Superlattice methods for the shortest vector problem and discrete Gaussian sampling

Minki Hhan[1] Jiseung Kim[2] Sunyeop Kim[3,4] Thijs Laarhoven[5]

[1]KAIST, Daejeon, South Korea
[2]Jeonbuk National University, Jeonju, South Korea
[3]Korea University, Seoul, South Korea
[4]Nanyang Technological University, Singapore
[5]NXP Semiconductors, Eindhoven, The Netherlands

minkihhan@kaist.ac.kr jiseungkim@jbnu.ac.kr sunyeopkim@korea.ac.kr mail@thijs.com

Abstract

We give worst-case algorithms for the shortest vector problem (SVP) and discrete Gaussian sampling (DGS) at arbitrary widths. Both algorithms generate lists on a dense superlattice and combine vectors along a chain down to the input lattice. Random shifts let us control the expected number of representations of a vector regardless of the lattice. For SVP we balance these counts with a novel midpoint-shell test to ensure output distributions are sufficiently uniform. For DGS we use Gaussian weights and correct the probabilities of selecting entries from the final list.

For SVP, our optimized algorithm runs in time $2^{0.409n+o(n)}$ and space $2^{0.259n+o(n)}$, improving the best $2^{n/2+o(n)}$ worst-case SVP complexities of Hhan [ePrint, 2026]. For Haar-random lattices, or assuming a natural upper bound on the kissing constant, the costs of the algorithm match the $2^{0.292n+o(n)}$ time and $2^{0.208n+o(n)}$ space of Becker–Ducas–Gama–Laarhoven [SODA 2016].

For quantum SVP, through a series of refinements and generalizations we obtain worst-case time and space $2^{0.247n+o(n)}$. This improves the best known heuristic quantum SVP time complexity of $2^{0.256n+o(n)}$ due to Bonnetain–Chailloux–Schrottenloher–Shen [Eurocrypt 2023], without requiring any heuristic assumptions.

For discrete Gaussian sampling, we provide a sampling algorithm at any width with distribution error $2^{-\kappa}$, running in worst-case time $2^{0.292n+o(n)}\,\mathrm{poly}(\kappa)$ and space $2^{0.208n+o(n)}\,\mathrm{poly}(\kappa)$, improving upon the previous best $2^{n+o(n)}$ cost for sampling at arbitrary widths of Aggarwal–Dadush–Regev–Stephens-Davidowitz [STOC 2015].

As further applications, we obtain fast algorithms for approximate-SVP/CVP, bounded distance decoding, and estimating the Gaussian mass and smoothing parameter of any lattice.

AI assistance: OpenAI ChatGPT-6.1 Sol and Codex, as well as Anthropic Claude Opus 5.5 assisted with writing, proof checking, and manuscript drafting and integration. In particular, the refined shell optimization for quantum SVP and the various techniques to speed up the quantum SVP algorithm were entirely the result of AI explorations. The authors are responsible for the correctness, originality, presentation, and integrity of the manuscript.

Contents

1 Introduction **4**
- 1.1 Related work . . . 4
- 1.2 Contributions . . . 5
- 1.3 Techniques . . . 11
- 1.4 Open problems . . . 15
- 1.5 Concurrent work . . . 15
- 1.6 Organization . . . 16

2 Preliminaries **17**
- 2.1 Lattice background . . . 17
- 2.2 Discrete and continuous Gaussians . . . 19
- 2.3 Lattice algorithms . . . 22
- 2.4 Probability bounds . . . 22
- 2.5 Fast search algorithms . . . 24

3 Discrete Gaussian Sampling in Time $2^{0.5n+o(n)}$ – Restriction **27**
- 3.1 Random extensions and the restriction sampler . . . 27
- 3.2 Cost of restriction . . . 29

4 Discrete Gaussian Sampling in Time $2^{0.2924\ldots n+o(n)}$ – Gaussian-Weighted Merges **32**
- 4.1 The sampler . . . 33
- 4.2 The guarantee of the sampler . . . 36
- 4.3 The chain and Gaussian-weighted lists . . . 38
- 4.4 Fixing an entry and correcting the selection probability . . . 43
- 4.5 Technical lemmas . . . 47
- 4.6 Proof of the sampling theorem . . . 58

5 Classical SVP in Time $2^{0.4385\ldots n+o(n)}$ – A Single Shell **59**
- 5.1 The superlattice chain . . . 60
- 5.2 The basic SVP algorithm . . . 68
- 5.3 Improved classical algorithms using spherical filtering . . . 73
- 5.4 Quantum search on the generated lists . . . 76
- 5.5 SVP specializations . . . 79

6 Classical SVP in Time $2^{0.4088\ldots n+o(n)}$ – Smaller Shells **81**
- 6.1 The modified fixed-radius construction . . . 81
- 6.2 Preparing the two shells and counting shortest differences . . . 83
- 6.3 Spherical filtering with the smaller shell . . . 84
- 6.4 Work limits and proof of the tradeoff . . . 85
- 6.5 The radius balance in the cost bound . . . 86
- 6.6 Quantum reporting at the same fixed radius . . . 86

7 Classical SVP in Time $2^{0.4085\ldots n+o(n)}$ – Fast Matrix Multiplication **89**
- 7.1 Matrix multiplication and grouped search . . . 89
- 7.2 Coordinates for the final BDGL buckets . . . 90
- 7.3 Numerical choice and total SVP cost . . . 92

8 Quantum SVP in Time $2^{0.2937...n+o(n)}$ – Variable Shells **93**
8.1 The variable-radius reference lists . . . 93
8.2 An asymmetric index and implicit queries . . . 94
8.3 The improved binary bound . . . 96

9 Quantum SVP in Time $2^{0.2652...n+o(n)}$ – Gaussian Splitting **98**
9.1 List construction by Gaussian splitting . . . 98
9.2 Gaussian correlation bounds . . . 101
9.3 Moments of the occurrence counts . . . 107
9.4 Finding a shortest vector from the top lists . . . 110
9.5 Quantum access and the dependent last coset . . . 112
9.6 Finite checks and completion . . . 114

10 Quantum SVP in Time $2^{0.2478...n+o(n)}$ – Multi-List Merges **117**
10.1 One list type and its prefix geometry . . . 117
10.2 Quantum work of a tuple merge . . . 119
10.3 Checking every shared set . . . 119
10.4 A fixed parameter instance . . . 121
10.5 Completion of the bounds . . . 122

11 Quantum SVP in Time $2^{0.2468...n+o(n)}$ – Selection with Gaussian Weights **124**
11.1 Several types and selection by seed identifiers . . . 124
11.2 Passing Gaussian weights down the merge tree . . . 125
11.3 Finding a shortest-vector representation after selection . . . 128
11.4 Search costs and bounded execution . . . 129
11.5 Choosing the merge geometry . . . 130

12 Applications **133**
12.1 Approximate SVP and CVP . . . 133
12.2 Bounded distance decoding . . . 133
12.3 Gaussian mass estimation . . . 134
12.4 Smoothing parameter estimation . . . 135

A Refined lattice point counts **142**
A.1 The analytic input . . . 143
A.2 Counts on a fixed radius interval . . . 144
A.3 Extension to all radii . . . 146

B Certificates for the selected quantum SVP construction **148**
B.1 Exact inputs . . . 148
B.2 Interval checks . . . 148

1 Introduction

As lattice-based cryptography has emerged as a prime candidate for quantum-safe cryptography, understanding the computational hardness of lattice problems is critical for designing efficient and secure post-quantum cryptographic schemes. Perhaps the most fundamental hard lattice problem is the shortest vector problem (SVP), which asks for a shortest non-zero vector of a lattice. Another central hard problem is discrete Gaussian sampling (DGS), asking to sample accurately from a Gaussian distribution restricted to the discrete set of lattice points, for any Gaussian width s. Discrete Gaussians can be constructed around any offset; unless stated otherwise, a discrete Gaussian in this paper refers to *centered* discrete Gaussian sampling, centered around the origin.

For both SVP and (centered) DGS, the best known algorithms take time exponential in the lattice dimension n. In this work we study worst-case algorithms for both problems, which provably work on any input lattice, and we study how the best complexities scale with the dimension n.

1.1 Related work

SVP algorithms. For solving (exact) high-dimensional SVP, there are two main families of algorithms, the first of which is lattice enumeration. After suitable preprocessing of a lattice basis [LLL82, SE94], these methods perform a recursive brute-force search over all possible combinations of basis vectors that could lead to a shorter vector [Kan83, FP85, SE94, GNR10, HS10, FGL26]. Its time complexity scales as $2^{\Theta(n \log n)}$, using $\mathrm{poly}(n)$ space.

The other family is lattice sieving, where by saturating the space with large lists of lattice vectors, one guarantees that suitable combinations of vectors will exist that form shorter vectors, guaranteeing convergence to a solution [AKS01, MV10, BDGL16]. These algorithms take $2^{\Theta(n)}$ time and $2^{\Theta(n)}$ space. The algorithm of Aggarwal–Dadush–Regev–Stephens-Davidowitz [ADRS15] long held the record for the fastest worst-case SVP algorithm, running in $2^{n+o(n)}$ time and space.

Various recent, concurrent improvements have decreased the worst-case SVP exponent [GFH26b, GFH26a, Kim26, Hha26b, Hha26a], making use of the fast above-smoothing DGS algorithm of [ADRS15]. Before this work, the best complexities were due to Hhan, solving worst-case SVP in $2^{n/2+o(n)}$ time and space [Hha26a]. Under heuristic assumptions [BDGL16], or on Haar-random lattices [ZWW26, DL26], the best known time complexity for SVP scales as $(3/2)^{n/2+o(n)}$.

DGS algorithms. For discrete Gaussian sampling, Klein described a randomized nearest-plane sampler running in polynomial time and space at high widths [Kle00], solving DGS at high accuracy above a threshold controlled by the Gram–Schmidt lengths of the basis [GPV08]. Klein's algorithm can informally be understood as a randomized version of lattice enumeration [Sch03].

The "sieving"-counterpart is the algorithm of Aggarwal–Dadush–Regev–Stephens-Davidowitz, who gave an arbitrary-width Gaussian sampling algorithm running in time and space $2^{n+o(n)}$, as well as a faster sampler above $\sqrt{2}$ times the smoothing parameter running in time $2^{n/2+o(n)}$ [ADRS15]. Aggarwal–Li–Stephens-Davidowitz later analyzed a simpler combination algorithm through mixtures of Gaussians along a lattice tower [ALS21]. For random lattices, recent work showed one can sample an exponentially large batch of discrete Gaussian samples at any width in time $(3/2)^{n/2+o(n)}$ [DL26].

SVP vs. DGS. Stephens-Davidowitz showed a strong connection between DGS and SVP [Ste16]; with an SVP oracle, one can solve DGS at arbitrary widths, but with inverse-polynomial (rather than exponentially small) distributional errors. Since the best reduction for solving SVP with a DGS oracle has exponential overhead [ADRS15, Proposition 4.3], it remains unclear whether DGS with exponentially small error or exact SVP is harder.

Sparsification and superlattices. Ideas of lattice sparsification and using sub/superlattices have a rich history in the study of lattices [Rog50, GMSS99, KS99, Kho03, Dad13, DRS14, DK16, Ste16, DADRT23]. Dadush–Kun developed lattice sparsification through random modular conditions [DK16]. The same random restriction appears in distance-guaranteed CVP reductions [DRS14].

Algorithmically, chains of superlattices were used for solving hard lattice problems in Aggarwal–Dadush–Regev–Stephens-Davidowitz's algorithm [ADRS15] (and the related [ALS21, ACKS25]), and the Becker–Gama–Joux overlattice sieve [BGJ14]. On a high level, both use chains of superlattices to reach a lattice regime for which sampling is easy. Then, by suitably combining vectors from a large list, one attempts to go back up the chain while preserving the nice properties from the superlattice, while not dropping too many vectors along the way.

Also closely related is the work of Ducas–Engelberts–Loyer [DEL25], who give a Gaussian version of Wagner's generalized birthday algorithm [Wag02] for q-ary lattices. Their algorithm follows a chain of projected lattices, lifts to auxiliary superlattices, and combines same-coset pairs while controlling conditional Gaussian distributions [DEL25].

The concurrent work of Gao–Feng–Hu [GFH26a, GFH26b] closely touches upon the techniques in this paper as well. Their techniques overlap with [Kim26] which is subsumed by this work.

Lattice point counts and kissing constants. Bounding lattice point counts in Euclidean balls is a common ingredient for analyzing lattice algorithms on worst-case lattices [AKS01, NV08, PS09, MV10, HPS11, ADRS15]. Pujol–Stehlé [PS09] showed how an upper bound on the kissing constant directly implies an upper bound on the number of lattice points within a ball of any radius. For our analyses, a tight point count bound directly affects the leading constants in the exponents: further lowering this bound would directly imply lower costs for our SVP algorithm.

For a long time, the Kabatiansky–Levenshtein upper bound on the kissing constant, $2^{0.4009\ldots n+o(n)}$, was the best known general upper bound [KL78, Corollary 1]. Recent work of OpenAI improved the upper bound to $2^{0.3966\ldots n+o(n)}$ [Ope26, Chapter 2], which directly translates to improved lattice point counts. The same paper described a packing density upper bound $(e/(2\pi))^{n/2+o(n)} = 2^{-0.6044\ldots n+o(n)}$ [Ope26, Ch. 1, Theorem 1.1], which unfortunately does not directly translate to an improved unconditional kissing constant. Kominers used this packing density bound to reduce the *lattice* kissing constant to $2^{0.3955\ldots n+o(n)}$ [Kom26]. Before this work, an open question was whether a tighter upper bound on the lattice point counts can be derived than the $2^{0.3966\ldots n+o(n)}$ that one would obtain from using the best-known kissing constant bound.

1.2 Contributions

In this work we study algorithmic approaches for SVP and DGS based on chains of superlattices. The three main contributions at a glance are:

- Worst-case SVP in time $2^{0.4085\ldots n+o(n)}$ and space $2^{0.2593\ldots n+o(n)}$;
- Worst-case quantum SVP in time and space $2^{0.2468\ldots n+o(n)}$;
- Worst-case DGS at any width in time $(3/2)^{n/2+o(n)}$ and space $(4/3)^{n/2+o(n)}$, up to factors polynomial in the accuracy parameter.

We extend the input lattice to a chain $\mathcal{L} \subset \Gamma_0 \subset \cdots \subset \Gamma_D$. The first extension controls the number of representations of a vector as a difference or sum. On the densest lattice Γ_D we generate lists by rounding. We then combine their entries to obtain lists on successively sparser lattices. For SVP the final search looks for a shortest vector. For DGS we choose the merge probabilities and correct the final selection to produce the required distribution.

1.2.1 Sharper lattice point counts: from $0.3966\ldots$ to $0.3955\ldots$

The SVP algorithms below use worst-case lattice point counts to bound their costs. For counting the number of non-zero lattice vectors of norm at most R, write $N_{\mathcal{L}}(R) = |\{\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\} : \|\mathbf{v}\| \leq R\}|$. Using similar arguments as Pujol–Stehlé [PS09, Lemma 3] with the improved OpenAI kissing constant bound, it immediately follows that for all lattices $\mathcal{L}$ and all $r \geq 1$:

$$N_{\mathcal{L}}(r \cdot \lambda_1(\mathcal{L})) \leq 2^{0.3966\ldots n+o(n)} r^n. \tag{1.1}$$

Kominers obtained the smaller *lattice* kissing constant upper bound $2^{0.3955\ldots n+o(n)}$ for arbitrary lattices [Kom26]. This directly gives a sharper upper bound on $N_{\mathcal{L}}(1 \cdot \lambda_1(\mathcal{L}))$. For $r > 1$ these lattice points are no longer the minimal vectors of the corresponding lattice, hence the bound need not apply. We show that this argument can (non-trivially) be extended to apply to all radii.

Proposition 1.1 (Refined lattice-point count)**.** *For any lattice $\mathcal{L}$ and for all $r \geq 1$,*

$$N_{\mathcal{L}}(r \cdot \lambda_1(\mathcal{L})) \leq \left(\frac{2e}{\pi}\right)^{n/2+o(n)} r^n = 2^{0.3955\ldots n+o(n)} r^n. \tag{1.2}$$

Its proof is the main result of Appendix A, which can be read independently of the rest of the paper. The displayed worst-case SVP bounds use this centered point-count estimate. The Gaussian quantum constructions also use the general spherical-code bound of [Ope26, Chapter 2, Remark 1.7] for separated projected shells.

1.2.2 Discrete Gaussian sampling in time $2^{0.2924\ldots n+o(n)}$ at arbitrary widths

One of the properties we generally require for Γ_0 is smoothness. In Section 3 we characterize how many cosets must be added to make a lattice smooth at a prescribed width. Concretely, for a lattice $\mathcal{L}$, denote its dual by $\mathcal{L}^*$, denote the standard Gaussian function by $\rho_s(\mathbf{x}) = e^{-\pi\|\mathbf{x}\|^2/s^2}$, and let the smoothing parameter $\eta_\varepsilon(\mathcal{L})$ denote the least s such that $\rho_{1/s}(\mathcal{L}^* \setminus \{\mathbf{0}\}) \leq \varepsilon$. We show that the minimum index $[\Gamma_0 : \mathcal{L}]$ required to achieve smoothness scales as $\Theta(\rho_{1/s}(\mathcal{L}^*))$ for each fixed $0 < \varepsilon < 1$.

Proposition 1.2 (Minimum smoothing index)**.** *For every full-rank lattice $\mathcal{L}$, width $s > 0$, and $0 < \varepsilon \leq 1$, let $I_{s,\varepsilon}(\mathcal{L})$ be the least index $[\Gamma_0 : \mathcal{L}]$ of a superlattice $\Gamma_0 \supseteq \mathcal{L}$ with $\eta_\varepsilon(\Gamma_0) \leq s$. Then*

$$(1-\varepsilon)\rho_{1/s}(\mathcal{L}^*) \leq I_{s,\varepsilon}(\mathcal{L}) \leq \left(\frac{18}{\varepsilon}\right)\rho_{1/s}(\mathcal{L}^*). \tag{1.3}$$

In particular, the minimum index for smoothing error $1/2$ lies between $\frac{1}{2}\rho_{1/s}(\mathcal{L}^*)$ and $36\rho_{1/s}(\mathcal{L}^*)$.

To sample at small widths, we first sample on a superlattice that is smooth at the requested width. We accept only samples that lie in the input lattice. The bounds on the extension index and the Gaussian masses control the probability of acceptance.

Proposition 1.3 (Discrete Gaussian sampling by restriction; previously [Kim26][1])**.** *Given a basis of an n-dimensional lattice $\mathcal{L}$ and a Gaussian squared width $s^2 > 0$, a randomized algorithm returns a single point $\mathbf{x} \in \mathcal{L}$ within total variation e^{-2n^3} of $D_{\mathcal{L},s}$ in time and space*

$$\mathrm{T} \leq 2^{n/2+o(n)}, \qquad \mathrm{S} \leq 2^{n/2+o(n)}. \tag{1.4}$$

[1] This paper subsumes [Kim26] by one of the authors.

This answers the question in [ADRS15, Section 1.3]. Section 3 also gives a lower bound for the cost of sampling by restriction.

The restriction sampler keeps only samples that already lie in $\mathcal{L}$. The sampler of Section 4 generates Gaussian-weighted lists on shifted superlattices and combines pairs to reach $\mathcal{L}$. Gaussian weights at the leaves and acceptance probabilities at each merge give the desired occurrence weights. A final correction removes the bias of selecting a uniform position from a random list. This gives the faster sampler below.

Theorem 1.4 (Discrete Gaussian sampling)**.** *For an input lattice $\mathcal{L}$ of rank n, a squared Gaussian width $s^2 > 0$ and an accuracy parameter $\kappa \geq 1$, a bounded randomized (classical) algorithm outputs a point $\mathbf{x} \in \mathcal{L}$ within total variation $2^{-\kappa}$ of $D_{\mathcal{L},s}$ in time* T *and space* S *satisfying*

$$\mathrm{T} = \left(\frac{3}{2}\right)^{n/2+o(n)} \mathrm{poly}(\kappa), \qquad \mathrm{S} = \left(\frac{4}{3}\right)^{n/2+o(n)} \mathrm{poly}(\kappa). \tag{1.5}$$

For κ polynomial in n, the algorithm can sample at any width with exponentially small error. Table 3 compares its cost with previous results.

1.2.3 Worst-case SVP in time $2^{0.4085\ldots n+o(n)}$

Section 5 first develops a fixed-radius algorithm whose list vectors have length close to $\lambda_1(\mathcal{L})$. Fresh coset shifts and a midpoint-shell test keep the expected multiplicities nearly uniform across the shell. Section 6 uses a smaller common radius. It chooses the first extension using the midpoint region for the final search. The smaller shell then needs fewer entries for a shortest vector to occur.

Theorem 1.5 (Classical SVP time–space tradeoff)**.** *Suppose a family of n-dimensional input lattices satisfies, for a constant $\beta \in [\frac{1}{2}\log_2(4/3),\ \frac{1}{2}]$ and for all $r \geq 1$,*

$$N_{\mathcal{L}}(r\lambda_1(\mathcal{L})) \leq 2^{\beta n+o(n)} r^n. \tag{1.6}$$

For every $\alpha \in [\frac{1}{2}\log_2(4/3),\ \frac{1}{2}\log_2(1+2^{2\beta-2})]$, a randomized algorithm solves exact SVP with success probability at least $2/3$, time $\mathrm{T} = 2^{t(\alpha)n+o(n)}$ and space $\mathrm{S} = 2^{s(\alpha)n+o(n)}$, where

$$t(\alpha) = \beta - \frac{1}{2}\log_2\left[1 - \left(1 - \frac{2^{2\alpha+1}}{1+2^{2-2\beta}}\right)^2\right], \qquad s(\alpha) = \alpha. \tag{1.7}$$

The time exponent decreases as α increases over the stated interval. Substituting the universal point-count exponent gives the following worst-case tradeoff.

Corollary 1.6 (Worst-case classical SVP)**.** *For $\beta = \frac{1}{2}\log_2(2e/\pi)$, Theorem 1.5 gives the minimum-time endpoint*

$$\mathrm{T} = \left(\frac{2\pi}{2\pi-e}\right)^{n/2+o(n)} = 2^{0.4088\ldots n+o(n)}, \qquad \mathrm{S} = \left(\frac{2\pi+e}{2\pi}\right)^{n/2+o(n)} = 2^{0.2593\ldots n+o(n)} \tag{1.8}$$

and the minimum-space endpoint

$$\mathrm{T} = \left(\frac{9\cdot(2\pi+e)^2}{8\pi\cdot(6\pi-e)}\right)^{n/2+o(n)} = 2^{0.4234\ldots n+o(n)}, \qquad \mathrm{S} = \left(\frac{4}{3}\right)^{n/2+o(n)} = 2^{0.2075\ldots n+o(n)}. \tag{1.9}$$

The shell radius $(\sqrt{1/4+\pi/(2e)}+o(1))\lambda_1(\mathcal{L})$, approximately $0.91\lambda_1(\mathcal{L})$, is used across the tradeoff.

The minimum-time endpoint can be improved slightly by a different search between the two final lists. Section 7 keeps the superlattice chain, shell radius and occurrence weights of Section 6, and the analyses of all earlier merges remain the same. Within the final spherical filters, it only replaces individual pair comparisons by a search that tests many pairs together using fast rectangular matrix multiplication. This follows the correlation-search strategy of [KKK18], also used for heuristic lattice sieving in [GH26]. For our final-search parameters, the current rectangular matrix-multiplication bounds already give an unconditional improvement.

Theorem 1.7 (Worst-case classical SVP with fast matrix multiplication)**.** *For every input lattice $\mathcal{L}$ of rank n, a randomized algorithm solves exact SVP with success probability at least $2/3$ in time and space*

$$\mathrm{T} = 2^{0.4085\ldots n+o(n)}, \qquad \mathrm{S} = \left(\frac{2\pi+e}{2\pi}\right)^{n/2+o(n)} = 2^{0.2593\ldots n+o(n)} \tag{1.10}$$

This improves the time exponent from $0.4088\ldots$ to $0.4085\ldots$, without increasing the memory. The only change that is applied is replacing the final nearest neighbor search routine with a fast-matrix-multiplication-optimized nearest neighbor search routine.

We further remark that, should the matrix multiplication constant be reduced to $\omega = 2$, initial calculations and reoptimizations suggest the worst-case SVP exponent with this method would decrease to $0.4064\ldots$.

1.2.4 Conditionally solving SVP in time $2^{0.2924\ldots n+o(n)}$

The volume-based lower bound on the general spherical kissing number is $(4/3)^{n/2+o(n)}$. It is natural to ask whether $\beta = \frac{1}{2}\log_2(\frac{4}{3})$ suffices for the centered lattice-point profile in (1.6). The same hypothesis was used in [AMS26] to analyze the costs of their SVP algorithm under this kissing constant assumption. In our case, this assumption would lead to worst-case SVP complexities matching the best heuristic SVP complexities of Becker–Ducas–Gama–Laarhoven [BDGL16].

Corollary 1.8 (SVP under a kissing constant assumption)**.** *If the general spherical kissing number is at most $(4/3)^{n/2+o(n)}$, then SVP can be solved on every input lattice in time and space*

$$\mathrm{T} = \left(\frac{3}{2}\right)^{n/2+o(n)} = 2^{0.2924\ldots n+o(n)}, \qquad \mathrm{S} = \left(\frac{4}{3}\right)^{n/2+o(n)} = 2^{0.2075\ldots n+o(n)}. \tag{1.11}$$

Quantumly, the complexities then become $\mathrm{T},\mathrm{S} = (13/9)^{n/2+o(n)} = 2^{0.2652\ldots n+o(n)}$.

Concurrently to this work, two papers independently showed that the heuristic SVP complexities of [BDGL16] can be made provable on Haar-random lattices [DL26, ZWW26]. Instantiating our algorithm on Haar-random lattices, we achieve the same asymptotic scaling. Note that Haar-random lattices satisfy the point counting bound $N_{\mathcal{L}}(r\lambda_1(\mathcal{L})) \le 2^{o(n)}r^n$, corresponding to $\beta = o(1)$; the following result can thus be read as weaker than Corollary 1.8, which only assumes $\beta \le \frac{1}{2}\log_2(4/3)$.

Corollary 1.9 (SVP on Haar-random lattices)**.** *There are classes $\mathcal{H}_n$ of unimodular lattices with Haar probability $1-o(1)$ such that SVP can be solved for every $\mathcal{L} \in \mathcal{H}_n$ in time and space*

$$\mathrm{T} = \left(\frac{3}{2}\right)^{n/2+o(n)} = 2^{0.2924\ldots n+o(n)}, \qquad \mathrm{S} = \left(\frac{4}{3}\right)^{n/2+o(n)} = 2^{0.2075\ldots n+o(n)}. \tag{1.12}$$

Quantumly, the complexities for these lattices are $\mathrm{T},\mathrm{S} = (13/9)^{n/2+o(n)} = 2^{0.2652\ldots n+o(n)}$.

An overview of the best SVP complexities is given in Tables 1 (classically) and 2 (quantumly).

1.2.5 Worst-case quantum SVP in time $2^{0.2468\ldots n+o(n)}$

The classical construction also gives a starting point for quantum SVP. Section 6 uses quantum reporting to generate the smaller fixed-radius lists. The subsequent sections give simpler variants of the final algorithm. Section 8 varies the shell radii and treats the two child lists asymmetrically. Section 9 introduces Gaussian splitting and a four-list final search. Section 10 extends the recursive merges to more than two children.

Theorem 1.10 (Worst-case quantum SVP)**.** *With coherent indexed access to stored classical arrays, a quantum algorithm solves exact SVP on every n-dimensional input lattice with success probability at least $2/3$, time T and space S satisfying*

$$\mathrm{T},\ \mathrm{S} \le 2^{\tau n+o(n)}, \qquad \tau = 0.2468\ldots. \tag{1.13}$$

Theorem 11.1 uses several list types. Of the representations that share their first two list entries, it keeps one. The proof follows the Gaussian weight of a shortest vector backwards through the merges to bound how much this selection removes. The preparation uses the centered lattice-point bound from Proposition 1.1 and the spherical-code bound from [Ope26, Chapter 2, Remark 1.7] for separated sets in projected shells.

1.2.6 Applications

Using existing reductions and straightforward extensions, we derive the following consequences from our results. For approximate-SVP with large approximation factors, this improves the kissing-constant-conditional $2^{0.307n+o(n)}$ time complexity of [AMS26]. For approximate-CVP, for large approximation factors this improves upon the $2^{n/2+o(n)}$ time for 3.67-approximate CVP of Hhan [Hha26a, DRS14]. For bounded distance decoding (CVP with a distance guarantee below $\lambda_1(\mathcal{L})/2$), the previous-best time complexity was $2^{n/2+o(n)}$ based on [ADRS15, Corollary 7.4].

Corollary 1.11 (Approximate SVP and CVP)**.** *For every $\gamma \ge 1$, worst-case γ-SVP and γ-CVP have randomized algorithms with success probability at least $2/3$ and time and space*

$$\mathrm{T} = (3/2)^{n/2+o(n)}2^{O(n/\gamma^2)}, \qquad \mathrm{S} = (4/3)^{n/2+o(n)}2^{O(n/\gamma^2)}. \tag{1.14}$$

Corollary 1.12 (Bounded-distance decoding)**.** *For $0 \le \gamma \le \sqrt{\pi/(6e)} \approx 0.4389$, the closest vector to a target $\mathbf{t}$ can be found with probability at least $2/3$ whenever $\mathrm{dist}(\mathbf{t},\mathcal{L}) \le \gamma \cdot \lambda_1(\mathcal{L})$, with costs*

$$\mathrm{T} = (3/2)^{n/2+o(n)}, \qquad \mathrm{S} = (4/3)^{n/2+o(n)}. \tag{1.15}$$

Using an efficient discrete Gaussian sampler, one can further efficiently estimate both the Gaussian mass $\rho_s(\mathcal{L})$ for any $s > 0$, as well as the smoothing parameter $\eta_\varepsilon(\mathcal{L})$ for $0 < \varepsilon \le 1/2$ up to arbitrary precision δ, with polynomial overhead in $1/\delta$. This improves the previous best $2^{n/2+o(n)}$ time for the Gaussian mass [Hha26a, Ste16] and smoothing parameter [ADRS15, Sec. 6]. Obtaining these costs with a polylogarithmic dependence on ε, similar to [CDLP13], is left for future work.

Corollary 1.13 (Gaussian mass estimation)**.** *Given a basis of $\mathcal{L}$, a Gaussian width $s > 0$, and a fixed accuracy parameter $0 < \delta < 1/2$, a randomized algorithm returns $\widehat{\rho} > 0$ such that $1-\delta \le \widehat{\rho}/\rho_s(\mathcal{L}) \le 1+\delta$ with probability at least $2/3$, in time and space*

$$\mathrm{T} = (3/2)^{n/2+o(n)}\,\mathrm{poly}(1/\delta), \qquad \mathrm{S} = (4/3)^{n/2+o(n)}\,\mathrm{poly}(1/\delta). \tag{1.16}$$

Classical SVP algorithm	Lattice model	Time	Space
Herold–Kirshanova–Laarhoven [HKL18]	Heuristic	0.377	**0.172**
Chailloux–Loyer [CL23]	Heuristic	0.338	0.189
Becker–Ducas–Gama–Laarhoven [BDGL16]	Heuristic	**0.292**	0.208
Pouly–Shen [PS26]	Haar-random	0.633	0.500
Zhou–Wang–Wang [ZWW26]	Haar-random	**0.292**	**0.208**
Doulgerakis–Laarhoven [DL26]	Haar-random	**0.292**	**0.208**
Corollary 1.9 – Haar-random application	Haar-random	**0.292**	**0.208**
Corollary 1.8 – Kissing constant assumption	Conditional	**0.292**	**0.208**
Aggarwal–Dadush–Regev–Stephens-Davidowitz [ADRS15]	Worst-case	1.000	1.000
Gao–Feng–Hu [GFH26b]	Worst-case	0.731	0.500
Hhan [Hha26b]	Worst-case	0.604	0.500
Gao–Feng–Hu [GFH26a]	Worst-case	0.560	0.500
Hhan [Hha26a]	Worst-case	0.500	0.500
Corollary 1.6 – Minimum space	Worst-case	0.423	**0.208**
Theorem 1.7 – Fast matrix multiplication	Worst-case	**0.409**	0.259

Table 1: Classical SVP bounds. The time and space columns give c in $2^{cn+o(n)}$, rounded to three decimals. Boldface shows the smallest exponents for each lattice model. Corollary 1.8 follows from the centered point-count bound implied by its assumption on the general spherical kissing number. The same point-count bound holds for typical Haar-random lattices, giving Corollary 1.9.

Quantum SVP algorithm	Lattice model	Time	Space
Kirshanova–Mårtensson–Postlethwaite–Moulik [KMPM19]	Heuristic	0.318	**0.172**
Engelberts–Chen–Gilani–van Hoof–Jeffery–de Wolf [ECG+25]	Heuristic	0.285	0.189
Laarhoven [Laa16a]	Heuristic	0.265	0.208
Heiser [Hei21]	Heuristic	0.257	0.208
Chailloux–Loyer [CL21]	Heuristic	0.257	0.208
Bonnetain–Chailloux–Schrottenloher–Shen [BCSS23]	Heuristic	**0.256**	0.208
Corollary 1.9 – Haar-random application	Haar-random	**0.265**	**0.265**
Corollary 1.8 – Kissing constant assumption	Conditional	**0.265**	**0.265**
Aggarwal–Chen–Kumar–Shen [ACKS25]	Worst-case	0.835	0.500
Hhan [Hha26b]	Worst-case	0.541	0.500
Theorem 5.1 – Single-shell minimum time	Worst-case	0.359	0.302
Theorem 6.1 – Smaller fixed radius	Worst-case	0.317	0.265
Theorem 8.1 – Variable shells	Worst-case	0.294	0.294
Theorem 9.1 – Gaussian splitting	Worst-case	0.265	0.265
Theorem 10.1 – Higher-arity merges	Worst-case	0.248	0.248
Theorem 11.1 – Multi-list construction	Worst-case	**0.247**	**0.247**

Table 2: Quantum SVP bounds. The time and space columns give c in $2^{cn+o(n)}$, rounded to three decimals. The unconditional multi-list bound applies to every lattice family in the table. The algorithms make different assumptions about quantum memory and access.

Discrete Gaussian sampler	Lattice model	Parameter s	Time	Space
Doulgerakis–Laarhoven [DL26]	Haar-random	Any s	**0.292**	**0.208**
Klein [Kle00]	Worst-case	Large s	*(poly)*	*(poly)*
Aggarwal–Dadush–Regev–Stephens-D. [ADRS15]	Worst-case	Any s	1.000	1.000
Aggarwal–Dadush–Regev–Stephens-D. [ADRS15]	Worst-case	Moderate s	0.500	0.500
Proposition 1.3 – Restriction	Worst-case	Any s	0.500	0.500
Theorem 1.4 – Discrete Gaussian sampling	Worst-case	Any s	**0.292**	**0.208**

Table 3: Discrete Gaussian sampling with error at most $2^{-\Theta(n)}$ in total variation. The time and space columns give c in $2^{cn+o(n)}$. The algorithms of [DL26, ADRS15] give exponentially many samples at these costs.

Corollary 1.14 (Smoothing parameter estimation)**.** *Given a basis of $\mathcal{L}$, a smoothing bound $0 < \varepsilon \le 1/2$, and an accuracy parameter $0 < \delta \le 1$, a randomized algorithm finds $\widehat{\eta}$ such that $1 - \delta \le \widehat{\eta}/\eta_\varepsilon(\mathcal{L}) \le 1 + \delta$ with probability at least $2/3$, in time and space*

$$\mathrm{T} = (3/2)^{n/2+o(n)} \operatorname{poly}(1/\varepsilon, 1/\delta), \qquad \mathrm{S} = (4/3)^{n/2+o(n)} \operatorname{poly}(1/\varepsilon, 1/\delta). \tag{1.17}$$

1.3 Techniques

The superlattice chain. A lattice sieve repeatedly combines vectors into smaller spherical shells. In the fixed-radius algorithms, we instead move through successively sparser lattices while keeping the shell radius fixed. Following earlier uses of superlattice chains [BGJ14, ADRS15, DK16, ALS21, ACKS25, DEL25], we use a chain with indices

$$\mathcal{L} \subset \Gamma_0 \subset \Gamma_1 \subset \cdots \subset \Gamma_D, \qquad [\Gamma_0 : \mathcal{L}] = p, \quad [\Gamma_{d+1} : \Gamma_d] = q. \tag{1.18}$$

The random prime-index extension Γ_0 makes the counts of representations vary little with the shift. The basis of Γ_D has small rounding cells, so we can sample list entries there and combine them down the chain. Sections 5 and 6 use subexponential q. Section 11 uses $q = 2^{\gamma n+o(n)}$ and logarithmic depth to control the Gaussian distributions for merges of several lists.

The basic SVP algorithm uses the same target expected count for each point in a shell near $\lambda_1(\mathcal{L})$. The DGS algorithm uses a shell chosen for the target output length and gives its points Gaussian weights. Both algorithms build the chain by successively halving basis vectors after constructing Γ_0.

Initial sampling. We draw uniformly from a slightly enlarged shell and apply Babai's nearest-plane rounding [Bab86]. We discard points outside the original shell. Each point in that shell has an equal-volume rounding cell contained in the enlarged shell, so each has the same probability of being drawn. For DGS we then accept with a probability proportional to the desired Gaussian weight.

Averaging over random shifts. To produce a list on $\Gamma_d + \mathbf{c}$, draw a fresh uniform shift $\mathbf{u}$ modulo Γ_{d+1}. Generate the two child lists on $\Gamma_{d+1} + \mathbf{u}$ and $\Gamma_{d+1} + \mathbf{u} - \mathbf{c}$, using independent randomness given the two translates. A difference belongs to $\Gamma_d + \mathbf{c}$ whenever its lattice part belongs to Γ_d. For any nonnegative measurable f, averaging a sum over a uniform shift $\mathbf{u}$ of a lattice Γ gives

$$\mathbb{E}_{\mathbf{u}}\left[\det(\Gamma) \sum_{\mathbf{x} \in \Gamma + \mathbf{u}} f(\mathbf{x})\right] = \int_{\mathbb{R}^n} f(\mathbf{x})\, d\mathbf{x}. \tag{1.19}$$

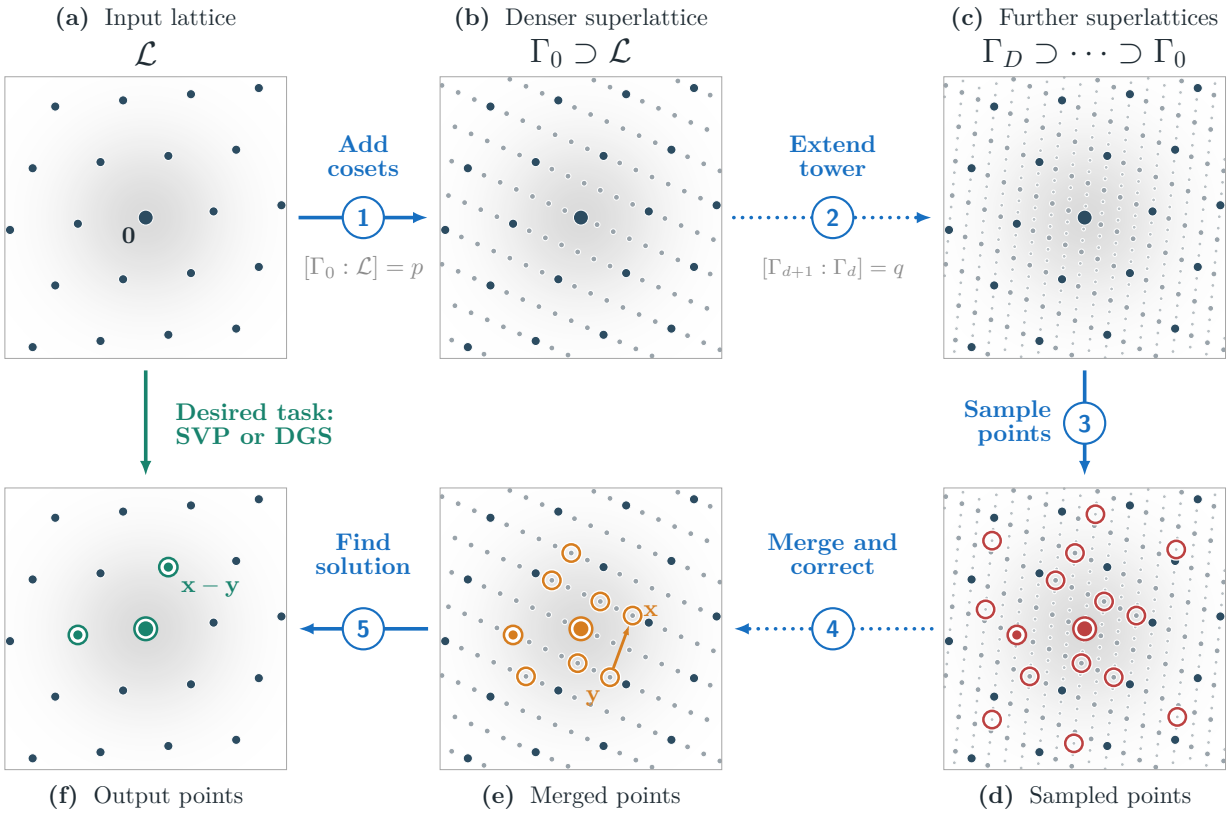


Figure 1: The superlattice construction for SVP and DGS (Sections 5 and 4). We extend $\mathcal{L}$ to Γ_0 and then to Γ_D, where we generate points. We combine the lists so that their differences lie in the required cosets, working back through Γ_0 to $\mathcal{L}$. Red and orange rings show points in the lists. Green rings show possible outputs, including $\mathbf{0}$ for DGS. Related constructions appear in [BGJ14, ADRS15, ALS21].

This converts the sum over lattice representations of a fixed difference into an integral. The shells stay centered at the origin as the translates move [PS09, NV08, MV10, HPS11]. The index of Γ_0 controls the fluctuations around this average.

A midpoint shell balances the representation counts. Let H be the shell of radius $\lambda_1(\mathcal{L})$. Write $\mathbf{Z} = \mathbf{X} - \mathbf{Y}$ and $\mathbf{W} = (\mathbf{X} + \mathbf{Y})/2$ for the difference and midpoint of two entries. When $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ all lie in H, the identity

$$\|\mathbf{W}\|^2 = \frac{\|\mathbf{X}\|^2 + \|\mathbf{Y}\|^2}{2} - \frac{\|\mathbf{Z}\|^2}{4} \approx \frac{3}{4}\lambda_1(\mathcal{L})^2. \tag{1.20}$$

places the midpoint near radius $\sqrt{3/4}\lambda_1(\mathcal{L})$. We require it to lie in a shell A at that radius, narrower than H. To see the effect, consider one proposal from each child, independent given the two translates. Each proposal assigns the same probability to every eligible point in its shell, independent of the shift, and may also fail. Write $\mathcal{P} = (\Gamma_d + \mathbf{c}) \cap H$. After averaging over the shift, the two probabilities below are given by volumes. We assume their conditioning events have positive

probability.

$$\Pr[\mathbf{Z} = \mathbf{z} \mid \mathbf{Z} \in \mathcal{P}] = \frac{\mathrm{vol}(H \cap (H + \mathbf{z}))}{\sum_{\mathbf{v} \in \mathcal{P}} \mathrm{vol}(H \cap (H + \mathbf{v}))}, \tag{1.21}$$

$$\Pr[\mathbf{Z} = \mathbf{z} \mid \mathbf{Z} \in \mathcal{P},\ \mathbf{W} \in A] = \frac{\mathrm{vol}(A \cap (H - \mathbf{z}/2) \cap (H + \mathbf{z}/2))}{\sum_{\mathbf{v} \in \mathcal{P}} \mathrm{vol}(A \cap (H - \mathbf{v}/2) \cap (H + \mathbf{v}/2))} = \frac{1 + o(1)}{|\mathcal{P}|}. \tag{1.22}$$

For every $\mathbf{z} \in \mathcal{P}$, a uniform midpoint in A gives two endpoints in H with probability $1 - o(1)$. The volume in the second numerator is therefore $(1 - o(1))\,\mathrm{vol}(A)$ uniformly in $\mathbf{z}$. Among pairs with difference in $\mathcal{P}$, requiring this midpoint test costs a subexponential factor.

Expected counts and pairs of entries. For DGS we prescribe Gaussian expected counts. Let ϕ_σ be the density of $N(\mathbf{0}, \sigma^2\mathbf{I})$. The weights of two child entries factor into a weight for the difference and a weight for the midpoint

$$\phi_\sigma(\mathbf{x})\phi_\sigma(\mathbf{y}) = \phi_{\sqrt{2}\sigma}(\mathbf{x} - \mathbf{y})\phi_{\sigma/\sqrt{2}}\left(\frac{\mathbf{x} + \mathbf{y}}{2}\right). \tag{1.23}$$

Averaging over the shift integrates the midpoint factor. An acceptance probability then adjusts the remaining factor to the desired weight. We choose the widths so that, for each allowed difference, the two children lie in the common shell with high probability. These shell tests also enforce the required midpoint cutoff.

Let $A_{\mathbf{z}}$ count entries equal to $\mathbf{z}$ and let $\ell_d(\mathbf{z})$ be the target expected count. In the binary SVP algorithms this target is constant across the prescribed shell. In DGS it follows the Gaussian weights. On chains with the required counting bounds, and before imposing the size and work limits, we show

$$(1 - o(1))\ell_d(\mathbf{z}) \le \mathbb{E}\left[A_{\mathbf{z}}\right] \le \ell_d(\mathbf{z}), \tag{1.24}$$

$$\mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{z}'} - \mathbf{1}_{\mathbf{z}=\mathbf{z}'})] \le (1 + o(1))\ell_d(\mathbf{z})\ell_d(\mathbf{z}'). \tag{1.25}$$

The first line gives the expected count. The second counts ordered pairs of distinct list positions with the prescribed values, even if the two values agree. The proof separates pairs of representations by which child entries they share.

Accelerating the merges. Equal-radius useful pairs have angle near $\pi/3$. BDGL product-code filters find them without scanning all pairs [BDGL16, Laa16b, GFH26a]. Quantum reporting and minimum finding further accelerate the indexed searches [BBHT98, vAGN24, DH96]. We mostly use these as black-box algorithms for reporting suitable pairs of vectors faster.

The final step for binary SVP. We generate two final lists on the same translate of Γ_0, independently given the translate. The final search compares differences that lie in $\mathcal{L}$ and returns the shortest nonzero one it finds. The midpoint test is used only in the internal merges.

Where the basic worst-case SVP overhead comes from. With a larger point-count exponent β, our preparation chooses a denser Γ_0 to control the fluctuations in the representation counts. A fixed shortest vector then has fewer representations for a given root-list size. Put $c = \frac{1}{2}\log_2(4/3)$. Lists of size $2^{\alpha n + o(n)}$ give an expected $2^{(2\alpha - \beta - c)n + o(n)}$ representations of a fixed shortest vector. We

can compensate by repeating trials or enlarging the lists. The final filtered search costs $2^{f(\alpha)n+o(n)}$ per trial, with

$$f(\alpha) = -\frac{1}{2}\log_2\left(\frac{4\cdot 2^{-2\alpha}-1}{3}\right), \qquad t_{\text{single}}(\alpha) = f(\alpha)+\beta+c-2\alpha. \tag{1.26}$$

For $\beta = c$, the usual list exponent $\alpha = c$ gives time exponent $0.2924\ldots$, equivalent to the best heuristic SVP algorithm [BDGL16]. Worst-case point counts give $\beta = 0.3955\ldots$. The minimum-time choice enlarges the lists to $\alpha = (\beta + c)/2 \approx 0.3016$, eliminating exponential repetitions. The larger lists increase the cost of the final search to $f(\alpha) \approx 0.4386$. The smaller-radius construction reduces both the required list size and this final cost. It keeps one list radius throughout the chain, but distinguishes the internal midpoint shell from the smaller midpoint region representing a final shortest difference. Preparing the chain for both regions and choosing the expected list length using the latter gives the improved classical bound. The quantum multi-list construction uses a different final tuple search.

Choosing a DGS output. For DGS the final merge combines two lists on a translate of Γ_0 and keeps differences in the input lattice $\mathcal{L}$ and a chosen shell S_j. Matching quotient labels identify differences in $\mathcal{L}$. Acceptance probabilities give target expected counts

$$\ell_{\text{out}}(\mathbf{z}) = \frac{\rho_s(\mathbf{z})}{4U}\,\mathbf{1}_{S_j}(\mathbf{z}), \tag{1.27}$$

Here U bounds the unknown Gaussian mass $\rho_s(\mathcal{L})$. Write L for the final list. The target counts sum to at most $1/4$, so $\mathbb{E}|L| \le 1/4$. By Markov's inequality L is empty with probability at least $3/4$.

If we select a uniform position from L, the probability of choosing $\mathbf{z}$ is $\mathbb{E}[A_{\mathbf{z}}/|L|]$, where $A_{\mathbf{z}}$ counts entries in L equal to $\mathbf{z}$ and the ratio is zero for an empty list. For example, suppose the list is $[\mathbf{x}]$ with probability $1/2$, $[\mathbf{y},\mathbf{y}]$ with probability $1/4$, and empty otherwise. Both points have expected count $1/2$, but are selected with probabilities $1/2$ and $1/4$. To measure this difference, we run the algorithm with one entry equal to $\mathbf{z}$ fixed in advance. We sample its midpoint from a continuous Gaussian and generate the two required child entries from it, then repeat this step down the tree. The other entries are generated as in an ordinary run.

Section 4 shows that the ordinary selection probability can be written in terms of a selection test with success probability $\varpi_{\mathbf{z}}$ as follows.

$$\Pr[\mathsf{Candidate}(j,U) = \mathbf{z}] = \ell_{\text{out}}(\mathbf{z})\varpi_{\mathbf{z}}. \tag{1.28}$$

The selection test succeeds with probability $1/|L|$ when the run with a fixed entry returns a list L. A failed run counts as a failed test. For valid U, the moment and work bounds give $\varpi_{\mathbf{z}} \ge 1/2$. We choose i uniformly from $\{0,\ldots,m-1\}$ and accept $\mathbf{z}$ only if all i independent tests fail. The resulting acceptance probability is

$$\frac{1}{m}\sum_{i=0}^{m-1}(1-\varpi_{\mathbf{z}})^i = \frac{1-(1-\varpi_{\mathbf{z}})^m}{m\varpi_{\mathbf{z}}}. \tag{1.29}$$

This cancels the unknown selection factor, up to relative error at most 2^{-m}. A related correction for unknown probabilities appears in [ADRS15, Claim 5.2].

Choosing the output shell. The DGS construction works one radial bin $S_0, \dots$ at a time. We choose j uniformly among $J = n + m$ bins, rather than according to the (unknown) Gaussian masses of these spherical shells. Every bin uses the same normalization $4U$, so, for $\mathbf{z} \in S_j$,

$$\Pr[\mathsf{Attempt} = \mathbf{z}] = \frac{\rho_s(\mathbf{z})}{4JUm}(1 - (1 - \varpi_{\mathbf{z}})^m). \tag{1.30}$$

The probability that an attempt succeeds from a particular bin is proportional to its Gaussian mass, up to the residual error in the last factor. Decreasing powers of two supply guesses for U; once a valid guess is within a factor two of the true mass, polynomially many attempts suffice. With negligible probability of reaching an invalid guess, $m = \kappa + 3$ gives error at most $2^{-\kappa}$. The weighted construction and correction fit the $(3/2)^{n/2+o(n)}\,\mathrm{poly}(\kappa)$ bound without a worst-case kissing-number overhead.

1.4 Open problems

Classical worst-case SVP in time $(3/2)^{n/2+o(n)}$. The exponent "0.292" keeps appearing in lattice algorithms: as the heuristic SVP exponent [BDGL16]; as the Haar-random exponent for SVP [DL26, ZWW26], CVP, and DGS [DL26]; and now as the conditional SVP exponent; the worst-case DGS exponent; the leading approximate-SVP/CVP exponents; and the leading terms for estimating the Gaussian mass and the smoothing parameter. While worst-case CVP may be out of reach [ADS15, ABGS21], a natural question to ask is: can classical worst-case SVP be solved in time $(3/2)^{n/2+o(n)}$? The quantum multi-list algorithm already runs below this exponent.

The main obstacle may lie in excluding the existence of extremal lattices with high kissing constants. Suppose a lattice exists with $2^{\beta n+o(n)}$ shortest vectors. Perturbing the space, one pair of points becomes the unique SVP solution. Unless the macroscopic structure of the lattice can be used to investigate the microscopic scale of the lattice at norm $\lambda_1(\mathcal{L})$, this suggests a natural conjectured barrier for solving worst-case SVP. If true, it would first require additional breakthroughs on the "kissing constant side" before we can get worst-case SVP time complexities as low as $(3/2)^{n/2+o(n)}$.

Conjecture 1.15 (Worst-case SVP lower bound)**.** *Let $\beta^* \le \frac{1}{2}\log_2(2e/\pi) = 0.3955\dots$ denote the minimum value of β such that, for all lattices $\mathcal{L}$ and all $r \ge 1$, we have $N_{\mathcal{L}}(r\lambda_1(\mathcal{L})) \le 2^{\beta n+o(n)} r^n$. Then no classical algorithm can solve worst-case SVP in time below $2^{\beta^* n+o(n)}$.*

For every admissible β and α in Theorem 1.5, the classical time exponent satisfies $t(\alpha) \ge \beta$. The quantum time exponent in Theorem 1.10 lies below the current best upper bound on β^*; the conjecture concerns only classical algorithms.

Classical tuple merging and simpler quantum constructions. Section 11 establishes a quantum tuple construction using Gaussian splitting kernels, several list types, and selection by seed identifiers. It remains open whether related geometry can improve the classical bound in Section 6, or whether a smaller type graph can give comparable quantum costs with a shorter analysis. The main classical difficulty is to bound the full candidate tree: quantum reporting uses a square root of its size times the output count, whereas classical reporting pays for every candidate. Extensions may be compared with heuristic tuple-sieving tradeoffs [BLS16, HKL18, KMPM19, CL23, ECG+25].

1.5 Concurrent work

The restriction result of Proposition 1.3 previously appeared in a manuscript of Kim [Kim26], which is subsumed by this work. Related concurrent contributions include the similar restriction

approach of [GFH26b] and their merging improvement [GFH26a]; the Hessian-based SVP methods of Hhan [Hha26b, Hha26a]; the Haar-random SVP/DGS methods of [DL26, ZWW26]; the kissing-constant-conditional analyses of [AMS26]; and the kissing constant improvements of [Ope26, Kom26].

Simultaneously to the publication of (the first version of) this manuscript, work by Gao–Han–Hu [GHH26] was published, using almost equivalent ideas and techniques to derive similar results. Given the substantial overlap, we briefly summarize some of the main observed differences between our work and theirs:

- They obtain a slightly worse optimized worst-case SVP constant of $0.4129\ldots$. Without the improved lattice point count bounds of Proposition 1.1, and instead using the Kabatiansky–Levenshtein bound 0.401 [KL78], our results in Section 6 would converge to the exact same constant $0.4129\ldots$, rather than $0.4088\ldots$. This difference is purely a matter of our sharper point count bounds. With the faster matrix multiplication of Section 7, an algorithmic difference compared to [GHH26], we further reduce the constant to $0.4085\ldots$.
- They do not consider the quantum SVP improvements described in this work.
- Similar to Doulgerakis–Laarhoven [DL26], they show that above the smoothing parameter, with the worst-case discrete Gaussian sampling costs of Theorem 1.4 one can actually obtain an exponentially large batch of discrete Gaussian samples, at no additional asymptotic costs (time or space) compared to just one sample.

1.6 Organization

Section 2 collects the preliminaries. Section 3 analyzes smoothing extensions and proves the restriction sampler. Section 4 gives the faster sampler by Gaussian-weighted merges. Sections 5 and 6 prove the SVP tradeoffs for two fixed shell radii, and Section 7 further improves the classical SVP time exponent using fast matrix multiplication. The quantum refinements use variable shells and asymmetric access in Section 8, Gaussian splitting in Section 9, higher-arity merges in Section 10, and selection by seed identifiers in Section 11. Section 12 derives the applications of the DGS sampler. Appendix A proves Proposition 1.1. Appendix B gives the parameter choices and interval checks for the selected quantum construction.

2 Preliminaries

We write $[n] = \{1, \dots, n\}$ and $x_+ = \max\{x, 0\}$. All logarithms are base two unless written ln. We use d_{TV} for total variation distance.

Parameters may depend on the dimension n and the input length. The stated bounds absorb the cost of using polynomially many bits of precision in arithmetic operations. Asymptotic bounds are uniform over the lattices and parameters in their stated ranges.

2.1 Lattice background

An n-dimensional lattice $\mathcal{L} \subset \mathbb{R}^n$ consists of the integer linear combinations of a basis $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$. We write $\mathcal{L} = \mathcal{L}(\mathbf{B}) = \mathbf{B}\mathbb{Z}^n$ and $\det \mathcal{L} = |\det \mathbf{B}|$. All lattices have full rank in their ambient space. In several proofs we rescale the lattice to have determinant one. The dual lattice is

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

If $\mathcal{L} \subseteq \Lambda$, then Λ is a superlattice of $\mathcal{L}$ and $\mathcal{L}$ is a sublattice of Λ. The index $[\Lambda : \mathcal{L}] = \det \mathcal{L} / \det \Lambda$ counts the cosets of $\mathcal{L}$ in Λ.

The i-th successive minimum $\lambda_i(\mathcal{L})$ is the smallest radius of a ball centered at the origin that contains i linearly independent lattice vectors. In particular, $\lambda_1(\mathcal{L})$ is the length of a shortest nonzero vector. For a target $\mathbf{y} \in \mathbb{R}^n$, its distance to the lattice is

$$\operatorname{dist}(\mathbf{y}, \mathcal{L}) = \min_{\mathbf{x} \in \mathcal{L}} \|\mathbf{y} - \mathbf{x}\|. \tag{2.1}$$

The covering radius is $\mu(\mathcal{L}) = \sup_{\mathbf{y} \in \mathbb{R}^n} \operatorname{dist}(\mathbf{y}, \mathcal{L})$. For a basis $(\mathbf{b}_i)$, we write $(\widetilde{\mathbf{b}}_i)$ for its Gram–Schmidt orthogonalization. Banaszczyk's covering-radius transference inequality [Ban93] gives

$$\mu(\mathcal{L})\lambda_1(\mathcal{L}^*) \le n/2. \tag{2.2}$$

Lattice problems. Given a basis of $\mathcal{L}$, the shortest vector problem (SVP) asks for a nonzero lattice vector of length $\lambda_1(\mathcal{L})$. The closest vector problem (CVP) also takes a target $\mathbf{y}$ and asks for a lattice point at distance $\operatorname{dist}(\mathbf{y}, \mathcal{L})$ from it. In the γ-approximate versions, the returned length or distance may be up to $\gamma \ge 1$ times the optimum. Bounded-distance decoding (BDD) is CVP with the promise $\operatorname{dist}(\mathbf{y}, \mathcal{L}) < \lambda_1(\mathcal{L})/2$.

Counting short lattice vectors. For a lattice $\mathcal{L}$, define

$$N_{\mathcal{L}}(R) = |\{\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\} : \|\mathbf{v}\| \le R\}|.$$

For a constant $\beta \ge 0$, write $b = 2^\beta$. We consider lattice families satisfying

$$N_{\mathcal{L}}(r \cdot \lambda_1(\mathcal{L})) \le A_n b^n r^n \qquad (r \ge 1), \tag{2.3}$$

Here $A_n = 2^{o(n)}$ is uniform over the lattices and radii in the family. When constructing a superlattice, we may replace b by $2^\beta A_n^{1/n}$ and use $A_n = 1$ in (2.3). This changes $\log_2 b$ by $o(1)$. For the Haar-random lattices considered below, $A_n = n^2$.

Let τ_n be the maximum size of a set of unit vectors in $\mathbb{R}^n$ with pairwise inner products at most $1/2$. For related lattice-point bounds, see [PS09, Lemma 3] and [ADRS15, Lemma 4.1].

Lemma 2.1 (Kissing-number point count)**.** *There is an absolute constant $C > 0$ such that, for every $n \geq 4$, every n-dimensional lattice $\mathcal{L}$, and every $r \geq 1$,*

$$1 + N_{\mathcal{L}}(r\lambda_1(\mathcal{L})) \leq Cn^3\tau_n r^n. \tag{2.4}$$

Consequently, $\tau_n \leq 2^{\beta n + o(n)}$ implies (2.3) uniformly in the lattice and the radius.

Proof. Normalize the lattice to have $\lambda_1(\mathcal{L}) = 1$. We first show that every unit ball contains at most $\tau_n + 1$ lattice points. Take distinct lattice points $\mathbf{x}, \mathbf{y}$ in the unit ball centered at $\mathbf{c}$, and put $\mathbf{u} = \mathbf{x} - \mathbf{c}$ and $\mathbf{v} = \mathbf{y} - \mathbf{c}$. Since $\|\mathbf{u} - \mathbf{v}\| \geq 1$ and both lengths are at most one,

$$2\langle \mathbf{u}, \mathbf{v}\rangle \leq \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2 - 1 \leq \min\{\|\mathbf{u}\|^2, \|\mathbf{v}\|^2\} \leq \|\mathbf{u}\|\|\mathbf{v}\|. \tag{2.5}$$

The nonzero centered vectors therefore give a spherical code with pairwise inner products at most $1/2$ after normalization. The center contributes at most one additional lattice point.

For $1 \leq r \leq n$, cover $[1, r^2]$ by at most $nr^2 + 1$ intervals $[a, a + 1/n]$, where $a = 1, 1 + 1/n, \ldots$ and $a \leq r^2$. For each interval, define the set of shell points $S_a = \{\mathbf{x} \in \mathcal{L} : a \leq \|\mathbf{x}\|^2 \leq a + 1/n\}$ and let $t = \sqrt{a + 1/n - 1}$. Let $\mathbf{u}$ be a point on the unit sphere $\mathbb{S}^{n-1}$. Then every $\mathbf{x} \in S_a$ for which $\langle \mathbf{u}, \mathbf{x}\rangle \geq t$ lies in the unit ball centered at $t\mathbf{u}$, because

$$\|\mathbf{x} - t\mathbf{u}\|^2 = \|\mathbf{x}\|^2 + t^2 - 2t\langle \mathbf{u}, \mathbf{x}\rangle \leq a + 1/n - t^2 = 1.$$

Hence, for every choice of $\mathbf{u}$, at most $\tau_n + 1$ points of S_a are selected.

On the other hand, fix any $\mathbf{x} \in S_a$. The directions $\mathbf{u}$ that select $\mathbf{x}$ form the spherical cap with $\langle \mathbf{u}, \mathbf{x}\rangle \geq t$. Consider uniform random $\mathbf{u}$. Since $1 - \frac{t^2}{\|\mathbf{x}\|^2} \geq \frac{1-1/n}{a} \geq \frac{1-1/n}{r^2}$, we have $\Pr_{\mathbf{u}}[\langle \mathbf{u}, \mathbf{x}\rangle \geq t] \geq cn^{-1/2}r^{-(n-1)}$ for an absolute constant $c > 0$ using the cap-area formula in [BM07, Section 3-C]. Summing these probabilities over $\mathbf{x} \in S_a$ yields

$$cn^{-1/2}r^{-(n-1)}|S_a| \leq \mathbb{E}_{\mathbf{u}}\#\{\mathbf{x} \in S_a : \langle \mathbf{u}, \mathbf{x}\rangle \geq t\} \leq \tau_n + 1.$$

Therefore $|S_a| \leq C\sqrt{n}\, r^{n-1}\tau_n$. Summing over the at most $nr^2 + 1$ shells, and using $r \leq n$, gives

$$N_{\mathcal{L}}(r) \leq \sum_a |S_a| \leq Cn^{3/2}r^{n+1}\tau_n \leq Cn^{5/2}r^n\tau_n.$$

For $r \geq n$, integrating the unit-ball count over centers in $B(\mathbf{0}, r + 1)$ gives

$$1 + N_{\mathcal{L}}(r) \leq (\tau_n + 1)(r + 1)^n \leq C\tau_n r^n.$$

Thus choosing an appropriate C proves the claim. □

Lemma 2.1 and the spherical-code bound of [Ope26, Chapter 2, Theorem 1.2 and Remark 1.7] give (2.3) with $\beta_{\text{sph}} = 0.39661$. Proposition 1.1 gives the sharper exponent $\beta = \frac{1}{2}\log_2(2e/\pi)$ for balls centered at the origin. Both bounds have a universal subexponential factor A_n. The sharper bound concerns all centered balls of radius at least $\lambda_1(\mathcal{L})$, while the lattice kissing number counts vectors of exactly that length.

2.2 Discrete and continuous Gaussians

For $s > 0$, define $\rho_s(\mathbf{x}) = e^{-\pi\|\mathbf{x}\|^2/s^2}$. For a discrete set $A \subset \mathbb{R}^n$, write $\rho_s(A) = \sum_{\mathbf{x}\in A} \rho_s(\mathbf{x})$. Whenever $0 < \rho_s(A) < \infty$, the discrete Gaussian on A is the probability distribution $D_{A,s}(\mathbf{x}) = \rho_s(\mathbf{x})/\rho_s(A)$. The Poisson summation formula gives

$$\rho_s(\mathcal{L}) = \frac{s^n}{\det \mathcal{L}} \rho_{1/s}(\mathcal{L}^*). \tag{2.6}$$

For $\varepsilon > 0$, the smoothing parameter of a lattice $\mathcal{L}$ is

$$\eta_\varepsilon(\mathcal{L}) = \inf\{t > 0 : \rho_{1/t}(\mathcal{L}^* \setminus \{\mathbf{0}\}) \le \varepsilon\}. \tag{2.7}$$

For $\alpha > 1$ and $t > 0$, the problem α-GapSPP$_\varepsilon$ asks to distinguish $\eta_\varepsilon(\mathcal{L}) \le t$ from $\eta_\varepsilon(\mathcal{L}) > \alpha t$.

We use the following relation between smoothing and successive minima.

Lemma 2.2 (Successive minima and the smoothing parameter [ALS21, Theorem 2.9])**.** *For every lattice $\mathcal{L}$ of rank $r \ge 20$, $\lambda_r(\mathcal{L}) \le \sqrt{r}\,\eta_{1/2}(\mathcal{L})$.*

Gaussian mass. The following standard Gaussian mass bounds are due to Banaszczyk [Ban93]; see also [ADS15, Lemmas 2.5–2.6]. For $s > 0$, $\gamma \ge 1$, $\mathbf{t} \in \mathbb{R}^n$, and $d = \operatorname{dist}(\mathbf{t}, \mathcal{L})$, we have

$$\rho_{\gamma s}(\mathcal{L}) \le \gamma^n \rho_s(\mathcal{L}), \qquad e^{-\pi d^2/s^2}\rho_s(\mathcal{L}) \le \rho_s(\mathcal{L} - \mathbf{t}) \le \rho_s(\mathcal{L}). \tag{2.8}$$

Lemma 2.3 (Gaussian tails and change of width)**.** *For $\mathbf{Y} \sim D_{\mathcal{L}-\mathbf{t},s}$ and $r > 0$,*

$$\Pr[\|\mathbf{Y}\| > r] \le 2^{n/2} \exp\left(\frac{\pi d^2}{s^2} - \frac{\pi r^2}{2s^2}\right), \tag{2.9}$$

For $t > 0$ and $\alpha \ge 1$, the dual Gaussian mass satisfies

$$\rho_{1/(\alpha t)}(\mathcal{L}^* \setminus \{\mathbf{0}\}) \le \rho_{1/t}(\mathcal{L}^* \setminus \{\mathbf{0}\})^{\alpha^2}. \tag{2.10}$$

Proof. Apply Markov's inequality to $e^{\pi\|\mathbf{Y}\|^2/(2s^2)}$. Its expectation is $\rho_{\sqrt{2}s}(\mathcal{L} - \mathbf{t})/\rho_s(\mathcal{L} - \mathbf{t})$, which is at most $2^{n/2}e^{\pi d^2/s^2}$ by the preceding bounds. For (2.10), increasing t to αt raises every summand in the dual mass to the power $\alpha^2 \ge 1$. The sum of these powers is at most the same power of the sum, as in [CDLP13, proof of Lemma 2.4]. □

Integrating (2.3) against the Gaussian and applying Stirling's estimate gives the following bound; see [ADRS15, Lemma 4.2] and [AS19, Theorem 3.1].

Lemma 2.4 (Gaussian estimate)**.** *For an n-dimensional full-rank lattice $\mathcal{L}$ with (2.3), and $s > 0$,*

$$\rho_s(\mathcal{L}) - 1 \le 3(n+1)A_n \left(\frac{b^2 s^2 n}{2\pi e \lambda_1(\mathcal{L})^2}\right)^{n/2}. \tag{2.11}$$

Comparison with the one-dimensional Gaussian integral gives

$$\rho_s(\mathbb{Z}) \le 1 + s \qquad \Longrightarrow \qquad \rho_s(\mathbf{B}\mathbb{Z}^n) \le (1 + s\|\mathbf{B}^{-1}\|_F)^n \tag{2.12}$$

where we use the inequality $\|\mathbf{B}\mathbf{z}\| \ge \|\mathbf{z}\|/\|\mathbf{B}^{-1}\|_F$ for a basis $\mathbf{B}$.

Continuous Gaussians. We write $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ for the normal distribution on $\mathbb{R}^n$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. For $\sigma > 0$, the centered Gaussian $N(\mathbf{0}, \sigma^2 \mathbf{I}_n)$ has density

$$\phi_\sigma(\mathbf{x}) = (2\pi\sigma^2)^{-n/2} e^{-\|\mathbf{x}\|^2/(2\sigma^2)}. \tag{2.13}$$

The two width conventions are related by $s = \sqrt{2\pi}\sigma$. We use the following identities to describe Gaussian vectors and their differences [GMPW20, Section 2].

Lemma 2.5 (Gaussian product)**.** *For every $\sigma, \tau > 0$ and $\mathbf{x}, \mathbf{z} \in \mathbb{R}^n$,*

$$\phi_\sigma(\mathbf{x})\phi_\sigma(\mathbf{x} - \mathbf{z}) = \phi_{\sqrt{2}\sigma}(\mathbf{z})\phi_{\sigma/\sqrt{2}}(\mathbf{x} - \mathbf{z}/2), \tag{2.14}$$

$$(\phi_\sigma * \phi_\tau)(\mathbf{z}) := \int \phi_\sigma(\mathbf{x})\phi_\tau(\mathbf{z} - \mathbf{x})\, d\mathbf{x} = \phi_{\sqrt{\sigma^2+\tau^2}}(\mathbf{z}). \tag{2.15}$$

Consequently, conditional on their difference being $\mathbf{z}$, two independent centered Gaussian vectors of covariance $\sigma^2\mathbf{I}$ have the form $\mathbf{W} + \mathbf{z}/2, \mathbf{W} - \mathbf{z}/2$, where $\mathbf{W} \sim N(\mathbf{0}, \sigma^2\mathbf{I}/2)$.

Periodization. For a full-rank lattice $\mathcal{L}$ and a nonnegative measurable function f, define its normalized periodization and second moment by

$$P_{\mathcal{L},f}(\mathbf{c}) = \det \mathcal{L} \sum_{\mathbf{x} \in \mathcal{L} + \mathbf{c}} f(\mathbf{x}), \qquad R_{\mathcal{L}}(f) = \mathbb{E}_{\mathbf{c} \bmod \mathcal{L}} P_{\mathcal{L},f}(\mathbf{c})^2. \tag{2.16}$$

Here we choose $\mathbf{c}$ uniformly modulo $\mathcal{L}$, equivalently uniformly in a fundamental domain. This measure on $\mathbb{R}^n/\mathcal{L}$ is invariant under translation and reflection. Thus $R_{\mathcal{L}}(f)$ is the mean square of $P_{\mathcal{L},f}$ over shifts.

The quotient integral formula [DADRT23, Section 2.3] gives

$$\mathbb{E}_{\mathbf{c} \bmod \mathcal{L}} \sum_{\mathbf{x} \in \mathcal{L} + \mathbf{c}} f(\mathbf{x}) = \frac{1}{\det \mathcal{L}} \int_{\mathbb{R}^n} f(\mathbf{x})\, d\mathbf{x}. \tag{2.17}$$

This can be shown as follows. Let F be a fundamental domain of $\mathcal{L}$. Since $\mathrm{vol}(F) = \det \mathcal{L}$ and the translates $F + \mathbf{v}$, $\mathbf{v} \in \mathcal{L}$, partition $\mathbb{R}^n$ up to sets of measure zero,

$$\begin{aligned}
\mathbb{E}_{\mathbf{c} \bmod \mathcal{L}} \sum_{\mathbf{x} \in \mathcal{L} + \mathbf{c}} f(\mathbf{x}) &= \frac{1}{\det \mathcal{L}} \int_F \sum_{\mathbf{v} \in \mathcal{L}} f(\mathbf{v} + \mathbf{c})\, d\mathbf{c} \\
&= \frac{1}{\det \mathcal{L}} \sum_{\mathbf{v} \in \mathcal{L}} \int_{F + \mathbf{v}} f(\mathbf{x})\, d\mathbf{x} \\
&= \frac{1}{\det \mathcal{L}} \int_{\mathbb{R}^n} f(\mathbf{x})\, d\mathbf{x}.
\end{aligned}$$

Multiplying by $\det \mathcal{L}$ gives the first identity below. For the second, expand the square and group pairs by their difference.

$$\mathbb{E}_{\mathbf{c}} P_{\mathcal{L},f}(\mathbf{c}) = \int_{\mathbb{R}^n} f(\mathbf{x})\, d\mathbf{x}, \qquad R_{\mathcal{L}}(f) = \det \mathcal{L} \sum_{\mathbf{u} \in \mathcal{L}} \int_{\mathbb{R}^n} f(\mathbf{w}) f(\mathbf{w} + \mathbf{u})\, d\mathbf{w}. \tag{2.18}$$

These identities hold for every nonnegative measurable f.

The same formula also averages sums over pairs with a prescribed difference coset.

