

Discrete Gaussian Sampling Meets BDGL Decoding: Solving the Shortest Vector Problem in $2^{0.5596n+o(n)}$ Time

Yiming Gao¹ Yansong Feng² Honggang Hu^{1,3}

¹School of Cyber Science and Technology, University of Science and Technology of China
qw1234567@mail.ustc.edu.cn

²Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China
fengyansong@amss.ac.cn

³Hefei National Laboratory, Hefei, China
hghu2005@ustc.edu.cn

Abstract

Aggarwal, Dadush, Regev, and Stephens-Davidowitz (ADRS) gave a $2^{n+o(n)}$ time algorithm for the Shortest Vector Problem (SVP) based on discrete Gaussian sampling (DGS), together with an honest sampler producing $2^{n/2}$ samples above the smoothing parameter in $2^{n/2+o(n)}$ time and space. Gao, Feng, and Hu (GFH) subsequently introduced DGS on random prime-index superlattices, making this sampler available at the shortest vector scale and obtaining a $2^{0.7314n+o(n)}$ time algorithm. In a different direction, the Becker–Ducas–Gama–Laarhoven (BDGL) sieve uses spherical product codes to find correlated pairs and runs in $2^{0.2925n+o(n)}$ time under the random list heuristic.

We combine the random superlattice DGS framework with a single BDGL product code decoding layer. The algorithm splits the DGS output into two lists. For a fixed shortest vector v , the Gaussian midpoint identity turns the event $X - Y = v$ into a birthday event, while equal quotient labels certify that the reported difference belongs to the input lattice. The product code decoder locates the corresponding pair without enumerating all pairwise differences.

Our analysis makes no random list assumption. For a fixed shortest vector v , once the retained lists contain a pair x, y with $x - y = v$, the BDGL product code finds that pair with high probability. We extend the product code analysis so that this guarantee is compatible with the claimed time and space bounds. A centered quotient line gives a $2^{0.5822n+o(n)}$ time algorithm. We then replace the line through the zero residue with a random affine translate. This lets us target a rarer midpoint shell. As a result, we obtain a randomized classical algorithm for SVP that runs in $2^{0.5596n+o(n)}$ time and uses $2^{n/2+o(n)}$ space.

Contents

1	Introduction	3
1.1	DGS meets one-shot BDGL decoding	3
1.2	Main results	4
2	Technique Overview	5
2.1	From direct hits to pair hits	5
2.2	A birthday pair from superlattice DGS	6
2.3	Labeled BDGL decoding	6
2.4	The centered cost balance	7
2.5	Sampling from a random shift of the superlattice	9
3	Preliminaries	10
3.1	Lattices, duals, and Gaussian mass	10
3.2	Honest discrete Gaussian sampling	11
3.3	Gaussian radial rates	12
3.4	Uniform consequences of the KL bound	14
3.5	The GFH random superlattice framework	14
4	The BDGL Product Code Decoder	15
4.1	Caps and wedges	16
4.2	Random product codes	16
4.3	The labeled near pair algorithm	18
5	The Centered Line Warmup: 0.5822	19
5.1	The centered quotient slice	20
5.2	A fixed shortest difference	21
5.3	The birthday lemma	22
5.4	Decoding the Centered Line List	22
5.5	Algorithm and proof of the centered theorem	25
6	The Affine Refinement: 0.5596	27
6.1	A random affine quotient line	27
6.2	Honest sampling from the affine lattice	28
6.3	A concentrated midpoint annulus	29
6.4	Birthday and decoding on the affine slice	31
6.5	Algorithm and main theorem	34
A	Centered Target Concentration	38
B	Finite Width Affine Annulus	39
C	A Uniform Reported Pair Bound	40
D	Product Code Coverage and Traversal	43
D.1	Uniform saturated coverage	43
D.2	Restricted Decoding with Bounded Space	45

1 Introduction

Let $\mathcal{L} \subset \mathbb{R}^n$ be a full-rank lattice. Given a basis of $\mathcal{L}$, the Shortest Vector Problem (SVP) asks us to find a nonzero vector $v \in \mathcal{L}$ of minimum Euclidean norm:

$$\|v\| = \lambda_1(\mathcal{L}) := \min_{x \in \mathcal{L} \setminus \{0\}} \|x\|.$$

SVP is a standard benchmark for exponential time lattice algorithms and a basic primitive in lattice reduction and cryptanalysis. The exact version also connects two algorithmic viewpoints. The first uses rigorous algorithms with worst-case guarantees. The second uses sieving algorithms whose best running times are predicted by random-list heuristics.

On the provable side, Ajtai, Kumar, and Sivakumar initiated randomized sieving with a $2^{O(n)}$ algorithm [AKS01]. Subsequent work developed both practical and rigorous variants [NV08, PS09]. Micciancio and Voulgaris later gave a deterministic $2^{2n+o(n)}$ -time algorithm based on Voronoi cells [MV13]. Aggarwal, Dadush, Regev, and Stephens-Davidowitz (ADRS) reached $2^{n+o(n)}$ time and space by reducing SVP to centered discrete Gaussian sampling [ADRS15].

The heuristic picture is much more encouraging. Under the random-list assumptions commonly used in lattice sieving, the Becker–Ducas–Gama–Laarhoven (BDGL) sieve runs in time

$$(3/2)^{n/2+o(n)} = 2^{0.2925n+o(n)}$$

by using spherical product codes to find correlated pairs [BDGL16]. The large gap between this heuristic bound and the rigorous bound naturally raises the following question: can we combine this pair-finding method with a sampler whose input distribution is rigorously controlled?

Two ingredients make this question approachable. ADRS gave an honest discrete Gaussian sampler [ADRS15]. Above the smoothing parameter, it produces $2^{n/2}$ independent samples in $2^{n/2+o(n)}$ time and space. Gao, Feng, and Hu (GFH) then made this sampler useful at the shortest-vector scale by sampling from a random prime-index superlattice $\Gamma \supset \mathcal{L}$ [GFH26]. At this scale, the superlattice Γ is smooth, even if $\mathcal{L}$ is not. Honest DGS is run on Γ , and each sample is tested for membership in $\mathcal{L}$. This gives an algorithm using $2^{0.7314n+o(n)}$ time and $2^{n/2+o(n)}$ space. Each sampled vector that falls in $\mathcal{L}$ is a direct chance to recover a shortest vector.

Our starting point is to use the samples in pairs. The GFH algorithm looks for a direct hit: it samples $X \sim D_{\Gamma,t}$ until X is a shortest vector of $\mathcal{L}$. Thus, a list of N samples gives N chances. In contrast, the same list gives $N(N-1)$ ordered differences. This suggests replacing

$$\text{direct hit: } X = v \quad \text{with} \quad \text{pair hit: } X - Y = v.$$

1.1 DGS meets one-shot BDGL decoding

For $X \in \Gamma$, denote $X + \mathcal{L} \in \Gamma/\mathcal{L}$ as $[X]$, i.e., the residue class of X modulo $\mathcal{L}$. This quotient label provides a lattice certificate for each pair. If $X, Y \in \Gamma$ have the same residue, then

$$[X] = [Y] \implies X - Y \in \mathcal{L}.$$

Thus the difference of equal label endpoints is in $\mathcal{L}$. For a fixed shortest vector v , completing the square gives

$$\rho_t(x)\rho_t(x-v) = e^{-\pi\|v\|^2/(2t^2)}\rho_{t/\sqrt{2}}(x-v/2).$$

At the critical scale of the GFH superlattice, the identity turns the appearance of a fixed shortest difference into a birthday-type event. The target pair appears after $2^{0.5578n+o(n)}$ samples: the rare event cost is paid over all pairs, while each endpoint is generated only once.

The birthday argument produces a target pair but does not locate it. Checking all pairs would take quadratic time and remove the gain. Instead, we use the BDGL product code decoder on a thin shell of the DGS list. Spherical filters find pairs with the required angular correlation, while quotient labels discard geometric pairs whose difference is not in $\mathcal{L}$. At a high level, the algorithm is

$$\underbrace{\text{superlattice DGS}}_{\text{proved input distribution}} + \underbrace{\text{one BDGL decoding layer}}_{\text{fast pair search}}.$$

The DGS analysis supplies the endpoint distribution, radial concentration, and target pair correlation. The product code analysis supplies the cap and wedge probabilities needed to search that distribution. Equal labels make every reported difference a lattice vector, so the algorithm can return the shortest nonzero difference it reports.

1.2 Main results

We now state two results. Define K_p as $p^{-1}\mathcal{L}$. The GFH superlattice determines a one-dimensional subspace $U = \Gamma/\mathcal{L}$ of the quotient space $K_p/\mathcal{L} \simeq \mathbb{F}_p^n$. We call U a quotient line. Our first result uses this line directly, which is a useful warm up because it isolates the decoding bottleneck. The second result uses a random affine translate of the same line, which separates the midpoint location from the target difference and improves the exponent.

As in ADRS and Gao–Feng–Hu [ADRS15, GFH26], the complexity bounds below suppress factors polynomial in the input encoding length. We use the standard real RAM model for spherical operations.

Theorem 1.1 (Centered one-shot Gaussian sieve). *There is a classical randomized algorithm that, for every sufficiently large n and every full-rank rational lattice $\mathcal{L} \subset \mathbb{R}^n$, returns a shortest nonzero vector of $\mathcal{L}$ with probability $1 - 2^{-\Omega(n \log n)}$. It uses*

$$2^{K_0 n + o(n)} \text{ time} \quad \text{and} \quad 2^{n/2 + o(n)} \text{ space.}$$

Put $\beta = 2^{0.401}$ and $a = e/\beta^2$. The centered exponent is

$$K_0 = \frac{1}{2} \log_2 \left(\frac{4a}{4a/\sqrt{e} - 1} \right) = 0.582151621 \dots < 0.5822.$$

The theorem separates two costs that will recur throughout the paper. The target pair appears in a $2^{0.5578n + o(n)}$ sample list, but finding it in the list costs more. After restricting to a norm shell, the dominant decoding costs are point and filter incidences and reported false pairs. Balancing them with the cap threshold gives the 0.5822 exponent.

The centered construction has one structural restriction: its quotient line passes through the zero residue. We remove that restriction by choosing a uniformly random affine line $z + U$ in $K_p/\mathcal{L} \simeq \mathbb{F}_p^n$. The direction of the line still controls which differences are available, while the offset controls where their midpoints lie. This allows the algorithm to target a midpoint shell slightly below the typical Gaussian radius. The shell is rarer, but it gives a smaller retained list and cheaper geometric decoding.

Theorem 1.2 (Affine one-shot Gaussian sieve). *There is a classical randomized algorithm that, for every sufficiently large n and every full-rank rational lattice $\mathcal{L} \subset \mathbb{R}^n$, returns a shortest nonzero vector of $\mathcal{L}$ with probability $1 - 2^{-\Omega(n \log n)}$. It uses*

$$2^{0.5596n + o(n)} \text{ time} \quad \text{and} \quad 2^{n/2 + o(n)} \text{ space.}$$

Optimizing the sampling and decoding costs gives

$$K_{\text{aff}} = 0.559568513615\dots < 0.5596.$$

The affine shell changes both costs. Its rare event requires more DGS samples, but the smaller endpoint shell reduces the retained list, point and filter incidences, and false pair work. Balancing these terms gives the 0.5596 exponent.

The main new ingredient is a concentration bound for the random affine line, which shows that a typical offset contains the desired midpoint mass. An honest shifted DGS then samples from this offset without bias. Section 2 gives the proof outline and the full derivation of both exponents.

Organization. Section 2 develops the common picture: a Gaussian midpoint identity creates the target pair, and a labeled BDGL decoding layer finds it. Section 3 introduces the Gaussian, lattice, and superlattice tools. Section 4 describes the product code decoder. Section 5 analyzes the centered warmup, and Section 6 proves the affine refinement. The appendices provide the finite width estimates, the uniform false pair bound, and the product-code details used in the main text.

Concurrent work. Concurrent with this work, Hhan obtained a $2^{0.6039n+o(n)}$ classical algorithm using midpoint Hessians, bounded distance decoding, and importance sampling [Hha26b].

Kim subsequently obtained an arbitrary parameter discrete Gaussian sampler in $2^{n/2+o(n)}$ time using random superlattices [Kim26]. This gives a $2^{0.7315n+o(n)}$ SVP algorithm close in spirit to the earlier $2^{0.7314n+o(n)}$ direct hit framework of Gao–Feng–Hu.

More recently, Hhan obtained a randomized algorithm for exact SVP using $2^{n/2+o(n)}$ time and space via a q -ary coset difference tree [Hha26a]. This improves the running-time exponent proved here from 0.5596 to $1/2$, but follows a different route. Hhan exploits the fact that the periodic-Gaussian gradient at v/q is nearly proportional to a shortest vector v , and computes an aggregated coset gradient along a chain of intermediate lattices through a Wagner-style generalized-birthday procedure. We instead draw honest discrete Gaussian samples from a random prime-index superlattice, certify candidate pairs by their quotient labels, and use one BDGL product-code decoding layer to locate a pair whose difference is a shortest vector.

2 Technique Overview

Fix a shortest vector v . The proof has two tasks:

1. Show that two DGS lists contain a pair x, y such that $x - y = v$.
2. Find this pair without enumerating all N^2 possible differences.

The first task uses a midpoint calculation and a birthday argument. The second uses one labeled BDGL decoding layer.

2.1 From direct hits to pair hits

For $t > 0$, write

$$\rho_t(x) = e^{-\pi\|x\|^2/t^2}.$$

Fix a shortest vector $v \in \mathcal{L}$ and let $\lambda = \|v\|$. To analyze the event $X - Y = v$, consider a coset C satisfying $C - v = C$. Let X, Y be independent samples from C . Completing the square gives

$$\rho_t(x)\rho_t(x - v) = e^{-\pi\lambda^2/(2t^2)}\rho_{t/\sqrt{2}}(x - v/2).$$

Summing this identity over $x \in C$ gives

$$\Pr[X - Y = v] = e^{-\pi\lambda^2/(2t^2)} \frac{\rho_{t/\sqrt{2}}(C - v/2)}{\rho_t(C)^2}.$$

This formula shows the three quantities that matter for a target pair: the cost of fixing the difference v , the Gaussian mass available to the midpoint, and the normalization of the two endpoint distributions. It also gives the midpoint a useful meaning. Once we condition on $X - Y = v$, the midpoint $M = (X + Y)/2$ is a discrete Gaussian with the smaller parameter $\sigma = t/\sqrt{2}$ on $C - v/2$.

This turns a direct hit search into a pair search. A direct hit algorithm gets one chance from each sample. Two lists of size N give N^2 cross pairs, so an event of probability p can appear once N is about $p^{-1/2}$. Both the centered and affine constructions use the same identity. They differ only in how they obtain enough Gaussian mass near the midpoint.

2.2 A birthday pair from superlattice DGS

The fast ADRS sampler works above the smoothing parameter. The input lattice, however, need not be smooth at the scale where a shortest difference is likely to appear. GFH fixes this by sampling from a random prime-index superlattice $\Gamma \supset \mathcal{L}$ [GFH26]. This superlattice is smooth at the required scale, and the quotient $\Gamma/\mathcal{L}$ gives each sample an exact residue label. Two samples with the same label have a difference in $\mathcal{L}$.

Let $\beta = 2^{0.401}$ be the constant from the Kabatiansky–Levenshtein bound, and put

$$a = \frac{e}{\beta^2}, \quad D = \frac{1}{a \ln 2}.$$

The corresponding critical scale satisfies

$$s_0^2 = \frac{2\pi a}{n} \lambda^2.$$

The algorithm runs over a polynomial-size grid of scales, one of which satisfies $s = (1 + O(1/n))s_0$. For this choice of s , the endpoint parameter is $t = (1 + o(1))\sqrt{2}s$. Hence the midpoint parameter is $\sigma = t/\sqrt{2} = (1 + o(1))s$. At this scale, a typical target midpoint has squared norm close to $a\lambda^2$.

The GFH mass estimates and the midpoint identity give

$$\Pr[X - Y = v] = 2^{-(1+D/8)n+o(n)}.$$

Thus two lists of size

$$N = 2^{\nu_0 n + o(n)}, \quad \nu_0 = \frac{1}{2} + \frac{D}{16} = 0.557834386 \dots$$

contain a target pair with constant probability. The honest sampler produces these samples in independent blocks.

2.3 Labeled BDGL decoding

The birthday argument shows that a target pair exists but does not locate it. Checking all cross pairs would take N^2 time and remove the gain. We use the BDGL spherical product code to find the pair.

The analysis has three costs: generating enough DGS samples, matching retained points with filters, and processing extra equal label pairs that share a filter. Restricting the samples to a shell

reduces the last two costs. It also makes the correlation depend only on the distance, up to $o(1)$. The cap threshold balances target coverage against this work. Narrow caps accept fewer points and pairs but need more filters. Wide caps need fewer filters but create more matches and reports.

Conditioned on $X - Y = v$, the midpoint has squared norm close to $a\lambda^2$ and is nearly orthogonal to v . Since $X = M + v/2$ and $Y = M - v/2$, both endpoints have squared norm close to

$$y\lambda^2, \quad y = a + \frac{1}{4}.$$

Their normalized correlation is therefore

$$\alpha_t = 1 - \frac{2}{b} + o(1), \quad b = 4y = 4a + 1.$$

Before decoding, we keep only points in a thin shell around this radius. Two points in the shell have almost the same norm, so their correlation is determined, up to $o(1)$, by their squared distance. The shell keeps the target pair with high probability and makes the lists smaller.

The BDGL decoder places directions in spherical cap filters. A cap with center $c \in S^{n-1}$ and threshold τ accepts a direction z when $\langle c, z \rangle \geq \tau$. We add one exact quotient label to each vector and index every bucket by both its filter and its label. Hence

$$\text{lab}(x) = \text{lab}(y) \iff x - y \in \mathcal{L}.$$

The geometric search uses only directions. The quotient labels enforce the lattice condition. Thus, every reported nonzero difference lies in $\mathcal{L}$.

