$$

The last inequality holds with $c_0 \le 1/(4\sqrt{2\pi e})$ and $A_0 \ge 2\sqrt{2\pi e}$. As $r_* < \ell$, a union bound shows that, except with probability 2^{-b},

$$\lambda_1(S^*) > r_*/q = c_0\sqrt{b}/a. \tag{4.3}$$

Disjoint balls of radius $\lambda_1(S^*)/2$ around dual lattice points show that the number of points of norm below $(j+1)\lambda_1(S^*)$ is at most $(2j+3)^b$. Consequently

$$\rho_{1/t}(S^* \setminus \{0\}) \le \sum_{j\ge 1}(2j+3)^b e^{-\pi t^2 \lambda_1(S^*)^2 j^2}.$$

If $t\lambda_1(S^*) \ge C\sqrt{b + \ln(1/\varepsilon)}$, this is at most ε: use $2j+3 \le 5^{j^2}$ and a large enough constant C. Combine this with (4.3) to obtain (4.2).

□

For $a\mathbb{Z}^b$, the usual smoothing estimate contains $\sqrt{\ln(b/\varepsilon)}$. The random construction replaces this by $\sqrt{1+\ln(1/\varepsilon)/b}$. When $\ln(1/\varepsilon)$ is much larger than b, this saves a factor of order $\sqrt{b}$ in the required width.

We also need to sample from a shifted copy of S. The following algorithm applies to the rational auxiliary lattices above.

Theorem 4.2 (Shifted Gaussian sampling, [ADS15, Theorem 4.7]). *Let $S \subset \mathbb{R}^b$ be a full-rank rational lattice. For rational $\mathbf{c}$, s satisfying*

$$s > \operatorname{dist}(-\mathbf{c}, S)/2^{\sqrt{b}}, \tag{4.4}$$

one can obtain a sample within total variation distance $2^{-\Omega(b^2)}$ of $D_{S+\mathbf{c},s}$ in time

$$2^{b+O(\sqrt{b}\log b)}\operatorname{poly}(\operatorname{bits}(S,\mathbf{c},s)).$$

This is the cited theorem with $f(b) = 2^{\sqrt{b}}$, retaining one output sample. As $S \supseteq q\mathbb{Z}^b$, every shift is at distance at most $q\sqrt{b}/2$ from S, so condition $s > q\sqrt{b}/(2\cdot 2^{\sqrt{b}})$ is easy to check.

4.2 Splitting the shifted Gaussian mass

The convolution formula in (2.3) contains a shifted Gaussian mass on the target lattice. To correct it, we first describe the part of this mass that we can compute. For a width $v > 0$ and a shift $\mathbf{t} \in \mathbb{R}^d$, define the *normalized shifted integer mass* by

$$T_v(\mathbf{t}) = v^{-d}\rho_v(\mathbb{Z}^d + \mathbf{t}) = \prod_{j=1}^{d}\left(1 + 2\sum_{k\ge 1} e^{-\pi v^2 k^2}\cos(2\pi k t_j)\right). \tag{4.5}$$

Lemma 4.3 (Computing the integer contribution). *For every $v > 0$ and $\mathbf{t} \in \mathbb{R}^d$,*

$$|T_v(\mathbf{t}) - 1| \le \rho_{1/v}(\mathbb{Z}^d) - 1.$$

For $v \ge 1$ and rational inputs $v^2, \mathbf{t}$, one can compute $T_v(\mathbf{t})$ to additive error $\xi \in (0,1)$ in time polynomial in d, the input bit length, and $\ln(1/\xi)$.

The inequality follows by bounding all nonzero Fourier terms by their absolute values. The computation and its precision bounds are proved in section A.5. In particular, $\rho_{1/v}(\mathbb{Z}^d) - 1 \le 1/10$ implies $9/10 \le T_v(\mathbf{t}) \le 11/10$ for every shift.

Now let $\Lambda \subseteq \mathbb{Z}^d$ be the target lattice. Its dual contains $\mathbb{Z}^d$, so we can split the Poisson sum into integer and noninteger dual vectors. Write

$$\delta_v(\Lambda) = \rho_{1/v}(\Lambda^* \setminus \mathbb{Z}^d), \qquad E_v(\mathbf{t}) = \sum_{\mathbf{y} \in \Lambda^* \setminus \mathbb{Z}^d} e^{-\pi v^2 \|\mathbf{y}\|^2} e^{2\pi i \langle \mathbf{y}, \mathbf{t} \rangle}.$$

The integer part of the sum is exactly $T_v(\mathbf{t})$. Thus (2.2) becomes

$$\rho_v(\Lambda + \mathbf{t}) = \frac{v^d}{\det \Lambda} \big(T_v(\mathbf{t}) + E_v(\mathbf{t}) \big), \qquad |E_v(\mathbf{t})| \le \delta_v(\Lambda). \tag{4.6}$$

We do not need to compute $E_v(\mathbf{t})$. Instead, lemma 2.11 bounds it uniformly over all shifts for the lattices in our chain.

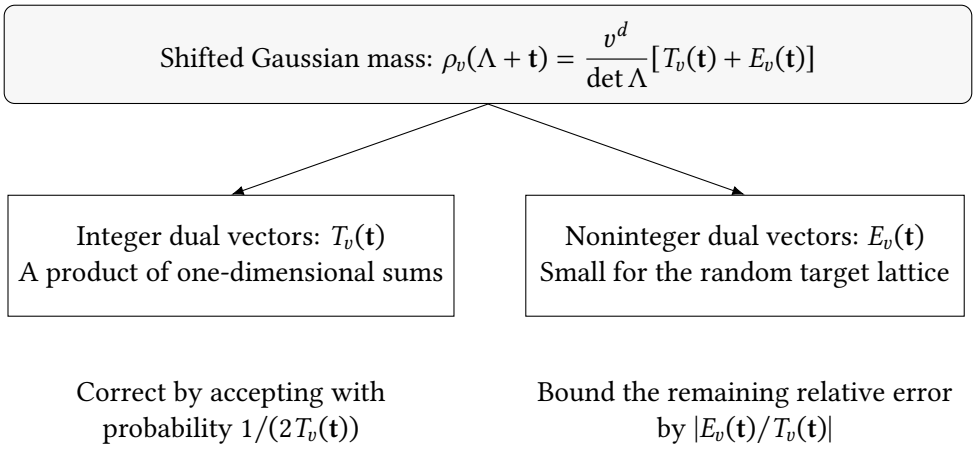

Figure 2: The computable integer contribution is removed by rejection sampling. The noninteger contribution remains, but is uniformly small.