Lemma 2.6 (Averaging over pairs)**.** *Let $\Gamma \subseteq \mathcal{L}$ be full-rank lattices, $d = \det \mathcal{L}$, and $q = [\mathcal{L} : \Gamma]$. Let $\mathbf{u} \bmod \mathcal{L}$ be uniform. For every nonnegative measurable $F : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and fixed $\mathbf{c}$,*

$$\mathbb{E}_{\mathbf{u}} \sum_{\substack{\mathbf{x} \in \mathcal{L}+\mathbf{u},\, \mathbf{y} \in \mathcal{L}+\mathbf{u}-\mathbf{c} \\ \mathbf{x}-\mathbf{y} \in \Gamma+\mathbf{c}}} F(\mathbf{x}, \mathbf{y}) = \frac{1}{d} \sum_{\mathbf{z} \in \Gamma+\mathbf{c}} \int_{\mathbb{R}^n} F(\mathbf{x}, \mathbf{x}-\mathbf{z})\, d\mathbf{x}. \tag{2.19}$$

If $\mathbf{c} \bmod \Gamma$ is independently uniform, then $\mathbf{u}$ and $\mathbf{u}-\mathbf{c}$ modulo $\mathcal{L}$ are independent and uniform, and

$$\mathbb{E}_{\mathbf{c},\mathbf{u}} \sum_{\substack{\mathbf{x} \in \mathcal{L}+\mathbf{u},\, \mathbf{y} \in \mathcal{L}+\mathbf{u}-\mathbf{c} \\ \mathbf{x}-\mathbf{y} \in \Gamma+\mathbf{c}}} F(\mathbf{x}, \mathbf{y}) = \frac{1}{qd^2} \int_{\mathbb{R}^n} \int_{\mathbb{R}^n} F(\mathbf{x}, \mathbf{y})\, d\mathbf{x}\, d\mathbf{y}. \tag{2.20}$$

Proof. Group pairs by $\mathbf{z} = \mathbf{x} - \mathbf{y}$ and apply Equation (2.17) to the sum over $\mathbf{x}$. Applying the same identity on Γ to the sum over $\mathbf{z}$, and using $\det \Gamma = qd$, gives the second formula. For independence, conditional on $\mathbf{u}$, the class $\mathbf{u}-\mathbf{c} \bmod \mathcal{L}$ is uniform because the image of a uniform class modulo Γ is uniform modulo $\mathcal{L}$. □

For a Gaussian density, Poisson summation expresses the periodized function as a Fourier series indexed by the dual lattice. For $\sigma > 0$ and $\mathbf{c} \in \mathbb{R}^n$,

$$P_{\mathcal{L},\phi_\sigma}(\mathbf{c}) = \sum_{\mathbf{y} \in \mathcal{L}^*} e^{-2\pi^2\sigma^2\|\mathbf{y}\|^2} e^{2\pi i \langle \mathbf{y}, \mathbf{c} \rangle}.$$

Setting $\mathbf{c} = \mathbf{0}$ gives the first equality below. Separating the constant term and applying the triangle inequality gives the second bound.

$$P_{\mathcal{L},\phi_\sigma}(\mathbf{0}) = \rho_{1/(\sqrt{2\pi}\sigma)}(\mathcal{L}^*), \qquad |P_{\mathcal{L},\phi_\sigma}(\mathbf{c}) - 1| \leq \rho_{1/(\sqrt{2\pi}\sigma)}(\mathcal{L}^*) - 1. \tag{2.21}$$

The Poisson summation formula and Parseval's identity also give

$$R_{\mathcal{L}}(\phi_\sigma) = \rho_{1/(2\sqrt{\pi}\sigma)}(\mathcal{L}^*). \tag{2.22}$$

Suppose f satisfies $f(\mathbf{x}) \leq Ce^{-\pi\|\mathbf{x}-\mathbf{c}_0\|^2/s^2}$ for all $\mathbf{x} \in \mathbb{R}^n$, for some $C, s > 0$ and $\mathbf{c}_0 \in \mathbb{R}^n$. Its periodization is bounded by Equation (2.21). For $\mathbf{y} \in \mathcal{L}^*$, integration over a fundamental region gives

$$\mathbb{E}_{\mathbf{c} \bmod \mathcal{L}} \left[P_{\mathcal{L},f}(\mathbf{c}) e^{-2\pi i \langle \mathbf{y}, \mathbf{c} \rangle} \right] = \int_{\mathbb{R}^n} f(\mathbf{x}) e^{-2\pi i \langle \mathbf{y}, \mathbf{x} \rangle}\, d\mathbf{x} = \widehat{f}(\mathbf{y}),$$

where $\widehat{f}$ denotes the Fourier transform of f. Parseval's identity therefore gives

$$R_{\mathcal{L}}(f) = \sum_{\mathbf{y} \in \mathcal{L}^*} |\widehat{f}(\mathbf{y})|^2, R_{\mathcal{L}}(f) - \left(\int_{\mathbb{R}^n} f(\mathbf{x})\, d\mathbf{x} \right)^2 = \sum_{\mathbf{y} \in \mathcal{L}^* \setminus \{\mathbf{0}\}} |\widehat{f}(\mathbf{y})|^2. \tag{2.23}$$

Translating f leaves the mean square unchanged because the shift is uniform modulo $\mathcal{L}$. For a full-rank sublattice $\Gamma \subseteq \mathcal{L}$ and $q = [\mathcal{L} : \Gamma]$, decomposition into cosets gives

$$P_{\mathcal{L},f}(\mathbf{c}) = q^{-1} \sum_{\mathbf{v} \in \mathcal{L}/\Gamma} P_{\Gamma,f}(\mathbf{c} + \mathbf{v}).$$

Jensen's inequality, averaged over $\mathbf{c} \bmod \Gamma$, therefore gives

$$R_{\mathcal{L}}(f) \leq R_{\Gamma}(f). \tag{2.24}$$

For any nonnegative measurable function g satisfying a Gaussian bound as above, $f \leq g$ implies $R_{\mathcal{L}}(f) \leq R_{\mathcal{L}}(g)$. The Cauchy–Schwarz inequality and translation invariance also give

$$\mathbb{E}_{\mathbf{c}}[P_{\mathcal{L},f}(\mathbf{c}-\mathbf{a}) P_{\mathcal{L},g}(\mathbf{c}-\mathbf{b})] \leq \sqrt{R_{\mathcal{L}}(f) R_{\mathcal{L}}(g)} \qquad (\mathbf{a}, \mathbf{b} \in \mathbb{R}^n). \tag{2.25}$$

Lemma 2.7 (Uniformity modulo a lattice)**.** *Let $\mathcal{L}$ be a full-rank lattice and let $g \geq 0$ be measurable and satisfy a Gaussian bound as above. If $m = \int_{\mathbb{R}^n} g(\mathbf{x})\,d\mathbf{x} > 0$, reducing the density g/m modulo $\mathcal{L}$ gives a distribution whose total variation distance from uniform is at most*

$$\tfrac{1}{2}\sqrt{R_{\mathcal{L}}(g)/m^2 - 1}. \tag{2.26}$$

Proof. The reduced distribution has density $P_{\mathcal{L},g}/m$ relative to uniform measure. By the Cauchy–Schwarz inequality and Equation (2.18), its total variation distance from uniform is

$$\frac{1}{2}\mathbb{E}_{\mathbf{c}}\left|\frac{P_{\mathcal{L},g}(\mathbf{c})}{m} - 1\right| \leq \frac{1}{2}\sqrt{\mathbb{E}_{\mathbf{c}}\left(\frac{P_{\mathcal{L},g}(\mathbf{c})}{m} - 1\right)^2} = \frac{1}{2}\sqrt{\frac{R_{\mathcal{L}}(g)}{m^2} - 1}. \qquad \square$$

2.3 Lattice algorithms

The following length estimates let us set the shell radii in the shortest-vector algorithms.

Lemma 2.8 (Guessing $\lambda_1(\mathcal{L})$)**.** *Given a basis of $\mathcal{L}$, one can construct polynomially many positive length guesses in polynomial time, one of which satisfies $\lambda_1(\mathcal{L}) \leq \widehat{\lambda} \leq (1 + n^{-2})\lambda_1(\mathcal{L})$.*

Schnorr's block reduction [Sch87], combined with the single-exponential-time SVP algorithm of Ajtai, Kumar, and Sivakumar [AKS01], gives the following guarantee.

Theorem 2.9 (Block reduction)**.** *Let $\mathcal{L}$ be an n-dimensional lattice given by a basis, let $2 \leq k \leq n$ be an integer, and let $0 < \delta < 1/2$. A randomized algorithm computes a basis $(\mathbf{b}_i)$ satisfying*

$$\max_i \|\widetilde{\mathbf{b}}_i\| \leq k^{n/k}\lambda_n(\mathcal{L}) \tag{2.27}$$

with probability at least $1 - \delta$, in time $2^{O(k)}\,\mathrm{poly}(n, \log(1/\delta))$.

The restriction sampler uses the Gaussian lists of Aggarwal, Dadush, Regev, and Stephens-Davidowitz.

Definition 2.10 (Gaussian list, [ADRS15, Definition 5.1])**.** To draw an ideal Gaussian list at width s, first choose its length $M \geq 0$, then draw M independent samples from $D_{\mathcal{L},s}$. The sampling randomness is independent of the choice of M. An implementation has error ν if the distribution of its entire output list is within total variation ν of that of an ideal list.

Theorem 2.11 (ADRS sampler, [ADRS15, Theorem 5.11])**.** *For an n-dimensional lattice $\mathcal{L}$, a width $s > 0$, and an accuracy parameter $\kappa \geq Cn$, the ADRS sampler produces a Gaussian list with error $e^{-\Omega(\kappa)}$ in time $2^{n/2+\mathrm{polylog}(\kappa)+o(n)}$, up to polynomial factors in the input length. If $s > \sqrt{2}\eta_{1/2}(\mathcal{L})$, the ideal list has $N_n = \lceil 2^{n/2} \rceil$ entries. At other widths, its length may be smaller, including zero; the Gaussian-list guarantee continues to hold.*

2.4 Probability bounds

For $X \geq 2$, a prime in $[X, 64X]$ can be found with failure $e^{-\Omega(n^2)}$ using polynomially many bit operations in $n, \log(2 + X)$ [RS62, AKS04].

We recall Markov's inequality and a special case of Paley–Zygmund inequality.

Lemma 2.12 (Moment bounds)**.** *For a nonnegative random variable X and $t > 0$, the following hold:*

$$\Pr[X \geq t] \leq \mathbb{E}X/t, \qquad \Pr[X > 0] \geq \frac{(\mathbb{E}X)^2}{\mathbb{E}X^2} \quad (0 < \mathbb{E}X^2 < \infty). \tag{2.28}$$

If each of N independent trials succeeds with probability p, the probability of at least one success is $1-(1-p)^N \ge \frac{1}{2}\min\{1, Np\}$. We use the following form of Chebyshev's inequality.

Lemma 2.13 (Mean estimation)**.** *Let $\overline{W}$ be the average of M independent $[0,1]$-valued random variables with common mean $\mu \ge c > 0$. Then, for every $\delta > 0$,*

$$\Pr[|\overline{W} - \mu| > \delta\mu] \le (4M\delta^2 c^2)^{-1}. \tag{2.29}$$

Proof. Since $\mathrm{Var}(\overline{W}) \le 1/(4M)$, Chebyshev's inequality gives the bound. □

For probability measures P, Q with densities p, q relative to a common dominating measure, total variation distance is $d_{\mathrm{TV}}(P,Q) = \sup_A |P(A) - Q(A)| = \frac{1}{2}\int |p - q|$.

Lemma 2.14 (Total variation bounds)**.** *Total variation is nonincreasing under a common randomized procedure and convex under mixtures. For nonnegative measurable weights $w, \widetilde{w}$, if $(1-\delta)w \le \widetilde{w} \le (1+\delta)w$, $0 \le \delta < 1$, and $0 < \int w < \infty$, then*

$$d_{\mathrm{TV}}\left(\frac{\widetilde{w}}{\int \widetilde{w}}, \frac{w}{\int w}\right) \le \frac{\delta}{1-\delta}. \tag{2.30}$$

Lemma 2.15 (Coupling)**.** *Suppose an algorithm makes at most N random yes-or-no decisions. If at every history the probability of a yes answer changes by at most δ, the output distributions differ by at most $N\delta$ in total variation. For probability measures P, Q, $d_{\mathrm{TV}}(P^{\otimes k}, Q^{\otimes k}) \le k\, d_{\mathrm{TV}}(P,Q)$.*

We use Gaussian concentration to bound how often a sample leaves a specified shell, including when the Gaussian has a fixed shift.

Lemma 2.16 (Gaussian concentration)**.** *Let $\mathbf{Z} \sim N(\mathbf{0}, \mathbf{I}_d)$, let $\mathbf{u}$ be a unit vector, and let $t \ge 0$. Then*

$$\Pr[|\langle \mathbf{u}, \mathbf{Z}\rangle| \ge t] \le 2e^{-t^2/2}, \qquad \Pr[|\|\mathbf{Z}\|^2 - d| \ge t] \le 2e^{-\frac{1}{8}\min\{t^2/d, t\}}. \tag{2.31}$$

Furthermore, let $\mathbf{W} \sim N(\mathbf{0}, \sigma^2\mathbf{I}_n/2)$ and $\|\mathbf{z}\|^2 \le Cn\sigma^2$. For every $0 < \delta \le 1$, both signs satisfy

$$\left| \frac{\|\mathbf{W} \pm \mathbf{z}/2\|^2}{n\sigma^2} - \left(\frac{1}{2} + \frac{\|\mathbf{z}\|^2}{4n\sigma^2}\right)\right| \le \delta \tag{2.32}$$

except with probability $4e^{-c_C n\delta^2}$, where $c_C > 0$ depends only on C. Finally, for $\mathbf{Z} \sim N(\mathbf{0}, \mathbf{I}_n)$ and $0 < r < 1$,

$$\Pr[\|\mathbf{Z}\|^2 \le rn] \le (re^{1-r})^{n/2}. \tag{2.33}$$

Proof. The bounds in (2.31) are standard [LM00]. For the shifted estimate, expand $\|\mathbf{W} \pm \mathbf{z}/2\|^2 = \|\mathbf{W}\|^2 \pm \langle \mathbf{W}, \mathbf{z}\rangle + \|\mathbf{z}\|^2/4$. Apply (2.31) to the centered squared norm and to $\langle \mathbf{W}, \mathbf{z}\rangle$, a centered Gaussian of variance at most $Cn\sigma^4/2$. With probability at least $1 - 4e^{-c_C n\delta^2}$ the two terms are each at most $\delta n\sigma^2/2$ in absolute value, which proves the bound simultaneously for both signs.

For the last bound, the Gaussian exponential moment is $\mathbb{E}e^{-t\|\mathbf{Z}\|^2} = (1+2t)^{-n/2}$. Markov's inequality at $t = \frac{1}{2r} - \frac{1}{2}$ gives $\Pr[\|\mathbf{Z}\|^2 \le rn] \le e^{trn}(1+2t)^{-n/2} = (re^{1-r})^{n/2}$, as desired. □

2.5 Fast search algorithms

2.5.1 Quantum search

Suppose $r > 0$ of M indexed entries meet a test. Quantum search finds one of them using $O(\sqrt{M/r})$ expected queries [BBHT98]. Listing every matching entry costs $\sqrt{M(r+1)}$ queries up to logarithmic factors [vAGN24]. Testing whether any entry matches and reducing the error probability to 2^{-n} cost polynomial factors. Dürr–Høyer minimum finding uses $O(\sqrt{M})$ queries with constant error [DH96]. These bounds assume coherent access to the indexed data. We charge separately for loading and maintaining the data.

2.5.2 Spherical pair enumeration

We find pairs with large normalized inner product by grouping the input vectors by direction [BDGL16]. A unit vector $\mathbf{s}$ and threshold u define the filter

$$\mathcal{C}_{\mathbf{s}}(u) = \{\mathbf{x} \neq \mathbf{0} : \langle \mathbf{s}, \mathbf{x} \rangle \geq \sqrt{u}\|\mathbf{x}\|\}.$$

We examine pairs contained in a common filter. For two fixed unit vectors with inner product $1/2$, a uniformly random direction $\mathbf{s}$ puts both in the filter with probability $(1 - 4u/3)^{n/2} 2^{O(\log n)}$ for fixed $0 < u < 3/4$ by [BDGL16, Lemma 2.2]. The reciprocal gives the scale of the number of filters needed to find such pairs.

Finding the filters containing a vector. The BDGL construction represents filter directions using a product of small sets. Append $O(\log n)$ zero coordinates to the inputs so that the resulting dimension d is divisible by $m = \Theta(\log n)$. For each block, choose $2^{O(n/\log n)}$ independent uniform vectors forming $C_i \subset m^{-1/2}\mathbb{S}^{d/m-1}$. Choose an independent uniform orthogonal transformation Q. One product family consists of the directions

$$\mathbf{s} = Q(\mathbf{c}_1, \ldots, \mathbf{c}_m), \qquad \mathbf{c}_i \in C_i.$$

For an input vector $\mathbf{x}$, write

$$Q^{\mathsf{T}}\mathbf{x}/\|\mathbf{x}\| = (\mathbf{z}_1, \ldots, \mathbf{z}_m).$$

The filters that contain $\mathbf{x}$ are exactly the tuples satisfying

$$\sum_{i=1}^{m} \langle \mathbf{c}_i, \mathbf{z}_i \rangle \geq \sqrt{u}.$$

Lemma 2.17 (Spherical-cap enumeration, BDGL parameter form)**.** *Let $h = \lceil \log_2(n+2) \rceil$, $0 < \varepsilon \leq h^{-2}$, and $q = 2^{\lceil n/h \rceil}$. Fix $\delta > 0$. Let $\rho_0 \in [1/3, 1/2]$ be a target correlation and $u \in [0, 0.342]$ a squared cap threshold such that*

$$\Delta := 1 - \frac{2u}{1+\rho_0} \geq \frac{1}{2} + \delta. \tag{2.34}$$

The bounds below hold uniformly on this parameter domain for fixed δ. All applications in this paper satisfy it with $\delta = 0.01$ for sufficiently large n.

Use the BDGL enumeration algorithm with target correlation $\rho_0 - C\varepsilon$ for a sufficiently large fixed C. Append $O(\log_2 n)$ zero coordinates, use a code-size multiplier between q and $8q$, and sample n independent filter families. The construction uses $\Delta^{-n/2} 2^{o(n)}$ filters. For every fixed pair of correlation at least $\rho_0 - O(\varepsilon)$, the probability that no filter contains both vectors is at most 2^{-n}.

For a single filter, the probabilities of containing a fixed point or both members of a fixed pair are bounded by

$$2^{o(n)}(1-u)^{n/2}, \qquad 2^{o(n)}\left(1-\frac{2u}{1+\rho}\right)_+^{(n-2)/2}, \tag{2.35}$$

respectively, where $\rho > -1$ is the pair correlation. An antipodal pair has probability zero of sharing a filter. The algorithm lists the vector-filter memberships one at a time, in time linear in the input and output sizes up to $2^{o(n)}$ factors, using $2^{o(n)}$ auxiliary space. If each part may contain at most B memberships and there are I in total, the product-code traversal groups the memberships into at most $2^{o(n)}(1 + I/B)$ parts.

Proof. We use the BDGL product-code construction [BDGL16] and the traversal bounds of [GFH26a, Theorem 4.2 and Proposition 4.3]. We first compute how the probability that a pair shares a filter depends on its correlation.

For unit vectors of correlation $-1 < \rho < 1$, the minimum squared norm of a vector having inner product at least $\sqrt{u}$ with both is

$$u(1,1)\begin{pmatrix}1 & \rho\\ \rho & 1\end{pmatrix}^{-1}\begin{pmatrix}1\\1\end{pmatrix} = \frac{2u}{1+\rho}. \tag{2.36}$$

The minimizer is parallel to the sum of the two unit vectors. Thus the common-cap factor at the target correlation is Δ. Substituting the corresponding Gram matrix and cap factor into the product-code analysis gives the stated bounds on the filter count and the failure probability. The target Gram determinant is at least $3/4$, and (2.34) gives the fixed margin above $1/2$ needed by the block estimates. Perturbing the target by $O(\varepsilon)$ changes the count by a factor $2^{O(n\varepsilon)}$. The code-size multiplier absorbs this factor because $n\varepsilon = o(\ln q)$. Independent families then give the claimed failure bound for each pair.

A random rotation gives the cap probabilities in (2.35). The cited product-code traversal gives the time, space, and partition bounds for listing the memberships. At $u = 0$, use four independent random hemispheres per family. A target pair has positive correlation for sufficiently large n, so the probability that all four hemispheres miss the pair is below $1/2$. The n families reduce this probability to at most 2^{-n}. Each membership can be checked directly and grouped into parts with the same size bound. □

Finding pairs in the same filter. Let X, Y be two lists. Entries may have labels, such as lattice cosets, and we compare only entries with the same label. Assume that labels have poly(n)-bit encodings. For each filter f and label a, let $X_{f,a}, Y_{f,a}$ be the corresponding sublists, and define

$$N = |X| + |Y|, \qquad I = \sum_{f,a}(|X_{f,a}| + |Y_{f,a}|), \qquad J = \sum_{f,a}|X_{f,a}|\,|Y_{f,a}|.$$

Thus J counts each pair once for each common filter. Repeated entries are counted with their multiplicities.

Lemma 2.18 (Pairs in common filters, [GFH26a, Proposition 4.3])**.** *For the filters of Lemma 2.17, all pairs from $X_{f,a} \times Y_{f,a}$ can be listed in time*

$$2^{o(n)}(N + I + J + 1)$$

and auxiliary space $O(N) + 2^{o(n)}$.

Proof. The claim is immediate if $N = 0$. Fix all traversal and subdivision rules deterministically. Process the independent product families sequentially. For one product, enumerate the filters containing each entry of X, stopping as soon as more than N records have been generated. If this happens, divide a nonsingleton set C_i into two nearly equal subsets and process the resulting products separately. Otherwise, group the records by filter and label. A single filter contains at most $|X| \le N$ entries, so this subdivision terminates.

For each product whose records fit, enumerate the filters containing each entry of Y and list its pairs with the corresponding entries of X. The products partition the filters, so every required pair is listed with its multiplicity. Processing the products sequentially gives the space bound.

The subdivision has depth $O(\log F) = O(n)$, where F is the total number of filters. At any fixed depth, the products have disjoint filter sets, so at most I/N of them generate more than N records. There are therefore $O(n(1 + I/N))$ products in total, including the independent families. At each depth, each vector-filter membership is generated at most once. The repeated input scans and filter computations consequently take $2^{o(n)}(N + I + 1)$ time. Grouping records uses their polynomial-length encodings, and listing the pairs takes an additional $2^{o(n)}J$ time. □

For fixed input lists and product families, the enumeration order does not depend on subsequent acceptance decisions. The enumeration may list the same pair of list entries through several filters. Any rule that processes a pair only once is applied separately by the calling algorithm.

Quantitative bounds for Gaussian-weighted lists. The sampler in Section 4 uses the following more precise bounds for enumerating pairs through spherical filters.

Lemma 2.19 (Quantitative cap enumeration, [DL26, Lemma A.1 and (58)–(60)])**.** *Let* $h = \lceil \log_2(n + 2) \rceil$ *and* $q = 2^{\lceil n/h \rceil}$*. For* $u = 1/4$*, the amplified construction uses at most* $(3/2)^{n/2} q^{1/2}$ *filters in total and* $q^{1/3}$ *words for each set* C_i*. After adding* $O(\log n)$ *zero coordinates, each filter direction is uniform. Every fixed pair with normalized inner product in* $[1/2 - 1/\lceil \ln n \rceil, 2/3]$ *shares a filter except with probability at most* 2^{-n}*.*

For input lists X, Y*, let* X_f, Y_f *be their sublists in filter* f*, and define*

$$I = \sum_f (|X_f| + |Y_f|), \qquad C = \sum_f |X_f|\,|Y_f|.$$

All pairs in these products can be listed in time

$$\mathrm{poly}(n) q^{1/3} (|X| + |Y| + I + C + 1) \tag{2.37}$$

and auxiliary space $O(1 + |X| + |Y| + \mathrm{poly}(n) q^{1/3})$*.*

Proof. The cited construction gives the total filter count, the sizes of the sets C_i, and the failure probability. Apply the subdivision procedure in the proof of Lemma 2.18, with all labels equal and filter-enumeration overhead $\mathrm{poly}(n) q^{1/3}$. This gives the time bound. The product families occupy $\mathrm{poly}(n) q^{1/3}$ words, and pair enumeration uses an additional $O(1 + |X| + |Y|)$ words. □

3 Discrete Gaussian Sampling in Time $2^{0.5n+o(n)}$ – Restriction

Theorem 3.1 (One discrete Gaussian sample in $2^{n/2+o(n)}$ time)**.** *Given a basis of an n-dimensional lattice $\mathcal{L}$, $s > 0$, and $\kappa \geq n$, Algorithm 1 runs in time and space $2^{n/2+\mathrm{polylog}(\kappa)+o(n)}$ and outputs a sample within total variation distance $e^{-\kappa}$ from $D_{\mathcal{L},s}$.*

A sample from a discrete Gaussian on a superlattice has the desired distribution once we condition on its belonging to the input lattice. The challenge is to find a superlattice on which sampling is efficient and on which this conditioning accepts often enough.

We choose a random extension of prime index. In the dual lattice, this removes most vectors outside the sublattice of prime multiples. Choosing the index in proportion to the dual Gaussian mass makes the extension smooth at the width required by the sampler. The same choice bounds how much larger the Gaussian mass of the extension is than that of the input lattice. On an appropriate extension, the probability of belonging to the input lattice is at least a constant times $2^{-n/2}$ per sample. The sampler supplies about $2^{n/2}$ independent samples in one call, so a constant fraction of the calls produce a usable sample.

Later sections use the same random extension to start a chain of superlattices.

Roadmap. Section 3.1 constructs the extension and analyzes the sampler. Section 3.2 bounds the minimum smoothing index and proves a matching lower bound on the sample cost of restriction.

3.1 Random extensions and the restriction sampler

Let $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$ be a full-rank lattice and p a prime. For $\mathbf{z} \in \mathbb{F}_p^n \setminus \{\mathbf{0}\}$ and any integer representative $\bar{\mathbf{z}}$, we define the superlattice

$$K_{\mathbf{z}} = \{\mathbf{a} \in \mathbb{Z}^n : \langle \mathbf{z}, \mathbf{a} \rangle \equiv 0 \bmod p\}, \qquad \mathcal{L}_{\mathbf{z}} = \mathcal{L} + \mathbb{Z}\,\frac{\mathbf{B}\bar{\mathbf{z}}}{p} \tag{3.1}$$

with index $[\mathcal{L}_{\mathbf{z}} : \mathcal{L}] = p$. Its dual is $\mathcal{L}_{\mathbf{z}}^* = \mathbf{B}^{-T} K_{\mathbf{z}}$. Indeed, for $\mathbf{a} \in \mathbb{Z}^n$ the pairing $\langle \mathbf{B}^{-T}\mathbf{a}, \mathbf{B}\bar{\mathbf{z}}/p \rangle = \langle \mathbf{a}, \bar{\mathbf{z}} \rangle / p$ is integral precisely when $\mathbf{a} \in K_{\mathbf{z}}$. Averaging over $\mathbf{z}$ controls the Gaussian mass of the surviving dual vectors. A similar construction using modulus 2 appears in [ADRS15, Lemma 5.12]. Using a larger prime modulus p replaces the dual sublattice $2\mathcal{L}^*$ by $p\mathcal{L}^*$, so short vectors in $2\mathcal{L}^* \setminus p\mathcal{L}^*$ are no longer forced to remain.

For every nonnegative ℓ^1 function g on $\mathcal{L}^* \setminus \{\mathbf{0}\}$, observe that

$$\begin{aligned} \mathbb{E}_{\mathbf{z}} \sum_{\mathbf{y} \in \mathcal{L}_{\mathbf{z}}^* \setminus \{\mathbf{0}\}} g(\mathbf{y}) &= \frac{p^{n-1}-1}{p^n-1} \sum_{\mathbf{y} \in \mathcal{L}^* \setminus \{\mathbf{0}\}} g(\mathbf{y}) + \left(1 - \frac{p^{n-1}-1}{p^n-1}\right) \sum_{\mathbf{y} \in p\mathcal{L}^* \setminus \{\mathbf{0}\}} g(\mathbf{y}) \\ &\leq \frac{1}{p} \sum_{\mathbf{y} \in \mathcal{L}^* \setminus \{\mathbf{0}\}} g(\mathbf{y}) + \sum_{\mathbf{y} \in p\mathcal{L}^* \setminus \{\mathbf{0}\}} g(\mathbf{y}) \end{aligned} \tag{3.2}$$

where the expectation is over a uniform random nonzero $\mathbf{z} \in \mathbb{F}_p^n$.

The next two lemmas bound the dual Gaussian mass of the extension.

Lemma 3.2 (Non-zero Gaussian mass of index-p extensions)**.** *For every $t > 0$,*

$$\mathbb{E}_{\mathbf{z}} \rho_{1/t}(\mathcal{L}_{\mathbf{z}}^* \setminus \{\mathbf{0}\}) \leq \frac{2\rho_{1/t}(\mathcal{L}^*)}{p}. \tag{3.3}$$

Algorithm 1 Arbitrary-width centered sampler $\mathsf{DirectDGS}(\mathbf{B}, s, \kappa)$

Input: a basis $\mathbf{B}$ of $\mathcal{L}$, $s > 0$, and $\kappa \geq n$

Set $J = \left\lceil 5 + n \log_2\left(1 + \frac{\sqrt{2}\|\mathbf{B}\|_F}{s}\right)\right\rceil$ and fix a sufficiently large constant $C > 0$.
for $r = 1, \ldots, \lceil C\kappa \rceil$ **do**
  for $m = 1, \ldots, J$ **do**
    Try up to $\lceil Cm \rceil$ independent uniform integers in $[2^m, 2^{m+1})$, taking the first prime p.
    If none is prime, continue to the next m.
    Choose uniform $\mathbf{z} \in \mathbb{F}_p^n \setminus \{\mathbf{0}\}$.
    Run the ADRS sampler on $(\mathcal{L}_\mathbf{z}, s, \lceil C(\kappa + \log(2 + J))\rceil)$.
    Return the first sample belonging to $\mathcal{L}$, if any.
return 0.

Proof. Apply (3.2) with $g(\mathbf{y}) = \rho_{1/t}(\mathbf{y})$. As $\rho_{1/t}(\mathcal{L}_\mathbf{z}^* \setminus \{\mathbf{0}\}) = \sum_{\mathbf{y} \in \mathcal{L}_\mathbf{z}^* \setminus \{\mathbf{0}\}} g(\mathbf{y})$, this gives

$$\mathbb{E}_\mathbf{z} \rho_{1/t}(\mathcal{L}_\mathbf{z}^* \setminus \{\mathbf{0}\}) \leq \frac{1}{p}\rho_{1/t}(\mathcal{L}^* \setminus \{\mathbf{0}\}) + \rho_{1/t}(p\mathcal{L}^* \setminus \{\mathbf{0}\}) \leq \frac{2}{p}\rho_{1/t}(\mathcal{L}^* \setminus \{\mathbf{0}\}) \leq \frac{2}{p}\rho_{1/t}(\mathcal{L}^*)$$

where the second inequality follows from the lemma below. This proves (3.3). □

Lemma 3.3 (Non-zero Gaussian masses of scaled lattices)**.** *For every lattice $\mathcal{L}$, parameter $s > 0$, and integer $k \geq 1$,*

$$\rho_s(k\mathcal{L} \setminus \{\mathbf{0}\}) \leq \frac{1}{k}\rho_s(\mathcal{L} \setminus \{\mathbf{0}\}). \tag{3.4}$$

Proof. Let $\mathcal{P}(\mathcal{L})$ contain one representative up to sign of each primitive vector of $\mathcal{L}$, where $\mathbf{v} \neq \mathbf{0}$ is primitive if $\mathbf{v}/d \notin \mathcal{L}$ for every integer $d \geq 2$. Every nonzero lattice vector is uniquely $m\mathbf{v}$ with $\mathbf{v} \in \mathcal{P}(\mathcal{L})$ and $m \in \mathbb{Z} \setminus \{0\}$. Then we have

$$k\rho_s(k\mathcal{L} \setminus \{\mathbf{0}\}) = 2 \sum_{\mathbf{v} \in \mathcal{P}(\mathcal{L})} \sum_{m \geq 1} k\rho_s(km\mathbf{v}) \leq 2 \sum_{\mathbf{v} \in \mathcal{P}(\mathcal{L})} \sum_{m \geq 1} \sum_{j=k(m-1)+1}^{km} \rho_s(j\mathbf{v}) = \rho_s(\mathcal{L} \setminus \{\mathbf{0}\})$$

where we use the fact that $\rho_s(j\mathbf{v})$ is nonincreasing in $j \geq 1$. □

Lemma 3.4 (Bounding the Gaussian mass of random index-p superlattices)**.** *Let $\mathcal{L}$ be full rank, $t > 0$, and p a prime with $32\rho_{1/t}(\mathcal{L}^*) \leq p < 128\rho_{1/t}(\mathcal{L}^*)$. For uniform $\mathbf{z} \in \mathbb{F}_p^n \setminus \{\mathbf{0}\}$, the extension $\mathcal{L}_\mathbf{z}$ satisfies, with probability at least $3/4$,*

$$\eta_{1/2}(\mathcal{L}_\mathbf{z}) < t, \qquad \rho_{\chi t}(\mathcal{L}_\mathbf{z}) \leq 160\chi^n \rho_t(\mathcal{L}) \quad \textit{for every } \chi \geq 1.$$

Proof. Lemma 3.2 gives an upper bound of $1/16$ on $\mathbb{E}_\mathbf{z}\rho_{1/t}(\mathcal{L}_\mathbf{z}^* \setminus \{\mathbf{0}\})$. By Markov's inequality, with probability at least $3/4$ this mass is at most $1/4$, which implies $\eta_{1/2}(\mathcal{L}_\mathbf{z}) < t$. When the nonzero dual mass is at most $1/4$, the Poisson summation formula and $[\mathcal{L}_\mathbf{z} : \mathcal{L}] = p$ give, for every $\chi \geq 1$,

$$\frac{\rho_{\chi t}(\mathcal{L}_\mathbf{z})}{\rho_t(\mathcal{L})} = \chi^n \frac{p\,\rho_{1/(\chi t)}(\mathcal{L}_\mathbf{z}^*)}{\rho_{1/t}(\mathcal{L}^*)} \leq \frac{5}{4}\chi^n \frac{p}{\rho_{1/t}(\mathcal{L}^*)} \leq 160\chi^n.$$

□

The restriction sampler. We try dyadic intervals for the prime index and use the first sample that belongs to $\mathcal{L}$. Figure 2 illustrates the restriction step on a suitable extension.

Proof of Theorem 3.1. First consider the algorithm with each ADRS output replaced by its ideal Gaussian list (Definition 2.10). For every extension and every ideal list length, the first sample in $\mathcal{L}$, conditional on there being one, has distribution $D_{\mathcal{L},s}$. This follows from independence of the samples and from restricting the Gaussian weights to $\mathcal{L}$. Earlier unsuccessful calls do not affect this conditional distribution.

We bound the probability that no ideal call produces such a sample. Each dyadic interval $[2^m, 2^{m+1})$ contains a fraction $\Omega(1/m)$ of primes, by the prime-counting bounds of [RS62] and Bertrand's postulate for the finitely many small intervals. The stated number of uniform trials, each using a deterministic primality test [AKS04], gives probability at least $3/4$ of finding a prime for sufficiently large C. Applying (2.12) to $\mathcal{L}^* = \mathbf{B}^{-T}\mathbb{Z}^n$ gives

$$\rho_{\sqrt{2}/s}(\mathcal{L}^*) \le \left(1 + \frac{\sqrt{2}\|\mathbf{B}\|_F}{s}\right)^n.$$

Thus $m^* := \lceil \log_2(32\rho_{\sqrt{2}/s}(\mathcal{L}^*)) \rceil \le J$. Every prime in $[2^{m^*}, 2^{m^*+1})$ meets the index condition of Lemma 3.4 at width $s/\sqrt{2}$. The lemma gives, with probability at least $3/4$ over $\mathbf{z}$,

$$\eta_{1/2}(\mathcal{L}_{\mathbf{z}}) < s/\sqrt{2}, \qquad \rho_s(\mathcal{L}_{\mathbf{z}}) \le 160\, 2^{n/2} \rho_{s/\sqrt{2}}(\mathcal{L})$$

For such an extension, the ideal sampler of Theorem 2.11 outputs $\lceil 2^{n/2} \rceil$ independent samples, each lying in $\mathcal{L}$ with probability at least

$$\rho_s(\mathcal{L})/\rho_s(\mathcal{L}_{\mathbf{z}}) \ge 2^{-n/2}/160.$$

Each pass therefore produces a sample in $\mathcal{L}$ with at least a constant probability, either at m^* or in an earlier call. With C sufficiently large, the probability that all $\lceil C\kappa \rceil$ passes fail is at most $e^{-2\kappa}$. Returning $\mathbf{0}$ on failure changes the output distribution by at most this amount.

Theorem 2.11 bounds the error for a full list, including calls below smoothing. Coupling the actual and ideal lists over at most $\lceil C\kappa \rceil J$ calls gives total error at most

$$\lceil C\kappa \rceil J\, e^{-\Omega(C(\kappa + \log(2+J)))} \le e^{-2\kappa} \tag{3.5}$$

for sufficiently large C. The total variation error is therefore at most $2e^{-2\kappa} \le e^{-\kappa}$.

There are $O(\kappa J)$ calls, each costing $2^{n/2 + \mathrm{polylog}(\kappa + \log(2+J)) + o(n)}$. Finding each prime, constructing the extension basis, and testing membership in $\mathcal{L}$ take polynomial time per trial or list entry. Reusing the storage between calls gives the stated space bound. As in Theorem 2.11, the bounds suppress factors polynomial in the input length. Taking $\kappa = 2n^3$ gives Proposition 1.3. □

3.2 Cost of restriction

We bound the minimum index needed for smoothing and the expected number of samples needed when sampling from one superlattice and restricting to $\mathcal{L}$ as in Figure 2.

Proposition 3.5 (Smoothing index and restriction cost)**.** *Let $\mathcal{L}$ be full rank and $t > 0$. For $0 < \varepsilon \le 1$, the minimum index of a superlattice $\Lambda \supseteq \mathcal{L}$ with $\eta_\varepsilon(\Lambda) \le t$ lies between*

$$(1 - \varepsilon)\rho_{1/t}(\mathcal{L}^*) \quad \textit{and} \quad \frac{18}{\varepsilon}\rho_{1/t}(\mathcal{L}^*).$$

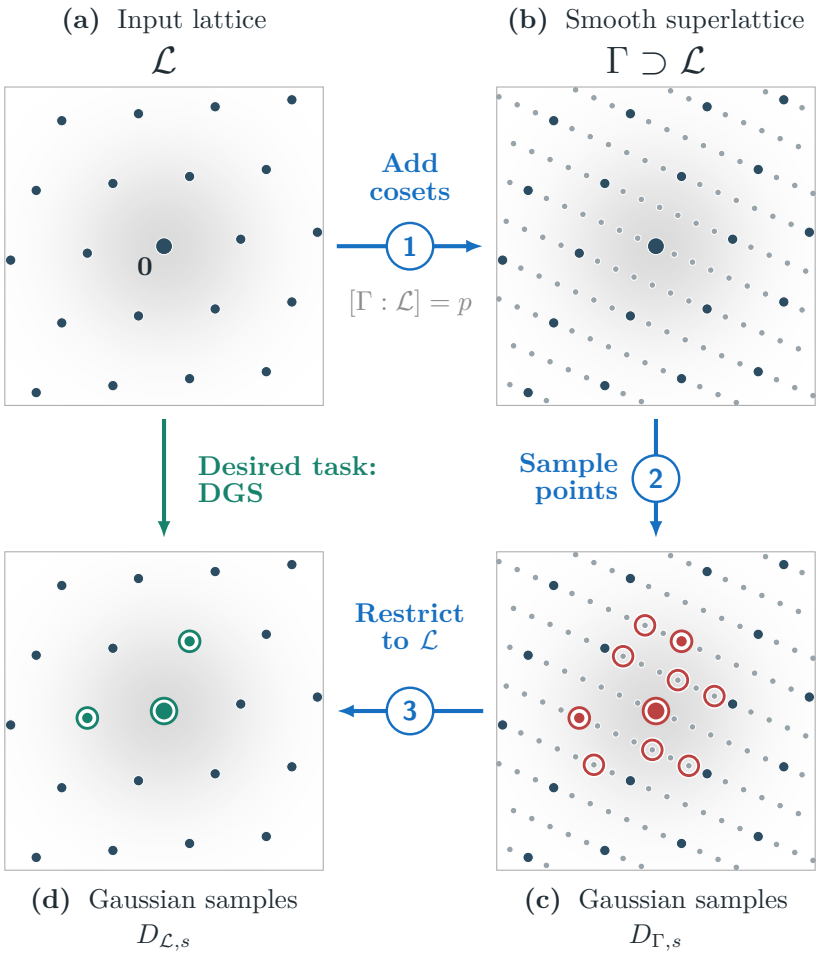


Figure 2: Discrete Gaussian sampling by restriction. Add cosets to obtain a smooth superlattice $\Gamma \supset \mathcal{L}$, sample from $D_{\Gamma,s}$, and keep the samples in $\mathcal{L}$. A sample from $D_{\Gamma,s}$ conditioned on lying in $\mathcal{L}$ has distribution $D_{\mathcal{L},s}$. Red rings in (c) show samples on Γ. Green rings in (d) show the samples that lie in $\mathcal{L}$. The grey shading represents the same Gaussian weights in all four panels. The enlarged dot is the origin. The displayed planar extension has index $p = 7$.

For every $\chi \geq 1$, the optimal expected number of samples needed to obtain a point of $\mathcal{L}$ by restriction satisfies

$$\max\left\{1, \frac{2}{3}\frac{\chi^n \rho_t(\mathcal{L})}{\rho_{\chi t}(\mathcal{L})}\right\} \leq \inf_{\substack{\Lambda \supseteq \mathcal{L} \\ \eta_{1/2}(\Lambda) \leq t}} \frac{\rho_{\chi t}(\Lambda)}{\rho_{\chi t}(\mathcal{L})} \leq 160\frac{\chi^n \rho_t(\mathcal{L})}{\rho_{\chi t}(\mathcal{L})}.$$

Proof. For the upper index bound, choose $Q = \lceil 8\rho_{1/t}(\mathcal{L}^*)/\varepsilon \rceil$ and a prime $Q < p < 2Q$. Lemma 3.2 gives an extension with nonzero dual mass at most $\varepsilon/4$, hence smooth at t, and index

$$p < 2Q \leq \frac{16}{\varepsilon}\rho_{1/t}(\mathcal{L}^*) + 2 \leq \frac{18}{\varepsilon}\rho_{1/t}(\mathcal{L}^*),$$

where the last step uses $\rho_{1/t}(\mathcal{L}^*) \geq 1$ and $\varepsilon \leq 1$.

We prove the lower bound. For $\Lambda \supseteq \mathcal{L}$ with $\eta_\varepsilon(\Lambda) \leq t$ and index $q = [\Lambda : \mathcal{L}] = \det \mathcal{L}/\det \Lambda$, the

Poisson summation formula and the definition of the smoothing parameter give

$$\rho_t(\Lambda) = \frac{t^n \rho_{1/t}(\Lambda^*)}{\det \Lambda} \le (1+\varepsilon)\frac{q\, t^n}{\det \mathcal{L}}, \qquad \rho_t(\Lambda) \ge \rho_t(\mathcal{L}) = \frac{t^n \rho_{1/t}(\mathcal{L}^*)}{\det \mathcal{L}}.$$

Combining these two inequalities gives the lower bound

$$q \ge \frac{\rho_{1/t}(\mathcal{L}^*)}{1+\varepsilon} \ge (1-\varepsilon)\rho_{1/t}(\mathcal{L}^*).$$

This proves the smoothing-index claim, also stated in Proposition 1.2.

For the sample cost, Lemma 3.4 gives the upper bound. For the lower bound, set $\varepsilon = 1/2$ in the index estimate to get $q \ge \frac{2}{3}\rho_{1/t}(\mathcal{L}^*)$. Therefore

$$\rho_{\chi t}(\Lambda) = \frac{(\chi t)^n \rho_{1/(\chi t)}(\Lambda^*)}{\det \Lambda} = \frac{q(\chi t)^n \rho_{1/(\chi t)}(\Lambda^*)}{\det \mathcal{L}} \ge \frac{q(\chi t)^n}{\det \mathcal{L}} \ge \frac{2}{3}\frac{(\chi t)^n \rho_{1/t}(\mathcal{L}^*)}{\det \mathcal{L}} = \frac{2}{3}\chi^n \rho_t(\mathcal{L}).$$

Dividing by $\rho_{\chi t}(\mathcal{L})$ gives the restriction lower bound; containment gives the bound 1. □

By the Poisson summation formula and monotonicity,

$$1 \le \frac{\rho_{1/t}(\mathcal{L}^*)}{\rho_{1/(\chi t)}(\mathcal{L}^*)} = \frac{\chi^n \rho_t(\mathcal{L})}{\rho_{\chi t}(\mathcal{L})} \le \chi^n.$$

By Proposition 3.5, this ratio also gives, up to constant factors, the ratio of the minimum 1/2-smoothing indices at widths t and χt. If $\rho_{1/t}(\mathcal{L}^*) = O(1)$, restriction needs only $O(1)$ samples in expectation. If $\rho_{\chi t}(\mathcal{L}) = 1 + o(1)$, the expected number needed is $\Theta(\chi^n)$.

The next theorem shows that the sample lower bound applies even when extensions may be chosen adaptively and the input lattice is $\mathbb{Z}^n$. At $\chi = \sqrt{2}$, the width ratio in the ADRS guarantee of Theorem 2.11, constant success probability requires $\Omega(2^{n/2})$ samples. This matches the sample bound for the restriction strategy.

Theorem 3.6 (Lower bound for restriction)**.** *Fix $\chi \ge 1$, $0 < \varepsilon \le 1$, and $N \ge 1$, and let $\mathcal{L} = \mathbb{Z}^n$ and $t = n^{-1/2}$. Consider a procedure that adaptively chooses superlattices $\Lambda_i \supseteq \mathcal{L}$ with $\eta_{1/2}(\Lambda_i) \le t$, receives at most N fresh independent samples from $D_{\Lambda_i, \chi t}$ per call, and outputs only a received sample in $\mathcal{L}$. Success probability at least ε requires, in expectation, at least*

$$\left(\frac{2}{3}\varepsilon - o(1)\right)\chi^n \quad \textit{samples and} \quad \left(\frac{2}{3}\varepsilon - o(1)\right)\frac{\chi^n}{N} \quad \textit{calls.}$$

Proof. Since χ is fixed,

$$1 \le \rho_t(\mathbb{Z}^n) \le \rho_{\chi t}(\mathbb{Z}^n) \le (1 + 3e^{-\pi n/\chi^2})^n = 1 + o(1).$$

By Proposition 3.5, each fresh sample belongs to $\mathcal{L}$ with conditional probability at most $(3/2 + o(1))\chi^{-n}$. Let M be the total number of received samples. Success requires at least one received sample in $\mathcal{L}$, so conditional expectation gives

$$\varepsilon \le \mathbb{E}[\text{number of received samples in } \mathcal{L}] \le (3/2 + o(1))\chi^{-n}\mathbb{E}[M].$$

Rearranging gives $\mathbb{E}[M] \ge (\frac{2}{3}\varepsilon - o(1))\chi^n$. Each call supplies at most N samples, so the expected number of calls is at least $\mathbb{E}[M]/N$. □

Section 4 improves the exponent by merging lists along a chain of extensions and correcting the distribution of the final sample.

4 Discrete Gaussian Sampling in Time $2^{0.2924\ldots n+o(n)}$ – Gaussian-Weighted Merges

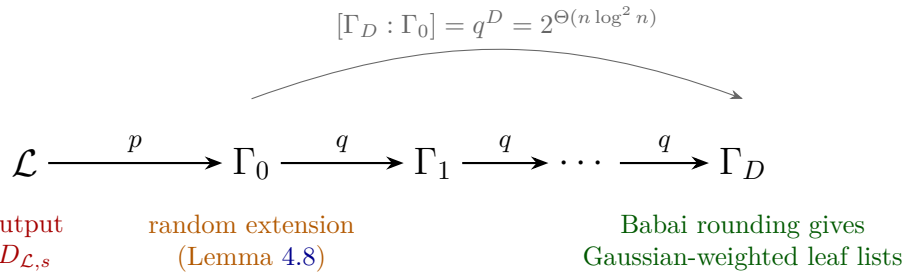
Theorem 1.4 (Discrete Gaussian sampling, restated)**.** *For an input lattice $\mathcal{L}$ of rank n, a squared Gaussian width $s^2 > 0$ and an accuracy parameter $\kappa \geq 1$, a bounded randomized (classical) algorithm outputs a point $\mathbf{x} \in \mathcal{L}$ within total variation $2^{-\kappa}$ of $D_{\mathcal{L},s}$ in time* T *and space* S *satisfying*

$$\mathrm{T} = \left(\frac{3}{2}\right)^{n/2+o(n)} \mathrm{poly}(\kappa), \qquad \mathrm{S} = \left(\frac{4}{3}\right)^{n/2+o(n)} \mathrm{poly}(\kappa). \tag{1.5}$$

The restriction sampler of Section 3.1 samples on one smooth extension and uses only samples already in $\mathcal{L}$. Here we combine samples in pairs. The difference of two points from suitable cosets can lie in a smaller lattice even when neither point does. Starting on a dense extension, repeated merges bring the lists back to $\mathcal{L}$.

1. **Give each difference the right weight.** The expected number of times a point appears is determined by its Gaussian weight. The product of the two child weights separates into a Gaussian in their difference and one in their midpoint. Averaging over a shared random shift integrates the midpoint term. The two child lists are sampled independently for the chosen shift. An acceptance probability then adjusts the weight of the difference. Each run aims for one range of output norms, so all intermediate lists can use spherical shells chosen for that range.

2. **Limit the number of pairs we have to examine.** The first extension makes the weighted counts stable over the random shift. Its index is chosen using an upper bound U on the Gaussian mass of $\mathcal{L}$, which also controls the expected length of the final list. A dense lattice at the other end of the chain permits sampling by Babai rounding. At each intermediate merge, spherical filters identify candidate pairs, and at the final merge we match coset labels. These searches give the time and space exponents in the theorem.



(4.1)

3. **Correct the choice from the final list.** A point may appear more than once. Choosing a random entry gives a point more weight when its occurrences come from short lists (Figure 4). The expected number of occurrences alone therefore does not give the desired sampling probability. To measure the extra factor, we make an auxiliary run in which one occurrence of that point is fixed in advance. The resulting list supplies a randomized test of the factor. Repeating the test allows us to cancel the factor to any prescribed accuracy.

More precisely, fix an output bin S_j and a mass bound $U \geq \rho_s(\mathcal{L})$. The probability of proposing $\mathbf{z} \in S_j$ is $\rho_s(\mathbf{z})\varpi_{\mathbf{z}}/(4U)$, where $\varpi_{\mathbf{z}} \in [1/2, 1]$ is the selection factor. The test returns 1 with probability $\varpi_{\mathbf{z}}$. We draw i uniformly from $\{0, \ldots, m-1\}$ and keep the proposal only if i independent tests all

return 0. Choosing the bin uniformly among J possibilities gives

$$\frac{1}{J}\cdot\frac{\rho_s(\mathbf{z})}{4U}\,\varpi_{\mathbf{z}}\cdot\frac{1-(1-\varpi_{\mathbf{z}})^m}{m\varpi_{\mathbf{z}}}=\frac{\rho_s(\mathbf{z})}{4JUm}(1-(1-\varpi_{\mathbf{z}})^m), \tag{4.2}$$

for each $\mathbf{z}$ in the sampled bins. The factor in parentheses differs from one by at most 2^{-m}. The bins omit only a small Gaussian tail, and trying successively smaller bounds U removes the need to know the Gaussian mass in advance. Choosing $m=\kappa+3$ gives the stated total variation error.

Roadmap. Section 4.1 describes the algorithm. Section 4.2 derives its sampling guarantee from the list counts and selection probabilities. Section 4.3 constructs the chain and lists, and Section 4.4 proves the selection correction. Section 4.5 proves the list-count and running-cost bounds. Section 4.6 combines these results to prove the theorem.

4.1 The sampler

This subsection describes the sampler at the level of its subroutines and states the properties of the recursion that it uses. Sections 4.3–4.4 give the details and the proofs.

Normalization. Scaling $\mathcal{L}$ and s by the same factor scales $D_{\mathcal{L},s}$, so Algorithm 2 rescales its input to $\det\mathcal{L}=1$. From now on, $\det\mathcal{L}=1$, $\mathcal{L}=\mathbf{B}\mathbb{Z}^n$ and $\sigma_s=s/\sqrt{2\pi}$. The lemmas below hold for all sufficiently large n.[2]

Radial bins and target weights. The bins are $S_j=\{\mathbf{z}\in\mathcal{L}: js^2\le\|\mathbf{z}\|^2<(j+1)s^2\}$, $j\ge 0$, with midpoints $r_j^2=(j+\frac{1}{2})s^2$. Attempt chooses j. A run is defined for every positive U; only the analysis assumes $U\ge\rho_s(\mathcal{L})$, until Section 4.2.1 removes this assumption. The *target weight* of $\mathbf{z}$ in the root list is its intended expected number of occurrences.

$$\ell_{\mathrm{out}}(\mathbf{z})=\frac{\rho_s(\mathbf{z})}{4U}\mathbf{1}_{S_j}(\mathbf{z}),\qquad\qquad\sum_{\mathbf{z}}\ell_{\mathrm{out}}(\mathbf{z})\le\frac{1}{4}, \tag{4.3}$$

The upper bound follows from $\rho_s(S_j)\le\rho_s(\mathcal{L})\le U$.

Runs and root lists. Fix a bin S_j and a mass bound U. A *run* $\mathsf{Run}(j,U)$ (Algorithm 6) first prepares a fresh superlattice chain $\mathcal{L}\subset\Gamma_0\subset\cdots\subset\Gamma_D$ as in (4.1). It generates lists from the leaves to the root of a complete binary tree (Figure 3). Each leaf on a coset of Γ_D draws N_{prop} independent proposals by rounding uniform points of an enlarged shell to that coset, then accepts each rounded point in the shell with probability proportional to the Gaussian density ϕ_τ. A merge takes two child lists on cosets of Γ_{d+1} and keeps differences in the parent coset and shell with acceptance probability α_d. The *root merge* takes two lists on the same coset of Γ_0, keeps differences in $\mathcal{L}\cap S_j$ with acceptance probability α_{out}, and outputs the *root list* L. Lists store occurrences, so a vector may appear more than once. A run fails if preparation fails or it exceeds a resource limit. Section 4.3 specifies these operations.

[2] Let n_0 be an absolute constant such that every statement of this section asserted for sufficiently large n holds for all $n\ge n_0$. It exists because there are finitely many such statements, and none of their thresholds depends on $\mathcal{L}$, s or κ. The proof of Theorem 1.4 handles $n<n_0$ by embedding $\mathcal{L}$ into $\mathcal{L}\oplus\mathbb{Z}^{n_0-n}$.

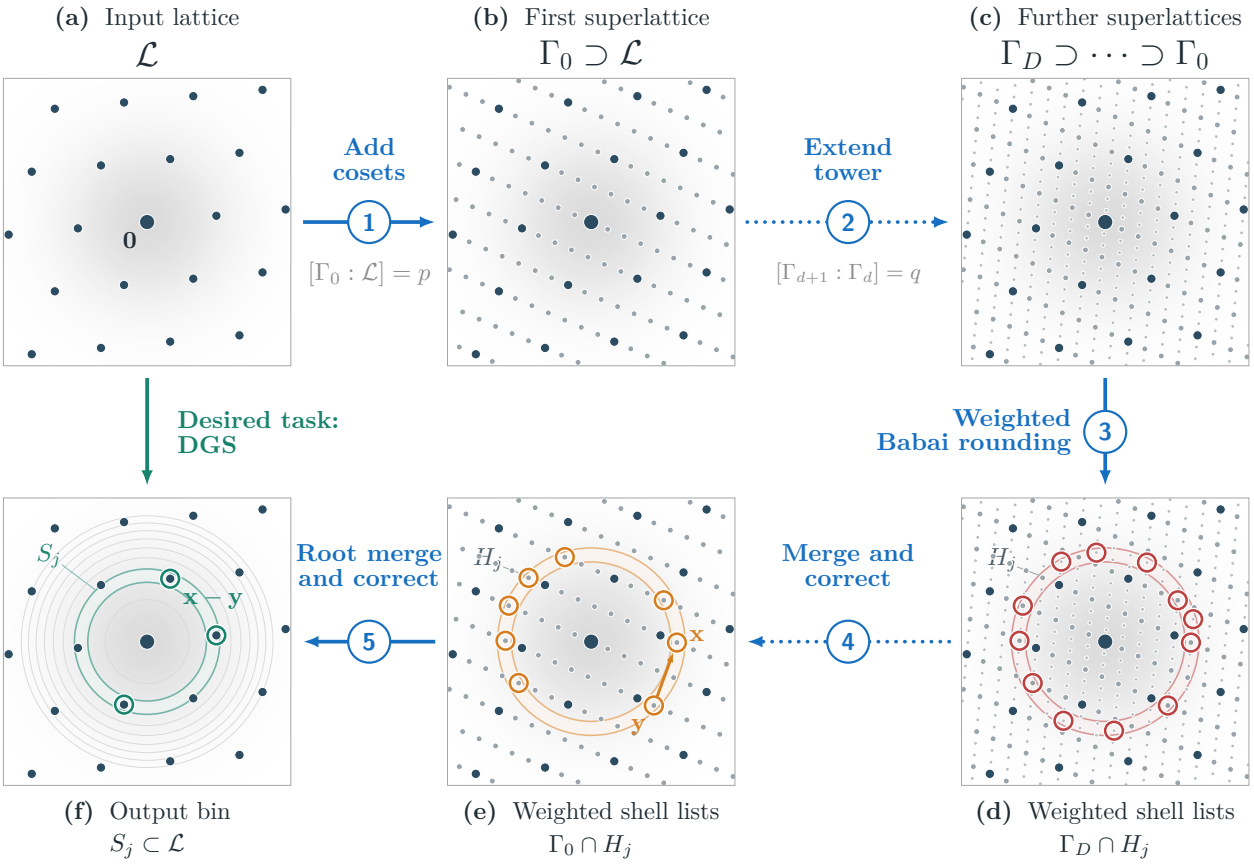


Figure 3: Gaussian-weighted recursion for one output bin S_j. The chosen bin fixes the widths and the common shell H_j for lists on $\Gamma_0, \ldots, \Gamma_D$. Red and orange rings show entries inside H_j. The last merge in (f) keeps differences in S_j and assigns them target weights proportional to $\rho_s(\mathbf{z})\mathbf{1}_{S_j}(\mathbf{z})$. A separate test corrects the bias of choosing an entry from this list. Green rings show possible outputs and faint circles show the other bins.

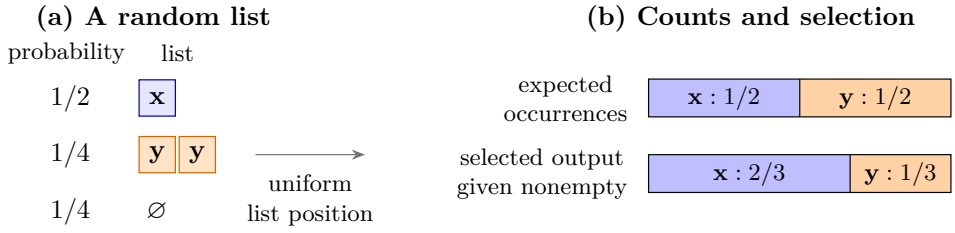


Figure 4: Expected occurrence counts do not determine the selected output. (a) A toy root list L equals $[\mathbf{x}]$ with probability $1/2$, $[\mathbf{y}, \mathbf{y}]$ with probability $1/4$, and is empty otherwise. (b) Both points have expected count $1/2$, but an entry of L chosen uniformly at random is $\mathbf{x}$ with probability $1/2$ and $\mathbf{y}$ with probability $1/4$, that is, $2/3$ and $1/3$ conditioned on a nonempty list, because the two occurrences of $\mathbf{y}$ share one list. Empty lists give failure. Hence $\varpi_{\mathbf{x}} = 1$ and $\varpi_{\mathbf{y}} = 1/2$ in Lemma 4.1.

Algorithm 2 Centered DGS at arbitrary width $\mathsf{DGS}(\mathbf{B}, s^2, \kappa)$

Input: a rational basis $\mathbf{B}$, rational $s^2 > 0$, and integer $\kappa \geq 1$.
Output: a point whose distribution is within $2^{-\kappa}$ of $D_{\mathcal{L},s}$, as in Theorem 1.4.

1: $m \leftarrow \kappa + 3$, $J \leftarrow n + m$, $M \leftarrow 16Jm^2$; compute K as in Section 4.2.1.
2: $\zeta \leftarrow |\det \mathbf{B}|^{-1/n}$; replace $\mathbf{B}$ by $\zeta\mathbf{B}$ and s^2 by $\zeta^2 s^2$.
3: **for** $U = 2^K, 2^{K-1}, \ldots, 1$ **do**
4: **for** M rounds **do**
5: $\mathbf{z} \leftarrow \mathsf{Attempt}(U, m)$.
6: **if** $\mathbf{z} \neq \mathsf{FAIL}$ **then**
7: **return** $\zeta^{-1}\mathbf{z}$.
8: **return 0**.

Algorithm 3 One attempt with rejection $\mathsf{Attempt}(U, m)$

Input: a mass bound $U > 0$ and an integer $m \geq 4$, with $J = n + m$. The basis $\mathbf{B}$ with $\det \mathcal{L} = 1$ and the squared width s^2 are the normalized ones of Algorithm 2.
Output: a point of $\mathcal{L}$, or FAIL.

1: Choose j uniformly from $\{0, \ldots, J-1\}$.
2: $L \leftarrow \mathsf{Run}(j, U)$; **return** FAIL if $L = \mathsf{FAIL}$ or L is empty. ▷ $\mathsf{Candidate}(j, U)$
3: $\mathbf{z} \leftarrow$ an entry of L chosen uniformly at random. ▷ $\Pr = \ell_{\text{out}}(\mathbf{z})\varpi_{\mathbf{z}}$
4: Choose i uniformly from $\{0, \ldots, m-1\}$.
5: **for** i rounds **do** ▷ $\mathsf{SelectionTest}(\mathbf{z}, j, U)$, mean $\varpi_{\mathbf{z}}$
6: $L' \leftarrow \mathsf{Run}(j, U, \mathbf{z})$.
7: **if** $L' \neq \mathsf{FAIL}$ **then**
8: **return** FAIL with probability $1/|L'|$.
9: **return** $\mathbf{z}$.

Each call of Run (Algorithms 6 and 8) resets its resource counters and prepares a fresh chain.

Runs with a fixed entry. For $\mathbf{z} \in S_j$, a *test run* $\mathsf{Run}(j, U, \mathbf{z})$ (Algorithm 8) fixes one occurrence of $\mathbf{z}$ in advance. At the root merge, it draws a Gaussian midpoint, writes $\mathbf{z} = \mathbf{x} - \mathbf{y}$, and fixes those entries in the two child lists. It continues in this way down to one uniformly chosen proposal slot in each leaf list, accepts each fixed pair whenever it finds it, and generates all other entries as a run does. It fails if a fixed value leaves the shell or the pair search of a merge misses a fixed pair. A successful test run therefore returns a root list that contains $\mathbf{z}$. Section 4.4 gives the details.

The two subroutines. Lines 2–3 of Attempt (Algorithm 3) form the subroutine $\mathsf{Candidate}(j, U)$, which performs a run and returns a uniformly random entry of its root list, the *candidate*. In one round of the loop, $\mathsf{SelectionTest}(\mathbf{z}, j, U)$ performs a test run for $\mathbf{z}$ and returns 1 with probability $1/|L|$ for its root list L (Figure 5). Each run uses a fresh chain, so the selection tests are independent given the candidate $\mathbf{z}$, the bin j and U. A selection test returns 0 if its run fails. Attempt chooses a bin j uniformly, takes one candidate $\mathbf{z}$, draws i uniformly from $\{0, \ldots, m-1\}$, and keeps $\mathbf{z}$ unless one of i selection tests returns 1. This rejection step cancels the selection factor, as in (4.2). DGS (Algorithm 2) makes up to M attempts for each mass bound $U = 2^K, 2^{K-1}, \ldots, 1$ and returns the first successful output.

Properties of root lists. Fix a returned chain and run the list generation that follows the preparation trial without the resource limits. Let $A_{\mathbf{z}}$ be the number of occurrences of $\mathbf{z}$ in its root list L. For $U \geq \rho_s(\mathcal{L})$, an attempt needs three properties.

(i) On every returned chain, each $\mathbf{z} \in S_j$ has $\mathbb{E}A_{\mathbf{z}} = \rho_s(\mathbf{z})/(4U)$ up to a factor $1 - e^{-\Omega(n/\log^4 n)}$. Hence the expected length of L is at most $1/4$.

(ii) On every good returned chain (defined in Section 4.3.1), which a preparation trial produces with probability $1 - o(1)$, the expected pair counts $\mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{z}'} - \mathbf{1}_{\mathbf{z}=\mathbf{z}'})]$ are at most $1 + e^{-\Omega(n/\log^4 n)}$ times the products of the target weights, as for independent Poisson counts.

(iii) For every nonnegative function F of the run record,

$$\mathbb{E}\big[A_{\mathbf{z}} F(\omega)\big] = \frac{\rho_s(\mathbf{z})}{4U}\, \mathbb{E}\big[\mathbf{1}_{\text{success}} F(\omega^{(\mathbf{z})})\big], \tag{4.4}$$

where ω and $\omega^{(\mathbf{z})}$ are ordinary and test run records, respectively. The right-hand side is zero when the test run fails. This is the case $f(\omega, i) = F(\omega)$ of Lemma 4.7 at the root. The identity also holds after the preparation trial is averaged in, with preparation failure contributing zero to both sides, and F may contain the indicator that both resource limits are respected.

Take for F the reciprocal length $1/|L|$, defined as zero when L is empty, times the indicator that both resource limits are respected. The left-hand side of (4.4) is then $\Pr[\mathsf{Candidate}(j, U) = \mathbf{z}]$, and the expectation on the right is the mean $\varpi_{\mathbf{z}}$ of $\mathsf{SelectionTest}(\mathbf{z}, j, U)$. Hence

$$\Pr[\mathsf{Candidate}(j, U) = \mathbf{z}] = \frac{\rho_s(\mathbf{z})}{4U}\, \varpi_{\mathbf{z}}$$

exactly. The approximation in (i) enters only through $\varpi_{\mathbf{z}}$. By (i) and (ii), a test run on a good chain returns the list $[\mathbf{z}]$ with probability at least $3/4 - o(1)$, so $\varpi_{\mathbf{z}} \geq 1/2$ (Lemma 4.1). Lemma 4.6 in Section 4.3 proves (i) and (ii), and Lemma 4.7 in Section 4.4 proves (iii).

4.2 The guarantee of the sampler

By property (i) of Section 4.1, on every returned chain, list generation without the resource limits produces root lists in which each $\mathbf{z} \in S_j$ occurs $\ell_{\text{out}}(\mathbf{z}) = \rho_s(\mathbf{z})/(4U)$ times in expectation, up to a factor $1 - e^{-\Omega(n/\log^4 n)}$. This subsection proves that Algorithms 2 and 3 (Section 4.1) turn such lists into one sample from $D_{\mathcal{L},s}$.

Candidates and selection tests. Each subroutine call prepares its own chain, so the selection tests in one call of Attempt are independent given the candidate $\mathbf{z}$, its bin, and U. Figure 5 compares the two runs.

The main result. The next two lemmas give the exact distribution of Attempt. The first one defines the *selection factor* $\varpi_{\mathbf{z}}$.

Lemma 4.1 (Selection factor)**.** *Let $U \geq \rho_s(\mathcal{L})$, fix a bin S_j and let $\varpi_{\mathbf{z}} = \Pr[\mathsf{SelectionTest}(\mathbf{z}, j, U) = 1]$. Then for every $\mathbf{z} \in S_j$, $\Pr[\mathsf{Candidate}(j, U) = \mathbf{z}] = \ell_{\text{out}}(\mathbf{z})\varpi_{\mathbf{z}}$ and $\frac{1}{2} \leq \varpi_{\mathbf{z}} \leq 1$.*

Lemma 4.2 (One attempt with rejection)**.** *Let $J = n + m$ and let $w(\mathbf{z}) = \rho_s(\mathbf{z})/(4JU)$ for $\mathbf{z} \in \bigcup_{j<J} S_j$ and $w(\mathbf{z}) = 0$ otherwise. For $U \geq \rho_s(\mathcal{L})$, Algorithm 3 has the exact distribution*

$$\Pr[\mathsf{Attempt} = \mathbf{z}] = \frac{w(\mathbf{z})}{m}\left(1 - (1 - \varpi_{\mathbf{z}})^m\right), \tag{4.5}$$

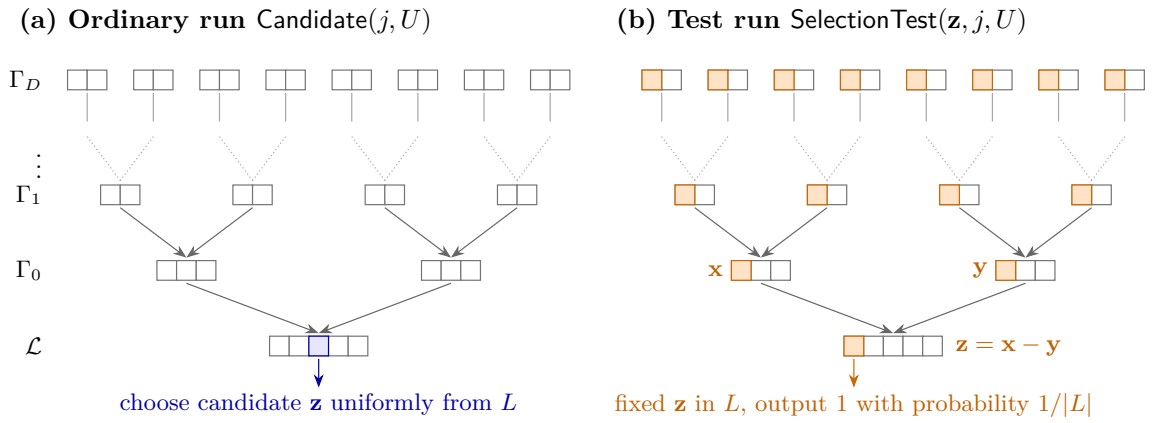


Figure 5: Candidate calls and selection tests. Each run starts from a fresh chain $\mathcal{L} \subset \Gamma_0 \subset \cdots \subset \Gamma_D$ and combines the lists on cosets of Γ_D into a root list L on $\mathcal{L}$ with entries in S_j. In (a), the candidate call obtains L from Algorithm 6 and chooses an entry uniformly if L is nonempty. The probability of obtaining $\mathbf{z}$ is $\ell_{\text{out}}(\mathbf{z})\varpi_{\mathbf{z}}$ (Lemma 4.1). In (b), Algorithm 8 fixes one occurrence of $\mathbf{z}$ at the root and one entry in each child list used to produce it (orange). The other entries are generated as in (a). The selection test returns 1 with probability $1/|L|$ when the run succeeds and returns 0 if it fails. Its mean is $\varpi_{\mathbf{z}}$. Algorithm 3 uses both calls.

where $\varpi_{\mathbf{z}}$ is the selection factor for the bin containing $\mathbf{z}$, and the right-hand side is zero outside $\bigcup_{j<J} S_j$. Conditional on success, the output is within $3 \cdot 2^{-m}$ of $D_{\mathcal{L},s}$ in total variation. If $\rho_s(\mathcal{L}) \le U \le 2\rho_s(\mathcal{L})$, its success probability is at least $1/(16Jm)$.

Lemma 4.1 is an identity. The error of Lemma 4.6 in $\mathbb{E}A_{\mathbf{z}}$ enters only $\varpi_{\mathbf{z}}$, through the success probability of the test run.

Lemma 4.2 follows from Lemma 4.1 and the rejection step, as in (4.2). This rejection step is similar to the square-root sampler of ADRS [ADRS15, Claim 5.2], which builds a Bernoulli variable with mean $\sqrt{p}$ from Bernoulli variables with unknown mean p. Section 4.4 proves both lemmas.

The radial tail. The error bound of Lemma 4.2 includes the Gaussian mass outside the sampled bins $S_0 \cup \cdots \cup S_{J-1}$. The centered case of (2.9) bounds this mass by

$$\Pr_{D_{\mathcal{L},s}}[\|\mathbf{X}\|^2 \ge Js^2] \le 2^{n/2} e^{-\pi J/2} \le 2^{-m}, \tag{4.6}$$

since the logarithm of the middle term is $n(\ln 2 - \pi)/2 - \pi m/2 \le -m \ln 2$.

Costs. For every positive U, Attempt makes at most m subroutine calls. Each preparation costs $2^{O(n/h)}$ (Lemma 4.9). Section 4.5.2 caps the work of each run at $W_0 = (3/2)^{n/2} 2^{O(n/\log n)}$ and its space at $(4/3)^{n/2} 2^{O(n/\log n)}$. The calls reuse their workspace, so

$$\mathrm{T} \le (3/2)^{n/2} 2^{O(n/\log n)} \operatorname{poly}(m), \qquad \mathrm{S} \le (4/3)^{n/2} 2^{O(n/\log n)} \operatorname{poly}(m). \tag{4.7}$$

4.2.1 Unknown Gaussian mass

Lemmas 4.1 and 4.2 assume $U \ge \rho_s(\mathcal{L})$, but $\rho_s(\mathcal{L})$ is unknown. Algorithm 2 therefore tries $U = 2^K, 2^{K-1}, \ldots, 1$ with $K = n\lceil \log_2(2 + s^2 \|\mathbf{B}^{-1}\|_F^2) \rceil$. The first bound is valid. The norm

comparison $\|\mathbf{Bz}\| \geq \|\mathbf{z}\|/\|\mathbf{B}^{-1}\|_F$ and (2.12) give

$$1 \leq \rho_s(\mathcal{L}) \leq \rho_{s\|\mathbf{B}^{-1}\|_F}(\mathbb{Z})^n \leq (1 + s\|\mathbf{B}^{-1}\|_F)^n \leq \left(2 + s^2\|\mathbf{B}^{-1}\|_F^2\right)^n \leq 2^K, \tag{4.8}$$

using $1 + x \leq 2 + x^2$ for $x \geq 0$. Since $s^2\|\mathbf{B}^{-1}\|_F^2$ does not change when $\mathbf{B}$ and s are scaled together, K can be computed from the original rational input. The search stops at the first bound at which an attempt succeeds. Once $U < \rho_s(\mathcal{L})$, the guarantee of Lemma 4.2 no longer holds, but the search reaches such bounds only with probability at most 2^{-m}.

Lemma 4.3 (Descending mass bounds)**.** *Algorithm 2 has total variation error at most* $4 \cdot 2^{-m}$*, where* $m = \kappa + 3$*. On every execution, its time* T *and space* S *satisfy*

$$\mathrm{T} \leq (3/2)^{n/2} 2^{O(n/\log n)} \operatorname{poly}(\kappa), \qquad \mathrm{S} \leq (4/3)^{n/2} 2^{O(n/\log n)} \operatorname{poly}(\kappa).$$

The dimension threshold n_0 *does not depend on* κ*.*

Proof. At each proposed bound, Algorithm 2 makes M independent calls of Attempt and returns its first successful output. Let U_* be the last power of two in the sequence with $U_* \geq \rho_s(\mathcal{L})$. By (4.8), it exists and satisfies $\rho_s(\mathcal{L}) \leq U_* < 2\rho_s(\mathcal{L})$. Every call up to this bound has the distributional guarantee of Lemma 4.2. At U_*, each succeeds with probability at least $1/(16Jm)$. Thus the probability of no output by the end of the stage U_* is at most $(1 - 1/(16Jm))^M \leq e^{-m} \leq 2^{-m}$. The first successful output before this event is a mixture of distributions within $3 \cdot 2^{-m}$ of $D_{\mathcal{L},s}$. Indeed, the calls use fresh randomness, and whether a call is made depends only on the failures of earlier calls. Reaching an invalid bound or returning fallback zero contributes at most 2^{-m}. The total error is therefore at most $4 \cdot 2^{-m}$.

There are at most $(K + 1)M$ calls of Attempt, each supplied bound has at most $K + 1$ bits, and the calls reuse their workspace. Hence (4.7) gives the claimed bound on time and space. All radial, geometric, and preparation estimates are uniform in the bin and in m, so their dimension thresholds do not depend on κ. □

4.3 The chain and Gaussian-weighted lists

A run builds its lists from the leaves to the root. At each leaf we sample in a spherical shell. At each merge we look for pairs whose difference lies on the prescribed lattice translate and accept each such difference with a probability chosen to give the next list its Gaussian weights. The last merge targets the output bin. The algorithms below give these steps and Lemma 4.6 states the bounds on the resulting counts. The notation groups the quantities by their role in the construction.

Symbol	Meaning
S_j, H	output bin; shell of the nonroot lists of a run for this bin
$\sigma_s = s/\sqrt{2\pi}$	width of the target distribution
τ_1, τ	widths of the lists on Γ_0 and on Γ_d, $d \geq 1$ (σ_d in general)
ℓ_{out}, ℓ_d	target weights (expected multiplicities) of the root list and of a list on Γ_d
N	weight scale of the nonroot target weights
α_{out}, α_d	acceptance probabilities of a merge
N_{prop}, W_0, M_0	proposals per leaf, work limit, and cap per output list or pair dictionary

Algorithm 4 Gaussian leaf sampling $\mathsf{GaussSample}(\mathbf{c})$

Input: a shift $\mathbf{c}$.

$L \leftarrow [\,]$.
for $r = 1, \ldots, N_{\text{prop}}$ **do**
  Draw $\mathbf{V}$ uniformly from H^+ and let $\mathbf{x} \leftarrow \mathsf{NP}_{\mathbf{B}^{(D)}}(\mathbf{V} - \mathbf{c}) + \mathbf{c}$.
  if $\mathbf{x} \in H$ **then**
    Append $\mathbf{x}$ to L with probability $N \operatorname{vol}(H^+)\phi_\tau(\mathbf{x})/N_{\text{prop}}$.
return L.

Algorithm 5 Gaussian merge $\mathsf{GaussMerge}(d, \mathbf{c}, X, Y)$

Input: a level $0 \le d < D$, a target shift $\mathbf{c}$, and two lists X, Y on $\Gamma_{d+1} + \mathbf{u}$ and $\Gamma_{d+1} + \mathbf{u} - \mathbf{c}$.

$L \leftarrow [\,]$.
for each occurrence pair $(\mathbf{x}, \mathbf{y}) \in X \times Y$, at its first report by the pair enumerator **do**
  if $\mathbf{x} - \mathbf{y} - \mathbf{c} \in \Gamma_d$ and $\mathbf{x} - \mathbf{y} \in H$ **then** ▷ no midpoint test
    Append $\mathbf{x} - \mathbf{y}$ to L with probability $\alpha_d(\mathbf{x} - \mathbf{y})$.
return L.

Algorithm 6 A run $\mathsf{Run}(j, U)$

Input: a bin index j and a mass bound $U > 0$.

Output: a root list on $\mathcal{L}$ in S_j, or FAIL.

Run a preparation trial for bin j and bound U; **return** FAIL if it fails.
Resource limits: return FAIL before exceeding W_0 operations after preparation or storing more than M_0 entries in any output list or pair dictionary (Section 4.5.2).
Draw $\mathbf{u}$ uniformly modulo Γ_0 and let $\mathbf{u}_0^{(0)}, \mathbf{u}_1^{(0)} \leftarrow \mathbf{u}$.
for $d = 0, \ldots, D-1$ and $i = 0, \ldots, 2^{d+1} - 1$ **do** ▷ split shifts, from the root down
  Draw $\mathbf{u}_{2i}^{(d+1)}$ uniformly modulo Γ_{d+1} and let $\mathbf{u}_{2i+1}^{(d+1)} \leftarrow \mathbf{u}_{2i}^{(d+1)} - \mathbf{u}_i^{(d)}$.
$L_i^{(D)} \leftarrow \mathsf{GaussSample}(\mathbf{u}_i^{(D)})$ for $i = 0, \ldots, 2^{D+1} - 1$. ▷ leaf lists
for $d = D-1, \ldots, 0$ **do** ▷ merges, from the leaves up
  $L_i^{(d)} \leftarrow \mathsf{GaussMerge}(d, \mathbf{u}_i^{(d)}, L_{2i}^{(d+1)}, L_{2i+1}^{(d+1)})$ for $i = 0, \ldots, 2^{d+1} - 1$.
return $\mathsf{RootMerge}(L_0^{(0)}, L_1^{(0)})$.