The product code stores the filter family in an implicit form and lists the filters that accept a given direction. For the target correlation, write

$$q = b(1 - \tau^2), \quad P = \frac{1}{2} \log_2 \frac{b}{q}, \quad K_t = \frac{1}{2} \log_2 \frac{b-1}{q-1}.$$

A fixed endpoint enters a random cap with probability $2^{-Pn+o(n)}$, while both endpoints of the target pair enter one with probability $2^{-K_t n+o(n)}$. Thus P is the point and filter cost, and K_t is the number of filters needed to cover the target. Two retained points at squared distance $u\lambda^2$ can share a cap only when $u < q$. The functions P and K_t are the special cases of the cap and wedge rates defined formally in Section 4.

With enough filters for the target wedge, the product code reports the target pair with high probability.

2.4 The centered cost balance

The Gaussian cost of the thin endpoint shell is

$$I_0 = J\left(\frac{y}{2a}\right), \quad J(z) = \frac{z - 1 - \ln z}{2 \ln 2}.$$

Thus, starting with $2^{\nu_0 n + o(n)}$ samples, each retained list has size

$$2^{\ell n + o(n)}, \quad \ell = \nu_0 - I_0 = 0.467954379 \dots < \frac{1}{2}.$$

The shell is needed for two reasons. Without it, the list exponent remains $\nu_0 = 0.5578 \dots > 1/2$, so the space bound fails. The endpoint norms would also vary, making the correlation depend on

both radii as well as the distance. At the parameters below, the point and filter incidence exponent would rise from 0.5822 to about 0.6721.

Choose a product code with $2^{K_t n + o(n)}$ filters, with enough sublinear slack for the fixed pair coverage guarantee. The target pair is then reported with probability $1 - \text{negl}(n)$. The point and filter incidence exponent is

$$T_{\text{point}} = \ell + K_t - P.$$

The same parameter q controls which extra pairs can be reported. For two retained points with

$$\|x - y\|^2 = u\lambda^2,$$

their correlation is

$$\alpha(u) = 1 - \frac{2u}{b} + o(1).$$

Two directions can share a cap only if $2\tau^2 < 1 + \alpha(u)$. Substituting the definitions of $\alpha(u)$ and q gives

$$u < q.$$

Thus a larger q makes target coverage easier, but also allows more differences to be reported. We choose q to balance target coverage, point and filter work, and reported pair work.

The KL bound controls how many lattice differences have squared length $u\lambda^2$. The midpoint shell and the common-cap probability contain opposite factors in $b - u$, and these factors cancel. This shell and wedge cancellation is why the false pair bound has a simple form. The remaining reported pair exponent is

$$0.401 + K_t + \frac{1}{2} \log_2 \frac{u(q - u)}{b - 1}.$$

The term $u(q - u)$ is maximized at $u = q/2$, so the total reported pair exponent is

$$T_{\text{pair}} = 0.401 - 1 + \log_2 q - \frac{1}{2} \log_2 (q - 1).$$

For the centered construction, choosing

$$q = \frac{4a}{\sqrt{e}}$$

gives $P = \ell$ and balances the two decoding terms:

$$T_{\text{point}} = T_{\text{pair}} = K_0 = \frac{1}{2} \log_2 \left(\frac{4a}{4a/\sqrt{e} - 1} \right) = 0.582151621 \dots$$

The complete leading-order balance is

$$T_{\text{sample}} = \nu_0 = 0.557834386 \dots, \quad T_{\text{point}} = K_0, \quad T_{\text{pair}} = K_0.$$

The target pair already appears at exponent 0.5578, but decoding raises the running time to 0.5822. This is the bottleneck addressed by the shifted construction.

The decoder streams its filter incidences and keeps only one bounded batching cell at a time. Since $\ell < 1/2$, the retained lists and one DGS block fit in $2^{n/2 + o(n)}$ space. Initializing the batching tree costs less than $2^{K_0 n + o(n)}$ time.

2.5 Sampling from a random shift of the superlattice

The centered construction has a clear bottleneck: the birthday step needs exponent $\nu_0 = 0.5578\dots$, but decoding raises the time exponent to $K_0 = 0.5822\dots$. The quotient line through zero restricts the midpoint to a special residue. We keep the same direction but choose a uniformly random shift $\bar{z} + U$. The direction controls which residue differences are available, while the shift controls where their midpoints lie.

Let

$$\pi : K_p \longrightarrow V = K_p / \mathcal{L}$$

be the quotient map. The centered superlattice is the preimage

$$\Gamma = \pi^{-1}(U)$$

of the line U . After choosing a coset $\bar{z} + U$ of U in V , choose $c \in K_p$ with $\pi(c) = \bar{z}$. The shifted sampling coset is

$$C = c + \Gamma = \{x \in K_p : x + \mathcal{L} \in \bar{z} + U\}.$$

For every nonnegative function f on K_p ,

$$\mathbb{E}_{U, \bar{z}} \left[\sum_{x \in c + \Gamma} f(x) \right] = p^{1-n} \sum_{x \in K_p} f(x).$$

The random shift removes the special role of the zero residue. Midpoint locations are now spread over the whole quotient, so the target difference can be paired with a lower midpoint shell.

For the chosen midpoint annulus, let F_r be its Gaussian mass in residue r and put $S_{U, \bar{z}} = \sum_{r \in \bar{z} + U} F_r$. The first moment is

$$\mathbb{E}[S_{U, \bar{z}}] = p^{1-n} \sum_r F_r.$$

The second moment controls collisions between residue classes and shows that most shifts have $S_{U, \bar{z}}$ close to its mean. It turns the quotient average into the mass of a typical lower shell, which is the main new correctness step.

We can still sample the shifted coset honestly. Add one extra coordinate and consider the lifted lattice

$$\hat{\Gamma}_c = \{(x + kc, kt) : x \in \Gamma, k \in \mathbb{Z}\}.$$

A centered DGS sample from this lattice, conditioned on height coordinate $K = 1$, has exactly the distribution $D_{c+\Gamma, t}$. Since $\Pr[K = 1]$ is constant, this conditioning increases the sampling cost by only a constant factor. Thus, the random shift changes the midpoint distribution without changing the leading sampling exponent.

Now we can use a lower midpoint shell. Require $\|M\|^2 \simeq aA\lambda^2$ with $A < 1$. Its Gaussian cost is $2^{-J(A)n+o(n)}$, so the birthday sample exponent becomes

$$\nu_A = \nu_0 + \frac{J(A)}{2}.$$

The factor $1/2$ comes from the birthday step: a shell probability paid by a pair costs only half as much in the exponent of each list. The endpoints now lie on the smaller shell

$$y_A = aA + \frac{1}{4}, \quad b_A = 4y_A = 4aA + 1.$$

This makes sampling slightly more expensive, but reduces the retained list and the geometric work.

Define

$$\ell_A = \nu_A - J\left(\frac{y_A}{2a}\right), \quad P = \frac{1}{2} \log_2 \frac{b_A}{q}, \quad K_t = \frac{1}{2} \log_2 \frac{b_A - 1}{q - 1},$$

and let the entire leading-order cost accounting be

$$\begin{aligned} T_{\text{sample}} &= \nu_A, \\ T_{\text{point}} &= \ell_A + K_t - P, \\ T_{\text{pair}} &= 0.401 - 1 + \log_2 q - \frac{1}{2} \log_2 (q - 1), \\ K(A, q) &= \max \{T_{\text{sample}}, T_{\text{point}}, T_{\text{pair}}\}. \end{aligned}$$

As in the centered construction, the decoder streams its work and uses $2^{n/2+o(n)}$ space because $\ell_A < 1/2$.

On the balanced curve $q(A) = 4a\sqrt{A/e}$, one has $\ell_A = P$ and $T_{\text{point}} = K_t = T_{\text{pair}}$. The unique crossing of the sampling and decoding curves occurs at $A_* = 0.905117379849\dots$ and $q_* = q(A_*) = 3.598600432817\dots$. At this point, $K(A_*, q_*) = 0.559568513615\dots < 0.5596$. Thus the affine shift trades a larger sampling cost for a smaller decoding cost, and the two costs meet below 0.5596.

3 Preliminaries

This section introduces the interfaces used in the rest of the paper. We first fix lattice, quotient, and Gaussian notation. We then state the sampling and radial mass estimates, followed by the KL and GFH facts that control the number of candidate differences, smoothness, and sampling cost.

All logarithms are natural unless a base is displayed. We write $[m] = \{1, \dots, m\}$ and use $o(n)$ uniformly over every fixed compact parameter range considered below.

3.1 Lattices, duals, and Gaussian mass

For a nonsingular matrix $B \in \mathbb{R}^{n \times n}$, the full-rank lattice generated by B is $\mathcal{L} = B\mathbb{Z}^n$. Its determinant and dual lattice are

$$\det \mathcal{L} = |\det B|, \quad \mathcal{L}^* = \{y \in \mathbb{R}^n : \langle x, y \rangle \in \mathbb{Z} \text{ for every } x \in \mathcal{L}\} = B^{-T}\mathbb{Z}^n.$$

For a lattice A of positive rank, its determinant is measured in $\text{span}(A)$, and

$$\lambda_1(A) = \min_{x \in A \setminus \{0\}} \|x\|.$$

We write $B_2^d(R)$ for the closed Euclidean ball of radius R in $\mathbb{R}^d$ and $V_d = \text{vol}(B_2^d(1))$.

The decoder needs an exact test for whether the difference of two sampled points lies in $\mathcal{L}$. We obtain it by recording the residue modulo $\mathcal{L}$. If p is prime, put $K_p = p^{-1}\mathcal{L}$. Every $x \in K_p$ has a unique representation $x = Bz/p$ with $z \in \mathbb{Z}^n$. Its quotient label is

$$\text{lab}_p(x) = z \bmod p \in \mathbb{F}_p^n.$$

Thus

$$\text{lab}_p(x) = \text{lab}_p(y) \iff x - y \in \mathcal{L}. \tag{1}$$

The label uses $O(n \log p)$ bits and is computed exactly from the lattice coordinates of x . Equality of labels is therefore an exact certificate, not a geometric test: it says that the difference belongs to $\mathcal{L}$. Every pair search below uses this same certificate.

The Gaussian weight below turns norm estimates into multiplicative mass estimates. This is what makes the midpoint identity and the shell estimates work together. For $s > 0$, $x \in \mathbb{R}^n$, and a countable set $S \subset \mathbb{R}^n$, define

$$\rho_s(x) = e^{-\pi\|x\|^2/s^2}, \quad \rho_s(S) = \sum_{x \in S} \rho_s(x), \quad D_{S,s}(x) = \frac{\rho_s(x)}{\rho_s(S)}.$$

Thus $D_{S,s}$ is the discrete Gaussian on S . For a full-rank lattice put $V_s(\mathcal{L}) = s^n / \det \mathcal{L}$. Poisson summation gives, for every $t \in \mathbb{R}^n$,

$$\rho_s(\mathcal{L} - t) = V_s(\mathcal{L}) \sum_{y \in \mathcal{L}^*} e^{-\pi s^2 \|y\|^2} e^{2\pi i \langle y, t \rangle}.$$

The smoothing parameter marks the range in which the Gaussian mass is nearly unchanged by shifts. This shift flatness is the property needed when we move between lattice cosets.

Definition 3.1 (Smoothing parameter). For $\varepsilon > 0$, the smoothing parameter $\eta_\varepsilon(\mathcal{L})$ is the unique positive value satisfying

$$\rho_{1/\eta_\varepsilon(\mathcal{L})}(\mathcal{L}^* \setminus \{0\}) = \varepsilon.$$

In particular, if $s \geq \eta_\varepsilon(\mathcal{L})$, then every shift satisfies

$$(1 - \varepsilon)V_s(\mathcal{L}) \leq \rho_s(\mathcal{L} - t) \leq (1 + \varepsilon)V_s(\mathcal{L}). \quad (2)$$

3.2 Honest discrete Gaussian sampling

The birthday step needs independent samples whose number is fixed before the sample values are seen. The following interface gives these properties, together with the time and space bound used later. We use the honest sampler of Aggarwal, Dadush, Regev, and Stephens-Davidowitz [ADRS15] through this interface.

Theorem 3.2. *There is an algorithm that, on input a rank- d lattice A , a parameter $s > 0$, and a confidence parameter $\kappa = \Omega(d)$, runs in*

$$2^{d/2 + \text{polylog}(\kappa) + o(d)}$$

time and space. Its output distribution is $\exp(-\Omega(\kappa))$ -close in joint statistical distance to an ideal sequence of independent samples from $D_{A,s}$ whose length is independent of the sample values. If

$$s > \sqrt{2} \eta_{1/2}(A),$$

then the ideal sequence has length exactly $\lceil 2^{d/2} \rceil$. For $\kappa = d^2$, the time and space are

$$2^{d/2 + O(d/\log d + \log^5 d)}.$$

Below the displayed threshold the ideal output may be shorter or empty.

In the main constructions, the sampler is called in independent blocks. The fixed block size lets us split the output into two lists. The statistical distance lets us first analyze ideal independent Gaussian samples and then transfer the bounds to the actual sampler.

3.3 Gaussian radial rates

Put

$$\beta = 2^{0.401}, \quad a = \frac{e}{\beta^2}, \quad D = \frac{1}{a \ln 2}, \quad \nu_0 = \frac{1}{2} + \frac{D}{16}.$$

The next function records the exponential cost of asking for a radius that is different from the typical one. The continuous Gaussian radial rate is

$$J(z) = \frac{z - 1 - \ln z}{2 \ln 2}, \quad z > 0.$$

If a centered n -dimensional Gaussian is restricted to squared radius z times its mean, a window of relative width $O(1/\log n)$ has mass $2^{-J(z)n + O(n/\log n)}$, uniformly for z in a fixed compact subset of $(0, \infty)$. The same statement holds on a lattice and every shift whenever the tilted Gaussian scale is flat. We record the precise form once, so later shell estimates can cite it instead of repeating the same calculation.

We will use the same rate on lattice cosets. The following lemma transfers the continuous shell estimate to a shifted lattice when its Gaussian mass is flat.

Lemma 3.3 (Gaussian shell transfer). *Let $\Lambda \subset \mathbb{R}^d$ be full rank, let $c \in \mathbb{R}^d$, and put $m_r = dr^2/(2\pi)$ and $\omega_d = 1/\log_2 d$. Suppose that*

$$\rho_u(\Lambda - c) = V_u(\Lambda)(1 + \xi_{u,d}), \quad |\xi_{u,d}| \leq \xi_d = o(1),$$

uniformly when u/r lies in a fixed compact interval. For z in a fixed compact subset of $(0, \infty)$ and every fixed $c_0 > 0$,

$$\sum_{\substack{x \in \Lambda - c \\ \|\|x\|^2 - zm_r\| \leq c_0 \omega_d m_r}} \rho_r(x) = V_r(\Lambda) 2^{-J(z)d + O(d\omega_d)}. \quad (3)$$

If $0 < z < 1$, the same argument gives

$$\sum_{\substack{x \in \Lambda - c \\ \|\|x\|^2 \leq zm_r}} \rho_r(x) \leq V_r(\Lambda) 2^{-J(z)d + O(\xi_d)}. \quad (4)$$

Both estimates are uniform in the displayed compact parameter ranges.

Proof. Set $r' = r\sqrt{z}$. The identity

$$\rho_r(x) = \rho_{r'}(x) \exp\left(-\frac{\pi(1 - 1/z)}{r^2} \|x\|^2\right) \quad (5)$$

shows that, on the shell, the second factor is $\exp(-d(z - 1)/2 + O(d\omega_d))$. It remains to locate the mass of $D_{\Lambda - c, r'}$. For $|\theta| = O(\omega_d)$,

$$\mathbb{E} \left[\exp\left(\frac{2\pi\theta}{r'^2} \|X\|^2\right) \right] = \frac{\rho_{r'/\sqrt{1-2\theta}}(\Lambda - c)}{\rho_{r'}(\Lambda - c)} \leq \frac{1 + \xi_d}{1 - \xi_d} (1 - 2\theta)^{-d/2}.$$

Chernoff's bound with $\theta = \pm\Theta(\omega_d)$ gives

$$\Pr[\|X\|^2 \text{ lies outside the stated shell}] \leq \exp(-\Omega(d\omega_d^2)).$$

The proof has three steps: change the Gaussian parameter, use concentration at the new parameter, and then transfer the mass back to the original one. Multiplying the change of measure factor by

$$\frac{V_{r'}(\Lambda)}{V_r(\Lambda)} = z^{d/2}$$

proves (3).

For the lower ball, $1 - 1/z < 0$, so the second factor in (5) is largest on the boundary. Bounding the entire r' -mass by $(1 + \xi_d)V_{r'}(\Lambda)$ gives (4). $\square$

We also need two simple bounds for shifted Gaussians. The first controls a linear tail, and the second compares two Gaussian parameters. They will be used for midpoint orthogonality and shell transfer, respectively.

Lemma 3.4 (Shifted slab and scale bounds). *If (2) holds for every shift of Λ at parameter r , then, for $X \sim D_{\Lambda-c,r}$, $v \neq 0$, and $R > 0$,*

$$\Pr[|\langle X, v \rangle| \geq R] \leq 2 \frac{1 + \varepsilon}{1 - \varepsilon} \exp\left(-\frac{\pi R^2}{r^2 \|v\|^2}\right). \quad (6)$$

Moreover, for every full-rank Λ , every shift c , and $r \geq s$,

$$\rho_r(\Lambda - c) \leq \left(\frac{r}{s}\right)^d \rho_s(\Lambda).$$

Proof. Completing the square gives

$$\mathbb{E}\left[e^{h\langle X, v \rangle}\right] = e^{h^2 r^2 \|v\|^2 / (4\pi)} \frac{\rho_r(\Lambda - c - h r^2 v / (2\pi))}{\rho_r(\Lambda - c)}.$$

Flatness bounds the ratio by $(1 + \varepsilon)/(1 - \varepsilon)$; optimizing Markov's inequality for the two signs proves (6). For the scale bound, shifted Poisson summation and the triangle inequality give

$$\rho_r(\Lambda - c) \leq V_r(\Lambda) \rho_{1/r}(\Lambda^*) \leq V_r(\Lambda) \rho_{1/s}(\Lambda^*) = \left(\frac{r}{s}\right)^d \rho_s(\Lambda).$$

$\square$

The final estimate in this group is a small second moment tool. It applies when one endpoint determines the other, which is exactly the situation for a fixed target difference in two independent lists.