4.3 Correcting a Gaussian difference

Suppose two independent Gaussian vectors belong to the same coset $\Lambda + \mathbf{c}$. Their difference belongs to Λ, but its distribution contains the extra mass factor in (2.3). We correct the integer part of this factor by the following rejection step.

Algorithm 2 Corrected difference of a same-coset pair

Require: $\mathbf{u}, \mathbf{v}$ in a common coset of $\Lambda \subseteq \mathbb{Z}^d$; sampling width s; probability error ξ.

1: Set $v = s/\sqrt{2}$ and $\mathbf{t} = (\mathbf{u} + \mathbf{v})/2$.
2: Compute $1/(2T_v(\mathbf{t}))$ to additive error ξ using lemma 4.3.
3: With that probability, return $\mathbf{u} - \mathbf{v}$; otherwise return Reject.

Lemma 4.4 (Correctness of algorithm 2). *Let* $\mathbf{u}, \mathbf{v}$ *be independent* $D_{\Lambda+\mathbf{c},s}$ *samples, where* $\Lambda \subseteq \mathbb{Z}^d$, *and put* $v = s/\sqrt{2} \geq 1$. *Suppose*

$$\rho_{1/v}(\mathbb{Z}^d) - 1 \leq 1/10, \qquad \delta_v(\Lambda) \leq \varepsilon \leq 1/10.$$

With exact probabilities, algorithm 2 accepts with probability at least $5/11$, *and its accepted output is* 3ε*-similar to* $D_{\Lambda,\sqrt{2}s}$. *Computing with error* ξ *costs polynomial time in the input length and* $\ln(1/\xi)$, *and changes the output distribution by at most* ξ.

Proof. We first analyze the exact probabilities. By lemma 4.3, $9/10 \leq T_v(\mathbf{t}) \leq 11/10$, so every acceptance probability lies between $5/11$ and $5/9$.

Fix a possible difference $\mathbf{w} \in \Lambda$, (2.3) gives

$$\mathbb{P}[\mathbf{u} - \mathbf{v} = \mathbf{w}] = \frac{\rho_{\sqrt{2}s}(\mathbf{w})\rho_v(\Lambda + \mathbf{c} - \mathbf{w}/2)}{\rho_s(\Lambda + \mathbf{c})^2}.$$

Since

$$\frac{\mathbf{u}+\mathbf{v}}{2} - \left(\mathbf{c} - \frac{\mathbf{w}}{2}\right) = \mathbf{u} - \mathbf{c} \in \Lambda \subseteq \mathbb{Z}^d.$$

The periodicity of T_v implies that the value used by the algorithm is $T_v(\mathbf{c} - \mathbf{w}/2)$.

Multiplying by the acceptance probability and then using (4.6), we get

$$\begin{aligned}
&\mathbb{P}[\mathbf{u} - \mathbf{v} = \mathbf{w}, \text{accept}] \\
&\quad = \frac{\rho_{\sqrt{2}s}(\mathbf{w})\rho_v(\Lambda + \mathbf{c} - \mathbf{w}/2)}{2\rho_s(\Lambda + \mathbf{c})^2 T_v(\mathbf{c} - \mathbf{w}/2)} \\
&\quad = \frac{v^d}{2 \det \Lambda\, \rho_s(\Lambda + \mathbf{c})^2}\rho_{\sqrt{2}s}(\mathbf{w})\big(1 + r_{\mathbf{c}}(\mathbf{w})\big), \qquad r_{\mathbf{c}}(\mathbf{w}) = \frac{E_v(\mathbf{c} - \mathbf{w}/2)}{T_v(\mathbf{c} - \mathbf{w}/2)}.
\end{aligned}$$

Hence the exact accepted distribution is

$$\mathbb{P}[\mathbf{u} - \mathbf{v} = \mathbf{w} \mid \text{accept}] = \frac{\rho_{\sqrt{2}s}(\mathbf{w})(1 + r_{\mathbf{c}}(\mathbf{w}))}{\sum_{\mathbf{z}\in\Lambda} \rho_{\sqrt{2}s}(\mathbf{z})(1 + r_{\mathbf{c}}(\mathbf{z}))}. \tag{4.7}$$

It remains to compare this expression with the target Gaussian. The bounds on E_v and T_v give $|r_{\mathbf{c}}(\mathbf{w})| \leq 10\varepsilon/9$. The denominator in (4.7), divided by $\rho_{\sqrt{2}s}(\Lambda)$, lies between $1 - 10\varepsilon/9$ and $1 + 10\varepsilon/9$. Consequently the ratio of the accepted probability to $D_{\Lambda,\sqrt{2}s}(\mathbf{w})$ lies in $\left[\frac{1-10\varepsilon/9}{1+10\varepsilon/9}, \frac{1+10\varepsilon/9}{1-10\varepsilon/9}\right]$. For $0 \leq \varepsilon \leq 1/10$, $\ln((1+10\varepsilon/9)/(1-10\varepsilon/9)) \leq 3\varepsilon$; this proves the claimed similarity. Finally, $T_v \geq 9/10$ can be computed to additive error ξ using lemma 4.3. Coupling the exact and approximate result bounds the output distribution by ξ. □

4.4 The full sampler and its correctness

We use the chain from section 2.6, with $n_i = \sum_{t=1}^{i} b_t$ and Λ_i of dimension $h + n_i$. Lifting uses the random auxiliary lattice S_i above its smoothing parameter. Pairing then uses algorithm 2. The factor 24 in the list sizes leaves enough vectors for both pairing and rejection, as lemma 3.4.

Algorithm 3 Gaussian sampling using random superlattices

Require: Matrix A; block sizes $b_1, \dots, b_r$ summing to n; widths $s_i = \sqrt{2}s_{i-1}$; code dimensions k_i; prime ℓ; $N = \lceil e^H \rceil$; numerical accuracy ξ.

Put A^T in systematic form $[B \mid I_n]$; return FAIL if the fixed pivot block is singular.
Sample $24^r N$ independent vectors on $\mathbb{Z}^h$ at width s_0.
for $i = 1, \dots, r$ **do**
    Choose an independent uniform k_i-dimensional subspace $C_i \subseteq \mathbb{F}_\ell^{b_i}$ and form S_i by (4.1).
    for each current vector $(\mathbf{x}, \mathbf{z})$ **do**
        Sample $\mathbf{y} \leftarrow D_{S_i - B_{\text{new}}\mathbf{x}, s_{i-1}}$ using theorem 4.2; append $\mathbf{y}$ to $(\mathbf{x}, \mathbf{z})$.
    end for
    Group the lifted vectors by coset modulo Λ_i and form disjoint pairs in input order.
    Apply algorithm 2 to each pair, using width s_{i-1} and probability error ξ.
    Retain the first $24^{r-i}N$ successful differences; return FAIL if there are too few.
end for
Return the N vectors in $\Lambda_r = \Lambda_q^\perp(A)$.