Algorithm 7 Root merge $\mathsf{RootMerge}(X, Y)$

Input: two lists X, Y on the same translate of Γ_0.

$L \leftarrow [\,]$.
for each occurrence pair $(\mathbf{x}, \mathbf{y}) \in X \times Y$ with $\mathbf{x} - \mathbf{y} \in \mathcal{L}$, reported once **do**
  if $\mathbf{x} - \mathbf{y} \in S_j$ **then**
    Append $\mathbf{x} - \mathbf{y}$ to L with probability $\alpha_{\text{out}}(\mathbf{x} - \mathbf{y})$.
return L.

The chain and the tree. Set $h = \lceil \log_2(n+2) \rceil$, $D = h^3$ and $q = 2^{\lceil n/h \rceil}$. The chain starts with a random extension Γ_0 of prime index p and continues with D extensions of index q, so $[\Gamma_D : \mathcal{L}] = pq^D$. Section 4.5.1 specifies the first extension and the terminal basis test. A *preparation trial* builds the chain and may fail. A chain it outputs is a *returned chain*. Each call of Candidate or SelectionTest runs a fresh preparation trial.

For a node whose list lies on $\Gamma_d + \mathbf{c}$, with $d < D$, draw a fresh *split shift* $\mathbf{u}$ uniformly modulo Γ_{d+1}. Its children have cosets $\Gamma_{d+1} + \mathbf{u}$ and $\Gamma_{d+1} + \mathbf{u} - \mathbf{c}$. Thus a difference of child vectors already lies in $\Gamma_{d+1} + \mathbf{c}$, and a quotient-label test decides whether it lies in $\Gamma_d + \mathbf{c}$. At the root both child lists lie on $\Gamma_0 + \mathbf{u}$, with $\mathbf{u}$ uniform modulo Γ_0. The root merge tests for differences in $\mathcal{L} \cap S_j$.

Gaussian weights. Let ϕ_σ be the density of $N(\mathbf{0}, \sigma^2 \mathbf{I})$, so $\rho_s = s^n \phi_{\sigma_s}$. From now on we work with these normalized densities. Recall that $\sigma_s = s/\sqrt{2\pi}$ is the width of the target distribution. The lists on Γ_d have width σ_d, and σ denotes a generic child width. With these normalized densities, the midpoint factor in the product identity

$$\phi_\sigma(\mathbf{x})\phi_\sigma(\mathbf{y}) = \phi_{\sqrt{2}\sigma}(\mathbf{x} - \mathbf{y})\, \phi_{\sigma/\sqrt{2}}\left(\frac{\mathbf{x}+\mathbf{y}}{2}\right) \tag{4.9}$$

then has total mass one.

The child shell. All nonroot lists of a run lie in one shell H, so that a merge pairs vectors of about the same length. The shell depends on the bin S_j; Figure 3 writes H_j. With Gaussian weights, the length of a child depends on the length of its parent, and we choose the widths so that this length is the same at every level. Let the midpoint $\mathbf{w} = \mathbf{x} - \mathbf{z}/2$ of a child pair $(\mathbf{x}, \mathbf{x} - \mathbf{z})$ have density $\phi_{\sigma/\sqrt{2}}$, as (4.9) suggests. Then

$$\mathbb{E}\|\mathbf{x}\|^2 = \mathbb{E}\|\mathbf{x} - \mathbf{z}\|^2 = \frac{n\sigma^2}{2} + \frac{\|\mathbf{z}\|^2}{4}, \tag{4.10}$$

since the cross term has mean zero. In the coordinate $t = \|\cdot\|^2/(n\sigma^2)$, with σ the child width, this maps a parent value t to the child value $1/2 + t/4$, with fixed point $2/3$. Below Γ_0 we take $\tau^2 = \frac{3}{2}R^2$, so the children of a parent of squared norm about nR^2 again have mean squared norm nR^2. At the root, the parent lies in the output bin S_j, whose radius is set by the output and not by the shell. The lists on Γ_0 therefore get their own width τ_1, chosen from the bin midpoint so that the root children again have mean squared norm nR^2. The acceptance probability α_0 of the merge onto Γ_0 converts the child width τ into the parent width τ_1. For $r^2 = r_j^2$ and the slack $\varepsilon = h^{-2}$,

$$\tau_1^2 = \max\left\{\frac{3}{4}\sigma_s^2, \frac{3r^2}{2n}\right\}, \quad R^2 = \frac{\tau_1^2}{2} + \frac{r^2}{4n}, \quad \tau^2 = \frac{3}{2}R^2, \quad H = \left\{\mathbf{x} : \left|\frac{\|\mathbf{x}\|^2}{nR^2} - 1\right| \le \varepsilon\right\}. \tag{4.11}$$

Lists on Γ_0 have width $\sigma_0 = \tau_1$, and lists on Γ_d, $d \ge 1$, have width $\sigma_d = \tau$. The first term of τ_1^2 gives $\tau_1, \tau \ge 3\sigma_s/4$ (Section 4.5). The proof of Lemma 4.8 needs this to bound the periodization second moment by the dual Gaussian mass $\rho_{1/s}(\mathcal{L}^*)$. The second term places the root children in the shell. Figure 6 shows the root and nonroot cases.

Weights and acceptance probabilities. The nonroot weights are $\ell_d = N \det \Gamma_d\, \phi_{\sigma_d} \mathbf{1}_H$ for $0 \le d \le D$, and the root weight is (4.3). The weight scale $N = \lceil q^2 2^{n/2} e^{-n/6} \rceil$ controls the target multiplicities. It is chosen so that two representations of a parent that share a child contribute only $2^{-\Omega(n/\log n)}$ to the second moments (Lemma 4.4 below): the integral in the second bound of

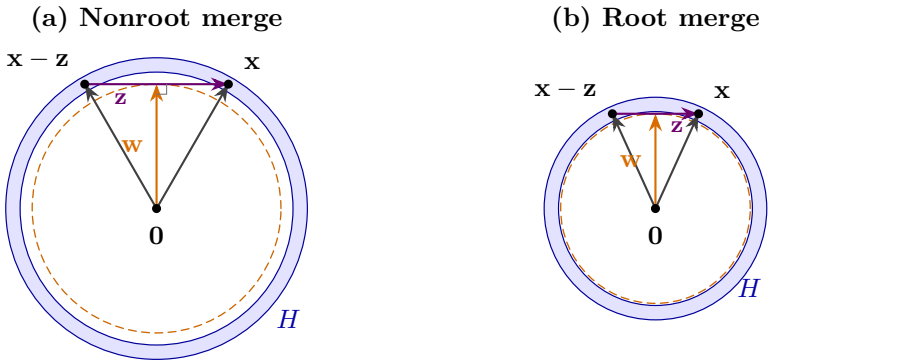


Figure 6: Shells in a Gaussian-weighted merge. A pair $(\mathbf{x}, \mathbf{x}-\mathbf{z})$ has midpoint $\mathbf{w} = \mathbf{x} - \mathbf{z}/2$. Under the midpoint density $\phi_{\sigma/\sqrt{2}}$, its mean squared norm is $n\sigma^2/2$ (dashed circle). Each child then has mean squared norm $n\sigma^2/2 + \|\mathbf{z}\|^2/4$, close to the shell center nR^2. In (a), an internal merge has $\mathbf{z} \in H$ and $\tau^2 = \frac{3}{2}R^2$. In (b), the last merge has $\mathbf{z} \in S_j$ and $\|\mathbf{z}\|^2 \approx r^2$. The width and shell radius are chosen so that the children fall in H with high probability. Both panels exaggerate shell widths.

that lemma is at most $2^{n/2}e^{-n/6+O(n\varepsilon)}$, since the shell radius satisfies $R^2 \le \frac{2}{3}\sigma^2$ for the child width σ, and dividing by N leaves $q^{-2}e^{O(n\varepsilon)}$. With this choice, a nonroot list has expected length at most $(4/3)^{n/2}2^{O(n/\log n)}$ over a uniform shift (Section 4.5.3). A merge keeps a difference $\mathbf{z}$ with the acceptance probability

$$\alpha_{\mathrm{out}}(\mathbf{z}) = \frac{\ell_{\mathrm{out}}(\mathbf{z})}{N^2 \det\Gamma_0\, \phi_{\sqrt{2}\tau_1}(\mathbf{z})}, \qquad \alpha_d(\mathbf{z}) = \frac{\ell_d(\mathbf{z})}{N^2 \det\Gamma_{d+1}\, \phi_{\sqrt{2}\sigma_{d+1}}(\mathbf{z})} \quad (0 \le d < D). \tag{4.12}$$

These probabilities turn the target weights of the children into the target weight of the parent. To see this, note that for a child width σ, (4.9) writes the weight $\phi_\sigma(\mathbf{x})\phi_\sigma(\mathbf{x}-\mathbf{z})$ of a child pair as $\phi_{\sqrt{2}\sigma}(\mathbf{z})$ times $\phi_{\sigma/\sqrt{2}}(\mathbf{x}-\mathbf{z}/2)$. Define

$$G_\sigma(\mathbf{w}) = \phi_{\sigma/\sqrt{2}}(\mathbf{w})\mathbf{1}_{\{|\|\mathbf{w}\|^2/(n\sigma^2)-1/2|\le 4\varepsilon\}}, \qquad g_{\sigma,\mathbf{z}}(\mathbf{x}) = G_\sigma(\mathbf{x}-\mathbf{z}/2)\mathbf{1}_H(\mathbf{x})\mathbf{1}_H(\mathbf{x}-\mathbf{z}).$$

Thus $g_{\sigma,\mathbf{z}}$ is the midpoint factor, restricted to pairs with both children in H and midpoint within the norm cutoff. The cutoff keeps the midpoint from being short, which the second bound of Lemma 4.4 uses. It holds automatically when both children and their difference lie in the shell, so the algorithm never tests it.

Fix the split shift $\mathbf{u}$ and a parent $\mathbf{z}$ at level d, and let $\sigma = \sigma_{d+1}$ and $g_\mathbf{z} = g_{\sigma,\mathbf{z}}$. If both child lists had expected counts equal to their target weights, then, since they are independent, (4.9) would give the expected number of child pairs with difference $\mathbf{z}$ as

$$\sum_{\mathbf{x}\in\Gamma_{d+1}+\mathbf{u}} \ell_{d+1}(\mathbf{x})\ell_{d+1}(\mathbf{x}-\mathbf{z}) = N^2(\det\Gamma_{d+1})^2\, \phi_{\sqrt{2}\sigma}(\mathbf{z}) \sum_{\mathbf{x}\in\Gamma_{d+1}+\mathbf{u}} g_\mathbf{z}(\mathbf{x}),$$

since the midpoint cutoff in $g_\mathbf{z}$ holds whenever both children and their difference lie in H (Lemma 4.4). Averaging over $\mathbf{u}$ replaces the last sum by $\int g_\mathbf{z}/\det\Gamma_{d+1}$, by (2.17). Multiplying by $\alpha_d(\mathbf{z})$ gives exactly $\ell_d(\mathbf{z})\int g_\mathbf{z}$, and α_{out} does the same at the root. Lemma 4.4 below shows $\int g_\mathbf{z} \ge 1 - e^{-\Omega(n/h^4)}$. The actual child lists only approximate their target weights, and pair enumeration may miss pairs. Lemma 4.6 accounts for both and runs this computation without the resource limits. The acceptance probabilities α_d and α_{out} are at most one by Lemma 4.10(ii)–(iii).

Runs. Algorithm 6 generates the merge tree level by level. It draws all split shifts from the root downward, samples the leaf lists by Algorithm 4, and builds the other lists from the leaves upward by Algorithms 5 and 7. Section 4.4 gives the test run (Algorithm 8). For the analysis we also view a run as a recursion. For a node whose list lies on $\Gamma_d + \mathbf{c}$, let $\mathsf{DGSList}(d, \mathbf{c})$ denote the part of the run below the node. For $d = D$, it is $\mathsf{GaussSample}(\mathbf{c})$. For $d < D$, it draws the split shift $\mathbf{u}$ uniformly modulo Γ_{d+1}, generates the child lists by $\mathsf{DGSList}(d+1, \mathbf{u})$ and $\mathsf{DGSList}(d+1, \mathbf{u} - \mathbf{c})$, and merges them by $\mathsf{GaussMerge}$. The root is the level root, with $\mathbf{c} = \mathbf{0}$ and both children generated by $\mathsf{DGSList}(0, \mathbf{u})$, and $\mathsf{DGSList}(\mathsf{root}, \mathbf{0})$ is a run after its preparation trial. Algorithms 6 and 8 run the tree level by level, while the proofs in Sections 4.3 and 4.4 argue by induction over the tree, so we check that the two views agree. Since the split shifts are independent, the two descriptions give the same distribution. For test runs, the split shifts depend on the fixed values of the ancestors, but the Gaussian midpoints are independent, and the two child subtrees of a node are independent given their fixed values. On successful unlimited executions, assigning randomness to the nodes gives the same record in either order. Corresponding operations have the same charges, and the sizes of all lists and dictionaries agree. Both orders therefore give the same event of respecting the resource limits. Every other such execution fails in both orders, possibly after different partial work. The target set is S_j at the root and $(\Gamma_d + \mathbf{c}) \cap H$ at level d.

In Algorithms 4–8, H^+ is H enlarged radially by r_H/n, where $r_H = \sqrt{n}\, R$, and $\mathsf{NP}_{\mathbf{B}^{(D)}}$ is Babai's nearest-plane rounding in the terminal basis. A pair $(\mathbf{x}, \mathbf{y}) \in X \times Y$ is *compatible* if $\mathbf{x} - \mathbf{y} \in \Gamma_d + \mathbf{c}$, with $\mathcal{L}$ in place of Γ_d at the root. The quotient labels $\mathbf{x} - \mathbf{u} + \Gamma_d$ and $\mathbf{y} - \mathbf{u} + \mathbf{c} + \Gamma_d$ decide this. The pair search works differently at the root. There we sort the labels and report each compatible pair once. At any other merge, the spherical-cap search from Lemma 2.19 reports candidate pairs. It may report a pair several times, and it misses a compatible pair with difference in H with probability at most 2^{-n} (Lemma 4.11). Only the first report of a pair receives an acceptance decision. Missed pairs lower the expected counts, and the lower bound of Lemma 4.6 includes this loss. A run returns FAIL before exceeding its work or storage limits. A test run also fails at the shell and coverage tests of Algorithm 8. Section 4.5.2 fixes the proposal count, the pair enumeration and both limits.

4.3.1 Moments of point multiplicities

Lemma 4.6 combines two inputs: a bound for one merge, which does not depend on the chain (Lemma 4.4), and a property of the chain, which holds for almost every returned chain (good chains, below).

Lemma 4.4 (Child shell)**.** *At a root merge with parent in S_j, or a nonroot merge with parent in H, the function $g_{\mathbf{z}} = g_{\sigma,\mathbf{z}}$, with $\sigma = \tau_1$ at the root and $\sigma = \tau$ at nonroot merges, satisfies*

$$\int g_{\mathbf{z}} \ge 1 - e^{-\Omega(n/h^4)}, \qquad \frac{1}{N} \int \frac{g_{\mathbf{z}}(\mathbf{x})^2}{\phi_\sigma(\mathbf{x})}\, d\mathbf{x} \le q^{-2} e^{O(n\varepsilon)}.$$

The first bound says that the restrictions lose only an $e^{-\Omega(n/h^4)}$ fraction of the midpoint mass. The second bounds the contribution of two representations of a parent that share one child entry, relative to the product of the parent weights. If both child lists have the expected counts specified by their weights, a child entry $\mathbf{x}$ paired with both $\mathbf{x} - \mathbf{z}$ and $\mathbf{x} - \mathbf{z}'$ contributes, after division by $\ell_d(\mathbf{z})\ell_d(\mathbf{z}')$, at most $N^{-1} \int g_{\mathbf{z}} g_{\mathbf{z}'}/\phi_\sigma$, which the Cauchy–Schwarz inequality reduces to the second bound. Since $q^{-2} = 2^{-2\lceil n/h \rceil}$ and $n\varepsilon = n/h^2$, this bound is $2^{-\Omega(n/\log n)}$. It fixes the weight scale N. By the last step of the proof (Section 4.5.3), the midpoint cutoff holds whenever both children lie in H and their difference lies in H, or in S_j at the root. Algorithm 6 therefore needs no midpoint test.

Good chains. The second moments of the lists depend on how much the number of representations of a parent fluctuates with the split shift. For a parent $\mathbf{z}$ and a uniform shift $\mathbf{c}$ of the child cosets, the periodization $P_{\Lambda,g_{\mathbf{z}}}(\mathbf{c})$ of (2.16) over the child lattice Λ is the normalized weighted number of child pairs with difference $\mathbf{z}$. Its mean over $\mathbf{c}$ is close to one for every $\mathbf{z}$ (Lemma 4.4), and its second moment $R_\Lambda(g_{\mathbf{z}})$ measures how much it fluctuates. A returned chain should keep these second moments at most $1+e^{-\Omega(n/h^4)}$ at every merge. Since $g_{\mathbf{z}} \le G_\sigma(\cdot - \mathbf{z}/2)$ and R_Λ is monotone and translation invariant (2.24), it suffices to bound $R_{\Gamma_d}(G_{\sigma_d})$ on every lattice of the chain.

Definition 4.5 (Good chain)**.** Let $c>0$ be the absolute constant of Lemma 4.8. A returned chain is *good* if

$$R_{\Gamma_d}(G_{\sigma_d}) \le 1 + e^{-cn/h^4} \qquad (0 \le d \le D). \tag{4.13}$$

Lemma 4.8 shows that a random first extension makes the chain good with high probability, and a preparation trial returns a good chain with probability $1-o(1)$ (Lemma 4.9). The lattice $\mathcal{L}$ itself does not work as Γ_0. By Poisson summation, $R_\Lambda(g) = \sum_{\mathbf{y}\in\Lambda^*} |\widehat{g}(\mathbf{y})|^2$, and the term $\mathbf{y}=\mathbf{0}$ is $(\int g)^2 \approx 1$. For an arbitrary $\mathcal{L}$, the dual $\mathcal{L}^*$ may have many short vectors, and the only available bound on the other terms is exponentially large in n (4.19). The dual of a random index-p extension Γ_0 is a random index-p sublattice of $\mathcal{L}^*$ containing $p\mathcal{L}^*$. It keeps each dual vector outside $p\mathcal{L}^*$ with probability about $1/p$, and the vectors of $p\mathcal{L}^*$ are p times longer. Choosing a large prime p therefore reduces the contribution of the nonzero terms.

The moment bounds. The expectation $\mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{z}'} - \mathbf{1}_{\mathbf{z}=\mathbf{z}'})]$ counts ordered pairs of distinct occurrences. For independent Poisson counts with means $\ell(\mathbf{z})$, it equals $\ell(\mathbf{z})\ell(\mathbf{z}')$.

Lemma 4.6 (Moments of point multiplicities)**.** *Fix a returned chain and run* $\mathsf{DGSList}(d,\mathbf{c})$ *without the resource limits. Let* $A_{\mathbf{z}}$ *count occurrences of* $\mathbf{z}$ *in the list it returns on* $\Gamma_d+\mathbf{c}$ *(at the root, use* $\mathcal{L}$ *and* ℓ_{out} *in place of* $\Gamma_d+\mathbf{c}$ *and* ℓ_d*, respectively). Then*

$$(1-e^{-\Omega(n/h^4)})\ell_d(\mathbf{z}) \le \mathbb{E}A_{\mathbf{z}} \le \ell_d(\mathbf{z}), \qquad \mathbf{z}\in\Gamma_d+\mathbf{c}. \tag{4.14}$$

If the chain is good, then

$$\mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{z}'} - \mathbf{1}_{\mathbf{z}=\mathbf{z}'})] \le (1+e^{-\Omega(n/h^4)})\ell_d(\mathbf{z})\ell_d(\mathbf{z}'), \qquad \mathbf{z},\mathbf{z}'\in\Gamma_d+\mathbf{c}.$$

Both bounds are uniform in the coset shift.

The lower bound in (4.14) is not an equality. This does not affect the output distribution, because the error enters only the selection factor of Section 4.2. Section 4.5.3 proves Lemma 4.6 by induction over the merge tree. It uses the leaf and pair-enumeration bounds in Section 4.5.2.

The proofs use ℓ_d only through the cancellation $\alpha_d(\mathbf{z})N^2 \det\Gamma_{d+1}\,\phi_{\sqrt{2}\sigma_{d+1}}(\mathbf{z}) = \ell_d(\mathbf{z})$ of (4.12). The root merge, with ℓ_{out}, α_{out}, child lattice Γ_0 and child width τ_1, satisfies the same cancellation. Hence neither the root nor the change of width from τ_1 to τ below Γ_0 needs a separate moment argument.

4.4 Fixing an entry and correcting the selection probability

This subsection proves Lemmas 4.1 and 4.2. The tool is the reweighting identity of Lemma 4.7. A test run fixes one occurrence of $\mathbf{z}$ and generates the rest of the execution around it. The identity states that its successful executions follow the distribution of $\mathsf{DGSList}(d,\mathbf{c})$ weighted by the number

Algorithm 8 A test run $\mathsf{Run}(j, U, \mathbf{z})$

Input: a bin index j, a mass bound $U > 0$ and a point $\mathbf{z} \in S_j$.
Output: a root list on $\mathcal{L}$ in S_j that contains $\mathbf{z}$, or FAIL.

1: Run a preparation trial for bin j and bound U; **return** FAIL if it fails. Use the resource limits of Algorithm 6.
2: Draw $\mathbf{W} \sim N(\mathbf{0}, \tau_1^2\mathbf{I}/2)$ and let $\mathbf{z}_0^{(0)} \leftarrow \mathbf{W} + \mathbf{z}/2$, $\mathbf{z}_1^{(0)} \leftarrow \mathbf{W} - \mathbf{z}/2$; **return** FAIL unless both lie in H.
3: $\mathbf{u} \leftarrow \mathbf{z}_0^{(0)} \bmod \Gamma_0$ and $\mathbf{u}_0^{(0)}, \mathbf{u}_1^{(0)} \leftarrow \mathbf{u}$.
4: **for** $d = 0, \dots, D-1$ and $i = 0, \dots, 2^{d+1} - 1$ **do** ▷ fixed values and split shifts
5: Draw $\mathbf{W} \sim N(\mathbf{0}, \tau^2\mathbf{I}/2)$ and let $\mathbf{z}_{2i}^{(d+1)} \leftarrow \mathbf{W} + \mathbf{z}_i^{(d)}/2$, $\mathbf{z}_{2i+1}^{(d+1)} \leftarrow \mathbf{W} - \mathbf{z}_i^{(d)}/2$; **return** FAIL unless both lie in H.
6: $\mathbf{u}_{2i}^{(d+1)} \leftarrow \mathbf{z}_{2i}^{(d+1)} \bmod \Gamma_{d+1}$ and $\mathbf{u}_{2i+1}^{(d+1)} \leftarrow \mathbf{u}_{2i}^{(d+1)} - \mathbf{u}_i^{(d)}$.
7: $L_i^{(D)} \leftarrow \mathsf{GaussSample}(\mathbf{u}_i^{(D)})$ for $i = 0, \dots, 2^{D+1} - 1$, except that one uniformly chosen proposal slot holds $\mathbf{z}_i^{(D)}$, without a proposal or an acceptance step.
8: Merge from the leaves up as in Algorithm 6, except that each merge returns FAIL unless the pair enumerator reports the pair of fixed child entries, and accepts this pair without a random decision.
9: **return** the root list.

$A_{\mathbf{z}}$ of occurrences of $\mathbf{z}$, normalized by $\ell_d(\mathbf{z})$. Applied to the reciprocal length of the root list, it shows that the mean of a selection test is the selection factor (Lemma 4.1). Lemma 4.2 then follows from the rejection step. The proofs also use the moment bounds (Lemma 4.6), the fact that a preparation trial returns a good chain with probability $1 - o(1)$ (Lemma 4.9), and the fact that a successful test run rarely reaches either resource limit (Lemma 4.12). Section 4.5 proves them.

Test runs. For a point $\mathbf{z}$ of the target set, let $\mathsf{DGSList}^{\mathbf{z}}(d, \mathbf{c})$ denote the part of a test run below a node on $\Gamma_d + \mathbf{c}$ whose fixed entry is $\mathbf{z}$. It fixes one occurrence of $\mathbf{z}$ first and generates the remaining lists around it, and $\mathsf{DGSList}^{\mathbf{z}}(\mathsf{root}, \mathbf{0})$ is a test run $\mathsf{Run}(j, U, \mathbf{z})$ after its preparation trial. Algorithm 8 lists the test run. It differs from a run in three places.

1. At a leaf, it chooses one of the N_{prop} proposal slots uniformly and places $\mathbf{z}$ there without a proposal or an acceptance step. This is the fixed occurrence. Every other slot is as in Algorithm 4.

2. At a merge with child width σ (τ_1 at the root, σ_{d+1} otherwise), it draws $\mathbf{W} \sim N(\mathbf{0}, \sigma^2\mathbf{I}/2)$ once and sets $\mathbf{x} = \mathbf{W} + \mathbf{z}/2$, $\mathbf{y} = \mathbf{W} - \mathbf{z}/2$. It fails unless $\mathbf{x}, \mathbf{y} \in H$. It sets the split shift $\mathbf{u}$ to $\mathbf{x}$ modulo the child lattice. Then $\mathbf{x}$ lies on the left child coset and $\mathbf{y} = \mathbf{x} - \mathbf{z}$ on the right one. It runs $\mathsf{DGSList}^{\mathbf{x}}$ and $\mathsf{DGSList}^{\mathbf{y}}$ on the same cosets as Algorithm 6.

3. It fails if either child fails. A nonroot merge then draws the random spherical codes of the pair enumerator (Lemma 2.19 and Section 4.5.2). It fails unless the pair of fixed child entries is *covered*, meaning that at least one filter reports it. It accepts this pair without a random decision, and the resulting difference is the fixed occurrence in the parent list. All other pairs are treated as in Algorithms 5 and 7.

The run $\mathsf{DGSList}^{\mathbf{z}}$ follows the same pair enumeration and resource limits as Algorithm 6. The fixed slot is charged as an accepted proposal: a proposal, its rounding and shell test, and one acceptance decision. An acceptance made without a random decision is charged like any other acceptance. The Gaussian split and its shell test are charged as the split of Algorithm 6.

The *record* ω of $\mathsf{DGSList}(d, \mathbf{c})$ stores its split shifts, lists, filters, occurrence indices and work counter. It omits the scalar random numbers that the execution discards. Both $\mathsf{DGSList}$ and $\mathsf{DGSList}^{\mathbf{z}}$ are run here without the resource limits. The capped run coincides with the unlimited run when total work is at most W_0 and each output list and pair dictionary has at most M_0 entries. Whether the run stays within these limits is determined by the lists, shifts, filters and occurrence indices in the record. For a value $\mathbf{z}$, let $\mathrm{Occ}_{\mathbf{z}}(\omega)$ be the set of positions of the returned list that hold $\mathbf{z}$, so $A_{\mathbf{z}} = |\mathrm{Occ}_{\mathbf{z}}(\omega)|$. Let $\mathsf{Q}_{\mathbf{z}}$ denote the run $\mathsf{DGSList}^{\mathbf{z}}(d, \mathbf{c})$, with record $\omega^{(\mathbf{z})}$ and fixed position $i^{(\mathbf{z})}$. Failed test runs contribute zero to expectations.

Lemma 4.7 (Reweighting by occurrences)**.** *For every* $\mathbf{z}$ *in the target set of the node and every nonnegative measurable function* f *of a record and a position,*

$$\mathbb{E} \sum_{i \in \mathrm{Occ}_{\mathbf{z}}(\omega)} f(\omega, i) = \ell_d(\mathbf{z})\, \mathbb{E}_{\mathsf{Q}_{\mathbf{z}}}[\mathbf{1}_{\mathrm{success}} f(\omega^{(\mathbf{z})}, i^{(\mathbf{z})})]. \tag{4.15}$$

The resource limits are functions of the record, so f may contain the indicator that both limits are respected. At the root, the identity also holds after a preparation trial is averaged in, with preparation failure contributing zero. The last paragraph of the proof justifies both remarks.

Proof. We argue by induction over the tree.

A leaf. The N_{prop} slots are independent, and each produces $\mathbf{z}$ with probability $\ell_D(\mathbf{z})/N_{\mathrm{prop}}$ by Lemma 4.10(i). Summing over the slot that holds the occurrence, the left-hand side of (4.15) is $N_{\mathrm{prop}} \cdot (\ell_D(\mathbf{z})/N_{\mathrm{prop}})$ times the expectation of f given that a uniformly chosen slot holds $\mathbf{z}$. This is the right-hand side, since $\mathsf{DGSList}^{\mathbf{z}}$ places $\mathbf{z}$ in a uniform slot and generates the other slots as in Algorithm 4.

A merge. Let σ be the child width, Λ the child lattice and $\ell_{\mathrm{ch}} = N \det \Lambda\, \phi_\sigma \mathbf{1}_H$, and assume the identity for both children. The idea is that averaging over the split shift turns the lattice sum over child pairs into an integral over their midpoint, and the Gaussian density of the midpoint is the distribution of $\mathbf{W}$ in the test run.

Condition on the split shift $\mathbf{u}$. The record of the merge is $\omega = (\mathbf{u}, \omega_L, \omega_R, \mathcal{C}, \beta)$. Here ω_L and ω_R are the records of the left and right child runs, on $\Lambda + \mathbf{u}$ and $\Lambda + \mathbf{u} - \mathbf{c}$, $\mathcal{C}$ is the code of the pair enumerator (absent at the root), and $\beta = (\beta_{i_L, i_R})$ are the acceptance bits of the eligible occurrence pairs, drawn at their first report. Given $\mathbf{u}$, the records ω_L, ω_R and $\mathcal{C}$ are independent. Given $(\omega_L, \omega_R, \mathcal{C})$, the bits are independent, and β_{i_L, i_R} has mean α_d of the difference of the pair. Let $\mathrm{rep}(i_L, i_R)$ be the event that the pair (i_L, i_R) is reported, and let $\iota(i_L, i_R)$ be the position of its difference in the parent list when it is accepted. In the sums below, a summand is zero unless its pair is reported and accepted. Each occurrence of $\mathbf{z}$ in the parent list comes from exactly one accepted pair, so

$$\sum_{i \in \mathrm{Occ}_{\mathbf{z}}(\omega)} f(\omega, i) = \sum_{\mathbf{x} \in \Lambda + \mathbf{u}} \sum_{i_L \in \mathrm{Occ}_{\mathbf{x}}(\omega_L)} \sum_{i_R \in \mathrm{Occ}_{\mathbf{x}-\mathbf{z}}(\omega_R)} \mathbf{1}_{\mathrm{rep}(i_L, i_R)}\, \beta_{i_L, i_R}\, f\big(\omega, \iota(i_L, i_R)\big).$$

Average over β first. The bit β_{i_L, i_R} contributes the factor $\alpha_d(\mathbf{z})$, and the remaining bits are averaged with $\beta_{i_L, i_R} = 1$. Their number depends on $(\omega_L, \omega_R, \mathcal{C})$, but given these records they are finitely many independent bits, so

$$h_{\mathbf{x}}(\omega_L, i_L; \omega_R, i_R; \mathcal{C}) = \mathbf{1}_{\mathrm{rep}(i_L, i_R)}\, \mathbb{E}_\beta\big[f(\omega, \iota(i_L, i_R)) \mid \mathbf{u}, \omega_L, \omega_R, \mathcal{C},\ \beta_{i_L, i_R} = 1\big]$$

is a nonnegative measurable function of its arguments, with $h_{\mathbf{x}} = 0$ when the pair is not reported; it also depends on $\mathbf{u}$, $\mathbf{z}$ and f, which are fixed. Fix $\mathbf{x}$, ω_R and $\mathcal{C}$, and apply the identity for the left

child to

$$f_L(\omega_L, i_L) = \sum_{i_R \in \mathrm{Occ}_{\mathbf{x}-\mathbf{z}}(\omega_R)} h_{\mathbf{x}}(\omega_L, i_L; \omega_R, i_R; \mathcal{C}).$$

Then condition on the left test run's record and fixed position $(\omega_L^{(\mathbf{x})}, i_L^{(\mathbf{x})})$ and $\mathcal{C}$, and apply the identity for the right child to $f_R(\omega_R, i_R) = h_{\mathbf{x}}(\omega_L^{(\mathbf{x})}, i_L^{(\mathbf{x})}; \omega_R, i_R; \mathcal{C})$. Tonelli's theorem justifies these steps. This gives

$$\mathbb{E}\Big[\sum_{i \in \mathrm{Occ}_{\mathbf{z}}(\omega)} f \;\Big|\; \mathbf{u} \Big] = \sum_{\mathbf{x} \in \Lambda + \mathbf{u}} \alpha_d(\mathbf{z}) \ell_{\mathrm{ch}}(\mathbf{x}) \ell_{\mathrm{ch}}(\mathbf{x}-\mathbf{z})\, F(\mathbf{x}),$$

where

$$F(\mathbf{x}) = \mathbb{E}\big[\mathbf{1}_{\mathrm{succ}_L} \mathbf{1}_{\mathrm{succ}_R}\, h_{\mathbf{x}}(\omega_L^{(\mathbf{x})}, i_L^{(\mathbf{x})}; \omega_R^{(\mathbf{x}-\mathbf{z})}, i_R^{(\mathbf{x}-\mathbf{z})}; \mathcal{C})\big].$$

If $\mathbf{x} \notin H$ or $\mathbf{x}-\mathbf{z} \notin H$, both sides vanish, since the child lists lie in H; for such $\mathbf{x}$, define $F(\mathbf{x}) = 0$. In F, the child test runs $\mathsf{DGSList}^{\mathbf{x}}$ on $\Lambda + \mathbf{u}$ and $\mathsf{DGSList}^{\mathbf{x}-\mathbf{z}}$ on $\Lambda + \mathbf{u} - \mathbf{c}$ are independent of each other and of $\mathcal{C}$, with success events succ_L and succ_R. For the fixed pair, rep is the event cov that the code covers it, and f is evaluated with this pair accepted at its first report. The function F depends on $\mathbf{u}$ only through $\mathbf{u} = \mathbf{x} \bmod \Lambda$. By the Gaussian product identity (4.9) and the definition (4.12) of α_d,

$$\alpha_d(\mathbf{z}) \ell_{\mathrm{ch}}(\mathbf{x}) \ell_{\mathrm{ch}}(\mathbf{x}-\mathbf{z}) = \ell_d(\mathbf{z}) \det \Lambda\, \phi_{\sigma/\sqrt{2}}(\mathbf{x} - \mathbf{z}/2) \mathbf{1}_H(\mathbf{x}) \mathbf{1}_H(\mathbf{x}-\mathbf{z}).$$

Since $\mathbf{u}$ is uniform modulo Λ, (2.17) gives $\mathbb{E}_{\mathbf{u}} \det \Lambda \sum_{\mathbf{x} \in \Lambda + \mathbf{u}} G(\mathbf{x}) = \int G$ for every nonnegative measurable G. Hence the left-hand side of (4.15) equals

$$\begin{aligned} &\ell_d(\mathbf{z}) \int \phi_{\sigma/\sqrt{2}}(\mathbf{x} - \mathbf{z}/2) \mathbf{1}_H(\mathbf{x}) \mathbf{1}_H(\mathbf{x}-\mathbf{z}) F(\mathbf{x})\, d\mathbf{x} \\ &\qquad = \ell_d(\mathbf{z})\, \mathbb{E}_{\mathbf{W}}\big[\mathbf{1}_H(\mathbf{W} + \mathbf{z}/2) \mathbf{1}_H(\mathbf{W} - \mathbf{z}/2) F(\mathbf{W} + \mathbf{z}/2)\big], \end{aligned}$$

with $\mathbf{W} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}/2)$. The right-hand side is $\ell_d(\mathbf{z})\, \mathbb{E}_{\mathsf{Q}_{\mathbf{z}}}[\mathbf{1}_{\mathrm{success}} f]$, because $\mathsf{DGSList}^{\mathbf{z}}$ draws $\mathbf{W}$ once, fails unless both fixed children lie in H, sets $\mathbf{u}$ to $\mathbf{W} + \mathbf{z}/2$ modulo Λ, runs the two child tests, fails unless the code covers the fixed pair, and accepts it. At the root, $\mathbf{c} = \mathbf{0}$ and both children lie on $\Gamma_0 + \mathbf{u}$. The root weight replaces ℓ_d, α_{out} replaces α_d, and the full pair enumeration covers the fixed pair. The same computation applies.

Work and preparation. $\mathsf{DGSList}$ and $\mathsf{DGSList}^{\mathbf{z}}$ have the same charges for splits, proposals and acceptances, so the total work is the same function of the record for both runs. At the root, $\ell_{\mathrm{out}}(\mathbf{z})$ does not depend on the returned chain, so one can also average over a preparation trial, including failure. □

Let $\mathrm{succ}_{\mathrm{lim}}$ be the event that the run succeeds and respects both resource limits. We now prove Lemmas 4.1 and 4.2.

Proof of Lemma 4.1. The identity. Apply the reweighting identity with $f = \mathbf{1}_{\mathrm{succ}_{\mathrm{lim}}}/|L|$, defined as zero when L is empty. It gives $\Pr[\mathsf{Candidate}(j, U) = \mathbf{z}] = \ell_{\mathrm{out}}(\mathbf{z})\, \mathbb{E}_{\mathsf{Q}_{\mathbf{z}}}[\mathbf{1}_{\mathrm{succ}_{\mathrm{lim}}}/|L|]$. This expectation is $\varpi_{\mathbf{z}}$, the probability that the selection test returns 1.

The lower bound on a good chain. Fix a good returned chain, and let $\Pr_{\mathsf{Q}}$ and $\mathbb{E}_{\mathsf{Q}}$ refer to $\mathsf{Q}_{\mathbf{z}}$ on it without the resource limits. With $f \equiv 1$, the reweighting identity and (4.14) give $\Pr_{\mathsf{Q}}[\mathrm{success}] = \mathbb{E} A_{\mathbf{z}} / \ell_{\mathrm{out}}(\mathbf{z}) \geq 1 - o(1)$. With $f(\omega, i) = (|L| - 1)_+$, the left-hand side of (4.15) is

$\mathbb{E}[A_{\mathbf{z}}(|L|-1)] = \sum_{\mathbf{y}} \mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{z}=\mathbf{y}})]$, since $|L| = \sum_{\mathbf{y}} A_{\mathbf{y}}$. The bound on expected pair counts in Lemma 4.6 and (4.3) give

$$\mathbb{E}_{\mathsf{Q}}[\mathbf{1}_{\text{success}}(|L|-1)] = \frac{1}{\ell_{\text{out}}(\mathbf{z})} \sum_{\mathbf{y}} \mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{z}=\mathbf{y}})] \le (1+o(1)) \sum_{\mathbf{y}} \ell_{\text{out}}(\mathbf{y}) \le \tfrac{1}{4} + o(1).$$

On success, L contains the fixed entry, so $|L|-1$ is a nonnegative integer. By Markov's inequality, $\Pr_{\mathsf{Q}}[\text{success}, |L| \ge 2] \le \frac{1}{4} + o(1)$, and hence $\Pr_{\mathsf{Q}}[\text{success}, |L| = 1] \ge \frac{3}{4} - o(1)$. By Lemma 4.12, the resource limits remove only an $o(1)$ part of this event. On $\text{succ}_{\text{lim}} \cap \{|L| = 1\}$ the test returns 1 with probability one. Hence, on every good returned chain, the test returns 1 with probability at least $3/4 - o(1)$.

Other chains. The test prepares its own chain. By Lemma 4.9, the chain is returned and good with probability at least $(1-o(1))(1-e^{-\Omega(n/h^4)}) = 1-o(1)$. On the remaining executions, the test returns 1 with probability at least 0. Hence $\varpi_{\mathbf{z}} \ge (1-o(1))(3/4-o(1)) \ge 1/2$. The upper bound is immediate from the definition. □

Proof of Lemma 4.2. With the uniform choice of the bin, the candidate equals $\mathbf{z}$ with probability $w(\mathbf{z})\varpi_{\mathbf{z}}$ by Lemma 4.1. Averaging the probabilities $(1-\varpi_{\mathbf{z}})^i$ over i gives the finite geometric sum (4.5). Let $\delta = 2^{-m}$. By $\varpi_{\mathbf{z}} \ge 1/2$, the last factor in (4.5) lies in $[1-\delta, 1]$. Hence, conditioned on success, the output is within $\delta/(1-\delta) \le 2\delta$ of $D_{\mathcal{L},s}$ restricted to $S_0 \cup \cdots \cup S_{J-1}$. The radial tail (4.6) adds at most δ. If $U \le 2\rho_s(\mathcal{L})$, then

$$\Pr[\mathsf{Attempt} \ne \mathsf{FAIL}] \ge \frac{1-\delta}{m} \sum_{\mathbf{z}} w(\mathbf{z}) \ge \frac{(1-\delta)^2}{8Jm} \ge \frac{1}{16Jm}.$$

□

4.5 Technical lemmas

This subsection proves the technical statements used in Sections 4.3–4.4. Section 4.5.1 completes the chain and proves its periodization bound (Lemma 4.8) and the preparation guarantee (Lemma 4.9). Section 4.5.2 fixes the leaf, pair-enumeration and resource-limit rules of Algorithm 6 and proves their properties (Lemmas 4.10 and 4.11). Section 4.5.3 proves the child-shell and moment bounds and the cost bound for test runs (Lemma 4.12). We first record elementary bounds on the child-shell parameters of (4.11).

Parameters of the child shell. The relative slack ε in the shell bounds and midpoint cutoff changes the exponential estimates below only by factors $e^{O(n\varepsilon)} = 2^{o(n)}$. For the Gaussian midpoints of (4.10), the child-shell tests fail with probability $e^{-\Omega(n\varepsilon^2)} = e^{-\Omega(n/h^4)}$. Let $b_1 = R^2/\tau_1^2$. Since $\tau_1^2 \ge 3r^2/(2n)$, we have $1/2 \le b_1 \le 2/3$. Hence $\tau^2 = \frac{3}{2} b_1 \tau_1^2 \ge \frac{3}{4}\tau_1^2 \ge \frac{9}{16}\sigma_s^2$, so $\tau_1, \tau \ge 3\sigma_s/4$. If $r^2 \ge n\sigma_s^2/2$, the second term of the maximum is active, $b_1 = 2/3$, and $\tau = \tau_1$.

4.5.1 The superlattice chain

The first superlattice Γ_0. Section 4.3.1 explains why $\mathcal{L}$ itself cannot serve as Γ_0, and (4.19) below gives the bound used there. By Definition 4.5, it suffices to bound $R_{\Gamma_d}(G_{\sigma_d})$ on every lattice of the chain. A random extension of large prime index brings this second moment down to $1+o(1)$.

Let $\widetilde{U} = U/s^n$. Since $\det \mathcal{L} = 1$, Poisson summation gives

$$\rho_{1/s}(\mathcal{L}^*) = \frac{\rho_s(\mathcal{L})}{s^n} \le \widetilde{U}, \tag{4.16}$$

and $\widetilde{U} \ge \rho_{1/s}(\mathcal{L}^*) \ge 1$, so $\widetilde{U}$ bounds the dual Gaussian mass. Choose a prime first index satisfying

$$\widetilde{U}(4/3)^{n/2}q^2 \le p \le 8\widetilde{U}(4/3)^{n/2}q^2, \tag{4.17}$$

and form a uniform random index-p extension $\Gamma_0 \supset \mathcal{L}$ by (3.1). The lower bound on p makes $R_{\mathcal{L}}(G_\sigma)/p$ and $\widetilde{U}/p$ exponentially small in the proof below. The upper bound enters the root acceptance probability in Lemma 4.10(iii).

Lemma 4.8 (Periodization second moment of the first extension)**.** *There is an absolute constant $c > 0$ with the following property. Fix a bin and let Γ_0 be the random extension above. Except with probability $e^{-\Omega(n/h^4)}$ over Γ_0, where the constant in Ω does not depend on the bin, every chain $\Gamma_0 \subseteq \Gamma_1 \subseteq \cdots \subseteq \Gamma_D$ satisfies (4.13) with this c.*

Proof. A second-moment step. Let $\sigma \ge 3\sigma_s/4$ and $0 \le f \le \phi_{\sigma/\sqrt{2}}$, and let $\|f\|_1 = \int f$. By Poisson summation and Parseval's identity, $R_\Lambda(f) = \sum_{\mathbf{y}\in\Lambda^*} |\widehat{f}(\mathbf{y})|^2$, and the term $\mathbf{y} = \mathbf{0}$ equals $\|f\|_1^2$. The dual of $\mathcal{L}/p$ is $p\mathcal{L}^*$. Applying (3.2) to $|\widehat{f}|^2$ gives $\mathbb{E}[R_{\Gamma_0}(f) - \|f\|_1^2] \le R_{\mathcal{L}}(f)/p + R_{\mathcal{L}/p}(f) - \|f\|_1^2$. By (2.18), $f \le \phi_{\sigma/\sqrt{2}}$, and (2.21) for the lattice $\mathcal{L}/p$,

$$R_{\mathcal{L}/p}(f) \le \int f(\mathbf{w}) P_{\mathcal{L}/p,\phi_{\sigma/\sqrt{2}}}(\mathbf{w})\, d\mathbf{w} \le \|f\|_1 \left(1 + \rho_{1/(p\sqrt{\pi}\sigma)}(\mathcal{L}^* \setminus \{\mathbf{0}\})\right).$$

Let $t = \lfloor p\sigma/(\sqrt{2}\sigma_s)\rfloor$, so $t \ge p/4$ since $\sigma \ge 3\sigma_s/4$. As $p\sqrt{\pi}\sigma \ge ts$, Lemma 3.3 and (4.16) give

$$\rho_{1/(p\sqrt{\pi}\sigma)}(\mathcal{L}^* \setminus \{\mathbf{0}\}) \le \rho_{1/s}(t\mathcal{L}^* \setminus \{\mathbf{0}\}) \le \frac{\rho_{1/s}(\mathcal{L}^*)}{t} \le \frac{4\widetilde{U}}{p}.$$

Hence, we get

$$\mathbb{E}[R_{\Gamma_0}(f) - \|f\|_1^2] \le \frac{R_{\mathcal{L}}(f)}{p} + \|f\|_1(1 - \|f\|_1) + \frac{4\|f\|_1\widetilde{U}}{p}. \tag{4.18}$$

Application to G_σ. Let $\sigma \in \{\tau_1, \tau\}$, so $\sigma \ge 3\sigma_s/4$ by the parameter bounds at the start of Section 4.5. On the support of G_σ, $\|\mathbf{w}\|^2 \ge n\sigma^2(1/2 - 4\varepsilon)$, and comparing the two Gaussian densities gives the pointwise bound

$$G_\sigma \le c_n\, \phi_{4\sigma/(3\sqrt{2})}, \qquad c_n = \left(\frac{16}{9}e^{-7/16}\right)^{n/2} e^{7n\varepsilon/4}.$$

By monotonicity of $R_{\mathcal{L}}$ and (2.22), we have

$$R_{\mathcal{L}}(G_\sigma) \le c_n^2 R_{\mathcal{L}}(\phi_{4\sigma/(3\sqrt{2})}) = c_n^2 \rho_{3/(4\sqrt{2\pi}\sigma)}(\mathcal{L}^*) \le c_n^2\widetilde{U} = \widetilde{U}\left(\frac{16}{9}e^{-7/16}\right)^n e^{O(n/h^2)}, \tag{4.19}$$

where the second inequality uses $4\sqrt{2\pi}\sigma/3 \ge s$ and (4.16). By the lower bound in (4.17),

$$\frac{R_{\mathcal{L}}(G_\sigma)}{p} \le \left(\frac{(16/9)e^{-7/16}}{\sqrt{4/3}}\right)^n q^{-2} e^{O(n/h^2)} = e^{-\Omega(n)},$$

since $(16/9)e^{-7/16} \approx 1.148 < \sqrt{4/3} \approx 1.155$, so the ratio is about 0.994, a constant below one. Also $\widetilde{U}/p \le (3/4)^{n/2}q^{-2}$. Let $\mu_n = \|G_\sigma\|_1$. By the concentration argument in the proof of Lemma 4.4,

$\mu_n = 1 - e^{-\Omega(n/h^4)}$, independently of the width. Since $G_\sigma \le \phi_{\sigma/\sqrt{2}}$ by definition, (4.18) with $f = G_\sigma$ gives

$$\mathbb{E}[R_{\Gamma_0}(G_\sigma) - \mu_n^2] \le \frac{R_{\mathcal{L}}(G_\sigma)}{p} + \mu_n(1-\mu_n) + \frac{4\mu_n \widetilde{U}}{p} \le e^{-\Omega(n)} + \mu_n(1-\mu_n) = e^{-\Omega(n/h^4)},$$

where the two terms with denominator p are absorbed in $e^{-\Omega(n)}$. Since $R_{\Gamma_0}(G_\sigma) \ge \mu_n^2$, Markov's inequality proves the second-moment bound on Γ_0 for both widths, except with probability $e^{-\Omega(n/h^4)}$. By (2.24), $R_{\Gamma_d} \le R_{\Gamma_0}$ for every superlattice $\Gamma_d \supseteq \Gamma_0$, which proves (4.13). The extension Γ_0 does not depend on the bin, and each call uses one bin, so no union bound over the J bins is needed. □

The remaining superlattice chain. We extend Γ_0 to Γ_D so that the terminal basis satisfies

$$\max_i \|\widetilde{\mathbf{b}}_i\| \le 3\sigma_s/(4n^2). \tag{4.20}$$

Since $\tau \ge 3\sigma_s/4$, this bounds the radius of a Babai cell by $\sqrt{n}\,\tau/(2n^2)$. Lemma 4.10(i) uses this to show that every Babai cell around a point of H lies in H^+, which makes the leaf weights exact.

Let $k = \lceil n/h \rceil$, so $q = 2^k$. Apply Theorem 2.9 to a basis of Γ_0 with block size k and failure probability n^{-2}, and then halve the next k basis vectors cyclically at each level. Each extension has index q, and the parities of the coordinates for these k columns give the quotient labels for the membership tests.

A preparation trial searches for a prime p in the window (4.17) with a bounded number of samples. It fails if the window contains no integer at least two, which can happen only if $U < \rho_s(\mathcal{L})$. It chooses $\mathbf{a} \in \mathbb{F}_p^n \setminus \{\mathbf{0}\}$ uniformly by rejection from $\mathbb{F}_p^n$ with at most n draws and forms Γ_0 from it by (3.1). It builds the chain and returns it if the terminal basis satisfies (4.20). Otherwise it reports failure. Since $\det \mathcal{L} = 1$, the returned lattices have $\det \Gamma_d = (pq^d)^{-1}$. They are not rescaled.

Lemma 4.9 (Preparation)**.** *For $U \ge \rho_s(\mathcal{L})$, a preparation trial returns a chain with probability $1 - o(1)$, and for each fixed bin, conditional on a return, the chain is good with probability $1 - e^{-\Omega(n/h^4)}$. For every positive U, every execution uses $2^{O(n/h)}$ time and space. The constants and the dimension threshold are uniform in $\mathcal{L}$, s, the bin and $U \ge \rho_s(\mathcal{L})$.*

Neither the construction nor the terminal test depends on the radial bin. The proof also shows the uniformity of the constants.

Proof. The bounded prime search succeeds with probability $1 - o(1)$ by prime density [RS62], and the rejection sampling of $\mathbf{a}$ fails with probability at most p^{-n}.

The terminal basis. By Lemma 3.2 and (4.16), $\mathbb{E}\,\rho_{1/s}(\Gamma_0^* \setminus \{\mathbf{0}\}) \le 2\rho_{1/s}(\mathcal{L}^*)/p \le 2\widetilde{U}/p \le 2q^{-2}$. By Markov's inequality, except with probability $4q^{-2}$, $\rho_{1/s}(\Gamma_0^* \setminus \{\mathbf{0}\}) \le 1/2$, that is, $\eta_{1/2}(\Gamma_0) \le s$. Lemma 2.2 then gives $\lambda_n(\Gamma_0) \le \sqrt{n}\,s$. Since $n/k \le h$ and $k \le 2n/h$, Theorem 2.9 gives, except with probability n^{-2}, a basis of Γ_0 with Gram–Schmidt lengths at most $k^{n/k}\lambda_n(\Gamma_0) \le (2n/h)^h \sqrt{n}\,s$. Halving a basis vector halves its Gram–Schmidt vector and leaves the others unchanged. Since $Dk \ge h^2 n$, the cyclic schedule halves every basis vector at least h^2 times, so

$$\max_i \|\widetilde{\mathbf{b}}_i\| \le \Big(\frac{2n}{h}\Big)^h 2^{-h^2} \sqrt{n}\,s \le \Big(\frac{2}{h}\Big)^h \sqrt{n}\,s \le \frac{3\sigma_s}{4n^2}.$$

The second inequality uses $h \ge \log_2 n$, and the last holds for all sufficiently large n because $s = \sqrt{2\pi}\sigma_s$ and $(2/h)^h = n^{-\omega(1)}$. None of these steps depends on the radial bin. The first basis has polynomial bit length in the input lengths and $\log(2 + \widetilde{U})$, so the block reduction and the halvings have the claimed cost on every execution.

Good chains. Lemma 4.8 bounds the probability that the chain is not good before conditioning on a return. A trial fails with probability $o(1)$, so conditioning on a return multiplies this bound by $1 + o(1)$. □

4.5.2 Leaves, acceptance probabilities, pair enumeration and resource limits

We fix the leaf proposal count, the order of pair enumeration, and the resource limits of Algorithm 6. Lemma 4.10 proves the leaf weights and bounds the acceptance probabilities. Lemma 4.11 shows that each eligible pair receives at most one acceptance decision.

Leaves. Algorithm 4 draws a uniform point of H^+ and rounds it by Babai's nearest-plane algorithm to the prescribed coset. Let $r_H = \sqrt{n}\,R$, and enlarge H radially by r_H/n to a shell H^+. With $M_H = N \operatorname{vol}(H^+) \max_{\mathbf{x}\in H} \phi_\tau(\mathbf{x})$, choose an integer N_{prop} with $M_H \le N_{\mathrm{prop}} \le 2M_H + 2$. A rounded point $\mathbf{x} \in H$ is accepted with probability $N \operatorname{vol}(H^+)\phi_\tau(\mathbf{x})/N_{\mathrm{prop}}$. The lower bound on N_{prop} makes this probability at most one (Lemma 4.10). The upper bound limits the number of proposals.

Pair enumeration. The decoder may report one pair through several filters. A repeated acceptance decision would change the moments computed in Section 4.3.1. The enumeration below gives each occurrence pair at most one acceptance decision. Nonroot pairs are enumerated by the spherical-cap decoder of Lemma 2.19, now with squared threshold $1/4$ and target correlation $1/2$. This correlation is that of Figure 6(a), where $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{x} - \mathbf{y}$ have nearly equal norms. Fix the decoder's Cartesian traversal, including the order of its code families. Each nonzero point is normalized to its direction before the cap queries, and quotient labels are tested only afterwards. The records carry occurrence indices. A pair is *compatible* if its quotient labels agree, and *eligible* if it is compatible and its difference lies in the target set. For each report, check these two deterministic tests first. For an eligible pair, look up its ordered occurrence-index key in a dictionary. If the key is absent, insert it and make an independent acceptance decision. Thus the dictionary contains only eligible keys, including the keys of rejected eligible pairs. At the root, both children are on $\Gamma_0 + \mathbf{u}$. Label $\mathbf{x} \in X$ and $\mathbf{y} \in Y$ by $\mathbf{x} - \mathbf{u} + \mathcal{L}$ and $\mathbf{y} - \mathbf{u} + \mathcal{L}$, sort these labels, and enumerate each equal-label occurrence pair once in index order.

Resource limits. After preparation, every run uses the limits

$$W_0 = \left\lceil (3/2)^{n/2} q^6 \right\rceil \operatorname{poly}(n), \qquad M_0 = \left\lceil (4/3)^{n/2} q^6 \right\rceil, \tag{4.21}$$

where the polynomial factor is fixed in the proof of Lemma 4.12. Stop before exceeding W_0 operations or storing more than M_0 entries in any output list or pair dictionary. The failure propagates through the entire call. Charge allocations and acceptance decisions equally in ordinary and test runs. The pair decoder streams the reports with auxiliary space linear in the child list sizes plus $\operatorname{poly}(n)q^{1/3}$ by Lemma 2.19. Root enumeration sorts the two child lists and streams their compatible pairs, and the leaf routine holds one proposal at a time. The tree has fewer than $V = 2^{D+2}$ nodes. Reserve space for shifts and fixed values at every node in both kinds of run. Storing the lists and this metadata uses at most $VM_0q^{o(1)} \operatorname{poly}(n)$ space, including the dictionaries and traversal workspace. Here $V = q^{o(1)}$. The limits apply on every execution, including for invalid mass bounds U. On a good chain with $U \ge \rho_s(\mathcal{L})$, Lemma 4.12 bounds the probability of exceeding either limit in a successful test run. Preparation is accounted for separately in Lemma 4.9.

At a leaf, the lattice is Γ_D and the width is τ. Let $\alpha_{\mathrm{leaf}}(\mathbf{x}) = N \operatorname{vol}(H^+)\phi_\tau(\mathbf{x})/N_{\mathrm{prop}}$ be the leaf acceptance probability, and recall the leaf weight $\ell_D(\mathbf{x}) = N \det \Gamma_D\, \phi_\tau(\mathbf{x})\mathbf{1}_H(\mathbf{x})$.

Lemma 4.10 (Acceptance probabilities)**.** *Fix a returned chain. In part (i), run the leaf routine without the resource limits. The following hold.*

(i) At a leaf, $\alpha_{\mathrm{leaf}} \le 1$*, and each proposal slot of Algorithm 4 produces* $\mathbf{x}$ *with probability* $\ell_D(\mathbf{x})/N_{\mathrm{prop}}$*. Consequently*

$$\mathbb{E}A_{\mathbf{x}} = \ell_D(\mathbf{x}), \qquad \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{x}=\mathbf{y}})] \le \ell_D(\mathbf{x})\ell_D(\mathbf{y}), \qquad \mathbf{x},\mathbf{y} \in \Gamma_D + \mathbf{c}.$$

(ii) For every $\mathbf{z} \in H$*, the nonroot acceptance probabilities satisfy* $\alpha_d(\mathbf{z}) \le q^{-1}e^{O(n\varepsilon)} < 1/2$ *for sufficiently large* n*.*

(iii) The root acceptance probability is at most $q^{-2}e^{O(1)}$*, uniformly over the radial bins and their points. In particular it is at most* $1/2$ *for sufficiently large* n*.*

Proof. (i) Let $\Lambda = \Gamma_D$ and $\sigma = \tau$. The terminal test gives Babai cell radius at most $\sqrt{n}\,\sigma/(2n^2) \le r_H/n$, because $r_H^2 = (2/3)n\sigma^2$. Each nearest-plane cell is a translate of the box with side vectors $\widetilde{\mathbf{b}}_i$ and volume $\det\Lambda$. The cell of every supported point $\mathbf{x}$ lies in H^+, giving proposal probability $\det\Lambda/\operatorname{vol}(H^+)$. The lower bound on N_{prop} makes $\alpha_{\mathrm{leaf}}(\mathbf{x})$ at most one. Multiplying by $\alpha_{\mathrm{leaf}}(\mathbf{x})$ gives the stated slot probability. To count occurrences we sum over all slots. To count ordered pairs of distinct occurrences we sum over two distinct slots, whose outcomes are independent. The radial enlargement changes the outer ball volume by only $e^{O(1)}$. On H, the maximal density has squared radius $(1-\varepsilon)(2/3)n\sigma^2$. The ball-volume estimate therefore gives

$$\begin{aligned}\frac{\log_2 N_{\mathrm{prop}}}{n} &= \frac{1}{2} - \frac{1}{6\ln 2} + \frac{\ln(2/3)+1/3}{2\ln 2} + O(h^{-2}) + \frac{2\log_2 q + O(\log n)}{n} \\ &= \tfrac{1}{2}\log_2(4/3) + O(h^{-2}) + \frac{2\log_2 q + O(\log n)}{n}.\end{aligned}$$

(ii) At the merge producing a list on Γ_0, the parent width is τ_1 and the child width is τ, where $\tau^2 = 3b_1\tau_1^2/2$. By (4.12) and $\det\Gamma_0 = q\det\Gamma_1$, $\alpha_0(\mathbf{z}) = (q/N)\,\phi_{\tau_1}(\mathbf{z})/\phi_{\sqrt{2}\tau}(\mathbf{z})$. For $\mathbf{z} \in H$, $\|\mathbf{z}\|^2 = nb_1\tau_1^2(1+O(\varepsilon))$, so the ratio of densities is $(3b_1)^{n/2}e^{n(1/6-b_1/2)+O(n\varepsilon)}$. With $N \ge q^2 2^{n/2}e^{-n/6}$ this gives

$$\ln\alpha_0(\mathbf{z}) \le -\ln q + n\left(\tfrac{1}{2}\ln(3b_1/2) - \tfrac{1}{2}b_1 + \tfrac{1}{3}\right) + O(n\varepsilon).$$

The expression in parentheses is at most $b_1/4 - 1/6 \le 0$, by $\ln t \le t-1$ and $b_1 \le 2/3$. At every later merge, both widths are τ. The same substitution gives $\ln\alpha_d(\mathbf{z}) \le -\ln q + O(n\varepsilon)$. Since $n\varepsilon = o(\ln q)$, both bounds are below $1/2$.

(iii) Insert the exact target weight (4.3) into (4.12) at child width τ_1. We bound α_{out} on S_j by replacing $\|\mathbf{z}\|^2$ with the bin midpoint r_j^2. Let $\xi = r_j^2/(n\sigma_s^2)$ and $v = \tau_1^2/\sigma_s^2 = \max\{3/4, 3\xi/2\}$. A point in S_j satisfies $|\|\mathbf{z}\|^2/(n\sigma_s^2) - \xi| \le \pi/n$. Since $|1/(4v) - 1/2| \le 1/2$, replacing its norm by the midpoint changes $\ln\alpha_{\mathrm{out}}(\mathbf{z})$ by at most $\pi/2$. Substituting $N^2 \ge 2^n e^{-n/3}q^4$ and the upper bound on p gives

$$\ln\alpha_{\mathrm{out}}(\mathbf{z}) \le -2\ln q + n\left(\tfrac{1}{2}\ln(2v/3) + \tfrac{1}{3} + \frac{\xi}{4v} - \frac{\xi}{2}\right) + \frac{\pi}{2} + \ln 8.$$

The expression in parentheses is nonpositive. For $\xi \le 1/2$ it equals $1/3 - \frac{1}{2}\ln 2 - \xi/6 < 0$. For $\xi \ge 1/2$ it equals $\frac{1}{2}(\ln\xi + 1 - \xi) \le 0$. Hence $\alpha_{\mathrm{out}}(\mathbf{z}) \le 8q^{-2}e^{\pi/2}$, uniformly in the bin, including $\mathbf{z} = \mathbf{0}$ in S_0. □

The ball-volume estimate in the proof of (i) also gives $N_{\mathrm{prop}} = (4/3)^{n/2}2^{O(n/h^2)}q^2\,\mathrm{poly}(n)$. Section 4.5.3 uses this rate.

For the nonroot decoder, let $\iota_{\mathcal{C}}(\mathbf{x})$ be the number of filters containing $\mathbf{x}$, and $m_{\mathcal{C}}(\mathbf{x},\mathbf{y})$ the number of filters containing both $\mathbf{x}$ and $\mathbf{y}$.

Lemma 4.11 (Pair enumeration)**.** *On a returned chain, the pair enumerator of Section 4.5.2 gives each compatible occurrence pair at most one acceptance decision. Without the resource limits, every compatible root pair is enumerated, and any eligible nonroot pair is omitted with probability at most 2^{-n}. For every nonzero $\mathbf{x}$, with expectation over the code, and for independent uniform directions $\mathbf{X},\mathbf{Y}$ in $\mathbb{R}^n$, with expectation over the code and $\mathbf{X},\mathbf{Y}$,*

$$\begin{aligned}\mathbb{E}\iota_{\mathcal{C}}(\mathbf{x}) &\le (9/8)^{n/2}q^{1/2}\,\mathrm{poly}(n),\\ \mathbb{E}m_{\mathcal{C}}(\mathbf{X},\mathbf{Y}) &\le (27/32)^{n/2}q^{1/2}\,\mathrm{poly}(n), \qquad m_{\mathcal{C}}(\mathbf{x},\mathbf{y}) \le \min\{\iota_{\mathcal{C}}(\mathbf{x}),\iota_{\mathcal{C}}(\mathbf{y})\}.\end{aligned} \tag{4.22}$$

Proof. For a nonroot eligible pair $\mathbf{z} = \mathbf{x} - \mathbf{y}$, all three vectors lie in H, so their correlation is $1/2 + O(\varepsilon)$. Since $\varepsilon = h^{-2}$, this lies in $[1/2 - 1/\lceil \ln n\rceil, 2/3]$ for sufficiently large n. Lemma 2.19 therefore gives coverage failure at most 2^{-n}.

The dictionary rule of Section 4.5.2 inserts each eligible ordered occurrence-index pair before the acceptance decision and skips all later reports of that key. Hence every eligible occurrence pair receives at most one acceptance decision, including when vector values repeat. At the root, the sorted equal-label enumeration covers every compatible occurrence pair once.

By Lemma 2.19, the code has at most $(3/2)^{n/2}q^{1/2}$ filters. Each filter is a cap $\{\mathbf{w} : \langle \mathbf{w}/\|\mathbf{w}\|, \mathbf{c}\rangle \ge 1/2\}$ around a center $\mathbf{c}$ on the unit sphere of the padded space $\mathbb{R}^{n+O(\log n)}$, and each center is marginally uniform. For $\mathbf{x} \in \mathbb{R}^n$, only the projection of $\mathbf{c}$ onto the first n coordinates enters $\langle \mathbf{x}, \mathbf{c}\rangle$, and this projection has norm at most one. Hence a fixed nonzero $\mathbf{x}$ lies in a given filter with probability at most $(3/4)^{n/2}\,\mathrm{poly}(n)$, the cap probability for squared threshold $1/4$. Summing over the filters gives the first bound in (4.22). For the second, fix a center. The events that $\mathbf{X}$ and $\mathbf{Y}$ lie in its filter are independent, and each has probability at most $(3/4)^{n/2}\,\mathrm{poly}(n)$. Summing the product over the filters gives the second bound. The third bound holds because a filter containing both points contains each of them. □

Filter membership is a cap condition on the normalized point. Hence $\iota_{\mathcal{C}}(\mathbf{x})$ and $m_{\mathcal{C}}(\mathbf{x},\mathbf{y})$ depend on nonzero arguments only through $\mathbf{x}/\|\mathbf{x}\|$ and $\mathbf{y}/\|\mathbf{y}\|$, and $\mathbb{E}\iota_{\mathcal{C}}(\mathbf{x})$ is the same for every nonzero $\mathbf{x}$. The enumeration uses the operations of (2.37), up to a polynomial in record lengths for dictionary lookups.

4.5.3 Proofs of the child-shell, moment and cost bounds

The child-shell and moment bounds.

Proof of Lemma 4.4. Let $\mathbf{x} - \mathbf{z}/2$ have density $\phi_{\sigma/\sqrt{2}}$. By (4.10), both children have normalized squared norm $\|\cdot\|^2/(n\sigma^2)$ with mean $1/2 + \|\mathbf{z}\|^2/(4n\sigma^2)$. The normalized center R^2/σ^2 of H is b_1 at the root and $2/3$ later, with half-width that center times ε. At the root the mean differs from b_1 by at most $\pi/(3n)$, since $|\|\mathbf{z}\|^2 - r^2| \le s^2/2$ and $\tau_1^2 \ge \frac{3}{4}\sigma_s^2$. At a nonroot merge it differs from $2/3$ by at most $\varepsilon/6$. Hence the $\varepsilon/4$-band about the mean lies in H. The concentration argument of Lemma 2.16, applied to both children and to the midpoint, gives the first bound.

For the second bound, let $b = R^2/\sigma^2 \le 2/3$. On the support of $g_{\mathbf{z}}$, $\|\mathbf{x} - \mathbf{z}/2\|^2 \ge n\sigma^2(1/2 - 4\varepsilon)$ and $\|\mathbf{x}\|^2 \le nb\sigma^2(1+\varepsilon)$, so

$$\frac{g_{\mathbf{z}}(\mathbf{x})^2}{\phi_\sigma(\mathbf{x})} \le 2^{n/2}(\pi\sigma^2)^{-n/2}e^{-n+nb/2+O(n\varepsilon)}.$$

The support lies in a ball of radius $\sigma\sqrt{n(1/2+4\varepsilon)}$ about $\mathbf{z}/2$, of volume at most $(\pi e\sigma^2)^{n/2}e^{O(n\varepsilon)}$. Hence

$$\frac{1}{N}\int \frac{g_{\mathbf{z}}(\mathbf{x})^2}{\phi_\sigma(\mathbf{x})}\,d\mathbf{x} \le \frac{2^{n/2}e^{(b-1)n/2+O(n\varepsilon)}}{N} \le q^{-2}e^{O(n\varepsilon)},$$

where the last step uses $b \le 2/3$ and $N \ge q^2 2^{n/2}e^{-n/6}$. Finally, $\|\mathbf{x}-\mathbf{z}/2\|^2 = (\|\mathbf{x}\|^2+\|\mathbf{x}-\mathbf{z}\|^2)/2 - \|\mathbf{z}\|^2/4$ bounds the normalized deviation of the midpoint from $1/2$ by $5\varepsilon/4$ at a nonroot merge and by $b_1\varepsilon+\pi/(3n)$ at the root. Both are within the cutoff 4ε of G_σ for sufficiently large n. □

Proof of Lemma 4.6. One merge. Let σ be the child width, Λ the child lattice, $\ell_{\mathrm{ch}} = N\det\Lambda\,\phi_\sigma \mathbf{1}_H$ and $g_{\mathbf{z}} = g_{\sigma,\mathbf{z}}$. Suppose the two child lists have first moments between $e^{-\theta_i}\ell_{\mathrm{ch}}$ and ℓ_{ch} and expected distinct-pair counts at most $e^{2\theta_i}\ell_{\mathrm{ch}}(\mathbf{x})\ell_{\mathrm{ch}}(\mathbf{y})$, for $i=1,2$. We show that the parent has the same bounds with $\theta_1+\theta_2+e^{-\Omega(n/h^4)}$ in place of θ_i, the distinct-pair bound under (4.13). Conditional on the split shift $\mathbf{u}$, the two child lists are independent. An occurrence pair $(\mathbf{x},\mathbf{x}-\mathbf{z})$ is accepted with probability $\alpha_d(\mathbf{z})$ ($\alpha_{\mathrm{out}}(\mathbf{z})$ at the root) when it is eligible and covered. By the Gaussian product identity (4.9) and the definition (4.12) of α_d,

$$\mathbb{E}[A_{\mathbf{z}} \mid \mathbf{u}] \le \alpha_d(\mathbf{z}) \sum_{\mathbf{x}\in\Lambda+\mathbf{u}} \ell_{\mathrm{ch}}(\mathbf{x})\ell_{\mathrm{ch}}(\mathbf{x}-\mathbf{z}) = \alpha_d(\mathbf{z})N^2(\det\Lambda)^2\phi_{\sqrt{2}\sigma}(\mathbf{z}) \sum_{\mathbf{x}\in\Lambda+\mathbf{u}} g_{\mathbf{z}}(\mathbf{x}) = \ell_d(\mathbf{z})P_{\Lambda,g_{\mathbf{z}}}(\mathbf{u}),$$

where the midpoint cutoff in $g_{\mathbf{z}}$ holds by the last step of the proof of Lemma 4.4. The same expression with the factor $e^{-\theta_1-\theta_2}(1-2^{-n})$, for the child lower bounds and the coverage of Lemma 4.11, is a lower bound. Averaging over the uniform shift $\mathbf{u}$ replaces $P_{\Lambda,g_{\mathbf{z}}}(\mathbf{u})$ by $\int g_{\mathbf{z}} \in [1-e^{-\Omega(n/h^4)},1]$. This gives the upper bound $\mathbb{E}A_{\mathbf{z}} \le \ell_d(\mathbf{z})$ on every returned chain, and a lower bound with error $\theta_1+\theta_2+e^{-\Omega(n/h^4)}$. The whole-tree argument below accumulates these errors.

For distinct output occurrences, distinguish disjoint child pairs, a shared left occurrence, and a shared right occurrence. After division by the two parent weights, the disjoint contribution is at most

$$e^{2\theta_1+2\theta_2}\mathbb{E}_{\mathbf{u}}[P_{\Lambda,g_{\mathbf{z}}}(\mathbf{u})P_{\Lambda,g_{\mathbf{z}'}}(\mathbf{u})] \le e^{2\theta_1+2\theta_2}\sqrt{R_\Lambda(g_{\mathbf{z}})R_\Lambda(g_{\mathbf{z}'})} \le e^{2\theta_1+2\theta_2}(1+e^{-\Omega(n/h^4)}).$$

For a shared left occurrence, the contribution from the left child is its expected entry count, and the contribution from the right child is its expected count of distinct pairs. The occurrence $\mathbf{x}$ then forms the pairs $(\mathbf{x},\mathbf{x}-\mathbf{z})$ and $(\mathbf{x},\mathbf{x}-\mathbf{z}')$. Applying the cancellation of (4.9) and (4.12) to each pair, and dividing by the single factor $\ell_{\mathrm{ch}}(\mathbf{x})$ that the two pairs share, gives for $\mathbf{x}\in H$

$$\alpha_d(\mathbf{z})\alpha_d(\mathbf{z}')\,\ell_{\mathrm{ch}}(\mathbf{x})\ell_{\mathrm{ch}}(\mathbf{x}-\mathbf{z})\ell_{\mathrm{ch}}(\mathbf{x}-\mathbf{z}') \le \ell_d(\mathbf{z})\ell_d(\mathbf{z}')\det\Lambda\,\frac{g_{\mathbf{z}}(\mathbf{x})g_{\mathbf{z}'}(\mathbf{x})}{N\phi_\sigma(\mathbf{x})}.$$

Summing over $\mathbf{x}\in\Lambda+\mathbf{u}$ and averaging over $\mathbf{u}$ with (2.17) replaces $\det\Lambda\sum_{\mathbf{x}}$ by $\int d\mathbf{x}$. After division by the two parent weights, Cauchy–Schwarz and the second bound of Lemma 4.4 leave at most

$$\frac{e^{2\theta_2}}{N}\int \frac{g_{\mathbf{z}}(\mathbf{x})g_{\mathbf{z}'}(\mathbf{x})}{\phi_\sigma(\mathbf{x})}\,d\mathbf{x} \le \frac{e^{2\theta_2}}{N}\left(\int\frac{g_{\mathbf{z}}^2}{\phi_\sigma}\int\frac{g_{\mathbf{z}'}^2}{\phi_\sigma}\right)^{1/2} \le e^{2\theta_2}q^{-2}e^{O(n\varepsilon)}.$$

The shared-right term has the same bound with the child roles reversed. Since $n\varepsilon = o(\log q)$, their sum is absorbed in $e^{-\Omega(n/h^4)}$. Each pair receives only one independent acceptance decision, so there is no fourth case. The same calculation applies to the root weight, whose only use is the same cancellation.

The whole tree. Apply the one-merge bounds from the leaves upward. On good chains, their second-moment hypothesis holds by (4.13). Leaves have zero error by Lemma 4.10(i). At a merge, the first-moment error is at most $\theta_1 + \theta_2 + \theta_{\text{local}}$ with $\theta_{\text{local}} = e^{-\Omega(n/h^4)} + O(2^{-n})$, on every returned chain, and on a good chain the exponent in the bound on expected distinct-pair counts accumulates in the same way. There are fewer than $V = 2^{D+2}$ calls and $\log V = O(h^3) = o(n/h^4)$, so the accumulated error is at most $V\theta_{\text{local}} = e^{-\Omega(n/h^4)}$. This proves both claims. The first-moment bounds use no second-moment hypothesis on the chain and hold on every returned chain. Both bounds are uniform in the coset shift: each merge step averages only over the fresh split shift, and the periodization second moments and the midpoint mass $\int g_{\mathbf{z}}$ it uses are translation invariant. □

Cost of test runs. Lemma 4.1 requires that the resource limits stop a test run only with probability $o(1)$, uniformly in $\mathbf{z}$. Lemma 4.12 proves this under the distribution of $\mathsf{DGSList}^{\mathbf{z}}$ conditioned on success. Let $V = 2^{D+2}$, an upper bound on the number of recursive calls.