Lemma 3.5 (Birthday bound for a fixed difference). *Let $X_1, \dots, X_m, Y_1, \dots, Y_m$ be independent with common law μ on a countable set. Let $\mathcal{R}$ be a relation such that each first coordinate determines at most one second coordinate and conversely. Put*

$$p = \Pr[(X, Y) \in \mathcal{R}], \quad r_* = \max\{\mu(z) : z \text{ occurs in } \mathcal{R}\},$$

and let Z count pairs (i, j) with $(X_i, Y_j) \in \mathcal{R}$. Then

$$\mathbb{E}[Z] = m^2 p, \quad \frac{\text{Var}[Z]}{(\mathbb{E}[Z])^2} \leq \frac{1}{m^2 p} + \frac{2r_*}{mp}. \quad (7)$$

Proof. Indicators with disjoint sample indices are independent. For a fixed first index, the probability that two indicators are simultaneously one is at most $r_* p$; the same holds for a shared second index. There are at most m^3 ordered overlaps of either kind, while the sum of the individual variances is at most $m^2 p$. Hence $\text{Var}[Z] \leq m^2 p + 2m^3 r_* p$, which is (7). $\square$

3.4 Uniform consequences of the KL bound

We use the KL sphere packing bound in two ways. A determinant bound controls Gaussian normalization, while a point count bound controls how many lattice differences can contribute to the false pair work. The quantitative margin we need was made explicit in Gao–Feng–Hu [GFH26].

Theorem 3.6. *For all sufficiently large d , the maximum sphere packing density in $\mathbb{R}^d$ satisfies*

$$\Delta_d \leq 2^{-0.599d} = 2^{-d}\beta^d.$$

Corollary 3.7. *For every sufficiently large d and every rank- d lattice A ,*

$$\det A \geq V_d \lambda_1(A)^d \beta^{-d}.$$

Proof. The radius- $\lambda_1(A)/2$ balls centered at A form a lattice packing of density $V_d(\lambda_1(A)/2)^d / \det A$. Apply Theorem 3.6 and rearrange. $\square$

The later shell sums require one dimension threshold that works throughout a fixed compact interval of normalized radii. The following quantitative form provides this uniformity.

Lemma 3.8 (Uniform KL point count [GFH26]). *For every fixed $R_0 \geq 1$, there are constants C_{R_0} and d_{R_0} such that the following holds. If $d \geq d_{R_0}$ and A is a rank- d lattice, then, simultaneously for every $1 \leq R \leq R_0$,*

$$|A \cap B_2^d(R\lambda_1(A))| \leq C_{R_0}(d+1)\beta^d R^d.$$

Equivalently, simultaneously for $1 \leq u \leq R_0^2$,

$$|A \cap B_2^d(\sqrt{u}\lambda_1(A))| \leq C_{R_0}(d+1)\beta^d u^{d/2}.$$

We also use the Gaussian mass bound of ADRS [ADRS15] in the following form uniform in the parameter. It follows by combining the preceding compact KL count with the elementary bound $|A \cap B_2^d(R\lambda_1(A))| \leq (1+2R)^d$ outside a fixed compact interval, as carried out in [GFH26].

Theorem 3.9. *There is an absolute constant $C_G > 0$ such that, for every rank- d lattice A and every $s > 0$,*

$$\rho_s(A) \leq 1 + \left(\frac{\beta^2 s^2 d}{2\pi e \lambda_1(A)^2} \right)^{d/2+1} (d+1)^{C_G}.$$

If $s \leq \sqrt{2\pi/d}\lambda_1(A)$, then also

$$\rho_s(A) \leq 1 + \exp\left(-\pi \frac{\lambda_1(A)^2}{s^2}\right) \beta^d (d+1)^{C_G}.$$

3.5 The GFH random superlattice framework

The last interface supplies the sampling space used by both constructions. At the correct scale it gives four facts: a smooth superlattice, a quotient line that labels its points, flat Gaussian mass on every shift, and a controlled search for the prime. We record the random superlattice construction of Gao–Feng–Hu [GFH26, Sections 3–5] in the form used below. Let $h = h_n$ satisfy $\log n \ll h \ll n$, and put

$$s_0^2 = \frac{2\pi e}{\beta^2 n} \lambda_1(\mathcal{L})^2.$$

An LLL reduced basis gives computable bounds $\lambda_{\text{lo}} \leq \lambda_1(\mathcal{L}) \leq \lambda_{\text{hi}}$ with $\lambda_{\text{hi}}/\lambda_{\text{lo}} = 2^{O(n)}$. Enumerate the multiplicative grid

$$\hat{\lambda}_j = \lambda_{\text{lo}}(1 + 1/n)^j, \quad s_j^2 = \frac{2\pi e}{\beta^2 n} \hat{\lambda}_j^2,$$

up to λ_{hi} . It has polynomial size, and one branch satisfies $\hat{\lambda}_j = (1 + O(1/n))\lambda_1(\mathcal{L})$. We call it the correct branch and fix it in all subsequent estimates.

Proposition 3.10 (GFH random superlattice interface [GFH26]). *The correct scale branch has $s = (1 + O(1/n))s_0$ and a prime*

$$p = \Theta\left(2^h \frac{\det \mathcal{L}}{s^n}\right). \quad (8)$$

Set $K_p = p^{-1}\mathcal{L}$. For a uniform nonzero linear functional

$$\varphi : \mathcal{L}^*/p\mathcal{L}^* \longrightarrow \mathbb{F}_p, \quad M_\varphi = \{z \in \mathcal{L}^* : \varphi(z + p\mathcal{L}^*) = 0\}, \quad \Gamma = M_\varphi^*,$$

one has

$$\mathcal{L} \subset \Gamma \subset K_p, \quad [\Gamma : \mathcal{L}] = p, \quad V_s(\Gamma) = 2^{h+O(1)}, \quad \rho_s(\mathcal{L}) \leq 2^{O(\log n)}.$$

Moreover, $U = \Gamma/\mathcal{L}$ is marginally a uniform quotient line in $K_p/\mathcal{L} \simeq \mathbb{F}_p^n$. Except with probability $2^{-2h/3+O(\log n)}$ over φ ,

$$\rho_{1/s}(\Gamma^* \setminus \{0\}) \leq \epsilon_h, \quad \epsilon_h = 2^{-h/3}. \quad (9)$$

On this event $\eta_{1/2}(\Gamma) < s$ and, uniformly over $c \in \mathbb{R}^n$,

$$\rho_u(\Gamma - c) = V_u(\Gamma)(1 + O(\epsilon_h)) \quad (u \geq s). \quad (10)$$

On the correct branch, independently of the event in (9), for every fixed $c_0 > 0$, uniformly over $c \in \mathbb{R}^n$,

$$\rho_u(K_p - c) = V_u(K_p)(1 + O(2^{-\Omega(h)})) \quad (u \geq c_0 s). \quad (11)$$

The bounded prime search fails on this branch with probability $2^{-\Omega(n \log n)}$.

The good event in (9) is used to justify all later shift-flat estimates. Conditioning on it may change the distribution of U , so for a nonnegative work variable Z we use the following comparison with the unconditioned expectation:

$$\mathbb{E}[Z \mid G] \leq \frac{\mathbb{E}[Z]}{\Pr[G]} = (1 + o(1))\mathbb{E}[Z], \quad (12)$$

where G denotes the event in (9).

4 The BDGL Product Code Decoder

This section gives the geometric search used later. The input is two lists of vectors, and each vector carries a quotient label. A cap is a filter for one direction; a wedge measures the chance that one filter accepts two directions. The decoder uses these filters to find near pairs and compares only entries with the same label.

4.1 Caps and wedges

For $z \in S^{d-1}$ and $0 < \tau < 1$, define

$$\text{Cap}_\tau(z) = \left\{ c \in S^{d-1} : \langle c, z \rangle \geq \tau \right\}.$$

The cap is the set of filters that accept the direction z . Its normalized surface measure depends only on τ ; write it as $C_d(\tau)$. For $z, z' \in S^{d-1}$ with $\langle z, z' \rangle = \alpha$, put

$$W_d(\tau, \alpha) = \mu_d(\text{Cap}_\tau(z) \cap \text{Cap}_\tau(z')).$$

Uniformly on compact subsets of the feasible parameter range,

$$C_d(\tau) = 2^{-P(\tau)d+O(\log d)}, \quad P(\tau) = -\frac{1}{2} \log_2(1 - \tau^2), \quad (13)$$

$$W_d(\tau, \alpha) = 2^{-K_\tau(\alpha)d+O(\log d)}, \quad K_\tau(\alpha) = -\frac{1}{2} \log_2 \left(1 - \frac{2\tau^2}{1 + \alpha} \right). \quad (14)$$

The wedge has positive measure precisely when $2\tau^2 < 1 + \alpha$.

We will also need a bound that stays valid when the two caps are close to being disjoint. This is the form used when we sum false candidates.

Lemma 4.1. *Let $d \geq 3$, $-1 < \alpha \leq 1$, and put*

$$\Delta(\tau, \alpha) = 1 - \frac{2\tau^2}{1 + \alpha}.$$

If $\Delta(\tau, \alpha) \leq 0$, then $W_d(\tau, \alpha) = 0$. Otherwise,

$$W_d(\tau, \alpha) \leq \Delta(\tau, \alpha)^{(d-2)/2}. \quad (15)$$

Proof. We project a common cap center onto the span of the two directions. The cap conditions force this projection into a spherical tail, whose measure gives the bound.

Let $z, z' \in S^{d-1}$ have correlation α and put $E = \text{span}\{z, z'\}$. Every c in both caps satisfies

$$\left\langle c, \frac{z + z'}{\sqrt{2(1 + \alpha)}} \right\rangle \geq \tau \sqrt{\frac{2}{1 + \alpha}},$$

and hence

$$\|\text{proj}_E(c)\|^2 \geq \frac{2\tau^2}{1 + \alpha}.$$

For uniform $c \in S^{d-1}$, the squared projection length onto E has distribution $\text{Beta}(1, (d-2)/2)$. Its upper tail at r^2 is $(1 - r^2)^{(d-2)/2}$. The two cap inequalities have an intersection of measure zero when $2\tau^2 \geq 1 + \alpha$; otherwise the projection tail gives (15). $\square$

4.2 Random product codes

Only the code is random. The input directions may be fixed and highly structured. After conditioning on the DGS lists, coverage of one target pair is a pointwise probability statement over the independent product code.

The product structure makes two tasks possible: it gives an implicit family of filters, and it lets us enumerate the filters accepted by a direction. Pad the ambient space by $O(\log d)$ zero coordinates

and write $d = mb$ with $m = \lceil \log_2 d \rceil$. For each $i \in [m]$, let C_i consist of B_{pc} independent uniform points on $m^{-1/2}S^{b-1}$. An independent Haar rotation Q gives the code

$$\mathcal{C} = Q(C_1 \times \cdots \times C_m), \quad M = |\mathcal{C}| = B_{\text{pc}}^m. \quad (16)$$

Rounding B_{pc} changes M by at most a $2^{O(\log d)}$ factor. For a Cartesian subcode

$$D = Q(S_1 \times \cdots \times S_m), \quad S_i \subseteq C_i,$$

write

$$\text{Decode}_D(z) = D \cap \text{Cap}_\tau(z).$$

We use three properties of this code. It supports output sensitive enumeration, gives the expected point and pair incidences, and covers a fixed target pair with high probability when the target wedge is large enough. The following theorem collects these properties. Parts of the statement are direct consequences of [BDGL16]; the uniform saturation and restricted subcode forms required by our algorithm are derived in Sections D.1 and D.2. Here saturation means that the product code has enough codewords in the target wedge that the chance of missing the target pair is negligible.

Theorem 4.2. *For the code in (16), the following statements hold.*

- (i) *For every Cartesian subcode $D = Q(S_1 \times \cdots \times S_m)$ and $z \in S^{d-1}$, the set $\text{Decode}_D(z)$ can be streamed in*

$$\text{poly}(d) (B_{\text{pc}} + |\text{Decode}_D(z)|)$$

time and $\text{poly}(d)B_{\text{pc}}$ auxiliary space.

- (ii) *Every codeword is marginally uniform on S^{d-1} . Hence, for fixed z, z' of correlation α ,*

$$\begin{aligned} \mathbb{E} [|\text{Decode}_\mathcal{C}(z)|] &= MC_d(\tau), \\ \mathbb{E} [|\text{Decode}_\mathcal{C}(z) \cap \text{Decode}_\mathcal{C}(z')|] &= MW_d(\tau, \alpha). \end{aligned}$$

- (iii) *Fix $\delta > 0$. Uniformly when*

$$\delta \leq \tau \leq 1 - \delta, \quad -1 + \delta \leq \alpha \leq 1 - \delta, \quad \tau(1 - \alpha) \geq \delta, \quad 1 - \frac{2\tau^2}{1 + \alpha} \geq \delta,$$

there is

$$H_d = O(\sqrt{d} \text{polylog } d)$$

such that, if $MW_d(\tau, \alpha) \geq 2^{H_d}$, then

$$\Pr[\text{Decode}_\mathcal{C}(z) \cap \text{Decode}_\mathcal{C}(z') = \emptyset] \leq \exp(-\Omega(\log^2 d)) + me^{-d^3}. \quad (17)$$

The theorem gives the three interfaces used below. Part (i) lets us enumerate accepted filters using bounded auxiliary space. Part (ii) gives the expected work for points and pairs. Part (iii) says that a fixed target pair is covered with high probability once $MW_d(\tau, \alpha)$ has subexponential slack. The next subsection combines this enumeration with recursive batching to enforce the overall space bound.

Part (ii) follows from the Haar rotation. The root code traversal underlying (i) is [BDGL16, Lemma 5.1], and Section D.2 extends it to restricted Cartesian subcodes. Finally, Section D.1 derives (iii) from the blockwise construction in [BDGL16, Theorem 5.1].

Algorithm 1: FindLabeledNearPairs($\mathcal{A}, \mathcal{B}, \tau, m, B_{\text{pc}}, S, Z_{\text{max}}$)

Input: Deduplicated labeled lists $\mathcal{A}, \mathcal{B}$; threshold τ ; product code parameters m, B_{pc} ; live incidence budget $S \geq |\mathcal{A}|$; online work budget Z_{max} .

Output: A stream, possibly with repetitions, of pairs (x, y) satisfying $\text{lab}(x) = \text{lab}(y)$, $x \neq y$, and $\langle \hat{x}, \hat{y} \rangle \geq 2\tau^2 - 1$, or abort.

```
1 Draw  $C_1, \dots, C_m$  and  $Q$ , and set  $\mathcal{C} = Q(C_1 \times \dots \times C_m)$ ;
2 Initialize a counter at zero; charge every streamed incidence, bucket access, and decoder
  initialization operation before performing it, and return abort if the next charge would
  exceed  $Z_{\text{max}}$ ;
3 Procedure Visit( $D = Q(S_1 \times \dots \times S_m)$ ):
4   Initialize an empty table  $H$  indexed by  $(c, \xi)$ ;
5   Stream  $c \in \text{Decode}_D(\hat{x})$  for  $x \in \mathcal{A}$  into  $H[c, \text{lab}(x)]$ , stopping when the table contains
     more than  $S$  records;
6   if  $H$  overflowed then
7     Split one  $S_i$  of size at least two into two nearly equal sets, obtaining  $D_0, D_1$ ;
8     discard  $H$ ; Visit( $D_0$ ); Visit( $D_1$ );
9   else
10    foreach  $y \in \mathcal{B}$  do
11      foreach  $c \in \text{Decode}_D(\hat{y})$  do
12        foreach  $x \in H[c, \text{lab}(y)]$  do
13          if  $x \neq y$  then
14            report  $(x, y)$ ;
15    discard  $H$ ;
16 Visit( $\mathcal{C}$ );
```

4.3 The labeled near pair algorithm

Let $\mathcal{A}$ and $\mathcal{B}$ be internally deduplicated lists of nonzero vectors. Each vector x carries a label $\text{lab}(x)$, while the decoder sees only its direction $\hat{x} = x / \|x\|$. The procedure first inserts the incidences from $\mathcal{A}$ into a label indexed table and then queries that table with $\mathcal{B}$. A cell is a node of the batching tree representing a Cartesian subcode. If a cell would exceed the live incidence budget, we split it along one product code coordinate and process its children.

For the realized code, let $Z_{\mathcal{A}}$ and $Z_{\mathcal{B}}$ count the work of building and querying the point-filter incidences:

$$Z_{\mathcal{A}} = \sum_{x \in \mathcal{A}} |\text{Decode}_{\mathcal{C}}(\hat{x})|, \quad Z_{\mathcal{B}} = \sum_{y \in \mathcal{B}} |\text{Decode}_{\mathcal{C}}(\hat{y})|.$$

For $c \in \mathcal{C}$ and a label ξ , let $\mathcal{A}(c, \xi)$ and $\mathcal{B}(c, \xi)$ be the corresponding cap incidences, and put

$$P_{\text{bucket}} = \sum_{c \in \mathcal{C}} \sum_{\xi} |\mathcal{A}(c, \xi)| |\mathcal{B}(c, \xi)|.$$

The quantity P_{bucket} counts the work for pairs that share a filter, including repeated reports through different filters and the bucket access for a common vector appearing on both sides. The last term in the time bound below is the cost of initializing the batching cells.

Proposition 4.3. *Algorithm 1 reports every distinct equal-label pair sharing a filter unless its online budget is exhausted. Its space usage is*

$$O(S + |\mathcal{A}| + |\mathcal{B}| + \text{poly}(d)B_{\text{pc}}), \quad (18)$$

and its running time is

$$\text{poly}(d) \left(dZ_{\mathcal{A}} + Z_{\mathcal{B}} + P_{\text{bucket}} + \left(1 + \frac{dZ_{\mathcal{A}}}{S} \right) (|\mathcal{A}| + |\mathcal{B}|)B_{\text{pc}} \right). \quad (19)$$

If $Z_{\max}$ is supplied, the algorithm returns abort before performing operation $Z_{\max} + 1$; otherwise the preceding completeness and time bounds hold unchanged.