Theorem 4.5 (Correctness and cost)**.** *Suppose $r = O(\ln n)$, $H \gg \ln n$, and the input data have polynomial bit length. Fix a nonsingular pivot block and auxiliary lattices for which, at every stage i,*

$$[S_i : q\mathbb{Z}^{b_i}] \le e^H, \qquad s_{i-1} \ge \eta_\varepsilon(S_i),$$
$$v_i = s_{i-1}/\sqrt{2} \ge 1, \qquad \rho_{1/v_i}(\mathbb{Z}^{h+n_i}) - 1 \le 1/10, \qquad \delta_{v_i}(\Lambda_i) \le \varepsilon \le 1/10.$$

Assume also that theorem 4.2 holds and each sampling or numerical operation has error at most ξ. Then the output distribution of algorithm 3, with failure included, is within total variation distance

$$e^{H+O(\ln n)}O(\varepsilon + \xi) + r\exp(-\Omega(N))$$

of $N = \lceil e^H \rceil$ independent $D_{\Lambda_q^\perp(A), s_r}$ samples. Its time and space are at most

$$e^{H+O(\ln n)}2^{b_{\max}+O(\sqrt{b_{\max}}\log b_{\max})}\operatorname{poly}\big(n, \ln(1/\xi)\big).$$

Proof. For ideal independent inputs, lemma 2.12 makes each lifted distribution 3ε-similar to the required superlattice Gaussian. Conditional on any coset label, the exact Gaussian inputs to a pair are independent samples on that coset. Lemma 4.4 gives acceptance at least $5/11 > 0.4$ and an accepted distribution 3ε-similar to the next Gaussian, whose width is $s_i = \sqrt{2}s_{i-1}$. These are precisely the hypotheses of lemma 3.4, which gives the stated joint-distribution.

There are $e^{H+O(\ln n)}$ vector operations because $24^r = \operatorname{poly}(n)$. The most expensive operation is a shifted sample in dimension at most $b_{\max}$; its cost is given by theorem 4.2. Gaussian mass evaluation, coset computation, and sorting add polynomial overhead. This proves the cost bound. □

4.5 Choosing the parameters

We choose the widths first. The initial width must control the noninteger dual mass; the final width must retain enough signal for LWE recovery. Under the assumptions of Result 2, set

$$g = \frac{h\ln q}{m}, \qquad s_0 = qe^{-g/2}, \qquad F = C_3\sigma\sqrt{\ln R}, \qquad a_0 = \frac{\ln 2}{2},$$

where R is defined in (1.1) and C_3 is a sufficiently large constant. We stop when the width reaches approximately q/F. More precisely, write $L_i = \ln(q/s_i)$ and choose

$$r = \left\lfloor \frac{\ln(q/s_0) - \ln F}{a_0} \right\rfloor, \qquad s_i = 2^{i/2} s_0 \quad (0 \le i \le r). \tag{4.8}$$

The remaining choice is the block size at each round. We allocate the blocks approximately in proportion to $1/L_{i-1}$, allowing an additional $O(\ln(L_{i-1}+2))$ per coordinate for auxiliary smoothing. The explicit choice is given in section B.2.

Lemma 4.6 (Random-superlattice parameters)**.** *Under the assumptions of Result 2, block sizes and auxiliary code dimensions can be computed in polynomial time so that the hypotheses of theorem 4.5 hold with probability* $1 - o(1)$*, with* $\varepsilon = \xi = e^{-100H}$ *and*

$$H = (a_0 + o(1))\frac{n}{\ln R}, \qquad r = O(\ln n), \qquad b_{\max} = o(H), \qquad b_{\min}^2/H \longrightarrow \infty.$$

The implemented endpoint satisfies

$$q/(2F) < s_r \le q/F. \tag{4.9}$$

The proof is in section B.2.

4.6 The resulting LWE algorithm

We can now combine the sampler with lemma 2.14. The choice $q/s_r \asymp \sigma\sqrt{\ln R}$ ensures that the list is large enough to recover the secret, while the block bounds keep the cost of each shifted sample below the main list cost.

Theorem 4.7 (Search-LWE with random superlattices)**.** *Under the assumptions of Result 2, algorithm 3 followed by lemma 2.14 solves search-LWE with probability* $1 - o(1)$*, using* m *samples, in classical time and space*

$$2^{(1/2+o(1))n/\ln R}.$$

Proof. Lemma 4.6 and theorem 4.5 give a list jointly $e^{-\Omega(H)}$-close to $N = \lceil e^H \rceil$ independent $D_{\Lambda_q^\perp(A),s_r}$ samples. Since $b_{\max} = o(H)$, its time and space are $e^{H+o(H)}$.

For recovery, write $u = q/s_r \in [F, 2F)$. Taking C_3 sufficiently large gives

$$\frac{m\sigma^2}{u^2} \le \frac{m}{C_3^2 \ln R} \le \frac{H}{100}, \qquad q^{-h}(1+u)^m = \exp\bigl(-(1-o(1))mg\bigr) \le N^{-1}.$$

Here $\ln(1+u) = o(g)$ and $mg/H \to \infty$ follow from $R \to \infty$. Also $H \gg \ln n$, $\sigma = n^{o(1)}$, and $s_r = n^{\kappa - o(1)}$, which verify the remaining size conditions of lemma 2.14. Its polynomial overhead does not change the exponent. Finally, $a_0 = (\ln 2)/2$ converts $e^{H+o(H)}$ to the stated bound. □

4.7 Polynomial errors and Regev's parameters

The same construction applies to polynomial error widths after changing the starting width to $s_0 = (\ln n)^2$ and the stopping threshold to $F = C_3\sigma$, for a sufficiently large constant C_3. The parameter and recovery arguments are given in section B.3.