Lemma 4.12 (Test runs rarely reach the resource limits)**.** *Assume $U \geq \rho_s(\mathcal{L})$. Fix a good returned chain and a root value $\mathbf{z} \in S_j$. Conditional on success, $\mathsf{DGSList}^{\mathbf{z}}(\mathsf{root}, \mathbf{0})$ run without the resource limits exceeds either limit in (4.21) with probability $o(1)$, uniformly in $\mathbf{z}$ and the bin.*

Proof. Let $\Pr_{\mathbf{z}}$ and $\mathbb{E}_{\mathbf{z}}$ denote probability and expectation for $\mathsf{DGSList}^{\mathbf{z}}(\mathsf{root}, \mathbf{0})$ run without the resource limits and conditioned on success. The recursion tree is complete and binary, with leaves at level D. Its node set $\mathcal{N}$ satisfies $|\mathcal{N}| < V$, and every node is executed under $\Pr_{\mathbf{z}}$. For $a \in \mathcal{N}$, let C_a be the work charged at a itself: the split, shell tests, pair enumeration, dictionary operations and acceptance decisions at a merge, or the proposals at a leaf. For a merge a at level d, write $\mathbf{c}_a$ for its coset shift ($\mathbf{0}$ at the root) and $\mathbf{u}_a$ for its split shift. Let $\mathbf{z}_a$ be the fixed value at a, and let $\mathbf{x}_a, \mathbf{y}_a$ be the two fixed child values. Let $E_a = (\mathbf{c}_a, \mathbf{u}_a)$.

For a nonroot merge a at level d, let Haar_a be the product Haar measure on $\mathbb{R}^n/\Gamma_d \times \mathbb{R}^n/\Gamma_{d+1}$. At the root, let Haar_a be the distribution with $\mathbf{c}_a = \mathbf{0}$ and $\mathbf{u}_a$ uniform modulo Γ_0. Define[3]

$$\begin{aligned}
\delta &= \max_{a \text{ a merge}} d_{\text{TV}}(\text{Dist}_{\Pr_{\mathbf{z}}}(E_a), \text{Haar}_a)\,, \\
\overline{T}_{\text{merge}} &= \max_{a \text{ a merge}} \mathbb{E}_{e \sim \text{Haar}_a}[\mathbb{E}_{\mathbf{z}}[C_a \mid E_a = e]]\,, \\
\overline{T}_{\text{leaf}} &= \max_{a \text{ a leaf}} \mathbb{E}_{\mathbf{z}} C_a.
\end{aligned}$$

Here $\text{Dist}_{\Pr_{\mathbf{z}}}(E_a)$ denotes the distribution of E_a in the successful test run. Thus δ is the largest total variation distance between a merge's shift distribution and Haar measure. The quantities $\overline{T}_{\text{merge}}$ and $\overline{T}_{\text{leaf}}$ are the largest expected merge and leaf costs, respectively, with the merge shifts averaged under Haar measure. By Lemma 2.14 and Markov's inequality, every merge a satisfies

$$\begin{aligned}
\Pr_{\mathbf{z}}[C_a > W_0/V] &\leq \mathbb{E}_{e \sim \text{Haar}_a}\Big[\Pr_{\mathbf{z}}[C_a > W_0/V \mid E_a = e]\Big] + \delta \\
&\leq \frac{V\overline{T}_{\text{merge}}}{W_0} + \delta.
\end{aligned}$$

Total variation is applied here only to the conditional probability, which lies in $[0, 1]$. It cannot transfer a bound on the expected work, which is unbounded. For a leaf, Markov's inequality gives $\Pr_{\mathbf{z}}[C_a > W_0/V] \leq V\overline{T}_{\text{leaf}}/W_0$. A union bound therefore gives

$$\Pr_{\mathbf{z}}\Big[\sum_{a \in \mathcal{N}} C_a > W_0\Big] \leq V\left(\frac{V \max\{\overline{T}_{\text{merge}}, \overline{T}_{\text{leaf}}\}}{W_0} + \delta\right).$$

[3]Where the density of the shifts in the test run vanishes, set the conditional costs and probabilities of exceeding a limit to zero.

The same transfer of a conditional probability applies to the event that a node exceeds its storage cap. Step 1 bounds δ, Step 2 bounds the work, and Step 3 handles the storage cap.

To describe the conditional distribution within a subtree, let $\mathcal{A}_a$ consist of the fixed values, split shifts, Gaussian splits and coverage events of the ancestors of a, excluding their lists.

We use the following two facts about successful test runs.

Fact A (subtree independence).

Under $\Pr_{\mathbf{z}}$, conditional on $\mathcal{A}_a$ and $(\mathbf{u}_a, \mathbf{x}_a, \mathbf{y}_a)$, the child lists X_a, Y_a are independent.

Proof of Fact A. The data $\mathcal{A}_a$ exclude the lists of the ancestors, so the success conditions of the ancestors depend on the subtree of a only through its success. Given $\mathcal{A}_a$, the conditions for success outside the subtree of a do not depend on the random choices within it. Once $(\mathbf{u}_a, \mathbf{x}_a, \mathbf{y}_a)$ is fixed, the remaining success conditions within the subtree are that both children succeed and, at a nonroot merge, that the code covers the fixed pair. These events depend on independent random choices, so conditioning on them preserves independence. The child lists therefore have the distributions returned by the unlimited runs $\mathsf{DGSList}^{\mathbf{x}_a}$ and $\mathsf{DGSList}^{\mathbf{y}_a}$ on the child cosets, each conditioned on success. At a nonroot merge, the code is likewise independent of these lists and is conditioned on covering the fixed pair.

The same argument shows that, given $\mathcal{A}_a$, the distribution of $\mathbf{u}_a$ is that of the split shift in the unlimited run $\mathsf{DGSList}^{\mathbf{z}_a}$ on its fixed coset, conditioned on success.

Fact B (conditional reweighting).

$$\text{For every nonnegative function } F \text{ of the record,} \quad \mathbb{E}_{\mathsf{Q}_{\mathbf{x}}}[F \mid \text{success}] = \frac{\mathbb{E}[A_{\mathbf{x}} F]}{\mathbb{E} A_{\mathbf{x}}}. \tag{4.23}$$

Proof of Fact B. Here $\mathsf{Q}_{\mathbf{x}}$ denotes an unlimited test run of $\mathsf{DGSList}^{\mathbf{x}}$, and the expectations on the right are for the corresponding run of $\mathsf{DGSList}$ without the resource limits. To prove the identity, let ℓ be its target weight. Equation (4.15) with $f \equiv 1$ and (4.14) give $\Pr_{\mathsf{Q}_{\mathbf{x}}}[\text{success}] = \mathbb{E} A_{\mathbf{x}}/\ell(\mathbf{x}) > 0$; dividing (4.15) by this probability gives (4.23).

Step 1: bounding δ using periodization. Fix a merge with child lattice Λ and child width σ: at the root these are Γ_0 and τ_1, and at level d they are Γ_{d+1} and σ_{d+1}. For a parent value $\mathbf{v}$, let $A_{\mathbf{v}}$ count its occurrences in the parent list. Applying (4.23) to indicators of events of the split shift identifies its distribution in a successful test run as $d\mu_{\mathbf{v}}(\mathbf{u}) \propto \mathbb{E}[A_{\mathbf{v}} \mid \mathbf{u}]\, d\mathbf{u}$, where $d\mathbf{u}$ is Haar measure on $\mathbb{R}^n/\Lambda$. By Fact A, $\mu_{\mathbf{v}}$ is also the conditional distribution of $\mathbf{u}_a$ under $\Pr_{\mathbf{z}}$ given $\mathcal{A}_a$ with $\mathbf{z}_a = \mathbf{v}$.

We show that $\mu_{\mathbf{v}}$ is within $\theta_n = e^{-\Omega(n/h^4)}$ of Haar measure, uniformly in $\mathbf{v}$ and the coset shift. Its density is proportional to a sum over $\mathbf{x} \in \Lambda + \mathbf{u}$, in which each term $g_{\mathbf{v}}(\mathbf{x})$ is multiplied by the ratios of the expected multiplicities in the two child lists to their target weights at $\mathbf{x}$ and $\mathbf{x} - \mathbf{v}$ and by the coverage probability of the pair (equal to one at the root). Lemmas 4.6 and 2.19 place these factors in $[1 - e^{-\Omega(n/h^4)}, 1]$ pointwise, so (2.30) reduces the comparison to the normalized periodization of $g_{\mathbf{v}}$, with error at most $e^{-\Omega(n/h^4)}$. Lemma 4.4 gives $\int g_{\mathbf{v}} = 1 - e^{-\Omega(n/h^4)}$, while $R_\Lambda(g_{\mathbf{v}}) \le 1 + e^{-\Omega(n/h^4)}$ follows from $g_{\mathbf{v}} \le G_\sigma(\cdot - \mathbf{v}/2)$, monotonicity and translation invariance of R_Λ, and the good-chain condition (4.13). Equation (2.26) applied to the normalized periodization now gives the claimed distance bound.

At a nonroot merge a with parent b, the coset shift $\mathbf{c}_a$ is $\mathbf{u}_b$ or $\mathbf{u}_b - \mathbf{c}_b$, so its distribution under $\Pr_{\mathbf{z}}$ is within θ_n of Haar measure on $\mathbb{R}^n/\Gamma_d$. The triangle inequality, applied first to $\mathbf{c}_a$ and then to $\mathbf{u}_a$ conditioned on $\mathcal{A}_a$, therefore gives $\delta \le 2\theta_n$.

Step 2: bounding $\overline{T}_{\text{merge}}$ and $\overline{T}_{\text{leaf}}$ using moment bounds. At each merge, we bound the conditional expected work by a function of its shifts and then bound its Haar average.

Let $\mathcal{P} = \mathrm{poly}(n)$ bound the $\mathrm{poly}(n)$ factor of (2.37) times the cost of arithmetic on records of polynomial bit length. A list at level d has at most $N_{\mathrm{prop}}^{2^{D-d}}$ entries, so occurrence indices have bit length at most $2^{D+1}\log_2(2+N_{\mathrm{prop}}) = q^{o(1)}$. Thus dictionary operations cost $q^{o(1)}\mathcal{P}$ per entry, and sorting has the same per-entry cost up to a logarithmic factor in the list length.

Nonroot merges. Fix a merge a at level d, and let $\Lambda = \Gamma_{d+1}$, $\sigma = \sigma_{d+1}$ and $\ell_{\mathrm{ch}} = N \det\Lambda\, \phi_\sigma \mathbf{1}_H$. With expectations over the code, let $\bar\iota = \mathbb{E}\iota_{\mathcal{C}}(\mathbf{x})$, which does not depend on $\mathbf{x} \in H$ by Lemma 4.11, and define

$$\bar\ell(\mathbf{u}) = \sum_{\mathbf{y}\in\Lambda+\mathbf{u}} \ell_{\mathrm{ch}}(\mathbf{y}), \qquad \Pi(\mathbf{c},\mathbf{u}) = \sum_{\mathbf{x}'\in\Lambda+\mathbf{u}} \sum_{\mathbf{y}'\in\Lambda+\mathbf{u}-\mathbf{c}} \ell_{\mathrm{ch}}(\mathbf{x}')\ell_{\mathrm{ch}}(\mathbf{y'})\, \mathbb{E}m_{\mathcal{C}}(\mathbf{x}',\mathbf{y}').$$

Condition on $\mathcal{A}_a$ and $(\mathbf{u}_a, \mathbf{x}_a, \mathbf{y}_a)$, and denote the resulting shifts and fixed values by $(\mathbf{c},\mathbf{u},\mathbf{x},\mathbf{y})$. Let $n_X(\mathbf{x}')$ be the number of entries of X_a equal to $\mathbf{x}'$ other than the fixed one. Define n_Y likewise. To bound this count using Fact B, take $F = (A_{\mathbf{x}'} - \mathbf{1}_{\mathbf{x}'=\mathbf{x}})_+$. It equals $n_X(\mathbf{x}')$ on every successful test run and satisfies $A_{\mathbf{x}}F = A_{\mathbf{x}}(A_{\mathbf{x}'} - \mathbf{1}_{\mathbf{x}'=\mathbf{x}})$. Facts A and B, together with the bounds on expected counts of entries and pairs in Lemma 4.6, give

$$\mathbb{E}[n_X(\mathbf{x}') \mid \cdot] = \frac{\mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{x}'} - \mathbf{1}_{\mathbf{x}'=\mathbf{x}})]}{\mathbb{E}A_{\mathbf{x}}} \le (1+o(1))\ell_{\mathrm{ch}}(\mathbf{x}'). \tag{4.24}$$

Here the expectations in the ratio are for the corresponding run of DGSList on the fixed child coset. The bound is uniform in the coset shift, and summing over $\mathbf{x}'$ gives

$$\mathbb{E}[|X_a| \mid \cdot] \le 1 + (1+o(1))\bar\ell(\mathbf{u}),$$

with the same bounds for Y_a using $\bar\ell(\mathbf{u}-\mathbf{c})$. These bounds also hold for the child lists at the root.

Under this conditioning, the code is independent of the lists and is conditioned on covering the fixed pair. Since $\iota_{\mathcal{C}}, m_{\mathcal{C}} \ge 0$, the coverage condition increases their means by at most a factor $(1-2^{-n})^{-1}$ (Lemma 4.11). The point–filter incidences $\sum_{\mathbf{x}'\in X_a}\iota_{\mathcal{C}}(\mathbf{x}') + \sum_{\mathbf{y}'\in Y_a}\iota_{\mathcal{C}}(\mathbf{y}')$ therefore have conditional mean at most $(1-2^{-n})^{-1}\bar\iota\,\mathbb{E}[|X_a|+|Y_a| \mid \cdot]$.

For the number of shared-filter pair visits $\sum_{(\mathbf{x}',\mathbf{y}')\in X_a\times Y_a} m_{\mathcal{C}}(\mathbf{x}',\mathbf{y}')$, including repetitions, we distinguish pairs according to whether a fixed entry is involved. Pairs of two other entries have conditional expected contribution at most $(1-2^{-n})^{-1}\sum_{\mathbf{x}',\mathbf{y}'}\mathbb{E}[n_X(\mathbf{x}') \mid \cdot]\mathbb{E}[n_Y(\mathbf{y}') \mid \cdot]\mathbb{E}m_{\mathcal{C}}(\mathbf{x}',\mathbf{y}') \le (1+o(1))^2(1-2^{-n})^{-1}\Pi(\mathbf{c},\mathbf{u})$. Since $m_{\mathcal{C}}(\mathbf{x}',\mathbf{y}') \le \min\{\iota_{\mathcal{C}}(\mathbf{x}'),\iota_{\mathcal{C}}(\mathbf{y}')\}$, the pairs containing a fixed entry contribute at most $(1-2^{-n})^{-1}\bar\iota\,\mathbb{E}[|X_a|+|Y_a|-1 \mid \cdot]$ to this conditional mean. The number of acceptance decisions is at most the number of pair visits plus one, and (2.37) bounds the traversal overhead by $q^{1/3}$.

For sufficiently large n, the table below collects these conditional mean bounds and the cost factor for traversal and dictionary operations.

Contribution at node a	Upper bound
list entries $\lvert X_a\rvert + \lvert Y_a\rvert$	$2 + (1+o(1))\big(\bar\ell(\mathbf{u}) + \bar\ell(\mathbf{u}-\mathbf{c})\big)$
point–filter incidences	$\frac{4}{3}\bar\iota\,\mathbb{E}[\lvert X_a\rvert + \lvert Y_a\rvert \mid \cdot]$
pair visits, two other entries	$2\Pi(\mathbf{c},\mathbf{u})$
pair visits, a fixed entry	$\frac{4}{3}\bar\iota\,\mathbb{E}[\lvert X_a\rvert + \lvert Y_a\rvert - 1 \mid \cdot]$
acceptance decisions	pair visits $+1$
traversal and dictionary	factor $q^{1/3+o(1)}\mathcal{P}$

To bound the sum of these contributions, define

$$\mathrm{T}_{\mathrm{node}}(\mathbf{c},\mathbf{u}) = C_0 q^{1/3+o(1)}\mathcal{P}\big[1 + \bar\iota + (1+\bar\iota)\big(\bar\ell(\mathbf{u}) + \bar\ell(\mathbf{u}-\mathbf{c})\big) + \Pi(\mathbf{c},\mathbf{u})\big],$$

where C_0 is a sufficiently large absolute constant. Since $\bar{\iota}$ and the bound in (4.24) do not depend on the fixed values, the resulting bound depends only on the shifts. The table gives the first inequality below uniformly in the conditioning variables. We establish the second by averaging over the shifts.

$$\mathbb{E}_{\mathbf{z}}[C_a \mid \mathcal{A}_a, \mathbf{u}_a, \mathbf{x}_a, \mathbf{y}_a] \le \mathrm{T}_{\mathrm{node}}(E_a), \qquad \mathbb{E}_{\mathrm{Haar}_a}\mathrm{T}_{\mathrm{node}} \le (3/2)^{n/2} q^5 \mathcal{P}. \tag{4.25}$$

For the Haar average, (2.17) and translation invariance give $\mathbb{E}_{\mathrm{Haar}_a}\bar{\ell}(\mathbf{u}) = \mathbb{E}_{\mathrm{Haar}_a}\bar{\ell}(\mathbf{u}-\mathbf{c}) = N\int \phi_\sigma \mathbf{1}_H$. At either child width the normalized shell has outer squared radius at most $2/3+\varepsilon$, so (2.33) gives $N\int \phi_\sigma \mathbf{1}_H \le (4/3)^{n/2} q^{2+o(1)}$.

To bound the Haar average of Π, Lemma 2.6 shows that $\mathbf{u}$ and $\mathbf{u}-\mathbf{c}$ are independent and uniform modulo Λ under Haar_a. Two applications of (2.17) therefore give

$$\mathbb{E}_{\mathrm{Haar}_a}\Pi = N^2 \iint \phi_\sigma(\mathbf{x}')\mathbf{1}_H(\mathbf{x}')\phi_\sigma(\mathbf{y}')\mathbf{1}_H(\mathbf{y}')\, \mathbb{E} m_{\mathcal{C}}(\mathbf{x}', \mathbf{y}')\, d\mathbf{x}' d\mathbf{y}'.$$

Since $\phi_\sigma \mathbf{1}_H$ is radial and $m_{\mathcal{C}}$ depends only on directions (Lemma 4.11), the two directions are independent and uniform, and (4.22) gives $\mathbb{E}_{\mathrm{Haar}_a}\Pi \le (3/2)^{n/2} q^{9/2+o(1)}$.

Using $\bar{\iota} \le (9/8)^{n/2} q^{1/2}\,\mathrm{poly}(n)$, the Haar average of $\bar{\iota}[1+\bar{\ell}(\mathbf{u})+\bar{\ell}(\mathbf{u}-\mathbf{c})]$ is at most $(3/2)^{n/2} q^{5/2+o(1)}$. Including the factor $q^{1/3+o(1)}\mathcal{P}$ gives the second inequality in (4.25).

The root. Both child cosets have the same shift and pairs are enumerated by their quotient labels. Let $\varphi_0 = \phi_{\tau_1}\mathbf{1}_H$, which is even, and $\ell_{\mathrm{ch}} = N \det\Gamma_0\, \varphi_0$. The corresponding list and pair sums are

$$\bar{\ell}_0(\mathbf{u}) = \sum_{\mathbf{y}\in\Gamma_0+\mathbf{u}} \ell_{\mathrm{ch}}(\mathbf{y}),$$
$$\Pi_0(\mathbf{u}) = \sum_{\substack{\mathbf{x}',\mathbf{y}'\in\Gamma_0+\mathbf{u} \\ \mathbf{x}'-\mathbf{y}'\in\mathcal{L}}} \ell_{\mathrm{ch}}(\mathbf{x}')\ell_{\mathrm{ch}}(\mathbf{y}').$$

Define $\mathrm{T}_{\mathrm{root}}(\mathbf{u}) = C_0 q^{o(1)}\mathcal{P}[1+\bar{\ell}_0(\mathbf{u})+\Pi_0(\mathbf{u})]$ and use $\mathrm{T}_{\mathrm{node}}(E_a) = \mathrm{T}_{\mathrm{root}}(\mathbf{u}_a)$ at the root. Sorting and enumeration cost $q^{o(1)}\mathcal{P}$ times the sum of $|X_a|+|Y_a|+1$ and the number of compatible pairs. By (4.23) and (4.24), pairs of two other entries have conditional mean at most $(1+o(1))^2\Pi_0(\mathbf{u})$, while the number of pairs containing a fixed entry is at most $|X_a|+|Y_a|+1$. Together with the list-size bound, this proves the first inequality in (4.25) at the root.

It remains to bound the Haar average of $\mathrm{T}_{\mathrm{root}}$. Equation (2.17) gives $\mathbb{E}_{\mathrm{Haar}_a}\bar{\ell}_0 = N\int\varphi_0$, and (2.19) with $\mathbf{c}=\mathbf{0}$ gives

$$\mathbb{E}_{\mathrm{Haar}_a}\Pi_0 = N^2 \det\Gamma_0 \sum_{\mathbf{v}\in\mathcal{L}} (\varphi_0 * \varphi_0)(\mathbf{v}).$$

The mean of $\bar{\ell}_0$ satisfies the bound already established for $\bar{\ell}$. To bound the mean of Π_0, we compare Gaussian densities:

$$\varphi_0 \le (2/3)^{n/2} e^{b_1 n/4 + O(n/h^2)} \phi_{\sqrt{2/3}\tau_1},$$
$$N^2 \det\Gamma_0 \sum_{\mathbf{v}\in\mathcal{L}} (\varphi_0*\varphi_0)(\mathbf{v}) \le \frac{N^2\widetilde{U}}{p}(2/3)^n e^{b_1 n/2 + O(n/h^2)} \le (4/3)^{n/2} q^{2+o(1)}.$$

Indeed, $2\tau_1/\sqrt{3} \ge \sigma_s$, so Poisson summation and (4.16), using $U \ge \rho_s(\mathcal{L})$, bound the lattice sum of $\phi_{2\tau_1/\sqrt{3}}$ by $\widetilde{U}$. These inequalities give the second bound in (4.25) at the root.

Taking the conditional expectation of the first inequality in (4.25) given E_a, then averaging over Haar_a and using the second inequality, gives

$$\overline{T}_{\mathrm{merge}} \le (3/2)^{n/2} q^5 \mathcal{P}.$$

The leaves. Each leaf has N_{prop} proposal slots, each charged at most $q^{o(1)}\mathcal{P}$, so

$$\overline{T}_{\mathrm{leaf}} \le q^{o(1)}\mathcal{P}N_{\mathrm{prop}} \le (3/2)^{n/2}q^5\mathcal{P},$$

since N_{prop} has rate $\frac{1}{2}\log_2(4/3) < \frac{1}{2}\log_2(3/2)$ by Lemma 4.10(i).

Total work. By Step 2, we may fix the polynomial factor in W_0 so that $\max\{\overline{T}_{\mathrm{merge}}, \overline{T}_{\mathrm{leaf}}\} \le W_0/q$. Substituting this bound and $\delta \le 2\theta_n$ into the union bound above gives

$$\Pr_{\mathbf{z}}\Big[\sum_{a\in\mathcal{N}} C_a > W_0\Big] \le V^2/q + 2Ve^{-\Omega(n/h^4)} = o(1),$$

since $\log V = O(h^3)$ is $o(\log q)$ and $o(n/h^4)$. Every bound above is uniform in $\mathbf{z}$ and the bin.

Step 3: the storage cap. At a nonroot merge, let B_a count eligible occurrence pairs, including those the decoder omits. Both the dictionary size and the parent list size are at most B_a. For the root, let B_a count compatible pairs, which bounds the root list size. A leaf has at most $N_{\mathrm{prop}} \le M_0$ entries for sufficiently large n. We bound the Haar average of $\mathbb{E}_{\mathbf{z}}[B_a \mid E_a]$ for each merge, then transfer its overflow probability back to the fixed shifts.

Use the conditioning in Step 2. Pairs with a fixed entry contribute at most $|X_a| + |Y_a|$ to B_a, whose Haar-averaged conditional expectation is $O(1) + (4/3)^{n/2}q^{2+o(1)}$ by (4.24) and averaging over the shifts. At a nonroot node, bound the contribution of pairs of two other entries by dropping compatibility but requiring $\mathbf{x}' - \mathbf{y}' \in H$. By Fact A and (4.24), their conditional mean is bounded by $(1+o(1))^2$ times the sum of $\ell_{\mathrm{ch}}(\mathbf{x}')\ell_{\mathrm{ch}}(\mathbf{y}')$ over such pairs. Under Haar_a, averaging over both shifts turns this sum into N^2 times an integral over $H \times H$ with density $\phi_\sigma(\mathbf{x}')\phi_\sigma(\mathbf{y}')$. Conditional on the radii, the directions are independent and uniform. If both points and their difference lie in H, their normalized inner product is at least $(1-3\varepsilon)/(2(1+\varepsilon)) \ge 1/2 - 2\varepsilon$. The spherical cap area formula [BM07, Section 3-C] then bounds the conditional probability by $(3/4)^{n/2}e^{O(n\varepsilon)}\,\mathrm{poly}(n)$. Using $N\int_H \phi_\sigma \le (4/3)^{n/2}q^{2+o(1)}$ bounds the Haar-averaged contribution by $(4/3)^{n/2}q^{4+o(1)}$.

At the root, Step 2 already bounds the Haar average of $(1+o(1))^2\Pi_0$, the conditional mean bound for compatible pairs of two other entries, by $(4/3)^{n/2}q^{2+o(1)}$. Thus for every merge a we have

$$\mathbb{E}_{e\sim\mathrm{Haar}_a}[\mathbb{E}_{\mathbf{z}}[B_a \mid E_a = e]] \le (4/3)^{n/2}q^{4+o(1)}, \qquad \Pr_{\mathbf{z}}[B_a > M_0] \le q^{-2+o(1)} + \delta. \tag{4.26}$$

The second bound follows from Markov's inequality and the total variation bound on E_a, applied to the conditional overflow probability. A union bound over fewer than $V = q^{o(1)}$ nodes gives $o(1)$ storage-cap loss. Combining with the work bound proves the lemma. □

4.6 Proof of the sampling theorem

Proof of Theorem 1.4. Algorithm 2 runs on the rescaled input with $\det\mathcal{L} = 1$ and returns ζ^{-1} times its output. The map $\mathbf{x} \mapsto \zeta^{-1}\mathbf{x}$ sends $D_{\zeta\mathcal{L},\zeta s}$ to $D_{\mathcal{L},s}$ and preserves total variation distance, so it suffices to treat $\det\mathcal{L} = 1$. Lemma 4.3 gives error at most $4\cdot 2^{-m} = 2^{-\kappa-1} < 2^{-\kappa}$ and the bounded resources. This resource bound is (1.5). The dimension threshold is independent of $\mathcal{L}$ and s by Lemma 4.9, and of κ by Lemma 4.3.

If $n < n_0$, with n_0 as in Section 4.1, run Algorithm 2 on the orthogonal direct sum $\mathcal{L}' = \mathcal{L} \oplus \mathbb{Z}^{n_0-n} \subset \mathbb{R}^{n_0}$, with basis $\mathbf{B} \oplus \mathbf{I}_{n_0-n}$, the same s and the same κ, and output the first n coordinates. Since $\rho_s(\mathbf{x},\mathbf{y}) = \rho_s(\mathbf{x})\rho_s(\mathbf{y})$, the first n coordinates of a sample from $D_{\mathcal{L}',s}$ have distribution $D_{\mathcal{L},s}$, and discarding coordinates does not increase total variation distance. The cost is that of dimension n_0, which is independent of $\mathcal{L}$, s and κ, so it is polynomial in κ. □

5 Classical SVP in Time $2^{0.4385\ldots n+o(n)}$ – A Single Shell

Theorem 5.1 (Single-shell SVP)**.** *Suppose a family of n-dimensional lattices satisfies the point-count bound (2.3) with $\frac{1}{2}\log_2(4/3) \le \beta \le \frac{1}{2}$. For every $\alpha \in [\frac{1}{2}\log_2(4/3), \alpha_{\max}]$, where $\alpha_{\max} = (\beta + \frac{1}{2}\log_2(4/3))/2$, there is a randomized classical algorithm for exact SVP with success probability at least $2/3$. Its time and space bounds are*

$$\mathrm{T_C} = 2^{t_{\beta,\mathrm{C}}(\alpha)n+o(n)}, \qquad t_{\beta,\mathrm{C}}(\alpha) = \beta - \alpha - \tfrac{1}{2}\log_2(1-2^{2\alpha-2}), \qquad \mathrm{S_C} = 2^{s(\alpha)n+o(n)}. \tag{5.1}$$

Here $s(\alpha) = \alpha$. In particular, for every lattice the algorithm achieves time $2^{0.438578\ldots n+o(n)}$ and space $2^{0.301559\ldots n+o(n)}$.

With coherent indexed access to stored classical arrays, the single-shell quantum algorithm solves exact SVP on every lattice with success probability at least $2/3$ in time $2^{\tau_{\mathrm{Q}}n+o(n)}$ and space $2^{s_{\mathrm{Q}}n+o(n)}$, where

$$\tau_{\mathrm{Q}} = \tfrac{1}{2}\log_2(13/9) + \tfrac{1}{4}\log_2(3e/(2\pi)) = 0.359297\ldots, \quad s_{\mathrm{Q}} = \tfrac{1}{4}\log_2(8e/(3\pi)) = 0.301559\ldots. \tag{5.2}$$

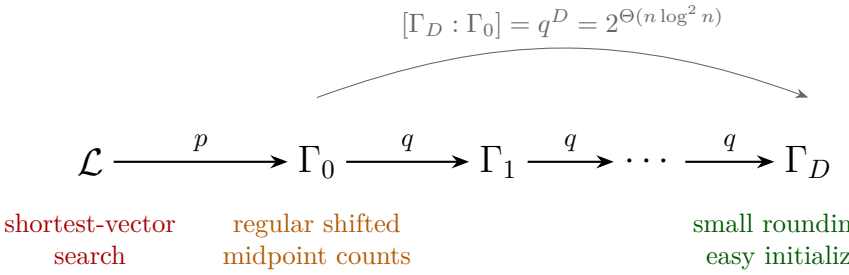
The aim is to produce two lists whose differences include a shortest vector of the input lattice. The list vectors themselves need not lie in the input lattice. They start in a denser superlattice, where sampling from a shell is easier, and each merge brings their differences closer to the input lattice. Every list uses the same shell, of radius close to the shortest-vector length.

1. **Prepare a chain for sampling and merging.** The first random extension makes counts in a thin midpoint shell concentrate over random shifts. At the other end of the chain, repeated basis halvings make the rounding cells small. This lets us sample each leaf list by rounding uniform points from a slightly enlarged shell.

2. **Balance the representations of each difference.** At every internal merge, draw a fresh shift shared by the two child lists, with an offset chosen so that the desired differences can lie in the parent coset. Averaging over the shared shift turns counts of representations into volumes. The number of pairs with a prescribed difference otherwise varies with the length of that difference. To make these volumes nearly equal, accept a pair only if its midpoint lies in a thin shell. When the two inputs and their difference have approximately the same length, the midpoint length is approximately $\sqrt{3}/2$ times that length.

3. **Use moments to show that a shortest difference appears.** For each vector, count how often it occurs in a list. We bound the expected count of each value and the expected number of pairs of distinct list entries with prescribed values. These estimates apply even when entries within a list are dependent. In the final search, the two lists are on the same shifted superlattice. We look for the shortest nonzero difference in the input lattice and use the moment bounds to show that such a pair is present with sufficient probability.

4. **Search close pairs by spherical filtering.** Examining every pair already gives an algorithm. Useful pairs have directions at an angle near $\pi/3$, so spherical caps can find them more quickly. Increasing the size of the two final lists raises the success probability and reduces the number of trials. The theorem balances the cost of larger lists against the cost of repeating a trial.



(5.3)

Roadmap. Section 5.1 constructs the superlattice chain and bounds the list moments. Section 5.2 analyzes exhaustive pair search. Section 5.3 uses spherical filtering to prove Theorem 5.1. Section 5.4 gives a quantum implementation. Section 5.5 evaluates the classical and quantum bounds.

5.1 The superlattice chain

Normalize the input lattice $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$ to determinant one. The algorithm tries the length guesses from Lemma 2.8. We analyze a guess λ satisfying $\lambda_1(\mathcal{L}) \le \lambda \le (1 + n^{-2})\lambda_1(\mathcal{L})$. In the point-count bound (2.3), take $b \ge \sqrt{4/3}$ and absorb the prefactor $A_n \ge 1$ into b as in Section 2.1. Write $\beta = \log_2 b$ for the resulting exponent, which differs from the exponent in the theorem by $o(1)$. For input classes with smaller point counts, such as Haar-random lattices, we use the bound with $b \ge \sqrt{4/3}$.

5.1.1 The first superlattice Γ_0

Let $q = 2^a$ be a power-of-two integer for $a = \lceil n/\log_2 n \rceil$. The midpoint shell in Figure 8 is

$$A = \{\mathbf{w} : \sqrt{3}\lambda/2 - \lambda n^{-2} \le \|\mathbf{w}\| \le \sqrt{3}\lambda/2 + \lambda n^{-2}\}, \qquad \mathrm{vol}(A) = \Theta(n^{-3/2}) \left(\frac{3\pi e \lambda^2}{2n}\right)^{n/2} \tag{5.4}$$

The volume estimate follows from Stirling's formula. We choose a prime satisfying

$$\frac{q(3b^2/4)^{n/2}}{\mathrm{vol}(A)} \le p \le \frac{2q(3b^2/4)^{n/2}}{\mathrm{vol}(A)}. \tag{5.5}$$

Following Section 3, draw a uniform $\mathbf{z} \in \mathbb{F}_p^n \setminus \{\mathbf{0}\}$ and define

$$\Gamma_0 = \mathcal{L} + \mathbb{Z}\frac{\mathbf{B}\bar{\mathbf{z}}}{p},$$

where $\bar{\mathbf{z}} \in \mathbb{Z}^n$ is a representative of $\mathbf{z}$. Then $[\Gamma_0 : \mathcal{L}] = p$ and $\det \Gamma_0 = 1/p$.

Lemma 5.2 (Bounding dual Gaussian mass of $\mathcal{L}^*$)**.** *For sufficiently large n, $\rho_{\sqrt{2n/(3\pi e)}/\lambda}(\mathcal{L}^*) \le p/q$. In particular, $p \ge q$.*

Proof. Lemma 2.4 with $s = \lambda\sqrt{3\pi e/(2n)}$ and the Poisson summation formula give

$$\rho_{1/s}(\mathcal{L}^*) = \frac{\rho_s(\mathcal{L})}{s^n} \le \frac{O(n^{-1/2})(3b^2/4)^{n/2}}{\mathrm{vol}(A)} \le \frac{(3b^2/4)^{n/2}}{\mathrm{vol}(A)} \le \frac{p}{q}$$

where we absorbed the term 1 in the asymptotic term using $b \ge \sqrt{4/3}$ in the first inequality. The inequality $p \ge q$ holds because $\rho_{1/s}(\mathcal{L}^*) \ge \rho_{1/s}(\mathbf{0}) = 1$. □

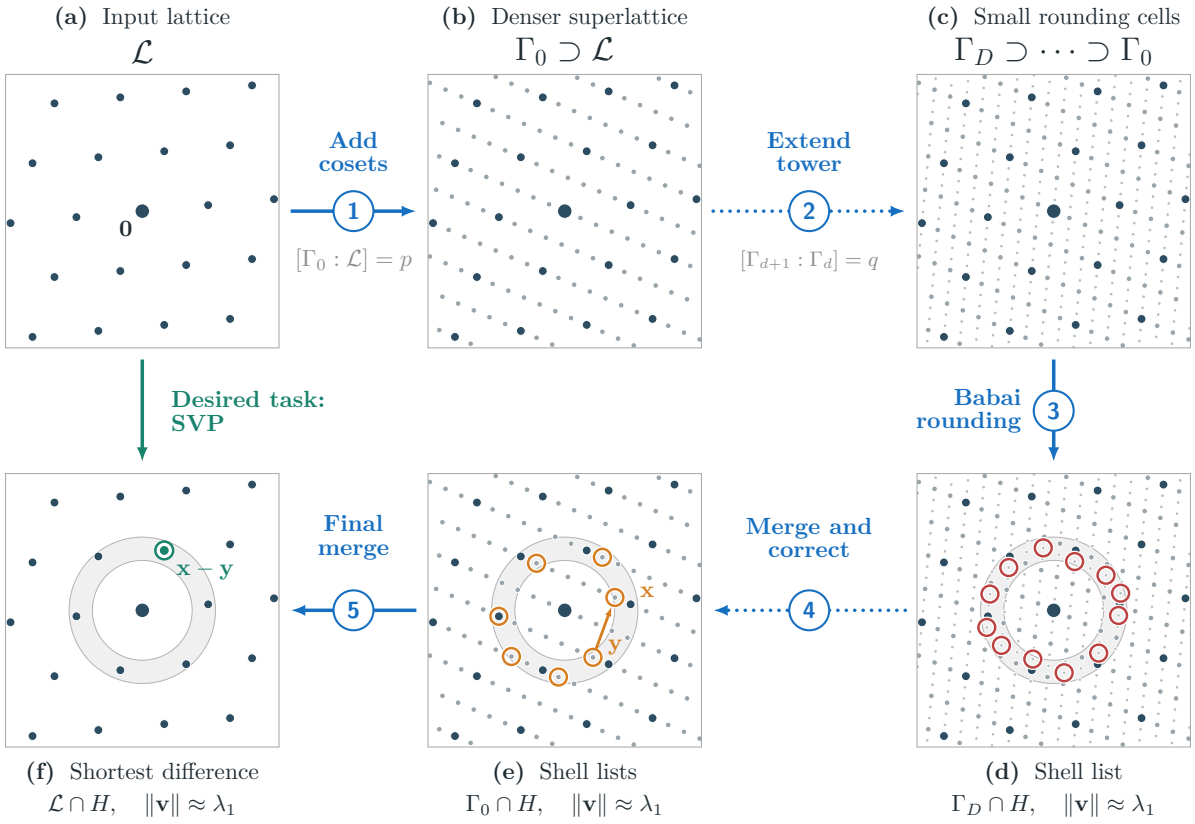


Figure 7: SVP with one shell H of radius $\lambda \approx \lambda_1(\mathcal{L})$. In (d), Babai rounding gives a list on Γ_D. Merges with a midpoint test give the lists on Γ_0 in (e). The final search in (f) finds a shortest nonzero difference $\mathbf{x} - \mathbf{y} \in \mathcal{L}$. All three lower panels use the same shell.

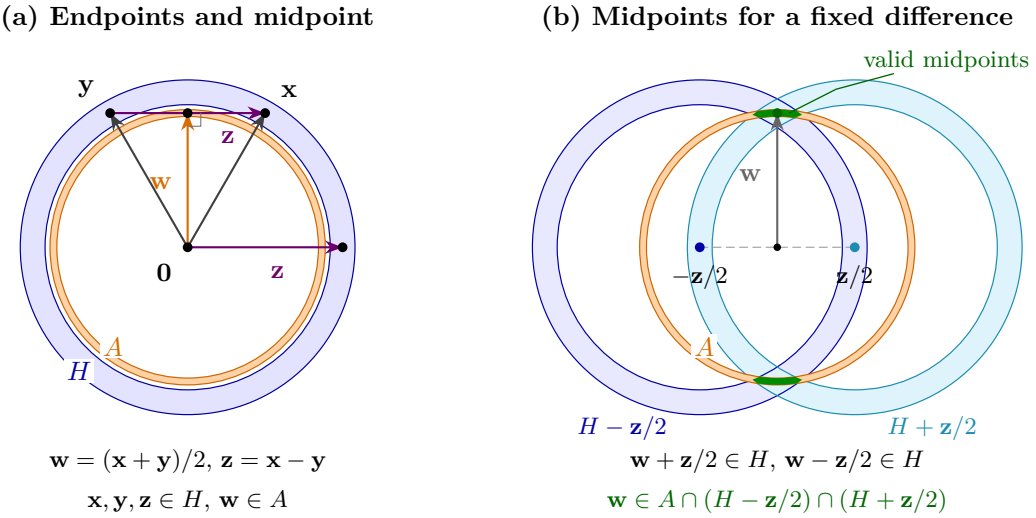


Figure 8: Midpoint-shell geometry. The list shell is $H = \{\mathbf{v} : 1 - n^{-1/3} \le \|\mathbf{v}\|/\lambda \le 1 + n^{-1/3}\}$, and the midpoint shell is $A = \{\mathbf{v} : \sqrt{3}/2 - n^{-2} \le \|\mathbf{v}\|/\lambda \le \sqrt{3}/2 + n^{-2}\}$. On the left, two list vectors have difference $\mathbf{z} = \mathbf{x} - \mathbf{y} \in H$ and midpoint $\mathbf{w} = (\mathbf{x} + \mathbf{y})/2 \in A$. On the right, the green region $A \cap (H - \mathbf{z}/2) \cap (H + \mathbf{z}/2)$ contains the valid midpoints for the fixed difference $\mathbf{z}$.

For a full-rank lattice Λ, define

$$K_\Lambda(\mathbf{u}) = \#((\Lambda + \mathbf{u}) \cap A), \qquad m_\Lambda = \frac{\mathrm{vol}(A)}{\det \Lambda}. \tag{5.6}$$

Expectations and variances of K_Λ are taken over $\mathbf{u}$ uniform modulo Λ. Equation (2.18) gives $\mathbb{E}K_\Lambda = m_\Lambda$.

Lemma 5.3 (Shell-count estimates)**.** *We have*

$$\frac{\mathbb{E}K_{\mathcal{L}}^2}{pm_{\mathcal{L}}^2} \le O(n^{3/2})q^{-1}, \qquad \frac{\mathrm{Var}\, K_{\mathcal{L}/p}}{m_{\mathcal{L}/p}^2} \le O(n^5)q^{-2}.$$

We first derive the following consequence and then prove Lemma 5.3.

Lemma 5.4 (Shell-count variance)**.** *With probability at least* $1 - e^{-n^{1/4}}$ *over* Γ_0*, every* $\Lambda \supseteq \Gamma_0$ *satisfies* $\mathrm{Var}\, K_\Lambda \le e^{-n^{1/4}} m_\Lambda^2$.

Proof. Let $f = \mathbf{1}_A/\mathrm{vol}(A)$. Since $P_{\Lambda,f} = K_\Lambda/m_\Lambda$ and $\int f = 1$, Parseval's identity gives

$$\frac{\mathrm{Var}\, K_\Lambda}{m_\Lambda^2} = \sum_{\mathbf{y} \in \Lambda^* \setminus \{\mathbf{0}\}} |\widehat{f}(\mathbf{y})|^2.$$

Applying Equation (3.2) to $|\widehat{f}|^2$ therefore gives

$$\mathbb{E}_{\Gamma_0} \frac{\mathrm{Var}\, K_{\Gamma_0}}{m_{\Gamma_0}^2} \le \frac{1}{p}\frac{\mathrm{Var}\, K_{\mathcal{L}}}{m_{\mathcal{L}}^2} + \frac{\mathrm{Var}\, K_{\mathcal{L}/p}}{m_{\mathcal{L}/p}^2} \le O(n^{3/2})q^{-1} + O(n^5)q^{-2} \le e^{-2n^{1/4}},$$

where we used $\mathrm{Var}\, K_{\mathcal{L}} \le \mathbb{E}K_{\mathcal{L}}^2$ and Lemma 5.3. By Markov's inequality, with probability at least $1 - e^{-n^{1/4}}$,

$$\frac{\mathrm{Var}\, K_{\Gamma_0}}{m_{\Gamma_0}^2} \le e^{-n^{1/4}}.$$

Whenever this bound holds, it extends simultaneously to every $\Lambda \supseteq \Gamma_0$ using the Parseval identity: since $\Lambda^* \subseteq \Gamma_0^*$, we have

$$\frac{\mathrm{Var}\, K_\Lambda}{m_\Lambda^2} \le \frac{\mathrm{Var}\, K_{\Gamma_0}}{m_{\Gamma_0}^2} \le e^{-n^{1/4}}. \qquad \square$$

Proof of Lemma 5.3. Let $r = \lambda(\sqrt{3}/2 + n^{-2})$. For $\mathbf{x} \in A \cap (A + \mathbf{v})$, both $\|\mathbf{x}\|$ and $\|\mathbf{x} - \mathbf{v}\|$ are at most r. The parallelogram identity gives

$$\left\|\mathbf{x} - \frac{\mathbf{v}}{2}\right\|^2 = \frac{\|\mathbf{x}\|^2 + \|\mathbf{x} - \mathbf{v}\|^2}{2} - \frac{\|\mathbf{v}\|^2}{4} \le r^2 - \frac{\|\mathbf{v}\|^2}{4}.$$

Thus the intersection lies in the ball centered at $\mathbf{v}/2$ with radius $\sqrt{r^2 - \|\mathbf{v}\|^2/4}$ if $\|\mathbf{v}\| \le 2r$, giving

$$\mathrm{vol}(A \cap (A + \mathbf{v})) \le V_n r^n \left(1 - \frac{\|\mathbf{v}\|^2}{4r^2}\right)_+^{n/2},$$

where V_n is the unit-ball volume and $a_+ = \max\{a, 0\}$. For every $R > 0$, integration by parts, Equation (2.3), and Stirling's formula give

$$\sum_{\mathbf{v}\in\mathcal{L}\setminus\{\mathbf{0}\}} \left(1 - \frac{\|\mathbf{v}\|^2}{4R^2}\right)_+^{n/2} = \frac{n}{4R^2}\int_0^{2R} N_{\mathcal{L}}(u)\, u \left(1 - \frac{u^2}{4R^2}\right)^{n/2-1} du \tag{5.7}$$

$$\leq \left(\frac{2bR}{\lambda_1(\mathcal{L})}\right)^n \frac{\Gamma(n/2+1)^2}{\Gamma(n+1)} \leq O(\sqrt{n}) \left(\frac{bR}{\lambda_1(\mathcal{L})}\right)^n. \tag{5.8}$$

Taking $R = r$ and using $\lambda/\lambda_1(\mathcal{L}) \leq 1 + n^{-2}$ bounds this sum by $O(\sqrt{n})(3b^2/4)^{n/2}$.

Since $\det \mathcal{L} = 1$, $m_{\mathcal{L}} = \mathrm{vol}(A)$ and Equation (2.18) with $f = \mathbf{1}_A/\mathrm{vol}(A)$ gives

$$\frac{\mathbb{E}K_{\mathcal{L}}^2}{m_{\mathcal{L}}^2} = \frac{1}{\mathrm{vol}(A)^2}\sum_{\mathbf{v}\in\mathcal{L}} \mathrm{vol}(A \cap (A+\mathbf{v})) = \frac{1}{\mathrm{vol}(A)} + \frac{1}{\mathrm{vol}(A)^2}\sum_{\mathbf{v}\in\mathcal{L}\setminus\{\mathbf{0}\}} \mathrm{vol}(A \cap (A+\mathbf{v}))$$

$$\leq \frac{1}{\mathrm{vol}(A)} + \frac{V_n r^n}{\mathrm{vol}(A)^2} O(\sqrt{n})(3b^2/4)^{n/2}.$$

We prove the first inequality by

$$\frac{\mathbb{E}K_{\mathcal{L}}^2}{pm_{\mathcal{L}}^2} \leq \frac{1 + O(n^{3/2})(3b^2/4)^{n/2}}{p\,\mathrm{vol}(A)} \leq O(n^{3/2})q^{-1}$$

where the first inequality uses $V_n r^n/\mathrm{vol}(A) = O(n)$, and the last inequality is due to the choice of p (Equation (5.5)) and $b^2 \geq 4/3$ to absorb the term 1.

Next, we prove the second inequality. For $s = \lambda\sqrt{3\pi e/(2n)}$, the one-dimensional Poisson summation formula applied to the multiples of a shortest dual vector gives

$$\rho_{1/s}(\mathcal{L}^*) \geq \sum_{k\in\mathbb{Z}} e^{-\pi s^2 k^2 \lambda_1(\mathcal{L}^*)^2} = \frac{1}{s\lambda_1(\mathcal{L}^*)}\sum_{k\in\mathbb{Z}} e^{-\pi k^2/(s^2\lambda_1(\mathcal{L}^*)^2)} \geq \frac{1}{s\lambda_1(\mathcal{L}^*)}.$$

We use the Voronoi cells of the full shifted lattice $\mathcal{L}/p + \mathbf{u}$, taking only those whose centres lie in A. For any shift $\mathbf{u}$, let U be the union of Voronoi cells centered at $(\mathcal{L}/p + \mathbf{u}) \cap A$. Each cell consists of points for which its centre is a nearest point of $\mathcal{L}/p + \mathbf{u}$. The cells have disjoint interiors and volume $\det(\mathcal{L}/p)$, so

$$\mathrm{vol}(U) = K_{\mathcal{L}/p}(\mathbf{u}) \det(\mathcal{L}/p).$$

Let R_{cov} be the covering radius of $\mathcal{L}/p$. Lemma 5.2 and the covering-radius transference bound [Ban93] give

$$R_{\mathrm{cov}} \leq \frac{n}{2p\lambda_1(\mathcal{L}^*)} \leq \frac{ns}{2q} = O(\lambda\sqrt{n})/q.$$

Every point in a cell is within distance R_{cov} of its centre. Hence $U \triangle A = (U \setminus A) \cup (A \setminus U)$ lies within distance R_{cov} of the two boundary spheres of A. Subtracting ball volumes gives

$$|\,\mathrm{vol}(U) - \mathrm{vol}(A)| \leq \mathrm{vol}(U \triangle A) \leq 4nV_n(r + R_{\mathrm{cov}})^{n-1} R_{\mathrm{cov}} = O(nV_n r^{n-1} R_{\mathrm{cov}}),$$

where the last equality uses $nR_{\mathrm{cov}}/r = O(n^{3/2}/q) = o(1)$, since $r = \Theta(\lambda)$. Dividing by $\mathrm{vol}(A)$ and using $V_n r^n/\mathrm{vol}(A) = O(n)$ gives

$$\left|\frac{K_{\mathcal{L}/p}(\mathbf{u})}{m_{\mathcal{L}/p}} - 1\right| \leq O(n^2 R_{\mathrm{cov}}/\lambda) \leq O(n^{5/2})/q$$

Squaring and averaging proves the second bound. □

5.1.2 The remaining superlattice chain

We extend Γ_0 by cyclically halving basis vectors as in the proof of [ADRS15, Claim 5.9]. Recall $q = 2^a$ for $a = \lceil n/\log_2 n \rceil$. Apply Theorem 2.9 to Γ_0 with block size a and failure probability n^{-2}, obtaining a basis $\mathbf{B}^{(0)} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$. Define successive bases by

$$\begin{aligned}\mathbf{B}^{(1)} &= (\mathbf{b}_1/2, \dots, \mathbf{b}_a/2, \mathbf{b}_{a+1}, \dots, \mathbf{b}_n)\,,\\ \mathbf{B}^{(2)} &= (\mathbf{b}_1/2, \dots, \mathbf{b}_{2a}/2, \mathbf{b}_{2a+1}, \dots, \mathbf{b}_n)\,,\\ &\vdots\end{aligned}$$

At each step, divide the next a basis vectors by two, returning to the first vector after the last. Continue for $D = \lceil (\log_2 n)^3 \rceil$ steps and define $\Gamma_j = \mathbf{B}^{(j)}\mathbb{Z}^n$. Then

$$\Gamma_0 \subseteq \cdots \subseteq \Gamma_D, \qquad [\Gamma_{j+1} : \Gamma_j] = q, \qquad \det \Gamma_j = (pq^j)^{-1}.$$

Abort unless $\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| \le 2\lambda/n^{3/2}$.

Lemma 5.5 (Properties of the superlattice chain)**.** *With probability $1 - o(1)$, the construction returns a chain satisfying*

$$\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| \le 2\lambda/n^{3/2}, \qquad \operatorname{Var} K_{\Gamma_j} \le e^{-n^{1/4}} m_{\Gamma_j}^2 \quad (0 \le j \le D).$$

Given Γ_0, constructing the remaining chain takes time $2^{O(n/\log n)}$.

Proof. Lemmas 3.2 and 5.2, together with Markov's inequality, give $\eta_{1/2}(\Gamma_0) \le \lambda\sqrt{3\pi e/(2n)}$ except with probability $4/q$. Hence $\lambda_n(\Gamma_0) = O(\lambda)$ by Lemma 2.2. Lemma 5.4 gives the variance bound.

The basis vectors are divided by two in the order $\mathbf{b}_1, \dots, \mathbf{b}_n, \mathbf{b}_1, \dots$, for a total of Da divisions. Thus each basis vector is divided by two at least $\lfloor Da/n \rfloor$ times. Scaling a basis vector leaves the spans of the preceding vectors unchanged, so its Gram–Schmidt vector is scaled by the same factor. Theorem 2.9 therefore gives

$$\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| \le 2^{-\lfloor Da/n \rfloor} \max_i \|\widetilde{\mathbf{b}}_i^{(0)}\| \le O(\lambda) 2^{-\lfloor Da/n \rfloor} a^{n/a} \le 2\lambda/n^{3/2}$$

except with probability n^{-2}. The last inequality holds for sufficiently large n by the choices of a and D. A union bound gives total failure probability at most $4/q + n^{-2} + e^{-n^{1/4}} = o(1)$. The block reduction takes time $2^{O(n/\log n)}$; all subsequent operations have polynomial cost. □

Sampling along the chain. To sample a list on a shifted superlattice, we draw uniform points in a shell and round them. The target shell for the rounded points is

$$H = \{\mathbf{x} \in \mathbb{R}^n : |\|\mathbf{x}\| - \lambda| \le \lambda n^{-1/3}\}. \tag{5.9}$$

The proposal shell is slightly larger to account for the rounding distance.

$$H^+ = \{\mathbf{x} : |\|\mathbf{x}\| - \lambda| \le \lambda(n^{-1/3} + n^{-1})\}, \qquad N = \left\lceil \frac{q \operatorname{vol}(H^+)}{\operatorname{vol}(A)} \right\rceil = (4/3)^{n/2+o(n)}. \tag{5.10}$$

Recall A is the midpoint shell defined in (5.4). At a leaf with shift $\mathbf{t}$, the sampler returns points in $(\Gamma_D + \mathbf{t}) \cap H$. The recursion below specifies the shifts. Let $\mathsf{NP}_{\mathbf{B}^{(D)}}$ denote Babai's nearest-plane rounding in the basis $\mathbf{B}^{(D)}$ [Bab86]. By Lemma 5.5, its rounding distance is at most λ/n.

For each intermediate lattice Γ_j, $j < D$, and target shift $\mathbf{t}$, a list on $\Gamma_j + \mathbf{t}$ is constructed by merging two child lists X, Y on

$$\Gamma_{j+1} + \mathbf{u}, \qquad \Gamma_{j+1} + \mathbf{u} - \mathbf{t}, \tag{5.11}$$

respectively, where $\mathbf{u}$ is uniform modulo Γ_{j+1}. This step is $\mathsf{Merge}(j, \mathbf{t}, X, Y)$.

Algorithm 9 Leaf sampling $\mathsf{Sample}(\mathbf{t})$

Input: a shift $\mathbf{t}$.

$L \leftarrow [\,]$.
for $r = 1, \dots, N$ **do**
  Draw $\mathbf{U}$ uniformly from H^+ and let $\mathbf{x} \leftarrow \mathsf{NP}_{\mathbf{B}^{(D)}}(\mathbf{U} - \mathbf{t}) + \mathbf{t}$.
  if $\mathbf{x} \in H$ **then**
    Append $\mathbf{x}$ to L with probability $q \operatorname{vol}(H^+)/(N \operatorname{vol}(A)) \le 1$.
return L.

Algorithm 10 Internal merge $\mathsf{Merge}(j, \mathbf{t}, X, Y)$

Input: a level $0 \le j < D$, a target shift $\mathbf{t}$, and two lists X, Y in (5.11).

$L \leftarrow [\,]$.
for each occurrence pair $(\mathbf{x}, \mathbf{y}) \in X \times Y$ **do**
  if $\mathbf{x} - \mathbf{y} - \mathbf{t} \in \Gamma_j$, $\mathbf{x} - \mathbf{y} \in H$, and $(\mathbf{x} + \mathbf{y})/2 \in A$ **then**
    Append $\mathbf{x} - \mathbf{y}$ to L.
return L.

The midpoint condition. For a candidate pair, the parallelogram identity gives

$$\left\| \frac{\mathbf{x} + \mathbf{y}}{2} \right\|^2 = \frac{\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2}{2} - \frac{\|\mathbf{x} - \mathbf{y}\|^2}{4} \approx \frac{3\lambda^2}{4} \tag{5.12}$$

when both inputs and their difference lie in H. To see why the midpoint test balances the output weights, consider two independent proposal slots $\mathbf{X}, \mathbf{Y}$ conditional on the common shift. Each slot assigns the same weight to each point of its list shell, independently of the shift. Average over the shift and condition on a compatible difference in H. For $\mathbf{z} \in (\Gamma_j + \mathbf{t}) \cap H$, the resulting distribution without the midpoint test is

$$\Pr[\mathbf{X} - \mathbf{Y} = \mathbf{z} \mid E] = \frac{\operatorname{vol}(H \cap (H + \mathbf{z}))}{\sum_{\mathbf{v} \in (\Gamma_j + \mathbf{t}) \cap H} \operatorname{vol}(H \cap (H + \mathbf{v}))}, \tag{5.13}$$

where E means that the difference lies in $(\Gamma_j + \mathbf{t}) \cap H$. With the midpoint test, the distribution is

$$\Pr\left[\mathbf{X} - \mathbf{Y} = \mathbf{z} \,\middle|\, E,\ \frac{\mathbf{X} + \mathbf{Y}}{2} \in A \right] = \frac{1 + o(1)}{|(\Gamma_j + \mathbf{t}) \cap H|}. \tag{5.14}$$

For every such $\mathbf{z}$, almost every midpoint in A places both endpoints in H, as shown in Lemma 5.9. The proof below applies this calculation to the expected counts of list entries.

Analysis of Sample**.** We first analyze the output lists of Sample. Let $\mathbf{1}_H$ be the indicator of H. For a list on $\Gamma_j + \mathbf{t}$, let $A_{\mathbf{x}}$ denote the number of times $\mathbf{x}$ occurs in the output list. Define

$$\ell_j(\mathbf{x}) = \frac{q \det \Gamma_j}{\operatorname{vol}(A)} \mathbf{1}_H(\mathbf{x}). \tag{5.15}$$

Lemma 5.6 (Sampling on $\Gamma_D + \mathbf{t}$)**.** *If* $\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| \le 2\lambda/n^{3/2}$*, the list returned by* $\mathsf{Sample}(\mathbf{t})$ *consists of independent uniform samples from* $(\Gamma_D + \mathbf{t}) \cap H$*. Moreover, for* $\mathbf{x}, \mathbf{y} \in \Gamma_D + \mathbf{t}$*,*

$$\mathbb{E} A_{\mathbf{x}} = \ell_D(\mathbf{x}), \qquad \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}})] \le \ell_D(\mathbf{x}) \ell_D(\mathbf{y}).$$

Proof. Points outside H are never appended, so $A_{\mathbf{x}} = \ell_D(\mathbf{x}) = 0$ for $\mathbf{x} \in (\Gamma_D + \mathbf{t}) \setminus H$. It therefore suffices to consider $\mathbf{x}, \mathbf{y} \in (\Gamma_D + \mathbf{t}) \cap H$.

For each $\mathbf{x} \in \Gamma_D + \mathbf{t}$, the set of points rounded to $\mathbf{x}$ is the translate by $\mathbf{x}$ of

$$\left\{ \sum_i v_i \widetilde{\mathbf{b}}_i^{(D)} : -\tfrac{1}{2} \le v_i < \tfrac{1}{2} \right\}. \tag{5.16}$$

This set has volume $\det \Gamma_D$, and every point in it is within distance λ/n of $\mathbf{x}$. The entire set therefore lies in H^+ because

$$\big|\|\mathbf{U}\| - \lambda\big| \le \|\mathbf{U} - \mathbf{x}\| + \big|\|\mathbf{x}\| - \lambda\big| \le \lambda/n + \lambda n^{-1/3}.$$

Thus, in each iteration, any given $\mathbf{x} \in (\Gamma_D + \mathbf{t}) \cap H$ is appended with probability

$$\frac{\det \Gamma_D}{\operatorname{vol}(H^+)} \cdot \frac{q \operatorname{vol}(H^+)}{N \operatorname{vol}(A)} = \frac{\ell_D(\mathbf{x})}{N}.$$

This probability is the same for every such $\mathbf{x}$, which proves uniformity. The proposal slots are independent, so, conditional on the list length, the accepted points are independent uniform samples. The two moment bounds follow from the independence of the proposal slots. □

Analysis of Merge**.** For a target shift $\mathbf{t}$ at level $j < D$, draw $\mathbf{u}$ uniformly modulo Γ_{j+1} and recursively construct lists X, Y on the translates in Equation (5.11), with target shifts $\mathbf{u}$ and $\mathbf{u} - \mathbf{t}$, respectively. Their output is combined by $\mathsf{Merge}(j, \mathbf{t}, X, Y)$ to obtain a list on $\Gamma_j + \mathbf{t}$. At level D, use $\mathsf{Sample}(\mathbf{t})$. We call the list from one such recursive construction a *base list* and its recursion tree a *base tree*.

Let $\xi = e^{-n^{1/4}}$. Assume that $\operatorname{Var} K_{\Gamma_j} \le \xi m_{\Gamma_j}^2$ for $0 \le j \le D$ and $\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| \le 2\lambda/n^{3/2}$. These bounds hold with probability $1 - o(1)$ by Lemma 5.5.

Lemma 5.7 (Moment bounds)**.** *For every list on $\Gamma_j + \mathbf{t}$ generated as above and $\mathbf{x}, \mathbf{y} \in \Gamma_j + \mathbf{t}$, uniformly in $\mathbf{t}$,*

$$(1 - o(1))\ell_j(\mathbf{x}) \le \mathbb{E} A_{\mathbf{x}} \le \ell_j(\mathbf{x}), \qquad \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}})] \le (1 + o(1))\ell_j(\mathbf{x})\ell_j(\mathbf{y}). \tag{5.17}$$

The second expectation counts ordered pairs of distinct list entries with values $\mathbf{x}$ and $\mathbf{y}$. If the values coincide, the subtraction excludes using the same entry twice. The upper bound $\mathbb{E} A_{\mathbf{x}} \le \ell_j(\mathbf{x})$ also holds without the variance assumption.

Proof. At level D, Lemma 5.6 gives the exact first moment and an upper bound on the expected number of pairs of distinct entries. Applying Lemma 5.8 below inductively gives

$$(1 - \xi)^{2^{D-j}-1}\ell_j(\mathbf{x}) \le \mathbb{E} A_{\mathbf{x}} \le \ell_j(\mathbf{x}), \qquad \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}})] \le (1 + 3\xi)^{2^{D-j}-1}\ell_j(\mathbf{x})\ell_j(\mathbf{y}).$$

The multiplicative factors tend to one because $2^D \xi = o(1)$. The upper first-moment bound follows in the same way without the variance assumption, as shown in the proof of Lemma 5.8. □

The induction uses the following estimate for a single application of Merge.

Lemma 5.8 (One Merge)**.** *Fix $0 \le j < D$ and a target shift $\mathbf{t}$. Let $\mathbf{u}$ be uniform modulo Γ_{j+1}. Conditional on $\mathbf{u}$, let X, Y be independent random lists on $\Gamma_{j+1} + \mathbf{u}$ and $\Gamma_{j+1} + \mathbf{u} - \mathbf{t}$, respectively.*

For either input list, let $B_{\mathbf{x}}$ count the number of times $\mathbf{x}$ occurs in that list. Suppose that, for every $\mathbf{u}$ and every $\mathbf{x}, \mathbf{y}$ in the corresponding translate,

$$\alpha_{\mathrm{mom}}\ell_{j+1}(\mathbf{x}) \le \mathbb{E}[B_{\mathbf{x}} \mid \mathbf{u}] \le \ell_{j+1}(\mathbf{x}), \qquad \mathbb{E}[B_{\mathbf{x}}(B_{\mathbf{y}} - \mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}}) \mid \mathbf{u}] \le f\ell_{j+1}(\mathbf{x})\ell_{j+1}(\mathbf{y})$$

for some $0 \le \alpha_{\mathrm{mom}} \le 1$ *and* $f \ge 1$. *Then the occurrence counts in* $\mathsf{Merge}(j, \mathbf{t}, X, Y)$ *satisfy*

$$\alpha_{\mathrm{mom}}^2(1-\xi)\ell_j(\mathbf{z}) \le \mathbb{E}A_{\mathbf{z}} \le \ell_j(\mathbf{z}), \qquad \mathbb{E}[A_{\mathbf{z}}(A_{\mathbf{z}'} - \mathbf{1}_{\{\mathbf{z}=\mathbf{z}'\}})] \le (1+3\xi)f^2\ell_j(\mathbf{z})\ell_j(\mathbf{z}')$$

for all $\mathbf{z}, \mathbf{z}' \in \Gamma_j + \mathbf{t}$. *The upper bound* $\mathbb{E}A_{\mathbf{z}} \le \ell_j(\mathbf{z})$ *also holds without the variance assumption.*

To prove Lemma 5.8, we study the number of pairs with a fixed difference. For $\mathbf{z} \in H$, define

$$E_{\mathbf{z}} = \{\mathbf{x} : \mathbf{x}, \mathbf{x}-\mathbf{z} \in H,\ \mathbf{x} - \mathbf{z}/2 \in A\}, \qquad K_{\Gamma_j,\mathbf{z}}(\mathbf{u}) = \#((\Gamma_j + \mathbf{u}) \cap E_{\mathbf{z}}) \tag{5.18}$$

analogous to (5.6). Thus $K_{\Gamma_j,\mathbf{z}}$ counts pairs $(\mathbf{x}, \mathbf{x}-\mathbf{z})$ whose two points lie in H and whose midpoint lies in A. Since $E_{\mathbf{z}} \subseteq A + \mathbf{z}/2$, $K_{\Gamma_j,\mathbf{z}}(\mathbf{u}) \le K_{\Gamma_j}(\mathbf{u} - \mathbf{z}/2)$ holds. For every fixed $\mathbf{z} \in H$, the next lemma shows that a uniform $\mathbf{W} \in A$ satisfies $\mathbf{W} + \mathbf{z}/2, \mathbf{W} - \mathbf{z}/2 \in H$ with probability at least $1 - \xi$.

Lemma 5.9 (Pairs with a fixed difference)**.** *Uniformly in* $\mathbf{z}, \mathbf{z}' \in H$ *and* $0 \le j \le D$,

$$(1-\xi)\,\mathrm{vol}(A) \le \mathrm{vol}(E_{\mathbf{z}}) \le \mathrm{vol}(A), \tag{5.19}$$

$$(1-\xi)m_{\Gamma_j} \le \mathbb{E}_{\mathbf{u}}K_{\Gamma_j,\mathbf{z}} \le m_{\Gamma_j}, \tag{5.20}$$

$$\mathbb{E}_{\mathbf{u}}[K_{\Gamma_j,\mathbf{z}}K_{\Gamma_j,\mathbf{z}'}] \le (1+\xi)m_{\Gamma_j}^2, \tag{5.21}$$

where $\mathbf{u}$ *is uniform modulo* Γ_j.

Proof. For $\mathbf{W}$ uniform in A, the definitions of A and H give $\mathbf{W} \pm \mathbf{z}/2 \in H$ whenever $|\langle \mathbf{W}, \mathbf{z}\rangle| \le n^{-1/3}\|\mathbf{W}\|\|\mathbf{z}\|$. Since $\mathbf{W}/\|\mathbf{W}\|$ is uniform on the unit sphere, the exceptional probability is at most $2\sqrt{n}\,e^{-(n-3)n^{-2/3}/2} \le \xi$. This proves the volume bound. Equation (2.17) then gives

$$\mathbb{E}_{\mathbf{u}}K_{\Gamma_j,\mathbf{z}} = \frac{\mathrm{vol}(E_{\mathbf{z}})}{\det\Gamma_j},$$

proving the mean bounds.

By translation invariance of uniform measure and the assumed variance bound,

$$\mathbb{E}_{\mathbf{u}}K_{\Gamma_j,\mathbf{z}}(\mathbf{u})^2 \le \mathbb{E}_{\mathbf{u}}K_{\Gamma_j}(\mathbf{u} - \mathbf{z}/2)^2 \le (1+\xi)m_{\Gamma_j}^2.$$

The Cauchy–Schwarz inequality therefore gives

$$\mathbb{E}_{\mathbf{u}}[K_{\Gamma_j,\mathbf{z}}K_{\Gamma_j,\mathbf{z}'}] \le \sqrt{\mathbb{E}_{\mathbf{u}}K_{\Gamma_j,\mathbf{z}}^2\,\mathbb{E}_{\mathbf{u}}K_{\Gamma_j,\mathbf{z}'}^2} \le (1+\xi)m_{\Gamma_j}^2. \qquad \square$$

Proof of Lemma 5.8. Points outside H are never appended, so it suffices to consider $\mathbf{z}, \mathbf{z}' \in (\Gamma_j + \mathbf{t}) \cap H$. Since $\det\Gamma_j = q\det\Gamma_{j+1}$, for $\mathbf{x} \in E_{\mathbf{z}}$ we have

$$\ell_{j+1}(\mathbf{x})\ell_{j+1}(\mathbf{x}-\mathbf{z}) = \left(\frac{q\det\Gamma_{j+1}}{\mathrm{vol}(A)}\right)^2 = \frac{\ell_j(\mathbf{z})}{m_{\Gamma_{j+1}}}. \tag{5.22}$$

Conditional independence of X, Y therefore gives

$$\alpha_{\mathrm{mom}}^2\frac{\ell_j(\mathbf{z})}{m_{\Gamma_{j+1}}}K_{\Gamma_{j+1},\mathbf{z}}(\mathbf{u}) \le \mathbb{E}[A_{\mathbf{z}} \mid \mathbf{u}] \le \frac{\ell_j(\mathbf{z})}{m_{\Gamma_{j+1}}}K_{\Gamma_{j+1},\mathbf{z}}(\mathbf{u}).$$

Averaging over $\mathbf{u}$ and applying Lemma 5.9 proves the first bound. Its upper bound uses only $\mathbb{E}_{\mathbf{u}}K_{\Gamma_{j+1},\mathbf{z}} = \mathrm{vol}(E_{\mathbf{z}})/\det\Gamma_{j+1} \le m_{\Gamma_{j+1}}$, so it does not require the variance assumption.

Two distinct output entries cannot arise from the same pair of input entries. Their generating pairs therefore share neither input entry, only an entry of X, or only an entry of Y. When the pairs share no input entry, the bounds for distinct entries in both lists and Lemma 5.9 bound the contribution by

$$(1+\xi) f^2 \ell_j(\mathbf{z}) \ell_j(\mathbf{z}'). \tag{5.23}$$

If the pairs share an entry of X with value $\mathbf{x}$, the two entries of Y have values $\mathbf{x}-\mathbf{z}$ and $\mathbf{x}-\mathbf{z}'$ and are distinct entries even when $\mathbf{z}=\mathbf{z}'$. Apply the first-moment bound to X and the bound for distinct entries to Y. Averaging over $\mathbf{u}$ using (2.17) bounds this contribution by

$$f \left(\frac{q \det \Gamma_{j+1}}{\mathrm{vol}(A)} \right)^3 \frac{\mathrm{vol}(E_{\mathbf{z}} \cap E_{\mathbf{z}'})}{\det \Gamma_{j+1}} \le \frac{f}{q} \, \ell_j(\mathbf{z}) \ell_j(\mathbf{z}').$$

The same bound holds when the pairs share an entry of Y. The choice of q gives $q^{-1} \le \xi$ for sufficiently large n. Since $f \ge 1$, the three contributions sum to at most $(1+3\xi) f^2 \ell_j(\mathbf{z}) \ell_j(\mathbf{z}')$. □

5.2 The basic SVP algorithm

We first analyze a trial that examines every pair at each merge. Later subsections implement the same searches more quickly.

The algorithm takes a basis $\mathbf{B}$ of $\mathcal{L}$ with $\det \mathcal{L} = 1$ and an estimate λ satisfying $\lambda_1(\mathcal{L}) \le \lambda \le (1+n^{-2})\lambda_1(\mathcal{L})$. Consider the lattice family satisfying the point-count bound (2.3) with $\beta \ge \frac{1}{2}\log_2(4/3)$.

Theorem 5.10 (Quadratic-search SVP)**.** *For a family of n-dimensional lattices satisfying the point-count bound (2.3) with $\beta \ge \frac{1}{2}\log_2(4/3)$, there exists a randomized algorithm solving SVP with constant probability in time $2^{(\beta+\frac{1}{2}\log_2(4/3))n+o(n)}$ and space $(4/3)^{n/2+o(n)}$. In particular, there exists a randomized algorithm solving SVP for any n-dimensional lattice in time and space*

$$\mathrm{T} = \left(\frac{8e}{3\pi}\right)^{n/2+o(n)} = 2^{0.6031\ldots n+o(n)}, \qquad \mathrm{S} = \left(\frac{4}{3}\right)^{n/2+o(n)} = 2^{0.2075\ldots n+o(n)}. \tag{5.24}$$

The filtered algorithms below allow any space parameter $\alpha \in [\frac{1}{2}\log_2(4/3), \alpha_{\max}]$, where $\alpha_{\max} = (\beta + \frac{1}{2}\log_2(4/3))/2$. As in Theorem 1.5, α is the classical space exponent.

Algorithm 11 constructs a tree of lists as illustrated in Figure 9. The subroutine Sample samples the leaf lists on Γ_D. The subroutine Merge combines child lists into parent lists on $\Gamma_{D-1}, \ldots, \Gamma_0$, and FinalMerge searches for a shortest vector. We impose a size limit of $M = 3q^2N$ entries on each list returned by Sample or Merge, where N is the number of proposals used by Sample. We repeat the construction of the two lists on Γ_0

$$k = \left\lfloor 2^{(\alpha - \frac{1}{2}\log_2(4/3))n} \right\rfloor \tag{5.25}$$

times, stopping just before FinalMerge in each repetition. Concatenate the lists from the repetitions into $L_0^{(0)}$ and $L_1^{(0)}$, which are then passed to FinalMerge. Note that we regard lists as multisets. For Theorem 5.10, it suffices to consider $k=1$ and $\alpha = \frac{1}{2}\log_2(4/3)$.

Lemma 5.11 (Success probability)**.** *Algorithm 11 outputs a shortest nonzero vector of $\mathcal{L}$ with probability at least $(1-o(1))\, 2^{-(\beta+\frac{1}{2}\log_2(4/3)-2\alpha)n}$.*

Algorithm 11 Complete SVP trial $\mathsf{SVPTrial}(\mathbf{B}, \lambda, \alpha)$

Input: a basis $\mathbf{B}$, a length estimate λ, and $\frac{1}{2}\log_2(4/3) \le \alpha \le \alpha_{\max}$.

Size limit: If a Sample or Merge call would produce more than M entries, discard the root tree containing that call. Append no entries from this tree and continue with the other trees.

Construct the chain; return FAIL if construction fails or $\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| > 2\lambda/n^{3/2}$.
Draw $\mathbf{u}$ uniformly modulo Γ_0 and set $\mathbf{u}_b^{(0)} = \mathbf{u}$ for $b = 0, 1$.
$L_b^{(0)} \leftarrow [\,]$ for $b = 0, 1$.
for $r = 1, \dots, k$ **do**
  for $\ell = 0, \dots, D-1$ **do** ▷ Generating random shifts
    Draw $\mathbf{u}_{2i}^{(\ell+1)}$ independently and uniformly modulo $\Gamma_{\ell+1}$ for each $i = 0, \dots, 2^{\ell+1}-1$.
    Set $\mathbf{u}_{2i+1}^{(\ell+1)} = \mathbf{u}_{2i}^{(\ell+1)} - \mathbf{u}_i^{(\ell)}$ for each $i = 0, \dots, 2^{\ell+1}-1$.
  $L_i^{(D)} \leftarrow \mathsf{Sample}(\mathbf{u}_i^{(D)})$ for $i = 0, \dots, 2^{D+1}-1$. ▷ Sampling leaf lists
  for $\ell = D-1, \dots, 1$ **do** ▷ Merging lists
    $L_i^{(\ell)} \leftarrow \mathsf{Merge}(\ell, \mathbf{u}_i^{(\ell)}, L_{2i}^{(\ell+1)}, L_{2i+1}^{(\ell+1)})$ for $i = 0, \dots, 2^{\ell+1}-1$.
  Append all entries of $\mathsf{Merge}(0, \mathbf{u}, L_{2b}^{(1)}, L_{2b+1}^{(1)})$ to $L_b^{(0)}$ for $b = 0, 1$.
return $\mathsf{FinalMerge}(L_0^{(0)}, L_1^{(0)})$.

Algorithm 12 Final search $\mathsf{FinalMerge}(X, Y)$

Input: two root lists X, Y on the same translate of Γ_0.

return a shortest nonzero difference $\mathbf{x} - \mathbf{y} \in \mathcal{L}$ with $\mathbf{x} \in X, \mathbf{y} \in Y$, or FAIL if none exists.

The size limit M bounds each intermediate list. Each list supplied to FinalMerge is the concatenation of at most k such lists.