The proof appears in [Section D.2](#). Splitting a cell changes the traversal order, but every full code filter remains in exactly one descendant cell. Thus the space bound comes from the split, while completeness comes from visiting all descendants.

Corollary 4.4. *Fix $\delta > 0$ and parameters τ, α in the parameter range of [Theorem 4.2\(iii\)](#), and let H_d be the corresponding threshold. Fix distinct $x \in \mathcal{A}$ and $y \in \mathcal{B}$ with $\text{lab}(x) = \text{lab}(y)$ and $\langle \hat{x}, \hat{y} \rangle = \alpha$. If $MW_d(\tau, \alpha) \geq 2^{H_d}$, then the code contains a common filter except with probability at most $\exp(-\Omega(\log^2 d)) + me^{-d^3}$. On every nonaborting execution, [Algorithm 1](#) then reports (x, y) .*

Proof. [Theorem 4.2\(iii\)](#) supplies a common filter with high probability. [Proposition 4.3](#) supplies the deterministic part: every such filter is visited and the pair is reported on a nonaborting execution. $\square$

5 The Centered Line Warmup: 0.5822

The centered construction is our baseline: it shows the full birthday argument for a fixed difference and the BDGL search before the affine refinement. The target difference appears at exponent $\nu_0 = 0.5578\dots$, but the decoder raises the running time to 0.5822. [Section 6](#) moves the midpoint shell inward with a random affine shift and reduces this decoding cost.

One trial has four steps. It draws the GFH quotient line, streams DGS endpoints into two lists restricted to a shell, searches the lists with one labeled product code, and keeps the shortest reported difference. The analysis answers four corresponding questions: where the samples come from, why the lists contain a prescribed shortest vector as a difference, how the decoder finds that pair, and which work account sets the final exponent.

Fix

$$\lambda = \lambda_1(\mathcal{L}).$$

The first step is to build a superlattice that is smooth at the sampling scale while keeping the quotient labels. Let

$$a_n = \left\lceil \frac{n}{\sqrt{\log_2 n}} \right\rceil, \quad \gamma_n = 1 + \frac{1}{\log_2 n}.$$

The leading sample exponent is

$$\nu_0 = \frac{1}{2} + \frac{D}{16}.$$

5.1 The centered quotient slice

Apply Proposition 3.10 with $h = a_n$. On the branch containing the critical scale

$$s_0^2 = \frac{2\pi e}{\beta^2 n} \lambda^2.$$

we have

$$s = (1 + O(1/n))s_0, \quad t = (1 + O(n^{-3}))\sqrt{2}\gamma_n s, \quad \sigma = \frac{t}{\sqrt{2}}.$$

The same proposition chooses

$$p = \Theta\left(2^{a_n} \frac{\det \mathcal{L}}{s^n}\right). \quad (20)$$

Put $K_p = p^{-1}\mathcal{L}$ and sample (Γ, U) by the pullback construction in Proposition 3.10, where $U = \Gamma/\mathcal{L}$. Marginally, U is a uniform quotient line, and

$$\mathcal{L} \subset \Gamma \subset K_p, \quad [\Gamma : \mathcal{L}] = p, \quad V_s(\Gamma) = 2^{a_n + O(1)}.$$

Write $\varepsilon_n = 2^{-a_n/3}$. Proposition 3.10 shows that (9) holds with probability $1 - 2^{-2a_n/3 + O(\log n)}$. On this event, $t > \sqrt{2}\eta_{1/2}(\Gamma)$, and the shifted flatness estimates (10) and (11) apply.

For later averages we also need the primal description of the random line.

Lemma 5.1. *For every nonnegative function $f : K_p \rightarrow \mathbb{R}_{\geq 0}$ and every $c \in \mathbb{R}^n$,*

$$\mathbb{E}_\Gamma \left[\sum_{x \in \Gamma \setminus \mathcal{L}} f(x - c) \right] = \theta_p \sum_{x \in K_p \setminus \mathcal{L}} f(x - c), \quad \theta_p = \frac{p-1}{p^n - 1}.$$

Proof. Every nonzero residue of $K_p/\mathcal{L}$ belongs to a uniform subspace of dimension one with probability $(p-1)/(p^n-1)$. Apply this identity to each summand and use Tonelli's theorem. $\square$

Lemma 5.1 is the bridge from the random quotient line to the dense ambient lattice K_p , where the flatness estimate (11) applies. We use it whenever we estimate an expectation over Γ . The identity is unconditional. For every nonnegative work variable Z , conditioning on the smoothing event gives

$$\mathbb{E}[Z \mid G] \leq \frac{\mathbb{E}[Z]}{\Pr[G]} = (1 + o(1))\mathbb{E}[Z].$$

Each DGS output is stored with its quotient label

$$\text{lab}_p(x) = x + \mathcal{L} \in K_p/\mathcal{L}.$$

We delete the zero label, namely the points $x \in \mathcal{L}$. Lemma 5.1 then averages every remaining endpoint onto the dense ambient lattice K_p . This deletion has no sampling cost and removes the forced coset that midpoint coordinates cannot control. It is what makes the false pair estimate in Section 5.4 uniform.

By Proposition 3.10 and Theorem 3.2, one DGS call produces $\lceil 2^{n/2} \rceil$ samples at parameter t in $2^{n/2 + o(n)}$ time and space. Repeating the call and retaining the first

$$N = \lceil 2^{\nu_0 n + 2a_n/3} \rceil \quad (21)$$

outputs gives the list used below.

5.2 A fixed shortest difference

We now choose the shell so that a target pair has two endpoints at nearly the same radius. Define

$$y = a + \frac{1}{4}, \quad z_0 = \frac{y}{2a}, \quad I_0 = \frac{z_0 - 1 - \ln z_0}{2 \ln 2}, \quad \ell = \nu_0 - I_0. \quad (22)$$

On the correct scale branch let $\hat{\lambda}$ be the grid value and put

$$R^2 = \frac{nt^2}{4\pi} + \frac{\hat{\lambda}^2}{4}, \quad \delta_n = \frac{1}{100 \log_2 n}, \quad \mathcal{S} = \left\{ x : \left| \|x\|^2 - R^2 \right| \leq \delta_n \hat{\lambda}^2 \right\}. \quad (23)$$

Thus $R^2 = (y + o(1))\lambda^2$.

Let X, Y be independent samples from $D_{\Gamma, t}$. For every $w \in \Gamma$, completing the square gives

$$\Pr[X - Y = w] = e^{-\pi \|w\|^2 / (2t^2)} \frac{\rho_\sigma(\Gamma - w/2)}{\rho_t(\Gamma)^2}. \quad (24)$$

Conditioned on $X - Y = w$, the midpoint $(X + Y)/2$ follows $D_{\Gamma - w/2, \sigma}$.

Lemma 5.2. *Fix a shortest vector $v \in \mathcal{L}$ and condition on (9). Then*

$$\Pr[X, Y \in \mathcal{S}, X \notin \mathcal{L} \mid X - Y = v] = 1 - o(1), \quad (25)$$

and

$$p_v := \Pr[X - Y = v, X, Y \in \mathcal{S}, X \notin \mathcal{L}] = 2^{-2\nu_0 n - a_n + O(n/\log n)}. \quad (26)$$

Proof. Let $W = (X + Y)/2$ under the conditioning $X - Y = v$. By (24),

$$W \sim D_{\Gamma - v/2, \sigma}.$$

The shifted Chernoff calculation in [Section A](#) gives

$$\|W\|^2 = \frac{n\sigma^2}{2\pi} + O(\delta_n \lambda^2), \quad \langle W, v \rangle = O(\delta_n \lambda^2)$$

except with probability $\exp(-\Omega(n/\log^2 n))$. Since $X = W + v/2$ and $Y = W - v/2$, both endpoints then lie in $\mathcal{S}$.

The conditional zero label mass is

$$\frac{\rho_\sigma(\mathcal{L} - v/2)}{\rho_\sigma(\Gamma - v/2)}.$$

The numerator is at most $(\sigma/s)^n \rho_s(\mathcal{L}) = 2^{O(n/\log n)}$ by [Lemma 3.4](#) and [Proposition 3.10](#). The denominator is

$$V_\sigma(\Gamma)(1 + o(1)) = 2^{a_n + O(n/\log n)}$$

by (10). Their ratio is $o(1)$, proving (25).

Finally,

$$e^{-\pi \lambda^2 / (2t^2)} = 2^{-Dn/8 + O(n/\log n)}, \quad \frac{V_\sigma(\Gamma)}{V_t(\Gamma)^2} = 2^{-n - a_n + O(n/\log n)}.$$

Substitute these estimates and the shifted flatness bounds into (24). Equation (25) then gives (26). $\square$

5.3 The birthday lemma

Split the N samples into two independent lists of size $\lfloor N/2 \rfloor$. Each cross pair is a target pair with probability p_v , so the expected number of target pairs is about $N^2 p_v$. The extra a_n in the choice of N makes this expectation large enough for the second moment bound below.

Lemma 5.3. *For every Γ satisfying (9), the two lists contain, with probability $1 - 2^{-\Omega(a_n)}$, points $x, y \in \mathcal{S}$ such that*

$$x - y = v, \quad \text{lab}_p(x) = \text{lab}_p(y) \neq 0.$$

Proof. Let $m = \lfloor N/2 \rfloor$ and let $\mathcal{R}$ be the target relation in (26). Its probability is p_v , and every endpoint in the relation uniquely determines the other endpoint. For every endpoint $x \in \mathcal{S}$, its DGS atom satisfies

$$\Pr[X = x] \leq 2^{-W_0 n - a_n + O(n/\log n)}, \quad W_0 = \frac{1}{2} + \frac{Dy}{4}. \quad (27)$$

Lemma 3.5, with r_* equal to the bound in (27), gives

$$\frac{\text{Var}[Z]}{(\mathbb{E}[Z])^2} \leq 2^{-a_n/3 + O(n/\log n)} + 2^{-(W_0 - \nu_0)n - 2a_n/3 + O(n/\log n)}.$$

Since $W_0 - \nu_0 = 1/(4 \ln 2)$, Chebyshev's inequality proves the claim. The relation already requires $x \notin \mathcal{L}$, while $v \in \mathcal{L}$ gives $\text{lab}_p(x) = \text{lab}_p(y) \neq 0$. $\square$

We have now shown that the two lists contain the target pair. Section 5.4 addresses the separate algorithmic task of recovering it without enumerating all pairwise differences.

5.4 Decoding the Centered Line List

The birthday lemma supplies the target pair. We now recover it with the decoder from Section 4 and bound the work needed to do so. The main issue is to search the two lists without scanning all pairwise differences.

There are three work accounts: incidences between retained points and filters, reported false pairs, and cell initialization. The first two are balanced by the cap threshold τ , through $q = b(1 - \tau^2)$. Put

$$b = 4a + 1 = 4y, \quad 1 - \tau^2 = \frac{4a}{b\sqrt{e}}, \quad q = b(1 - \tau^2) = \frac{4a}{\sqrt{e}}. \quad (28)$$

The centered time constant is

$$K_0 = \frac{1}{2} \log_2 \left(\frac{4a}{4a/\sqrt{e} - 1} \right).$$

This is the cost of decoding the list, rather than producing the target pair. We now account for each part of this cost.

5.4.1 Retention and target coverage

The shell in (23) lies below the typical radius of $D_{\Gamma,t}$. Its large deviation rate is I_0 from (22).

Lemma 5.4. *For $X \sim D_{\Gamma,t}$ on the correct branch,*

$$\mathbb{E}_\Gamma [\Pr[X \in \mathcal{S}, X \notin \mathcal{L}]] \leq 2^{-I_0 n + O(n/\log n)}.$$

Consequently either retained list has expected size at most

$$2^{\ell n + 2a_n/3 + O(n/\log n)}. \quad (29)$$

Proof. The deterministic bound $\rho_t(\Gamma) \geq V_t(\Gamma)$ and Lemma 5.1, applied to $f(x) = \rho_t(x)\mathbf{1}_{\mathcal{S}}(x)$, give

$$\mathbb{E}_{\Gamma} [\Pr[X \in \mathcal{S}, X \notin \mathcal{L}]] \leq \frac{\theta_p}{V_t(\Gamma)} \rho_t(K_p \cap \mathcal{S}).$$

The volume factors cancel up to $1 + o(1)$ because

$$\theta_p \frac{V_t(K_p)}{V_t(\Gamma)} = 1 + o(1).$$

The normalized squared shell radius is $z_0 + O(1/\log n)$ relative to the continuous Gaussian mean. Exponential tilting to $t\sqrt{z_0 + O(1/\log n)}$ and (11) yield

$$\frac{\rho_t(K_p \cap \mathcal{S})}{V_t(K_p)} \leq \exp\left(\frac{n}{2}(1 - z_0 + \ln z_0) + O(n/\log n)\right) = 2^{-I_0 n + O(n/\log n)}.$$

Multiplying by half the sample count (21) proves (29). $\square$

The shell also makes the target geometry stable. The midpoint concentration from Lemma 5.2 gives

$$\alpha_1 = \frac{\langle x_1, x_2 \rangle}{\|x_1\| \|x_2\|} = 1 - \frac{2}{b} + O(1/\log n). \quad (30)$$

The choices above satisfy the exact leading-order identities

$$P(\tau) = \ell, \quad K_{\tau} \left(1 - \frac{2}{b}\right) = K_0, \quad (31)$$

where P and K_{τ} are defined in (13) and (14).

Pad the endpoint directions to dimension $d = n + O(\log n)$ as in Section 4. Choose the product code parameters so that

$$M_f = B_{\text{pc}}^m = 2^{K_0 n + a_n/10 + O(\log n)}. \quad (32)$$

Lemma 5.5. *Suppose a retained list has size at most*

$$L_{\text{ret}} = 2^{\ell n + 5a_n/6}. \quad (33)$$

Its expected number of incidences between points and filters is at most

$$2^{K_0 n + 14a_n/15 + O(n/\log n)}. \quad (34)$$

Moreover, a fixed target pair supplied by Lemma 5.3 shares a product code filter with probability $1 - \text{negl}(n)$ and is reported on every nonaborting execution.

Proof. Theorem 4.2(ii), (31), and (32) give

$$\mathbb{E}[Z_{\text{pt}}] = L_{\text{ret}} M_f C_d(\tau) \leq 2^{K_0 n + 14a_n/15 + O(n/\log n)}.$$

For the target, (30) and the wedge estimate give

$$\log_2(M_f W_d(\tau, \alpha_1)) \geq \frac{a_n}{10} - O(n/\log n) \geq \frac{a_n}{20}.$$

This exceeds the $H_d = O(\sqrt{n} \text{polylog } n)$ threshold in Theorem 4.2. The fixed centered parameters have positive margins in every condition of that theorem, so Corollary 4.4 completes the proof. $\square$

5.4.2 Reported Pair Work

We next bound all reported pairs other than the target. Consider two retained endpoints of equal nonzero label with difference $w \in \mathcal{L} \setminus \{0\}$. Write

$$u = \frac{\|w\|^2}{\lambda^2}.$$

At the limiting shell radius, their correlation is

$$\alpha(u) = 1 - \frac{2u}{b}.$$

A common cap has positive measure only for $1 \leq u < q$.

The zero label deletion removes exactly the endpoints $x \in \mathcal{L}$. For a fixed difference w and midpoint $M = x - w/2$, these endpoints form the coset $\mathcal{L} - w/2$, which is contained in $\Gamma - w/2$ for every Γ . The remaining midpoint sum ranges over $(\Gamma \setminus \mathcal{L}) - w/2$. Lemma 5.1 averages this sum onto $(K_p \setminus \mathcal{L}) - w/2$. The denser lattice K_p is flat under shifts at the smaller Gaussian parameters used in shell transfer, so the rare midpoint shell can be estimated by changing to the corresponding narrower Gaussian.

Lemma 5.6. *Let $P_{\neq 0}$ be the number of incidences between filters and nonzero equal label pairs from the two retained lists. Conditional on (9), the expectation over the quotient line, the ideal DGS lists, and the independent product code satisfies, with the code size (32),*

$$\mathbb{E}[P_{\neq 0}] \leq 2^{K_0 n + 13a_n/30 + O(n/\log n)}.$$

Proof. For a fixed normalized squared distance u , the KL count, the two list sizes, the common volume factor, the pair penalty, and the midpoint shell contribute, in that order,

$$0.401 + \frac{1}{2} \log_2 u + 2\nu_0 - 1 - \frac{Du}{8} - J\left(\frac{b-u}{4a}\right).$$

Since $b = 4a + 1$ and $2\nu_0 = 1 + D/8$,

$$\frac{D(1-u)}{8} - J\left(\frac{b-u}{4a}\right) = \frac{1}{2} \log_2 \frac{b-u}{4a}.$$

The wedge factor is $((q-u)/(b-u))^{n/2+o(n)}$, so $b-u$ cancels and the filtered exponent becomes

$$F(u) = 0.401 + K_0 + \frac{1}{2} \log_2 \frac{u(q-u)}{4a}, \quad 1 \leq u < q.$$

It is maximized at $u = q/2$, and

$$0.401 + \frac{1}{2} \log_2 \frac{q^2}{16a} = \log_2 \beta - \log_2 \beta = 0.$$

Thus $\max_u F(u) = K_0$. Near the feasibility boundary $u = q$, the wedge probability tends to zero and the limiting approximation is not uniform. In finite dimension the boundary is $q_n = q + O(1/\log n)$. Partitioning $[1, q_n)$ into bins of width $1/n$, measured backward from q_n , gives a uniform bound for each bin. Summing the bins changes the estimate only by a subexponential factor and also covers the finite shell width. The exact conditional expectation and the envelope for unequal radii are recorded in Proposition C.1. $\square$

The choice $q = 4a/\sqrt{e}$ in (28) places the peak of the filtered exponent at K_0 . The cap threshold enters both sides of the balance: target coverage through the wedge and false pair work through q .