Lemma 4.8 (Polynomial-error parameters). *Let $q = n^{\kappa+o(1)}$ be prime, $\sigma = n^{\beta+o(1)}$, $0 < \beta < \kappa$, and $m = (c + o(1))n$ with $c > 1 + 1/\ln(\kappa/\beta)$. The allocation in section B.2, with the widths above, satisfies the hypotheses of theorem 4.5 with probability $1 - o(1)$, with $\varepsilon = \xi = e^{-100H}$ and*

$$H = \left(\frac{a_0}{\ln(\kappa/\beta)} + o(1)\right) n, \qquad b_i = \Theta(n/\ln n), \qquad \frac{q}{2F} < s_r \le \frac{q}{F}. \tag{4.10}$$

Theorem 4.9 (Polynomial-error LWE). *Under the assumptions of lemma 4.8, search-LWE can be solved with probability $1 - o(1)$ in classical time and space*

$$2^{(1/[2\ln(\kappa/\beta)]+o(1))n}.$$

Corollary 4.10 (Regev's parameters). *For prime $q = n^{2+o(1)}$ and $\sigma = n^{3/2+o(1)}$, the theorem gives*

$$T, S \le 2^{(1/[2\ln(4/3)]+o(1))n}, \qquad m = (c + o(1))n, \quad c > 1 + 1/\ln(4/3).$$

In particular, $m = 5n$ suffices, with time exponent approximately 1.73803.

Acknowledgements

This research was supported by the National Key Research Project of China under Grant No.2025YFA1017200.

Use of AI. The authors used OpenAI's GPT-6 Astra model to assist with working out, checking, and implementing the technical details of their initial ideas, as well as with drafting and revising the manuscript. The authors verified and revised the content and conclusions of the manuscript and take full responsibility.

References

[Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC 1996*, pp. 99–108, 1996. https://doi.org/10.1145/237814.237838.

[AR05] Dorit Aharonov and Oded Regev. Lattice problems in NP ∩ coNP. *Journal of the ACM*, 52(5):749–765, 2005. https://cims.nyu.edu/~regev/papers/cvpconp.pdf.

[ADRS15] Divesh Aggarwal, Daniel Dadush, Oded Regev, and Noah Stephens-Davidowitz. Solving the Shortest Vector Problem in 2^n Time via Discrete Gaussian Sampling. In *STOC 2015*, 2015. https://arxiv.org/abs/1412.7994.

[ADS15] Divesh Aggarwal, Daniel Dadush, and Noah Stephens-Davidowitz. Solving the Closest Vector Problem in 2^n Time—The Discrete Gaussian Strikes Again! In *FOCS 2015*, 2015. https://arxiv.org/abs/1504.01995.

[BKW03] Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM*, 50(4):506–519, 2003. https://doi.org/10.1145/792538.792543.

[BLPRS13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical Hardness of Learning with Errors. In *STOC 2013*, pp. 575–584, 2013. https://arxiv.org/abs/1306.0281.

[Bra12] Zvika Brakerski. Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. In *Advances in Cryptology—CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 868–886. Springer, 2012. https://doi.org/10.1007/978-3-642-32009-5_50.

[BV11] Zvika Brakerski and Vinod Vaikuntanathan. Efficient Fully Homomorphic Encryption from (Standard) LWE. In *FOCS 2011*, pp. 97–106, 2011. https://eprint.iacr.org/2011/344.

[CJWZ26] Zhili Chen, Kaijie Jiang, Anyu Wang, and Chuanqi Zhang. A Subexponential Algorithm for Binary LWE with Sublinear Samples. Cryptology ePrint Archive, Paper 2026/2289, 2026. https://eprint.iacr.org/2026/2289.

[CKKS17] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. Homomorphic Encryption for Arithmetic of Approximate Numbers. In *ASIACRYPT 2017*, pp. 409–437, 2017. https://doi.org/10.1007/978-3-319-70694-8_15.

[DEL25] Léo Ducas, Lynn Engelberts, and Johanna Loyer. Wagner's Algorithm Provably Runs in Subexponential Time for SIS^∞. In *CRYPTO 2025*, 2025. https://eprint.iacr.org/2025/575.

[DP26] Léo Ducas and Ludo N. Pulles. Accurate Score Prediction for Dual-Sieve Attacks. *Journal of Cryptology*, 39(1), article 8, 2026. https://doi.org/10.1007/s00145-025-09560-7.

[EWY23] Thomas Espitau, Alexandre Wallet, and Yang Yu. On Gaussian Sampling, Smoothing Parameter and Application to Signatures. In *ASIACRYPT 2023*, pp. 65–97, 2023. https://doi.org/10.1007/978-981-99-8739-9_3.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *STOC 2009*, pp. 169–178, 2009. Full thesis: https://crypto.stanford.edu/craig/.

[GFH26] Yiming Gao, Yansong Feng, and Honggang Hu. Solving the Shortest Vector Problem in $2^{0.7314n+o(n)}$ Time via Discrete Gaussian Sampling on Superlattices. Cryptology ePrint Archive, Paper 2026/1587, 2026. Preprint. https://eprint.iacr.org/2026/1587.

[GJS15] Qian Guo, Thomas Johansson, and Paul Stankovski. Coded-BKW: Solving LWE Using Lattice Codes. In *CRYPTO 2015*, pp. 23–42, 2015. https://doi.org/10.1007/978-3-662-47989-6_2.

[GJMS17] Qian Guo, Thomas Johansson, Erik Mårtensson, and Paul Stankovski. Coded-BKW with Sieving. In *ASIACRYPT 2017*, pp. 323–346, 2017. https://doi.org/10.1007/978-3-319-70694-8_12.

[GJMS19] Qian Guo, Thomas Johansson, Erik Mårtensson, and Paul Stankovski Wagner. On the Asymptotics of Solving the LWE Problem Using Coded-BKW with Sieving. *IEEE Transactions on Information Theory*, 2019. Author manuscript: https://bora.uib.no/bora-xmlui/handle/1956/23273.

[GMPW20] Nicholas Genise, Daniele Micciancio, Chris Peikert, and Michael Walter. Improved Discrete Gaussian and Subgaussian Analysis for Lattice Cryptography. In *PKC 2020*, 2020. https://www.iacr.org/archive/pkc2020/12110222/12110222.pdf.

[GSW13] Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based. In *CRYPTO 2013*, pp. 75–92, 2013. https://www.iacr.org/archive/crypto2013/80420297/80420297.pdf.

[HGH26] Xuyuan Han, Yiming Gao, and Honggang Hu. A Provable Subexponential-Time Algorithm for LWE from $n + o(n)$ Samples via Wagner-Style Gaussian Sampling. Cryptology ePrint Archive, Paper 2026/2261, 2026. https://eprint.iacr.org/2026/2261.

[Hhan26] Minki Hhan. Finding a Shortest Vector and More in $2^{n/2+o(n)}$ Time using q-ary Coset Difference Tree. arXiv:2609.02764, 2026. Preprint. https://arxiv.org/abs/2609.02764.