Lemma 5.12 (Memory bound)**.** *With the list-size bound specified in Algorithm 11, every initial or intermediate list has at most $(4/3)^{n/2+o(n)}$ entries, and each final list $L_0^{(0)}, L_1^{(0)}$ has at most $2^{\alpha n+o(n)}$ entries.*

Proof of Theorem 5.10. Set $\alpha = \frac{1}{2}\log_2(4/3)$. Algorithm 11 constructs two trees of depth D. By Lemma 5.12, each list contains at most $(4/3)^{n/2+o(n)}$ entries. Checking all pairs in each call to Merge or FinalMerge therefore takes $(4/3)^{n+o(n)}$ time. There are $2^{O(D)} = 2^{o(n)}$ nodes, so the total time is $(4/3)^{n+o(n)}$, and storing all lists uses $(4/3)^{n/2+o(n)}$ space.

By Lemma 5.11, repeating $2^{(\beta-\frac{1}{2}\log_2(4/3))n+o(n)}$ independent trials gives constant success probability in time $2^{(\beta+\frac{1}{2}\log_2(4/3))n+o(n)}$ and the same working space. The other steps including the chain construction take only subexponential factors. □

Counting shortest differences. Fix a shortest vector $\mathbf{v} \in \mathcal{L}$. Let $A_{\mathbf{x}}, B_{\mathbf{x}}$ count occurrences of $\mathbf{x}$ in the two final lists before imposing the list-size limit M. Set

$$Z = \sum_{\mathbf{x} \in \Gamma_0 + \mathbf{u}} A_{\mathbf{x}} B_{\mathbf{x}-\mathbf{v}} \mathbf{1}_A(\mathbf{x} - \mathbf{v}/2), \tag{5.26}$$

Thus Z counts pairs $(\mathbf{x}, \mathbf{y})$ in the two lists with $\mathbf{x} - \mathbf{y} = \mathbf{v}$ and midpoint in A.

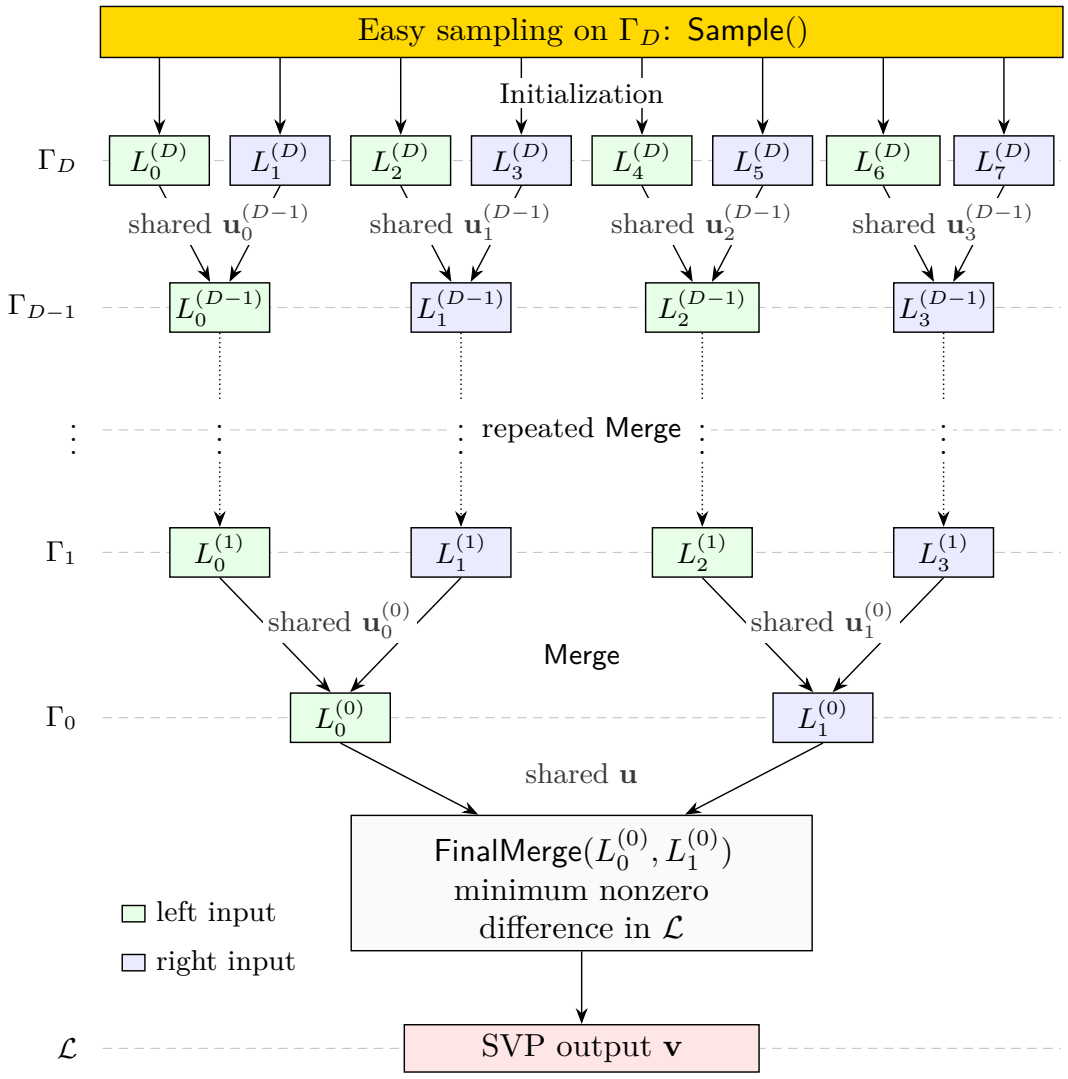


Figure 9: The two independent merge trees share a root shift $\mathbf{u}$. A node on $\Gamma_j + \mathbf{t}$ combines children on $\Gamma_{j+1} + \mathbf{u}'$ and $\Gamma_{j+1} + \mathbf{u}' - \mathbf{t}$, using a fresh split shift $\mathbf{u}'$. Dotted stretches omit intermediate levels and their other branches. Leaves use nearest-plane sampling; the final search returns a shortest nonzero difference in $\mathcal{L}$.

Lemma 5.13 (Moments for shortest-vector representations)**.** *Fix a chain satisfying Lemma 5.5 and a shortest vector $\mathbf{v} \in \mathcal{L}$. Let*

$$\nu = \frac{k^2 q^2 \det \Gamma_0}{\mathrm{vol}(A)} \in [q/8, q]\, 2^{(2\alpha - \beta - \frac{1}{2}\log_2(4/3))n}.$$

The moments over the uniform shift and the list construction satisfy

$$\mathbb{E}Z \geq (1 - o(1))\nu, \qquad \mathbb{E}Z^2 \leq (1 + o(1))(\nu^2 + 3\nu). \tag{5.27}$$

Proof. Throughout the proof, the list-size bound M is not imposed. Write $c = \frac{1}{2}\log_2(4/3)$ and $x = 2^{(\alpha - c)n} \geq 1$. Equation (5.25) gives $x/2 \leq k \leq x$, while (5.5) gives $q2^{(\beta - c)n} \leq p\,\mathrm{vol}(A) \leq 2q2^{(\beta - c)n}$.

Since $\det\Gamma_0 = 1/p$, these imply the stated interval for ν. For $1 \le r \le k$, let $A_{\mathbf{x}}^{(r)}$ and $B_{\mathbf{x}}^{(r)}$ denote the multiplicities of $\mathbf{x}$ in the two lists generated in iteration r. Then

$$A_{\mathbf{x}} = \sum_{r=1}^{k} A_{\mathbf{x}}^{(r)}, \qquad B_{\mathbf{x}} = \sum_{r=1}^{k} B_{\mathbf{x}}^{(r)}.$$

For $\mathbf{x}, \mathbf{y} \in (\Gamma_0 + \mathbf{u}) \cap H$, Lemma 5.7 gives $(1 - o(1))w \le \mathbb{E}[A_{\mathbf{x}} \mid \mathbf{u}] \le w$ for $w = \frac{kq \det\Gamma_0}{\mathrm{vol}(A)}$. Using the same lemma and independence between iterations,

$$\begin{aligned}
\mathbb{E}[A_{\mathbf{x}} A_{\mathbf{y}} \mid \mathbf{u}] &= \sum_{r=1}^{k} \mathbb{E}[A_{\mathbf{x}}^{(r)} A_{\mathbf{y}}^{(r)} \mid \mathbf{u}] + \sum_{r \ne s} \mathbb{E}[A_{\mathbf{x}}^{(r)} \mid \mathbf{u}]\, \mathbb{E}[A_{\mathbf{y}}^{(s)} \mid \mathbf{u}] \\
&\le k\left((1 + o(1))(w/k)^2 + (w/k)\mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}}\right) + k(k-1)(w/k)^2 \\
&\le (1 + o(1))w^2 + w\mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}}.
\end{aligned}$$

The term with $\mathbf{x} = \mathbf{y}$ follows from $(A_{\mathbf{x}}^{(r)})^2 = A_{\mathbf{x}}^{(r)}(A_{\mathbf{x}}^{(r)} - 1) + A_{\mathbf{x}}^{(r)}$. The same bounds hold for $B_{\mathbf{x}}$.

Let $K = K_{\Gamma_0,\mathbf{v}}(\mathbf{u})$, the number of $\mathbf{x} \in \Gamma_0 + \mathbf{u}$ satisfying $\mathbf{x}, \mathbf{x} - \mathbf{v} \in H$ and $\mathbf{x} - \mathbf{v}/2 \in A$. Lemma 5.9 gives that, for $m = \frac{\mathrm{vol}(A)}{\det\Gamma_0}$,

$$(1 - o(1))m \le \mathbb{E}_{\mathbf{u}} K \le m, \qquad \mathbb{E}_{\mathbf{u}} K^2 \le (1 + o(1))m^2.$$

In the following sums, $\mathbf{x}$ and $\mathbf{y}$ range over these K points. Independence of the two lists conditional on $\mathbf{u}$ gives

$$\mathbb{E}[Z \mid \mathbf{u}] = \sum_{\mathbf{x}} \mathbb{E}[A_{\mathbf{x}} \mid \mathbf{u}]\, \mathbb{E}[B_{\mathbf{x}-\mathbf{v}} \mid \mathbf{u}] \ge (1 - o(1))w^2 K.$$

Consequently, $\mathbb{E}Z \ge (1 - o(1))w^2 m = (1 - o(1))\nu$. This proves the first inequality.

Expanding Z^2 gives

$$\begin{aligned}
\mathbb{E}[Z^2 \mid \mathbf{u}] &= \sum_{\mathbf{x},\mathbf{y}} \mathbb{E}[A_{\mathbf{x}} A_{\mathbf{y}} \mid \mathbf{u}]\, \mathbb{E}[B_{\mathbf{x}-\mathbf{v}} B_{\mathbf{y}-\mathbf{v}} \mid \mathbf{u}] \\
&\le (1 + o(1))\left(w^4 K(K-1) + (w^2 + w)^2 K\right) = (1 + o(1))\left(w^4 K^2 + (2w^3 + w^2)K\right).
\end{aligned}$$

Here $K(K-1)$ counts the terms with $\mathbf{x} \ne \mathbf{y}$, and K counts those with $\mathbf{x} = \mathbf{y}$. Averaging over $\mathbf{u}$,

$$\mathbb{E}Z^2 \le (1 + o(1))\left(w^4 m^2 + (2w^3 + w^2)m\right) = (1 + o(1))\left(\nu^2 + (1 + 2w)\nu\right) \le (1 + o(1))(\nu^2 + 3\nu),$$

where the last inequality uses $w \le 2^{(\alpha-\beta)n} \le 1$, which follows from (5.5) and the choice of k. □

We next bound how many shortest-vector representations are lost when a list exceeds the size limit.

Lemma 5.14 (Effect of the list-size bound)**.** *Fix a chain satisfying Lemma 5.5. Let $\widehat{Z}$ count the same pairs when a root tree is discarded if any list generated in that tree exceeds M entries, as in Algorithm 11. Then $\mathbb{E}\widehat{Z} \ge (1 - o(1))\mathbb{E}Z$.*

Proof. Consider executions without the list-size bound. Define a probability measure $\Pr_*$ by weighting each execution by $Z/\mathbb{E}Z$ and then choosing uniformly one of the Z counted pairs. Under this measure, $\Pr_*[\text{the chosen pair is removed}] = \frac{\mathbb{E}[Z - \widehat{Z}]}{\mathbb{E}Z}$. It therefore suffices to show that this probability is $o(1)$.

Starting from the two entries of the pair chosen under $\Pr_*$, trace the input pairs that generated them through the merges until reaching Sample. This identifies one entry in every list of the two recursion trees.

Consider one of these lists at level j. Let $\mathbf{t}$ be its shift, so its entries lie in $(\Gamma_j + \mathbf{t}) \cap H$. Let $\mathbf{x}$ be the value of the entry identified in this list. For each $\mathbf{y} \in \Gamma_j + \mathbf{t}$, let $C_\mathbf{y}$ count the entries equal to $\mathbf{y}$, and let $N = \sum_\mathbf{y} C_\mathbf{y}$ be the list size.

Conditional on $\mathbf{t}$ and $\mathbf{x}$, the distribution of this list's construction under $\Pr_*$ has density $\frac{C_\mathbf{x}}{\mathbb{E}[C_\mathbf{x} | \mathbf{t}]}$ relative to its original distribution conditional on $\mathbf{t}$.

Consequently, Lemma 5.7 gives

$$\mathbb{E}_*[N \mid \mathbf{t}, \mathbf{x}] = 1 + \sum_{\mathbf{y} \in \Gamma_j + \mathbf{t}} \frac{\mathbb{E}[C_\mathbf{x}(C_\mathbf{y} - \mathbf{1}_{\{\mathbf{x}=\mathbf{y}\}}) \mid \mathbf{t}]}{\mathbb{E}[C_\mathbf{x} \mid \mathbf{t}]} \le 1 + (1 + o(1)) \sum_{\mathbf{y} \in \Gamma_j + \mathbf{t}} \ell_j(\mathbf{y}). \tag{5.28}$$

For sufficiently large n, Markov's inequality bounds the conditional probability that $N > M$ by

$$F_j(\mathbf{t}) := \min\left\{1, \frac{1 + 2\sum_{\mathbf{y} \in \Gamma_j + \mathbf{t}} \ell_j(\mathbf{y})}{M}\right\}.$$

Writing $\mathbb{E}_\mathbf{t}$ for expectation over a uniform shift modulo Γ_j, Lemma 2.6 and the definition of M give

$$\mathbb{E}_\mathbf{t} F_j(\mathbf{t}) \le \frac{1 + 2q \operatorname{vol}(H)/\operatorname{vol}(A)}{M} \le q^{-2}.$$

We next account for the distribution of the shifts under $\Pr_*$. The common shift $\mathbf{u}$ of the two selected output lists has density

$$\frac{\mathbb{E}[Z \mid \mathbf{u}]}{\mathbb{E}Z} \le (1 + o(1)) \frac{K_{\Gamma_0, \mathbf{v}}(\mathbf{u})}{\mathbb{E}_\mathbf{u} K_{\Gamma_0, \mathbf{v}}}$$

relative to the uniform distribution, by the first-moment bounds used in Lemma 5.13.

Similarly, consider a merge with target shift $\mathbf{t}$ and selected output value $\mathbf{z}$. If $C_\mathbf{z}$ counts its output occurrences and $\mathbf{u}$ is its first input shift, then the conditional density of $\mathbf{u}$ is

$$\frac{\mathbb{E}[C_\mathbf{z} \mid \mathbf{u}, \mathbf{t}]}{\mathbb{E}[C_\mathbf{z} \mid \mathbf{t}]} \le (1 + o(1)) \frac{K_{\Gamma_{j+1}, \mathbf{z}}(\mathbf{u})}{\mathbb{E}_\mathbf{u} K_{\Gamma_{j+1}, \mathbf{z}}},$$

by the proof of Lemma 5.8. The second input shift is $\mathbf{u} - \mathbf{t}$, so its density satisfies the same bound with a translated count function.

Thus, in each case, the relevant shift density h satisfies

$$h \le (1 + o(1)) \frac{K}{\mathbb{E}K}, \qquad \frac{\mathbb{E}K^2}{(\mathbb{E}K)^2} \le \frac{1 + \xi}{(1 - \xi)^2} = 1 + o(1),$$

where expectations are over the corresponding uniform shift. The second inequality follows from Lemma 5.9. Since $0 \le F_j \le 1$, Cauchy–Schwarz gives

$$\mathbb{E}_\mathbf{t}[h(\mathbf{t}) F_j(\mathbf{t})] \le (1 + o(1)) \frac{\mathbb{E}_\mathbf{t}[K(\mathbf{t}) F_j(\mathbf{t})]}{\mathbb{E}_\mathbf{t} K} \le (1 + o(1)) \sqrt{\frac{\mathbb{E}_\mathbf{t} K^2}{(\mathbb{E}_\mathbf{t} K)^2} \mathbb{E}_\mathbf{t} F_j^2} \le \frac{1 + o(1)}{q}.$$

These bounds are uniform in the selected parent value and target shift. Hence every list in the two selected trees exceeds M with probability at most $(1 + o(1))/q$ under $\Pr_*$.

The two trees contain fewer than 2^{D+2} lists. A union bound therefore gives

$$\frac{\mathbb{E}[Z - \widehat{Z}]}{\mathbb{E}Z} \le 2^{D+2} \frac{1 + o(1)}{q} = o(1). \qquad \square$$

Proof of Lemma 5.11. By Lemma 5.5, the chain satisfies its guarantees with probability $1 - o(1)$. Condition on these guarantees, and let $Z, \widehat{Z}$ be as in Lemma 5.14.

If $\widehat{Z} > 0$, the two lists supplied to FinalMerge contain a pair with difference $\mathbf{v}$. Thus FinalMerge returns a shortest vector. Since $0 \le \widehat{Z} \le Z$, Cauchy–Schwarz and Lemmas 5.13 and 5.14 give

$$\Pr[\widehat{Z} > 0] \ge \frac{(\mathbb{E}\widehat{Z})^2}{\mathbb{E}\widehat{Z}^2} \ge (1 - o(1))\frac{(\mathbb{E}Z)^2}{\mathbb{E}Z^2} \ge (1 - o(1))\frac{\nu}{\nu + 3}.$$

Since $\alpha \le \alpha_{\max}$, we have $2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n} \le 1$. Therefore

$$\frac{\nu}{\nu + 3} \ge \frac{q\, 2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n}}{q\, 2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n} + 24} \ge \frac{q}{q + 24}\, 2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n}$$

where we use $\nu \ge (q/8)\, 2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n}$ from the bound on ν. Multiplying by the probability that the chain satisfies its guarantees, and $q \to \infty$, gives success probability at least $(1 - o(1))\, 2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n}$, as required. □

5.3 Improved classical algorithms using spherical filtering

We now implement the pair searches in the exhaustive algorithm using the spherical filters of [BDGL16] and the analysis of [GFH26a]. This gives the time–space tradeoff in Theorem 5.1.

Let $\beta_{\mathrm{lat}} = \frac{1}{2}\log_2(2e/\pi)$, the universal centered point-count exponent of Proposition 1.1. We first analyze the filter implementation for constant exponents $\beta \le \beta_{\mathrm{lat}}$. We use the filters of Lemma 2.17, with $u = 1/4$ for Merge. For FinalMerge, choose

$$u_\alpha = 1 - 2^{-2\alpha}, \qquad \Delta_\alpha = 1 - \tfrac{4}{3}u_\alpha = \frac{4\,2^{-2\alpha} - 1}{3}, \qquad f_{\mathrm{filt}}(\alpha) = -\tfrac{1}{2}\log_2 \Delta_\alpha. \tag{5.29}$$

The following bound combines the probability of finding a shortest difference with the cost of the filters.

Lemma 5.15 (One trial)**.** *Assume $\lambda_1(\mathcal{L}) \le \lambda \le (1 + n^{-2})\lambda_1(\mathcal{L})$ and $\beta \le \beta_{\mathrm{lat}} + o(1)$. Algorithm 11 can be implemented so that one trial outputs a shortest vector with probability at least $(1 - o(1))\, 2^{-(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n}$, in expected time $2^{f_{\mathrm{filt}}(\alpha)n + o(n)}$ and space $2^{\alpha n + o(n)}$.*

Proof of Theorem 5.1. Repeat the trial in Lemma 5.15 independently $O(2^{(\beta + \frac{1}{2}\log_2(4/3) - 2\alpha)n})$ times and output the shortest vector found, choosing the constant for success probability at least $9/10$. The total expected time is $2^{t_{\beta,\mathrm{C}}(\alpha)n + o(n)}$, where

$$t_{\beta,\mathrm{C}}(\alpha) = \beta + \tfrac{1}{2}\log_2(4/3) - 2\alpha + f_{\mathrm{filt}}(\alpha) = \beta - \alpha - \tfrac{1}{2}\log_2(1 - 2^{2\alpha - 2}). \tag{5.30}$$

Limit the total work to ten times this upper bound. Markov's inequality bounds the additional failure probability by $1/10$, leaving success probability at least $4/5 > 2/3$.

Run trials and length guesses sequentially. This preserves the space bound and increases the time complexity only by subexponential factors.

For a stated $\beta > \beta_{\mathrm{lat}}$, run the same algorithm with the universal profile at β_{lat} and space parameter $\alpha' = \min\{\alpha, (\beta_{\mathrm{lat}} + c)/2\}$, where $c = \frac{1}{2}\log_2(4/3)$. Its space exponent is at most α. If $\alpha \le (\beta_{\mathrm{lat}} + c)/2$, then $t_{\beta_{\mathrm{lat}},\mathrm{C}}(\alpha) \le t_{\beta,\mathrm{C}}(\alpha)$ by (5.30). Otherwise, on the full stated interval $c \le \alpha \le (\beta + c)/2$ the derivative of $t_{\beta,\mathrm{C}}(\alpha)$ is $-1 + 2^{2\alpha-2}/(1 - 2^{2\alpha-2}) < 0$, because $(\beta + c)/2 < 1/2$. Thus $t_{\beta,\mathrm{C}}(\alpha)$ is at least its endpoint value $f_{\mathrm{filt}}((\beta + c)/2)$. This endpoint increases with β, so it is at least $f_{\mathrm{filt}}((\beta_{\mathrm{lat}} + c)/2) = t_{\beta_{\mathrm{lat}},\mathrm{C}}(\alpha')$. This proves the stated bound over the entire profile range. □

Implementation. For a unit vector $\mathbf{s}$, define

$$H_{\mathbf{s}}(u) = \{\mathbf{x} \in H : \langle \mathbf{s}, \mathbf{x} \rangle \geq \sqrt{u}\|\mathbf{x}\|\}.$$

We use filters defined by this inequality, with $u = 1/4$ for Merge and $u = u_\alpha$ from (5.29) for FinalMerge. Thus the test for Merge is $\langle \mathbf{s}, \mathbf{x} \rangle \geq \|\mathbf{x}\|/2$.

For input lists X, Y, let X_f, Y_f be the entries contained in filter f, and let

$$I = \sum_f (|X_f| + |Y_f|).$$

Let J count pairs from $X_f \times Y_f$, summed over all filters, satisfying $\mathbf{x}-\mathbf{y}-\mathbf{t} \in \Gamma_j$ for $\mathsf{Merge}(j, \mathbf{t}, X, Y)$, or $\mathbf{x} - \mathbf{y} \in \mathcal{L}$ for FinalMerge. A pair contained in several filters is counted once for each such filter.

Lemma 5.16 (Filter-based implementation)**.** *Assume $\lambda_1(\mathcal{L}) \leq \lambda \leq (1 + n^{-2})\lambda_1(\mathcal{L})$ and $\beta \leq \beta_{\mathrm{lat}} + o(1)$. Each merging call can be implemented in time $2^{o(n)}(|X| + |Y| + I + J + 1)$ and auxiliary space $O(|X| + |Y| + M) + 2^{o(n)}$. The resulting trial satisfies the success-probability bound in Lemma 5.11.*

Proof. In Merge, group the entries of X by $\mathbf{x}-\mathbf{u}+\Gamma_j$ and those of Y by $\mathbf{y}-\mathbf{u}+\mathbf{t}+\Gamma_j$. These cosets agree exactly when $\mathbf{x}-\mathbf{y}-\mathbf{t} \in \Gamma_j$. In FinalMerge, group each entry $\mathbf{z}$ by $\mathbf{z}-\mathbf{u}+\mathcal{L}$. Lemma 2.18 finds the pairs in corresponding groups that share a filter. For Merge, check $\mathbf{x}-\mathbf{y} \in H$ and $(\mathbf{x}+\mathbf{y})/2 \in A$, and use each accepted pair of list entries at most once. Store the accepted occurrence pairs to prevent repeated outputs. If an additional pair would exceed the bound M, discard the affected root tree as in Algorithm 11. For FinalMerge, output the shortest nonzero difference found. The time and space bounds follow from Lemma 2.18 and the additional storage for at most M pairs.

We next bound the probability of missing a required pair. For $\beta \leq \beta_{\mathrm{lat}} + o(1)$, the upper endpoint $\alpha_{\max} = (\beta + c)/2$ gives

$$\tfrac{1}{4} \leq u_\alpha \leq 1 - 2^{-(\beta_{\mathrm{lat}}+c)} + o(1) < 0.342, \qquad \Delta_\alpha \geq \frac{4\,2^{-(\beta_{\mathrm{lat}}+c)} - 1}{3} - o(1) > 0.51 \tag{5.31}$$

for sufficiently large n. The constant c here is $\frac{1}{2}\log_2(4/3)$. Thus both filter settings satisfy Lemma 2.17 with $\delta = 0.01$. Let $(\mathbf{x}, \mathbf{y})$ be a pair accepted by Merge, or a pair whose difference is a shortest vector in FinalMerge. The three lengths $\|\mathbf{x}\|, \|\mathbf{y}\|, \|\mathbf{x} - \mathbf{y}\|$ are $(1 + O(n^{-1/3}))\lambda$, so

$$\left\langle \frac{\mathbf{x}}{\|\mathbf{x}\|}, \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle = \frac{\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2}{2\|\mathbf{x}\|\|\mathbf{y}\|} = \tfrac{1}{2} + O(n^{-1/3}). \tag{5.32}$$

Thus we apply Lemma 2.17 with $\varepsilon = \lceil \log_2(n+2) \rceil^{-2}$ and $2n$ independent repetitions. Each required pair then shares a filter except with probability at most $2^{-2n^2} \leq e^{-n^2}$.

Recall ν from Lemma 5.13 where $(1 - o(1))\nu/(\nu + 3)$ is the success-probability lower bound used in the proof of Lemma 5.11. Fix a chain satisfying Lemma 5.5, and let $\widehat{Z}$ be as in Lemma 5.14. A union bound over at most $2^{O(n)}$ pairs, followed by Cauchy–Schwarz and Lemmas 5.13 and 5.14, gives

$$\begin{aligned}\Pr[\text{a shortest vector is found}] &\geq \Pr[\widehat{Z} > 0] - e^{-n^2+O(n)} \\ &\geq \frac{(\mathbb{E}\widehat{Z})^2}{\mathbb{E}Z^2} - e^{-n^2+O(n)} \geq (1 - o(1))\frac{\nu}{\nu + 3} - e^{-n^2+O(n)}.\end{aligned}$$

The bound on ν, together with $\alpha \leq \alpha_{\max}$ and $q \to \infty$, gives the claimed success probability. Lemma 5.5 contributes only a further factor $1 - o(1)$. □

Overlaps of translated caps. To bound I and J, we estimate the overlap of translated caps. To count the pairs with difference $\mathbf{v} \in \mathcal{L}$, we bound the volume of $H_{\mathbf{s}}(u) \cap (H_{\mathbf{s}}(u) + \mathbf{v})$.

Lemma 5.17 (Cap-overlap sum)**.** *Assume $\lambda_1(\mathcal{L}) \le \lambda \le (1 + n^{-2})\lambda_1(\mathcal{L})$ and (2.3), with $b = 2^{\beta}$. Uniformly over unit vectors $\mathbf{s}$ and $3/16 \le u \le 0.4$,*

$$\frac{1}{\mathrm{vol}(A)} \sum_{\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\}} \mathrm{vol}(H_{\mathbf{s}}(u) \cap (H_{\mathbf{s}}(u) + \mathbf{v})) \le \left(\frac{4b^2(1-u)^2}{3} \right)^{n/2} 2^{o(n)}.$$

Proof. Let $C = H_{\mathbf{s}}(u)$ and $R^2 = \lambda^2(1 - u + 3n^{-1/3})$. For $\mathbf{x} \in C$,

$$\|\mathbf{x} - \sqrt{u}\,\lambda \mathbf{s}\|^2 \le \|\mathbf{x}\|^2 - 2u\lambda\|\mathbf{x}\| + u\lambda^2 \le R^2.$$

The parallelogram identity therefore gives

$$\mathrm{vol}(C \cap (C + \mathbf{v})) \le V_n R^n \left(1 - \frac{\|\mathbf{v}\|^2}{4R^2}\right)_+^{n/2}.$$

By Equation (5.8),

$$\frac{1}{\mathrm{vol}(A)} \sum_{\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\}} \mathrm{vol}(C \cap (C + \mathbf{v})) \le \frac{O(\sqrt{n}) V_n R^n}{\mathrm{vol}(A)} \left(\frac{bR}{\lambda_1(\mathcal{L})} \right)^n = \left(\frac{4b^2(1-u)^2}{3} \right)^{n/2} 2^{o(n)}.$$

Here we used $\mathrm{vol}(A) = V_n(\sqrt{3}\lambda/2)^n 2^{o(n)}$, $R^2 = \lambda^2(1 - u + o(1))$, and $\lambda/\lambda_1(\mathcal{L}) = 1 + O(n^{-2})$. □

We can now bound the expected number of stored filter entries and examined pairs.

Lemma 5.18 (Expected counts)**.** *Assume $\lambda_1(\mathcal{L}) \le \lambda \le (1 + n^{-2})\lambda_1(\mathcal{L})$. Fix a chain on which Algorithm 11 generates lists, and choose the filters independently of the input lists.*

For Merge*, use $3/16 \le u \le 0.342$ and $F(u) = (1 - 4u/3)^{-n/2} 2^{o(n)}$ filters. Then*

$$\mathbb{E}I \le F(u) \left(\frac{4(1-u)}{3} \right)^{n/2} 2^{o(n)}, \qquad \mathbb{E}J \le F(u) \left(\frac{16(1-u)^2}{9} \right)^{n/2} 2^{o(n)}.$$

In particular, at $u = 1/4$, we have $\mathbb{E}I, \mathbb{E}J \le (3/2)^{n/2 + o(n)}$.

For FinalMerge*, using $u = u_\alpha$ and $F(u_\alpha) = 2^{f_{\mathrm{filt}}(\alpha)n + o(n)}$ filters gives $\mathbb{E}I, \mathbb{E}J \le 2^{f_{\mathrm{filt}}(\alpha)n + o(n)}$. Expectations are over the shifts, list generation, and filters.*

Proof. Fix the filters for one execution of Merge or FinalMerge. The expectations below are over the shifts and list generation. We analyze the input lists generated without filtering or size bounds, since deleting entries can only decrease I, J.

For Merge, let C be the intersection of one filter with H, and let I_C, J_C be the counts restricted to this filter. Lemma 5.7 bounds each input multiplicity by $q \det \Gamma_{j+1} / \mathrm{vol}(A)$ in expectation. Lemma 2.6 gives

$$\mathbb{E}I_C \le \frac{2q\,\mathrm{vol}(C)}{\mathrm{vol}(A)}, \qquad \mathbb{E}J_C \le \left(\frac{q \det \Gamma_{j+1}}{\mathrm{vol}(A)} \right)^2 \frac{\mathrm{vol}(C)^2}{\det \Gamma_j \, \det \Gamma_{j+1}} = q \left(\frac{\mathrm{vol}(C)}{\mathrm{vol}(A)} \right)^2,$$

using $\det \Gamma_j = q \det \Gamma_{j+1}$. Projecting and normalizing the filter direction gives $C \subseteq H_{\mathbf{s}}(u)$ for a unit vector $\mathbf{s} \in \mathbb{R}^n$. Therefore,

$$\frac{\mathrm{vol}(C)}{\mathrm{vol}(H)} \le (1-u)^{n/2} 2^{o(n)}, \qquad \frac{q\,\mathrm{vol}(H)}{\mathrm{vol}(A)} = (4/3)^{n/2 + o(n)}.$$

Summing over the $F(u)$ filters proves the bounds for Merge.

For FinalMerge, each input list combines k independently generated lists and has expected size at most $kq\,\mathrm{vol}(H)/\,\mathrm{vol}(A) = 2^{\alpha n+o(n)}$. The same volume estimate gives

$$\mathbb{E}I \le F(u_\alpha)\,2^{\alpha n}(1-u_\alpha)^{n/2}2^{o(n)} = F(u_\alpha)2^{o(n)},$$

since $1-u_\alpha = 2^{-2\alpha}$.

For J, group pairs by their difference and average over the common shift. Lemma 2.6 gives

$$\mathbb{E}J \le \frac{\nu}{\mathrm{vol}(A)} \sum_f \sum_{\mathbf{v}\in\mathcal{L}} \mathrm{vol}(C_f \cap (C_f+\mathbf{v})), \tag{5.33}$$

where C_f is the intersection of filter f with H. The coefficient follows from $\frac{1}{\det\Gamma_0}\left(\frac{kq\det\Gamma_0}{\mathrm{vol}(A)}\right)^2 = \frac{\nu}{\mathrm{vol}(A)}$. Recall that $\det\Gamma_0 = 1/p$; the bound on p in (5.5) and the definition of k imply

$$\nu = \frac{k^2q^2}{p\,\mathrm{vol}(A)} \le k^2q\left(\frac{4}{3b^2}\right)^{n/2} = 2^{(2\alpha-\beta-\frac{1}{2}\log_2(4/3))n+o(n)}.$$

Lemma 5.17 bounds the contribution of $\mathbf{v}\neq\mathbf{0}$ by

$$F(u_\alpha)\nu\left(\frac{4b^2(1-u_\alpha)^2}{3}\right)^{n/2} 2^{o(n)} \le F(u_\alpha)\,2^{2\alpha n}(1-u_\alpha)^n 2^{o(n)} = F(u_\alpha)2^{o(n)}.$$

For $\mathbf{v}=\mathbf{0}$, the intersection is C_f, so its contribution is at most

$$F(u_\alpha)\nu\frac{\mathrm{vol}(H_{\mathbf{s}}(u_\alpha))}{\mathrm{vol}(A)} \le F(u_\alpha)2^{(\alpha-\beta)n+o(n)} \le F(u_\alpha)2^{o(n)},$$

using $\alpha \le \alpha_{\max} \le \beta$. This proves the bound on $\mathbb{E}J$. All bounds are uniform in the filters, so they also hold after averaging over their choice. □

Proof of Lemma 5.15. The success probability follows from Lemma 5.16.

Lemmas 5.16 and 5.18 give expected time $(3/2)^{n/2+o(n)}$ for each Merge and $2^{f_{\mathrm{filt}}(\alpha)n+o(n)}$ for FinalMerge. Each Sample takes time $(4/3)^{n/2+o(n)}$ by (5.10). There are $k2^{O(D)}$ sampling and merging calls, so list generation takes expected time at most

$$k(3/2)^{n/2+o(n)} \le 2^{f_{\mathrm{filt}}(\alpha)n+o(n)}.$$

This follows from the definition of k, the identity $f_{\mathrm{filt}}(\frac{1}{2}\log_2(4/3)) = \frac{1}{2}\log_2(3/2)$, and $f'_{\mathrm{filt}}(\alpha) > 1$. Preparation takes subexponential time.

Generate the k lists on each side sequentially. Lemma 5.12 bounds the stored lists by $O(kM + 2^DM) = 2^{\alpha n+o(n)}$ entries. The additional space in Lemma 5.16 satisfies the same bound. □

5.4 Quantum search on the generated lists

Theorem 5.19 (Quantum tradeoff)**.** *Under the same point-count assumption as Theorem 5.1, for every $\frac{1}{2}\log_2(4/3) \le \alpha \le \alpha_{\max}$, there is a quantum algorithm for exact SVP with coherent indexed access to stored classical arrays and success probability at least $2/3$. Its time and space bounds are*

$$\mathrm{T_Q} = 2^{(\frac{1}{2}\log_2(13/9)+\beta-\alpha)n+o(n)}, \qquad \mathrm{S_Q} = 2^{\max\{\alpha,\frac{1}{2}\log_2(13/9)\}n+o(n)}. \tag{5.34}$$

For Merge, use the filters of Lemma 2.17 with $u = 3/16$ and quantum enumeration to find the accepted pairs. For FinalMerge, use quantum minimum finding over all input pairs.

Lemma 5.20 (One trial)**.** *Assume $\lambda_1(\mathcal{L}) \le \lambda \le (1+n^{-2})\lambda_1(\mathcal{L})$. Algorithm 11 has a quantum implementation that outputs a shortest vector with probability at least*

$$(1-o(1))\,2^{-(\beta+\frac{1}{2}\log_2(4/3)-2\alpha)n},$$

in time $2^{(\alpha+\frac{1}{2}\log_2(13/12))n+o(n)}$ and space $2^{\max\{\alpha,\frac{1}{2}\log_2(13/9)\}n+o(n)}$.

Proof of Theorem 5.19. Run $\lceil 4\,2^{(\beta+\frac{1}{2}\log_2(4/3)-2\alpha)n}\rceil$ independent trials from Lemma 5.20 and output the shortest vector found. For sufficiently large n, the probability that all trials fail is at most $e^{-2} < 1/3$. The total time exponent is

$$\alpha + \tfrac{1}{2}\log_2(13/12) + \beta + \tfrac{1}{2}\log_2(4/3) - 2\alpha = \tfrac{1}{2}\log_2(13/9) + \beta - \alpha.$$

Run trials and length guesses sequentially to preserve the space bound. Preparation and the number of guesses contribute only subexponential factors. □

Quantum enumeration. For one execution of Merge, let I, J be the counts in Lemma 5.18. Let R count those pairs in J that also satisfy $\mathbf{x}-\mathbf{y} \in H$ and $(\mathbf{x}+\mathbf{y})/2 \in A$.

Lemma 5.21 (Expected work)**.** *Assume $\lambda_1(\mathcal{L}) \le \lambda \le (1+n^{-2})\lambda_1(\mathcal{L})$, and fix a chain on which Algorithm 11 generates lists. With $u = 3/16$, quantum enumeration of the accepted pairs contained in a common filter takes expected time $(13/9)^{n/2+o(n)}$.*

Proof. Group the input entries by filter and by the cosets used in Lemma 5.16. Store the corresponding groups, their Cartesian-product sizes, and the cumulative sums of these sizes. This allows coherent access to each of the J pairs in polynomial time.

Quantum enumeration [vAGN24], with failure probability at most e^{-n^2}, finds the accepted pairs in expected time $2^{o(n)}\left(|X|+|Y|+I+\sqrt{J(R+1)}+1\right)$. Lemma 5.18 with $u = 3/16$ gives

$$\mathbb{E}I \le (13/9)^{n/2+o(n)}, \qquad \mathbb{E}J \le (169/108)^{n/2+o(n)}. \tag{5.35}$$

By (5.32) and (2.35), each accepted pair belongs on average to at most $2^{o(n)}$ common filters. Averaging the output multiplicities from Lemma 5.7 over the target shift therefore gives

$$\mathbb{E}R \le 2^{o(n)}\frac{q\,\mathrm{vol}(H)}{\mathrm{vol}(A)} = (4/3)^{n/2+o(n)}.$$

Cauchy–Schwarz now gives

$$\mathbb{E}\sqrt{J(R+1)} \le \sqrt{\mathbb{E}J\,\mathbb{E}(R+1)} \le (13/9)^{n/2+o(n)}.$$

Together with $\mathbb{E}(|X|+|Y|) \le (4/3)^{n/2+o(n)}$, this proves the expected-time bound. □

To bound the running time, we modify the generation of each of the k lists on either side of FinalMerge. If any Merge takes more than q^2T steps, we stop generating that list and use an empty list instead. Here $T = (13/9)^{n/2+o(n)}$ is chosen in Lemma 5.22. Each list requires $2^{O(D)}$ sampling and merging procedures, so its generation uses time and space at most $2^{O(D)}q^2T = (13/9)^{n/2+o(n)}$.

Lemma 5.22 (Expected count)**.** *Using the same shifts and samples, let Z_{Q} count the pairs in $\widehat{Z}$ from Lemma 5.14 that are also generated by quantum enumeration with the stopping rule above. For a suitable choice of $T = (13/9)^{n/2+o(n)}$, whenever the chain satisfies Lemma 5.5, we have $0 \le Z_{\mathrm{Q}} \le \widehat{Z} \le Z$ and $\mathbb{E}Z_{\mathrm{Q}} \ge (1-o(1))\mathbb{E}Z$.*

Proof. Use the probability measure $\Pr_*$ from the proof of Lemma 5.14: weight each execution by $Z/\mathbb{E}Z$ and choose one of its counted pairs uniformly. Then $\Pr_*[\text{the chosen pair is removed}] = \frac{\mathbb{E}[Z-Z_{\mathrm{Q}}]}{\mathbb{E}Z}$. We bound this probability by examining how the two entries of the chosen pair were generated.

Starting from each of these entries, follow the two input entries that produced it at a merge, and continue until reaching Sample. Consider a merge at level j. Write $\Gamma_j + \mathbf{t}$ for the translate containing its output list. As in (5.11), its input lists lie in $X \subseteq \Gamma_{j+1} + \mathbf{u}, Y \subseteq \Gamma_{j+1} + \mathbf{u} - \mathbf{t}$, where $\mathbf{u}$ is the shift used to generate the first input list. Let $\mathbf{x} \in X$ and $\mathbf{y} \in Y$ be the input pair that produced the output entry being followed.

Under $\Pr_*$, conditional on $\mathbf{t}, \mathbf{u}, \mathbf{x}, \mathbf{y}$, the constructions of X and Y are independent, each weighted by the multiplicity of its selected value, as in the proof of Lemma 5.14. The moment bounds used in (5.28) show that, after removing the selected entry from each list, every remaining expected multiplicity is at most $(1+o(1))\ell_{j+1}$. Hence the same first-moment estimates as in Lemma 5.21 apply to these remaining entries and to pairs formed from them.

We next bound the contributions involving $\mathbf{x}$ or $\mathbf{y}$. By (2.35), the expected number of filters containing $\mathbf{x}$ is at most

$$F(3/16)(13/16)^{n/2}2^{o(n)} = (13/12)^{n/2+o(n)}.$$

Averaging the multiplicity bound over uniform shifts bounds the expected number of other entries in Y by $(4/3)^{n/2+o(n)}$. Their pairs with $\mathbf{x}$ therefore contribute at most

$$(13/12)^{n/2+o(n)}(4/3)^{n/2+o(n)} = (13/9)^{n/2+o(n)}$$

to $\mathbb{E}J$. Their contribution to $\mathbb{E}R$ is at most $(4/3)^{n/2+o(n)}$, since each accepted pair belongs on average to at most $2^{o(n)}$ common filters. The same bounds hold for pairs involving $\mathbf{y}$ and the other entries of X. The pair $(\mathbf{x}, \mathbf{y})$ is itself accepted, so its contribution to $\mathbb{E}J$ and $\mathbb{E}R$ is at most $2^{o(n)}$.

For fixed shifts, the multiplicity bounds and filter estimates above are uniform in the selected input values $\mathbf{x}, \mathbf{y}$. They therefore give an upper bound $W_j(\mathbf{t}, \mathbf{u})$ on the conditional expected work. Applying Cauchy–Schwarz as in Lemma 5.21 gives

$$\mathbb{E}_{\mathbf{t},\mathbf{u}} W_j(\mathbf{t}, \mathbf{u}) \le T = (13/9)^{n/2+o(n)},$$

where the two shifts are independently uniform modulo Γ_j and Γ_{j+1}. Choose T to satisfy this bound for every j.

Let $h(\mathbf{t}, \mathbf{u})$ be the joint density of these shifts under $\Pr_*$ relative to the product of the two uniform measures. Write $h(\mathbf{t}, \mathbf{u}) = h_0(\mathbf{t})h_1(\mathbf{u} \mid \mathbf{t})$ for its marginal and conditional densities. The proof of Lemma 5.14 gives $\mathbb{E}_{\mathbf{t}} h_0^2 \le 1 + o(1)$. It also bounds the second moment of the input-shift density conditional on the target shift and the selected output value, uniformly in both. Averaging over that output value and applying Jensen's inequality gives $\mathbb{E}_{\mathbf{u}} h_1(\mathbf{u} \mid \mathbf{t})^2 \le 1 + o(1)$. Therefore,

$$\mathbb{E}_{\mathbf{t},\mathbf{u}} h^2 = \mathbb{E}_{\mathbf{t}}\Big[h_0(\mathbf{t})^2 \mathbb{E}_{\mathbf{u}} h_1(\mathbf{u} \mid \mathbf{t})^2\Big] \le 1 + o(1).$$

Markov's inequality and the Cauchy–Schwarz inequality imply

$$\begin{aligned}\Pr_*[\text{this merge requires more than } q^2T \text{ steps}] &\le \mathbb{E}_{\mathbf{t},\mathbf{u}}\left[h \min\left\{1, \frac{W_j}{q^2T}\right\}\right] \\ &\le \left((1+o(1))\frac{\mathbb{E}_{\mathbf{t},\mathbf{u}} W_j}{q^2T}\right)^{1/2} \le \frac{1+o(1)}{q}.\end{aligned}$$

The stopping rule can remove the pair chosen under $\Pr_*$ only by discarding one of the two lists that produce its entries. Generating those two lists uses $2^{O(D)}$ merges. A union bound therefore bounds the removal probability from this rule by $2^{O(D)}/q = o(1)$.

It remains to bound failures of the filters and quantum enumeration. At each merge, apply Lemma 2.17 with $\varepsilon = \lceil \log_2(n+2) \rceil^{-2}$ and $2n$ independent repetitions, using filters independent of the shifts and samples. By (5.32), each fixed accepted pair shares a filter except with probability at most e^{-n^2}. If every list used to generate the two chosen entries has at most M entries, these merges involve at most $2^{O(D)}M^2 = 2^{O(n)}$ input pairs. A union bound over these pairs and the quantum enumeration errors gives an additional removal probability of at most $e^{-n^2+O(n)}$.

Lemma 5.14 bounds the removal probability from the list-size condition by $o(1)$. The total removal probability under $\Pr_*$ is therefore $o(1)$, and hence $\mathbb{E}Z_{\mathrm{Q}} \geq (1-o(1))\mathbb{E}Z$. □

Proof of Lemma 5.20. Generate the k lists using Lemma 5.22. The total time and space are $k(13/9)^{n/2+o(n)} = 2^{(\alpha+\frac{1}{2}\log_2(13/12))n+o(n)}$ and $O(kM) + (13/9)^{n/2+o(n)} = 2^{\max\{\alpha, \frac{1}{2}\log_2(13/9)\}n+o(n)}$.

For FinalMerge, assign each input pair its squared difference norm if the difference is nonzero and belongs to $\mathcal{L}$, and assign $+\infty$ otherwise. Dürr–Høyer minimum finding [DH96], with failure probability at most e^{-n^2}, finds a minimum in time $\sqrt{|X||Y|}\,2^{o(n)} \leq 2^{\alpha n+o(n)}$. Output its difference if the minimum is finite.

Condition on the guarantees of Lemma 5.5. Cauchy–Schwarz and Lemmas 5.13 and 5.22 give

$$\Pr[Z_{\mathrm{Q}} > 0] \geq \frac{(\mathbb{E}Z_{\mathrm{Q}})^2}{\mathbb{E}Z^2} \geq (1-o(1))\frac{\nu}{\nu+3} \geq (1-o(1))\frac{q}{q+24}\,2^{-(\beta+\frac{1}{2}\log_2(4/3)-2\alpha)n}$$

where we use the inequality used for the success probability in Lemma 5.11. If $Z_{\mathrm{Q}} > 0$, the final lists contain a shortest-vector difference. The chain guarantees hold with probability $1-o(1)$, $q \to \infty$, and minimum finding fails with probability at most e^{-n^2}. This proves the success probability. □

5.5 SVP specializations

Classical worst-case endpoints. Apply Theorem 5.1 with the universal exponent $\beta = 0.3955\ldots + o(1)$ from Proposition 1.1. On $[\frac{1}{2}\log_2(4/3), \alpha_{\max}]$, where $\alpha_{\max} < 1/2$,

$$t'_{\beta,\mathrm{C}}(\alpha) = -1 + \frac{2^{2\alpha-2}}{1-2^{2\alpha-2}} < 0. \tag{5.36}$$

Thus $\alpha = \alpha_{\max}$ minimizes time, giving time exponent $0.4385783636\ldots + o(1)$ and space exponent $0.3015591026\ldots + o(1)$. At the minimum-space endpoint $\alpha = \frac{1}{2}\log_2(4/3)$,

$$t_{\beta,\mathrm{C}}(\tfrac{1}{2}\log_2(4/3)) = \beta + \tfrac{1}{2}\log_2(9/8) = 0.4805619564\ldots + o(1), \tag{5.37}$$

$$s(\tfrac{1}{2}\log_2(4/3)) = \tfrac{1}{2}\log_2(4/3) = 0.2075187496\ldots. \tag{5.38}$$

Proof of Corollary 1.8. Lemma 2.1 converts the assumed bound on the general angle-$\pi/3$ spherical kissing number into the centered point-count profile. After absorbing its subexponential prefactor, use $\beta_{\mathrm{eff}} = \frac{1}{2}\log_2(4/3) + o(1)$ and $\alpha = \frac{1}{2}\log_2(4/3) + o(1)$ in Theorems 5.1 and 5.19. The classical time and space exponents are $\frac{1}{2}\log_2(3/2) + o(1)$ and $\frac{1}{2}\log_2(4/3) + o(1)$, respectively. Both quantum exponents are $\frac{1}{2}\log_2(13/9) + o(1)$. □

Haar-typical input classes. Let μ_n be the invariant probability measure on the space of unimodular lattices $\mathrm{SL}_n(\mathbb{R})/\mathrm{SL}_n(\mathbb{Z})$.

Lemma 5.23 (Uniform point-count bound for Haar-random lattices, [ZY26, Theorem 1.2])**.** *There is an absolute constant $C_{\mathrm{H}} > 0$ such that, for every $n \geq 3$ and every $u > 0$,*

$$\Pr_{\mathcal{L}\sim\mu_n}\left[\sup_{r\geq 1}\frac{N_{\mathcal{L}/\lambda_1(\mathcal{L})}(r)}{r^n} > u\right] \leq \min\{1, C_{\mathrm{H}}/u\}. \tag{5.39}$$

Taking $u = n^2$ gives, with probability $1 - O(n^{-2})$, $N_{\mathcal{L}/\lambda_1(\mathcal{L})}(r) \leq n^2 r^n$ for all $r \geq 1$. Let $\mathcal{H}_n$ be this class of lattices. Equation (2.3) then holds on $\mathcal{H}_n$ with $\beta = 0$, $b = 1$, and $A_n = n^2$.

Proof of Corollary 1.9. For lattices in $\mathcal{H}_n$, use the effective exponent $\beta_{\mathrm{eff}} = \frac{1}{2}\log_2(4/3) + o(1)$ in Theorems 5.1 and 5.19, as in the preceding specialization. This gives

$$\mathrm{T_C} = (3/2)^{n/2+o(n)}, \qquad \mathrm{S_C} = (4/3)^{n/2+o(n)}, \tag{5.40}$$

$$\mathrm{T_Q} = (13/9)^{n/2+o(n)}, \qquad \mathrm{S_Q} = (13/9)^{n/2+o(n)}. \tag{5.41}$$

□

Quantum worst-case endpoints. The time exponent in Theorem 5.19 decreases with α. Using the universal point-count exponent $\beta = \frac{1}{2}\log_2(2e/\pi)$ from Proposition 1.1, we have $\alpha_{\max} = \frac{1}{4}\log_2(8e/(3\pi)) > \frac{1}{2}\log_2(13/9)$. Choosing $\alpha = \alpha_{\max}$ gives the quantum bounds in Theorem 5.1. For minimum quantum space, choose $\alpha = \frac{1}{2}\log_2(13/9)$, which lies in $[\frac{1}{2}\log_2(4/3), \alpha_{\max}]$. Up to this value, the space exponent is constant and the time exponent decreases. This choice gives time exponent $\beta = 0.3955\ldots + o(1)$ and space exponent $\frac{1}{2}\log_2(13/9) = 0.2652\ldots$.

6 Classical SVP in Time $2^{0.4088...n+o(n)}$ – Smaller Shells

Theorem 6.1 (Smaller fixed-radius SVP endpoints)**.** *Exact SVP on every input lattice can be solved with success probability at least* $2/3$ *in classical time* $2^{\tau_C n+o(n)}$ *and space* $2^{s_C n+o(n)}$*, or quantum time* $2^{\tau_Q n+o(n)}$ *and space* $2^{s_Q n+o(n)}$ *with coherent indexed access to stored classical arrays, where*

$$\tau_C = \tfrac{1}{2}\log_2 \frac{2\pi}{2\pi - e} = 0.408816\ldots, \qquad s_C = \tfrac{1}{2}\log_2\left(1 + \frac{e}{2\pi}\right) = 0.259332\ldots,$$
$$\tau_Q = \tfrac{1}{2}\log_2\left[\frac{13}{12}\left(1 + \frac{e}{2\pi}\right)\right] = 0.317070\ldots, \qquad s_Q = \tfrac{1}{2}\log_2 \frac{13}{9} = 0.265257\ldots. \tag{6.1}$$

Both algorithms use one common shell radius throughout the chain.

Theorem 5.1 uses lists whose vectors have approximately the shortest-vector length. Shorter list vectors can still produce a shortest difference. Using a smaller common shell saves time and space, provided that the two final lists are large enough to contain such a pair.

1. **Use different midpoint shells for different tasks.** An internal merge produces another vector of the list radius. The final search needs a longer difference, so its useful midpoints lie in a smaller shell. This second shell enters the analysis of the final search. The search itself only checks the difference.
2. **Choose the radius by balancing two count estimates.** The first superlattice must make shifted midpoint counts concentrate for both shells. Shrinking the common list radius first reduces the list size. Beyond a certain point, the small final midpoint shell forces a denser superlattice, and producing a shortest difference requires more entries again. The two requirements balance at a list radius of about 0.91 times the shortest-vector length.
3. **Set the filters for the final search.** A shortest difference between vectors on a smaller shell comes from a wider angle than an internal difference. Filters chosen for the wider angle also find the internal pairs. The point-count bound controls how many pairs the final search examines.
4. **Balance list size against the number of trials.** With smaller lists, a trial succeeds less often. We bound the fraction of shortest-vector representations lost when a trial reaches its list or time limit. This gives the classical time–space tradeoff. The quantum implementation can form many small lists and join them only at the final search, so the larger lists need not be used at every level.

Roadmap. Section 6.1 defines the two midpoint shells. Section 6.2 proves the list moment estimates. Section 6.3 analyzes the spherical filters. Section 6.4 derives the classical time–space tradeoff. Section 6.5 justifies the common shell radius. Section 6.6 gives the quantum implementation and its time–space tradeoff.

6.1 The modified fixed-radius construction

Normalize the input to determinant one. Let $c = \frac{1}{2}\log_2(4/3)$ and $\beta_{\mathrm{lat}} = \frac{1}{2}\log_2(2e/\pi)$. We first treat point-count exponents $c \le \beta \le \beta_{\mathrm{lat}}$. Absorb the subexponential prefactor in (2.3) into $b = b_n \ge 2^\beta$, so that $b = 2^{\beta+o(1)}$. The list and midpoint radii below depend on b and a guess $\widehat{\lambda}$ for the shortest-vector length.

We use shell width parameter $\varepsilon = h^{-2}$, where $h = \lceil \log_2(n+2) \rceil$, and take length guesses of relative accuracy $\varepsilon/16$ as in Lemma 2.8. Choose a rational $a = 1 + b^2/4 + O(\varepsilon) > 1 + b^2/4$. Let

r be the midpoint radius for a shortest difference, R the common list radius, and r_{i} the midpoint radius of an internal pair. The upper list-size exponent m_a is determined by a.

$$r^2 = \frac{\widehat{\lambda}^2}{4(a-1)}, \qquad R^2 = ar^2, \qquad r_{\mathrm{i}}^2 = \frac{3}{4}R^2, \qquad m_a = \tfrac{1}{2}\log_2 a. \tag{6.2}$$

For sufficiently large n, $4/3 < a < 3/2$. Every list uses radius R; the midpoint identities are

$$r_{\mathrm{i}}^2 + R^2/4 = R^2, \qquad r^2 + \widehat{\lambda}^2/4 = R^2. \tag{6.3}$$

Thus $r_{\mathrm{i}} \geq r$, and $R/\widehat{\lambda} = \sqrt{1/4 + 2^{-2\beta}} + o(1)$. At the lower endpoint $\beta = c$ the limiting radii recover Section 5; otherwise the final difference is longer than the list vectors.

Define the list shell, the two midpoint shells, and their volume scale by

$$H = \{\mathbf{x} : |\|\mathbf{x}\|^2/r^2 - a| \leq \varepsilon\},$$
$$A_t = \{\mathbf{w} : (1-n^{-2})t^2 \leq \|\mathbf{w}\|^2 \leq t^2\} \quad (t \in \{r, r_{\mathrm{i}}\}), \qquad \mathcal{V}(t) = \left(\frac{2\pi e t^2}{n}\right)^{n/2}, \qquad \mathcal{V} = \mathcal{V}(r). \tag{6.4}$$

As in Section 5, take a random prime-index extension and then halve basis vectors cyclically. Use $D = h^3$ halvings of batches of $\lceil n/h \rceil$ vectors, so each step has index $q = 2^{\lceil n/h \rceil}$. The first index and the scale used in the terminal rounding test are

$$p \in [q^{10}/\mathcal{V},\, 2q^{10}/\mathcal{V}], \qquad \sigma^2 = 2r^2/n. \tag{6.5}$$

Here p is prime. The terminal test is $\max_i \|\widetilde{\mathbf{b}}_i^{(D)}\| \leq \sigma/n^2$.

The parameter $\alpha \in [c, m_a]$ controls the list size. Define a weight coefficient C_α, the target expected count $\ell_{d,\alpha}(\mathbf{x})$ of a vector, and the maximum list length M_α.

$$C_\alpha = \frac{q^6}{\mathcal{V}}\, 2^{(\alpha - m_a)n}, \qquad \ell_{d,\alpha}(\mathbf{x}) = C_\alpha \det \Gamma_d\, \mathbf{1}_H(\mathbf{x}), \qquad M_\alpha = \left\lceil q^{12} 2^{\alpha n} (1 + \varepsilon/a)^{n/2} \right\rceil. \tag{6.6}$$

Write $A_\beta = 1 + 2^{2\beta}/4$ and $m_\beta = \frac{1}{2}\log_2 A_\beta$ for the limiting values. Theorem 1.5 uses $\alpha \leq m_\beta \leq m_a$. The variable-radius refinement also uses $\alpha = m_a$.

At an internal merge, test $\mathbf{x} - \mathbf{y} \in H$ and $(\mathbf{x} + \mathbf{y})/2 \in A_{r_{\mathrm{i}}}$, then accept the pair with probability $q/(C_\alpha \operatorname{vol}(A_{r_{\mathrm{i}}}))$. Assign each ordered pair of list positions a distinct integer e. At each merge, draw two independent uniform values $U, V \in [0, 1)$. Accept a pair that passes the tests when the fractional part of $eU + V$ is below the acceptance probability. The values for two distinct pairs are independent and uniform because the associated linear map on $(\mathbb{R}/\mathbb{Z})^2$ is surjective. Each repeated filter visit uses the same decision, and each accepted pair is output only once. At a leaf, use the uniform nearest-plane sampler with

$$H^+ = \{\mathbf{u} : |\|\mathbf{u}\|^2/r^2 - a| \leq \varepsilon + n^{-1}\}, \qquad N = \lceil C_\alpha \operatorname{vol}(H^+) \rceil. \tag{6.7}$$

A proposal rounding into H is accepted with probability $C_\alpha \operatorname{vol}(H^+)/N$. Lists are limited by M_α. Generate two independent root lists conditional on one common uniform shift modulo Γ_0, and search their equal labels modulo $\mathcal{L}$ for a shortest nonzero difference. The final search checks only whether the difference is nonzero and lies in $\mathcal{L}$.

6.2 Preparing the two shells and counting shortest differences

The first extension is chosen to control shifted counts in both midpoint shells. We use the random extension and dual-lattice argument of Section 5 for each shell.

Lemma 6.2 (Simultaneous midpoint preparation)**.** *With probability* $1-o(1)$*, preparation passes the terminal test and, with* $\xi = e^{-n/h^5}$*, satisfies*

$$\frac{\mathcal{V}}{2q^{10}} \le \det \Gamma_0 \le \frac{\mathcal{V}}{q^{10}}, \qquad \operatorname{Var}_{\mathbf{u}} K_{\Gamma_d,t}(\mathbf{u}) \le \xi m_{\Gamma_d,t}^2 \quad (0 \le d \le D,\ t \in \{r, r_{\mathrm{i}}\}), \tag{6.8}$$

where $K_{\Lambda,t}(\mathbf{u}) = \#((\Lambda+\mathbf{u}) \cap A_t)$ *and* $m_{\Lambda,t} = \operatorname{vol}(A_t)/\det\Lambda$*. Its cost is* $2^{O(n/h)}$ *on every execution.*

Proof. For either midpoint radius t, the radial integration in (5.8) gives

$$\sum_{\mathbf{v}\in\mathcal{L}\setminus\{\mathbf{0}\}} \left(1 - \frac{\|\mathbf{v}\|^2}{4t^2}\right)_+^{n/2} \le \operatorname{poly}(n)(bt/\lambda_1(\mathcal{L}))^n, \qquad \operatorname{vol}(A_t) = (1+O(n^{-1}))\frac{\mathcal{V}(t)}{2\sqrt{\pi}n^{3/2}}. \tag{6.9}$$

After the random extension, the diagonal and nonzero terms in the relative second moment are bounded, respectively, by polynomial multiples of

$$\frac{1}{p\mathcal{V}(t)} \le q^{-10}, \qquad \frac{(bt/\lambda_1(\mathcal{L}))^n}{p\mathcal{V}(t)} \le q^{-10}(br/\lambda_1(\mathcal{L}))^n \le q^{-10}e^{O(n\varepsilon)}. \tag{6.10}$$

Here $t \ge r$ and $br \le \widehat{\lambda}$. The powers of q absorb both the polynomial factors and scale-guess errors. For each shell apply the fine-lattice estimate of Lemma 5.3 at radius t. The random-line average (3.2) and the dual-lattice inclusion in Lemma 5.4 transfer the bounds to the chain. A union bound gives simultaneous preparation.

For basis preparation use $s_0 = \sqrt{2\pi e}\, r/\sqrt{n}$. The same Gaussian-mass estimate and Poisson summation give

$$\rho_{1/s_0}(\mathcal{L}^*) \le \frac{\operatorname{poly}(n)}{\mathcal{V}} \max\{1, (br/\lambda_1(\mathcal{L}))^n\} \le p/q. \tag{6.11}$$

The primal mass estimate ensures that the prime window tends to infinity. The reduction and cyclic halvings of Lemma 5.5, applied at length scale r, give the terminal test. Finally, $\det\Gamma_0 = 1/p$ gives the determinant bounds. □

Lemma 6.3 (Moments and shortest-vector representations)**.** *For a chain satisfying Lemma 6.2, the lists before imposing size limits satisfy the two moment bounds of Section 5.1.2, with target expected counts* $\ell_{d,\alpha}$*. If* Z *counts representations of a fixed shortest vector in the two root lists with midpoint in* A_r*, and*

$$\nu_\alpha = C_\alpha^2 \det\Gamma_0 \operatorname{vol}(A_r) = 2^{2(\alpha-m_a)n+o(n)}, \tag{6.12}$$

then, at a correct length guess,

$$\mathbb{E}Z \ge (1-o(1))\nu_\alpha, \qquad \mathbb{E}Z^2 \le (1+o(1))(\nu_\alpha^2+3\nu_\alpha), \qquad \Pr[Z>0] \ge (1-o(1))\frac{\nu_\alpha}{\nu_\alpha+3}. \tag{6.13}$$

Proof. The only new representation-volume calculation uses (6.3). For a prescribed internal difference $\mathbf{z} \in H$, a uniform $\mathbf{W} \in A_{r_{\mathrm{i}}}$ gives both endpoints $\mathbf{W} \pm \mathbf{z}/2$ in H with probability $1-O(\xi)$. For a

fixed shortest difference, the same assertion uses A_r. Indeed, the deterministic squared-length error is at most $(\varepsilon/4 + O(n^{-2}))r^2$, and spherical concentration controls the cross term by $e^{-\Omega(n\varepsilon^2)} \le \xi$.

The terminal rounding cells have radius $O(r/n^2)$ and fit inside H^+, so the leaf first moment is exact. The internal sharing terms are controlled uniformly in the space choice by

$$C_\alpha \operatorname{vol}(A_{r_\mathrm{i}}) = \Theta(q^6 n^{-3/2}) 2^{(\alpha - c)n}. \tag{6.14}$$

This makes the acceptance probability at most one and bounds each contribution from pairs sharing an input entry by $\mathrm{poly}(n) q^{-6} \le \xi$. The disjoint-pair case uses the prepared variance for A_{r_i}. Thus the same moment induction has total error $O(2^D \xi) = o(1)$.

For a shortest difference, use the variance bound for A_r. Since $\alpha \le m_a$, the maximum root weight is at most q^{-4}. The expansion in Lemma 5.13, with A_r as the midpoint shell, then gives the first and second moment bounds with normalizer ν_α. The second-moment bound gives the stated probability. The exact normalizer includes a subexponential prefactor of order $q^2 n^{-3/2}$. □

6.3 Spherical filtering with the smaller shell

Internal pairs that pass both tests have correlation $1/2 + O(\varepsilon)$. Pairs with a shortest-vector difference have correlation $\rho_f + O(\varepsilon)$, where $\rho_f = 2/a - 1 \in [1/3, 1/2]$. Use the BDGL product-code filters and enumeration procedure from Section 5, with target correlation $1/2$ at internal merges and ρ_f in the final search. Lemma 2.17 applies with common-cap factor $1 - 2u/(1+\rho)$ for the corresponding target correlation ρ.

For the fixed space parameter α, use one common threshold and cost bound:

$$u_\alpha = 1 - 2^{-2\alpha}, \qquad \Delta_\alpha = 1 - a u_\alpha, \qquad K_\alpha = \Delta_\alpha^{-n/2} 2^{o(n)}, \qquad f_a(\alpha) = -\tfrac{1}{2} \log_2 \Delta_\alpha. \tag{6.15}$$

The internal searches use

$$K_\mathrm{i} = (1 - \tfrac{4}{3} u_\alpha)^{-n/2} 2^{o(n)} \le K_\alpha,$$

since $a \ge 4/3$. Thus the bounds below apply to both kinds of search with the common upper bound K_α. For $c \le \alpha \le m_a$, we have $u_\alpha \le 1 - 1/a < 1/3$ and $\Delta_\alpha \ge 2 - a > 0.56$ for sufficiently large n, since $a \le 1 + e/(2\pi) + o(1)$. These parameters satisfy the decoder domain in Lemma 2.17. Independent repetitions reduce coverage errors to e^{-n^2} with polynomial overhead.

Let I count assignments of points to filters and J_i count pairs of entries with matching labels within each filter at an internal merge. A pair appearing in several filters is counted once per filter. The shift-averaging calculation of Lemma 5.18 becomes

$$\mathbb{E} I \le K_\alpha 2^{o(n)} [2^{2\alpha}(1 - u_\alpha)]^{n/2}, \qquad \mathbb{E} J_\mathrm{i} \le K_\alpha 2^{o(n)} [2^{4\alpha}(1 - u_\alpha)^2]^{n/2}. \tag{6.16}$$

Both bracketed factors equal one. The same calculation gives these bounds with K_i in place of K_α for internal searches. The count J_i includes all visits to a pair that shares several filters. The merge outputs each accepted pair of list entries only once.

Counting pairs in the final search. The final lists share one shift, so we count pairs by their difference and average over that shift. For a difference $\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\}$ put $T = \|\mathbf{v}\|^2 / \lambda_1(\mathcal{L})^2$ and $Q = a(1 - u_\alpha)/(a - 1)$. The translated-cap argument of Lemma 5.17 places its midpoint, after subtracting the cap centre, in a ball of squared radius

$$r^2 \{ a(1 - u_\alpha) - (a - 1) T + O(\varepsilon) \}. \tag{6.17}$$

Averaging over the common shift gives coefficient $C_\alpha^2 \det \Gamma_0$. The same centered point-count integration gives

$$\sum_{\mathbf{v}\neq\mathbf{0}} \left(\frac{Q - \|\mathbf{v}\|^2/\lambda_1(\mathcal{L})^2}{Q-1} \right)_+^{n/2} \le 2^{o(n)} \left(\frac{b^2 Q^2}{4(Q-1)} \right)^{n/2}, \tag{6.18}$$

$$\mathbb{E} J_f^{\neq 0} \le K_\alpha 2^{o(n)} \left(\frac{b^2}{4(a-1)} \right)^{n/2} \le K_\alpha 2^{o(n)}. \tag{6.19}$$

In the last line the factor $2^{2(\alpha - m_a)n}$ cancels the remaining list and cap factors because $2^{2\alpha}(1-u_\alpha) = 1$. Zero differences are charged to incidences as before. Only centered input-lattice counts enter this calculation.

Averaging the expected list size over the shift gives at most $q^6 2^{\alpha n}(1+\varepsilon/a)^{n/2}$. Since $f_a(\alpha) \ge \alpha$, initialization and list processing fit within the search work. With only $2^{o(n)}$ recursive calls, one uncapped reference trial has expected work $2^{f_a(\alpha)n + o(n)}$.

6.4 Work limits and proof of the tradeoff

As in Lemma 5.14, weight each execution by the number of shortest-vector representations it produces, then choose one such representation uniformly. Follow its input entries down the recursion. The final pair has midpoint in A_r, and the pair at each internal merge has midpoint in A_{r_i}. The variance bounds for both shells let us compare the distribution of these shifts with uniform shifts by Cauchy–Schwarz, as in Lemma 5.14.

For pairs involving neither of the two entries being followed, the work bounds are (6.16) and (6.19). Pairs involving one of these entries contribute at most $2^{\alpha n + o(n)} K_\alpha (1-u_\alpha)^{n/2} = K_\alpha 2^{o(n)}$ visits. The pair of the two chosen entries contributes at most $K_\alpha 2^{o(n)}$ visits, including repeated visits across filters. The moment bounds of Lemma 6.3 give the corresponding bounds on list sizes. We can thus set the list and work limits at $2^{\alpha n + o(n)}$ and $2^{f_a(\alpha)n + o(n)}$, respectively, so that the count $\widehat{Z}$ after imposing both limits satisfies

$$\mathbb{E}\widehat{Z} \ge (1 - o(1))\mathbb{E}Z. \tag{6.20}$$

As in Lemma 5.16, repetitions reduce the probability of missing a required pair to $e^{-n^2 + O(n)} = o(\nu_\alpha)$. Imposing the limits can only decrease the second moment. The probability bound in (6.13) therefore holds up to a factor $1 - o(1)$.

Proof of Theorem 1.5. For $c \le \beta \le \beta_{\mathrm{lat}}$, the capped trials use space $2^{\alpha n + o(n)}$, cost $2^{f_a(\alpha)n + o(n)}$, and succeed with probability at least $2^{-2(m_a - \alpha)n + o(n)}$. Independent repetition gives success at least $2/3$ with time exponent

$$f_a(\alpha) + 2m_a - 2\alpha = 2m_a - 2\alpha - \tfrac{1}{2}\log_2(1 - a + a2^{-2\alpha}). \tag{6.21}$$

Since $a = A_\beta + o(1)$ and $m_a = m_\beta + o(1)$, this is the exponent in Theorem 1.5. As in Section 5, run trials and length guesses sequentially, enforce work limits, and use a standard exact algorithm in finitely many small dimensions.

For a stated $\beta > \beta_{\mathrm{lat}}$, use the universal point-count bound with β_{lat} and space parameter $\min\{\alpha, m_{\beta_{\mathrm{lat}}}\}$. This satisfies the claimed space bound. For fixed α in its domain, the time exponent increases with A_β, because, writing $X = 2^{2\alpha}$ and $D = A_\beta - (A_\beta - 1)X$,

$$\frac{\partial}{\partial A_\beta} \ln \frac{A_\beta^2}{XD} = \frac{2}{A_\beta} + \frac{X-1}{D} > 0. \tag{6.22}$$

If $\alpha > m_{\beta_{\mathrm{lat}}}$, the universal minimum-time endpoint also suffices: $t_{\beta,\mathrm{C}}(\alpha)$ decreases on $[c, m_\beta]$ and is at least $-\frac{1}{2}\log_2(2 - A_\beta)$, which increases with β. This proves the full stated range without changing radii along the chain or as a function of the chosen space parameter. □

6.5 The radius balance in the cost bound

Keep $\det \mathcal{L} = 1$ and write $y = (R/\lambda_1(\mathcal{L}))^2$. In these units the final and internal midpoint radii are $\sqrt{y - 1/4}$ and $\sqrt{3y/4}$, respectively. Write $\mathcal{V}(t)$ for the volume scale in (6.4) at physical radius t. In the second moment of the count in the midpoint shell, the term for two equal points contributes $1/\mathcal{V}(t)$. The nonzero input-lattice differences contribute at most $b^n/\mathcal{V}(\lambda_1(\mathcal{L}))$, up to subexponential factors. Thus a sufficient first index has scale

$$p(y) = \frac{2^{o(n)}}{\mathcal{V}(\lambda_1(\mathcal{L}))} \max\{b, (y - 1/4)^{-1/2}\}^n. \tag{6.23}$$

The two terms agree at $y_\star = 1/4 + b^{-2}$. At the universal point-count exponent this gives radius $0.909870\ldots$. For another common radius in the decoder range $3/4 \le y \le 1$, put $x = 2^{-2\alpha}$. Equation (6.23) gives

$$B_y = \max\{b, (y - 1/4)^{-1/2}\}, \qquad m_y = \tfrac{1}{2}\log_2 \frac{B_y y}{\sqrt{y - 1/4}}, \qquad a_y = \frac{y}{y - 1/4}. \tag{6.24}$$

Here m_y is the list-size exponent at which a trial succeeds with probability $2^{-o(n)}$. For threshold $u_\alpha = 1 - x$, the time exponent including repetitions is

$$T_y(\alpha) = 2m_y - 2\alpha - \tfrac{1}{2}\log_2(1 - a_y + a_y x). \tag{6.25}$$

For $y \le y_\star$, both $m_y = \frac{1}{2}\log_2 a_y$ and a_y decrease as y increases, so this bound decreases towards $y_\star$. For $y \ge y_\star$, it simplifies to

$$T_y(\alpha) = \log_2(byx) - \tfrac{1}{2}\log_2(yx - 1/4). \tag{6.26}$$

Its derivative with respect to y has the sign of $yx - 1/2$, which is positive because $\alpha \le m_\beta$ gives $yx \ge b^{-2} > 1/2$ in the directly analyzed range $\beta \le \beta_{\mathrm{lat}}$. Hence $y_\star$ minimizes these fixed-radius bounds for every space choice under consideration. The same choice minimizes the required root-list exponent used in the quantum construction below. This is an optimization of this construction and its proved cost bounds.

6.6 Quantum reporting at the same fixed radius

Theorem 6.4 (Smaller fixed-radius quantum SVP tradeoff)**.** *Under the point-count hypothesis* (1.6) *of Theorem 1.5, use* $c = \frac{1}{2}\log_2(4/3)$ *and* $m_\beta = \frac{1}{2}\log_2(1 + 2^{2\beta}/4)$ *as above and set* $q_0 = \frac{1}{2}\log_2(13/9)$. *With coherent indexed access to stored classical arrays, for every space exponent* $s \in [c, \max\{m_\beta, q_0\}]$ *there is a quantum exact SVP algorithm with success probability at least* $2/3$, *space* $2^{sn+o(n)}$, *and time* $2^{t_{\beta,\mathrm{Q}}(s)n+o(n)}$. *Writing* $\bar{s} := \min\{s, q_0\}$, *its time exponent is*

$$\begin{aligned} \psi(s) &:= \min\left\{\tfrac{1}{2}\log_2(9/8), \bar{s} - \tfrac{1}{4}\log_2\Big(4(2^{2\bar{s}} - 1)\Big)\right\}, \\ t_{\beta,\mathrm{Q}}(s) &= 2m_\beta - \min\{s, m_\beta\} + \psi(s). \end{aligned} \tag{6.27}$$

Its minimum-time endpoint has time exponent $m_\beta + \frac{1}{2}\log_2(13/12)$ *and space exponent* $\max\{m_\beta, q_0\}$. *The construction uses one common physical shell radius independent of the lattice level and the chosen space exponent; it may depend on* β.

Use the construction of Section 5.4, generating lists with size exponent c and concatenating them at the final level. Every vector lies in the same shell H of radius R. Quantum enumeration speeds up the internal merges, and quantum minimum finding searches the two final lists directly.

Let s be the total space exponent and put $\alpha = \min\{s, m_a\}$. Each base list uses the coefficient C_c from (6.6). Concatenate $k = 2^{(\alpha-c)n+o(n)}$ independent lists on each side, conditional on the common shift. Both moment bounds hold with coefficient kC_c. This gives the same exponential rate $\nu_\alpha = 2^{2(\alpha-m_a)n+o(n)}$ for the expected number of shortest-vector representations.