After deduplication within each list, a common point $x = y$ creates at most one bucket access per query incidence. Hence diagonal bucket work is covered by (34).

5.4.3 The centered work ledger

Apply Proposition 4.3 with

$$S_{\text{cell}} = 2^{n/2+a_n}. \quad (35)$$

The leading exponents are

$$\begin{aligned} T_{\text{sample}} &= \nu_0, & T_{\text{point}} &= K_0, \\ T_{\text{pair}} &= K_0. \end{aligned}$$

Cell initialization has exponent $T_{\text{point}} - 1/2 + \ell < T_{\text{point}}$ because $\ell < 1/2$. Therefore

$$\max\{T_{\text{sample}}, T_{\text{point}}, T_{\text{pair}}\} = K_0.$$

The ledger identifies the goal of the affine refinement. The sampling term $\nu_0 = 0.5578$ determines when the target pair appears, but point incidences and reported pair work determine the running time $K_0 = 0.5822$. We therefore move the midpoint shell below its typical radius to reduce the decoding terms ℓ and P . The centered line cannot move this shell independently: its zero residue couples the target direction to the midpoint location. Section 6 removes this coupling with a random affine translate and makes the extra birthday cost explicit.

5.5 Algorithm and proof of the centered theorem

We now turn the centered construction into an algorithm. One trial chooses a scale and a smooth superlattice, samples two labeled lists, keeps the points in the shell, and runs the labeled BDGL search. Equal labels make every reported difference a lattice vector; work counters enforce the time and space bounds; independent repetitions give the final success probability. The scale branches use the definitions from Section 5, and the work estimates come from Section 5.4.

Within the repetition loop, ending the current trial discards all trial-local state and continues with the next repetition. Sampler calls and retention are interleaved block by block: each returned block is consumed immediately and released before the next call, and only the prefix needed to reach N outputs is consumed from the last block. The decoder's report stream is likewise consumed online.

By (1), every nonzero update is a lattice vector. The scale grid has polynomial size, and the bounded prime search fails on the correct branch with probability $2^{-\Omega(n \log n)}$.

5.5.1 Resource bounds and operation budget

Proposition 5.7. *On the correct scale branch, conditional on (9), the ideal DGS version of one trial exceeds one of its retained list, point incidence, bucket work, or initialization caps with probability $2^{-\Omega(a_n)}$. Irrespective of the scale branch and random choices, every trial, including an aborted one, uses at most*

$$2^{K_0 n + o(n)} \text{ time} \quad \text{and} \quad 2^{n/2 + o(n)} \text{ space.}$$

Proof. Lemma 5.4 and Markov's inequality against (33) bound the retained list overflow probability by $2^{-a_n/6 + O(n/\log n)}$ before conditioning on the smoothing event; (12) changes this by only a $1 + o(1)$ factor. Conditional on the list cap, put

$$W_{\text{root}} = Z_A + Z_B + P_{\text{bucket}}$$

for the root product code. Equation (34), Lemma 5.6, and the bound of diagonal bucket accesses by query incidences give the following conditional bound (conditioning on the list-cap event costs only a $1 + o(1)$ factor):

$$\mathbb{E}[W_{\text{root}}] \leq 2^{K_0 n + 14a_n/15 + O(n/\log n)}.$$

Algorithm 2: CenteredOneShotSVP($\mathcal{L}$)

Input: A rational basis of a full-rank lattice $\mathcal{L} \subset \mathbb{R}^n$.
Output: The shortest reported nonzero vector, or $\perp$.

- 1 Set $\kappa \leftarrow n^2$, $N \leftarrow \lceil 2^{\nu_0 n + 2a_n/3} \rceil$, and $Q_{\text{rep}} \leftarrow \lceil C_Q n \log_2 n \rceil$;
- 2 Set $m_{\text{DGS}} \leftarrow \lceil 2^{n/2} \rceil$ and $R_{\text{DGS}} \leftarrow \lceil N/m_{\text{DGS}} \rceil$;
- 3 Set $L_{\text{ret}} \leftarrow 2^{\ell n + 5a_n/6}$, $S_{\text{cell}} \leftarrow 2^{n/2 + a_n}$, and $Z_{\text{max}} \leftarrow 2^{K_0 n + a_n}$;
- 4 Set $v_{\text{best}} \leftarrow \perp$;
- 5 **foreach** *scale grid branch* j **do**
- 6 Compute s_j, t_j , the shell $\mathcal{S}_j$, and a prime p_j satisfying the guarded version of (20); skip a failed branch;
- 7 **for** $r = 1, \dots, Q_{\text{rep}}$ **do**
- 8 Draw (Γ, U) by the pullback construction of Proposition 3.10;
- 9 Initialize empty deduplicated retained lists $\mathcal{A}, \mathcal{B}$;
- 10 **for** $k = 1, \dots, R_{\text{DGS}}$ **do**
- 11 Call Theorem 3.2 on (Γ, t_j, κ) ; if it returns fewer than m_{DGS} outputs, end the current trial and continue with the next repetition;
- 12 Consume as many of its first m_{DGS} outputs as needed to reach N outputs in total, assigning the first $\lfloor N/2 \rfloor$ to side $\mathcal{A}$ and the rest to side $\mathcal{B}$; immediately discard points with zero label or outside the shell and otherwise insert the point with label $\text{lab}_{p_j}(x)$, ending the current trial before an insertion would make either retained list exceed L_{ret} ;
- 13 Release the returned block;
- 14 Run Algorithm 1 on $(\mathcal{A}, \mathcal{B}, \tau, m, B_{\text{pc}}, S_{\text{cell}}, Z_{\text{max}})$, with B_{pc}^m chosen as in (32);
- 15 **foreach** *reported pair* (x, y) **do**
- 16 Set $w \leftarrow x - y$ and update v_{best} if $w \neq 0$ is shorter;
- 17 If Algorithm 1 returns abort, end the current trial and continue with the next repetition;
- 18 **return** v_{best} ;

By Proposition 4.3, there is an absolute constant $c_{\text{bat}} \geq 1$ such that all streamed-incidence and bucket-access charges in the batching traversal are at most $c_{\text{bat}} dW_{\text{root}}$. Therefore Markov's inequality gives

$$\Pr \left[c_{\text{bat}} dW_{\text{root}} > \frac{Z_{\text{max}}}{4} \right] \leq \frac{4c_{\text{bat}} d \mathbb{E}[W_{\text{root}}]}{Z_{\text{max}}} = 2^{-\Omega(a_n)},$$

because $\log d + O(n/\log n) = o(a_n)$.

On the complementary good event $c_{\text{bat}} dW_{\text{root}} \leq Z_{\text{max}}/4$, Proposition 4.3 bounds cell initialization deterministically by

$$\text{poly}(n) \left(1 + \frac{dZ_{\text{max}}}{S_{\text{cell}}} \right) 2L_{\text{ret}} B_{\text{pc}}.$$

Its two leading exponents are ℓ and $K_0 - 1/2 + \ell$, both below K_0 because $\ell < 1/2$. Thus it is at most $Z_{\text{max}}/4$ for all sufficiently large n , and the entire online counter stays below Z_{max} on the displayed event.

The resource bound itself needs no correct-branch distributional estimate. On every branch, a trial makes at most R_{DGS} sampler calls, so Theorem 3.2 bounds sampling by $2^{\nu_0 n + o(n)}$ time. The retained lists have a hard cap, and Algorithm 1 performs at most Z_{max} charged operations, returning abort before operation $Z_{\text{max}} + 1$. Its polynomial overhead and $B_{\text{pc}} = 2^{O(n/\log n)}$ are absorbed in the $o(n)$ term. At most one fitting cell, the two retained lists, the product code description, and one DGS block are live simultaneously. Equation (35) and $\ell < 1/2$ give the claimed time and space bounds, also for aborted trials. $\square$

5.5.2 Correctness and resources

Proof of Theorem 1.1. Fix a shortest vector v and the correct scale branch. Proposition 3.10 supplies the good random superlattice except with probability $2^{-\Omega(a_n)}$. Conditional on this event, Lemma 5.3 supplies retained endpoints x, y satisfying $x - y = v$ and $\text{lab}_p(x) = \text{lab}_p(y) \neq 0$, again except with probability $2^{-\Omega(a_n)}$.

The target pair is fixed before the independent product code is drawn. Lemma 5.5 and Corollary 4.4 supply a common filter with probability $1 - \text{negl}(n)$; every nonaborting traversal then reports the pair. Proposition 5.7 bounds the cap overflow probability by $o(1)$. A union bound shows that one trial on the correct branch succeeds with probability bounded below by an absolute constant. Whenever it succeeds, v is among the reported differences. All nonzero reports lie in $\mathcal{L}$, so the shortest report has norm exactly $\lambda_1(\mathcal{L})$.

The repetitions use fresh superlattices, DGS calls, and product codes. Choosing C_Q sufficiently large makes the probability that every correct branch trial fails at most $2^{-\Omega(n \log n)}$. The bounded prime search has the same failure scale.

We now transfer the ideal-sampler analysis to the actual sampler. Order all calls and condition on their preceding histories. Theorem 3.2 gives $e^{-\Omega(n^2)}$ statistical distance for the next call. There are $2^{O(n)}$ calls, so the adaptive hybrid has total distance $e^{-\Omega(n^2)}$.

The deterministic part of Proposition 5.7 applies to every scale branch, including incorrect and aborted ones. Polynomially many scale branches and repetitions are therefore absorbed by the $o(n)$ terms. This proves the theorem. $\square$

6 The Affine Refinement: 0.5596

Section 5.4 leaves a precise bottleneck. The centered trial finds the target difference after $2^{\nu_0 n + o(n)}$ samples with $\nu_0 = 0.5578\dots$, but its decoder costs $2^{K_0 n + o(n)}$ with $K_0 = 0.5822$. The quotient line through the zero residue controls both which differences are allowed and where their midpoints lie. These two roles are tied together, so the midpoint shell cannot be moved inward on its own.

We separate these roles by translating the quotient line. Choose a random affine line and sample from its lift $c + \Gamma$. Since $v \in \mathcal{L}$, translation by v preserves the affine lattice, so the birthday identity for a fixed difference is unchanged. The offset moves the midpoint mass and lets us target a lower shell, while the equal label certificate, birthday argument, and BDGL decoder stay the same as in Section 5.

The refinement has a clear tradeoff. If the midpoint squared radius is chosen to be A times its centered value, the birthday exponent rises by $J(A)/2$, while the retained list and decoding exponents fall. This section shows that a typical affine line supports the lower shell and computes the resulting work ledger.

Fix a shortest vector $v \in \mathcal{L}$ and write

$$\lambda = \|v\| = \lambda_1(\mathcal{L}), \quad \mathcal{I} = [0.85, 0.95], \quad A \in \mathcal{I}.$$

As before, $K_p = p^{-1}\mathcal{L}$, $U = \Gamma/\mathcal{L}$, and $K_p/\mathcal{L} \simeq \mathbb{F}_p^n$.

6.1 A random affine quotient line

Put $V = K_p/\mathcal{L}$. After choosing the uniform line $U \leq V$, choose a uniform coset $\bar{z} + U \in V/U$. Let $c \in K_p$ lift $\bar{z}$ and define

$$C = c + \Gamma = \{x \in K_p : x + \mathcal{L} \in \bar{z} + U\}.$$

Every sampled point carries the quotient label

$$\text{lab}_p(x) = x + \mathcal{L} \in V.$$

The line direction controls which residue differences are allowed; the affine offset controls where their midpoints occur. We need a typical offset to carry the expected midpoint mass. The following first and second moment calculation gives this concentration. Write

$$\theta = p^{1-n}.$$

Lemma 6.1. *For nonnegative weights $(F_r)_{r \in V}$, let*

$$S_{U,\bar{z}} = \sum_{r \in \bar{z} + U} F_r, \quad T = \sum_{r \in V} F_r.$$

Then

$$\mathbb{E}_{U,\bar{z}}[S_{U,\bar{z}}] = \theta T, \quad \text{Var}_{U,\bar{z}}[S_{U,\bar{z}}] \leq \theta \sum_{r \in V} F_r^2. \quad (36)$$

Equivalently, for every nonnegative $f : K_p \rightarrow \mathbb{R}_{\geq 0}$,

$$\mathbb{E}_{U,\bar{z}} \left[\sum_{x \in c + \Gamma} f(x) \right] = \theta \sum_{x \in K_p} f(x). \quad (37)$$

Proof. A fixed residue lies on the random affine line with probability θ . For distinct $r, r' \in V$,

$$\Pr[r, r' \in \bar{z} + U] = \theta \frac{p-1}{p^n-1} < \theta^2.$$

Expanding the first two moments proves (36); summing the point weights within each residue gives (37). $\square$

The mean says that a random affine line sees the right amount of midpoint mass on average. The variance bound is what lets us replace this average by a statement about a typical line.

6.2 Honest sampling from the affine lattice

The affine trial samples from $C = c + \Gamma$ rather than Γ . We therefore need an honest sampler for $D_{C,t}$ with a fixed output budget that does not bias the list. We derive the required shifted wrapper from the one-dimensional embedding of ADRS [ADRS15, Section 5.4]. The proposition below verifies the fixed output budget, the absence of conditioning bias, and the uniform transcript bound needed for exponentially many adaptive calls. Define

$$\hat{\Gamma}_c = \{(x + kc, kt) : x \in \Gamma, k \in \mathbb{Z}\} \subset \mathbb{R}^{n+1}. \quad (38)$$

Proposition 6.2. *For every rational (Γ, c, t) , there is a wrapper that makes 32 ADRS calls with confidence parameter $\kappa = n^2$, uses $2^{n/2+o(n)}$ time and space, and returns either $\perp$ or $m_n = \lceil 2^{n/2} \rceil$ points. Define its ideal transcript by replacing each ADRS call by the ideal random length transcript of Theorem 3.2. Uniformly in the input, the actual and ideal wrapper transcripts are $e^{-\Omega(n^2)}$ -close. The joint distance remains $e^{-\Omega(n^2)}$ for every adaptive sequence of $2^{O(n)}$ wrapper calls.*

If (9) holds at a parameter s with $t \geq \sqrt{2}s$, the ideal wrapper returns $\perp$ with probability $\exp(-\Omega(2^{n/2}))$. Conditional on success, its output consists of m_n independent samples from $D_{C,t}$; in particular, conditioning on having accepted m_n samples introduces no bias.

Proof. The dual of (38) is

$$\hat{\Gamma}_c^* = \left\{ \left(y, \frac{k - \langle y, c \rangle}{t} \right) : y \in \Gamma^*, k \in \mathbb{Z} \right\}.$$

Indeed, pairing with $(x, 0)$ forces $y \in \Gamma^*$, and pairing with (c, t) gives the second coordinate.

Assume now that (9) holds at a parameter s with $t \geq \sqrt{2}s$. At Gaussian parameter $\sqrt{2}/t$, the terms with $y = 0$ contribute

$$\zeta_{\text{vert}} = 2 \sum_{k \geq 1} e^{-\pi k^2/2} < \frac{1}{2}.$$

For each $y \neq 0$, the shifted theta sum in one dimension is maximized at an integer shift. Since $t \geq \sqrt{2}s$, (9) gives

$$\rho_{\sqrt{2}/t}(\hat{\Gamma}_c^* \setminus \{0\}) \leq \zeta_{\text{vert}} + \rho_{\sqrt{2}}(\mathbb{Z})\rho_{1/s}(\Gamma^* \setminus \{0\}) < \frac{1}{2}$$

for all sufficiently large n . Hence $t > \sqrt{2}\eta_{1/2}(\hat{\Gamma}_c)$.

For $(\hat{X}, Kt) \sim D_{\hat{\Gamma}_c, t}$, the weight of $(x + kc, kt)$ is $e^{-\pi k^2} \rho_t(x + kc)$. Conditional on $K = 1$, the first n coordinates therefore follow $D_{C, t}$. Moreover,

$$\Pr[K = 1] = \frac{e^{-\pi} \rho_t(\Gamma + c)}{\sum_{k \in \mathbb{Z}} e^{-\pi k^2} \rho_t(\Gamma + kc)} = (1 + O(\varepsilon_n)) \frac{e^{-\pi}}{\rho_1(\mathbb{Z})}, \quad (39)$$

where $\varepsilon_n = 2^{-an/3}$ and the last equality follows from (10). This probability is bounded below by an absolute positive constant.

The wrapper always makes 32 calls on $\hat{\Gamma}_c$, accepts samples with height coordinate $K = 1$, and keeps the first m_n accepted values, returning $\perp$ if fewer than m_n samples are accepted. Under (9), the embedded lattice is above smoothing, so every ideal call contains exactly $\lceil 2^{(n+1)/2} \rceil$ independent embedded Gaussian samples. A Chernoff bound and (39) give the stated probability that fewer than m_n samples are accepted. Conditional on the sequence of height coordinates, the accepted first coordinates are still independent with law $D_{C, t}$; consequently conditioning on having accepted m_n samples introduces no bias.

For arbitrary rational input, apply the per call transcript guarantee of Theorem 3.2 with $\kappa = n^2$ to the fixed sequence of 32 calls and then apply the deterministic acceptance map. This proves the unconditional wrapper distance. Adaptive telescoping over $2^{O(n)}$ wrapper invocations still gives $2^{O(n)} e^{-\Omega(n^2)} = e^{-\Omega(n^2)}$. $\square$

6.3 A concentrated midpoint annulus

The purpose of the affine offset is to move the midpoint shell inward without losing too much birthday mass. The parameter $A < 1$ measures this move: the target midpoint has squared radius $aA\lambda^2$ instead of the centered value $a\lambda^2$. Lowering A makes the target pair rarer, but it also shrinks the endpoint list and reduces the decoding work. We now measure both effects.