[HKM18] Gottfried Herold, Elena Kirshanova, and Alexander May. On the asymptotic complexity of solving LWE. *Designs, Codes and Cryptography*, 86(1):55–83, 2018. https://doi.org/10.1007/s10623-016-0326-0. Author manuscript: https://crypto-kantiana.com/elena.kirshanova/Papers/on_the_asymptotic_complexity.pdf.

[KF15] Paul Kirchner and Pierre-Alain Fouque. An Improved BKW Algorithm for LWE with Applications to Cryptography and Lattices. In *CRYPTO 2015*, pp. 43–62, 2015. https://arxiv.org/abs/1506.02717.

[Kim26] Jiseung Kim. One Discrete Gaussian Sample in $2^{n/2+o(n)}$ Time. arXiv:2608.03220, 2026. Preprint. https://arxiv.org/abs/2608.03220.

[LMK+23] Yongwoo Lee, Daniele Micciancio, Andrey Kim, Rakyong Choi, Maxim Deryabin, Jieun Eom, and Donghoon Yoo. Efficient FHEW Bootstrapping with Small Evaluation Keys, and Applications to Threshold Homomorphic Encryption. In *EUROCRYPT 2023*, pp. 227–256, 2023. https://doi.org/10.1007/978-3-031-30620-4_8.

[LV26] Haoxing Lin and Prashant Nalini Vasudevan. Provably Subexponential Algorithms for Bounded-Error LWE with Few Samples. arXiv:2610.13043v1, 2026. https://arxiv.org/abs/2610.13043v1.

[LZ26] Yechen Li and Qunxiong Zheng. What Happens When integrating Modulus Switching and Lossy Source Coding: A New Dual Attack Variant on LWE. Cryptology ePrint Archive, Paper 2026/1400, 2026. Preprint. https://eprint.iacr.org/2026/1400.

[MR07] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007. Preliminary version in *FOCS 2004*. https://cims.nyu.edu/~regev/papers/average.pdf.

[M19] Erik Mårtensson. The Asymptotic Complexity of Coded-BKW with Sieving Using Increasing Reduction Factors. arXiv:1901.06558, 2019. https://arxiv.org/abs/1901.06558.

[NIST24a] National Institute of Standards and Technology. *Module-Lattice-Based Key-Encapsulation Mechanism Standard*. Federal Information Processing Standards Publication 203, August 2024. https://doi.org/10.6028/NIST.FIPS.203.

[PS24] Amaury Pouly and Yixin Shen. Provable Dual Attacks on Learning with Errors. In *EUROCRYPT 2024*, pp. 256–285, 2024. https://eprint.iacr.org/2023/1508.

[PT26] Paola de Perthuis and Filip Trenkić. Refined Modelling of the Primal Attack, and Variants Against Module-LWE. In *PQCrypto 2026*, pp. 342–376, 2026. https://doi.org/10.1007/978-3-032-22698-3_11. Author manuscript: https://ir.cwi.nl/pub/36238/paper.pdf.

[QX25] Hongyuan Qu and Guangwu Xu. On the Provable Dual Attack for LWE by Modulus Switching. Cryptology ePrint Archive, Paper 2025/859, 2025. https://eprint.iacr.org/2025/859.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC 2005*, pp. 84–93, 2005. Full version: *Journal of the ACM*, 56(6), article 34, 2009. https://cims.nyu.edu/~regev/papers/qcrypto.pdf.

[WBLW25] Yu Wei, Lei Bi, Xianhui Lu, and Kunpeng Wang. Memory-Efficient BKW Algorithm for Solving the LWE Problem. In *PKC 2025*, pp. 331–362, 2025. https://eprint.iacr.org/2025/266.

[WWZZ26] Rui-Jie Wang, Zhong-Xiao Wang, Qun-Xiong Zheng, and Xuan Zhao. LaMS: A p-adic Layered Modulus Switching for Provable Dual Attacks on LWE. Cryptology ePrint Archive, Paper 2026/1326, 2026. Preprint. https://eprint.iacr.org/2026/1326.

A Proofs of distributional and sampling estimates

A.1 Gaussian tail bounds

We record the tail estimates used for the errors and sampled vectors.

Lemma A.1 (Gaussian tail bounds). *Let $\mathbf{x} \sim D_{\Lambda,s}$, where $\Lambda \subset \mathbb{R}^d$ is a full-rank lattice. For every coordinate j and $r > 0$,*

$$\mathbb{E}x_j^2 \le \frac{s^2}{2\pi}, \qquad \mathbb{P}[|x_j| > rs] \le 2e^{-\pi r^2}.$$

If $e_1, \dots, e_m$ are independent samples from $D_{\mathbb{Z},\sigma}$, then, for $a > 0$ and $B_0 > \ln 2/\pi$,

$$\mathbb{P}[\|\mathbf{e}\|_\infty > \sigma\sqrt{a \ln n}] \le 2mn^{-\pi a},$$
$$\mathbb{P}[\|\mathbf{e}\|^2 > B_0 m\sigma^2] \le \exp[-(\pi B_0 - \ln 2)m/2].$$

Proof. Completing the square and applying lemma 2.6, with ζ_j the jth unit vector, gives

$$\mathbb{E}e^{tx_j} = e^{s^2t^2/(4\pi)}\frac{\rho_s(\Lambda - s^2t\zeta_j/(2\pi))}{\rho_s(\Lambda)} \le e^{s^2t^2/(4\pi)}.$$

Symmetry and the quadratic term give the second moment. Markov's inequality at $t = 2\pi r/s$, applied to x_j and $-x_j$, gives the coordinate tail. A union bound gives the infinity-norm bound.

For the squared norm, an independent standard real Gaussian g gives, for $0 < t < \pi$,

$$\mathbb{E}e^{te_j^2/\sigma^2} = \mathbb{E}_g\mathbb{E}_{e_j}e^{\sqrt{2t}\,ge_j/\sigma} \le \mathbb{E}_g e^{tg^2/(2\pi)} = (1 - t/\pi)^{-1/2}.$$

Multiply over the independent coordinates and apply Markov's inequality at $t = \pi/2$ to obtain $e^{-\pi B_0 m/2}2^{m/2}$. □

