A Block Enumeration Technique for Lattice Basis Reduction

Huck Bennett*

November 2, 2018

Abstract

We present a technique that yields algorithms for computing (nearly) optimally reduced lattice bases with respect to a wide class of basis quality measures. Namely, we get algorithms for computing lattice bases with minimum orthogonality defect, bases with minimum Gram-Schmidt decay, Babai-optimal bases, and bases with $(1 + \varepsilon)$-approximately minimum Seysen condition number. These algorithms are all slight variations of the same technique, and all run in polynomial time for lattices of fixed rank. To the best of our knowledge, these are the first algorithms for any of these problems to beat the naive approach, which requires exponential time in the bit length of the input.

The technique works by breaking a lattice into pieces according to large gaps in its successive minima, enumerating bases for each of these pieces, and then lifting the bases for each piece to a basis of the whole lattice. This technique may be of independent interest.

1 Introduction

A lattice $\mathcal{L} = \mathcal{L}(B) = \{\sum_{i=1}^{n} a_i \boldsymbol{b}_i : a_1, \ldots, a_n \in \mathbb{Z}\}$ of rank n is the set of integer combinations of some linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^n$ which form the columns of the basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \in \mathbb{R}^{n \times n}$. Lattices are well-studied mathematical objects [CS98], and in the last few decades have found many applications within computer science including in integer programming (e.g. [Len83, Kan87, Dad12]), coding theory (e.g. [LB14, CDL17]), and especially cryptography (e.g. [Ajt96, AD97, GGH97, GPV08, Reg09, Gen09]).

Any given basis B of a lattice $\mathcal{L}$ is not unique, and a common goal is to compute a "reduced basis" of $\mathcal{L}$ which satisfies useful properties such as having short and nearly orthogonal vectors. The theory of basis reduction is intimately related to solving lattice problems both approximately and exactly, and is therefore a major area of study. However, different notions of a "reduced" basis are useful for solving different computational problems on lattices. These notions are either specified in terms of a basis B meeting certain "local", greedy conditions (such as the conditions for a basis to be LLL-, HKZ-, or Minkowski-reduced), or in terms of (approximately) minimizing a "global" *basis quality function* Q (such as the orthogonality defect or Gram-Schmidt decay of the basis).

The two most important computational problems on lattices are the Shortest Vector Problem (SVP), which is to find a shortest non-zero lattice vector, and the Closest Vector Problem (CVP), which is to find a closest lattice vector to an input target vector $\boldsymbol{t}$. A number of algorithms for

*Department of Electrical Engineering and Computer Science, Northwestern University. Email: hbennett@eecs.northwestern.edu. Part of this work appears in the author's Ph.D. thesis, and was completed while the author was a student at New York University and was visiting CWI, Amsterdam.

these problems work by finding reduced lattice bases. In terms of approximation algorithms, the seminal LLL algorithm [LLL82] efficiently computes a basis which yields an approximate solution to SVP. Such LLL-reduced bases can also be used to solve approximate CVP efficiently via Babai's nearest-plane algorithm [Bab86], and have many other applications. In terms of slower but exact algorithms, Kannan's algorithm for exact SVP and CVP [Kan87] relies on computing HKZ-reduced bases [KZ73], which give a greedy way of formalizing of what it means to be a short lattice basis.

The focus of this paper is on giving algorithms for computing bases which are *optimally* reduced with respect to various "global" basis quality functions. The global basis quality functions that we consider are related to fundamental geometric quantities associated with a lattice, and appear in a wide variety of applications from analyzing the security of cryptosystems to solving approximate CVP efficiently to constructing low-distortion mappings between lattices. Algorithms that use bases as preprocessing (such as Babai's algorithm, discussed below) give particular motivation for studying algorithms that compute such optimally reduced bases. Indeed, computing an optimally reduced basis may be computationally expensive, but it can be used many times as preprocessing after being computed once.

We next introduce and survey the functions that we will consider in this paper: the orthogonality defect δ, the multiplicative Gram-Schmidt decay η (together with the closely related Babai basis quality function Q_{Babai}), and the Seysen condition number S.

One general way of formalizing what it means for a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ to be short and orthogonal is according to its *orthogonality defect*, defined as

$$\delta(B) := \prod_{i=1}^{n} \|\boldsymbol{b}_i\| / \|\tilde{\boldsymbol{b}}_i\|, \tag{1}$$

where $\tilde{\boldsymbol{b}}_i$ is the ith Gram-Schmidt vector of B (i.e., $\tilde{\boldsymbol{b}}_i$ is the projection of $\boldsymbol{b}_i$ onto the orthogonal complement of $\text{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1})$). It holds that $\delta(B) \geq 1$ with equality if and only if the vectors in B are pairwise orthogonal; this is known as Hadamard's inequality.