Put

$$\omega_n = \frac{1}{\log_2 n}, \quad \Delta_n = \frac{\omega_n}{8}, \quad y_A = aA + \frac{1}{4}, \quad b_A = 4y_A. \quad (40)$$

The target midpoint event and endpoint shell are

$$\begin{aligned} \mathcal{E}_A(v) &= \left\{ w : \left| \|w\|^2 - aA\lambda^2 \right| \leq \Delta_n \lambda^2, \left| \langle w, v \rangle \right| \leq \Delta_n \lambda^2 \right\}, \\ \mathcal{S}_A &= \left\{ x : \left| \|x\|^2 - y_A \lambda^2 \right| \leq \frac{\omega_n}{2} \lambda^2 \right\}. \end{aligned} \quad (41)$$

The identity

$$w \in \mathcal{E}_A(v) \implies w + v/2, w - v/2 \in \mathcal{S}_A \quad (42)$$

follows by expanding the two squared norms.

For each residue $r \in V$, let F_r be the midpoint mass carried by that residue. Formally, define

$$F_r = \sum_{\substack{q \in K_p \\ q + \mathcal{L} = r}} \rho_\sigma(q - v/2) \mathbf{1}_{\mathcal{E}_A(v)}(q - v/2),$$

and put

$$T_A = \sum_{r \in V} F_r, \quad S_{U, \bar{z}} = \sum_{r \in \bar{z} + U} F_r, \quad \mu_A = \theta T_A.$$

The first moment gives the large deviation cost $J(A)$. To show that a typical affine line has this mass, we use the second moment; its main estimate reduces to one scalar saddle. For $1 \leq u < 4aA$, set

$$H_A(u) = 0.401 + \frac{1}{2} \log_2 u - \frac{1}{2} - \frac{Du}{4} - J\left(2A - \frac{u}{2a}\right). \quad (43)$$

Lemma 6.3. *For $x = u/(2a)$,*

$$H_A(u) = -J(x) - J(2A - x) \leq -2J(A). \quad (44)$$

Equality holds only at $u = 2aA$. In particular,

$$H_A(2aA) = -2J(A).$$

Proof. Substitute $u = 2ax$, $D = 1/(a \ln 2)$, and $a = e/\beta^2$ into (43); the terms outside the last J combine to $-J(x)$. This proves (44). The function J is strictly convex, so

$$J(x) + J(2A - x) \geq 2J(A),$$

with equality only when $x = A$. Equivalently, $H_A(u) + 2J(A) = \frac{1}{2} \log_2(x(2A - x)/A^2) \leq 0$. $\square$

Proposition 6.4. *On the correct scale branch, uniformly for $A \in \mathcal{I}$,*

$$T_A = V_\sigma(K_p) 2^{-J(A)n + O(n/\log n)}, \quad (45)$$

$$\sum_{r \in V} F_r^2 \leq V_\sigma(K_p) 2^{-2J(A)n + O(n/\log n)}. \quad (46)$$

Conditional on (9), with probability $1 - 2^{-\Omega(a_n)}$ over U and $\bar{z}$,

$$S_{U, \bar{z}} = V_\sigma(\Gamma) 2^{-J(A)n + O(n/\log n)}. \quad (47)$$

Proof. The sum T_A is the ρ_σ -mass of $\mathcal{E}_A(v)$ in $K_p - v/2$. Lemma 3.3 at radial ratio $A + O(1/\log n)$ gives the factor $2^{-J(A)n + O(n/\log n)}$. Lemma 3.4 shows that the excluded midpoint slab has probability $\exp(-\Omega(n/\log^2 n))$ under the tilted law. Together with (11), this proves (45).

For the second moment, put $f(w) = \rho_\sigma(w) \mathbf{1}_{\mathcal{E}_A(v)}(w)$. Equality of residues means that two midpoint witnesses differ by a vector of $\mathcal{L}$. Hence their collision mass has the exact expansion

$$\sum_{r \in V} F_r^2 = \sum_{z \in \mathcal{L}} \sum_{w \in K_p - v/2} f(w) f(w - z), \quad (48)$$

and

$$\rho_\sigma(w)\rho_\sigma(w-z) = e^{-\pi\|z\|^2/(2\sigma^2)}\rho_{\sigma/\sqrt{2}}(w-z/2). \quad (49)$$

The term $z = 0$ has exponent $-1/2 - J(2A)$ relative to $V_\sigma(K_p)$, which is smaller than $-2J(A)$ by $(1 + \ln A)/(2 \ln 2)$. For $z \neq 0$, binning $u = \|z\|^2/\lambda^2$ and applying the KL count gives the exponent $H_A(u) + O(1/\log n)$. Lemma 6.3 proves the required maximum. The finite width radial boundary is handled in Section B.

We now pass from the ambient estimates to a typical affine line. Lemma 6.1 and $\theta V_\sigma(K_p) = V_\sigma(\Gamma)$ give

$$\frac{\text{Var}[S_{U,\bar{z}}]}{\mu_A^2} \leq \frac{2^{O(n/\log n)}}{V_\sigma(\Gamma)} = 2^{-a_n + O(n/\log n)}.$$

Let G denote (9) and $g = \Pr[G]$. For every fixed U , averaging over $\bar{z} \in V/U$ gives μ_A , independently of U . Therefore

$$\mathbb{E}[S_{U,\bar{z}} \mid G] = \mu_A, \quad \text{Var}[S_{U,\bar{z}} \mid G] \leq \frac{\theta \sum_r F_r^2}{g} + (g^{-1} - 1)\mu_A^2.$$

Since $g = 1 - 2^{-\Omega(a_n)}$, the conditional relative variance remains $2^{-\Omega(a_n)}$. Chebyshev's inequality proves (47). $\square$

We have shown that a typical affine line carries the midpoint mass in the lower shell needed by the birthday argument. The next section pays for this rarer event through the birthday sample count and analyzes the resulting decoding ledger.

6.4 Birthday and decoding on the affine slice

For a typical affine line, the remaining analysis has two stages: establish the target pair and recover it efficiently. The lower shell changes these stages in opposite ways: it raises the birthday cost but lowers the retained list and decoding costs. We track all three exponents in one ledger.

Fix $A \in \mathcal{I}$ and condition on the good superlattice event and the affine mass concentration event in (47). Recall

$$\nu_A = \nu_0 + \frac{J(A)}{2}, \quad \nu_0 = \frac{1}{2} + \frac{D}{16}.$$

The term $J(A)/2$ is the birthday cost of requiring the target midpoint to lie in the lower annulus. This is the price of the decoding savings below.

6.4.1 The affine birthday pair

Let X, Y be independent samples from $D_{C,t}$. Since $v \in \mathcal{L} \subset \Gamma$, translation by v preserves C . The target pair probability is the product of three factors: the cost of fixing the difference, the midpoint mass in the chosen annulus, and the two endpoint normalizers. Completing the square gives

$$\Pr[X - Y = v, (X + Y)/2 \in \mathcal{E}_A(v)] = e^{-\pi\lambda^2/(2t^2)} \frac{S_{U,\bar{z}}}{\rho_t(C)^2}. \quad (50)$$

Equations (47) and (10) give the three exponents

$$e^{-\pi\lambda^2/(2t^2)} = 2^{-Dn/8 + O(n/\log n)}, \quad \frac{V_\sigma(\Gamma)}{V_t(\Gamma)^2} = 2^{-n - a_n + O(n/\log n)}, \quad \frac{S_{U,\bar{z}}}{V_\sigma(\Gamma)} = 2^{-J(A)n + O(n/\log n)}.$$

Since $2\nu_A = 1 + D/8 + J(A)$, their product is

$$p_v(C) = 2^{-2\nu_A n - a_n + O(n/\log n)}.$$

The shared endpoint terms in the birthday variance are controlled by the largest DGS atom on the endpoint shell.

Lemma 6.5. *For every $x \in C \cap \mathcal{S}_A$,*

$$\Pr[X = x] \leq 2^{-W(A)n - a_n + O(n/\log n)}, \quad W(A) = \frac{1}{2} + \frac{y_A}{4a \ln 2}. \quad (51)$$

Moreover,

$$W(A) - \nu_A = \frac{1 + \ln A}{4 \ln 2} > 0.30 \quad (A \in \mathcal{I}). \quad (52)$$

Proof. Use the inner radius of (41) in the numerator and $\rho_t(C) \geq (1 - \varepsilon_n)V_t(\Gamma)$ in the denominator. This gives (51); substituting the definitions of y_A , D , and $J(A)$ gives (52). $\square$

Lemma 6.6. *Let*

$$N = \left\lceil 2^{\nu_A n + 2a_n/3} \right\rceil$$

independent samples be split evenly into two lists. With probability $1 - 2^{-\Omega(a_n)}$, the lists contain $x_1, x_2 \in \mathcal{S}_A$ such that

$$x_1 - x_2 = v, \quad \text{lab}_p(x_1) = \text{lab}_p(x_2).$$

Proof. Let $m = \lfloor N/2 \rfloor$ and use the relation in (50); each endpoint determines the other uniquely. Lemma 3.5, with $p = p_v(C)$ and r_* from (51), gives

$$\frac{\text{Var}[Z]}{(\mathbb{E}[Z])^2} \leq 2^{-a_n/3 + O(n/\log n)} + 2^{-(W(A) - \nu_A)n - 2a_n/3 + O(n/\log n)}.$$

Chebyshev's inequality proves the claim. The shell inclusion follows from (42), and $v \in \mathcal{L}$ gives equal quotient labels by (1). $\square$

6.4.2 Retention and target coverage

Moving the shell inward makes a single endpoint rarer, so the retained list is smaller. We choose the filter parameter through

$$q = b_A(1 - \tau^2), \quad P = \frac{1}{2} \log_2 \frac{b_A}{q}, \quad K_t = \frac{1}{2} \log_2 \frac{b_A - 1}{q - 1},$$

where $\max\{2, b_A/2\} < q < b_A$. Define

$$\ell_A = \nu_A - J\left(\frac{y_A}{2a}\right).$$

Lemma 6.7. *For $X \sim D_{C,t}$,*

$$\Pr[X \in \mathcal{S}_A] = 2^{-J(y_A/(2a))n + O(n/\log n)}. \quad (53)$$

A target pair from Lemma 6.6 has correlation

$$\frac{\langle x_1, x_2 \rangle}{\|x_1\| \|x_2\|} = 1 - \frac{2}{b_A} + O(1/\log n). \quad (54)$$

Consequently, a retained point has cap exponent P , while the target pair has wedge exponent K_t .

Proof. The typical squared radius under $D_{C,t}$ is $(2a + O(1/\log n))\lambda^2$. Tilting to the shell radius gives (53); the tilted parameter is at least s , so (10) applies to the shift C . For the target pair, write $x_1 = w + v/2$ and $x_2 = w - v/2$ with $w \in \mathcal{E}_A(v)$. Expanding the inner product and the two norms gives (54). Finally,

$$1 - \tau^2 = \frac{q}{b_A}, \quad 1 - \frac{2\tau^2}{1 + (1 - 2/b_A)} = \frac{q - 1}{b_A - 1},$$

which gives the cap and wedge exponents from (13)–(14). $\square$

Choose the product code size

$$M_f = B_{\text{pc}}^m = 2^{K_t n + a_n/10 + O(\log n)} \quad (55)$$

and cap either retained list at

$$L_{\text{ret}} = 2^{\ell_A n + 5a_n/6}. \quad (56)$$

Put

$$T_{\text{point}} = \ell_A + K_t - P.$$

Lemma 6.8. *Subject to (56), the expected number of incidences between points and filters is at most*

$$2^{T_{\text{point}} n + 14a_n/15 + O(n/\log n)}. \quad (57)$$

A fixed target pair supplied by Lemma 6.6 shares a product code filter with probability $1 - \text{negl}(n)$ and is reported on every nonaborting execution.

Proof. The expected incidence count is $L_{\text{ret}} M_f C_d(\tau)$, which gives (57). For the target,

$$\log_2(M_f W_d(\tau, 1 - 2/b_A + O(1/\log n))) \geq \frac{a_n}{10} - O(n/\log n) \geq \frac{a_n}{20}.$$

This dominates the threshold H_d in Theorem 4.2. The compact parameter range and the constraint $\max\{2, b_A/2\} < q < b_A$ leave fixed positive margins in the application box, so Corollary 4.4 completes the proof. $\square$

6.4.3 Reported Pair Work and the Affine Ledger

We now bound the extra pairs that the decoder may report. For a nonzero equal label difference $w \in \mathcal{L}$, put $u = \|w\|^2/\lambda^2$. The limiting filter exponent for a pair at distance u is

$$G_{A,q}(u) = 0.401 + K_t + \frac{1}{2} \log_2 \frac{u(q - u)}{b_A - 1}, \quad 1 \leq u < q. \quad (58)$$

It is maximized at $u = q/2$. Define

$$T_{\text{pair}} = 0.401 - 1 + \log_2 q - \frac{1}{2} \log_2(q - 1).$$

Lemma 6.9. *Let $P_{\neq 0}$ count incidences between filters and nonzero equal label pairs from the two retained lists. Conditional on the good superlattice event and the affine mass concentration event,*

$$\mathbb{E}[P_{\neq 0}] \leq 2^{T_{\text{pair}} n + 13a_n/30 + O(n/\log n)}.$$

Proof. This is Proposition C.1. Its sum over fixed differences uses affine averaging (37). The midpoint shell and wedge terms share a factor $b_A - u$, which cancels after the Gaussian midpoint and KL terms are expanded; this gives (58). Since $q > 2$, $u(q - u)$ is maximized at $q/2$, which gives T_{pair} and the stated finite width bound. $\square$

After deduplication within each list, a pair with $x_1 = x_2$ causes at most one bucket access per query incidence. This diagonal work is covered by (57). With the cell cap

$$S_{\text{cell}} = 2^{n/2+a_n},$$

Proposition 4.3 gives cell initialization exponent $T_{\text{point}} - 1/2 + \ell_A < T_{\text{point}}$ because $\ell_A < 1/2$. The affine running time exponent is therefore

$$K(A, q) = \max \{ \nu_A, T_{\text{point}}, T_{\text{pair}} \}. \quad (59)$$

This is the central affine tradeoff. Decreasing A raises the birthday term ν_A through the large deviation penalty $J(A)$, but lowers the retained list exponent and hence T_{point} . The cap parameter q balances the point and filter incidence and false pair terms. The improvement comes from choosing (A, q) so that the increased birthday cost meets the reduced decoding costs.

The ledger has reduced the proof to this two parameter choice. The next section makes the choice and turns it into the final algorithm.

6.5 Algorithm and main theorem

We now choose the shell displacement and cap threshold to balance the three costs in the affine ledger. The ledger (59) reduces this choice to two scalar parameters. On the balanced curve

$$q(A) = 4a\sqrt{A/e},$$

the sampling and decoding costs meet at the unique point

$$A_* = 0.905117379849\dots, \quad q_* = q(A_*) = 3.598600432817\dots$$

At this point,

$$K_{\text{aff}} = K(A_*, q_*) = 0.559568513615\dots < 0.5596. \quad (60)$$

The point (A_*, q_*) lies in the interior of the parameter range used above, and $\ell_{A_*} < 1/2$.

Remark 6.10 (Rational check). The rounded exponent does not require the exact optimizer. Replacing (A_*, q_*) throughout by the exact rational values

$$A_0 = \frac{181}{200} = 0.905, \quad q_0 = \frac{1799}{500} = 3.598$$

gives, by direct substitution in (59), $K(A_0, q_0) < 0.559573 < 0.5596$. These values also satisfy $\max\{2, b_{A_0}/2\} < q_0 < b_{A_0}$ and $\ell_{A_0} < 0.442341 < 1/2$.

The algorithm may use the rational values (A_0, q_0) from Remark 6.10. We use (A_*, q_*) below to record the balanced limiting values; this changes only the displayed $o(n)$ terms.

Compared with the centered algorithm, the affine version adds two steps: it chooses a random affine offset and samples from its shifted coset. The list, filter, and reporting steps then follow the same pattern.

Algorithm 3: AffineOneShotSVP($\mathcal{L}$)

Input: A rational basis of a full-rank lattice $\mathcal{L} \subset \mathbb{R}^n$.
Output: The shortest reported nonzero vector, or $\perp$.

- 1 Set $\kappa \leftarrow n^2$, $N \leftarrow \lceil 2^{\nu_{A_*} n + 2a_n/3} \rceil$, and $Q_{\text{rep}} \leftarrow \lceil C_Q n \log_2 n \rceil$;
- 2 Set $m_{\text{DGS}} \leftarrow \lceil 2^{n/2} \rceil$ and $R_{\text{DGS}} \leftarrow \lceil N/m_{\text{DGS}} \rceil$;
- 3 Set $L_{\text{ret}} \leftarrow 2^{\ell_{A_*} n + 5a_n/6}$, $S_{\text{cell}} \leftarrow 2^{n/2 + a_n}$, and $Z_{\text{max}} \leftarrow \lceil 2^{0.5596n + a_n} \rceil$;
- 4 Set $\tau^2 \leftarrow 1 - q_*/b_{A_*}$;
- 5 Choose $d = n + O(\log n)$ and $m = \lceil \log_2 d \rceil$ as in Section 4, and choose B_{pc} so that B_{pc}^m satisfies (55);
- 6 Set $v_{\text{best}} \leftarrow \perp$;
- 7 **foreach** *scale grid branch* j **do**
- 8 Compute s_j and a polynomial-bit rational t_j satisfying $\sqrt{2}\gamma_n s_j < t_j \leq (1 + n^{-3})\sqrt{2}\gamma_n s_j$, the branchwise shell $\mathcal{S}_{A_*, j}$, and a prime p_j satisfying (8); skip a failed branch;
- 9 Put $K_{p_j} = p_j^{-1}\mathcal{L}$ and $V_j = K_{p_j}/\mathcal{L}$;
- 10 **for** $r = 1, \dots, Q_{\text{rep}}$ **do**
- 11 Draw (Γ, U) by the pullback construction of Proposition 3.10;
- 12 Draw a uniform affine coset $\bar{z} + U \in V_j/U$, choose a lift $c \in K_{p_j}$, and put $C = c + \Gamma$;
- 13 Initialize empty deduplicated retained lists $\mathcal{A}, \mathcal{B}$;
- 14 **for** $k = 1, \dots, R_{\text{DGS}}$ **do**
- 15 Call the wrapper of Proposition 6.2, using confidence parameter κ for its internal ADRS calls; if it returns $\perp$, end the current trial and continue with the next repetition;
- 16 Consume as many of its outputs as needed to reach N outputs in total, assigning the first $\lfloor N/2 \rfloor$ to side $\mathcal{A}$ and the rest to side $\mathcal{B}$; immediately discard points outside the shell and otherwise insert the point with label $\text{lab}_{p_j}(x)$, ending the current trial before an insertion would make either retained list exceed L_{ret} ;
- 17 Release the returned block;
- 18 Run Algorithm 1 on $(\mathcal{A}, \mathcal{B}, \tau, m, B_{\text{pc}}, S_{\text{cell}}, Z_{\text{max}})$; **foreach** *reported pair* (x_1, x_2) **do**
- 19 Set $w \leftarrow x_1 - x_2$ and update v_{best} if $w \neq 0$ is shorter;
- 20 If Algorithm 1 returns abort, end the current trial and continue with the next repetition;
- 21 **return** v_{best} ;

Within the repetition loop, ending the current trial discards all trial-local state and continues with the next repetition. Wrapper calls and retention are interleaved block by block: each returned block is consumed immediately and released before the next call, and only the prefix needed to reach N outputs is consumed from the last block. The decoder's report stream is likewise consumed online.