A.2 Proof of lemma 2.13

Proof. Put $I = [\Lambda' : \Lambda]$. Each coset has at most one vector not paired, so there are at least $(M - I)/2 \ge M/3$ disjoint pairs. Fixing all coset labels fixes the pairing rule and leaves the inputs independent, each with distribution $D_{\Lambda+\mathbf{c},s}$ in its coset. By (2.3), the difference has Gaussian weights multiplied by $\rho_{s/\sqrt{2}}(\Lambda + \mathbf{c} - \mathbf{w}/2)$. By lemma 2.9, this factor is within $1 \pm \varepsilon$ of a constant independent of $\mathbf{w}$. Normalizing gives

$$\frac{1-\varepsilon}{1+\varepsilon} \le \frac{\mathbb{P}[\mathbf{u} - \mathbf{v} = \mathbf{w} \mid \mathbf{c}]}{D_{\Lambda,\sqrt{2}s}(\mathbf{w})} \le \frac{1+\varepsilon}{1-\varepsilon}.$$

Each difference is therefore at distance at most $\varepsilon/(1-\varepsilon) \le 2\varepsilon$ from the target. Conditional independence, lemma 2.4, and averaging over the labels give a joint bound of $2M_0\varepsilon$, which implies the stated bound. □

A.3 Proof of lemma 2.14

Proof. Put $u = q/s$. By lemma A.1, with probability $1 - o(1)$,

$$\|\mathbf{e}\|^2 \le m\sigma^2, \qquad \|\mathbf{e}\|_\infty = O(\sigma\sqrt{\ln n}) = o(q).$$

Fix such an error, independently of A, and put $\omega = \rho_u(\mathbf{e})$. The first hypothesis gives $\omega \ge N^{-\pi/100} \ge N^{-1/25}$.

For the supplied vectors $\mathbf{x}_1, \dots, \mathbf{x}_N$, compute

$$\widehat{f} = \frac{1}{N}\sum_i \cos\frac{2\pi\langle \mathbf{x}_i, \mathbf{b}\rangle}{q}, \qquad \widehat{\mathbf{g}} = \frac{1}{N}\sum_i \mathbf{x}_i \sin\frac{2\pi\langle \mathbf{x}_i, \mathbf{b}\rangle}{q}. \tag{A.1}$$

Round $q\widehat{\mathbf{g}}/(s^2\widehat{f})$ coordinatewise to obtain an error candidate, then solve $A\mathbf{s} = \mathbf{b} - \mathbf{e} \pmod q$. Report failure if the denominator vanishes, or the solution is not unique.

For exact Gaussian samples, let f and $\mathbf{g}$ denote the expectations. Since $\langle \mathbf{x}_i, \mathbf{b}\rangle = \langle \mathbf{x}_i, \mathbf{e}\rangle \pmod q$, Poisson summation and differentiation give

$$f = \frac{\rho_u(\Lambda_q(A) + \mathbf{e})}{\rho_u(\Lambda_q(A))}, \qquad \frac{q\mathbf{g}}{s^2 f} = \frac{\sum_{\boldsymbol{\ell}\in\Lambda_q(A)}(\mathbf{e}+\boldsymbol{\ell})\rho_u(\mathbf{e}+\boldsymbol{\ell})}{\rho_u(\Lambda_q(A)+\mathbf{e})} =: \boldsymbol{\mu}.$$

Indeed, the derivatives of the cosine average and Gaussian mass have factors $-2\pi/q$ and $-2\pi/u^2$, respectively. Thus $\boldsymbol{\mu}$ is a weighted average of possible errors; the correct error has weight ω.

We bound the other weights. For nonzero $\mathbf{a} \in \mathbb{F}_q^n$, $A\mathbf{a}$ is uniform, so

$$\begin{aligned}\mathbb{E}_A \sum_{\boldsymbol{\ell}\in\Lambda_q(A)\setminus q\mathbb{Z}^m} (1 + \|\boldsymbol{\ell}\|_1)\rho_u(\mathbf{e}+\boldsymbol{\ell}) &\le q^{n-m}\sum_{\mathbf{v}\in\mathbb{Z}^m}(1+\|\mathbf{v}-\mathbf{e}\|_1)\rho_u(\mathbf{v}) \\ &\le \operatorname{poly}(n) q^{n-m}(1+u)^m \le \operatorname{poly}(n) N^{-1}.\end{aligned}$$

The middle bound follows coordinatewise from $\rho_u(\mathbb{Z}) \le 1 + u$ and $\sum_{k\in\mathbb{Z}} |k|\rho_u(k) \le u^2/\pi + u$. Markov's inequality bounds this contribution by $\omega/32$ except with probability $\operatorname{poly}(n)N^{-24/25} = o(1)$. The same argument without the shift or weight bounds the unshifted contribution by $o(1)$ with probability $1 - o(1)$.

For the other representatives of the zero codeword, $\|\mathbf{e}+q\mathbf{v}\|^2 - \|\mathbf{e}\|^2 \ge (q^2/2)\|\mathbf{v}\|^2$ when $\|\mathbf{e}\|_\infty \le q/4$. Factoring the Gaussian sums gives

$$\sum_{\mathbf{v}\in\mathbb{Z}^m\setminus\{0\}} (1 + q\|\mathbf{v}\|_1)\rho_u(\mathbf{e}+q\mathbf{v}) \le \omega \operatorname{poly}(n) e^{-\pi s^2/2} = o(\omega),$$

where $s^2/\ln n \to \infty$. At zero shift, this also gives $\rho_u(q\mathbb{Z}^m) = 1 + o(1)$. Consequently, for sufficiently large n,

$$\|\boldsymbol{\mu} - \mathbf{e}\|_\infty \le 1/16, \qquad \rho_u(\Lambda_q(A)) \le 2, \qquad f \ge \omega/2.$$

The cosine statistic has variance at most one, and each coordinate of the sine statistic has variance at most $s^2/(2\pi)$ by lemma A.1. Chebyshev's inequality and a union bound give

$$|\widehat{f} - f| \le \frac{\omega}{64(1 + \|\mathbf{e}\|_\infty)}, \qquad \|\widehat{\mathbf{g}} - \mathbf{g}\|_\infty \le \frac{s^2\omega}{64q},$$

except with probability $\mathrm{poly}(n)/(N\omega^2) = o(1)$. Substitution bounds the error of the quotient relative to $\boldsymbol{\mu}$ by $1/8$. Its distance from $\mathbf{e}$ is less than $1/2$, so rounding recovers $\mathbf{e}$. Rank failure has probability at most $q^{n-m}/(q-1) \le N^{-1}$.

Coordinate tails allow truncation of large inputs with $o(1)$ failure, and the rounding margin permits precision polynomial in n and $\ln N$, giving the stated cost. Coupling the entire approximate list to exact samples adds at most δ to the failure probability. The list is generated from A alone, so this coupling preserves its independence from the error conditional on A. □

A.4 Proof of lemma 3.4

Proof. Consider a stage with target count N' and $24N'$ independent ideal inputs. Since lifting succeeds with probability at least $4/5$, a Chernoff bound gives $12N'$ successes except with probability $e^{-\Omega(N')}$. Coupling the retained lifts to exact Gaussians costs $O(N'\varepsilon)$.