The space used inside a base call. Write $q_0 = \frac{1}{2}\log_2(13/9)$ and $\bar{s} = \min\{s, q_0\}$. For $c \le \bar{s} \le q_0$, use squared internal cap threshold

$$w_{\bar{s}} = \frac{3\,2^{2\bar{s}} - 4}{4(2^{2\bar{s}} - 1)}, \qquad \Delta_{\bar{s}} = 1 - \frac{4}{3}w_{\bar{s}} = \frac{1}{3(2^{2\bar{s}} - 1)}. \tag{6.28}$$

The useful internal correlation is $1/2$ and $0 \le w_{\bar{s}} \le 3/16$. Store the list entries in each filter and index the pairs with matching labels as in Lemma 5.21. Each pair of list positions has a fixed acceptance decision. Let I count the stored entries, J the indexed pairs, and J_{acc} the indexed pairs accepted by the merge. The exponents i, j bound the corresponding means of I, J over the shifts.

$$i = c + \tfrac{1}{2}\log_2 \frac{1 - w_{\bar{s}}}{\Delta_{\bar{s}}} = \bar{s}, \qquad j = 2c + \log_2(1 - w_{\bar{s}}) - \tfrac{1}{2}\log_2 \Delta_{\bar{s}}, \qquad \mathbb{E}J_{\mathrm{acc}} \le 2^{cn+o(n)}. \tag{6.29}$$

Quantum enumeration and Cauchy–Schwarz give a base time exponent at most $\max\{\bar{s}, (j+c)/2\}$, including construction and loading of the tables. This equals

$$q_{\mathrm{base}}(\bar{s}) = c + \bar{s} - \tfrac{1}{4}\log_2\Big(4(2^{2\bar{s}} - 1)\Big). \tag{6.30}$$

The filter count is $K = \Delta_{\bar{s}}^{-n/2} 2^{o(n)} \le 2^{cn+o(n)}$, so constructing empty filters is covered as well. The table space has exponent $\bar{s}$. At $\bar{s} = q_0$, time and space are both q_0. Alternatively, the streamed classical base merge costs $g = \frac{1}{2}\log_2(3/2)$ in space exponent c. We use whichever base implementation is faster under the space limit. The excess time over the base list size is therefore $\psi(s)$ from Theorem 6.4.

Success probability under work limits. To bound how often the work limits remove a shortest-vector pair, follow the two entries of a chosen representation as in Lemma 5.22. Use $A_{r_{\mathrm{i}}}$ inside the base calls and A_r for the final pair. A failed base call returns an empty list and the remaining calls continue. Only the two calls supplying the chosen entries can remove that representation. The union bound therefore covers only these two calls, even when k is exponential.

Internally the normalized geometry and base-list exponent are exactly those of Section 5.4. The same conditional bounds for I, J, J_{acc} apply with the threshold in (6.28). The auxiliary code is tuned to the actual internal angle, so each eligible pair has subexponential expected filter multiplicity. Cauchy–Schwarz bounds the enumeration time conditional on the entries being followed. The limits on list size, stored filter assignments and work therefore leave a fraction $1 - o(1)$ of the expected count of representations. Repetition reduces filter and enumeration failure probabilities to e^{-n^2} as in Lemma 5.22.

Proof of Theorem 6.4. Generating the k base lists sequentially costs $2^{(\alpha+\psi(s))n+o(n)}$. Their root-list storage and the base incidence tables fit in space $2^{sn+o(n)}$. Final minimum finding on the Cartesian product costs $2^{\alpha n+o(n)}$ and has constant success conditional on a shortest-vector pair being present.

The probability of finding a shortest vector after imposing the work limits is $2^{-2(m_a-\alpha)n+o(n)}$. Independent repetition therefore gives total time exponent

$$2m_a - \alpha + \psi(s) = 2m_a - \min\{s, m_a\} + \psi(s). \tag{6.31}$$

Taking limiting parameters proves the stated quantum curve for $\beta \le \beta_{\mathrm{lat}}$. For larger supplied β, use the same universal-profile reduction as in the classical proof. The bound $2m_\beta - \min\{s, m_\beta\} + \psi(s)$ increases with m_β; if the space choice exceeds the universal minimum-time space, its endpoint already satisfies the claimed bounds. □

Proof of Theorem 6.1. Set $\beta = \frac{1}{2}\log_2(2e/\pi)$ using Proposition 1.1. In the classical tradeoff of Theorem 1.5, choose $\alpha = m_\beta$. This gives space exponent m_β and time exponent $-\frac{1}{2}\log_2(2 - A_\beta)$. The smallest quantum time exponent in Theorem 6.4 is $m_\beta + \frac{1}{2}\log_2(13/12)$ at space exponent $\max\{m_\beta, q_0\} = q_0$ for this β. These are the four exponents in (6.1). □

At $\beta = c$, the quantum exponents both equal q_0, as in Section 5. Section 8 improves list generation by varying the radii and searching the two child lists differently.

7 Classical SVP in Time $2^{0.4085...n+o(n)}$ – Fast Matrix Multiplication

We use the superlattice chain, list constructions, shell radii, midpoint tests, and acceptance rules of Section 6. The internal merges are unchanged. In the final search we still use BDGL filters to form buckets, but change the search inside each bucket. We form random signed sums of groups of vectors and use repeated matrix products to identify candidate pairs. We then check the corresponding original vectors.

By reducing the costs of a search inside a BDGL bucket, we can reoptimize the parameters for the final BDGL nearest neighbor search. Instead of having $2^{o(n)}$ vectors per filter, the time complexity of the final merge step now becomes slightly smaller when we choose the parameters to get on average $2^{0.0105...n+o(n)}$ vectors per filter. Since the final merge step dominates the overall time complexity, this leads to a reduced worst-case SVP time exponent as well.

Theorem 1.7 (Classical SVP with a batched final search, restated)**.** *For every input lattice $\mathcal{L}$ of rank n, a randomized algorithm solves exact SVP with success probability at least $2/3$ in time $\mathrm{T} = 2^{\tau_{\mathrm{C}}^\star n+o(n)}$ and space $\mathrm{S} = 2^{s_{\mathrm{C}} n+o(n)}$, where*

$$\tau_{\mathrm{C}}^\star = 0.408502827340608\ldots, \qquad s_{\mathrm{C}} = \tfrac{1}{2}\log_2\left(1+\frac{e}{2\pi}\right).$$

This approach follows the correlation-search strategy of Karppa–Kaski–Kohonen [KKK18]: strengthen the desired correlations inside the buckets, form random signed sums of groups, and use a matrix product to locate promising pairs. Gao–Hu [GH26] recently used the same strategy inside spherical filters for heuristic lattice sieving, where they obtained a conditional reduction in the heuristic SVP exponent if the matrix multiplication exponent ω is reduced below $9/4$. In our case, the improvement applies unconditionally: already with the current rectangular matrix multiplication upper bounds we obtain a reduced SVP time exponent.

Roadmap. Section 7.1 gives the matrix search, Section 7.2 constructs its polynomial coordinates, and Section 7.3 evaluates the total SVP cost.

7.1 Matrix multiplication and grouped search

Lemma 7.1 (Rectangular multiplication [Uzn26, Corollary 1.2])**.** *For fixed $x \geq 0$, multiplying a $\lceil d^x \rceil \times d$ real matrix by a $d \times \lceil d^x \rceil$ matrix costs $d^{\mu(x)+o(1)}$ operations, where $\mu(x) = 1 + x + x^2/4$ for $0 \leq x \leq 2$, and $\mu(x) = 2x$ for $x \geq 2$.*

If rows of A, B are signed sums of groups of vectors, each entry of AB^{T} is the sum of their pairwise inner products with the corresponding signs. Repeated matrix products, including tests on bits of the point indices, find pairs whose inner products stand out.

Lemma 7.2 (Searching grouped vectors)**.** *Let two lists contain N unit vectors in total, each with D_Φ coordinates. Put $P = \tau^{-2}\sum_{i,j}\langle a_i, b_j\rangle^2$ and $\tau^2 = D_\Phi^{-b_\Phi}$, with $0 \leq b_\Phi \leq 1$. For each fixed $\vartheta \geq 1$, all pairs of score magnitude at least τ occur among the candidates, with high probability, in time*

$$\widetilde{O}((ND_\Phi^\vartheta + P)D_\Phi^{g+o(1)}), \quad g = (3 + b_\Phi - 2\vartheta)_+^2/16. \tag{7.1}$$

There are $\widetilde{O}(ND_\Phi^\vartheta + P)$ candidates. For replayable inputs and streamed checks, working memory is $\widetilde{O}(N + \sqrt{P + ND_\Phi^\vartheta + 1}\, D_\Phi + D_\Phi^{4+o(1)})$.

Proof. Write $D = D_\Phi$ and $b = b_\Phi$. For independent signs σ_j, put

$$X_i = \Big\langle a_i, \sum_j \sigma_j b_j \Big\rangle, \qquad R_i = \sum_j \langle a_i, b_j\rangle^2.$$

Then $\mathbb{E}X_i^2 = R_i$ and $\mathbb{E}X_i^4 \le 3R_i^2$. Constant-size block averages of X_i^2, followed by median amplification over $O(\log(ND+2))$ blocks, estimate all row and column sums within constant factors with high probability, in $\widetilde{O}(ND)$ time. Multiplying the sum of the row estimates by τ^{-2} and a suitable absolute constant gives a constant-factor upper estimate of P. Fix the resulting dyadic classes before drawing further randomness. Discard classes whose normalized upper bound is below one. On a rectangle of sizes $u \le v$, normalized row and column bounds are then $O(P/u)$ and $O(P/v)$.

A signed coordinate hash to dimension d adds expected normalized squared score $O(uv/(d\tau^2))$. Use $d \ge \tau^{-2}$ with polylogarithmic guards, and exact coordinates when $d = D$. For budget W, use group counts $(u, W/u)$ if $u \le \sqrt{W}$, and $(\sqrt{W}, \sqrt{W})$ otherwise, with no more groups than vectors. Endpoint signs give noise variance $O(W/(m_1m_2) + W/(um_2) + W/(vm_1))$ for group counts m_1, m_2; singleton sides remove their collision terms.

For the signed coordinate hash S,

$$\mathbb{E}\big|\langle Sa, Sb\rangle - \langle a, b\rangle\big|^2 \le 2/d.$$

Use guards of size $C\log^2(ND+2)$ in the compressed dimension and group counts, switching to exact coordinates or singleton groups when their full sizes are reached. For sufficiently large C, a fixed target has compressed score of magnitude at least $3\tau/4$, while the noise in all its index-bit tests is at most $\tau/8$, with constant probability. Restricting a test to indices whose specified bit is one includes either the full target contribution or none of it. Threshold $\tau/4$ therefore recovers that bit. Tests on both sides recover the two indices. Each cell emits at most one candidate. Independent $O(\log(ND+2))$ repetitions find all target pairs with high probability. These factors are absorbed by $\widetilde{O}$.

Write $u = D^{r_1}$, $v = D^{r_2}$, $P = D^p$, $M = \max\{r_2 + \vartheta, p\}$. Emit all pairs if $r_1 \le \vartheta$ or $r_1 + r_2 \le M$. Otherwise choose

$$W = D^M, \qquad \ell = \min\{r_1, M/2\}, \qquad d = D^\delta, \quad \delta = \min\{1, 2\ell - 2\vartheta + b\}.$$

Here $\ell > \vartheta$ and $\delta \ge b$. For $\delta < 1$, added squared scores fit within W; at $\delta = 1$ they are exact. Rectangular tiles in Lemma 7.1 cost $D^{M+(2\delta-\ell)_+^2/(4\delta)+o(1)}$. Substitution bounds the extra exponent by g. Summing the classes gives (7.1). Store the smaller grouped list and stream tiles of at most d^4 entries; this gives the stated memory and candidate bounds. Original pairs can be checked after their indices are emitted. □

7.2 Coordinates for the final BDGL buckets

Use $\alpha = m_a$ and the limiting value $a = 1 + e/(2\pi)$ from Section 6.1. For the final cap threshold put

$$u = 1 - \frac{a-1}{a}Q, \quad \Delta = (a-1)(Q-1), \quad \rho_Q = 1 - \frac{2}{Q}, \quad \iota = \tfrac{1}{2}\log_2\frac{Q}{Q-1}, \quad 2 < Q < \frac{a}{a-1}. \tag{7.2}$$

Here $\Delta = 1 - au$ is the common-cap factor for a shortest difference. Thus $K = \Delta^{-n/2}2^{o(n)}$ filters give expected total assignments $2^{\iota n + o(n)}$. A bucket from one list has mean $2^{m_a n + o(n)}(1-u)^{n/2} = 2^{\frac{1}{2}\log_2((a-1)Q)n + o(n)}$. The old $Q = 2\pi/e$ gives zero bucket exponent; the new value below gives $0.0105446904\ldots$.

Randomly rotate each cap code and keep projection heights in $[R\sqrt{u}, R\sqrt{u}+\delta_n\widehat{\lambda}]$, with $\delta_n \to 0$ slowly enough to dominate shell and code errors. Common-cap mass outside this band decays by $e^{-\Omega(n\delta_n)}$, so target coverage changes by $o(1)$. Subtract the component along the filter centre and normalize. Target pairs then have inner product $\rho_Q + o(1)$. The calculation of (6.17) and the point-count bound (2.3) give the expected pair profile

$$F_Q(T) = \beta + \tfrac{1}{2}\log_2 T + \tfrac{1}{2}\log_2 \frac{Q-T}{Q-1} = B_Q + \tfrac{1}{2}\log_2(1-t^2), \tag{7.3}$$

where $T = \|w\|^2/\lambda_1^2$, $1 \le T < Q$, $t = 1-2T/Q$, $\beta = \frac{1}{2}\log_2(2e/\pi)$ and $B_Q = \beta - \frac{1}{2}\log_2(1-\rho_Q^2)$. This bound sums over filters and equal lattice labels, and permits nonnegative radial weights. For every pair in the projection bands, let $\widetilde{t}$ denote its actual inner product after projection and normalization. Uniformly over these pairs,

$$\widetilde{t} = 1 - \frac{2T}{Q} + O(\delta_n + \varepsilon).$$

Put $Q_n = Q + C(\delta_n + \varepsilon)$ for a sufficiently large fixed constant C. Then $T \le Q_n$, $B_{Q_n} = B_Q + o(1)$, and $\widetilde{t} = t_0 + O(\delta_n + \varepsilon)$, where $t_0 = 1 - 2T/Q_n$. For the upper bounds below, use (7.3) with Q_n in place of Q, including B_{Q_n}. Give each vector value a consistent random red/blue colour and search only different colours. This excludes zero differences; a fixed nonzero target survives with constant probability.

Lemma 7.3 (Polynomial coordinates)**.** *For the parameters below there is an explicit map to unit vectors with $D_\Phi = 2^{\gamma n + o(n)}$ coordinates, computed in $D_\Phi\,\mathrm{poly}(n)$ time. Target scores are at least $\tau_n = 2^{\chi n - o(n)}$, and the total squared-score quantity in Lemma 7.2 satisfies $\mathbb{E}P_{\mathrm{tot}} \le 2^{j_Q n + o(n)}$. The rates can be evaluated by*

$$\begin{aligned} \gamma &= (1+\kappa)\log_2(1+\kappa) - \kappa\log_2\kappa, \\ \chi &= -\tfrac{1}{2}\log_2(1 - 2\rho_Q v + v^2) - \kappa\log_2 v - \gamma, \\ j_Q &= B_Q - \gamma - 2\chi, \qquad b_\Phi = -2\chi/\gamma, \end{aligned} \tag{7.4}$$

where v is the smaller positive root of $(1+\kappa)v^2 - (1+2\kappa)\rho_Q v + \kappa = 0$.

Proof. Let $n' = n + o(n)$ be the ambient dimension after the zero padding of Lemma 2.17. Use the standard degree-k polynomial map on the unit sphere from [DX13, Sections 2 and 5], with $k = \kappa(n'-1) + O(1)$. Its normalized inner-product polynomial is

$$p_{n,k}(t) = \frac{[z^k](1 - 2tz + z^2)^{-\nu}}{[z^k](1-2z+z^2)^{-\nu}}, \qquad \nu = (n'-3)/2,$$

where $[z^k]$ denotes a coefficient. The cited coordinate construction costs $D_\Phi\,\mathrm{poly}(n)$; Stirling's bounds give γ. On $|z| = \varrho$, use $|1 - 2tz + z^2|^2 \ge (1-\varrho^2)^2(1-t^2)$ and $\varrho^2 = k/(k+2\nu)$ in the coefficient bound. After normalization,

$$|p_{n,k}(t)|^2 \le \min\{1, 2^{-\gamma n}(1-t^2)^{-n/2}\}\, 2^{o(n)}. \tag{7.5}$$

At $t = \rho_Q$, the smaller critical radius is $v + O(1/n)$. The strict inequality $\rho_Q > 2\sqrt{\kappa(1+\kappa)}/(1+2\kappa)$ gives a unique point of maximal modulus at the positive real point and a positive nondegenerate leading term. Evaluating it gives χ, uniformly near ρ_Q.

Apply the counting profile to the supremum of (7.5) over $|\widetilde{t} - t_0| \le \eta_n$, where $\eta_n = O(\delta_n + \varepsilon) = o(1)$. This is a nonnegative radial weight. Uniformly over these pairs,

$$(1-t_0^2)^{n/2}\min\{1, 2^{-\gamma n}(1-\widetilde{t}^2)^{-n/2}\} \le (2^{-2\gamma} + O(\eta_n))^{n/2} = 2^{-\gamma n + o(n)}.$$

This follows from $|(1-\widetilde{t}^2)-(1-t_0^2)| = O(\eta_n)$ and the clipped bound 1. Since $B_{Q_n} = B_Q + o(1)$, radial summation and division by τ_n^2 give $j_Q = B_Q - \gamma - 2\chi$. No independent-direction assumption is used. □

7.3 Numerical choice and total SVP cost

Applying Lemma 7.2 in each bucket and lattice label gives

$$t(Q,\kappa,\vartheta) = \max\{\iota + \vartheta\gamma, j_Q\} + \gamma(3 + b_\Phi - 2\vartheta)_+^2/16. \tag{7.6}$$

The analytic optimization gives

$$Q \approx 2.3454918202, \qquad \kappa \approx 0.0004734417391, \qquad \vartheta \approx 1.2286357509.$$

Only a feasible choice is needed for the SVP upper bound. Direct evaluation of (7.2) and (7.4) gives

Quantity	Numerical value
Assignments ι	0.400878424616434
Coordinates γ	0.005912133061544
Target score χ	−0.001315650319066
Squared scores j_Q	0.408142282659857
Matrix overhead $g\gamma$	0.000360544680751749

The printed decimals are approximate. At the limiting choice, $\iota + \vartheta\gamma = j_Q$ and $t = j_Q + g\gamma = 0.408502827340608\ldots$, compared with $0.408816556928413\ldots$ in Section 6.

To define the constant without fixed rounding, let $\tau_C^\star$ be the minimum of (7.6) on

$$[2.34549, 2.34550] \times [0.0004734, 0.0004735] \times [1.2286, 1.2287].$$

On this box $0.4445 < b_\Phi < 0.4456$, $u < 0.292$, $\Delta > 0.582$, and the target discriminant exceeds 0.0198. The space bounds below also hold throughout the box, so the estimates are uniform. An n^{-2} grid, with $O(n^{-2})$ accuracy in evaluating the cost, has polynomial size and approaches this minimum. Thus finite parameter computation causes no loss in the exponent. The numerical value above comes from the analytic optimization; its optimality derivation is unnecessary for the algorithmic bound.

Proof of Theorem 1.7. Lemma 6.3, at $\alpha = m_a$, gives constant probability of representing a fixed shortest vector; the normalizer contains $\Theta(q^2 n^{-3/2})$. Filtering, colouring and the grouped search keep it with constant probability. A sufficiently large constant-factor work cutoff and repetition give success at least 2/3. Length guesses and exact minimization are as in Section 6.4.

For earlier merges, use correlation 1/2 and the existing u_α in Lemma 2.17. Equation (6.16) gives internal exponent $0.3716670890\ldots < \tau_C^\star$; the costs of the internal merges are negligible compared to the costs of the final search. Store both root lists and use the product partitioning from the proof of Lemma 2.18. For each product, enumerate memberships from both lists, stopping when their combined incidence buffer exceeds N records. Then split a nonsingleton code factor and process its two parts sequentially. A single filter contains at most N records, so this terminates. At each depth the products have disjoint filter sets, and at most I/N products exceed the buffer limit. The same charging argument gives $2^{o(n)}(N + I + 1)$ time and $O(N)$ incidence records. This procedure rescans the stored root lists. For each completed buffer, group by filter and lattice label and apply Lemma 7.2 instead of enumerating all pairs.

Throughout the parameter range, the search lemma adds space exponents $\max\{\iota+\vartheta\gamma, j_Q\}/2+\gamma < 0.210$ and $4\gamma < 0.024$, both below s_C. The original list bank therefore determines space; the final bucket search determines time. All remaining factors are $2^{o(n)}$. □

8 Quantum SVP in Time $2^{0.2937\ldots n+o(n)}$ – Variable Shells

Theorem 8.1 (Variable-radius binary quantum SVP)**.** *Exact SVP on every input lattice admits a quantum algorithm with success probability at least* $2/3$ *and time and space*

$$\mathrm{T_Q} = 2^{t_\star n+o(n)}, \qquad \mathrm{S_Q} = 2^{s_\star n+o(n)}, \qquad s_\star = t_\star = 0.2937623291\ldots. \tag{8.1}$$

More precisely, put $a_\star = 1 + e/(2\pi)$ *and* $v_\star = 1 + a_\star/4$*, and let* $\tau_\star$ *be the unique root in* $(0.929, 0.930)$ *of*

$$a_\star(1+\tau_\star)\left(1 - \frac{v_\star(1+\tau_\star)}{4}\right)\left(v_\star - (v_\star - 1)\tau_\star^2\right) = 1. \tag{8.2}$$

Then

$$t_\star = -\tfrac{1}{2}\log_2\left[(1+\tau_\star)\left(1 - \frac{v_\star(1+\tau_\star)}{4}\right)\right]. \tag{8.3}$$

The cap thresholds use the exact root $\tau_\star$ *of this equation.*

Section 6 uses one shell radius throughout the chain. Quantum minimum finding makes its final search cost about as much as writing the two top lists. Generating those lists is the more expensive step. We reduce that cost in two stages.

Smaller shells at deeper levels. Keep the radius of the top lists, so the analysis of the final shortest-vector search still applies. Use the final search's smaller midpoint shell at every internal merge. The parallelogram identity then determines the child radius from the parent radius. The radii decrease down the chain, so producing the lists is cheaper. With the same filter threshold for both child lists this already gives exponent $0.298756\ldots$.

Different thresholds for data and queries. Store the cap memberships for one child list. For a vector from the other list, compute its matching cap addresses as the search visits them. Thus storage and construction pay directly for the first list, while the second list's cap matches enter the square-root cost of quantum reporting. Using two thresholds balances these costs and gives the exponent in the theorem.

The search also applies when the queries represent partial sums in Sections 9–11.

Roadmap. Section 8.1 derives the shell radii and bounds the cost with equal filter thresholds. Section 8.2 develops the asymmetric index and query procedure. Section 8.3 combines these bounds to obtain the stated time and space.

8.1 The variable-radius reference lists

Use the parameters and prepared chain of Section 6 at the universal centered point-count bound. Thus $a = 1 + b^2/4 + o(1)$, $b^2 = 2e/\pi + o(1)$, and $C = q^6/\mathcal{V}$. Keep the root shell, but use the smaller midpoint shell A_r at every internal merge. The parallelogram identity gives

$$\begin{aligned} &R_0^2 = ar^2, && R_{d+1}^2 = r^2 + R_d^2/4, && \nu_d := R_d^2/r^2 = \frac{4}{3} + \frac{a - 4/3}{4^d}, \\ &H_d = \{\mathbf{x} : |\|\mathbf{x}\|^2/r^2 - \nu_d| \le \varepsilon\}, && \ell_d(\mathbf{x}) = C \det \Gamma_d \, \mathbf{1}_{H_d}(\mathbf{x}), && m_d = \tfrac{1}{2}\log_2 \nu_d. \end{aligned} \tag{8.4}$$

The radii decrease from $0.909870\ldots\widehat{\lambda}$ at the root to $0.877772\ldots\widehat{\lambda}$ at the dense end. An eligible pair is accepted with probability $q/(C\,\mathrm{vol}(A_r))$. Leaves use the sampler of (6.7), with H_D and its enlargement by n^{-1}.

Lemma 8.2 (Binary reference moments)**.** *On a prepared chain, these uncapped lists satisfy the first and second moment bounds for distinct occurrences in Lemma 6.3, with weights (8.4). Their mean sizes are $2^{m_d n+o(n)}$. Two root lists drawn independently given their common uniform shift contain a shortest-vector difference with probability $1-o(1)$.*

Proof. For a prescribed $\mathbf{z}\in H_d$, a uniform $\mathbf{W}\in A_r$ places both endpoints $\mathbf{W}\pm\mathbf{z}/2$ in H_{d+1} except with probability $O(\xi)$. The radius identity leaves deterministic squared-length error at most $(\varepsilon/4+O(n^{-2}))r^2$, and the cross term is controlled by spherical concentration. The leaf cells fit inside the enlarged shell. Since $C\,\mathrm{vol}(A_r)=\Theta(q^6n^{-3/2})$, the same three sharing cases and variance bound from the prepared chain in Lemma 6.3 propagate both moments with total error $O(2^D\xi)=o(1)$. Averaging over a uniform shift gives the expected list sizes. Since the root shell is unchanged, the normalizing factor for the shortest-vector count is

$$C^2\det\Gamma_0\,\mathrm{vol}(A_r)=\Theta(q^2n^{-3/2})\longrightarrow\infty. \tag{8.5}$$

The second-moment bound from Section 6 therefore gives success probability $1-o(1)$. □

At a level-d merge, put $v=\nu_{d+1}=1+\nu_d/4$ and $l=\frac{1}{2}\log_2 v$. Useful child pairs have correlation $\rho=2/v-1+O(\varepsilon)$. With equal squared cap thresholds $u=(1-1/\nu_d)/v$, their common-cap factor is $\Delta=1/\nu_d$. The explicit incidence reporter of Section 6.6 then has exponents

$$k_d=m_d,\qquad i_d=Q_d,\qquad j_d=2Q_d-m_d,\qquad Q_d=\tfrac{1}{2}\log_2(1+\nu_d^2/4) \tag{8.6}$$

for filters, incidences and candidate addresses. Reporting the accepted addresses costs $2^{Q_d n+o(n)}$. The rates decrease with d; the root gives $Q_0=0.2987564049\ldots$. The following access construction reduces the cost of querying the second child list.

8.2 An asymmetric index and implicit queries

For a unit direction $\mathbf{w}$, write $\mathcal{C}_t(\mathbf{w})=\{\mathbf{z}:\|\mathbf{z}\|=1,\ \langle\mathbf{z},\mathbf{w}\rangle\ge t\}$. Lists and query families below are indexed by occurrences. A query may be a stored vector or a vector computed from an existing prefix. For each cap address, store a list of the data occurrences in that cap. The query tree has one vertex for each query and matching cap address. Its descendants visit that cap's data list. A data occurrence can therefore be visited more than once when it lies in several matching caps.

Lemma 8.3 (Asymmetric cap access)**.** *Let a stored data family and a query family of nonzero vectors have mean sizes at most $2^{un+o(n)}$ and $2^{\pi n+o(n)}$. Queries are accessible at designated vertices of a fixed binary tree of polynomial depth, and their vectors are reversibly computable with $2^{o(n)}$ work. Data ranks have polynomial bit length. Choose thresholds $\alpha_{\mathrm{d}},\alpha_{\mathrm{q}}$ and target correlation ρ in a fixed compact subset of the strict two-active-cap region defined below. Put*

$$\begin{aligned} A&=-\tfrac{1}{2}\log_2(1-\alpha_{\mathrm{d}}^2), & B&=-\tfrac{1}{2}\log_2(1-\alpha_{\mathrm{q}}^2),\\ \Delta&=1-\frac{\alpha_{\mathrm{d}}^2+\alpha_{\mathrm{q}}^2-2\rho\alpha_{\mathrm{d}}\alpha_{\mathrm{q}}}{1-\rho^2}, & F&=-\tfrac{1}{2}\log_2\Delta. \end{aligned} \tag{8.7}$$

In this region $\alpha_{\mathrm{d}}>\rho\alpha_{\mathrm{q}}$, $\alpha_{\mathrm{q}}>\rho\alpha_{\mathrm{d}}$ and $\Delta>0$. The two inequalities mean that the minimum-norm vector satisfying the two cap constraints saturates both constraints. The positive Δ gives a nonempty

intersection of the caps. Eligible pairs have correlation at least $\rho - \eta_n$, for a specified $\eta_n \geq 0$ with $\eta_n = o(1)$. For a fresh uniform $\mathbf{w}$, let $D(\mathbf{w})$ and $Q(\mathbf{w})$ count data and query directions in their respective caps. Assume

$$\mathbb{E}D(\mathbf{w}) \leq 2^{(u-A)n+o(n)}, \quad \mathbb{E}Q(\mathbf{w}) \leq 2^{(\pi-B)n+o(n)}, \quad \mathbb{E}[D(\mathbf{w})Q(\mathbf{w})] \leq 2^{(\pi+u-A-B)n+o(n)}. \tag{8.8}$$

There is a data index with mean construction and storage rate $\max\{u, I\}$, where

$$I = u + F - A, \qquad H = \pi + F - B, \qquad N = \pi + u + F - A - B. \tag{8.9}$$

It extends the query tree with $2^{o(n)}$ reversible navigation work per edge and polynomial depth. The mean numbers of additional query–cap vertices and candidate visits have rates H and N, respectively; the query vertices themselves are also charged. The bound includes empty cap lists and repeated visits. Every fixed eligible pair is covered with failure at most 2^{-n^2}. Choosing the first common address lets each pair of occurrences continue at most once.

The code is independent of the families. Expectations in (8.8) may include the indicator of an event fixed before the code is drawn. The same exponential rates hold for tree depth and data-rank bit lengths $2^{o(n)}$, provided navigation still costs $2^{o(n)}$ per edge.

Proof. Use the block product construction of Lemma 2.17. With $a_0 = \lceil \log_2(n+2) \rceil$ blocks of dimension $k_0 = \lceil n/a_0 \rceil$, pad to $N_0 = a_0 k_0$ coordinates. Put $\rho_n = \rho - \eta_n - Cn^{-1/4}$, for a sufficiently large fixed C, and let Δ_n be (8.7) with ρ_n in place of ρ. In each of n^2 independent rotated families, store in every block

$$M = \left\lceil \Delta_n^{-k_0/2} 2^{N_0^{9/10}/a_0} \right\rceil \tag{8.10}$$

uniform unit directions. A global address selects one per block. Its local thresholds are $\alpha_{\mathrm{d}}/\sqrt{a_0}$ and $\alpha_{\mathrm{q}}/\sqrt{a_0}$. The two-vector Gram calculation gives common-cap factor Δ_n. After a random rotation, block norms and correlations are accurate to $O(n^{-1/4})$ except with probability $o(1)$; the margin in ρ_n absorbs these errors. The slack makes a common local choice exist in every block with probability at least $1/2$ per family. The n^2 families give the stated coverage. Since $\Delta_n = \Delta - o(1)$, all rates change by $o(1)$. The conceptual number of addresses is $2^{Fn+o(n)}$, while the local dictionaries have size $2^{o(n)}$.

Local membership implies global cap membership. Each concatenated centre is marginally uniform after its independent rotation. Projecting away the padded coordinates only strengthens the cap tests relative to the normalized projected centre. Thus summing (8.8) over addresses gives (8.9). Only pairs of data occurrences and cap addresses are stored and sorted. A nonempty cap list is located by its offset and length. An address absent from the index has an empty cap list.

For a query, scan the local dictionaries to recompute its qualifying sets. Their Cartesian product gives the matching cap addresses. Every partial address extends unless a qualifying set is empty, which can be detected before traversal. Binary rank trees for these sets and for the corresponding cap list give polynomial depth. The tree size is at most $2^{o(n)}$ times one plus the number of matching addresses and candidate visits. A navigation step recomputes local sets and performs addressed reads in $2^{o(n)}$ work. No query incidence array is constructed. Finally, local intersection scans find the first common address of a fixed pair. Only this first visit can pass the acceptance test or extend the search. The acceptance test uses the random value assigned to that pair. □

Lemma 8.4 (Quantum reporting on an implicit tree)**.** *Let a fixed binary tree have S vertices, depth $2^{o(n)}$ and D_{out} qualifying leaves. Suppose we can navigate its edges and test whether a leaf qualifies*

reversibly in $2^{o(n)}$ work. All qualifying leaves can be reported with bounded error in $2^{o(n)}\sqrt{S(D_{\mathrm{out}}+1)}$ work, apart from separately charged index construction and output writes. One qualifying leaf can be found, or their absence established, in $2^{o(n)}\sqrt{S}$ work. The tree size need not be known beforehand. Auxiliary workspace is $2^{o(n)}$, apart from the stored access data and reported output records.

Proof. Quantum backtracking can find a qualifying leaf in a subtree or verify that none exists, in work proportional to the square root of its size up to subexponential factors [Mon18, Theorem 1.2 and Section 2.3]. A subtree of size S_v therefore costs $2^{o(n)}\sqrt{S_v}$. Test subtrees depth first and descend after positive answers. With correct tests, at most $2D_{\mathrm{out}}(h+1)+1$ subtrees are tested, where h is the depth. Each vertex has at most $h+1$ tested ancestors. Consequently

$$\sum_v \sqrt{S_v} \le \sqrt{(2D_{\mathrm{out}}(h+1)+1)S(h+1)}. \tag{8.11}$$

Amplification absorbs all detector and size-estimation errors. The same backtracking guarantee bounds the work to find one leaf and the auxiliary space. Its polynomial dependence on depth also allows depth $2^{o(n)}$. □

8.3 The improved binary bound

Proof of Theorem 8.1. Use the list construction of Lemma 8.2. At an internal merge, write $\nu = \nu_d$, $v = 1+\nu/4$, $l = \frac{1}{2}\log_2 v$ and $m = \frac{1}{2}\log_2 \nu$. In Lemma 8.3, take $u = \pi = l$, target $\rho = 2/v - 1$, and

$$\tau = \tau_\star, \qquad \alpha_{\mathrm{d}}^2 = \frac{v-1}{v}, \qquad \alpha_{\mathrm{q}} = \tau\alpha_{\mathrm{d}}, \qquad \Delta = (1+\tau)\left(1 - \frac{v(1+\tau)}{4}\right). \tag{8.12}$$

These parameters stay in a compact strict two-active-cap region. The cap-load assumptions follow from the upper first moments: averaging the assigned parent shift and its fresh split shift makes the two child shifts independent and uniform. The expected counts of points in caps then follow by integrating over the corresponding shell regions. We only need independence between the two child lists given their shifts.

Here $A = l$, so the stored index and the implicit tree have rates

$$I = F = -\tfrac{1}{2}\log_2 \Delta, \qquad H = N = F + \tfrac{1}{2}\log_2\Big(v(1-\tau^2)+\tau^2\Big). \tag{8.13}$$

A candidate qualifies at its first common cap address if it passes the shell, midpoint, coset and acceptance tests. Each such leaf corresponds to a distinct output occurrence in the reference construction, giving $\mathbb{E}D_{\mathrm{out}} \le 2^{mn+o(n)}$. The tree has mean size at most $2^{Hn+o(n)}$, including query–cap vertices with empty data lists, since $H = N \ge l$. Lemma 8.4 and Cauchy–Schwarz give mean reporting work at most

$$\mathbb{E}\sqrt{S(D_{\mathrm{out}}+1)} \le \sqrt{\mathbb{E}S\,\mathbb{E}(D_{\mathrm{out}}+1)} \le 2^{(H+m)n/2+o(n)}. \tag{8.14}$$

Index construction, stored lists and output writing therefore give local time and space exponents at most

$$T(\nu) = \max\{m, l, F, (H+m)/2\}, \qquad S(\nu) = \max\{m, l, F\}. \tag{8.15}$$

Both increase with ν: m, l, F increase, and $v(1-\tau^2)+\tau^2$ increases because $\tau < 1$. Thus the cost at the shallowest internal merge bounds all internal calls.

At the limiting root geometry $\nu = a_\star$, put $G_\star = v_\star - (v_\star - 1)\tau_\star^2$. Equation (8.2) is $a_\star G_\star \Delta_\star = 1$, so substitution gives exactly

$$F = t_\star, \qquad H = F + \tfrac{1}{2}\log_2 G_\star, \qquad m = \tfrac{1}{2}\log_2 a_\star, \qquad \frac{H+m}{2} = t_\star. \tag{8.16}$$

Both m and l are smaller than $t_\star$. The defining polynomial is strictly decreasing on $(0.929, 0.930)$ and has opposite signs at the endpoints, proving existence and uniqueness there. On this interval F increases with τ and the reporting rate $(H+m)/2$ decreases; the equation balances the two costs. Directed evaluation gives $\tau_\star = 0.9292476653\ldots$ and the stated $t_\star$. The finite root parameter is $a = a_\star + o(1)$, so continuity adds only $o(1)$ to these rates at the same exact threshold ratio $\tau_\star$. There are $2^{o(n)}$ recursive calls. Leaf initialization has exponent $\frac{1}{2}\log_2(4/3)$; depth-first list storage and final minimum finding have exponent at most m_0, also below these bounds. First impose list limits with subexponential slack above their reference means. Markov's inequality makes any such failure have total probability $o(1)$. On this bounded recursion, address lengths and tree depths are polynomial; the stated index and work limits fail with probability $o(1)$ by the same mean bounds. Pointwise coverage and reporting errors are amplified to 2^{-n^2} before a union bound over the bounded lists and calls. Coupling to the reference construction gives a shortest-vector representation with probability $1 - o(1)$. Amplify the final minimum finding to obtain success at least $2/3$. Preparation, incorrect length guesses, numerical precision and the bounded-dimensional fallback are treated as in Section 6. □

Section 9 uses the same search procedure for trees of partial sums.

9 Quantum SVP in Time $2^{0.2652...n+o(n)}$ – Gaussian Splitting

Theorem 9.1 (Gaussian binary generation and a four-list root)**.** *Exact SVP on every input lattice admits a quantum algorithm with success probability at least* $2/3$ *and time and space*

$$\mathrm{T_Q} = \mathrm{S_Q} = (13/9)^{n/2+o(n)} = 2^{\tau_{\mathrm{Q}} n+o(n)}, \qquad \tau_{\mathrm{Q}} = \tfrac{1}{2}\log_2 \frac{13}{9} = 0.2652573583\ldots. \tag{9.1}$$

Each internal merge combines two lists. The final search uses four top lists. The analysis uses the centered lattice counts of Proposition 1.1 and the spherical-code bound of [Ope26, Chapter 2, Remark 1.7] for separated projected sets.

Section 8 uses pairs both to build the lists and to find a shortest vector in the final search. We keep the pair merges and make two changes.

Search four lists at the end. The extra lists give more ways to represent a fixed shortest vector, so each list can be smaller. This reduces the cost of every internal merge. The final search builds a partial sum by appending one vector at a time. The filters from Section 8 limit the number of extensions to explore. Generating the lists now determines the leading time exponent.

Use a Gaussian to weight representations. For a proposed sum, a conditional Gaussian assigns a weight to each way of splitting it among the children. Accepting a representation with probability proportional to its Gaussian density gives the desired expected count for each point on the output shell. We count every accepted representation during list generation, including different representations of the same vector. We remove repeated vectors only from the top lists, where agreeing in all but one coordinate of a tuple then forces the entire tuple to agree.

The proof needs bounds on the expected counts and on the dependence between two representations. We obtain these by preparing the chain so that Gaussian sums over its cosets have controlled variation. Conditioning on one or two coordinates at a time reduces the required bounds to low-dimensional covariance calculations. The final search also needs a separate estimate for the last child coset, whose shift is determined by the other shifts.

Roadmap. Section 9.1 gives the splitting rule and the final four-list search. Section 9.2 bounds Gaussian correlations on the superlattice chain. Section 9.3 derives the list moments, and Section 9.4 uses them to bound the probability of finding a shortest vector. Section 9.5 bounds the search work, including the last coset. Section 9.6 checks the parameter choices and proves the claimed bounds.

9.1 List construction by Gaussian splitting

Normalize the lattice to have determinant one and write $\lambda = \lambda_1(\mathcal{L})$. Use the shortest-length guesses of Section 5 with relative accuracy $O(n^{-3})$. The guess sets the overall scale σ_0 through $\lambda^2/(n\sigma_0^2) = \chi + O(n^{-3})$ for a correct guess. The normalized squared radius X determines the geometry of each binary split. The two point-counting bounds used in the preparation apply to centered lattice counts and to general separated sets, respectively. We use the exact constants

$$\begin{aligned} X &= 1.45, & \chi &= 0.639615, & \beta_{\mathrm{lat}} &= \tfrac{1}{2}\log_2(2e/\pi), \\ \beta_{\mathrm{sph}} &= 0.39661, & \beta_{\mathrm{cert}} &= 0.39662. \end{aligned} \tag{9.2}$$

The numerical checks use β_{cert} as an upper bound on β_{sph}. Section 9.6 checks the bounds for these choices.

To make the acceptance probabilities valid at finite n, we use a list-size exponent b_n slightly above its limit c. The shell radius decreases by the factor $f_n^{-1/2}$ at each depth, and γ_n controls how quickly the chain grows denser relative to the shells. Write η_n for the exponential part of the expected count per point and ω_n for the relative shell width. Set

$$c = \tfrac{1}{2}\log_2(4/3), \qquad \gamma_n = \frac{1}{\ln(n+e)}, \qquad f_n = 1+\gamma_n,$$
$$b_n = c + 2\gamma_n, \qquad \eta_n = b_n - \tfrac{1}{2}\log_2 X, \qquad \omega_n = n^{-1/4}. \tag{9.3}$$

For a fixed dimension write f, γ, b, η for the corresponding parameters without their subscripts. We work at sufficiently large dimensions. Bounded-dimensional cases are handled as in Section 5. Put

$$t_j = \frac{\sigma_0}{\sqrt{ef^j}}, \qquad \mathcal{V}_j = (\sqrt{2\pi e}\, t_j)^n, \qquad D = \lceil K \log_2(n+2)/\gamma_n \rceil. \tag{9.4}$$

The fixed constant K is chosen in the terminal-basis proof. Then $D = O((\ln n)^2)$, $f^{D/2} = n^{O(1)}$, and the expanded recursion has $2^{O((\ln n)^2)} = 2^{o(n)}$ calls. Lemma 9.4 constructs $\mathcal{L} \subseteq \Gamma_0 \subseteq \cdots \subseteq \Gamma_D$, with $\Delta_j = \det \Gamma_j$ and

$$\delta_j = \Delta_j/\mathcal{V}_j, \qquad a_j = \delta_j 2^{\eta n}, \qquad H_j = \left\{ \mathbf{x} : \left| \frac{\|\mathbf{x}\|^2}{nXt_j^2} - 1 \right| \le \omega_n \right\}. \tag{9.5}$$

A depth-j list has target occurrence weight $a_j \mathbf{1}_{H_j}$ and expected size $2^{bn+o(n)}$. Its physical radius is $\sqrt{nX}\, t_j$; this decreases by the factor $f^{-1/2}$ at each depth.

The splitting rule also applies to later constructions with more than two children. We call a merge that builds a list a *source merge* to distinguish it from the final search. Such a merge takes $k \ge 2$ child lists on prescribed cosets of Γ_{c} and generates a parent list on a coset of $\Gamma_{\mathrm{o}} \subseteq \Gamma_{\mathrm{c}}$. In units of the child scale t_{c}^2, choose a positive definite covariance Σ with diagonal $X_1, \dots, X_k$, and put

$$\mathbf{s} = \Sigma \mathbf{1}, \qquad P = \mathbf{1}^T \Sigma \mathbf{1}, \qquad R = \Sigma - \frac{\mathbf{s}\mathbf{s}^T}{P}. \tag{9.6}$$

The parent shell has squared radius nPt_{c}^2. Choose the first $k-1$ child shifts independently and uniformly modulo Γ_{c}. The output coset determines the sum of all shifts modulo Γ_{c}, so it fixes the last child shift. Given the chain and all the child cosets, the calls use independent randomness.

For a prescribed parent $\mathbf{z}$, let $g_{\mathbf{z}}$ be the normalized Gaussian density of the first $k-1$ children given their sum $\mathbf{z}$. Its means are $(s_i/P)\mathbf{z}$, and its free-coordinate covariance is the corresponding principal minor of $t_{\mathrm{c}}^2 R$. If the incoming point coefficients are $a_1, \dots, a_k$ and the desired parent coefficient is a_{o}, accept a proposed ordered occurrence tuple with probability

$$p_{\mathbf{z}}(\mathbf{x}_1, \dots, \mathbf{x}_k) = \frac{a_{\mathrm{o}} \Delta_{\mathrm{c}}^{k-1}}{\prod_i a_i}\, g_{\mathbf{z}}(\mathbf{x}_1, \dots, \mathbf{x}_{k-1}) G_{\mathbf{z}}(\mathbf{x}_1, \dots, \mathbf{x}_k). \tag{9.7}$$

The factor $G_{mathbfz}$ applies the coset and shell tests. A Gram test also bounds the difference between each scaled inner product and its prescribed value $Sigma_{ih}$. A partial-sum test restricts the squared length of a sum of the first few children. These tests can be indicators or smooth cutoffs taking values in $[0,1]$. Their product weights the conditional Gaussian in the expected output count.

Lemma 9.2 (Averaging over the child cosets)**.** *For every parent* $\mathbf{z}$ *in its prescribed coset and every nonnegative integrable function* F*,*

$$\mathbb{E}_{\text{cosets}} \sum_{\substack{\mathbf{x}_i \in \Gamma_{\mathrm{c}} + \mathbf{c}_i \\ \sum_i \mathbf{x}_i = \mathbf{z}}} F(\mathbf{x}_1, \dots, \mathbf{x}_{k-1}) = \Delta_{\mathrm{c}}^{-(k-1)} \int F. \tag{9.8}$$

With the nominal expected counts in the child lists, the expected output count in (9.7) is exactly a_{o} times the integral of the conditional Gaussian weighted by the tests.

Proof. Averaging the sum over the first $k-1$ uniform shifts gives the integral with factor $\Delta_{\mathrm{c}}^{-(k-1)}$. The sum fixes the last vector, and its coset was chosen to contain it. The change of coordinates from the first $k-1$ vectors and their sum to the full tuple has absolute determinant one. Multiplying the expected child counts by the acceptance probability cancels $\Delta_{\mathrm{c}}^{-(k-1)} \prod_i a_i$, leaving $a_{\mathrm{o}} g_{\mathbf{z}} G_{\mathbf{z}}$. □

Write $\ell_i = \eta_i + \frac{1}{2}\log_2 X_i$ for the child list-size exponents. If the parent scale satisfies $t_{\mathrm{o}}^2 = f_{\mathrm{o}} t_{\mathrm{c}}^2$ and its shell variance is $X_{\mathrm{o}} = P/f_{\mathrm{o}}$, use $a_{\mathrm{o}} = (\Delta_{\mathrm{o}}/\mathcal{V}_{\mathrm{o}}) 2^{\eta_{\mathrm{o}} n}$. For tuples satisfying the Gram test, the base-two logarithm of the acceptance probability divided by n is

$$\ell_{\mathrm{o}} - \sum_i \ell_i - \tfrac{1}{2}\log_2 \det G + \frac{1}{n}\log_2(\Delta_{\mathrm{o}}/\Delta_{\mathrm{c}}) + o(1), \qquad G = \operatorname{diag}(X_i)^{-1/2} \Sigma \operatorname{diag}(X_i)^{-1/2}. \tag{9.9}$$

Indeed $\det R_{\mathrm{free}} = \det \Sigma / P$; evaluating the density under the Gram test in (9.7) gives this expression. The scale factors f_{o} cancel. The acceptance probability is at most one whenever the negative leading term dominates the $O(\omega_n)$ error from the Gram test.

Assign each ordered tuple of child occurrences a single address and a single random value, even if several filters report it. Pairwise independent uniform values suffice. Draw independent uniform U, V in $\mathbb{R}/\mathbb{Z}$ per call and assign the value $eU + V$ to integer address e. For eee', the integer torus map with rows $(e,1),(e',1)$ is surjective, so the two assigned values are independent and uniform. The filter codes and these values are independent of the child calls. We count different tuples with the same output as distinct occurrences.

For the binary source in this section, choose one fresh split coset and put

$$\Sigma = X \begin{pmatrix} 1 & f/2-1 \\ f/2-1 & 1 \end{pmatrix}, \qquad P = fX, \qquad p_\star = X(1-f/4) \longrightarrow 1.0875. \tag{9.10}$$

Thus $\mathbf{x}_1 = \mathbf{z}/2 + \mathbf{e}$ and $\mathbf{x}_2 = \mathbf{z}/2 - \mathbf{e}$, with $\mathbf{e} \sim N(\mathbf{0}, p_\star t_{j+1}^2 I_n)$. Use (9.7) with $a_{\mathrm{o}} = a_j$, $a_1 = a_2 = a_{j+1}$, the shells H_j, H_{j+1}, and the entrywise Gram tests $|\langle \mathbf{x}_i, \mathbf{x}_h\rangle/(n t_{j+1}^2) - \Sigma_{ih}| \le X\omega_n$. The acceptance probability has exponent $-\kappa_n + o(1)$, where

$$\begin{aligned} \kappa_n &= M_\star - \gamma_n, \\ M_\star &= \eta + \tfrac{1}{2}\log_2 p_\star = b + \tfrac{1}{2}\log_2(1-f/4), \\ \kappa_n &= \gamma_n + \tfrac{1}{2}\log_2(1-\gamma_n/3) = \left(1 - \frac{1}{6\ln 2} + o(1)\right)\gamma_n. \end{aligned} \tag{9.11}$$

Thus $\kappa_n > \frac{3}{4}\gamma_n$ for sufficiently large n. The margin dominates the uniform $O(\omega_n) + O(\log_2 n/n)$ error in the acceptance exponent.

At depth D, use the uniform-shell Babai initializer proved below. To form the root, choose three independent Haar cosets modulo Γ_0, choose the fourth so that the shifts sum to zero modulo Γ_0, and generate the four lists with independent randomness given their cosets. Remove repeated vector values only in these top lists. Put $C = X/e$ and $V = 0.76$. In σ_0^2 units choose

$$Q_{ii} = C, \qquad Q_{ih} = (V - 3C)/6 \quad (i \ne h \le 3), \qquad Q_{i4} = Q_{4i} = (\chi - V - C)/6 \quad (i \le 3). \tag{9.12}$$

The four prefix variances are

$$y_1 = C, \qquad y_2 = C + V/3, \qquad y_3 = V, \qquad y_4 = \chi. \tag{9.13}$$

The search appends one top vector at a time, testing prefix shells and Gram conditions before advancing. A final tuple qualifies if its sum is nonzero, lies in $\mathcal{L}$ and has squared norm below the search threshold.

The constants describe a feasible instance of these inequalities. The variance X controls both the limiting conditional variance $3X/4$ of a binary split and the diagonal of the root covariance. The total root variance χ controls the preparation margins and the required shortest-vector length, while V sets the penultimate prefix radius. The exchangeability of the first three root lists leaves only this one intermediate variance to specify. Section 9.6 checks the preparation and search bounds and the probability of finding a shortest-vector representation.

9.2 Gaussian correlation bounds

Unlike the centered counts sufficient for the earlier binary midpoint shells, the present preparation also counts separated projected shells. We use the two distinct counting exponents in (9.2): the refined centered lattice bound follows from the Fourier estimates of [Ope26, Kom26], and the general spherical-code bound of [Ope26] controls arbitrary separated sets in the projected shells.

Lemma 9.3 (Separated sets and lattice sections)**.** *Every rank-m lattice Λ of minimum at least λ satisfies*

$$\begin{aligned} N_\Lambda(R\lambda) &\le 2^{(\beta_{\mathrm{lat}}+o(1))m}R^m \quad (R\ge 1),\\ \det\Lambda &\ge \mathrm{vol}(B_m)\lambda^m 2^{-(\beta_{\mathrm{lat}}+o(1))m}. \end{aligned} \tag{9.14}$$

For translated balls and finite λ-separated sets, the corresponding bound is $2^{(\beta_{\mathrm{sph}}+o(1))m}R^m$.

Proof. The centered count is Proposition 1.1 in the span of Λ. Divide by R^m and let $R\to\infty$; the normalized count tends to $\mathrm{vol}(B_m)\lambda^m/\det\Lambda$, proving the determinant bound. For separated sets normalize $\lambda=1$ and lift points of a unit ball to the sphere in one extra coordinate by $\mathbf{x}\mapsto(\mathbf{x},\sqrt{1-\|\mathbf{x}\|^2})$. Distances do not decrease, giving an angle-$\pi/3$ code. For larger radii, apply the radial-shell and cap-averaging argument of Lemma 2.1. It uses only separation, so it also applies to finite separated sets and to translated balls. The polynomial radial and cap factors are absorbed in $o(m)$. The general bound $A(m,1/2)\le 2^{(0.39661+o(1))m}$ [Ope26, Chapter 2, Remark 1.7] supplies the stated exponent. In particular the determinant bound applies to every rational section $\mathcal{L}\cap W$. □

Lemma 9.4 (A chain at a fixed or decreasing physical scale)**.** *Allow $f=f(n)\ge 1$ and $\gamma=\gamma(n)>0$, with $\gamma=O(1)$, $\gamma^{-1}=O(\ln n)$ and $\ln f=O(\gamma)$; fixed positive parameters are included. Take $D=\lceil K\log_2(n+2)/\gamma\rceil$. For any sufficiently large fixed C_0, one can construct*

$$\begin{aligned} \Gamma_0&=\mathcal{L}+\mathbb{Z}\frac{\mathbf{B}\bar{\mathbf{z}}_p}{p}, & \Delta_0&=\frac{\det\mathcal{L}}{p}, & \delta_0&=\Delta_0/\mathcal{V}_0=\Theta(n^{-C_0}),\\ \Gamma_j&=\Gamma_0+\mathbb{Z}\frac{\mathbf{B}_0\bar{\mathbf{z}}_q}{q^j}, & \Delta_j&=\Delta_0 q^{-j}, & \delta_j&=\delta_0(f^{n/2}/q)^j, \end{aligned} \tag{9.15}$$

where p,q are primes, $\log_2 q=(\frac12\log_2 f+\gamma)n+O(1)$, $\mathbf{z}_p$ is uniform nonzero modulo p, and $\mathbf{z}_q=(1,z_2,\dots,z_n)$ has its last coordinates uniform modulo q^D, independently of $\mathbf{z}_p$. The construction works uniformly over the allowed prime choices.

Proof. Lemma 9.3 and Stirling's formula give

$$\frac{\det\mathcal{L}}{\mathcal{V}_0}\ge 2^{(\frac12\log_2(e\chi)-\beta_{\mathrm{lat}}-o(1))n}, \qquad \tfrac12\log_2(e\chi)-\beta_{\mathrm{lat}}>0.00338590. \tag{9.16}$$

Choose p in a constant-factor interval around $\det \mathcal{L}/(n^{-C_0}\mathcal{V}_0)$, and q around $2^{(\frac{1}{2}\log_2 f+\gamma)n}$, using the prime-extension procedure of Section 5. Fix the primes before choosing their directions. The nonzero line modulo p has index p. The first coordinate of $\mathbf{z}_q$ gives order exactly q^j modulo q^j, proving the determinant identities. The volume relation $\mathcal{V}_j = \mathcal{V}_0 f^{-jn/2}$ proves the last identity. In particular $\delta_j = \delta_0\, 2^{-\gamma jn+O(j)}$. Also $\ln(t_0/t_j) \le D\ln f/2 = O(\ln n)$ uniformly over these schedules. □

For a positive definite $d\times d$ coefficient covariance C, let $\varphi_{j,C}$ be the density of independent rows with covariance $t_j^2 C$. Choose a smooth cutoff ψ_C equal to one on an inner Gram window and zero outside a larger window. We call the normalized Gaussian restricted by this cutoff a *broad reference* and define

$$h_{j,C} = \varphi_{j,C}\psi_C/\mu_C, \qquad \mu_C = \int \varphi_{j,C}\psi_C = 1 - e^{-\Omega(\sqrt{n})}. \tag{9.17}$$

The inner and outer Gram windows have widths that are fixed multiples of ω_n. They are chosen with successive fixed margins. Coefficient dimensions are fixed, and all covariance eigenvalues and inverse eigenvalues stay uniformly bounded. Gaussian concentration makes the mass estimate uniform over such families. For any normalized density h put

$$\begin{aligned}\operatorname{Corr}(h)(\mathbf{w}) &= \int h(\mathbf{x})h(\mathbf{x}+\mathbf{w})\,d\mathbf{x},\\ \mathcal{R}_{\Gamma^d}(h) &= (\det\Gamma)^d \sum_{\mathbf{w}\in\Gamma^d} \operatorname{Corr}(h)(\mathbf{w}), \qquad \mathcal{P}_{\mathcal{L},j}(h) = \mathcal{V}_j^d \sum_{\mathbf{w}\in\mathcal{L}^d} \operatorname{Corr}(h)(\mathbf{w}).\end{aligned} \tag{9.18}$$

Lemma 9.5 (Phases and ordinary integer pushforwards)**.** *For a smooth compactly supported probability density h, Poisson summation gives*

$$\begin{aligned}\mathcal{R}_{\Gamma^d}(h) &= \sum_{\boldsymbol{\xi}\in(\Gamma^*)^d} |\widehat{h}(\boldsymbol{\xi})|^2 \ge 1,\\ (\det\Gamma)^d \sum_{\mathbf{w}\in\Gamma^d} \operatorname{Corr}(h)(\mathbf{w}+\mathbf{u}) &\le \mathcal{R}_{\Gamma^d}(h) \qquad (\mathbf{u}\in\mathbb{R}^{n\times d}).\end{aligned} \tag{9.19}$$

If $U\in\mathbb{Z}^{d\times r}$ has full column rank and h_U is the ordinary density pushforward under $\mathbf{x}\mapsto\mathbf{x}U$, then $\mathcal{R}_{\Gamma^r}(h_U) \le \mathcal{R}_{\Gamma^d}(h)$, also with real phases.

Proof. The Fourier transform of the autocorrelation is $|\widehat{h}|^2$. Its zero coefficient is one; removing all Fourier phases proves the second inequality. Moreover $\widehat{h_U}(\boldsymbol{\xi}) = \widehat{h}(\boldsymbol{\xi}U^T)$. The map $\boldsymbol{\xi}\mapsto\boldsymbol{\xi}U^T$ injects the projected dual lattice into the original one, so the Fourier sum for h_U is a subseries of the sum for h. □

For the physical bounds, fix a depth j and suppress its subscripts on $t, \mathcal{V}, h_C$. Put $A = e\chi/4$ and $\rho_\star = 0.4999999999$. The power $\rho_\star$ is an exact conservative choice below the natural volume power $1/2$, used in the interval bound for the overlap. The actual coefficient $\lambda^2/(4nt_j^2)$ is $Af^j(1+O(n^{-3})) \ge A - o(1)$; replacing it by A only enlarges the overlap bounds below.

Lemma 9.6 (Overlap of Gaussian densities restricted to Gram windows)**.** *For $Q = \mathbf{w}^T\mathbf{w}/\lambda^2$, broad references satisfy*

$$\mathcal{V}^d \operatorname{Corr}(h_C)(\mathbf{w}) \le 2^{o(n)} \det(C)^{-n/2} \det\Bigl(I - AC^{-1/2}QC^{-1/2} + O(\omega_n)I\Bigr)^{\rho_\star n} \tag{9.20}$$

on their overlap support. Exponential maxima may use the unperturbed positive domain, with an $o(1)$ change in rate.

Proof. Set $\mathbf{y} = \mathbf{x} + \mathbf{w}/2$. The two Gram constraints imply

$$\frac{\mathbf{y}^T\mathbf{y}}{nt^2} = C - \frac{\mathbf{w}^T\mathbf{w}}{4nt^2} + O(\omega_n)C. \tag{9.21}$$

The product of the two Gaussian factors has exponent $-nd+O(n\omega_n)$ on this intersection. Discarding all remaining constraints increases its volume. For sufficiently large fixed c, this volume is bounded by

$$\det(C - AQ + c\omega_n C)^{n/2}\big(\mathrm{vol}(B_n)(\sqrt{n}\,t)^n\big)^d = 2^{O(\log_2 n)}\mathcal{V}^d \det(C - AQ + c\omega_n C)^{n/2}. \tag{9.22}$$

Indeed apply the inverse square root of the displayed coefficient matrix and enclose its columns in balls of radius $\sqrt{n}\,t$. Combining the Gaussian normalizers and the window normalizers proves the bound with power $n/2$. The determinant after conjugation is at most $(1 + O(\omega_n))^d$, so the slightly smaller power $\rho_\star n$ also gives a valid bound up to $2^{o(n)}$. Fixed covariance dimension and compact positive-definite domains justify the vanishing enlargement, including at the zero-overlap boundary. □

Lemma 9.7 (Scalar preparation)**.** *Define*

$$F_{\mathrm{lat}} = \beta_{\mathrm{lat}} + \rho_\star \log_2(2\rho_\star) - (\rho_\star + \tfrac{1}{2})\log_2(1 + 2\rho_\star) - \tfrac{1}{2}\log_2 A, \quad H_1(a) = \max\{0, -\tfrac{1}{2}\log_2 a\}. \tag{9.23}$$

For the certified pair bounds below write $F_{\mathrm{cert}} = F_{\mathrm{lat}} + \beta_{\mathrm{cert}} - \beta_{\mathrm{lat}}$. Then $F_{\mathrm{cert}} < -0.00236535926728$. For every scalar variance $a > 0$,

$$\mathcal{P}_{\mathcal{L},j}(h_a) \le 2^{(\max\{F_{\mathrm{lat}}, -\frac{1}{2}\log_2 a\} + o(1))n} \le 2^{(H_1(a)+o(1))n}. \tag{9.24}$$

The exponent is strictly negative if $a > 1$ by a fixed margin. Here $F_{\mathrm{lat}} < -0.00338590$.

Proof. The zero displacement has exponent $-\frac{1}{2}\log_2 a$. For a nonzero displacement set $u = \|\mathbf{w}\|^2/\lambda^2$. Lemma 9.3 and (9.20) bound its radial sum by the maximum of $\beta_{\mathrm{lat}} + \frac{1}{2}\log_2 u - \frac{1}{2}\log_2 a + \rho_\star \log_2(1 - Au/a)$. Enlarge the maximum to $0 < Au/a < 1$. Its maximizer is $Au/a = (1 + 2\rho_\star)^{-1}$ and its value is F_{lat}. Inverse-polynomial radial cells and the vanishing window enlargement cost only $2^{o(n)}$. These estimates are uniform over all depths because the actual overlap coefficient is at least $A - o(1)$. □

The newer spherical-code bound enters quantitatively through F_{cert}. At the chosen χ, replacing β_{cert} by the classical exponent $0.4009\ldots$ of [KL78] would make this radial upper-bound rate positive. The all-radius centered lattice estimate and the spherical-code estimate serve different roles.

Lemma 9.8 (Pair preparation)**.** *For positive definite $C \in \mathbb{R}^{2\times 2}$ put*

$$m_{\mathbb{Z}}(C) = \min_{\mathbf{z}\in\mathbb{Z}^2\setminus\{\mathbf{0}\}} \mathbf{z}^T C\mathbf{z}, \qquad H_2(C) = \max\{0, -\tfrac{1}{2}\log_2\det C, -\tfrac{1}{2}\log_2 m_{\mathbb{Z}}(C)\}. \tag{9.25}$$

Then $\mathcal{P}_{\mathcal{L},j}(h_C) \le 2^{(H_2(C)+o(1))n}$. The same allowance bounds the actual ordinary rank-one integer pushforwards. If both $\det C$ and $m_{\mathbb{Z}}(C)$ exceed one by fixed margins, all these physical exponents are strictly negative.

Proof. Apply Gauss reduction in an integer unimodular chart and write $C = \left(\begin{smallmatrix} a & c \\ c & v \end{smallmatrix}\right)$, where $a = m_{\mathbb{Z}}(C) \le v$, $2|c| \le a$, and $\mathfrak{d} = \det C$. Conditioning on the first coordinate gives scalar variances a and $\mathfrak{d}/a$. If $a \le 1$, their two bounds from Lemma 9.7 sum to at most $H_2(C)$. The same holds if both variances exceed one. Lemma 9.10 combines the bounds on the two scalar sums. The remaining

case is $a > 1$ and $\mathfrak{d}/a \le 1$. It gives $1 < a \le v \le 1 + a/4 \le 4/3$ and $\mathfrak{d} > 3/4$. The smaller conditional variance $p = \mathfrak{d}/v$ satisfies

$$p \ge 1 - \frac{1}{4v}, \qquad\qquad p \ge \frac{4\mathfrak{d}}{4\mathfrak{d}+1}. \tag{9.26}$$

For rank-two displacements write $Q = \left(\begin{smallmatrix} u & h \\ h & w \end{smallmatrix}\right)$, ordering the columns so that $u \ge w \ge 1$ and permuting C accordingly. In this ordering denote its diagonal entries by a, v and its off-diagonal entry by c. Both diagonal entries lie in $(1, 4/3]$ and $2|c| \le \min(a, v)$. The two bounds in (9.26) still hold: $a - c^2/v \ge a - a^2/(4v) \ge 1 - 1/(4v)$, since the last quadratic increases over $1 \le a \le 4/3$ and $v > 1$. Multiplying by v gives $v \le \mathfrak{d} + 1/4$. Separation gives $|h| \le (u + w - 1)/2$. On the overlap support $u, w \le U = 4/(3A)$, and hence the projected squared radius satisfies

$$r = u - h^2/w \ge \min\{3/4, U - U^2/4\} > 0.715107 > 0.7. \tag{9.27}$$

To obtain this bound, insert the bound on $|h|$ and use concavity successively on $1 \le w \le u \le U$. Direct expansion gives

$$\det(C - AQ) = (v - Aw)(p - Ar) - Avw(h/w - c/v)^2 \le (v - Aw)(p - Ar). \tag{9.28}$$

For the interval calculation we allow the weaker exponent β_{cert} also for the first lattice vector. Counting it and maximizing its radial contribution costs at most F_{cert}. Restrict its norm and its inner product with the second vector to inverse-polynomial cells. Projection of the second cell has separation $1 - o(1)$, since the removed coordinate of a difference is $o(1)$. If $r \ge 1$, Lemma 9.3 gives the same bound for the second contribution. For $0.7 \le r \le 1$, use the general-angle spherical-code bound [KL78] with exponent

$$\mathcal{K}(r) = \frac{(1+z)\ln(1+z) - z\ln z}{\ln 2}, \qquad z = \tfrac{1}{2}((1-s^2)^{-1/2} - 1), \qquad s = 1 - \frac{1}{2r}. \tag{9.29}$$

Writing $h_r(p) = \rho_\star \log_2(1 - Ar/p) - \frac{1}{2}\log_2 p$, the required strict one-variable inequality is

$$F_{\mathrm{cert}} + \sup_{0.7 \le r \le 1} \left\{ \min\{\beta_{\mathrm{cert}}, \mathcal{K}(r)\} + \max_{p \ge 4/5} h_r(p) \right\} < 0. \tag{9.30}$$

For completeness it has a finite rational interval verification. Start with $[7/10, 1]$. On $[l, h]$ upper-bound the expression by $F_{\mathrm{cert}} + \min\{\beta_{\mathrm{cert}}, \mathcal{K}(h)\} + h_l(\max\{4/5, (1+2\rho_\star)Al\})$. With directed interval arithmetic, bisect cells whose upper endpoint is nonnegative. This gives 993 accepted cells, maximum depth 19, and upper bounds below $-3.84 \cdot 10^{-9}$. Square roots can be enclosed by bisection and logarithms by range reduction and $\ln x = 2\sum_{j \ge 0} y^{2j+1}/(2j+1)$, $y = (x-1)/(x+1)$, with its geometric tail bound. Thus the verification requires only rational inequalities and convergent series with explicit tails. The supplied `scripts/verify_quantum_sections.py -section pair` reproduces this interval cover. For the first lattice vector we can use the centered rate F_{lat}, which gives an additional margin $\beta_{\mathrm{cert}} - \beta_{\mathrm{lat}} > 0.001$ in the rank-two estimate. The certificate keeps the conservative estimate to verify the same bound for both factors at $r \ge 1$.

This proves the desired strict bound for $\mathfrak{d} \ge 1$. For $3/4 < \mathfrak{d} < 1$, replace $p \ge 4/5$ by $p \ge 4\mathfrak{d}/(4\mathfrak{d}+1)$. The corresponding maximum plus $\frac{1}{2}\log_2 \mathfrak{d}$ is increasing in $\mathfrak{d}$: in the active case $-\frac{1}{2} \le dh_r/d\log_2 p \le 0$ and $d\log_2(4\mathfrak{d}/(4\mathfrak{d}+1))/d\log_2 \mathfrak{d} = 1/(4\mathfrak{d}+1)$. The rank-two contribution is therefore below $-\frac{1}{2}\log_2 \mathfrak{d}$ by a fixed margin.

The zero displacement has exponent $-\frac{1}{2}\log_2 \mathfrak{d}$. A rank-one displacement has columns $z_1\mathbf{w}, z_2\mathbf{w}$ for primitive $\mathbf{w}$ and integral $\mathbf{z} \ne 0$. The determinant lemma and the scalar maximum give exponent

at most $F_{\mathrm{cert}} - \frac{1}{2}\log_2(\mathfrak{d}\,\mathbf{z}^T C^{-1}\mathbf{z}) \le F_{\mathrm{cert}} - \frac{1}{2}\log_2 a$, where $a = m_{\mathbb{Z}}(C)$ refers to the reduced form. Here $\min_{\mathbf{z}\neq 0}\mathbf{z}^T C^{-1}\mathbf{z} = a/\mathfrak{d}$, because the adjugate of a binary symmetric form is its unimodular quarter-turn. Only boundedly many coefficient pairs meet the overlap support. These three ranks prove the physical bound.

Finally, Gaussian disintegration of an ordinary integer pushforward in direction $\mathbf{z}$ gives variance $\mathbf{z}^T C\mathbf{z} \ge a$. Its actual normalized density is bounded by a broader reference at this variance times a normalizer ratio $1 + e^{-\Omega(\sqrt{n})}$. Lemma 9.7 proves the pushforward assertion. The strict margins give the final statement. □

Lemma 9.9 (Dual separation and Fourier-rank averaging)**.** *Put* $K_j = \Gamma_j^*$ *and* $\Theta_K(s) = \sum_{\mathbf{w}\in K} e^{-2\pi^2 s^2\|\mathbf{w}\|^2}$. *With probability* $1 - o(1)$ *the event* $E_0 = \{\Theta_{K_0}(\sigma_0) \le 2\}$ *holds. On it*

$$\lambda_1(K_0) \ge \frac{\sqrt{\ln 2}}{\sqrt{2}\pi\sigma_0}, \qquad \sigma_0\lambda_1(p\mathcal{L}^*) \ge 2^{\Omega(n)}/\operatorname{poly}(n). \tag{9.31}$$

For a broad reference in fixed coefficient dimension d*, there are polynomial-size families* $\mathcal{U}_r$ *of integer* $d \times r$ *matrices whose rows generate* $\mathbb{Z}^r$ *such that*

$$\mathbb{E}\Big[\mathbf{1}_{E_0}(\mathcal{R}_{\Gamma_j^d}(h_{j,C}) - 1)\Big] \le \sum_{r=1}^{d} p^{-r} q^{-rj} (\det\mathcal{L})^r \sum_{U\in\mathcal{U}_r}\sum_{\mathbf{w}\in\mathcal{L}^r} \operatorname{Corr}((h_{j,C})_U)(\mathbf{w}) + e^{-\Omega(n\log_2 n)}. \tag{9.32}$$

Proof. Let $S = \mathcal{L}^* \cap W$ be the saturation of a rank-r span. Complementary Gram determinants give $\det S = \det(\mathcal{L}\cap W^\perp)/\det\mathcal{L}$. Lemma 9.3 and the first-index choice imply

$$p\det S \ge \sigma_0^{-r} 2^{(\frac{1}{2}\log_2(e\chi) - \beta_{\mathrm{lat}} - o(1))n}/\operatorname{poly}(n). \tag{9.33}$$

Consequently any fixed-rank independent tuple in $\mathcal{L}^*$ of lengths at most $n^{O(1)}/\sigma_0$ has full rank modulo p: otherwise its index in S is divisible by p, contradicting Hadamard's inequality. Its probability of surviving the first extension is $(p^{n-r}-1)/(p^n-1) < p^{-r}$. For rank one, (9.33) also gives the second inequality in (9.31). The remaining frequencies of the theta sum survive with probability at most p^{-1}. Poisson summation and the Gaussian mass estimate of Section 5 therefore give

$$\mathbb{E}[\Theta_{K_0}(\sigma_0) - 1] \le \delta_0 \sum_{\mathbf{x}\in\mathcal{L}} e^{-\|\mathbf{x}\|^2/(2\sigma_0^2)} + o(1) = O(\delta_0) + o(1). \tag{9.34}$$

The nonzero primal contribution has exponent at most $\beta_{\mathrm{lat}} - \frac{1}{2}\log_2(e\chi) < 0$. Markov's inequality proves E_0. Its two frequencies $\pm\mathbf{w}$ give the first separation bound.

Choose the smooth plateaus from a fixed Gevrey function. Their directional derivatives of order m have L^1 norm at most $n^{C_1 m} t_j^{-m}$, for $m = \lceil A_0 n\rceil$ and fixed sufficiently large A_0. Integration by parts and packing in the separated lattice K_0^d make the Fourier tail outside n^B/t_j at most $e^{-\Omega(n\log_2 n)}$ when B is sufficiently large. This holds uniformly over $j \le D$, since t_0/t_j is polynomial, and for every $K_j^d \subseteq K_0^d$.

For a rank-r frequency tuple within the cutoff, take a reduced basis $\mathbf{w}$ of its exact generated module and write the tuple as $\mathbf{w}U^T$. The rows of U generate $\mathbb{Z}^r$. Fixed-dimensional reduction, the frequency cutoff and dual separation give $\|U\|_\infty = n^{O(1)}$, leaving polynomially many choices. Using the exact generated module, rather than its saturation, makes membership of the original tuple equivalent to membership of these r basis vectors.

Their short independent coordinate rows in a basis of K_0 have full rank modulo the super-polynomial prime q. Otherwise their generated module would have volume at least q times its

saturation volume; packing in the rank-r section of K_0 contradicts their polynomial lengths. Write their coordinates as $[\mathbf{a}_1 \mid A']$. The remaining line condition is $A'\mathbf{z}' = -\mathbf{a}_1 \bmod q^j$. If A' has rank r modulo q, a unit minor gives probability q^{-rj}. Otherwise the full augmented rank makes the system inconsistent already modulo q. Combine this with the p^{-r} bound. Keep $\mathbf{1}_{E_0}$ through this separation argument, remove it only inside the resulting nonnegative sum, and apply Poisson summation to each ordinary pushforward. This proves (9.32). □

Lemma 9.10 (Scalar and pair Schur paths)**.** *Choose an integer change of coordinates of determinant* ± 1 *and an order in which to condition on successive blocks of one or two coordinates. We call this choice a* Schur path*. The resulting conditional covariance matrices are its Schur blocks* $C_1, \dots, C_m$*. With suitably nested cutoffs,*

$$\mathcal{R}_{\Gamma^d}(h_{j,C}) \le (1 + e^{-\Omega(\sqrt{n})}) \prod_{\nu=1}^{m} \mathcal{R}_{\Gamma^{\dim C_\nu}}(h_{j,C_\nu}). \tag{9.35}$$

For a density $\varphi_{j,C} G/Z$ *normalized by the total mass* Z*, the multiplier is at most* $Z^{-2} \prod_\nu \mu_{C_\nu}^2$ *if* $0 \le G \le 1$ *and* G *is supported on the corresponding block windows.*

Proof. Write $C = L \operatorname{diag}(C_\nu) L^T$ with real unit block-triangular L. The residual coordinates $\mathbf{e} = \mathbf{x} L^{-T}$ have Jacobian one and factor the Gaussian. The cutoff G implies the larger residual windows, so the restricted density is bounded by $(\prod_\nu \mu_{C_\nu}/Z) \prod_\nu h_{j,C_\nu}(\mathbf{e}_\nu)$. Use this in both autocorrelation factors and sum displacement blocks backwards. Each residual displacement is its lattice block plus a real shift determined by earlier blocks. Equation (9.19) removes that phase. The integer chart preserves the product lattice and the real shear changes only integration variables. Squaring the normalizer multiplier proves the claim. □

Lemma 9.11 (One common Gaussian preparation event)**.** *Fix a deterministic polynomial-size family of broad references in bounded coefficient dimension, with eigenvalues and inverse eigenvalues uniformly bounded. Matrices may depend on* n *within these compact domains. Specify a scalar/pair Schur path for each, and set*

$$\mathcal{H}(C) = \sum_{\dim C_\nu = 1} H_1(C_\nu) + \sum_{\dim C_\nu = 2} H_2(C_\nu). \tag{9.36}$$

For $\xi = \xi_n > 0$ *with* $\xi_n \gg \omega_n + \log_2 n/n$*, with probability* $1 - o(1)$ *over the chain, simultaneously for this family and* $0 \le j \le D$*,*

$$\mathcal{R}_{\Gamma_j^d}(h_{j,C}) \le 2^{(\mathcal{H}(C)+\xi)n}. \tag{9.37}$$

If every block allowance is zero, then at child depths $j \ge 1$ *the stronger bound is* $1 + e^{-\Omega(\sqrt{n})}$*. At* $j = 0$ *this stronger bound holds when every scalar variance exceeds one and every pair has determinant and integer minimum exceeding one, by fixed margins. All statements hold for ordinary integer pushforwards and real phases.*

Proof. Apply Lemma 9.9 separately to each scalar or pair block. Its density cancellation is

$$p^{-r} q^{-rj} \left(\frac{\det \mathcal{L}}{\mathcal{V}_j} \right)^r = \delta_j^r = \delta_0^r \, 2^{-r\gamma j n + O(j)}. \tag{9.38}$$

Lemmas 9.7 and 9.8 control the physical rank contributions. The pair's actual rank-one pushforwards are covered by Lemma 9.8; a primitive square rank-two map is unimodular. The vanishing slack

requires a quantified bound. The zero physical displacement contributes at most $2^{Hn+O(n\omega_n+\log_2 n)}$, without a lattice-count prefactor. Every nonzero scalar, pair or rank-one term has a fixed physical gap below H, from $F_{\mathrm{lat}} < 0$, the weaker $F_{\mathrm{cert}} < 0$, and (9.30). The universal $2^{o(n)}$ point-count prefactors are absorbed in this fixed gap. Thus each physical rank term is bounded by $2^{Hn+O(n\omega_n+\log_2 n)} + 2^{(H-\delta)n}$ for some fixed $\delta > 0$, uniformly on the compact covariance domains. Polynomially many integer charts add $O(\log_2 n)$. Markov at $2^{(H+\xi_n/m)n}$ makes failure $e^{-\Omega(n\xi_n)}$. For zero allowance at $j \geq 1$, the density factor $2^{-r\gamma jn+O(j)}$ instead makes the excess $e^{-\Omega(\gamma n)}$, because $\gamma \gg \omega_n + \log_2 n/n$. At depth zero the strict physical block bounds give the same decrease. Markov at $e^{-c\sqrt{n}}$ proves the near-one assertions. A union bound over the blocks, reference densities and depths shows that all bounds and E_0 hold simultaneously. Combine the blocks by Lemma 9.10. Lemma 9.5 supplies the phase and pushforward conclusions. □

These conditions depend only on the chain, which is fixed before choosing the coset shifts and sampling the lists. The family can include any prescribed source, root or Schur-complement covariances needed later. Compact one-parameter covariance families can be covered by an n^{-3} mesh and a Loewner enlargement: the Gaussian majorization factor is $1 + O(n^{-2})$. This suffices for the exponent slack in (9.37). The near-one assertion uses the actual finite reference family.