The affine offset is represented by a vector in $\mathbb{F}_p^n/U$, and a lift in K_p has polynomial encoding length.

6.5.1 Resource bounds and operation budget

The parameter choice above bounds the three work terms in the ledger. The following proposition turns those bounds into a fixed operation budget and a space bound for one trial.

Proposition 6.11. *On the correct branch, conditional on the good superlattice and affine mass events, the ideal DGS version of one trial exceeds one of its caps with probability $2^{-\Omega(a_n)}$. Irrespective of the scale branch and random choices, every trial, including an aborted one, uses at most*

$$2^{0.5596n + o(n)} \text{ time} \quad \text{and} \quad 2^{n/2 + o(n)} \text{ space.}$$

Proof. The first resource to control is the retained list size. Equation (53) gives expected size

$$2^{\ell_{A_*}n+2a_n/3+O(n/\log n)}.$$

Markov's inequality against (56) leaves an $a_n/6 - O(n/\log n)$ margin. Conditional on the list cap, put

$$W_{\text{root}} = Z_{\mathcal{A}} + Z_{\mathcal{B}} + P_{\text{bucket}}$$

for the root product code. Lemmas 6.8 and 6.9, together with the bound of diagonal bucket work by query incidences, give the following conditional bound (conditioning on the list-cap event costs only a $1 + o(1)$ factor):

$$\mathbb{E}[W_{\text{root}}] \leq 2^{0.5596n+14a_n/15+O(n/\log n)}.$$

By Proposition 4.3, there is an absolute constant $c_{\text{bat}} \geq 1$ such that all streamed-incidence and bucket-access charges in the batching traversal are at most $c_{\text{bat}}dW_{\text{root}}$. Hence

$$\Pr \left[c_{\text{bat}}dW_{\text{root}} > \frac{Z_{\text{max}}}{4} \right] \leq \frac{4c_{\text{bat}}d\mathbb{E}[W_{\text{root}}]}{Z_{\text{max}}} = 2^{-\Omega(a_n)},$$

because $\log d + O(n/\log n) = o(a_n)$.

On the complementary good event $c_{\text{bat}}dW_{\text{root}} \leq Z_{\text{max}}/4$, Proposition 4.3 bounds cell initialization deterministically by

$$\text{poly}(n) \left(1 + \frac{dZ_{\text{max}}}{S_{\text{cell}}} \right) 2L_{\text{ret}}B_{\text{pc}}.$$

Its two leading exponents are ℓ_{A_*} and $0.5596 - 1/2 + \ell_{A_*}$, both strictly below 0.5596 because $\ell_{A_*} < 1/2$. Thus it is at most $Z_{\text{max}}/4$ for all sufficiently large n , and the entire online counter stays below Z_{max} on the displayed event.

The resource bound itself needs no correct-branch distributional estimate. On every branch, a trial makes at most R_{DGS} wrapper calls, so Proposition 6.2 bounds sampling by $2^{\nu_{A_*}n+o(n)}$ time. The retained lists have a hard cap, and Algorithm 1 performs at most Z_{max} charged operations, returning abort before operation $Z_{\text{max}} + 1$. Its polynomial overhead and $B_{\text{pc}} = 2^{O(n/\log n)}$ are absorbed in the $o(n)$ term. At most one fitting product code cell, the retained lists, and one shifted DGS block are live at the same time. The cell and DGS block have $2^{n/2+o(n)}$ size, while $\ell_{A_*} < 1/2$. This proves the time and space bounds, also for aborted trials. $\square$

6.5.2 Correctness and resources

Proof of Theorem 1.2. Fix a shortest vector v and the correct scale branch. The proof has two affine specific steps: choose a typical offset and sample its coset without bias. Proposition 3.10 supplies the good quotient direction except with probability $2^{-\Omega(a_n)}$. Conditional on this event, Proposition 6.4 supplies a typical affine offset with the same failure scale. Proposition 6.2 then supplies the ideal DGS lists analyzed in Section 6.4.

The birthday lemma places a pair with difference v in the two retained lists. Lemma 6.8 and Corollary 4.4 supply a common filter with probability $1 - \text{negl}(n)$; every nonaborting traversal then reports the pair. Proposition 6.11 bounds the cap overflow probability by $o(1)$. A union bound with the birthday and decoder failures shows that one correct branch trial succeeds with probability bounded below by an absolute constant. Equal quotient labels make every reported difference a lattice vector by (1), so the shortest nonzero report has norm exactly $\lambda_1(\mathcal{L})$.

The repetitions use fresh quotient directions, offsets, DGS calls, and product codes. For sufficiently large C_Q , the probability that all correct branch trials fail is $2^{-\Omega(n \log n)}$. The bounded

prime search has the same failure scale. Proposition 6.2 transfers the entire adaptive execution from the ideal sampler transcripts to the actual ones within $e^{-\Omega(n^2)}$ statistical distance.

Finally, Equation (59), the choice of (A_*, q_*) , and (60) give the claimed leading exponent on the correct branch. The deterministic part of Proposition 6.11 applies to every scale branch, including incorrect and aborted ones. Polynomially many branches and repetitions are therefore absorbed by the $o(n)$ terms. This proves the theorem. $\square$

Use of Generative AI

During the early preparation of this manuscript, the authors used OpenAI Codex (GPT-5.6 sol) for drafting assistance and exploratory discussion of proof details. All mathematical claims and arguments in the final manuscript were independently verified by the authors, who take full responsibility for the content.

References

- [ADRSD15] Divesh Aggarwal, Daniel Dadush, Oded Regev, and Noah Stephens-Davidowitz. Solving the shortest vector problem in 2^n time via discrete gaussian sampling. In *Proceedings of the 47th Annual ACM Symposium on Theory of Computing*, STOC '15, pages 733–742, 2015.
- [AKS01] Miklós Ajtai, Ravi Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 601–610, 2001.
- [BDGL16] Anja Becker, Léo Ducas, Nicolas Gama, and Thijs Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 10–24, 2016.
- [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.
- [Hha26a] Minki Hhan. Finding a shortest vector and more in $2^{n/2+o(n)}$ time using q -ary coset difference tree. Cryptology ePrint Archive, Paper 2026/1859, 2026.
- [Hha26b] Minki Hhan. Solving the shortest vector problem in time $2^{0.6039n}$ via mid-point hessian. arXiv:2608.02478, 2026.
- [Kim26] Jiseung Kim. One discrete gaussian sample in $2^{n/2+o(n)}$ time. arXiv:2608.03220, 2026.
- [Laa16] Thijs Laarhoven. *Search Problems in Cryptography: From Fingerprinting to Lattice Sieving*. PhD thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, February 2016.
- [Laa18] Thijs Laarhoven. Graph-based time–space trade-offs for approximate near neighbors. In *34th International Symposium on Computational Geometry*, volume 99 of *Leibniz International Proceedings in Informatics*, pages 57:1–57:14, 2018.

- [LK14] Yongjae Lee and Woo Chang Kim. Concise formulas for the surface area of the intersection of two hyperspherical caps. Technical Report IE-TR-2014-01, Department of Industrial and Systems Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea, February 2014.
- [MV13] Daniele Micciancio and Panagiotis Voulgaris. A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations. *SIAM Journal on Computing*, 42(3):1364–1391, 2013.
- [NV08] Phong Q. Nguyen and Thomas Vidick. Sieve algorithms for the shortest vector problem are practical. *Journal of Mathematical Cryptology*, 2(2):181–207, 2008.
- [PS09] Xavier Pujol and Damien Stehlé. Solving the shortest lattice vector problem in time $2^{2.465n}$. Cryptology ePrint Archive, Paper 2009/605, 2009.
- [PS20] Liudmila Prokhorenkova and Aleksandr Shekhovtsov. Graph-based nearest neighbor search: From practice to theory. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7803–7813. PMLR, 2020.

Appendix A Centered Target Concentration

Let $W = (X + Y)/2$ under the conditioning $X - Y = v$. By (24),

$$W \sim D_{\Gamma-v/2, \sigma}.$$

We bound the radial and directional deviations of this midpoint. These are the two facts needed to show that both endpoints enter the centered shell. Put $\mathcal{R}_n = (1 + \varepsilon_n)/(1 - \varepsilon_n) = 1 + o(1)$. For every $\theta < 1/2$ whose tilted parameter $\sigma/\sqrt{1 - 2\theta}$ is at least s , shifted flatness gives

$$\mathbb{E} \left[\exp \left(\frac{2\pi\theta}{\sigma^2} \|W\|^2 \right) \right] \leq \mathcal{R}_n (1 - 2\theta)^{-n/2}.$$

Take $\theta = \pm\Theta(1/\log n)$ and optimize the two Chernoff bounds. The negative tilt is admissible because $\sigma \geq \gamma_n s$. We obtain

$$\Pr \left[\left| \|W\|^2 - \frac{n\sigma^2}{2\pi} \right| > \frac{\delta_n \lambda^2}{3} \right] \leq \exp(-\Omega(n/\log^2 n)). \quad (61)$$

Completing the square for the linear moment gives, for $h \in \mathbb{R}$,

$$\mathbb{E} \left[e^{h\langle W, v \rangle} \right] \leq \mathcal{R}_n \exp \left(\frac{h^2 \sigma^2 \lambda^2}{4\pi} \right).$$

For $R = \delta_n \lambda^2/3$, choose $h = 2\pi R/(\sigma^2 \lambda^2)$ and repeat with $-h$. Then

$$\Pr[|\langle W, v \rangle| > \delta_n \lambda^2/3] \leq \exp(-\Omega(n/\log^2 n)). \quad (62)$$

Finally,

$$\|W \pm v/2\|^2 = \|W\|^2 + \frac{\lambda^2}{4} \pm \langle W, v \rangle.$$

The grid error $\hat{\lambda}^2 - \lambda^2 = O(\lambda^2/n)$ is smaller than the unused shell margin. Equations (61)–(62) therefore give the concentration invoked in Lemma 5.2.

Appendix B Finite Width Affine Annulus

We prove the two estimates in Proposition 6.4 with the windows from (40). All bounds are uniform for $A \in \mathcal{I}$.

For the first moment, the typical squared radius at parameter σ is

$$R_\sigma = \frac{n\sigma^2}{2\pi} = (a + O(\omega_n))\lambda^2.$$

The midpoint window therefore has radial ratio $A + O(\omega_n)$. Applying Lemma 3.3 to $K_p - v/2$ gives

$$V_\sigma(K_p)2^{-J(A)n+O(n\omega_n)}.$$

At the tilted parameter $\sigma\sqrt{A + O(\omega_n)}$, ambient shifted flatness holds uniformly. Lemma 3.4 with $R = \Delta_n\lambda^2$ shows that the midpoint slab complement has probability $\exp(-\Omega(n\omega_n^2))$. Removing it proves (45) with matching upper and lower bounds.

We now prove the residue square estimate. Fix a nonzero difference $z \in \mathcal{L}$ in (48) and put

$$u = \frac{\|z\|^2}{\lambda^2}, \quad z_{2,A}(u) = 2A - \frac{u}{2a}.$$

If both factors in the product are nonzero, then

$$\|w - z/2\|^2 = \left(aA - \frac{u}{4} + O(\omega_n)\right)\lambda^2. \quad (63)$$

The triangle inequality confines u to a fixed compact interval.

First assume $z_{2,A}(u) \geq 2\epsilon$ for a fixed $\epsilon > 0$. Partition this interval into bins of width ω_n . In one bin, the cumulative KL bound contributes $2^{0.401n}u^{n/2}\text{poly}(n)$, the product identity (49) contributes the difference penalty and the volume ratio $V_{\sigma/\sqrt{2}}/V_\sigma = 2^{-n/2}$, and Lemma 3.3 at the radius in (63) contributes the remaining radial rate. Relative to $V_\sigma(K_p)$, the resulting exponent is

$$H_A(u) + O(\omega_n) = -J\left(\frac{u}{2a}\right) - J\left(2A - \frac{u}{2a}\right) + O(\omega_n).$$

Lemma 6.3 bounds this by $-2J(A) + O(\omega_n)$.

It remains to control $z_{2,A}(u) < 2\epsilon$, including the $O(\omega_n)$ layer where the limiting radial ratio may be slightly negative. Equation (63) places $w - z/2$ in a shifted ball whose radial ratio under parameter $\sigma/\sqrt{2}$ is at most 3ϵ for all sufficiently large n . Lemma 3.3, in its version for a lower ball, (4), gives the normalized bin exponent

$$0.401 + \frac{1}{2}\log_2 u - \frac{1}{2} - \frac{Du}{4} - J(3\epsilon) + O(\omega_n). \quad (64)$$

All terms except $-J(3\epsilon)$ are uniformly bounded on the compact distance range. Since $J(3\epsilon) \rightarrow \infty$ as $\epsilon \downarrow 0$, choose ϵ once so that (64) is below $-2J(A)$ by a fixed constant throughout $\mathcal{I}$.

Finally, the $z = 0$ term in (48) has exponent

$$-\frac{1}{2} - J(2A) + O(\omega_n) = -2J(A) - \frac{1 + \ln A}{2 \ln 2} + O(\omega_n),$$

again with a fixed gap. There are $O(\log n)$ interior bins and one boundary range; their multiplicity is absorbed by $O(n/\log n)$. This proves (46).

Appendix C A Uniform Reported Pair Bound

The main text uses one reported pair estimate in both constructions. This appendix proves that estimate once, keeping the centered and affine cases in one calculation. Set $A = 1$ for the centered construction and take $A \in [0.85, 0.95]$ for the affine construction. In both cases write

$$y_A = aA + \frac{1}{4}, \quad b_A = 4y_A = 4aA + 1, \quad \nu_A = \nu_0 + \frac{J(A)}{2}.$$

The endpoint shell has squared radius $(y_A + O(1/\log n))\lambda^2$. Choose

$$q = b_A(1 - \tau^2), \quad 2 < q < b_A, \quad K_t = \frac{1}{2} \log_2 \frac{b_A - 1}{q - 1}.$$

Proposition C.1 (Uniform reported pair work). *On the correct branch, let $N = 2^{\nu_A n + 2a_n/3 + O(1)}$ independent DGS samples be split into two lists of sizes $m_1 = \lfloor N/2 \rfloor$ and $m_2 = N - m_1$, and use a product code of size $M_f = 2^{K_t n + a_n/10 + O(\log n)}$. In the centered case delete the zero quotient label; in the affine case retain every label. Conditional on the smoothing event, and in the affine case also on the affine mass concentration event in (47), the expected number $P_{\neq 0}$ of filter incidences of nonzero equal label pairs satisfies*

$$\mathbb{E}[P_{\neq 0}] \leq 2^{T_{\text{pair}}(q)n + 13a_n/30 + O(n/\log n)}, \quad (65)$$

where

$$T_{\text{pair}}(q) = 0.401 - 1 + \log_2 q - \frac{1}{2} \log_2(q - 1).$$

The estimate is uniform over the displayed compact range of A and every fixed compact feasible range of q .

Proof. Fix $w \in \mathcal{L} \setminus \{0\}$ and put $u = \|w\|^2/\lambda^2$. If $g_w(x)$ denotes the nonnegative contribution of the ordered pair $(x, x - w)$, we first write this contribution before taking exponents.

For a quotient slice, put

$$C = \Gamma, \quad \mathcal{X}_C = \Gamma \setminus \mathcal{L} \quad \text{in the centered case,}$$

and

$$C = c + \Gamma, \quad \mathcal{X}_C = C \quad \text{in the affine case.}$$

Because $w \in \mathcal{L} \subseteq \Gamma$, membership of x in $\mathcal{X}_C$ implies membership of $x - w$ in the same set. Define

$$\Phi_w(x) = \mathbf{1}_{\mathcal{S}_A}(x) \mathbf{1}_{\mathcal{S}_A}(x - w) W_d \left(\tau, \frac{\langle x, x - w \rangle}{\|x\| \|x - w\|} \right),$$

where $\mathcal{S}_A$ denotes the branchwise endpoint shell. For a fixed slice, the two ideal lists and the marginal uniformity of each codeword give

$$\mathbb{E}[P_w \mid C] \leq \frac{m_1 m_2 M_f}{\rho_t(C)^2} \sum_{x \in \mathcal{X}_C} \rho_t(x) \rho_t(x - w) \Phi_w(x). \quad (66)$$

The inequality only discards repeated samples when the two lists are deduplicated. Before deduplication, the right hand side is the exact expected number of filter incidences of ordered pairs with difference w .