Fix their coset labels. The entries remain independent, and disjoint pairing gives at least $(12N' - N)/2 \ge 11N'/2$ pairs. Each succeeds with probability at least $2/5$, so another Chernoff bound gives N' successes except with probability $e^{-\Omega(N')}$, uniformly over the labels. Conditional on the labels and success indicators, accepted differences are independent, and each is $O(\varepsilon)$-close to the target Gaussian. By lemma 2.4, the first N' outputs are jointly $O(N'\varepsilon)$-close, also after averaging. A step without rejection has success probability one.

Couple each implemented stage to this ideal experiment. By contraction, existing error does not increase, and each call adds at most ξ. Summing over $\sum_i O(24^{r-i+1}N) = O(24^r N)$ calls and the shortage events gives (3.3). Since the ideal distribution never returns FAIL, this also bounds the failure probability. □

A.5 Finite-precision Gaussian corrections

Both corrections use one-dimensional Gaussian sums. We give their precision bounds together, using the Gaussian-sum evaluation of [BLPRS13, Section 5.2].

Proposition A.2 (Finite Gaussian evaluations)**.** *In the parameter ranges of lemmas 3.3 and 4.3, the stated masses and acceptance probabilities can be computed to additive error ξ with their stated polynomial costs. If each sampling or rejection call has statistical error at most ξ, at most Q such calls add at most $Q\xi$ to the joint output error.*

Proof. The cited evaluation uses the positive Gaussian series for widths below one and its Poisson expansion otherwise. Truncating these rapidly convergent series gives any prescribed number of accuracy bits in polynomial time.

For lifting, reduce each shift to $y \in [-1/2, 1/2]$ and put $t = s/a \ge 1$. Since $\rho_t(\mathbb{Z} + y) \ge e^{-\pi/4}$, additive accuracy $e^{-\pi/4}\xi/(32b)$ gives relative error at most $\xi/(32b)$ per factor. The ratio of the two b-fold products then has relative error at most $\xi/4$. The exact ratio lies in $[0, 1]$, so its additive error is also at most $\xi/4$. For (4.5), each Fourier factor has absolute value below $1 + 2\sum_{k\ge1} e^{-\pi k^2} < 2$. Computing each factor to error $\xi/(d3^d)$ gives product error at most ξ by telescoping. This requires only $O(d + \ln(d/\xi))$ accuracy bits per factor. When $T_v \ge 9/10$, evaluation of $1/(2T_v)$ loses only a constant factor in accuracy.

Set error $\xi/2$ each to evaluation and rounding the acceptance probability to a multiple of 2^{-k}. Clip to $[0,1]$ and use k random bits. Each decision can be coupled to its exact version with error at most ξ, uniformly conditional on previous calls. The coordinate samplers in lemma 2.7 are handled with the same per-call error setting. A union bound gives total error $Q\xi$. Real widths may be replaced by rational squared widths of sufficient precision: after truncation, their exponents have polynomial sensitivity in our parameter ranges. The strict parameter inequalities permit this rounding. □

B Proofs of the parameter bounds

We first check the warm-up schedule, then give a choice of parameters for both random-superlattice schedules.

B.1 Proof of lemma 3.6

Proof. Put $\ell = \ln\ln n$, $a_0 = \ln 2/2$, and $u_i = \ln(q/s_{i-1}) = u_1 - a_0(i-1)$, where $u_1 = (3/2+o(1))\ln n$. While $s_{i-1} \le q/(2\ln n)$,

$$\ln p_i \le u_i + \tfrac{1}{2}\ln\ln(b_i+2) + O(1) \le u_i + \ell, \qquad b_i \ln p_i \le H/2.$$

To show that the schedule finishes in this range, temporarily omit the cap on the final block. Integral comparison gives

$$\sum_{i\colon s_{i-1}\le q/(2\ln n)} \left\lfloor \frac{H}{2(u_i+\ell)} \right\rfloor = \frac{H}{2a_0}\ln\frac{u_1+\ell}{2\ell+O(1)} + O(H/\ell + \ln n) = \left(\frac{4}{\ln 2} + o(1)\right) n > n.$$

Hence $r = O(\ln n)$ and $s_r \le q/\ln n$. The same comparison restricted to input widths at most n gives only $O(H) = o(n)$ coordinates, so $s_r \ge n$.

For target smoothing, put $d_i = 3n + n_i$ and $v_i = s_{i-1}/\sqrt{2}$. By lemma 2.11,

$$\mathbb{E}_B \rho_{1/v_i}(\Lambda_i^* \setminus \{0\}) \le \rho_{1/v_i}(\mathbb{Z}^{d_i}) - 1 + q^{-3n}(1+q/v_i)^{d_i}. \tag{B.1}$$

We need $v_i \ge 4q^{n_i/d_i}$. For $v_i \ge 8\sqrt{n}$, this follows from $n_i/d_i \le 1/4$ and $q^{1/4} \le 2^{1/4}\sqrt{n}$. Otherwise, only $O(1+\ln\ell)$ rounds have occurred since $v_1 = (C/\sqrt{2})\sqrt{4n/\ell}$. Each has block size $O(H/\ln n)$, giving

$$(n_i/d_i)\ln q = O((1+\ln\ell)/\ell) = o(1).$$

Since $v_i \to \infty$, the bound holds in these rounds as well. The noninteger term in (B.1) is therefore at most

$$q^{-3n}(2q/v_i)^{d_i} = \left(2q^{n_i/d_i}/v_i\right)^{d_i} \le 2^{-d_i}.$$

Since $d_i \ge 3n$, $H = o(n)$, and $r = O(\ln n)$, Markov's inequality and a union bound make all noninteger masses at most e^{-40H} except with probability $e^{-\Omega(n)}$. Finally, $v_i^2 \ge C^2H/2$, so lemma 2.10 bounds the integer mass by $O(me^{-\pi C^2H/2}) \le e^{-100H}$ for large enough C. The total is at most e^{-30H}, as required. □

B.2 Proof of lemma 4.6

Proof. The following parameter setting also applies to lemma 4.8. Write $L_0 = \ln(q/s_0)$, $L_F = \ln F$, and $L_i = L_0 - ia_0$. Choose a prime $\ell \in [n^D, 2n^D]$ with fixed $D > 4\kappa + 20$. For sufficiently large absolute constants C_4, C_5, put

$$D_i = L_{i-1} + C_4 \ln(L_{i-1} + 2) + C_5, \qquad H = \left\lceil 4 \ln \ell + \frac{n + 4r + 8}{\sum_{i=1}^{r} D_i^{-1}} \right\rceil.$$