Lemma 9.12 (Terminal basis and uniform shell initialization)**.** *For sufficiently large fixed K, a basis of Γ_D with $\max_i \|\widetilde{\mathbf{b}}_i\| \leq t_D/n^2$ can be found and tested in time $2^{0.20n+o(n)}$, with success probability $1 - o(1)$. Given such a basis, for a fixed shell variance $X > 0$, coefficient $a_D = (\Delta_D/\mathcal{V}_D)2^{\eta n}$ and $\ell = \eta + \frac{1}{2}\log_2 X \geq 0$, the uniform-shell initializer has*

$$\mathbb{E}A_{\mathbf{x}} = a_D \mathbf{1}_H(\mathbf{x}), \qquad \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{x}=\mathbf{y}})] \leq a_D^2 \mathbf{1}_H(\mathbf{x})\mathbf{1}_H(\mathbf{y}) \tag{9.39}$$

on every prescribed coset, using $2^{\ell n+o(n)}$ proposals.

Proof. Use block reduction with a sufficiently small fixed block fraction α so its time is below $2^{0.20n+o(n)}$, as in Section 5. Put $w_\star = t_D/(Cn^{3+2/\alpha})$ and choose $K > 4 + 2/\alpha$. On E_0, Poisson summation gives $\Theta_{K_0}(w_\star) \leq 2(\sigma_0/w_\star)^n$. Frequencies outside qK_0 survive to K_D with probability at most q^{-D}; bound the rest by their entire theta sum. Therefore

$$\mathbb{E}[\Theta_{K_D}(w_\star) - 1 \mid K_0] \leq 2q^{-D}(\sigma_0/w_\star)^n + \Theta_{qK_0}(w_\star) - 1 = 2^{-\Omega(n\log_2 n)}. \tag{9.40}$$

The $\frac{1}{2}\log_2 f$ term in $\log_2 q/n$ cancels the shrinking t_D in the first term. Dual separation makes the second term negligible. Markov, the smoothing-to-minima bound and the block-reduction guarantee of Section 5 give the asserted basis test.

Its Babai cells have volume Δ_D and radius at most $t_D/(2n^{3/2})$. Enlarge the shell by this radius to a region H^+ containing every rounding cell of a supported point. Uniform sampling in H^+ and Babai rounding to the prescribed coset proposes each supported point with probability $\Delta_D/\operatorname{vol}(H^+)$. Take $N = \lceil a_D \operatorname{vol}(H^+)/\Delta_D \rceil$ independent proposals and accept a rounded point in H with probability $a_D \operatorname{vol}(H^+)/(N\Delta_D)$. The expected count is exact. Independence of distinct proposals gives the second moment bound even when both proposals round to the same point. Finally $\operatorname{vol}(H^+) = \mathcal{V}_D X^{n/2} 2^{o(n)}$, so $N = 2^{\ell n+o(n)}$. Proposals can be processed one at a time. □

9.3 Moments of the occurrence counts

We first analyze the lists without work and size limits. Fix a chain satisfying the preparation bounds and average over the shifts and sampling randomness below each merge. The bounds hold for every prescribed output coset. A vector can occur more than once in a list. In a second moment we count pairs of distinct list entries, including entries with the same vector value. For a count $A_{\mathbf{x}}$ this means using $A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{x}=\mathbf{y}})$.

Lemma 9.13 (Upper bound on the expected output count)**.** *If the expected count in child i is at most ha_iw_i, where $h \geq 1$ and $0 \leq w_i \leq 1$ are its shell weights, the output count before any deletions satisfies*

$$\mathbb{E}A_{\mathbf{z}}^{\mathrm{raw}} \leq h^k a_{\mathrm{o}} w_{\mathrm{o}}(\mathbf{z}). \tag{9.41}$$

Deleting histories can only decrease this upper bound.

Proof. Given the child cosets, the child calls are independent. The expected number of representations therefore uses a product of expected child counts. Apply Lemma 9.2 and (9.7). The remaining Gaussian integral has mass at most the parent weight because the tests include that weight. □

Lemma 9.14 (Normalization of each shared-coordinate pattern)**.** *For the source covariance (9.6), let two distinct accepted histories share exactly the child occurrence identifiers in a proper nonempty set S. Relative to the nominal parent product, their shared-coordinate coefficient is at most*

$$2^{-M(S)n+o(n)}, \qquad M(S) = \sum_{i\in S} \eta_i + \tfrac{1}{2}\log_2 \det R_{SS}. \tag{9.42}$$

Write $F = S^c$. After fixing shared coordinates and the total, the remaining free dimension is $|F|-1$. In any determinant-one integer free-coordinate chart, let its conditional covariance be C_F and let $H(S) = \mathcal{H}(C_F)$ for a specified scalar/pair Schur path. The remaining correlation is at most $2^{(H(S)+\xi)n}$ on the preparation event. For $|F| = 1$ it is one and $H(S) = 0$. For $S = \varnothing$, a near-one bound for the full conditional reference gives the corresponding near-one two-history contribution.

Proof. For two tuples with the same entry at positions in S, use first moments at those positions and second moments for distinct entries at the other positions. The shared coordinates have conditional covariance $t_{\mathrm{c}}^2 R_{SS}$. The Gram test puts their residuals in the prescribed window. One copy of their density is bounded there by $2^{o(n)}\mathcal{V}_{\mathrm{c}}^{-|S|}\det(R_{SS})^{-n/2}$; the other copy integrates to at most one. The occurrence normalization is exactly

$$\frac{\Delta_{\mathrm{c}}^{|S|}}{\prod_{i\in S} a_i}\,\mathcal{V}_{\mathrm{c}}^{-|S|}\det(R_{SS})^{-n/2} = 2^{-M(S)n}. \tag{9.43}$$

The sum constraint leaves $|F|-1$ lattice displacements. For example, successive sums of the free coordinates form an integer chart of determinant one. Regression gives covariance C_F in this chart. The difference between the two conditional means shifts the autocorrelation sum. Lemma 9.11 bounds the sum uniformly in this shift. The Gram tests imply the wider windows of the reference densities, whose normalizers are $1 - e^{-\Omega(\sqrt{n})}$. This proves the asserted bound. When S is empty, both full conditional densities are bounded directly by the normalized reference density. The mean square of its periodized density is close to one, and no shared-coordinate factor is needed. The restriction to the Gram window is used in the density bound that gives (9.43). □

Lemma 9.15 (Second moment bound for a merge)**.** *Suppose the first and second moments in each child are at most h times their nominal values. Suppose the mean square of the periodized conditional reference for pairs sharing no entries is at most $1+\varepsilon_n$. If every nonempty proper S satisfies $M(S) - H(S) \geq \kappa_n$, where $\kappa_n \gg \omega_n + \log_2 n/n$, and the preparation slack is at most $\kappa_n/4$, then the parent second moment for distinct entries before deletions is at most*

$$h^{2k} a_{\mathrm{o}}^2 w_{\mathrm{o}}(\mathbf{z}) w_{\mathrm{o}}(\mathbf{z}')(1 + \varepsilon_n + 2^{-\kappa_n n/3}) \tag{9.44}$$

Fixed positive margins are included. The same conclusion holds after a deterministic deletion rule if the positive-margin hypothesis is imposed only on patterns that can remain between two tuples that survive the deletions.

Proof. Group pairs of distinct tuples by the set of positions where they use the same entry. For pairs sharing no entries, use the assumed mean-square bound. Apply Lemma 9.14. Restricting to the Gram window costs $2^{O(n\omega_n+\log_2 n)}$ uniformly on compact covariance domains, which is absorbed by $\kappa_n n$. There are only a fixed number of patterns. Pairwise independence of the random values assigned to distinct tuples lets us multiply their acceptance probabilities. If two tuples use the same entries at every position, they produce the same occurrence. Such pairs are excluded from the second moment, so only proper shared sets occur. The product of all child moment factors is at most h^{2k}. Deletion decreases every remaining count and removes the eliminated patterns, proving the last assertion. □

Lemma 9.16 (Uniform pointwise lower bound under the Gaussian tests)**.** *Give every type shell the same fixed relative halfwidth $c\omega_n$. Choose prefix and full Gram windows with larger fixed halfwidths. The factor $G_{\mathbf{z}}$ consists of these tests and the output coset test. For positive definite source covariances in a fixed compact domain, uniformly over every supported parent,*

$$\int g_{\mathbf{z}} G_{\mathbf{z}} \geq 1 - e^{-c'\sqrt{n}} \tag{9.45}$$

for some $c' > 0$, apart from the deterministic parent membership test. Consequently a source without internal history deletion maps pointwise child first lower factors a to a parent lower factor at least $a^k(1 - e^{-c'\sqrt{n}})$.

Proof. Write $\|\mathbf{z}\|^2/(nt_{\mathrm{c}}^2 P) = 1 + \delta$. Conditional Gaussian means and covariances give

$$\frac{\mathbb{E}\|\mathbf{X}_i\|^2}{nt_{\mathrm{c}}^2 X_i} = 1 + \alpha_i \delta, \qquad \alpha_i = \frac{s_i^2}{PX_i} < 1. \tag{9.46}$$

Strict inequality follows from positive definiteness and $k \geq 2$. Compact positive-definite bounds give a common maximum $\alpha_\star < 1$. Even at the parent shell boundary, the child expectation is separated from its shell boundary by $c(1 - \alpha_\star)\omega_n$. The full expected Gram matrix is $\Sigma + \delta\, \mathbf{s}\mathbf{s}^T/P$; its whitened perturbation has operator norm $|\delta|$. Thus a wider full Gram window contains its expectation with fixed relative margin. Every proper prefix is another fixed quadratic form of this Gaussian tuple and has the same concentration at the chosen partial-sum width. Gaussian norm and quadratic-form concentration at width ω_n gives failure $e^{-\Omega(n\omega_n^2)} = e^{-\Omega(\sqrt{n})}$, uniformly in the parent and its direction. Averaging over the cosets by Lemma 9.2 then multiplies the lower factors for the child counts. This argument applies before any internal deletion rule. □

Lemma 9.17 (Moment bounds through the recursion)**.** *Consider a fixed finite collection of source types expanded into N_n calls, with $\ln N_n = o(\sqrt{n})$. Suppose all acceptance probabilities are at most one and the mean squares of the periodized references for pairs sharing no entries are close to one. Suppose also that all proper shared-coordinate margins dominate $\omega_n + \log_2 n/n$, with preparation slack at most one quarter of the smallest margin. With the terminal initialization of Lemma 9.12, sources without internal deletion satisfy, uniformly in every prescribed coset,*

$$\begin{aligned} (1 - e_n) a_{\mathrm{o}} w_{\mathrm{o}}(\mathbf{x}) \leq \mathbb{E} A_{\mathbf{x}} \leq (1 + e_n) a_{\mathrm{o}} w_{\mathrm{o}}(\mathbf{x}), \\ \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{x}=\mathbf{y}})] \leq (1 + e_n) a_{\mathrm{o}}^2 w_{\mathrm{o}}(\mathbf{x}) w_{\mathrm{o}}(\mathbf{y}), \qquad e_n = e^{-\Omega(\sqrt{n})}. \end{aligned} \tag{9.47}$$

Proof. Induct through the expanded call graph. The exponent of the first moment factor is at most the number of children, and that of the second moment factor is at most twice that number. The lower bound follows from Lemma 9.16. At the terminal level the expected count equals its

target, and the second moment bound for distinct entries has no error factor. At each merge the additional error is $e^{-\Omega(\sqrt{n})}$ by preparation and Lemma 9.15. The amplification through the expanded call graph is $\exp(O(\ln N_n))$, which preserves this error form. In particular $D = O((\ln n)^2)$ and $N_n = \exp(O((\ln n)^2))$ are allowed. □

For the binary construction, the only nonempty proper patterns are its two singleton children. Their conditional marginal variance is $p_\star$ and both have margin $M_\star > \gamma_n \gg \omega_n$ eventually; there is no remaining free displacement. The empty pattern uses the scalar reference $p_\star > 1$ at a positive child depth. This verifies all hypotheses of Lemma 9.17. For the narrow tests used in the binary construction, one can check (9.45) directly. With $\mathbf{x}_{1,2} = \mathbf{z}/2 \pm \mathbf{e}$, except with $e^{-\Omega(\sqrt{n})}$ probability, $|\|\mathbf{e}\|^2/(np_\star t_{j+1}^2) - 1| \le \omega_n/4$ and $|\langle \mathbf{z}, \mathbf{e}\rangle|/(nXt_{j+1}^2) \le \omega_n/8$. The resulting child norm error is at most $(f/4 + p_\star/(4X) + 1/8)\omega_n < \omega_n$; the cross-Gram error is at most $(f/4 + p_\star/(4X))X\omega_n < X\omega_n$.

9.4 Finding a shortest vector from the top lists

The following root statements allow different list variances. At depth zero take k top calls with coefficients $a_i = \delta_0\, 2^{\eta_i n}$ and shell variances X_i in t_0^2 units. Choose their first $k-1$ cosets independently Haar modulo Γ_0 and choose the last so that the shifts sum to zero modulo Γ_0. Let Σ be a positive definite root covariance with diagonal X_i and total $P = \mathbf{1}^T\Sigma\mathbf{1} = e\chi$. Put $\mathbf{s} = \Sigma\mathbf{1}$ and $R = \Sigma - \mathbf{s}\mathbf{s}^T/P$. Its principal $(k-1)$-coordinate minors have common determinant $\det\Sigma/P$.

Lemma 9.18 (A polynomial-mass conditional root window)**.** *For a shortest* $\mathbf{v}$ *at a correct length guess, the conditional Gaussian with means* $(s_i/P)\mathbf{v}$ *and residual covariance* $t_0^2 R$ *has a narrow window* $W_{\mathbf{v}}$ *contained in the region allowed by the search tests. In* $k-1$ *free coordinates its volume satisfies*

$$I_{\mathbf{v}} = \mathcal{V}_0^{k-1}(\det\Sigma/P)^{n/2} n^{O(1)}, \tag{9.48}$$

with both polynomial upper and inverse-polynomial lower factors. The uniform density on this window is at most a polynomial factor times the conditional Gaussian density.

Proof. Require the residual Gram entries and residual inner products with $\mathbf{v}$, normalized by nt_0^2, to be within fixed multiples of n^{-2} of their Gaussian values. Separate the $k-1$ axial coordinates along $\mathbf{v}$. Their joint density is fixed-dimensional Gaussian. In $\mathbf{v}^\perp$, the whitened residual Gram matrix has a Wishart density of fixed dimension. Stirling's formula shows that these inverse-polynomial windows about its typical Gram matrix have at least inverse-polynomial probability. On this window the Gaussian quadratic form is $(k-1)n + O(n^{-1})$. Its density is therefore within polynomial factors of $\mathcal{V}_0^{-(k-1)}(\det\Sigma/P)^{-n/2}$, proving the volume estimate and the comparison of the two densities. The coordinate change to free coordinates and the total has determinant one. Fixed larger broad windows absorb the length-guess error. □

Lemma 9.19 (Root moments, with separate upper and lower hypotheses)**.** *Let* $Z_{\mathbf{v}}$ *count tuples in the unique top lists with sum* $\mathbf{v}$ *and free coordinates in* $W_{\mathbf{v}}$*. Averaging over uniform shifts at the nominal counts gives the mean*

$$m_{\mathbf{v}} = \Delta_0^{-(k-1)} \prod_i a_i\, I_{\mathbf{v}} = \delta_0\, 2^{\mu_{\mathrm{w}} n} n^{O(1)}, \qquad \mu_{\mathrm{w}} = \sum_i \eta_i + \tfrac{1}{2}\log_2(\det\Sigma/P). \tag{9.49}$$

It is independent of the extension directions conditional on the fixed indices.

Suppose the first and second moments of the top counts have upper factors $1+e_n$, and define, for $1 \le |S| \le k-2$,

$$M_{\mathrm{r}}(S) = \sum_{i \in S} \eta_i + \tfrac{1}{2} \log_2 \det R_{SS}, \tag{9.50}$$

Let K_S be the remaining free conditional covariance given the coordinates in S and the total. If the mean square of the periodized reference for root pairs sharing no entries is at most $2^{o(n)}$ and every $M_{\mathrm{r}}(S) - \mathcal{H}(K_S)$ has a fixed positive margin, then

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}} Z_{\mathbf{v}}^2] \le 2^{o(n)} m_{\mathbf{v}}^2 + (1+o(1)) m_{\mathbf{v}}. \tag{9.51}$$

This upper bound does not require pointwise first lower bounds. If, additionally, the expected top counts have lower factors $1-e_n$, then

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}} Z_{\mathbf{v}}] \ge (1-o(1)) m_{\mathbf{v}}. \tag{9.52}$$

The same lower conclusion holds if the first-moment deficit under the normalized window functional is proved to be $o(1)$ separately. Whenever $m_{\mathbf{v}} \ge 2^{-o(n)}$, these conclusions give a representation of a shortest vector with probability $2^{-o(n)}$. If this mean square is bounded by a fixed power of n and the nonempty sharing margins stay uniformly positive, then

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}} Z_{\mathbf{v}}^2] \le n^A m_{\mathbf{v}}^2 + O(m_{\mathbf{v}}) \tag{9.53}$$

for a fixed A, uniformly on compact root covariance families. In particular $m_{\mathbf{v}} \ge n^{-B}$ gives inverse-polynomial success.

Proof. Average over the first $k-1$ uniform root shifts. The last shift is fixed by their sum, giving the coefficient $\Delta_0^{-(k-1)} \prod_i a_i$ in the integral. Lemma 9.18 proves (9.49). Pointwise first upper bounds give the corresponding actual mean upper bound, and pointwise lower bounds give its lower bound on each prepared chain. Average over the chain event, whose probability is $1-o(1)$ uniformly in the fixed prime choices. The nominal integral is independent of those remaining directions. This proves (9.52). Alternatively transport a relative deficit under the normalized window functional; its polynomial Gaussian domination is paid only once.

For the second moment expand pairs by their exact agreeing top coordinates. Distinct top calls have independent streams conditional on the complete coset tuple. Within a call, use the first moment for an entry shared by the two tuples and the second moment for distinct entries. For the empty shared set, compare each narrow-window density once with the broad conditional Gaussian, costing polynomial factors by Lemma 9.18. The remaining lattice displacement has dimension $k-1$, and the corresponding periodized reference has mean square at most $2^{o(n)}$. For nonempty proper S, the same disintegration as in Lemma 9.14 gives coefficient $2^{-M_{\mathrm{r}}(S)n+o(n)}$ and remaining correlation $\mathcal{R}_{\Gamma_0^{k-1-|S|}}(h_{K_S})$. Preparation bounds it by $2^{(\mathcal{H}(K_S)+\xi)n}$, uniformly in the regression phase. The positive margins absorb ξ and the polynomial window comparisons. If $k-1$ top values agree, the common sum forces the last value to agree. Top uniqueness then forces its identifier to agree as well. Thus these are identical tuples and contribute only the diagonal actual mean, at most $(1+o(1))m_{\mathbf{v}}$. This proves (9.51). The second-moment inequality gives the last assertion. For (9.53), keep the actual polynomial window comparisons in the empty-sharing term. Every nonempty term still has a fixed negative exponential rate after these comparisons and the factors from restricting the Gaussian densities. Its contribution is absorbed in $n^A m_{\mathbf{v}}^2$. Applying the second-moment inequality to $\mathbf{1}_{\mathcal{G}} Z_{\mathbf{v}}$ bounds the probability of success together with successful preparation. □

Removing repeated vectors costs little in the expected top counts. For a count $A_{\mathbf{x}}$ before removal,

$$\mathbb{E}[A_{\mathbf{x}} - \mathbf{1}_{\{A_{\mathbf{x}} \geq 1\}}] \leq \tfrac{1}{2}\mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{x}} - 1)] \leq \tfrac{1}{2}(1 + e_n)a_i^2 w_i(\mathbf{x}). \tag{9.54}$$

If $\eta_i < 0$ and δ_0 is inverse polynomial, this is exponentially smaller than $a_i w_i(\mathbf{x})$. Removing repetitions also preserves the second moment bound. Consequently the lower hypotheses of Lemma 9.19 hold for the top lists generated here.

9.5 Quantum access and the dependent last coset

We use the cap access and tree reporter of Lemmas 8.3 and 8.4. For a stored child list of rate u, valid prefixes of rate π, and cap exponents F, A, B, the exponents for the index, the query–cap vertices and the candidate visits are

$$I = u + F - A, \qquad H = \pi + F - B, \qquad N = \pi + u + F - A - B. \tag{9.55}$$

The counts include query–cap vertices with empty data lists and repeated visits to the same candidate. A pair extends the search only through its first common cap. We build and store the data index and compute the query–cap vertices as the search visits them. Use independent fresh codes at successive joins. To bound the valid-prefix population, drop earlier coverage conditions after canonicalization: each geometric prefix history has at most one continuing address.

Every top coset is marginally Haar, including the last one. If a source output coset is Haar modulo its parent lattice, its complete tuple of child cosets is product Haar modulo the child lattice. Indeed the change from the parent and the first $k-1$ child shifts to all k child shifts is an invertible torus change. Induction gives Haar marginals throughout the source calls. The conditional independence and the upper bounds on the expected counts then let us estimate the number of cap memberships and partial sums by integration. At the root only the first $k-1$ shifts are independent; its last stage requires a separate bound.

Lemma 9.20 (Candidate work at a dependent last root coset)**.** *Suppose the final search uses a fixed number of lists, with the upper bounds on the expected counts for uniform shell weights. Assume the partial-sum and Gram tests are rotation invariant and the cap codes are drawn independently of the lists. Assume its prefix covariances range over a fixed compact positive-definite domain. Let its penultimate sum have variance V and its last child variance C in σ_0^2 units, with $(\sqrt{V} - \sqrt{C})^2 > 0$. Require this gap uniformly over the allowed shell windows. After averaging the chain with the indicator $\mathbf{1}_{\mathcal{G}}$, the expected number of query–cap vertices and candidate visits at the last stage satisfies (9.55) up to $2^{o(n)}$. This bound counts visits before the final norm and membership tests.*

Proof. The query–cap vertices depend only on the independent earlier shifts. For candidate visits, apply the upper bounds on the expected list counts and average over those shifts. Count all prefixes that satisfy the earlier rotation-invariant tests, since each such prefix extends through at most one common cap per join. Enlarge the last local Cartesian cap conditions to ordinary global caps. Each full center is marginally uniform and independent of the lists. After averaging it, the resulting two-shell reference is rotationally invariant.

Normalize its cap-weighted sum density F by the single-cap measures, so $\int F = 1$. Positive mixtures over allowed prefix Gram matrices and shell radii preserve the bounds below. The ratio of lattice to continuous incidences is $\Delta_0 \sum_{\mathbf{z} \in \Gamma_0} F(\mathbf{z})$. The zero displacement is absent by separation of the two radii. For other displacements put

$$Z = \frac{\|\mathbf{y} + \mathbf{x}\|^2}{n\sigma_0^2}, \qquad c = \frac{V + C - Z}{2\sqrt{VC}}, \qquad A_{\mathrm{c}} = (1 - \alpha^2)(1 - \beta^2), \tag{9.56}$$

where the query direction is $-\mathbf{y}$. Coarea for two uniform spheres bounds their unconstrained sum density by

$$\mathcal{V}_0 f_{\mathrm{sum}}(\mathbf{z}) \le 2^{o(n)} \left(\frac{1-c^2}{eZ}\right)^{n/2}. \tag{9.57}$$

Shell widths and polynomial dimensional factors affect only $o(n)$. In the active region the shared-cap factor relative to the two single-cap measures is at most $(\Delta(c)/A_{\mathrm{c}})^{n/2}2^{o(n)}$. The identity

$$\frac{(1-c^2)\Delta(c)}{A_{\mathrm{c}}} = 1 - \frac{(c-\alpha\beta)^2}{A_{\mathrm{c}}} \le 1 \tag{9.58}$$

cancels its possible radial cost. If one cap is inactive, for example $\beta < \alpha$ and $c \ge \beta/\alpha$, the factor is instead $(1-c^2)/(1-\beta^2) \le 1$; the other case is symmetric. Lemma 9.3 on inverse-polynomial radial cells supplies $(Z/\chi)^{n/2}$, canceling $Z^{-n/2}$ in (9.57). Consequently

$$\mathcal{V}_0 \sum_{\mathbf{z}\in\mathcal{L}\setminus\{\mathbf{0}\}} F(\mathbf{z}) \le 2^{[\beta_{\mathrm{lat}} - \frac{1}{2}\log_2(e\chi)+o(1)]n}. \tag{9.59}$$

At an endpoint of the range of feasible angles, the intersection numerator tends to zero with power $(n - O(1))/2$. The same exponent bound therefore applies to cells meeting these endpoints.

To control the dense-grid term, bound the indicators by smooth cutoffs supported on an n^{-B_0} enlargement, for sufficiently large fixed B_0. Their normalized mass changes by $1 + o(1)$; the positive gap and (9.59) persist. Differentiation in the last vector of the fiber integral gives, for unit $\mathbf{u}$ and $0 \le l \le 2n+2$,

$$\|D^l_{\mathbf{u}} F\|_1 \le \sigma_0^{-l} \exp(O(n \log_2 n)). \tag{9.60}$$

The coefficient covariances lie in a fixed compact positive-definite domain. Gevrey cutoff derivatives, inverse-polynomial transition widths and the positive shell norm lower bounds give this estimate; all coarea and single-cap normalizers are already included in F.

Put $a_p = (p-1)/(p^n-1)$. Each nonzero class in $(\mathcal{L}/p)/\mathcal{L}$ belongs to the random extension with probability a_p, while input-lattice points always belong. Hence

$$\mathbb{E}_{\Gamma_0} \sum_{\mathbf{z}\in\Gamma_0} F(\mathbf{z}) = (1-a_p) \sum_{\mathbf{z}\in\mathcal{L}} F(\mathbf{z}) + a_p \sum_{\mathbf{z}\in\mathcal{L}/p} F(\mathbf{z}). \tag{9.61}$$

The separation bound for $p\mathcal{L}^*$ in (9.33) holds for the chosen index even after we drop the preparation indicator. Using this bound, integration by parts through order $2n+2$, and packing, the sum of the absolute values of the nonzero Fourier terms is at most $e^{-\Omega(n^2)}$. Poisson summation therefore gives

$$\Delta_0 a_p \sum_{\mathbf{z}\in\mathcal{L}/p} F(\mathbf{z}) = a_p p^{n-1}(1 + e^{-\Omega(n^2)}) = 1 + o(1). \tag{9.62}$$

After multiplication by $\Delta_0 = \delta_0 \mathcal{V}_0$, the other term is $o(1)$ by (9.59) and $\beta_{\mathrm{lat}} - \frac{1}{2}\log_2(e\chi) < -0.00338590$. The preparation indicator is removed only inside this nonnegative upper bound, after applying the expected-count bounds valid for prepared chains. Thus the deterministic reference is averaged without conditioning on the preparation event. The expected incidence ratio is at most $1 + o(1)$. □

For binary sources, reverse the second child direction, giving $\rho = 1 - f/2 = \frac{1}{2} - \gamma_n/2 \to \frac{1}{2}$. Use equal cap thresholds $\alpha = \beta = \sqrt{3/16}$ and put

$$\Delta = 1 - \frac{2\alpha^2}{1+\rho}, \qquad F = -\tfrac{1}{2}\log_2 \Delta, \qquad A = -\tfrac{1}{2}\log_2(1-\alpha^2),$$
$$I = b + F - A, \qquad J_{\mathrm{pair}} = 2b + F - 2A, \qquad T_{\mathrm{rep}} = \tfrac{1}{2}(J_{\mathrm{pair}} + b). \tag{9.63}$$

Prefix sums of actual bucket products index the compatible pair addresses as in Section 5.4. At each address, test whether it is the pair's first common cap address, apply the shell and coset conditions, and compare the assigned random value to the acceptance probability. For J candidate addresses and R_{out} accepted pairs, reporting costs $2^{o(n)}\sqrt{J(R_{\mathrm{out}}+1)}$. The expected-count bound averaged over a uniform output shift gives $\mathbb{E}[\mathbf{1}_{\mathcal{G}} R_{\mathrm{out}}] \le 2^{bn+o(n)}$. The Cauchy–Schwarz inequality gives

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}}\sqrt{J(R_{\mathrm{out}}+1)}] \le \sqrt{\mathbb{E}[\mathbf{1}_{\mathcal{G}} J]\,\mathbb{E}[\mathbf{1}_{\mathcal{G}}(R_{\mathrm{out}}+1)]}. \tag{9.64}$$

The time exponent for constructing and loading the index, reporting the pairs, writing the output, and sorting is at most $\max\{b, F, I, T_{\mathrm{rep}}\}$.

For the root define, for $i = 2, 3, 4$,

$$a_i = \frac{y_{i-1} + C - y_i}{2y_{i-1}}, \qquad v_i = C - a_i^2 y_{i-1}, \qquad d_i = v_i/C, \qquad \rho_i = \sqrt{1-d_i},$$
$$P_1 = b, \qquad P_i = P_{i-1} + b + \tfrac{1}{2}\log_2 d_i. \tag{9.65}$$

Use $s = \tau_{\mathrm{Q}}$, $U = 2^{-2b}$, $W = 2^{-2s}$ and

$$\alpha = \sqrt{1-U}, \qquad \beta_i = \rho_i\alpha + \sqrt{d_i(U-W)},$$
$$Q_i = \max\{0, b + \tfrac{1}{2}\log_2 d_i,\ s + \tfrac{1}{2}\log_2(1-\beta_i^2)\}. \tag{9.66}$$

Here $F = s$ and $A = b$. Building the indices costs $2^{sn+o(n)}$. The exponents for query–cap vertices and candidate visits are at most $P_{i-1} + Q_i$. The signs satisfy $a_i > 0$, so the query direction is the opposite prefix direction. The active-cap inequalities and $U > W$ are strict. Lemma 8.4 gives search rate at most $\frac{1}{2}\max\{b, P_{i-1} + Q_i : 2 \le i \le 4\}$. Lemma 9.20 covers the last stage because $(\sqrt{V} - \sqrt{C})^2 > 0.01999966117$.

9.6 Finite checks and completion

The companion source archive contains verifier version 1.2, `scripts/verify_quantum_sections.py`, the corresponding input and dependency files, and the certified interval output. The manifest `scripts/quantum-verification.sha256` identifies these files by SHA-256 digest. The checks certify finite inequalities for the pair bound, the covariance margins for the final search and list construction, and the access rates used in Sections 9–11. The lemmas give the analytic reductions from the probabilistic and geometric estimates to these inequalities.

At the limiting parameters $f = 1$, $b = c$, the checks use only the exact inputs in (9.2), (9.12) and (9.66). The native root covariance is $\Sigma = eQ$, with total $P = e\chi$. For its empty reference, eliminate the first free coordinate as a scalar and use the remaining conditional pair. The resulting bounds are

$$\eta_\infty < -0.06050770048, \qquad \mu_{\mathrm{w},\infty} > 0.00956955218,$$
$$\text{empty-root scalar variance} > 1.36142738192,$$
$$\text{final-pair determinant} > 1.04107981405, \qquad \text{final-pair integer minimum} > 1.13736684481. \tag{9.67}$$

The covariance of the final search is independent of n. Increasing b_n above c only increases the expected shortest-vector count and the margins for tuples that share entries. The ten nontrivial proper shared classes are covered by symmetry below. Lower bounds are rounded downward and upper bounds upward.

S	$M_{\mathrm{r}}(S) >$	$\mathcal{H}(K_S) \le$	$M_{\mathrm{r}}(S) - \mathcal{H}(K_S) >$
$\{1\},\{2\},\{3\}$	0.16205231	0	0.16205231
$\{4\}$	0.11119088	0	0.11119088
Pairs in $\{1,2,3\}$	0.23287175	0.10228681	0.13058494
Pairs containing 4	0.19439344	0.06380849	0.13058494

For the first three singleton classes the pair determinant and integer minimum equal those above; for $\{4\}$ they exceed 1.11713500419 and 1.22045620115. The remaining scalar variance for a pair in $\{1,2,3\}$ exceeds 0.86779513084, and for a pair containing 4 exceeds 0.91534215086, giving the displayed allowances.

The supplied `scripts/verify_quantum_sections.py -section gaussian` checks these inequalities with outward 80-decimal intervals and positive lower pivots. Pair minima use integer Gauss reduction, with a finite integer box and a spectral exclusion bound when a reducedness comparison is inconclusive. The same command also checks the root access rates.

At the limit, the source cap formulas give the exact identities

$$F_\infty = c, \qquad I_\infty = T_{\mathrm{rep},\infty} = 2c - \tfrac{1}{2}\log_2(16/13) = \tfrac{1}{2}\log_2(13/9) = \tau_{\mathrm{Q}}. \tag{9.68}$$

Use $s = \tau_{\mathrm{Q}}$ in (9.66). At $b = c$, root search has rate below 0.256562709, with active-cap margins above 0.2061. Continuity keeps search below τ_{Q} near this limit; loading has rate τ_{Q}. For $0 < \gamma \le 0.01$, the source rates are at most $\tau_{\mathrm{Q}} + 4\gamma$: on $1 \le f \le 1.01$, $F'(f) < 0.1$, while $T_{\mathrm{rep}} = \frac{3}{2}b + \frac{1}{2}F - A$.

Choose preparation slack $\xi_n = \gamma_n/10$. The margin in the acceptance exponent and the margins for pairs sharing entries dominate ω_n, and the empty conditional variance stays above one by a fixed margin. Lemma 9.17 therefore applies through all $\exp(O((\ln n)^2))$ calls. Top uniqueness, the positive limiting exponent of the expected shortest-vector count, and (9.53) give a fixed B such that the probability of both successful preparation and a shortest-vector representation is at least n^{-B}.

The source cap construction has overhead $2^{O(n^{9/10})}$, and the expanded recursion costs a factor $\exp(O((\ln n)^2))$. These are smaller than $2^{n\gamma_n}$ for all sufficiently large n. The resulting expected work bound before imposing the limit is

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}} W] \le U_n := 2^{\tau_{\mathrm{Q}} n + 10 n \gamma_n}. \tag{9.69}$$

Leaf work has rate $b_n \to c$ and preprocessing rate below 0.20. Sorting and uniqueness are charged before deletion.

Lemma 9.21 (Bounded execution and exact root search)**.** *Suppose we have computable bounds $U_n \ge \mathbb{E}[\mathbf{1}_{\mathcal{G}} W]$ and $p_{0,n} \le \Pr[\mathcal{G}$ and a shortest-vector representation exists$]$. Then bounded trials with constant success use $O(U_n/p_{0,n}^2)$ time and no more space, apart from subexponential amplification factors. In particular $U_n = 2^{cn+o(n)}$ and $p_{0,n} = 2^{-o(n)}$ give $2^{cn+o(n)}$ when these bounds are explicit. For every fixed $\zeta > 0$, expected work $2^{cn+o(n)}$ and success probability $2^{-o(n)}$ also give a work bound of $2^{(c+3\zeta)n+o(n)}$. The final search solves exact SVP using polynomially many length guesses and norm thresholds.*

Proof. Stop each trial after $4U_n/p_{0,n}$ charged operations. Markov bounds lost successful prepared trials by $p_{0,n}/4$. Amplify coverage, sampling and quantum errors so their total within the cap is at

most $p_{0,n}/4$. Pointwise coverage can be integrated under every nonnegative history functional; after a cap one can also use the bounded address sets. A capped trial succeeds with probability at least $p_{0,n}/2$. Repeating $O(p_{0,n}^{-1})$ times proves the bound. The counter applies to failed preparations as well and bounds stored records. For the fixed-slack consequence take $p_{0,n} = 2^{-\zeta n}$ and $U_n = 2^{(c+\zeta)n}$ for sufficiently large n.

During each final search, fix the arrays, filter codes and assigned random values. Vary only the norm test for the final sum. For rational input clear denominators by Q_{in}, so $\|Q_{\mathrm{in}}\mathbf{x}\|^2$ is integral for $\mathbf{x} \in \mathcal{L}$. Binary search this threshold and find a qualifying leaf at the smallest feasible value. The input bounds and length grid of Section 5 require only polynomially many searches. Exact checks reject zero and nonlattice outputs. The shortest valid output is a shortest vector whenever a trial within the work limit finds one. □

Proof of Theorem 9.1. Apply Lemma 9.21 with $p_{0,n} = n^{-B}$ and (9.69). The resulting work is $O(U_n n^{2B}) = 2^{\tau_{\mathrm{Q}} n + o(n)}$, since $\gamma_n \to 0$. Amplification and exact threshold search give success at least $2/3$. □

10 Quantum SVP in Time $2^{0.2478...n+o(n)}$ – Multi-List Merges

Theorem 10.1 (Quantum SVP with multi-list merges)**.** *There is a quantum algorithm for exact SVP on every input lattice, with success probability at least* $2/3$ *and time and space bounded by*

$$\mathrm{T_Q} \le 2^{t_\mathrm{Q} n+o(n)}, \qquad \mathrm{S_Q} \le 2^{s_\mathrm{Q} n+o(n)}. \tag{10.1}$$

With 24 *child lists per merge and ten lists in the final search, the time and space exponents are*

$$t_\mathrm{Q} = s_\mathrm{Q} = 0.247895 < 0.2479. \tag{10.2}$$

A version with four child lists per merge and four lists in the final search has $t_\mathrm{Q} = s_\mathrm{Q} = 0.2546$*. Both versions use one list type and the lattice point counts in Proposition 1.1 and Lemma 9.3.*

Section 9 combines two lists at each merge and uses several lists in the final search. Here we also combine several lists at each merge. The sum is built one vector at a time, and we set a radius for each partial sum. Choosing these radii lets us distribute the angular constraints across successive joins. This controls the number of prefixes searched and allows shorter child lists. All lists still use the same radius and expected size throughout the lattice chain.

We store each child list and a spherical-filter index. When the search visits a partial sum, its index queries identify suitable entries in the next list. Quantum reporting finds all accepted tuples at an internal merge, while the final search only needs one shortest-vector representation. The Gaussian acceptance rule again gives constant target weights across the output shell. Because every accepted tuple is kept, the expected count of each output can be bounded below as in the preceding section.

The main new estimate concerns pairs of tuples that use some of the same list entries. For each such choice we must bound its contribution to the second moment. Conditional Gaussian distributions have a simple form in partial-sum coordinates, which lets us evaluate these bounds recursively.

Comparison with heuristic pair sieving. The heuristic quantum sieving exponent of about 0.2563 in [BCSS23] comes from finding shortening pairs in a sieve list. Our construction can distribute the angular constraints needed to obtain a short sum over several joins. The prefix radii then control the size of each intermediate search domain while the incoming lists stay smaller. For ideal independent uniform shell points, the same calculation would multiply the population of each prefix level by the next list size and the angular-window probability $d_h^{n/2+o(n)}$. This gives the exponent in (10.6). Lemma 10.2 bounds the cost of reporting accepted tuples at a merge and of finding one shortest-vector representation in the final search for the lists produced by the algorithm.

Roadmap. Section 10.1 describes the lists and the partial sums. Section 10.2 bounds the work of finding tuples, and Section 10.3 bounds the contribution of shared entries to the second moment. Section 10.4 checks the parameters for both constructions, and Section 10.5 proves the theorem.

10.1 One list type and its prefix geometry

Use the chain and normalization of Lemma 9.4, with fixed physical scale t, normalizer $\mathcal{V} = (\sqrt{2\pi e}\, t)^n$, and $\delta_d = \det \Gamma_d / \mathcal{V}$. Thus $f = 1$. Centered lattice counts use $\beta_{\mathrm{lat}} = \frac{1}{2}\log_2(2e/\pi)$. Pair preparation

for projected separated sets uses the certified slack β_{cert} above β_{sph} as in Section 9.2. Put

$$P_{\mathrm{r}} = e\chi, \qquad \chi = 0.639615,$$
$$H = \left\{ \mathbf{x} : \left| \frac{\|\mathbf{x}\|^2}{nt^2 X} - 1 \right| \le \omega_n \right\}, \qquad \eta = \ell - \tfrac{1}{2}\log_2 X, \qquad a_d = \delta_d 2^{\eta n}. \tag{10.3}$$

Here $\omega_n = n^{-1/4}$, with the fixed window constants allowed by Lemma 9.16. For a list on a prescribed coset of Γ_d, the target expected number of occurrences of $\mathbf{x}$ is $a_d \mathbf{1}_H(\mathbf{x})$. The target list-size exponent is ℓ. Use

$$\tau_\star = 10^{-6}, \qquad \gamma_n = \min\left\{ \frac{\tau_\star}{4}, \frac{1}{\ln(n+e)} \right\},$$
$$D = \left\lceil \frac{K \log_2(n+2)}{\gamma_n} \right\rceil, \qquad \xi_n = \gamma_n. \tag{10.4}$$

Here $\tau_\star$ is a certified margin for the acceptance probability, and K is the fixed terminal-depth constant. The actual index between successive lattices has rate $\gamma_n + o(1)$. It stays positive while $\gamma_n \to 0$; the source geometry specified below uses zero index rate. The same-type source has k independent child calls at depth $d+1$; the cosets and the rule for accepting a tuple are those of the Gaussian splitting construction in Section 9. The terminal sampler is Lemma 9.12.

Specify a merge by its prefix variances $Y_0, \dots, Y_{k-1}$ in units of t^2. Every child has variance X, so $Y_0 = X$. For a source take $Y_{k-1} = X$; for an r-list root use $k = r$ and $Y_{r-1} = P_{\mathrm{r}}$. Define

$$A_h = \frac{Y_{h-1} + X - Y_h}{2Y_{h-1}}, \qquad g_h = 1 - A_h, \qquad q_h = X - A_h^2 Y_{h-1} \quad (h \ge 1),$$
$$q_0 = X, \qquad T_{hh} = X, \qquad T_{ih} = -A_h \sum_{j<h} T_{ij} \quad (i < h). \tag{10.5}$$

Symmetry specifies the other entries of $\boldsymbol{T}$. If all $q_h > 0$, this is positive definite, its determinant is $\prod_h q_h$, and $\mathbf{1}^T \boldsymbol{T} \mathbf{1} = Y_{k-1}$. Indeed, independent innovations $\mathbf{e}_h$ of variance $q_h t^2$ realize $\mathbf{x}_h = -A_h \sum_{i<h} \mathbf{x}_i + \mathbf{e}_h$. For the parameters below $A_h > 0$, and the useful correlation between the opposite prefix and the next child is

$$d_h = q_h / X, \qquad \rho_h = \sqrt{1 - d_h}, \qquad \Pi_0 = \ell, \qquad \Pi_h = \Pi_{h-1} + \ell + \tfrac{1}{2}\log_2 d_h. \tag{10.6}$$

Thus $2^{\Pi_h n + o(n)}$ bounds the expected number of valid prefixes of length $h+1$ before acceptance. At internal merges and before the final child at the root, averaging over shifts and using the bounds on expected occurrence counts gives the factor $d_h^{n/2+o(n)}$ for the shell and angle condition of the new child. Tests on the full Gram matrix can only decrease the count. The dependent last root coset is treated below. On tuples passing the Gram-matrix tests, the acceptance probability is at most one when

$$\tau_0 := \Pi_{k-1} - \ell > \tau_\star, \qquad \tau_{\mathrm{acc}}(n) = \tau_0 - \gamma_n > 3\tau_\star/4. \tag{10.7}$$

Its logarithmic probability rate is $-\tau_{\mathrm{acc}}(n) + o(1)$, by the Gaussian formula for the expected occurrence count in Lemma 9.13.

10.2 Quantum work of a tuple merge

For an index rate $c > \ell$, use the cap access construction of Lemma 8.3. At join h, set

$$\begin{aligned} U &= 2^{-2\ell}, \qquad W = 2^{-2c}, \qquad \alpha_{\mathrm{d}} = \sqrt{1-U}, \\ \alpha_{\mathrm{q},h} &= \rho_h\sqrt{1-U} + \sqrt{d_h(U-W)}, \\ Q_h &= \max\left\{0, \ell + \tfrac{1}{2}\log_2 d_h, c + \tfrac{1}{2}\log_2(1-\alpha_{\mathrm{q},h}^2)\right\}, \\ C &= \max\{\ell, \Pi_{h-1} + Q_h : 1 \le h < k\}. \end{aligned} \tag{10.8}$$

Write C_{s} for this expression evaluated on the source prefixes with index rate c_{s}, and C_{r} for its root evaluation with index rate c_{r} and r children in place of k. Both cap constraints are active in the intersection calculation for each instance below. Choosing the filters to cover the lower endpoint of the allowed correlation interval changes the rates by $o(1)$.

For the two certified geometries, the time bound accounts for basis preparation, list output, the two filter indices, reporting the accepted tuples at an internal merge, and finding one tuple in the final search. The corresponding time and space exponents are

$$t_{\mathrm{Q}} = \max\left\{0.20, \ell, c_{\mathrm{s}}, c_{\mathrm{r}}, \frac{C_{\mathrm{s}}+\ell}{2}, \frac{C_{\mathrm{r}}}{2}\right\}, \qquad s_{\mathrm{Q}} = \max\{0.20, \ell, c_{\mathrm{s}}, c_{\mathrm{r}}\}. \tag{10.9}$$

Lemma 10.2 (Tuple work)**.** *Suppose the occurrence upper bounds of Lemma 9.17 hold. A source merge with output rate ℓ has expected work exponent at most $\max\{c, (C+\ell)/2\}$. A root search has expected work exponent at most $\max\{c, C/2\}$. Construction, storage and loading of the indices, and source output writes, are included.*

Proof. A point lies in a data cap with probability $2^{-\ell n+o(n)}$, and the determinant for the common-cap calculation is W. Each index therefore has exponent c. At join h, the number of filter queries and candidate entries has exponent $\Pi_{h-1} + c + \frac{1}{2}\log_2(1-\alpha_{\mathrm{q},h}^2)$. The other terms in Q_h include prefix navigation and valid continuations. Thus the search tree has at most $2^{Cn+o(n)}$ vertices in expectation, including queries to empty filters. The output coset of each call is uniform modulo its lattice. Averaging over it and the fresh shifts gives independent uniform child cosets, which justify these counts at internal merges. The first $r-1$ root cosets are independent and uniform. For the last child, Lemma 9.20 applies under the separation condition $Y_{r-2} \neq X$.

Codes are fresh after the lists. At each join we continue through a single designated common filter, so each accepted tuple corresponds to one leaf. Its acceptance decision uses the identifiers of its entries. The bound on expected occurrence counts gives at most $2^{\ell n+o(n)}$ accepted leaves in expectation. If S is the tree size and R the number of accepted leaves, Cauchy–Schwarz gives

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}}\sqrt{S(R+1)}] \le \sqrt{\mathbb{E}[\mathbf{1}_{\mathcal{G}}S]\,\mathbb{E}[\mathbf{1}_{\mathcal{G}}(R+1)]} \le 2^{(C+\ell)n/2+o(n)}. \tag{10.10}$$

Apply Lemma 8.4. At the root only one shortest-vector representation is needed, giving rate $C/2$. A fixed arity contributes only a constant number of indices. The search computes prefixes as needed. □

10.3 Checking every shared set

Each internal merge keeps all accepted tuples. We therefore check the inequality of Lemma 9.14 for every proper nonempty set of shared identifiers, including sets containing the first two children. The Markov form (10.5) makes these checks explicit.

Let $P = Y_{k-1}$, $\mathbf{s} = \boldsymbol{T}\mathbf{1}$, and $\boldsymbol{R} = \boldsymbol{T} - \mathbf{s}\mathbf{s}^T/P$. For a proper shared set S, put $F = S^c = (f_0 < \cdots < f_{m-1})$. Express the free coordinates by their successive partial sums. The inverse of the resulting conditional covariance is the tridiagonal matrix $\boldsymbol{J}_F$, with

$$\begin{aligned} D(a,c) &= q_a^{-1} + \sum_{a<h<c} \frac{(1-g_h)^2}{q_h} + \frac{g_c^2}{q_c}, & E_a &= -g_a/q_a, \\ (J_F)_{jj} &= D(f_j, f_{j+1}), & (J_F)_{j-1,j} &= E_{f_j}, \\ L_0 &= k\eta + \tfrac{1}{2}\log_2 \frac{\prod_h q_h}{P}, & M(S) &= L_0 - m\eta + \tfrac{1}{2}\log_2 \det \boldsymbol{J}_F. \end{aligned} \tag{10.11}$$

The last identity is Jacobi's identity applied to $\boldsymbol{R}_{SS}$. This change of coordinates is unimodular. The first $m-1$ partial sums are free, and the fixed total determines the last free child. For $m = 1$ its dimension is zero and $M(S) = L_0 - \eta$.

For $m \geq 2$ compute pivots

$$d_0 = D(f_0, f_1), \qquad d_j = D(f_j, f_{j+1}) - E_{f_j}^2/d_{j-1}. \tag{10.12}$$

For $m = 2$ use the bound $H_1(1/d_0)$. For $m \geq 3$, eliminate the partial sums in reverse order. The resulting scalar variances are $1/d_{m-2}, \ldots, 1/d_2$, followed by the first pair conditional on those later coordinates. With $\boldsymbol{K}_2 = (\boldsymbol{J}_F[0:2, 0:2])^{-1}$, the resulting bound is

$$H(S) = H_2(\boldsymbol{K}_2) + \sum_{j=2}^{m-2} \max\{0, \tfrac{1}{2}\log_2 d_j\}. \tag{10.13}$$

Here H_1, H_2 are the scalar and pair bounds of Lemmas 9.7 and 9.8. This is a valid Schur path in Lemma 9.11.

Lemma 10.3 (Checking all choices of shared entries)**.** *The following monotone dynamic program gives a certified lower bound for every $M(S) - H(S)$. Check one- and two-free-coordinate sets directly. Initialize every triple $f_0 < f_1 < f_2$ with*

$$d = d_1, \qquad L = L_0 - 3\eta + \tfrac{1}{2}\log_2(d_0 d_1) - H_2(\boldsymbol{K}_2). \tag{10.14}$$

If the last free index is a, append any $c > a$ by

$$d' = D(a,c) - E_a^2/d, \qquad L' = L - \eta + \min\{0, \tfrac{1}{2}\log_2 d'\}. \tag{10.15}$$

For each possible last index, keep the componentwise smallest states. Compare states separately according to whether an index was omitted. Record a state when the shared set is nonempty. Sources include every starting index f_0 and all one-free-coordinate cases. At an r-list root check shared sizes $1, \ldots, r-2$, as required by Lemma 9.19.

Proof. Equations (10.12)–(10.13) give the initialization and update. Both updated components are nondecreasing in the previous d, L whenever $d' > 0$. A componentwise smaller lower-bound state therefore bounds every continuation of a larger state. Directed rounding downwards preserves this property. An omitted index, including any index after the last one chosen, makes S nonempty. This test keeps $S = \varnothing$ out of the bounds for nonempty sets. Nonpositive or unresolved pivots reject a certificate. □

Remark 10.4 (Unequal child parameters). For child variances X_h and population exponents ℓ_h, use the individual X_h in (10.5), with $q_0 = X_0$, and put $\eta_h = \ell_h - \frac{1}{2}\log_2 X_h$. In (10.11) replace $k\eta$ by $\sum_h \eta_h$ and $m\eta$ by $\sum_{h\in F}\eta_h$. In (10.14) replace 3η by $\eta_{f_0} + \eta_{f_1} + \eta_{f_2}$, and the append update subtracts η_c. The one-free-coordinate case is $L_0 - \eta_{f_0}$. The inverse-covariance formulas and the state comparisons are unchanged. When only certain shared sets are allowed, record the information needed to distinguish them and compare states separately.

We check tuples with no shared entries separately using the same elimination order. The scalar variances exceed one, and for the final pair both the determinant and the minimum of its quadratic form on nonzero integer vectors exceed one. Hence these tuples need no correction in the preparation bound. The pair minimum is checked over all integer vectors by finite enumeration with a spectral tail bound, as in Lemma 9.8.

10.4 A fixed parameter instance

We specify the source geometry by balancing the reporting costs at successive prefixes. The backward recurrence determines the correlations for all but the first join. The forward recurrence then gives the prefix variances, with the terminal variance prescribed by the next list in the recursion. The root prefix variances are given explicitly and checked together with the source. The numbers of children and lists set the lengths of the two prefix sequences. Their scale and expected list size are given by X and ℓ. The work exponents set the target for the source recurrence and the two filter index sizes. All terminating decimals below are exact inputs.

$$\begin{aligned} &k = 24, && r = 10, && X = 1.4, && \ell = 0.17392,\\ &s = 0.2478915, && c_{\mathrm{s}} = 0.247895, && c_{\mathrm{r}} = 0.244. \end{aligned} \tag{10.16}$$

Here s is the target used in the source recurrence, and c_{s} is the exponent for the index that the algorithm builds. Set $U = 2^{-2\ell}$ and $W = 2^{-2s}$. Starting with $\widehat{\Pi}_{k-1} = \ell$, work backwards for $h = k-1, \dots, 2$:

$$\begin{aligned} &K_h = 2^{2(s-\widehat{\Pi}_h)}, \qquad x_h = \frac{\sqrt{(1-U)(U-W)} + \sqrt{UK_h - W}}{U},\\ &\widehat{d}_h = (1+x_h^2)^{-1}, \quad \widehat{\rho}_h = x_h\sqrt{\widehat{d}_h}, \qquad \widehat{\Pi}_{h-1} = \widehat{\Pi}_h - \ell - \tfrac{1}{2}\log_2 \widehat{d}_h. \end{aligned} \tag{10.17}$$

The chosen branch equalizes the reporting cost at these joins. To connect two full child lists to this tail, define

$$\begin{aligned} &\widehat{d}_1 = 2^{2(\widehat{\Pi}_1 - 2\ell)}, \qquad \widehat{\rho}_1 = \sqrt{1-\widehat{d}_1}, \quad Y_0 = X, \quad Y_1 = 2X(1-\widehat{\rho}_1),\\ &Y_h = Y_{h-1} + X - 2\widehat{\rho}_h\sqrt{XY_{h-1}} \quad (2 \le h < k). \end{aligned} \tag{10.18}$$

Finally prescribe $Y_{k-1} = X$ exactly. Reconstruct all actual correlations, populations and costs from (10.5); the last join uses this prescribed endpoint. In particular, we check the acceptance margin τ_0 directly after fixing the last radius. The first join is filtered, and every child list has exponent ℓ.

At the root set $Y_0^{\mathrm{r}} = X$, $Y_9^{\mathrm{r}} = P_{\mathrm{r}}$ and use Table 4. Reconstruct its covariance by the same prefix formulas. Choose nine independent Haar cosets modulo Γ_0, and determine the tenth by requiring their sum to vanish modulo Γ_0. Generate independent main calls on this complete coset tuple. Remove repeated vector values within each of the top lists as in Lemma 9.19, then search the root tree.

h	Y_h^{r}	h	Y_h^{r}
1	1.969341	5	1.902406
2	1.926397	6	1.831689
3	1.884000	7	1.821689
4	1.891813	8	1.697365

Table 4: Root prefix variances in units of t^2.

Quantity	Source	Root
Minimum scalar variance with no shared entries	> 1.0486	> 1.1013
Final pair determinant	> 1.0880	> 1.0538
Final pair integer minimum	> 1.1392	> 1.1610
Reporting or search rate	< 0.247892	< 0.241513
Index construction and storage rate	0.247895	0.244

Table 5: Certified preparation and work bounds for the 24-child source and ten-list root.

Lemma 10.5 (Verified parameter inequalities)**.** *The instance* (10.16) *gives acceptance probabilities at most one and satisfies both strict cap conditions. The margin for every nonempty proper set of shared source entries exceeds* $1.3193 \cdot 10^{-6}$*. At the root, every required margin exceeds* 0.0008131 *and the exponent* L_0 *for representations of the shortest vector exceeds* 0.0154*. The work and preparation bounds in Table 5 hold.*

Proof. The supplied `scripts/verify_quantum_sections.py -section tuples` reads the exact inputs from this section and implements (10.5)–(10.18) with directed 80-decimal intervals. Every division, square root and logarithm requires a positive certified argument. Lemma 10.3 checks all $2^{24} - 2$ proper nonempty source sets and all 1012 required root sets. The scalar and pair conditions for tuples without shared entries are checked separately, including the minima over integer vectors. The cap conditions and $\eta < 0$ are included in the interval checks. The last-root separation $Y_8^{\mathrm{r}} \neq X$ follows from Table 4. Separately, the source acceptance margin before accounting for the index is $\tau_0 > 1.3193 \cdot 10^{-6}$. By (10.4), accounting for the index still leaves a fixed positive margin. The maximum child and proper-prefix norm contractions are below 0.14048 and 0.31260, respectively, so the bounds for the shell and Gram-matrix tests apply uniformly. All bounds in the table are rounded outwards from these certificates. □

For a smaller instance with wider source margins, take four children and four root lists. Use

$$\begin{aligned} X &= 1.3801, \qquad \ell = 0.19835, \qquad c_{\mathrm{s}} = 0.2546, \qquad c_{\mathrm{r}} = 0.243, \\ (Y_0, Y_1, Y_2, Y_3) &= (X, 1.4546, 1.4824, X), \\ (Y_0^{\mathrm{r}}, Y_1^{\mathrm{r}}, Y_2^{\mathrm{r}}, Y_3^{\mathrm{r}}) &= (X, 2.0568, 2.0276, P_{\mathrm{r}}). \end{aligned} \tag{10.19}$$

The accompanying four-list certificate verifies all fourteen source sharing patterns. Its minimum source margin exceeds 0.0003749, compared with $1.3193 \cdot 10^{-6}$ for the larger instance. The shortest-vector representation exponent at the root exceeds 0.0008410, source reporting is below 0.254435, and root search is below 0.251850. The conditions for tuples without shared entries and for the cap intersections are also strict.

10.5 Completion of the bounds

Proof of Theorem 10.1. Prepare the finite covariance family with Lemma 9.11 and slack $\xi_n = \gamma_n$, below every certified sharing margin. Our schedule satisfies $n\xi_n \gg n\omega_n + (\ln n)^2$ and $D = O((\ln n)^2)$.

There are $2^{O((\ln n)^2)}$ recursive calls. The moment induction multiplies its local $e^{-c\sqrt{n}}$ errors by at most $e^{O((\ln n)^2)}$, so the error remains $e^{-\Omega(\sqrt{n})}$. The acceptance probabilities, the bounds for tuples without shared entries, and the remaining sharing checks give the moment bounds of Lemma 9.17. The lower bound for the expected count of each vector applies because the construction keeps every accepted tuple. The complete identifiers in the reference analysis have subexponential length, whose overhead is included in $2^{o(n)}$. Under the global work limit, call paths and occurrence ranks give compact addresses of length $O(n + (\ln n)^2)$.

Lemmas 9.18 and 9.19 give a shortest-vector representation with at least inverse-polynomial probability. The factors from the comparison of windows are polynomial and all other margins are fixed. There is no randomized acceptance step in the final search. Lemma 10.2 and Table 5 bound expected work on prepared chains, indices, stored sibling lists and outputs by $2^{0.247895n+o(n)}$; leaf work and basis preparation are smaller. Apply Lemma 9.21 to bound the work on every trial. The code overhead $O(n^{9/10})$ and recursion overhead $O((\ln n)^2)$ are $o(n\gamma_n)$. Thus, for all sufficiently large n, we can use the success-probability lower bound $p_{0,n} = n^{-C}$ for a fixed sufficiently large C, and the expected-work upper bound $U_n = 2^{0.247895n+10n\gamma_n}$. The capped trials have total work and storage bounded by

$$O(U_n/p_{0,n}^2) = 2^{0.247895n+o(n)}. \tag{10.20}$$

The four-child instance gives $2^{0.2546n+o(n)}$ by the same argument. Repeating the trials and minimizing the output norm gives success probability at least $2/3$ for both versions. □

11 Quantum SVP in Time $2^{0.2468\ldots n+o(n)}$ – Selection with Gaussian Weights

Theorem 11.1 (Quantum SVP with selection by entry identifiers)**.** *There is a quantum algorithm for exact SVP on every input lattice, with success probability at least* $2/3$ *and time and space bounded by*

$$\mathrm{T_Q} \le 2^{t_\mathrm{Q} n+o(n)}, \qquad \mathrm{S_Q} \le 2^{s_\mathrm{Q} n+o(n)}. \tag{11.1}$$

The time and space exponents are

$$t_\mathrm{Q} = s_\mathrm{Q} = 0.24689646860581055 < 0.2469. \tag{11.2}$$

The construction uses six list types and sixteen lists in the final search. It uses the lattice point counts in Proposition 1.1 and Lemma 9.3.

The preceding section uses one type of list at every merge. Here we allow different radii and list sizes, which lets us use shorter lists in the final search. With shorter lists, however, pairs of tuples that share certain entries can contribute too much to the second moment. To remove these pairs, we designate the first two child lists at each merge and call them the seed lists. Among the accepted tuples with the same pair of seed entries, we keep exactly one. Distinct tuples that remain can share at most one of these two entries.

This choice can also remove tuples needed to produce a shortest vector. To measure that loss, start with the Gaussian distribution of the sixteen entries conditioned on their sum being the desired shortest vector. Each entry has a Gaussian marginal distribution. Use this marginal to weigh the loss in the merge that produced the entry. Repeating this procedure down the merge tree gives a Gaussian weight for every child. We bound the chance of competing tuples under these weights and sum the losses over the tree. This gives the lower bound on the expected number of shortest-vector representations needed for the final search.

Roadmap. Section 11.1 introduces the list types and the rule for choosing one tuple per seed pair. Section 11.2 bounds the loss from this choice. Section 11.3 bounds the probability of finding a shortest vector, and Section 11.4 bounds the search cost. Section 11.5 specifies the shell radii and completes the proof using the checks in Appendix B.

11.1 Several types and selection by seed identifiers

Each list type has a fixed radius and an expected size. We use the fixed-scale chain of Lemma 9.4 with $\gamma = 10^{-6}$ and the same χ and counting exponents as in Section 9. Write $\Delta_d = \det \Gamma_d$, $\delta_d = \Delta_d/\mathcal{V}$ and

$$X_\tau = s_\star \varrho_\tau^2, \qquad \eta_\tau = \ell_\tau - \tfrac{1}{2}\log_2 X_\tau, \qquad a_{\tau,d} = \delta_d 2^{\eta_\tau n}. \tag{11.3}$$

Here ℓ_τ is the list-size exponent, and the target expected count of a point $\mathbf{x}$ is $a_{\tau,d}\mathbf{1}_{H_\tau}(\mathbf{x})$. Each type has one physical radius $\sqrt{nX_\tau}\,t$ throughout the chain. Fixed shell and Gram-window constants are chosen with successive margins as in Section 9.

There are a main type B and five auxiliary types $U_1, \ldots, U_5$, with $U_0 = \mathsf{B}$. Generate them in the order

$$\begin{aligned} U_{i,d} &\longleftarrow 38 \text{ fresh } \mathsf{B}_d \text{ calls and one } U_{i-1,d} \quad (1 \le i \le 5), \\ \mathsf{B}_d &\longleftarrow 40 \text{ fresh } \mathsf{B}_{d+1} \text{ calls and one } U_{5,d+1}. \end{aligned} \tag{11.4}$$

Only main calls occur at depth zero, while auxiliary calls occur at positive depth. Expanding the auxiliary chain gives $40 + 5 \cdot 38 + 1 = 231$ next-level main calls. With the depth of Lemma 9.4, the resulting number of calls is $O(231^D) = O(n^{K \log_2(231)/\gamma})$. At the last lattice in the chain, generate main lists using Lemma 9.12.

Generate each of the first two child lists at its full main-list size, then independently keep entries so that its expected size has exponent ξ_i. The cost includes generating the full lists. The remaining list sizes and the Gaussian covariance are specified below. With Σ in units of t^2, $P = \mathbf{1}^T \Sigma \mathbf{1} = X_o$, use the same exact acceptance rule as before:

$$p_{\mathbf{z}}(\mathbf{x}) = \frac{a_o \Delta_c^{k-1}}{\prod_{h=0}^{k-1} a_h} g_{\mathbf{z}}(\mathbf{x}_0, \ldots, \mathbf{x}_{k-2}) G_{\mathbf{z}}(\mathbf{x}). \tag{11.5}$$

All distinct child calls have private randomness conditional on their complete coset tuple. The acceptance decision is indexed by the tuple of entries, and each tuple is considered through one designated sequence of filters, as in the preceding sections.

After reporting the accepted tuples, choose one of them by a fixed rule for each ordered pair of seed entries. Two distinct tuples in the resulting list therefore cannot share both entries. At the root, generate sixteen main lists on compatible Haar cosets, with the last coset determined by the other fifteen, and remove repeated vector values in these top lists. The root prefix search is otherwise unchanged.

Lemma 11.2 (Upper occurrence moments after selection)**.** *If the chain satisfies the preparation conditions, there is $h_n = 1 + n^{O(1)} e^{-c\sqrt{n}}$ such that every source list after selection satisfies*

$$\begin{aligned} \mathbb{E} A_{\mathbf{x}} &\le h_n a_o w_o(\mathbf{x}), \\ \mathbb{E}[A_{\mathbf{x}}(A_{\mathbf{y}} - \mathbf{1}_{\mathbf{x}=\mathbf{y}})] &\le h_n a_o^2 w_o(\mathbf{x}) w_o(\mathbf{y}). \end{aligned} \tag{11.6}$$

The first bound also holds for the lists before selection.

Proof. Independent subsampling preserves the bounds on the expected count and on the expected number of pairs of distinct entries, with the correspondingly reduced target weights. The Gaussian calculation in Lemma 9.13 gives the bound on the expected count before selection. Selection can only decrease that count. Two distinct tuples after selection cannot share both seed entries. For every other nonempty proper set S, Lemma 9.14 gives relative rate $-M(S) + H(S)$; Lemma B.1 certifies a fixed positive margin. For tuples with no shared entries, the reference density at positive lattice depth needs no correction and contributes $1 + e^{-\Omega(\sqrt{n})}$. Sharing all entries would give the same tuple, which is excluded from the pair count in (11.6). We apply the previous induction to the remaining choices of shared entries. The number of calls is polynomial, so the accumulated error is still exponentially small in $\sqrt{n}$. □

11.2 Passing Gaussian weights down the merge tree

To bound the number of shortest-vector representations below, we measure the loss from selection under a Gaussian weight on the output. Draw the output from this weight and split it using the conditional Gaussian used for splitting. The marginal distribution at a child is the weight used to measure losses in that child list. The preparation in Lemma 9.11 includes the resulting covariances, using a grid of polynomial size for their contraction parameters. For a coefficient covariance C, write $\varphi_{C,\mathbf{m}}$ for the density with mean $\mathbf{m}$ and covariance $t^2 C \otimes I_n$, and omit the mean when it is zero. For each source put

$$\mathbf{s} = \Sigma \mathbf{1}, \qquad P = \mathbf{1}^T \Sigma \mathbf{1}, \qquad R = \Sigma - \frac{\mathbf{s}\mathbf{s}^T}{P}. \tag{11.7}$$

Lemma 11.3 (Child weights and shell tests)**.** *For the source covariance in* (11.7)*, let the incoming normalized Gaussian weight be*

$$\psi(\mathbf{z}) = \varphi_{(1-\theta)P,\mathbf{b}}(\mathbf{z}), \qquad 0 \le \theta < 1. \tag{11.8}$$

Under its Gaussian splitting map, child i has marginal $\varphi_{(1-\theta_i)X_i,\mathbf{b}_i}$, where

$$\mathbf{b}_i = \frac{s_i}{P}\mathbf{b}, \qquad \theta_i = \theta\frac{s_i^2}{PX_i}. \tag{11.9}$$

The factor $s_i^2/(PX_i)$ is strictly less than one. If $\|\mathbf{b}\|^2/(nt^2P) = \theta + O(n^{-2})$, the probability of failing any of the broad shell or Gram-matrix tests is $e^{-\Omega(\sqrt{n})}$ under this weight. The same compatibility condition is preserved at every child.

Proof. Draw the total according to ψ and then draw the children conditionally on that total. The resulting joint Gaussian has mean $(s_i\mathbf{b}/P)_i$ and coefficient covariance $\Sigma - \theta\mathbf{s}\mathbf{s}^T/P$. Its diagonal gives (11.9). Positive definiteness of Σ gives $s_i^2 < PX_i$, since $k \ge 2$. The claimed compatibility at a child follows by substitution.

The covariance contraction is canceled in the expected Gram matrix by the squared mean: for all i, j,

$$\frac{\mathbb{E}\langle \mathbf{X}_i, \mathbf{X}_j\rangle}{nt^2} = \Sigma_{ij} - \theta\frac{s_is_j}{P} + \frac{s_is_j}{P^2}\frac{\|\mathbf{b}\|^2}{nt^2} = \Sigma_{ij} + O(n^{-2}). \tag{11.10}$$

Gaussian quadratic and axial concentration, in the fixed coefficient dimension, give failure $e^{-\Omega(\sqrt{n})}$ for the windows of width $n^{-1/4}$. Each shell or partial-sum test uses a fixed quadratic form of this Gram matrix. The estimates are uniform over the compact covariance family and the bounded incoming contractions in Lemma B.1. □

Lemma 11.4 (Distribution of the conditional residual)**.** *Let V be a nonempty set of nonseed coordinates, $S = V^c$, and put*

$$C = ((\Sigma^{-1})_{VV})^{-1}, \qquad a_V = \frac{\mathbf{1}^TC\mathbf{1}}{P}, \qquad \mathbf{E} = \mathbf{X}_V - \mathbf{X}_S\Sigma_{SS}^{-1}\Sigma_{SV}. \tag{11.11}$$

Before changing the output weight, $\mathbf{E}$ has coefficient covariance C. Under (11.8)*, its density is $\nu = \varphi_{C_\theta,\mathbf{m}}$, with*

$$C_\theta = C - \frac{\theta}{P}C\mathbf{1}\mathbf{1}^TC, \qquad \mathbf{m} = \frac{\mathbf{b}(C\mathbf{1})^T}{P}, \tag{11.12}$$

$$0 < a_V \le 1, \qquad \frac{\det C_\theta}{\det C} = 1 - \theta a_V. \tag{11.13}$$

Proof. Under the original Gaussian distribution, $\mathbf{E}$ is independent of $\mathbf{X}_S$. Its covariance with the total is $t^2C\mathbf{1}$, while the total has variance t^2P. Taking its marginal after changing the distribution of the total gives (11.12). Equivalently, for fixed $\mathbf{E}$, Gaussian integration gives the exact density identity

$$\int \varphi_{\Sigma_{SS}}(\mathbf{X}_S)\,\frac{\psi(\sum_i \mathbf{X}_i)}{\varphi_P(\sum_i \mathbf{X}_i)}\,d\mathbf{X}_S = \frac{\nu(\mathbf{E})}{\varphi_C(\mathbf{E})}. \tag{11.14}$$

The conditional variance of the total given $\mathbf{X}_S$ is $t^2\mathbf{1}^TC\mathbf{1} \le t^2P$. This proves $a_V \le 1$, and the matrix determinant lemma proves the determinant identity. In particular $(1-\theta)C \preceq C_\theta \preceq C$. □

Lemma 11.5 (Mixed correlation inequality)**.** *Let V, C, a_V be as above, and let $B_0(V)$ be the scalar preparation allowance of C in the fixed original-coordinate elimination order. Define*

$$M(V) = -\sum_{i\in V} \eta_i - \tfrac{1}{2}\log_2 \det C. \tag{11.15}$$

Under each of the Gaussian weights above, the relative loss caused by competing tuples with the same seed entries and different vector values exactly in V is at most

$$e^{-\Omega(\sqrt{n})} + 2^{[-M(V)+B_0(V)-\frac{1}{4}\log_2(1-\theta a_V)+o(1)]n}. \tag{11.16}$$

The estimate is uniform in the mean $\mathbf{b}$ and is measured relative to the expected count under the incoming weight before the geometric tests.

Proof. For an accepted tuple h, let $N_{h,V}$ count the competing accepted tuples of this form. The indicator that the rule deletes h is bounded by $\sum_V \min(1, N_{h,V})$. Write Q_h for the condition that $\mathbf{E}_h - \mathbf{m}$ lies in the broad Gram window for C_θ. Then

$$\min(1, N_{h,V}) \le \mathbf{1}_{Q_h^c} + \mathbf{1}_{Q_h} N_{h,V}. \tag{11.17}$$

The bound on expected occurrence counts and (11.14) show that the first term contributes at most an $e^{-\Omega(\sqrt{n})}$ fraction of the incoming expected count. This term depends only on the chance of being outside the window, even if a tuple there has many competitors.

For the second term, use the bound for pairs of distinct entries in the changed child lists, and bound the competitor's acceptance probability by one. Averaging over the shifts leaves a sum over differences $\mathbf{w} \in \Gamma_c^{|V|}$. Dropping other constraints on the competitor increases this nonnegative integral. The two Gram-matrix tests imply that $\mathbf{E}$ and $\mathbf{E}+\mathbf{w}$ lie in the broad windows for C. On the latter window,

$$\varphi_C(\mathbf{E}+\mathbf{w}) \ge 2^{-o(n)} p_0(C), \qquad p_0(C) = \mathcal{V}^{-|V|}\det(C)^{-n/2}. \tag{11.18}$$

Let h_{C_θ} and h_C be smooth Gaussian reference densities whose cutoff functions equal one on these windows. Their normalizing constants are $1 - e^{-\Omega(\sqrt{n})}$. Using the new density inside the integral gives

$$\nu(\mathbf{E}) G_{\mathrm{old}} G_{\mathrm{new}} \mathbf{1}_{Q_h} \le \frac{2^{o(n)}}{p_0(C)} h_{C_\theta}(\mathbf{E}-\mathbf{m}) h_C(\mathbf{E}+\mathbf{w}). \tag{11.19}$$

Poisson summation and Cauchy–Schwarz on the Fourier side imply, uniformly in $\mathbf{m}$,

$$\Delta_c^{|V|} \sum_{\mathbf{w}\in\Gamma_c^{|V|}} \int h_{C_\theta}(\mathbf{x}-\mathbf{m}) h_C(\mathbf{x}+\mathbf{w})\, d\mathbf{x} \le \sqrt{\mathcal{R}_{\Gamma_c^{|V|}}(h_{C_\theta}) \mathcal{R}_{\Gamma_c^{|V|}}(h_C)}. \tag{11.20}$$

Indeed a translation only multiplies a Fourier coefficient by a phase. Restoring the occurrence coefficients in (11.19) gives exactly

$$\frac{\prod_{i\in V} a_i}{\Delta_c^{|V|} p_0(C)} = 2^{-M(V)n}. \tag{11.21}$$

Finally let p_i, p_i^θ be the forward scalar conditional variances of C, C_θ in the same order. Schur-complement monotonicity gives $0 < p_i^\theta \le p_i$. For $H_1(p) = \max\{0, -\frac{1}{2}\log_2 p\}$,

$$\begin{aligned} 0 \le H_1(p_i^\theta) - H_1(p_i) &\le -\tfrac{1}{2}\log_2(p_i^\theta/p_i), \\ \frac{B_\theta + B_0}{2} &\le B_0 - \tfrac{1}{4}\log_2(1-\theta a_V). \end{aligned} \tag{11.22}$$

Both reference densities satisfy Lemma 9.11 whenever the chain satisfies $\mathcal{G}$. The grid in that lemma gives bounds uniform over the range of θ. Equation (11.20) is already uniform in the means. Combining these inequalities proves (11.16). □

Lemma 11.6 (Small selection loss)**.** *At each merge, the relative loss from selecting one tuple per seed pair is $e^{-\Omega(\sqrt{n})}$ under each Gaussian weight used to count shortest-vector representations. Removing repeated vectors from each top list has exponentially small relative loss under every nonnegative weight on the vector value.*

Proof. Lemma B.1 bounds the contractions in (11.9) by a fixed constant below one and certifies a positive reserve in

$$M(V) - B_0(V) + \tfrac{1}{4}\log_2(1 - \theta a_V) > 0 \tag{11.23}$$

for every nonempty set of coordinates with different vector values. Lemma 11.5 therefore applies. If two entries have different identifiers and the same vector value, the bound on pairs of distinct entries introduces an additional factor a_i. The exponents η_i are strictly negative. We can omit these factors when other vector values differ. If all vector values coincide, the factors themselves give an exponentially small bound. The number of such patterns is fixed.