Let H be the smoothing event, together with the affine mass concentration event in the affine case. Shifted flatness gives, on H ,

$$\rho_t(C) \geq (1 - o(1))V_t(\Gamma).$$

Inserting $\mathbf{1}_H$ into (66), dividing by $\Pr[H] = 1 - o(1)$, and then using $\mathbf{1}_H \leq 1$ gives

$$\mathbb{E}[P_w \mid H] \leq \frac{(1 + o(1))m_1 m_2 M_f}{V_t(\Gamma)^2} \mathbb{E} \left[\sum_{x \in \mathcal{X}_C} \rho_t(x) \rho_t(x - w) \Phi_w(x) \right], \quad (67)$$

where the remaining expectation is over the random quotient slice. Thus the function to be averaged is explicitly

$$g_w(x) = \rho_t(x) \rho_t(x - w) \Phi_w(x).$$

The two constructions now give, respectively,

$$\begin{aligned} \mathbb{E}_\Gamma \left[\sum_{x \in \Gamma \setminus \mathcal{L}} g_w(x) \right] &= \theta_p \sum_{x \in K_p \setminus \mathcal{L}} g_w(x) \leq \theta_p \sum_{x \in K_p} g_w(x), \\ \mathbb{E}_{U, \bar{z}} \left[\sum_{x \in c + \Gamma} g_w(x) \right] &= \theta \sum_{x \in K_p} g_w(x). \end{aligned}$$

Here $\theta_p = (p - 1)/(p^n - 1)$ and $\theta = p^{1-n}$ differ by a $1 + o(1)$ factor. Equation (67) shows where the two DGS normalizers, list multiplicity, code multiplicity, and conditioning loss enter before this averaging step.

Completing the square in the two endpoint weights gives

$$\rho_t(x) \rho_t(x - w) = e^{-\pi \|w\|^2 / (2t^2)} \rho_\sigma(x - w/2), \quad \sigma = \frac{t}{\sqrt{2}}. \quad (68)$$

The common volume factor in both constructions is

$$\frac{(p^{1-n} + o(p^{1-n}))V_\sigma(K_p)}{V_t(\Gamma)^2} = 2^{-n-a_n+O(n/\log n)}. \quad (69)$$

We now sum over w by the normalized squared distance u . The shell condition places the midpoint $x - w/2$ at squared radius

$$\left(y_A - \frac{u}{4} + O(1/\log n) \right) \lambda^2.$$

Relative to the typical squared radius $a\lambda^2$ at parameter σ , the radial ratio is

$$z(u) = \frac{b_A - u}{4a} + O(1/\log n).$$

Lemma 3.3 gives its Gaussian mass, and Lemma 3.8 bounds the number of differences up to radius $\sqrt{u}\lambda$ by $2^{0.401n} u^{n/2} \text{poly}(n)$. Including the two list sizes, the volume factor (69), and the Gaussian penalty in (68), the unfiltered exponent of a distance bin is

$$0.401 + \frac{1}{2} \log_2 u + 2\nu_A - 1 - \frac{Du}{8} - J \left(\frac{b_A - u}{4a} \right) + O(1/\log n). \quad (70)$$

Since $2\nu_A = 1 + D/8 + J(A)$ and $A = (b_A - 1)/(4a)$, direct substitution into the definition of J gives the exact identity

$$\frac{D(1-u)}{8} + J(A) - J\left(\frac{b_A - u}{4a}\right) = \frac{1}{2} \log_2 \frac{b_A - u}{b_A - 1}.$$

Thus (70) becomes

$$0.401 + \frac{1}{2} \log_2 \frac{u(b_A - u)}{b_A - 1} + O(1/\log n). \quad (71)$$

It remains to include the common filter probability without losing uniformity at its boundary. If the endpoint squared radii are $R\lambda^2$ and $S\lambda^2$, their correlation at distance u is

$$\alpha(R, S, u) = \frac{R + S - u}{2\sqrt{RS}}.$$

Define

$$\mathfrak{b}(R, S) = (\sqrt{R} + \sqrt{S})^2, \quad \mathfrak{q}(R, S) = R + S + 2(1 - 2\tau^2)\sqrt{RS}.$$

An exact calculation gives

$$1 - \frac{2\tau^2}{1 + \alpha(R, S, u)} = \frac{\mathfrak{q}(R, S) - u}{\mathfrak{b}(R, S) - u}.$$

Because $R, S = y_A + O(1/\log n)$, there are $b_n = b_A + O(1/\log n)$ and $q_n = q + O(1/\log n)$ such that every positive wedge has $u < q_n$ and, by Lemma 4.1,

$$W_d(\tau, \alpha(R, S, u)) \leq \left(\frac{q_n - u}{b_n - u}\right)^{(d-2)/2}. \quad (72)$$

For $q_n - u \geq 1/n$, multiplying (71) by the code size and the wedge factor gives the following upper bound on the normalized finite-width distance-bin exponent:

$$0.401 + K_t + \frac{1}{2} \log_2 \left(\frac{u(b_A - u)(q_n - u)}{(b_A - 1)(b_n - u)} \right) + O(1/\log n). \quad (73)$$

This is the finite-width form of the structural cancellation used in the body of the paper.

For completeness at $u = q_n$, partition $[1, q_n]$ into reverse bins

$$I_k = \left[\max \left\{ 1, q_n - \frac{k+1}{n} \right\}, q_n - \frac{k}{n} \right), \quad 0 \leq k < \lceil n(q_n - 1) \rceil.$$

At a nonempty left endpoint u_k , $q_n - u_k \geq 1/n$. Replacing the power $(d-2)/2$ in (72) by $n/2$ therefore costs only $O((\log n)^2)$. All other terms are smooth because $b_n - q_n$ has a fixed positive margin. The cumulative KL bound may be evaluated at the right endpoint of each width- $1/n$ bin at a cost $O(\log n)$. Hence (73), with an $O(n/\log n)$ error in the total exponent, holds uniformly in all $O(n)$ bins. Moreover,

$$\frac{b_A - u}{b_n - u} = 1 + O(1/\log n)$$

uniformly for $1 \leq u < q_n$. Thus the normalized bin exponent is at most

$$0.401 + K_t + \frac{1}{2} \log_2 \frac{u(q_n - u)}{b_A - 1} + O(1/\log n). \quad (74)$$

Since $q_n > 2$ for all sufficiently large n , the quadratic $u(q_n - u)$ is maximized on $u \geq 1$ at $u = q_n/2$. Using $q_n = q + O(1/\log n)$ and the definition of K_t , the maximum in (74) is

$$0.401 + \frac{1}{2} \log_2 \frac{b_A - 1}{q - 1} + \frac{1}{2} \log_2 \frac{q_n^2}{4(b_A - 1)} = T_{\text{pair}}(q) + O(1/\log n).$$

Finally, the sample padding, volume slack, and code padding contribute $4a_n/3$, $-a_n$, and $a_n/10$, whose sum is $13a_n/30$. Summing the bins proves (65). $\square$

Appendix D Product Code Coverage and Traversal

D.1 Uniform saturated coverage

We prove the two product code properties used in the main text: saturation of the target wedge and output sensitive traversal of a restricted Cartesian subcode. We begin with the coverage estimate.

Fix $\delta > 0$ and restrict to the parameter box in Theorem 4.2(iii). Put

$$\Delta(\tau, \alpha) = 1 - \frac{2\tau^2}{1 + \alpha}, \quad \Psi(\tau, \alpha) = -\frac{1}{2} \log_2 \Delta(\tau, \alpha).$$

We first record the wedge estimate with unequal thresholds, because the block decomposition perturbs the two thresholds separately.

For unit vectors of correlation α , let $W_k(r, s; \alpha)$ be the probability that a uniform point on S^{k-1} has inner products at least r and s with the two vectors. Define

$$R(r, s; \alpha) = \frac{r^2 + s^2 - 2\alpha rs}{1 - \alpha^2}, \quad \Delta(r, s; \alpha) = 1 - R(r, s; \alpha).$$

Let $\mathcal{P}_\delta$ be a compact parameter set on which

$$\delta \leq r, s \leq 1 - \delta, \quad |\alpha| \leq 1 - \delta, \quad r - \alpha s \geq \delta, \quad s - \alpha r \geq \delta, \quad \Delta(r, s; \alpha) \geq \delta.$$

The conditions $r - \alpha s \geq \delta, s - \alpha r \geq \delta$ ensure that both cap constraints are active at the closest feasible projection. The underlying positivity conditions $r - \alpha s > 0, s - \alpha r > 0$ are missing from the statement of [BDGL16, Lemma 2.2], although they are implicitly used in its boundary-intersection proof. Lemma D.1 imposes the stronger versions of these conditions to ensure uniformity of the estimate.

The equivalent condition $\min\{r/s, s/r\} > \alpha$ is imposed explicitly in [Laa16, Lemma 13.4]. If $r < \alpha s$ or $s < \alpha r$, one constraint is inactive and, up to polynomial factors, the wedge has the exponential rate of a single cap; at equality, the two rates coincide. Exact finite-dimensional formulas covering all configurations are given in [LK14], while complete asymptotic case distinctions appear in [Laa18, Lemma 3] and [PS20, Supplementary Material, Lemma 2]; the latter also explicitly identifies the additional cases missing from [BDGL16, Lemma 2.2].

For $r = s = \tau$, the strict activity conditions reduce to $\tau(1 - \alpha) > 0$, so the equal-threshold specialization used in the main text is unaffected.

Lemma D.1 (Uniform wedge estimate). *Uniformly on $\mathcal{P}_\delta$,*

$$\log_2 W_k(r, s; \alpha) = \frac{k}{2} \log_2 \Delta(r, s; \alpha) + O_\delta(\log k).$$

For $r = s = \tau$, this becomes

$$\log_2 W_k(\tau, \alpha) = -k\Psi(\tau, \alpha) + O_\delta(\log k).$$

Proof. Rotate the cap directions into the first two coordinates. The closest point satisfying both projected halfspace constraints is their boundary intersection, whose squared norm is $R(r, s; \alpha)$. The density of the two dimensional projection of a uniform point on S^{k-1} is

$$\frac{k-2}{2\pi} (1 - \|z\|^2)^{(k-4)/2} \mathbf{1}_{\{\|z\| \leq 1\}}.$$

The feasible projection is contained in $\{z : \|z\|^2 \geq R(r, s; \alpha)\}$, whose spherical projection tail is $\Delta(r, s; \alpha)^{(k-2)/2}$. This gives the upper bound.

For the lower bound, the angle and active constraint margins give a feasible rectangle of area $k^{-2}\Theta_\delta(1)$ adjacent to the closest point and directed into both halfspaces. Throughout this rectangle,

$$\|z\|^2 \leq R(r, s; \alpha) + O_\delta(1/k).$$

Because $\Delta(r, s; \alpha) \geq \delta$, integrating the projection density over this rectangle gives

$$k^{-O_\delta(1)} \Delta(r, s; \alpha)^{k/2}.$$

The upper and lower bounds differ only by a polynomial factor, proving the lemma uniformly. $\square$

Write $d = mb$. After the Haar rotation, decompose the two fixed unit directions as $z = (z_1, \dots, z_m)$ and $z' = (z'_1, \dots, z'_m)$ with blocks in $\mathbb{R}^b$.

Lemma D.2 (Uniform block subwedge). *On an event of probability at least $1 - \Delta_d$, where $\Delta_d = \exp(-\Omega(\log^2 d))$, the following holds simultaneously for every block:*

$$\left| m \|z_i\|^2 - 1 \right| \leq \epsilon_d, \quad \left| m \|z'_i\|^2 - 1 \right| \leq \epsilon_d, \quad |m \langle z_i, z'_i \rangle - \alpha| \leq \epsilon_d, \quad \epsilon_d = O\left(\frac{\text{polylog } d}{\sqrt{d}}\right). \quad (75)$$

For each regular block, put

$$\hat{z}_i = \frac{z_i}{\|z_i\|}, \quad \hat{z}'_i = \frac{z'_i}{\|z'_i\|}, \quad r_i = \frac{\tau}{\sqrt{m} \|z_i\|}, \quad s_i = \frac{\tau}{\sqrt{m} \|z'_i\|}, \quad \alpha_i = \langle \hat{z}_i, \hat{z}'_i \rangle,$$

and define

$$U_i = \left\{ u \in S^{b-1} : \langle u, \hat{z}_i \rangle \geq r_i, \langle u, \hat{z}'_i \rangle \geq s_i \right\}.$$

Every concatenation $c = (u_1/\sqrt{m}, \dots, u_m/\sqrt{m})$ with $u_i \in U_i$ lies in the global common wedge. If w_i is the normalized measure of U_i , then

$$w_i \geq W_d(\tau, \alpha)^{1/m} 2^{-L_d}, \quad L_d = O_\delta(\sqrt{d} \text{polylog } d). \quad (76)$$

Proof. The simultaneous concentration statement is the block norm and block inner product estimate from the Haar decomposition in [BDGL16, Appendix C], with deviation parameter $\Theta(\log^2 d)$ and a union bound over $m = O(\log d)$ blocks. Equation (75) implies

$$r_i = \tau + O(\epsilon_d), \quad s_i = \tau + O(\epsilon_d), \quad \alpha_i = \alpha + O(\epsilon_d).$$

The fixed margins in Theorem 4.2(iii) therefore put all $(r_i, s_i; \alpha_i)$ in one set $\mathcal{P}_{\delta'}$ for some $\delta' > 0$ and all sufficiently large d .

If $u_i \in U_i$, then

$$\langle u_i/\sqrt{m}, z_i \rangle = \frac{\|z_i\|}{\sqrt{m}} \langle u_i, \hat{z}_i \rangle \geq \frac{\tau}{m}.$$

Summing over the blocks gives $\langle c, z \rangle \geq \tau$; the same calculation gives $\langle c, z' \rangle \geq \tau$. This proves the Cartesian product containment.

Lemma D.1 and the bounded derivatives of $\log \Delta$ on $\mathcal{P}_{\delta'}$ give

$$\log_2 w_i = -b\Psi(\tau, \alpha) + O_\delta(b\epsilon_d + \log b).$$

The same lemma in dimension d gives

$$\frac{1}{m} \log_2 W_d(\tau, \alpha) = -b\Psi(\tau, \alpha) + O_\delta(\log d/m).$$

Their difference is bounded by

$$L_d = O_\delta(b\epsilon_d + \log d) = O_\delta(\sqrt{d} \text{ polylog } d),$$

which proves (76). $\square$

Lemma D.3 (Uniform product code saturation). *With L_d and Δ_d from Lemma D.2,*

$$\Pr[a \text{ common codeword exists}] \geq (1 - \Delta_d) \left(1 - \exp\{-B_{\text{pc}} W_d(\tau, \alpha)^{1/m} 2^{-L_d}\}\right)^m. \quad (77)$$

Consequently (17) holds with

$$H_d = mL_d + 3m \log_2 d = O(\sqrt{d} \text{ polylog } d).$$

Proof. Lemma D.2 supplies an event of probability $1 - \Delta_d$ on which every block has a subwedge of measure at least $W_d(\tau, \alpha)^{1/m} 2^{-L_d}$ and whose Cartesian product is contained in the global wedge.

Each block code contains B_{pc} independent uniform points, and the block codes are independent. Conditional on block regularity, the probability that every block code hits its subwedge is at least

$$\prod_{i=1}^m (1 - (1 - w_i)^{B_{\text{pc}}}) \geq \left(1 - \exp\{-B_{\text{pc}} W_d(\tau, \alpha)^{1/m} 2^{-L_d}\}\right)^m.$$

Any choice of one hit per block concatenates to a point in the original wedge, proving (77).

If $MW_d(\tau, \alpha) \geq 2^{H_d}$, then

$$B_{\text{pc}} W_d(\tau, \alpha)^{1/m} 2^{-L_d} = (MW_d(\tau, \alpha))^{1/m} 2^{-L_d} \geq d^3.$$

The conditional miss probability is at most me^{-d^3} ; adding the block regularity failure proves (17). $\square$

D.2 Restricted Decoding with Bounded Space

We first justify the restricted subcode decoder used by the batching tree. For a fixed query direction and each block, sort the allowed block points by their inner product with that direction. During enumeration in depth first order, a prefix is visited only when its current score plus the maximum contribution from every unassigned block can still reach the cap threshold. The children of a prefix are scanned in sorted order, so scanning stops after the first infeasible child.

Every feasible prefix has at least one output completion. Charge it to its lexicographically first output completion. At each depth, an output receives at most one charge, and every feasible parent incurs at most one additional test of an infeasible child. Hence the traversal visits $O(m(1 + |\text{Decode}_D(z)|))$ prefixes after sorting. Sorting and forming the rotated block coordinates costs $\text{poly}(d)B_{\text{pc}}$ time and space. This proves Theorem 4.2(i) for every Cartesian subcode, not only for the root code.

Proof of Proposition 4.3. A cell is a Cartesian subcode $Q(S_1 \times \cdots \times S_m)$. Fix the binary tree obtained by repeatedly splitting one S_i of size at least two into two nearly equal index sets. Its leaves are individual codewords, and its depth is at most

$$D_T \leq m \lceil \log_2 B_{\text{pc}} \rceil = O(d).$$

For a cell D , let $z_{\mathcal{A}}(D)$ be its insertion side incidence count. A cell is retained precisely when $z_{\mathcal{A}}(D) \leq S$. An overflowing cell is discarded and recomputed in both children. A singleton cell has load at most $|\mathcal{A}| \leq S$, so the recursion terminates.

At each tree depth, the processed cells are disjoint and their insertion weights sum to at most $Z_{\mathcal{A}}$. Every internal processed cell has weight larger than S , so the total number of processed cells is

$$O\left(1 + \frac{D_T Z_{\mathcal{A}}}{S}\right).$$

Each insertion incidence is streamed at most once per depth, for a total of $O(D_T Z_{\mathcal{A}})$ outputs. The retained cells partition the full code, so the query outputs over all retained cells total $Z_{\mathcal{B}}$. Their label buckets perform exactly P_{bucket} accesses. Adding the $\text{poly}(d)B_{\text{pc}}$ initialization cost for each point and processed cell gives (19).

At most $S + 1$ insertion records, the two input lists, and one decoder stack are live at once, giving (18). Every full code word belongs to exactly one retained descendant, so every qualifying pair is reported unless the online budget is exhausted. Because the counter is charged before each operation, an exhausted budget returns abort before operation $Z_{\text{max}} + 1$. $\square$