Start with $b_i = \lfloor (H - 4\ln \ell)/D_i \rfloor$ and remove coordinates until $\sum_i b_i = n$. The initial total is between $n + 3r + 8$ and $n + 4r + 9 + \sum_i D_i^{-1}$, requiring only $O(r + \ln \ln n)$ removals. Each initial block has size $\Omega(H/\ln n)$, so even concentrating the removals in one block gives, uniformly in i,

$$b_i = (1 - o(1))H/D_i, \qquad b_i D_i \le H - 4\ln \ell.$$

Choose

$$k_i = \left\lceil \frac{b_i}{\ln \ell} \left[L_{i-1} + \ln C_2 + \frac{1}{2} \ln \left(1 + \frac{100H}{b_i} \right) \right] \right\rceil, \tag{B.2}$$

where C_2 is from lemma 4.1. Since $H/b_i = O(D_i)$, the logarithmic correction in D_i bounds the last two terms, giving

$$k_i \ln \ell \le b_i D_i + \ln \ell < H, \qquad C_2 q \ell^{-k_i/b_i} \sqrt{1 + 100H/b_i} \le s_{i-1}.$$

These are the index and auxiliary-smoothing requirements.

For both schedules, $L_F \to \infty$ and $L_0 = O(\ln n)$. Integral comparison gives

$$\sum_{i=1}^{r} D_i^{-1} = \frac{1 + o(1)}{a_0} \ln(L_0/L_F), \qquad H = \frac{(a_0 + o(1))n}{\ln(L_0/L_F)}.$$

Hence $b_{\min} = \Omega(H/\ln n)$, $b_{\max} = O(H/L_F) = o(H)$, and $b_{\min}^2/H \to \infty$. Here $L_0 = g/2$ and $L_F = \ln(C_3 \sigma \sqrt{\ln R})$, so $\ln(L_0/L_F) = (1 + o(1)) \ln R$: after division by $\ln(2 + \sigma)$, the extra term in L_F changes this logarithm by at most $O(1 + \ln \ln \ln R) = o(\ln R)$.

We check the remaining hypotheses of theorem 4.5. Equation (B.2) gives $e^{L_F} \le \ell^{k_i/b_i} \le n^{\kappa + o(1)}$, $k_i/b_i < 1/4$, and $\ell/(\sqrt{b_i}\ell^{k_i/b_i}) \to \infty$. Thus lemma 4.1 applies, with total failure probability at most $r2^{-b_{\min}} = o(1)$. For the targets, put $v_i = s_{i-1}/\sqrt{2}$ and $n_i = \sum_{t \le i} b_t$. By lemma 2.11,

$$\mathbb{E}\delta_{v_i}(\Lambda_i) \le q^{-h}(1 + q/v_i)^{h + n_i} \le \exp\{-mg/2 + O(m)\}.$$

Since $mg/H \to \infty$, Markov's inequality and a union bound make all these masses at most e^{-100H} with probability $1 - o(1)$. Also $s_0 \ge \sqrt{q}$, so lemma 2.10 bounds the integer-dual mass by $1/10$.

Every shift is within $q\sqrt{b_i}/2$ of S_i. Since $2^{\sqrt{b_{\min}}}$ dominates every polynomial in n and $b_{\min}^2 \gg H$, theorem 4.2 gives error below e^{-100H} and cost $e^{o(H)}$ per sample. Use section A.5 for numerical evaluations. A certified lower bound for r loses at most one round, changing H by $o(H)$ and giving $q/(2F) < s_r \le q/F$. The $O(r)$ surplus coordinates and $4\ln \ell$ reserve allow integer rounding, with an additional $O(\ln n)$ budget if needed. □

B.3 Proof of lemma 4.8

Proof. Now $s_0 = (\ln n)^2$, $F = C_3\sigma$, and $\ln(L_0/L_F) \to \ln(\kappa/\beta) > 0$. The same allocation gives

$$H = \left(\frac{a_0}{\ln(\kappa/\beta)} + o(1)\right) n, \qquad b_i = \Theta(n/\ln n).$$

The auxiliary smoothing, shifted sampling, and rounding arguments are unchanged. We only need a new target-mass bound for the smaller initial width.

Since $b_t \le H/L_{t-1}$, integral comparison and $\ln x \le x - 1$ give

$$n_i \le \frac{H}{a_0} \ln \frac{L_0}{L_i}, \qquad n_i L_i \le \frac{H}{a_0}(L_0 - L_i).$$

At $v_i = s_i/2$, we have $\ln(1 + q/v_i) \le L_i + O(1)$, and hence

$$\begin{aligned} h \ln q - (h + n_i)\ln(1 + q/v_i) &\ge h \ln s_0 + (h - H/a_0)(L_0 - L_i) - O(m) \\ &\ge h \ln s_0 - O(m) = \Omega(n \ln\ln n). \end{aligned}$$

The second inequality uses $c - 1 > 1/\ln(\kappa/\beta)$, which gives $h > H/a_0$. By lemma 2.11, the expected noninteger mass is at most $e^{-\Omega(n \ln\ln n)}$. Markov's inequality and a union bound make every such mass at most e^{-100H} with probability $1 - o(1)$. Since $v_i \ge (\ln n)^2/\sqrt{2}$, the integer mass is below $1/10$ as well. □

Proof of theorem 4.9. Lemma 4.8 and theorem 4.5 give $N = \lceil e^H \rceil$ jointly $o(1)$-close Gaussian samples at cost $e^{H+o(H)}$. Since $q/s_r \in [C_3\sigma, 2C_3\sigma)$, choosing C_3 sufficiently large gives $m\sigma^2 s_r^2/q^2 \le m/C_3^2 \le H/100$. Moreover,

$$\ln\big(q^{n-m}(1 + q/s_r)^m\big) = -\big((c-1)\kappa - c\beta + o(1)\big) n \ln n \le -H.$$

Indeed, $\ln x < x - 1$ at $x = \kappa/\beta > 1$ gives $1 + 1/\ln(\kappa/\beta) > \kappa/(\kappa - \beta)$, so the coefficient is positive. Finally, $s_r = n^{\kappa-\beta+o(1)}$, $\sigma\sqrt{\ln n} = o(q)$, and $\ln N = \Theta(n)$ verify the remaining conditions of lemma 2.14. Recovery costs $e^{H+o(H)}$; (4.10) and $a_0 = (\ln 2)/2$ give the claimed exponent. □