When removing repeated values from the top lists, the number of deleted occurrences of $\mathbf{x}$ is at most $A_{\mathbf{x}}(A_{\mathbf{x}} - 1)/2$. Lemma 11.2 bounds its expectation by $h_n a_{\mathsf{B},0}^2 w_{\mathsf{B}}(\mathbf{x})/2$. Since $\eta_{\mathsf{B}} < 0$ and δ_0 is inverse polynomial, this is exponentially smaller than the target expected count $a_{\mathsf{B},0} w_{\mathsf{B}}(\mathbf{x})$. The inequality is pointwise and remains true after multiplication by any nonnegative weight depending only on the parent value. □

Lemma 11.7 (Accumulating the losses over the merge tree)**.** *Suppose the Gaussian weight used for the final search satisfies Lemma 11.3 when passed down to the child lists. The relative loss in the weighted expected count, compared with the Gaussian splitting maps before selection and geometric tests, is at most $n^{O(1)} e^{-\Omega(\sqrt{n})}$. The same bound holds when the weight for the final search is bounded above by a polynomial multiple of this Gaussian reference.*

Proof. The splitting maps are positive multilinear maps of the expected child occurrence counts. Replace the inputs by their target counts one at a time, bounding the other inputs by (11.6). The contribution of the difference at a child is measured under the Gaussian weight from (11.9). Lemmas 11.3, 9.12, and 11.6 bound losses from the geometric tests, leaf sampling, and selection, respectively. The filter construction also bounds the probability of missing each eligible tuple. Averaging that probability under the same nonnegative weight bounds the resulting loss.

At a node the relative loss is at most the inherited loss plus the smaller of one and the local bound. This is the bound used in (11.17) to handle tuples outside the Gram window. Across polynomially many nodes, the product of the upper factors is $1 + o(1)$. Summing the losses gives the claim. If the final weight is bounded by a polynomial multiple of the reference density, this introduces one polynomial factor at the final search. The estimates hold for each chain satisfying $\mathcal{G}$ and can then be averaged over the chain. □

11.3 Finding a shortest-vector representation after selection

Let $\boldsymbol{T}_{\mathrm{r}}$ be the sixteen-list root covariance in $\sigma_\star^2$ units, so its diagonal is X_{B}/e and $\mathbf{1}^T \boldsymbol{T}_{\mathrm{r}} \mathbf{1} = \chi$. For a fixed shortest vector $\mathbf{v}$ at a correct length guess, put

$$\mathbf{r} = \boldsymbol{T}_{\mathrm{r}} \mathbf{1}, \qquad C = e\left(\boldsymbol{T}_{\mathrm{r}} - \frac{\mathbf{r}\mathbf{r}^T}{\chi}\right), \qquad \mu_{\mathrm{w}} = 16\eta_{\mathsf{B}} + \tfrac{1}{2}\log_2 \det C_{\mathrm{free}}. \tag{11.24}$$

Here C_{free} is a principal fifteen-coordinate minor. Apply Lemma 9.18 to the conditional root Gaussian. The window $W_{\mathbf{v}}$ restricts the residual Gram matrix and the projections along $\mathbf{v}$. It has at least inverse-polynomial Gaussian mass, and its uniform density is at most a polynomial multiple of the Gaussian density. The change of coordinates has determinant one, giving $\det C_{\mathrm{free}} = e^{15} \det \boldsymbol{T}_{\mathrm{r}}/\chi$.

Let $Z_{\mathbf{v}}$ count tuples of the distinct top vector values with total $\mathbf{v}$, residuals in $W_{\mathbf{v}}$ and the required filter coverage. The expected count under the reference construction before the filter tests is

$$m_{\mathbf{v}} = \delta_0\, 2^{\mu_{\mathrm{w}} n} n^{O(1)}, \tag{11.25}$$

with both polynomial upper and inverse-polynomial lower factors. Indeed, averaging over fifteen independent shifts gives $a_{\mathrm{B},0}^{16}\Delta_0^{-15}$, and the window volume is $\mathcal{V}^{15} \det(C_{\mathrm{free}})^{n/2} n^{O(1)}$. Conditional on the fixed prime indices this nominal quantity is independent of the extension directions. Moreover, Lemma 9.11 gives $\Pr(\mathcal{G} \mid \text{fixed indices}) = 1 - o(1)$ uniformly; averaging over the prepared directions therefore preserves this nominal mass up to $1 - o(1)$.

The marginal Gaussian weight at child i in the final search has mean $(r_i/\chi)\mathbf{v}$ and contraction $r_i^2/(\chi(T_{\mathrm{r}})_{ii})$. It is compatible with Lemma 11.3; the correct length guess gives its norm identity to accuracy $O(n^{-3})$. Applying Lemma 11.7 to the final search and accounting for the removal of repeated top entries gives

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}} Z_{\mathbf{v}}] \geq (1 - o(1)) m_{\mathbf{v}}. \tag{11.26}$$

The comparison of the narrow window with the Gaussian weight contributes one polynomial factor. The exponentially small probability of missing a fixed tuple under the filters contributes only $o(m_{\mathbf{v}})$ to the loss. The upper bounds for expected counts give the matching upper estimate.

The second-moment part of Lemma 9.19 uses only upper bounds on expected counts and on pairs of distinct entries. It therefore applies to the top lists after selection. The finite checks give positive margins for all 65 518 nonempty shared sets of size at most fourteen, and tuples with no shared entries need no correction. Agreement in fifteen top values forces the sixteenth. Consequently

$$\begin{aligned} \mathbb{E}[\mathbf{1}_{\mathcal{G}} Z_{\mathbf{v}}^2] &\leq 2^{o(n)} m_{\mathbf{v}}^2 + 2m_{\mathbf{v}}, \\ \Pr[\mathcal{G} \text{ and } Z_{\mathbf{v}} > 0] &\geq \frac{(1 - o(1)) m_{\mathbf{v}}^2}{2^{o(n)} m_{\mathbf{v}}^2 + 2m_{\mathbf{v}}} = 2^{-o(n)}. \end{aligned} \tag{11.27}$$

The last equality uses the certified $\mu_{\mathrm{w}} > 0$ and inverse-polynomial δ_0. This bounds the probability of finding a shortest-vector representation before imposing the work limit. Entries within a list can be dependent.

11.4 Search costs and bounded execution

The search tests use the fixed tuple identifiers, acceptance decisions, and filter indices from Section 8. Each source output coset is uniform modulo its lattice. Averaging over this coset and the fresh shifts makes the child cosets independent and uniform. This follows by induction from the sixteen compatible top cosets, as in Section 9. The bounds in (11.6) therefore give the expected numbers of filter queries and candidate entries used in Lemma 8.3. The last dependent root coset is handled by Lemma 9.20.

At a source with output exponent ℓ_o, the number M_{out} of accepted tuples before selection satisfies $\mathbb{E}[\mathbf{1}_{\mathcal{G}} M_{\mathrm{out}}] \leq 2^{\ell_o n + o(n)}$. If the implicit tree has mean-size exponent C, quantum reporting and Cauchy–Schwarz give time exponent $(C + \ell_o)/2$. We also account for building the indices, writing

the outputs, sorting, and choosing one tuple per seed pair. The construction of Section 11.5 uses index exponents s_{src} and s_{root}, and expected tree-size exponents C_i and C_{r} from (11.41) and (11.42). With output exponent ℓ_i for type i, the overall bound is

$$\begin{aligned} c_\star &= \max\left\{0.20, s_{\mathrm{src}}, s_{\mathrm{root}}, \max_i \ell_i, \max_i \frac{C_i+\ell_i}{2}, \frac{C_{\mathrm{r}}}{2}\right\}, \\ t_{\mathrm{Q}} = s_{\mathrm{Q}} &= c_\star + 3\zeta, \qquad \zeta = 10^{-6}. \end{aligned} \tag{11.28}$$

The formulas and finite bounds below imply

$$\mathbb{E}[\mathbf{1}_{\mathcal{G}} W] \le 2^{c_\star n + o(n)}. \tag{11.29}$$

There are polynomially many recursive calls; preprocessing has exponent below 0.20. Combining (11.27) with this bound, we limit the work of each trial and repeat it as in Section 9, with $\zeta = 10^{-6}$. Searching for the exact minimum norm adds a polynomial factor. The final time and storage exponent is

$$c_\star + 3\zeta = 0.24689646860581055 < 0.2469. \tag{11.30}$$

It remains to specify and certify the finite geometry used in these bounds.

11.5 Choosing the merge geometry

The geometry is determined by two recursions. At a source, work backwards from the tuple count needed for acceptance, then compute the shell radii forwards from the two orthogonal seeds. At the root, the terminal radius is prescribed by the shortest-vector length, so both populations and radii can be computed backwards. The finite choices for the two recursions and their certificates are given in Appendix B.

Inverting a join. Let p be the population exponent after a join, u the appended list's exponent, s' the index exponent, and K the target exponent for the number of queries at this join. Write ρ for the correlation between the prefix and the negative of the appended child, and $d = 1-\rho^2$. The thresholds in (11.39) give a query count with exponent $p^- + s' + \frac{1}{2}\log_2(1-\alpha_{\mathrm{q}}^2)$, where p^- is the preceding population and $p = p^- + u + \frac{1}{2}\log_2 d$. Setting the query exponent to K and solving for $z = \rho/\sqrt{d}$ gives the backward map

$$\begin{aligned} U &= 2^{-2u}, & W &= 2^{-2s'}, \\ z &= \frac{\sqrt{(1-U)(U-W)} + \sqrt{2^{2(K-p-s')} - W}}{U}, & \rho &= \frac{z}{\sqrt{1+z^2}}, \quad p^- = p + \tfrac{1}{2}\log_2(U(1+z^2)). \end{aligned} \tag{11.31}$$

We use the positive branch of the balance equation

$$\left(Uz - \sqrt{(1-U)(U-W)}\right)^2 + W = 2^{2(K-p-s')}. \tag{11.32}$$

The full cost in (11.41) and (11.42) also accounts for the numbers of prefixes before and after the join. We evaluate it using the radii and list sizes of the algorithm.

Source recursions. Put $k_i = 39$ for $1 \le i \le 5$ and $k_6 = 41$. The first two child lists have expected size with exponent ξ_i, and the joint radius of their sum is $R_1 = \sqrt{2}\varrho_{\mathsf{B}}$. The next twenty children are main lists, appended with correlation $c_{\mathsf{B}} = \sqrt{1 - 2^{-2\ell_{\mathsf{B}}}}$, so the population exponent stays at $2\xi_i$. The remaining joins append main lists and then the last child from (11.4). For phase i, apply (11.31) to this tail with $s' = s_{\mathrm{src}}$ and $K = 2\tau_{\mathrm{s}} - \ell_i$, where τ_{s} is the reporting target. Start from $\ell_i + \varepsilon + \gamma \mathbf{1}_{i=6}$, with acceptance margin $\varepsilon > 0$. At the first join of the tail the preceding population is fixed at $2\xi_i$. When the preceding population p^- is fixed, use $\rho = \sqrt{1 - 2^{2(p-p^- - u)}}$ instead of the backward map. Appendix B gives the one additional boundary population used in the first phase.

In order of increasing phase, use the resulting correlations to compute the prefix radii, starting with $R_0 = \varrho_{\mathsf{B}}$ and the orthogonal seed pair. When the appended child has radius a_h, set

$$R_h = \sqrt{R_{h-1}^2 + a_h^2 - 2\rho_h R_{h-1} a_h}. \tag{11.33}$$

The last radius defines ϱ_i in the auxiliary phases. In the main phase prescribe $R_{k_6 - 1} = \varrho_{\mathsf{B}}$ and recompute the last correlation from the two radii and the endpoint. This closes the recursion across lattice levels.

For each source put $v_h = s_\star a_h^2$ and $y_h = s_\star R_h^2$. The Gaussian covariance is determined by

$$\begin{aligned} &\Sigma_{hh} = v_h, && A_h = \frac{y_{h-1} + v_h - y_h}{2y_{h-1}}, && \Sigma_{ah} = -A_h \sum_{c<h} \Sigma_{ac} \quad (a < h), \\ &g_h = 1 - A_h, && q_h = v_h - A_h^2 y_{h-1} \quad (h \ge 1), && q_0 = v_0. \end{aligned} \tag{11.34}$$

By symmetry this specifies all entries. Its determinant is $\prod_h q_h$ and $\mathbf{1}^{\mathsf{T}} \Sigma \mathbf{1} = y_{k_i - 1}$. The exponents for child occurrence counts and numbers of prefixes are

$$\eta_h = u_h - \tfrac{1}{2} \log_2 v_h, \qquad P_1 = 2\xi_i, \qquad P_h = P_{h-1} + u_h + \tfrac{1}{2} \log_2 (q_h / v_h) \quad (h \ge 2). \tag{11.35}$$

Here $u_0 = u_1 = \xi_i$, and all other child exponents are those of their types. The acceptance margin is $P_{k_i - 1} - \ell_i - \gamma \mathbf{1}_{i=6}$. For a shared set S containing every child except one seed, the total fixes the single remaining coordinate. Thus $H(S) = 0$, and the margin is

$$M(S) - H(S) = P_{k_i - 1} - \xi_i + \log_2 (\varrho_{\mathsf{B}} / \varrho_i). \tag{11.36}$$

Both choices of omitted seed give the same margin.

Root recursion. All sixteen children have variance $C = X_{\mathsf{B}}/e$ in $\sigma_\star^2$ units. Set $Y_{15} = \chi$ and start the backward map with population $\frac{1}{2} \log_2 (e\chi) + \mu_0$. For $h = 15, \dots, 4$, use $s' = s_{\mathrm{root}}$, $u = \ell_{\mathsf{B}}$ and the root query target K_{r}. Invert the radius update on its larger branch:

$$Y_{h-1} = \left(\rho_h \sqrt{C} + \sqrt{Y_h - C(1 - \rho_h^2)} \right)^2. \tag{11.37}$$

For the first four children use a covariance invariant under permutations, with diagonal C and off-diagonal $(Y_3 - 4C)/12$. This gives Y_0, Y_1, Y_2 and connects the recursion to four full lists. Apply (11.34) with $v_h = C$ and $y_h = Y_h$ to obtain $\boldsymbol{T}_{\mathrm{r}}$. Recompute the actual populations from $P_0 = \ell_{\mathsf{B}}$ using this covariance. The covariance in units of t^2 is $\boldsymbol{T}^{\mathrm{r}} = e\boldsymbol{T}_{\mathrm{r}}$, where $\boldsymbol{T}_{\mathrm{r}}$ is the root covariance in $\sigma_\star^2$ units. With $\boldsymbol{r}^{\mathrm{r}} = \boldsymbol{T}_{\mathrm{r}} \mathbf{1}$, check

$$\theta_{\mathrm{root}} = \max_h \frac{(r_h^{\mathrm{r}})^2}{C\chi}, \qquad \mu_{\mathrm{w}} = 16\ell_{\mathsf{B}} + \tfrac{1}{2} \log_2 \frac{\det \boldsymbol{T}_{\mathrm{r}}}{C^{16}} - \tfrac{1}{2} \log_2 (e\chi). \tag{11.38}$$

The certificate also checks $Y_{14} \ne C$, so sufficiently narrow shells exclude zero sums at the last join before the lattice-membership test. The cost bounds use the actual populations after connecting to the seeds and to the root's first four children.

Actual access costs. For a join with next-child exponent u, squared sine $d = q_h/v_h$, and index rate s' equal to s_{src} or s_{root}, the thresholds are

$$U = 2^{-2u}, \qquad W = 2^{-2s'}, \qquad \alpha_{\mathrm{d}} = \sqrt{1-U}, \qquad \alpha_{\mathrm{q}} = \sqrt{1-d}\sqrt{1-U} + \sqrt{d(U-W)}. \tag{11.39}$$

Both cap constraints are active in the intersection calculation at each filtered join. The initial seed pair is traversed directly and has population exponent $2\xi_i$. The thresholds give cap-intersection exponent s', insertion exponent u, and per-prefix access exponent

$$Q(u, d, s') = \max\{0, u + \tfrac{1}{2}\log_2 d, s' + \tfrac{1}{2}\log_2(1-\alpha_{\mathrm{q}}^2)\}, \tag{11.40}$$

$$C_i = \max\{2\xi_i, P_{h-1} + Q(u_h, q_h/v_h, s_{\mathrm{src}}) : 2 \le h < k_i\}, \qquad c_i^{\mathrm{rep}} = \tfrac{1}{2}(C_i + \ell_i). \tag{11.41}$$

For the root set $P_0 = \ell_{\mathsf{B}}$ and $P_h = P_{h-1} + \ell_{\mathsf{B}} + \frac{1}{2}\log_2(q_h/v_h)$. Its tree-size and search exponents are

$$C_{\mathrm{r}} = \max\{\ell_{\mathsf{B}}, P_{h-1} + Q(\ell_{\mathsf{B}}, q_h/v_h, s_{\mathrm{root}}) : 1 \le h < 16\}, \qquad c_{\mathrm{root}}^{\mathrm{search}} = C_{\mathrm{r}}/2. \tag{11.42}$$

The index exponents account for construction and storage. The reporting exponents also account for writing the accepted tuples and choosing one per seed pair.

Proof of Theorem 11.1. Appendix B verifies the moment and loss bounds used in (11.27). That probability bound, the work limit in (11.30), and repetition give the stated costs and success probability. The type counts, arities and numerical inputs are fixed independently of n. □

12 Applications

Discrete Gaussian samples yield approximate shortest and closest vectors, and they let us estimate Gaussian masses and smoothing parameters. For bounded-distance decoding, we instead use the SVP algorithm on a lattice formed by adding one coordinate.

12.1 Approximate SVP and CVP

Corollary 1.11 (Approximate SVP and CVP, restated)**.** *For every $\gamma \geq 1$, worst-case γ-SVP and γ-CVP have randomized algorithms with success probability at least $2/3$ and time and space*

$$\mathrm{T} = (3/2)^{n/2+o(n)} 2^{O(n/\gamma^2)}, \qquad \mathrm{S} = (4/3)^{n/2+o(n)} 2^{O(n/\gamma^2)}. \tag{1.14}$$

Proof of Corollary 1.11. We assume $2 \leq \gamma \leq \sqrt{n}$. For $\gamma < 2$, known exact algorithms take $2^{O(n)}$ time and space, which suffices for the stated bounds. For $\gamma > \sqrt{n}$, run the procedure below with approximation factor $\sqrt{n}$.

The embedding reduction of [ADRS15, Theorem 7.1 and the subsequent discussion] uses polynomially many width guesses and $M = 2^{O(n/\gamma^2)}$ centered DGS samples per guess in dimension $n+1$. It keeps the shortest eligible sample for each guess and returns the closest candidate. Generate the samples one at a time in the same workspace. Choose κ so that the joint distribution differs from that of ideal samples by a sufficiently small constant in total variation. A constant number of repetitions gives success probability at least $2/3$. The sample count contributes the stated overhead in time. The reduction and its stored candidates fit the same overhead in space. The SVP result follows from the approximation- and dimension-preserving reduction of [GMSS99]. □

12.2 Bounded distance decoding

Corollary 1.12 (Bounded-distance decoding, restated)**.** *For $0 \leq \gamma \leq \sqrt{\pi/(6e)} \approx 0.4389$, the closest vector to a target $\mathbf{t}$ can be found with probability at least $2/3$ whenever $\operatorname{dist}(\mathbf{t}, \mathcal{L}) \leq \gamma \cdot \lambda_1(\mathcal{L})$, with costs*

$$\mathrm{T} = (3/2)^{n/2+o(n)}, \qquad \mathrm{S} = (4/3)^{n/2+o(n)}. \tag{1.15}$$

Proof of Corollary 1.12. The Kannan embedding makes the difference between the target and the closest lattice point into a shortest vector of a larger lattice. We bound the point counts in the embedded lattice so that the faster SVP tradeoff applies.

Let $c = \frac{1}{2}\log_2(4/3)$ and $\beta_{\mathrm{lat}} = \frac{1}{2}\log_2(2e/\pi)$. The hypothesis on the decoding radius is $\gamma \leq \sqrt{\pi/(6e)} = 2^{c-\beta_{\mathrm{lat}}-1}$. Return $\mathbf{t}$ if it belongs to $\mathcal{L}$. Otherwise let $d = \operatorname{dist}(\mathbf{t}, \mathcal{L}) > 0$ and let $\mathbf{u}$ be the unique closest lattice vector. Its uniqueness follows from $\gamma < 1/2$. First suppose we have a height $h \in [d, (1+1/n)d]$.

Consider the Kannan embedding

$$\Lambda_h = \{(\mathbf{v} - k\mathbf{t}, kh) : \mathbf{v} \in \mathcal{L},\ k \in \mathbb{Z}\}. \tag{12.1}$$

The vectors $\pm(\mathbf{u} - \mathbf{t}, h)$ have length $\ell = \sqrt{d^2 + h^2}$. For $k = 0$, every nonzero vector has length at least $\lambda_1(\mathcal{L}) > \ell$, since $h \leq 2d$ and $5\gamma^2 \leq 5\pi/(6e) < 1$. For $|k| \geq 2$, the last coordinate alone has absolute value at least $2h > \ell$. In the two remaining layers, $k = 1$ and $k = -1$, uniqueness of the closest vector gives the two shortest vectors above. Thus $\lambda_1(\Lambda_h) = \ell$.

We bound the number of embedded vectors at every radius. Let $r \geq 1$ and $R = r\ell$. A vector $(\mathbf{v} - k\mathbf{t}, kh)$ of norm at most R has $|k| \leq R/h \leq \sqrt{2}r$. Let $\mathbf{e} = \mathbf{t} - \mathbf{u}$ and $\mathbf{w} = \mathbf{v} - k\mathbf{u} \in \mathcal{L}$. Then

$$\|\mathbf{w}\| \leq \|\mathbf{v} - k\mathbf{t}\| + |k|\|\mathbf{e}\| \leq \sqrt{R^2 - k^2h^2} + |k|d \tag{12.2}$$

$$\leq R\sqrt{1 + d^2/h^2} = rd\left(\frac{h}{d} + \frac{d}{h}\right) \tag{12.3}$$

$$\leq \big(2 + O(n^{-2})\big)\gamma r \lambda_1(\mathcal{L}). \tag{12.4}$$

For each k, the correspondence to $\mathbf{w}$ is injective. The vector $\mathbf{w} = \mathbf{0}$ contributes at most one point per layer. Apply Proposition 1.1 to count the remaining points whenever the radius in (12.4) is at least $\lambda_1(\mathcal{L})$. At smaller radii only $\mathbf{w} = \mathbf{0}$ is possible. Since there are $O(r)$ layers,

$$N_{\Lambda_h}(r\ell) \leq 2^{o(n)} r \left[1 + 2^{\beta_{\mathrm{lat}} n}\big((2 + O(n^{-2}))\gamma r\big)^n\right]. \tag{12.5}$$

This implies, with $m = n + 1$, $N_{\Lambda_h}(r\lambda_1(\Lambda_h)) \leq 2^{cm + o(m)} r^m$ for $r \geq 1$. Theorem 1.5, with $\beta = \alpha = c$, therefore solves SVP on this embedding with success probability at least $2/3$ and costs

$$\mathrm{T} = (3/2)^{m/2 + o(m)}, \qquad \mathrm{S} = (4/3)^{m/2 + o(m)}. \tag{12.6}$$

To obtain a suitable height, take a geometric grid with ratio $1 + 1/n$. If Q is a common denominator of the basis and target entries, then $1/Q \leq d \leq \|\mathbf{t}\|$. The grid covering these bounds and one additional step has polynomial size and contains a suitable height. Run the SVP algorithm on each embedding with the time and space limits above. For an output $(\mathbf{z}, kh)$ with $k \in \{-1, 1\}$, test that $\mathbf{t} + k\mathbf{z} \in \mathcal{L}$ and keep the closest valid candidate. A suitable height yields $\mathbf{u}$ with probability at least $2/3$. The other trials can only supply candidates at least as far from $\mathbf{t}$. The grid preserves the stated exponents. □

12.3 Gaussian mass estimation

Corollary 1.13 (Gaussian mass estimation, restated)**.** *Given a basis of $\mathcal{L}$, a Gaussian width $s > 0$, and a fixed accuracy parameter $0 < \delta < 1/2$, a randomized algorithm returns $\widehat{\rho} > 0$ such that $1 - \delta \leq \widehat{\rho}/\rho_s(\mathcal{L}) \leq 1 + \delta$ with probability at least $2/3$, in time and space*

$$\mathrm{T} = (3/2)^{n/2 + o(n)}\,\mathrm{poly}(1/\delta), \qquad \mathrm{S} = (4/3)^{n/2 + o(n)}\,\mathrm{poly}(1/\delta). \tag{1.16}$$

Proof of Corollary 1.13. At a sufficiently small width, the origin accounts for almost all the Gaussian mass. We estimate the mass at the desired width by increasing the width in small steps and estimating the ratio at each step. The ratios multiply to give the mass change [ŠVV09].

Let $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$ and $f_{\mathbf{B}} = \|\mathbf{B}^{-1}\|_F^2$. Choose

$$s_0^2 = \min\{s^2, \delta^2/(64n^2 f_{\mathbf{B}})\}.$$

Since $\|\mathbf{B}\mathbf{z}\| \geq \|\mathbf{z}\|/\sqrt{f_{\mathbf{B}}}$, (2.12) gives

$$1 \leq \rho_{s_0}(\mathcal{L}) \leq \rho_{s_0\sqrt{f_{\mathbf{B}}}}(\mathbb{Z})^n \leq (1 + \delta/(8n))^n \leq e^{\delta/8}.$$

Increase the squared width by factors $1 + 1/n$, shortening the last step to end at $s_\nu = s$. There are polynomially many steps. Their squared widths are rational and have polynomial encoding length in the original input units. For $\mathbf{X}_i \sim D_{\mathcal{L}, s_i}$, the random variable

$$W_i = \exp\Big[-\pi\|\mathbf{X}_i\|^2(s_{i-1}^{-2} - s_i^{-2})\Big] \in [0, 1]$$

has mean $r_i = \rho_{s_{i-1}}(\mathcal{L})/\rho_{s_i}(\mathcal{L})$. The step size and (2.8) give $r_i \geq e^{-1/2}$. Use Lemma 2.13 with polynomially many independent samples to estimate each r_i to relative error $\delta/(8\nu)$, with probability at least $11/12$ that all estimates satisfy this bound. Replace any estimate outside $[1/2, 1]$ by the nearest endpoint of the interval. The product $\widehat{\rho} = \prod_{i=1}^{\nu} \widehat{r}_i^{-1}$ then has logarithmic error at most $\delta/4$ from the ratios and $\delta/8$ from the initial mass. Since $\delta < 1/2$, this gives $1 - \delta \leq \widehat{\rho}/\rho_s(\mathcal{L}) \leq 1 + \delta$. If $\nu = 0$, return 1.

For a total of N_{smp} samples, choose sampler accuracy $2^{-\kappa} \leq 1/(12N_{\mathrm{smp}})$. By Lemma 2.15, this changes the success probability by at most $1/12$. Generate samples one at a time in the same workspace, storing the estimates and the running product. Theorem 1.4 gives the stated bounds. □

12.4 Smoothing parameter estimation

Corollary 1.14 (Smoothing parameter estimation, restated)**.** *Given a basis of $\mathcal{L}$, a smoothing bound $0 < \varepsilon \leq 1/2$, and an accuracy parameter $0 < \delta \leq 1$, a randomized algorithm finds $\widehat{\eta}$ such that $1 - \delta \leq \widehat{\eta}/\eta_\varepsilon(\mathcal{L}) \leq 1 + \delta$ with probability at least $2/3$, in time and space*

$$\mathrm{T} = (3/2)^{n/2+o(n)} \operatorname{poly}(1/\varepsilon, 1/\delta), \qquad \mathrm{S} = (4/3)^{n/2+o(n)} \operatorname{poly}(1/\varepsilon, 1/\delta). \tag{1.17}$$

Proof of Corollary 1.14. The smoothing parameter is the width at which the Gaussian mass on nonzero dual vectors falls to ε. First compare a trial width with the smoothing parameter up to a multiplicative factor, then try widths in increasing order until the comparison returns YES.

Let $\alpha = 1 + \delta$ and $\psi(t) = \rho_{1/t}(\mathcal{L}^* \setminus \{\mathbf{0}\})$. We seek a test that answers YES when $t \geq \eta_\varepsilon(\mathcal{L})$ and NO when $\alpha t < \eta_\varepsilon(\mathcal{L})$. Either answer is allowed in the intervening interval. A YES instance has $\psi(t) \leq \varepsilon$. A NO instance has $\psi(\alpha t) > \varepsilon$, so (2.10) gives $\psi(t) > \varepsilon^{1/\alpha^2}$. The two thresholds differ by at least

$$\varepsilon^{1/\alpha^2} - \varepsilon \geq \varepsilon(1 - \alpha^{-2}) \ln(1/\varepsilon) \geq \tfrac{3}{4}(\ln 2)\varepsilon\delta.$$

Apply Corollary 1.13 to $\mathcal{L}^*$ at squared width $1/t^2$ and relative error $\varepsilon\delta/64$. Answer YES if the estimate is less than $1 + \varepsilon + \varepsilon\delta/8$. Whenever the mass estimate has the stated accuracy, this threshold separates the two cases.

To bound the search interval from below, take a nonzero dual vector $\mathbf{y}$, for instance a column of $\mathbf{B}^{-\mathsf{T}}$. Its mass alone exceeds ε for $t < t_{\mathrm{lo}} = \sqrt{\ln(1/\varepsilon)/\pi}/\|\mathbf{y}\|$. To bound the interval from above, use $\|\mathbf{B}^{-\mathsf{T}}\mathbf{k}\| \geq \|\mathbf{k}\|/\|\mathbf{B}\|_F$ for $\mathbf{k} \in \mathbb{Z}^n$. Put $a = e^{-\pi t^2/\|\mathbf{B}\|_F^2}$. For $a \leq 1/(8n)$, the product over coordinates gives

$$\psi(t) \leq \left(1 + 2\sum_{k \geq 1} a^{k^2}\right)^n - 1 \leq (1 + 4a)^n - 1 \leq e^{4na} - 1 \leq 8na.$$

Thus $\psi(t) \leq \varepsilon$ for $t \geq t_{\mathrm{hi}} = \|\mathbf{B}\|_F \sqrt{\ln(8n/\varepsilon)/\pi}$. Round the endpoints outward to rationals. Their logarithmic ratio is polynomial in the input length and $\log(1/\varepsilon)$.

Search the increasing grid from the lower endpoint through the first point at or above the upper endpoint, with ratio $g = 1 + \delta/4$. Use the test above with factor g and repeat its calls enough times that their joint error probability is at most $1/3$. If the first YES occurs at t, the preceding NO gives $\eta_\varepsilon(\mathcal{L}) > t/g$. If t is the initial endpoint, then $\eta_\varepsilon(\mathcal{L}) \geq t > t/g$. The YES answer itself gives $\eta_\varepsilon(\mathcal{L}) \leq gt$, so returning gt overestimates the parameter by at most a factor $g^2 \leq 1 + \delta$. The number of calls and the repetition cost are polynomial in the input length, ε^{-1} and δ^{-1}, which gives (1.17). □

References

[ABGS21] D. Aggarwal, H. Bennett, A. Golovnev, and N. Stephens-Davidowitz. Fine-grained hardness of CVP(P)—Everything that we can prove (and nothing else). In *Proceedings of the 32nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1816–1835. SIAM, 2021. URL: https://arxiv.org/abs/1911.02440, doi:10.1137/1.9781611976465.109.

[ACKS25] D. Aggarwal, Y. Chen, R. Kumar, and Y. Shen. Improved classical and quantum algorithms for the shortest vector problem via bounded distance decoding. *SIAM Journal on Computing*, 54(2):233–278, 2025. doi:10.1137/22M1486959.

[ADRS15] D. Aggarwal, D. Dadush, O. Regev, and N. 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)*, pages 733–742, 2015. URL: https://arxiv.org/abs/1412.7994v5.

[ADS15] D. Aggarwal, D. Dadush, and N. Stephens-Davidowitz. Solving the closest vector problem in 2^n time—the discrete Gaussian strikes again! In *Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 563–582, 2015. doi:10.1109/FOCS.2015.41.

[AKS01] M. Ajtai, R. 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 (STOC)*, pages 601–610, 2001. doi:10.1145/380752.380857.

[AKS04] M. Agrawal, N. Kayal, and N. Saxena. PRIMES is in P. *Annals of Mathematics*, 160(2):781–793, 2004.

[ALS21] D. Aggarwal, Z. Li, and N. Stephens-Davidowitz. A $2^{n/2}$-time algorithm for $\sqrt{n}$-SVP and $\sqrt{n}$-Hermite SVP, and an improved time-approximation tradeoff for (H)SVP. In *Advances in Cryptology – EUROCRYPT 2021, Part I*, volume 12696 of *LNCS*, pages 467–497, 2021. URL: https://arxiv.org/abs/2007.09556v1, doi:10.1007/978-3-030-77870-5_17.

[AMS26] D. Aggarwal, A. Morolia, and N. Stephens-Davidowitz. Faster provable lattice sieving with spherical codes. Cryptology ePrint Archive, Paper 2026/2388, 2026. URL: https://eprint.iacr.org/2026/2388.

[AS19] D. Aggarwal and N. Stephens-Davidowitz. An improved constant in Banaszczyk's transference theorem. arXiv:1907.09020, 2019.

[Bab86] L. Babai. On Lovász' lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, 1986. doi:10.1007/BF02579403.

[Ban93] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296:625–635, 1993.

[BBHT98] M. Boyer, G. Brassard, P. Høyer, and A. Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4–5):493–505, 1998. doi:10.1002/(SICI)1521-3978(199806)46:4/5<493::AID-PROP493>3.0.CO;2-P.

[BCSS23] X. Bonnetain, A. Chailloux, A. Schrottenloher, and Y. Shen. Finding many collisions via reusable quantum walks. In *Advances in Cryptology – EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 221–251, 2023. URL: https://arxiv.org/abs/2205.14023, doi:10.1007/978-3-031-30589-4_8.

[BDGL16] A. Becker, L. Ducas, N. Gama, and T. Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 10–24, 2016.

[BGJ14] A. Becker, N. Gama, and A. Joux. A sieve algorithm based on overlattices. *LMS Journal of Computation and Mathematics*, 17(A):49–70, 2014.

[BLS16] S. Bai, T. Laarhoven, and D. Stehlé. Tuple lattice sieving. *LMS Journal of Computation and Mathematics*, 19(A):146–162, 2016. doi:10.1112/S1461157016000292.

[BM07] A. Barg and O. R. Musin. Codes in spherical caps. *Advances in Mathematics of Communications*, 1(1):131–149, 2007. doi:10.3934/amc.2007.1.131.

[CDLP13] K.-M. Chung, D. Dadush, F.-H. Liu, and C. Peikert. On the lattice smoothing parameter problem. In *Proceedings of the 28th IEEE Conference on Computational Complexity (CCC)*, pages 230–241, 2013. URL: https://arxiv.org/abs/1412.7979, doi:10.1109/CCC.2013.31.

[CL21] A. Chailloux and J. Loyer. Lattice sieving via quantum random walks. In *Advances in Cryptology – ASIACRYPT 2021, Part IV*, volume 13093 of *LNCS*, pages 63–91, 2021. URL: https://arxiv.org/abs/2105.05608, doi:10.1007/978-3-030-92068-5_3.

[CL23] A. Chailloux and J. Loyer. Classical and quantum 3 and 4-sieves to solve SVP with low memory. In *Post-Quantum Cryptography – PQCrypto 2023*, volume 14154 of *LNCS*, pages 225–255, 2023. URL: https://eprint.iacr.org/2023/200, doi:10.1007/978-3-031-40003-2_9.

[Dad13] D. Dadush. A deterministic polynomial space construction for ε-nets under any norm. arXiv:1311.6671, 2013. URL: https://arxiv.org/abs/1311.6671, arXiv:1311.6671, doi:10.48550/arXiv.1311.6671.

[DADRT23] T. Debris-Alazard, L. Ducas, N. Resch, and J. Tillich. Smoothing codes and lattices: Systematic study and new bounds. *IEEE Transactions on Information Theory*, 69(9):6006–6027, 2023.

[DEL25] L. Ducas, L. Engelberts, and J. Loyer. Wagner's algorithm provably runs in subexponential time for SIS$^\infty$. In *Advances in Cryptology – CRYPTO 2025, Part I*, volume 16000 of *LNCS*, pages 353–384, 2025. URL: https://arxiv.org/abs/2503.23238v1, doi:10.1007/978-3-032-01855-7_12.

[DH96] C. Dürr and P. Høyer. A quantum algorithm for finding the minimum. arXiv:quant-ph/9607014v2, 1996.

[DK16] D. Dadush and G. Kun. Lattice sparsification and the approximate closest vector problem. *Theory of Computing*, 12(2):1–34, 2016. doi:10.4086/toc.2016.v012a002.

[DL26] E. Doulgerakis and T. Laarhoven. Smooth sailing through spherical shells: Provable random-lattice sieving in time $2^{0.292n}$. arXiv:2609.37482v1, 2026.

[DRS14] D. Dadush, O. Regev, and N. Stephens-Davidowitz. On the closest vector problem with a distance guarantee. In *Proceedings of the 29th Conference on Computational Complexity (CCC)*, pages 98–109, 2014.

[DX13] Feng Dai and Yuan Xu. Spherical harmonics, 2013. Chapter 1 of Approximation Theory and Harmonic Analysis on Spheres and Balls. URL: https://arxiv.org/abs/1304.2585, arXiv:1304.2585.

[ECG+25] L. Engelberts, Y. Chen, A. S. Gilani, M. van Hoof, S. Jeffery, and R. de Wolf. An improved quantum algorithm for 3-tuple lattice sieving. arXiv:2510.08473, 2025.

[FGL26] Y. Feng, Y. Gao, and J. Liu. Faster SVP in Polynomial Space, 2026. URL: https://arxiv.org/abs/2609.21612.

[FP85] U. Fincke and M. Pohst. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. *Mathematics of Computation*, 44(170):463–471, 1985. doi:10.1090/s0025-5718-1985-0777278-8.

[GFH26a] Y. Gao, Y. Feng, and H. Hu. Discrete Gaussian sampling meets BDGL decoding: Solving the shortest vector problem in $2^{0.5596n+o(n)}$ time. IACR ePrint 2026/1844, 2026.

[GFH26b] Y. Gao, Y. Feng, and H. Hu. Solving the shortest vector problem in $2^{0.7314n+o(n)}$ time via discrete Gaussian sampling on superlattices. IACR ePrint 2026/1587, 2026.

[GH26] Yiming Gao and Honggang Hu. Faster matrix multiplication leads to faster lattice sieving. Cryptology ePrint Archive, Paper 2026/2622, 2026. URL: https://eprint.iacr.org/2026/2622.

[GHH26] Yiming Gao, Xuyuan Han, and Honggang Hu. Gaussian difference dynamics: Breaking the $2^{n/2}$ barrier for the shortest vector problem and discrete gaussian sampling. Cryptology ePrint Archive, Paper 2026/2690, 2026. URL: https://eprint.iacr.org/2026/2690.

[GMPW20] N. Genise, D. Micciancio, C. Peikert, and M. Walter. Improved discrete Gaussian and subgaussian analysis for lattice cryptography. In *Public-Key Cryptography – PKC 2020, Part I*, volume 12110 of *LNCS*, pages 623–651, 2020. URL: https://par.nsf.gov/servlets/purl/10157346, doi:10.1007/978-3-030-45374-9_21.

[GMSS99] O. Goldreich, D. Micciancio, S. Safra, and J.-P. Seifert. Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. *Information Processing Letters*, 71(2):55–61, 1999. doi:10.1016/S0020-0190(99)00083-6.

[GNR10] N. Gama, P. Q. Nguyen, and O. Regev. Lattice Enumeration Using Extreme Pruning. In *Advances in Cryptology – EUROCRYPT 2010*, pages 257–278. Springer Berlin Heidelberg, 2010. doi:10.1007/978-3-642-13190-5_13.

[GPV08] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC)*, pages 197–206, 2008. URL: https://eccc.weizmann.ac.il/report/2007/133/download.

[Hei21] M. Heiser. Improved quantum hypercone locality sensitive filtering in lattice sieving. IACR ePrint 2021/1295, 2021.

[Hha26a] M. Hhan. Finding a shortest vector and more in $2^{n/2+o(n)}$ time using q-ary coset difference tree. arXiv:2609.02764v1, 2026.

[Hha26b] M. Hhan. Solving the shortest vector problem in $2^{0.6039n}$ time via mid-point hessian. arXiv:2608.02478; IACR ePrint 2026/1597, 2026.

[HKL18] G. Herold, E. Kirshanova, and T. Laarhoven. Speed-ups and time–memory trade-offs for tuple lattice sieving. In *Public-Key Cryptography – PKC 2018, Part I*, volume 10769 of *LNCS*, pages 407–436, 2018. URL: https://eprint.iacr.org/2017/1228, doi:10.1007/978-3-319-76578-5_14.

[HPS11] G. Hanrot, X. Pujol, and D. Stehlé. Algorithms for the shortest and closest lattice vector problems. In *Coding and Cryptology – IWCC 2011*, volume 6639 of *Lecture Notes in Computer Science*, pages 159–190. Springer, 2011. URL: https://xpujol.net/publications/SVPCVP.pdf, doi:10.1007/978-3-642-20901-7_10.

[HS10] G. Hanrot and D. Stehlé. A Complete Worst-Case Analysis of Kannan's Shortest Lattice Vector Algorithm, 2010. URL: https://perso.ens-lyon.fr/damien.stehle/downloads/KANNAN_EXTENDED.pdf.

[Kan83] R. Kannan. Improved algorithms for integer programming and related lattice problems. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing - STOC '83*, STOC '83, pages 193–206. ACM Press, 1983. doi:10.1145/800061.808749.

[Kho03] S. Khot. Hardness of approximating the shortest vector problem in high L_p norms. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 290–297. IEEE Computer Society, 2003. doi:10.1109/SFCS.2003.1238203.

[Kim26] J. Kim. One discrete Gaussian sample in $2^{n/2+o(n)}$ time. arXiv:2608.03220v1, 2026.

[KKK18] Matti Karppa, Petteri Kaski, and Jukka Kohonen. A faster subquadratic algorithm for finding outlier correlations. *ACM Transactions on Algorithms*, 14(3):31:1–31:26, 2018. URL: https://arxiv.org/abs/1510.03895, arXiv:1510.03895, doi:10.1145/3174804.

[KL78] G. A. Kabatiansky and V. I. Levenshtein. Bounds for packings on a sphere and in space. *Problems of Information Transmission*, 14(1):1–17, 1978.

[Kle00] P. Klein. Finding the closest lattice vector when it's unusually close. In *Proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 937–941, 2000.

[KMPM19] E. Kirshanova, E. Mårtensson, E. W. Postlethwaite, and S. R. Moulik. Quantum algorithms for the approximate k-list problem and their application to lattice sieving. In *Advances in Cryptology – ASIACRYPT 2019, Part I*, volume 11921 of *LNCS*, pages 521–551, 2019. URL: https://eprint.iacr.org/2019/1016, doi:10.1007/978-3-030-34578-5_19.

[Kom26] S. D. Kominers. A Fourier consequence for lattice kissing numbers and average contacts. Hexagon 2610.00021v1, https://hexagonmath.org/2610.00021v1, 2026.

[KS99] R. Kumar and D. Sivakumar. A note on the shortest lattice vector problem. In *Proceedings of the 14th Annual IEEE Conference on Computational Complexity (CCC)*, pages 200–204. IEEE Computer Society, 1999.

[Laa16a] T. Laarhoven. *Search Problems in Cryptography: From Fingerprinting to Lattice Sieving*. PhD thesis, Eindhoven University of Technology, 2016. URL: https://pure.tue.nl/ws/files/14673128/20160216_Laarhoven.pdf.

[Laa16b] T. Laarhoven. Tradeoffs for nearest neighbors on the sphere. arXiv:1511.07527v2, 2016.

[LLL82] A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982. doi:10.1007/bf01457454.

[LM00] B. Laurent and P. Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, 28(5):1302–1338, 2000.

[Mon18] A. Montanaro. Quantum-walk speedup of backtracking algorithms. *Theory of Computing*, 14(15):1–24, 2018. URL: https://theoryofcomputing.org/articles/v014a015/, doi:10.4086/toc.2018.v014a015.

[MV10] D. Micciancio and P. Voulgaris. Faster exponential time algorithms for the shortest vector problem. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2010)*, pages 1468–1480. SIAM, 2010. doi:10.1137/1.9781611973075.119.

[NV08] P. Q. Nguyen and T. Vidick. Sieve algorithms for the shortest vector problem are practical. *Journal of Mathematical Cryptology*, 2(2):181–207, 2008. doi:10.1515/JMC.2008.009.

[Ope26] OpenAI. Ten advances in mathematics and theoretical computer science. https://cdn.openai.com/pdf/ten-proofs-oai.pdf, 2026.

[PS09] X. Pujol and D. Stehlé. Solving the shortest lattice vector problem in time $2^{2.465n}$. IACR ePrint 2009/605, 2009.

[PS26] A. Pouly and Y. Shen. Solving the shortest vector problem in $2^{0.63269n+o(n)}$ time on random lattices. In *Advances in Cryptology – EUROCRYPT 2026, Part IV*, volume 16544 of *LNCS*, pages 92–123, 2026. URL: https://eprint.iacr.org/2024/1805, doi:10.1007/978-3-032-25327-9_4.

[Rog50] C. A. Rogers. A note on coverings and packings. *Journal of the London Mathematical Society*, 25(4):327–331, 1950. doi:10.1112/jlms/s1-25.4.327.

[RS62] J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, 6(1):64–94, 1962.

[Sch87] C. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical Computer Science*, 53(2–3):201–224, 1987. doi:10.1016/0304-3975(87)90064-8.

[Sch03] C. Schnorr. Lattice Reduction by Random Sampling and Birthday Methods. In *STACS 2003*, pages 145–156. Springer Berlin Heidelberg, 2003. doi:10.1007/3-540-36494-3_14.

[SE94] C. P. Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Mathematical Programming*, 66(1-3):181–199, 1994. doi:10.1007/bf01581144.

[Ste16] N. Stephens-Davidowitz. Discrete Gaussian sampling reduces to CVP and SVP. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1748–1764, 2016. URL: https://arxiv.org/abs/1506.07490v4.

[ŠVV09] D. Štefankovič, S. Vempala, and E. Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. *Journal of the ACM*, 56(3), 2009. doi:10.1145/1516512.1516520.

[Uzn26] Przemysław Uznański. Rectangular matrix multiplication from polynomial products, 2026. Version 2, Corollary 1.2. URL: https://arxiv.org/abs/2610.10494v2, arXiv:2610.10494.

[vAGN24] J. van Apeldoorn, S. Gribling, and H. Nieuwboer. Basic quantum subroutines: finding multiple marked elements and summing numbers. *Quantum*, 8:1284, 2024. doi:10.22331/q-2024-03-14-1284.

[Wag02] D. Wagner. A generalized birthday problem. In *Advances in Cryptology – CRYPTO 2002*, volume 2442 of *LNCS*, pages 288–304, 2002. doi:10.1007/3-540-45708-9_19.

[ZWW26] Y. Zhou, Z. Wang, and M. Wang. Exact SVP on Haar-random lattices at the heuristic sieving exponents. IACR ePrint 2026/2271, 2026.

[ZY26] Y. Zhang and Y. Yang. A uniform random-lattice tail bound for the SVP kissing-profile parameter. arXiv:2605.21966v3, 2026.

A Refined lattice point counts

This appendix proves a bound on the number of lattice points in any ball centered at the origin. The proof uses an explicit Fourier construction from [Ope26, Chapter 1, Section 4] and its evaluation at the shortest-vector radius by Kominers [Kom26, Lemma 3 and proof of Theorem 1]. We first bound the size of the Fourier construction's negative values over a fixed range of radii. A Gaussian mass bound then extends the count to all larger radii.

A full-rank lattice in $\mathbb{R}^n$ is a set $\mathcal{L} = \mathbf{B}\mathbb{Z}^n$ for an invertible real matrix $\mathbf{B}$. Its determinant is $\det \mathcal{L} = |\det \mathbf{B}|$, its dual is $\mathcal{L}^* = \mathbf{B}^{-T}\mathbb{Z}^n$, and its shortest nonzero vector has Euclidean length

$$\lambda_1(\mathcal{L}) = \min_{\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\}} \|\mathbf{v}\|. \tag{A.1}$$

For $r, T \geq 0$, write $N_{\mathcal{L}}(r)$ for the number of nonzero points within radius r and $C_{\mathcal{L}}(T)$ for the same count at squared radius T.

$$N_{\mathcal{L}}(r) = \big|\{\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\} : \|\mathbf{v}\| \leq r\}\big|, \qquad C_{\mathcal{L}}(T) = N_{\mathcal{L}}(\sqrt{T}). \tag{A.2}$$

Throughout this appendix, ln and $\log_2$ denote natural and binary logarithms, respectively. A factor $A_n = 2^{o(n)}$ means $\log_2 A_n / n \to 0$ as $n \to \infty$.

Theorem A.1 (Uniform centered lattice-point bound)**.** *Put*

$$\beta_{\mathrm{lat}} = \frac{1}{2} \log_2 \left(\frac{2e}{\pi} \right) = 0.3955\ldots. \tag{A.3}$$

There is a sequence $A_n \geq 1$ with $\log_2 A_n = o(n)$, independent of the lattice and the radius, such that every full-rank lattice $\mathcal{L} \subset \mathbb{R}^n$ satisfies

$$1 + C_{\mathcal{L}}(T) \leq A_n \left(\frac{2eT}{\pi \lambda_1(\mathcal{L})^2} \right)^{n/2} \qquad (T \geq \lambda_1(\mathcal{L})^2). \tag{A.4}$$

In radius notation, the same inequality is $1 + N_{\mathcal{L}}(r\lambda_1(\mathcal{L})) \leq A_n 2^{\beta_{\mathrm{lat}} n} r^n$ for $r \geq 1$. Equivalently, for every fixed $\delta > 0$, all sufficiently large dimensions satisfy

$$1 + N_{\mathcal{L}}(r\lambda_1(\mathcal{L})) \leq 2^{(\beta_{\mathrm{lat}} + \delta)n} r^n \qquad (r \geq 1), \tag{A.5}$$

uniformly over all input lattices and all radii.

Spherical-code and packing bounds. The spherical-code approach gives the exponent $\beta_{\mathrm{sph}} = 0.39661$ [Ope26, Chapter 2, Remark 1.7]. The proof of [PS09, Lemma 3] divides lattice points into thin shells and applies a spherical-code bound at the angle determined by each shell radius. For a shell of normalized radius $r \geq 1$, the limiting maximum inner product between normalized vectors is

$$s_r = 1 - \frac{1}{2r^2}. \tag{A.6}$$

This argument needs spherical-code bounds at fixed thresholds, including the kissing threshold $s_1 = 1/2$. The density theorem in [Ope26, Chapter 1, Theorem 1.1] instead concerns the packing density Δ_n in dimension n.

$$\Delta_n \leq 2^{-[\delta_{\mathrm{F}} + o(1)]n}, \qquad \delta_{\mathrm{F}} = \frac{1}{2} \log_2 \frac{2\pi}{e} = 1 - \beta_{\mathrm{lat}}. \tag{A.7}$$

Kominers uses the explicit functions behind this density bound to obtain $\tau(\mathcal{L}) \leq 2^{(\beta_{\mathrm{lat}} + o(1))n}$ for the lattice kissing number $\tau(\mathcal{L}) = N_{\mathcal{L}}(\lambda_1(\mathcal{L}))$ in [Kom26, Theorem 1]. Extending this lattice bound to other shells requires estimates of those functions at larger radii.

Proof outline. The Fourier transform is positive and the function itself is negative on all nonzero lattice vectors after scaling. Poisson summation bounds the sum of their absolute contributions by the value at the origin. A lower bound on each contribution therefore gives an upper bound on the count. The saddle formula from OpenAI and Kominers's evaluation at the shortest-vector radius provide the starting point for this estimate.

We then convert the count on a fixed radius interval into a Gaussian mass bound, using the summation method of [ADRS15, Lemma 4.2]. An elementary bound suffices for the tail outside that interval. Comparing Gaussian masses at different widths then bounds the point counts at larger radii.

Scale the lattice so that $\lambda_1(\mathcal{L}) = 1$. We use the Fourier transform

$$\widehat{f}(\mathbf{y}) = \int_{\mathbb{R}^n} f(\mathbf{x}) e^{-2\pi i \langle \mathbf{x}, \mathbf{y} \rangle} \, d\mathbf{x}. \tag{A.8}$$

For a radial function, we also write $f(r)$ for its value at any point of norm r. A Schwartz function is a smooth function all of whose derivatives decay faster than every inverse polynomial. We write $\mathbf{1}_I$ for the indicator of an interval I.

A.1 The analytic input

The analytic construction uses two signed contributions to a real function H. The first is negative, and the second is positive and supported farther out. The second contribution makes the saddle radii cover the range needed for our point count. We start by specifying the two contributions and the change of variables from the saddle parameter u to the radius.

Fix a sufficiently small $\varepsilon > 0$ before letting n tend to infinity, and put $\nu = n/2$. The cited construction denotes this parameter by λ. Set

$$u_0 = 1 + \varepsilon/4, \qquad u_1 = 1 + \varepsilon/2, \tag{A.9}$$

$$a_- = \varepsilon^2, \qquad a_+ = \ln(1/\varepsilon), \qquad a_B = \varepsilon^{-3}, \tag{A.10}$$

$$\vartheta_\varepsilon = 3\varepsilon/8, \qquad \omega_\varepsilon = e^{-\vartheta_\varepsilon a_B}. \tag{A.11}$$

The two signed shell densities and their real saddle potential are

$$w_s(a) = -\frac{[1 - 2\varepsilon(1+a)]e^{-2a}}{2a^2 \cosh a} \mathbf{1}_{[a_-, a_+]}(a), \tag{A.12}$$

$$w_B(a) = \frac{\omega_\varepsilon}{\cosh a} \mathbf{1}_{[a_B, a_B+1]}(a), \tag{A.13}$$

$$H(u) = \int_0^\infty [w_s(a) + w_B(a)] \, [\cosh(au) - 1] \, da. \tag{A.14}$$

These are equations (34)–(36) of the cited construction, with the cutoff and amplitude notation changed. Write H_s and H_B for the two contributions to H. With Γ denoting the Euler gamma function and $\psi = (\ln \Gamma)'$ its logarithmic derivative, set

$$v_\nu(u) = -\frac{1}{2} \ln \pi + \frac{1}{2} \psi\Big(\frac{\nu(1+u)}{2}\Big) + H'(u), \qquad R_\nu = e^{v_\nu(u_0)}, \tag{A.15}$$

$$P_-(iu) = \varepsilon/4 + (1-u)(1+u)^2, \tag{A.16}$$

$$\mathcal{A}_\nu(u) = -\frac{\nu u}{2} \ln \pi + \ln \Gamma\Big(\frac{\nu(1+u)}{2}\Big) + \nu H(u) - \nu(1+u) v_\nu(u). \tag{A.17}$$

Lemma A.2 (Fourier sign and saddle estimates)**.** *For every sufficiently small fixed $\varepsilon > 0$ and all sufficiently large n, the construction in [Ope26, Chapter 1, Section 4] supplies real radial Schwartz functions f_-, f_+ with*

$$\widehat{f_-} = f_+ > 0, \qquad f_-(r) < 0 \quad (r \geq R_\nu), \tag{A.18}$$

$$f_-(0) = 2\pi^{\nu/2} e^{\nu H(1)}(\varepsilon/4) > 0. \tag{A.19}$$

On $[u_0, u_1]$, $v_\nu'(u)$ is positive and bounded above and away from zero by constants depending only on ε. Uniformly on this interval,

$$-f_-\left(e^{v_\nu(u)}\right) = e^{\mathcal{A}_\nu(u)} \frac{\nu[-P_-(iu)]}{2\pi} \sqrt{\frac{2\pi}{\nu v_\nu'(u)}}\,[1 + o_\varepsilon(1)]. \tag{A.20}$$

The contact-radius evaluation of [Kom26, Lemma 3 and proof of Theorem 1, equation (7)], in the same normalization, gives

$$\frac{1}{n} \ln \frac{f_-(0)}{-f_-(R_\nu)} = \frac{1}{2} \ln \frac{2e}{\pi} + O(\varepsilon) + O_\varepsilon(n^{-1}). \tag{A.21}$$

The constant in $O(\varepsilon)$ is absolute.

Proof. The Fourier identities and origin value are equations (40) and (42), while the global signs are Theorem 4.1 and Corollary 4.9 of the cited chapter. Lemma 4.5, equation (56), supplies the positive curvature bounds on the compact interval. Lemma 4.8, equations (70)–(71), gives (A.20); its relative error is uniform, so the formula also applies when u depends on n. For the final estimate, Kominers uses precisely $f = f_-$ and $R = R_\nu$, with $g_\varepsilon = H$, $u = u_0$, and his parameter $\lambda = \nu$. His Lemma 3 and the cancellation in the proof of his Theorem 1 give (A.21). Thus the contact-value calculation need not be repeated here. □

A.2 Counts on a fixed radius interval

Lemma A.3 (Coverage of a fixed radius interval)**.** *For every fixed $M > 1$, sufficiently small fixed $\varepsilon > 0$ and then sufficiently large n satisfy*

$$[R_\nu, R_\nu\sqrt{M}] \subseteq \{e^{v_\nu(u)} : u_0 \leq u \leq u_1\}. \tag{A.22}$$

Proof. The saddle radius is strictly increasing by Lemma A.2. For fixed ε, the digamma asymptotic gives

$$v_\nu(u_1) - v_\nu(u_0) \longrightarrow \frac{1}{2} \ln \frac{1+u_1}{1+u_0} + H'(u_1) - H'(u_0). \tag{A.23}$$

The second derivative H_s'' is uniformly bounded on $[1, u_1]$ as $\varepsilon \downarrow 0$. Indeed, its integrand in absolute value is bounded by a constant multiple of $e^{-2a}\cosh(u_1 a)/\cosh a$, whose integral is uniformly finite. Thus $H_s'(u_1) - H_s'(u_0) = O(\varepsilon)$. For the remote positive shell, the explicit density gives absolute constants $c, C > 0$ such that

$$H_B'(u_1) \geq c a_B e^{(u_1 - 1 - \vartheta_\varepsilon)a_B} = c a_B e^{1/(8\varepsilon^2)}, \tag{A.24}$$

$$H_B'(u_0) \leq C(a_B + 1) e^{-1/(8\varepsilon^2) + O(\varepsilon)}. \tag{A.25}$$

Consequently, the right-hand side of (A.23) tends to infinity as $\varepsilon \downarrow 0$. Choose ε so that this limit exceeds $\frac{1}{2}\ln M$, and then choose n sufficiently large. Monotonicity proves the inclusion. □

Lemma A.4 (Bound on the radial values over a fixed interval)**.** *For each fixed $M > 1$, the preceding functions satisfy*

$$\frac{f_-(0)}{-f_-(R_\nu\sqrt{T})} \le 2^{[\beta_{\mathrm{lat}}+O_M(\varepsilon)+o_\varepsilon(1)]n}T^{n/2} \qquad (1 \le T \le M), \tag{A.26}$$

uniformly in T, for sufficiently small fixed ε and then sufficiently large n.

Proof. The saddle equation gives the exact finite-dimensional identity

$$\mathcal{A}'_\nu(u) = -\nu(1+u)v'_\nu(u). \tag{A.27}$$

Here the derivative of the gamma term, together with $\nu H'(u) - \frac{\nu}{2}\ln\pi$, equals $\nu v_\nu(u)$ and cancels the corresponding term from the last summand in (A.17). On $[u_0, u_1]$, $-P_-(iu)$ and $v'_\nu(u)$ are positive and bounded above and away from zero for fixed ε. Thus (A.20) gives

$$\ln[-f_-(e^{v_\nu(u)})] = \mathcal{A}_\nu(u) + O_\varepsilon(\ln n) \tag{A.28}$$

uniformly on this interval. Integrating (A.27), rather than differentiating an asymptotic remainder, yields

$$\ln\frac{-f_-(R_\nu)}{-f_-(e^{v_\nu(u)})} \le \nu(1+u_1)[v_\nu(u) - v_\nu(u_0)] + O_\varepsilon(\ln n). \tag{A.29}$$

By Lemma A.3, each $R_\nu\sqrt{T}$ has a saddle $u \in [u_0, u_1]$. Substituting $v_\nu(u) - v_\nu(u_0) = \frac{1}{2}\ln T$ into (A.29) gives the factor

$$T^{\nu(1+u_1)/2} = T^{n/2+\varepsilon n/8}. \tag{A.30}$$

Since $T \le M$, the additional factor is at most $\exp[(\varepsilon n/8)\ln M]$. Combining this with (A.21) proves the bound. The constants in $O_M(\varepsilon)$ may depend on M and are uniform in n and T. □

Lemma A.5 (Centered count on a fixed radius interval)**.** *For every fixed $M > 1$, there is a sequence $B_{n,M} \ge 1$ with $\log_2 B_{n,M} = o(n)$ such that every lattice with $\lambda_1(\mathcal{L}) = 1$ satisfies*

$$C_{\mathcal{L}}(T) \le B_{n,M}2^{\beta_{\mathrm{lat}}n}T^{n/2} \qquad (1 \le T \le M). \tag{A.31}$$

The factor is independent of the lattice and of T.

Proof. Set $F_n(\mathbf{x}) = f_-(R_\nu\mathbf{x})$. Fourier scaling preserves $\widehat{F}_n > 0$, and $F_n(\mathbf{v}) < 0$ for every nonzero $\mathbf{v} \in \mathcal{L}$. Poisson summation therefore gives

$$0 \le \sum_{\mathbf{v}\in\mathcal{L}} F_n(\mathbf{v}) = \frac{1}{\det\mathcal{L}}\sum_{\mathbf{w}\in\mathcal{L}^*}\widehat{F}_n(\mathbf{w}), \tag{A.32}$$

$$\sum_{\mathbf{v}\in\mathcal{L}\setminus\{\mathbf{0}\}}[-F_n(\mathbf{v})] \le F_n(\mathbf{0}). \tag{A.33}$$

All sums converge absolutely because the functions are Schwartz. For every nonzero $\mathbf{v}$ of squared norm at most $T \le M$, Lemma A.4, applied at $\|\mathbf{v}\|^2$, bounds its contribution below by

$$-F_n(\mathbf{v}) \ge F_n(\mathbf{0})2^{-[\beta_{\mathrm{lat}}+O_M(\varepsilon)+o_\varepsilon(1)]n}T^{-n/2}. \tag{A.34}$$

Summing proves the counting estimate with exponent $\beta_{\mathrm{lat}} + O_M(\varepsilon) + o_\varepsilon(1)$. For any desired positive error in the exponent, first choose ε sufficiently small and then take n sufficiently large. The estimate holds uniformly over lattices and radii, because the functions depend only on n, ε and M. The elementary packing bound below makes the supremum of $C_{\mathcal{L}}(T)/(2^{\beta_{\mathrm{lat}}n}T^{n/2})$ finite in each dimension. Taking $B_{n,M}$ to be the larger of one and this supremum gives the stated sequence. □

A.3 Extension to all radii

Lemma A.6 (Gaussian mass bound)**.** *There is a sequence $C_n \geq 1$ with $\log_2 C_n = o(n)$, independent of the lattice, such that every lattice with $\lambda_1(\mathcal{L}) = 1$ satisfies*

$$Z_{\sqrt{e/n}}(\mathcal{L}) \leq C_n 2^{\beta_{\mathrm{lat}} n}, \qquad Z_\sigma(\mathcal{L}) = \sum_{\mathbf{v}\in\mathcal{L}} e^{-\|\mathbf{v}\|^2/(2\sigma^2)} = \rho_{\sqrt{2\pi}\sigma}(\mathcal{L}). \tag{A.35}$$

Proof. We use the Gaussian summation argument of [ADRS15, Lemma 4.2]. Since Lemma A.5 covers a fixed range of radii, we estimate the tail separately. Fix $M = 16e$. Radial integration gives

$$Z_{\sqrt{e/n}}(\mathcal{L}) = 1 + \frac{n}{2e}\int_1^\infty C_{\mathcal{L}}(T) e^{-nT/(2e)}\, dT. \tag{A.36}$$

Write $\varphi(T) = \frac{1}{2}\ln T - T/(2e)$. Its maximum over $T \geq 1$ is $\varphi(e) = 0$. The integral over $[1, M]$ therefore contributes at most $O(n)B_{n,M}2^{\beta_{\mathrm{lat}} n}$.

For the remaining tail, the elementary disjoint-ball estimate gives

$$1 + C_{\mathcal{L}}(T) \leq (2\sqrt{T}+1)^n \leq 3^n T^{n/2} \qquad (T \geq 1). \tag{A.37}$$

Since $\varphi'(T) \leq -15/(32e)$ for $T \geq M$, the tail is at most

$$\frac{n}{2e}3^n \int_{16e}^\infty e^{n\varphi(T)}\, dT \leq \frac{16}{15}\left(12e^{-15/2}\right)^n. \tag{A.38}$$

The base is less than one. The two integrals and the zero vector therefore give the desired bound with $C_n = 2^{o(n)}$. □

Lemma A.7 (Gaussian mass scaling)**.** *For every full-rank lattice $\mathcal{L} \subset \mathbb{R}^n$, every $\sigma > 0$, and every $a \geq 1$,*

$$Z_{a\sigma}(\mathcal{L}) \leq a^n Z_\sigma(\mathcal{L}). \tag{A.39}$$

Proof. The Fourier transform of $e^{-\|\mathbf{x}\|^2/(2\sigma^2)}$ is $(\sqrt{2\pi}\sigma)^n e^{-2\pi^2\sigma^2\|\mathbf{y}\|^2}$. Poisson summation therefore gives

$$\frac{Z_\sigma(\mathcal{L})}{(\sqrt{2\pi}\sigma)^n} = \frac{1}{\det\mathcal{L}} \sum_{\mathbf{w}\in\mathcal{L}^*} e^{-2\pi^2\sigma^2\|\mathbf{w}\|^2}. \tag{A.40}$$

Every summand on the right is nonincreasing in σ. Comparing σ and $a\sigma$ proves the inequality. □

Proof of Theorem A.1. Continue to assume $\lambda_1(\mathcal{L}) = 1$. Lemma A.7, with scale factor $R/\sqrt{e} \geq 1$, gives

$$Z_{R/\sqrt{n}}(\mathcal{L}) \leq (R/\sqrt{e})^n Z_{\sqrt{e/n}}(\mathcal{L}) \qquad (R \geq \sqrt{e}). \tag{A.41}$$

Every point of norm at most R has weight at least $e^{-n/2}$ in $Z_{R/\sqrt{n}}$. Lemma A.6 therefore implies

$$1 + C_{\mathcal{L}}(R^2) \leq e^{n/2} Z_{R/\sqrt{n}}(\mathcal{L}) \tag{A.42}$$

$$\leq e^{n/2}(R/\sqrt{e})^n Z_{\sqrt{e/n}}(\mathcal{L}) \leq C_n 2^{\beta_{\mathrm{lat}} n} R^n. \tag{A.43}$$

For $1 \le R \le \sqrt{e}$, Lemma A.5 bounds the nonzero points by $B_{n,e}2^{\beta_{\mathrm{lat}}n}R^n$. Since $B_{n,e} \ge 1$ and $\beta_{\mathrm{lat}} > 0$, adding the zero vector costs at most a factor two. For a single factor valid in every dimension, define

$$A_n = \max\left\{1, \sup_{\substack{\lambda_1(\mathcal{L})=1 \\ R\ge 1}} \frac{1 + C_{\mathcal{L}}(R^2)}{2^{\beta_{\mathrm{lat}}n}R^n}\right\}. \tag{A.44}$$

The elementary packing estimate (A.37) bounds this supremum by $3^n 2^{-\beta_{\mathrm{lat}}n}$, so it is finite for every n. For each fixed positive exponent margin, the preceding estimates are uniform over the lattice and the entire unbounded radius range for all sufficiently large n. They therefore imply $\log_2 A_n = o(n)$. This definition includes the finitely many dimensions preceding each analysis threshold without choosing a dimension-dependent ε in the construction.

Finally, scale an arbitrary lattice by $1/\lambda_1(\mathcal{L})$ and set $R = \sqrt{T}/\lambda_1(\mathcal{L})$. This proves (A.4). Since $\log_2 A_n = o(n)$, for each fixed $\delta > 0$ one has $A_n \le 2^{\delta n}$ for all sufficiently large n, and $C_{\mathcal{L}}(r^2\lambda_1(\mathcal{L})^2) = N_{\mathcal{L}}(r\lambda_1(\mathcal{L}))$, which also proves (A.5). □

B Certificates for the selected quantum SVP construction

B.1 Exact inputs

All terminating decimals in this appendix denote exact rational inputs or explicitly rounded bounds. The recursions in Section 11.5 specify all intermediate values. The main-list exponent ℓ_{B} and radius ϱ_{B} set the size and radius of the lists that recur between lattice levels. The constants $s_\star, f, \gamma, \chi$ set the scale and the chain of lattices. The two filter indices have exponents s_{src} and s_{root}. The constant ε is the acceptance margin used to start the source recursion. Their exact values are

$$\begin{aligned} \ell_{\mathsf{B}} &= 0.16115531333523020, & \varrho_{\mathsf{B}} &= 0.96024591000000000, & s_\star &= 1.6, \\ f &= 1, & \gamma &= 0.000001, & \chi &= 0.639615, \\ s_{\mathrm{src}} &= 0.24689346860581055, & s_{\mathrm{root}} &= 0.2462966, & \varepsilon &= 0.000002. \end{aligned} \tag{B.1}$$

We use the same β_{cert} as in Section 9.2. In the source recursion use $\tau_{\mathrm{s}} = s_{\mathrm{src}} - 10^{-6}$. For the last join of phase 1, set its preceding population to $p^- = 0.2660631413630137$ and use the prescribed-population rule. The source seed and output exponents are given below.

Type	ξ_i	ℓ_i
U_1	0.12161067072574366	0.18032354268758470
U_2	0.12385816510738004	0.17745829041004027
U_3	0.12300122293101495	0.17917217476277042
U_4	0.12462866771593330	0.17591728519293280
U_5	0.11751375879847031	0.18428977232480756
B	0.13200965364478510	0.16115531333523020

Table 6: Exact source parameters. Row B is phase 6.

At the root use $K_{\mathrm{r}} = 0.49259300765$ and $\mu_0 = 0.0000001$. The upper bound for the contraction when passing the weights down the merge tree is $\theta_\star = 0.169211$.

B.2 Interval checks

Finite preparation tests. The scalar and pair corrections H_1, H_2 are from Lemmas 9.7 and 9.8. Use the formulas in Remark 10.4 with variances v_h, exponents η_h, covariance Σ, and total variance y_{k_i-1}. Lemma 10.3 then bounds $M(S) - H(S)$ for each possible set of shared entries. At an internal merge, the only nonempty proper sets to check omit at least one seed. The two seeds are symmetric, so the first free index can be taken to be $f_0 = 0$. We check one and two free indices directly and compare states separately according to whether the shared set is empty. At the root, apply the same recurrence to the covariance $\boldsymbol{T}^{\mathrm{r}}$, with common exponent $\ell_{\mathsf{B}} - \frac{1}{2}\log_2 X_{\mathsf{B}}$ and total variance $e\chi$. This covers all 65 518 shared sets of sizes $1, \dots, 14$.

To bound the loss caused by selecting one tuple per seed pair, take W to be a nonempty set of nonseed coordinates where two tuples have different vector values. Put $\boldsymbol{K}_W = ((\Sigma^{-1})_{WW})^{-1}$. Before passing down the Gaussian weight, the margin is $-\sum_{h\in W}\eta_h - \frac{1}{2}\log_2\det \boldsymbol{K}_W - H(\boldsymbol{K}_W)$, where H uses scalar elimination in the original order of the children. To compute this margin, start with $(J, L) = (0, 0)$ and process $h = k_i - 1, \dots, 2$ backwards.

$$h \notin W: \quad (J, L) \longmapsto (J + (1 - g_h)^2/q_h, L), \tag{B.2}$$

$$h \in W: \quad p = q_h/(1 + q_h J), \qquad (J, L) \longmapsto \left(\frac{g_h^2 J}{1 + q_h J}, L - \eta_h + \min\{0, -\tfrac{1}{2}\log_2 p\}\right). \tag{B.3}$$

The final L is the margin before including the Gaussian weight. The updates are monotone, so componentwise comparisons apply. Keep $W = \varnothing$ separate from the nonempty sets.

The factors in (11.9) for passing weights down the merge tree are

$$a_{ih} = \frac{(\Sigma \mathbf{1})_h^2}{y_{k_i-1} v_h}, \qquad p_i = a_{i,k_i-1}, \qquad a_i = \max_{h<k_i-1} a_{ih}. \tag{B.4}$$

Multiplying these factors along the recursion gives

$$\pi_i = p_6 \prod_{c=i+1}^{5} p_c \quad (1 \le i \le 5), \qquad \pi_{\mathsf{B}} = \max\{a_6, \pi_1 p_1, \pi_i a_i : 1 \le i \le 5\}. \tag{B.5}$$

The incoming contraction is at most $\theta_\star \pi_i$ for U_i, $\theta_\star \pi_{\mathsf{B}}$ for a main list below the root, and $\theta_\star$ for a main list at the root. All multipliers are below one. Add $\frac{1}{4}\log_2(1-\theta)$ to the final L in (B.3) and subtract 10^{-6}. The logarithmic term uses $a_V \le 1$ in (11.16).

The verifier uses outward-rounded intervals and exact integer changes of coordinates for the pair minima. In the recurrences, use the certified lower endpoints of both state components. Since each update is nondecreasing in those components, any continuation from a smaller state bounds the corresponding continuation from a larger state. A nonpositive lower pivot prevents certification. For a pair, Gauss reduction certifies the integer minimum when $2|K_{12}| \le \min(K_{11}, K_{22})$. Otherwise a finite integer box suffices because the smallest eigenvalue bounds the quadratic form outside the box. Compare empty and nonempty sets of changed or shared entries separately.

Lemma B.1 (Finite parameter inequalities)**.** *For the exact parameters above, the acceptance margins and the margins $M(S) - H(S)$ for shared entries all exceed $1.9 \cdot 10^{-6}$, including the two choices where S omits exactly one child and that child is a seed. After including the Gaussian weights and subtracting 10^{-6}, the margins for different vector values exceed 0.04 below the root and 0.029 in the top main lists. For tuples with no shared entries, all source scalar variances exceed 1.13. The covariance of the final pair has determinant greater than 1.0036 and its quadratic form has minimum greater than 1.0778 over nonzero integer vectors. The corresponding root bounds are 1.1595, 1.0210, and 1.1668. Furthermore,*

$$\theta_{\mathrm{root}} < 0.169211, \qquad \mu_{\mathrm{w}} > 9 \cdot 10^{-8}, \qquad \min_{1 \le |S| \le 14} \{M_{\mathrm{r}}(S) - H_{\mathrm{r}}(S)\} > 0.0148, \tag{B.6}$$

$$\max_i c_i^{\mathrm{rep}} < 0.246893, \quad c_{\mathrm{root}}^{\mathrm{search}} < 0.246297, \qquad c_\star = s_{\mathrm{src}} < 0.246894. \tag{B.7}$$

Proof. Lemma 10.3 and recurrence (B.3) reduce the subset tests to finite recurrences. At the root the recurrence has no seed restriction. For tuples without shared entries, eliminate the partial sums in reverse order, leaving a two-coordinate covariance to check. Equations (11.39)–(11.41) give the access and reporting tests. The verifier `scripts/verify_quantum_sections.py` with option `-section selected` reads the exact inputs, computes the radii from the two recursions, and checks the bounds with 80-decimal interval arithmetic. A complete run includes the pair check from Section 9.2 and writes the results to `scripts/quantum-verification.json`. Section 9.6 describes the archive and the input hashes. □