The orthogonality defect is a widely-used measure of the quality of lattice bases, and captures the quality of standard notions of reduced bases. It holds that LLL-reduced bases B have $\delta(B) \leq 2^{n(n-1)/4}$ (see, e.g., [Vaz01]), and that HKZ-reduced bases B have $\delta(B) \leq n^n$ and are within a $n^{n/2}$ factor of optimal (see [LLS90, MG02]). Furthermore, Minkowski-reduced bases (another greedy way of formalizing short lattice bases) have orthogonality defect at most $2^{O(n^2)}$ [vdWG68], a characterization which is crucial to Helfrich's algorithm for computing them [Hel85]. The orthogonality defect also appears directly in applications. For example, the original security analysis of the well-known GGH encryption and signature schemes [GGH97] depends on the difficulty of computing a basis with low orthogonality defect.

Another key basis quality function is the *multiplicative Gram-Schmidt decay*

$$\eta(B) := \max_{1 \leq i \leq j \leq n} \|\tilde{\boldsymbol{b}}_i\| / \|\tilde{\boldsymbol{b}}_j\| \tag{2}$$

of a basis B, which is closely related to Babai's nearest-plane algorithm [Bab86]. The core of Babai's algorithm is an algorithm for solving γ-approximate Closest Vector Problem with Preprocessing (γ-CVPP), where the preprocessing is a basis B.[1] The approximation factor γ guaranteed by Babai's

[1] Babai's algorithm is often presented as a polynomial time algorithm for solving approximate CVP. This interpretation comes from first computing an LLL-reduced basis B, and then using B as the preprocessing for the core CVPP algorithm.

algorithm depends on B, and so a natural objective is to compute a basis B which yields the smallest possible value of γ. Babai's analysis of his algorithm shows that the value of γ is upper bounded by a function $Q_{\text{Babai}}(B)$ of the Gram-Schmidt vectors of B with $\eta(B) \leq Q_{\text{Babai}}(B) \leq \sqrt{n+1} \cdot \eta(B)$, and as such $\eta(B)$ is a strong proxy for how well B works as preprocessing.

Standard notions of basis reduction, such as those with low orthogonality defect, focus on ensuring that the vectors in a basis B are relatively short, but make no explicit guarantees about the lengths of vectors in the *dual basis* $B^* := (B^{-1})^T$ of B. Some applications require short primal bases B, some require short dual bases B^*, and some require B to be well-conditioned so that B and B^* both have short vectors simultaneously. To study the question of finding well-conditioned lattice bases, Seysen [Sey93] defined the matrix condition number

$$S(B) := \max_{i \in [n]} \|\boldsymbol{b}_i\| \|\boldsymbol{b}_i^*\| \ . \tag{3}$$

By the Cauchy-Schwarz inequality, $\|\boldsymbol{b}_i\| \|\boldsymbol{b}_i^*\| \geq |\langle \boldsymbol{b}_i, \boldsymbol{b}_i^* \rangle| = 1$ for primal-dual basis vector pairs $\boldsymbol{b}_i, \boldsymbol{b}_i^*$, so $S(B)$ is a measure of how tight the Cauchy-Schwarz inequality is for such pairs.

Seysen showed how to compute a basis B for every lattice $\mathcal{L}$ of rank n so that $S(B) \leq n^{O(\log n)}$. This improved on a result of Håstad and Lagarias [HL90], who initiated the study of computing well-conditioned lattice bases and gave an upper bound of $\exp(O(n^{1/3}))$ on a slightly different condition number which in turn is upper bounded by $S(B)$. There have been a number of follow-up papers to Seysen's work ([Sey99, ZAM08, Maz10]), and it has appeared frequently in the coding theory literature (e.g., [SMH07, ZMS10, YH15]). In another recent application of Seyesn's work, Bennett, Dadush, and Stephens-Davidowitz [BDS16] showed how to use well-conditioned bases to construct low-distortion mappings between lattices.

Generalized basis reduction

We next define a general notion of basis quality function, and the associated computational problem of finding a basis which minimizes it. This notion captures the orthogonality defect, the Gram-Schmidt decay, and the Seysen condition number described above.

Definition 1.1. A *basis quality function* is a mapping $Q : \cup_{n=1}^{\infty} \mathrm{GL}_n(\mathbb{R}) \to \mathbb{R}^+$ from full-rank lattice bases to the positive reals.

For a basis quality function Q, let $Q(\mathcal{L}) := \inf Q(B)$, where the infimum is taken over all bases B of $\mathcal{L}$, and let $Q(n) := \sup Q(\mathcal{L})$, where the supremum is taken over all lattices of rank at most n. Call a basis B of $\mathcal{L}$ which achieves $Q(B) = Q(\mathcal{L})$ *optimal* with respect to Q. We will study basis quality functions Q for which $Q(n)$ is bounded.

The search version of the associated computational problem is then as follows.

Definition 1.2 (Basis Reduction Problem)**.** The γ-approximate Basis Reduction Problem with respect to a basis quality function Q is the search problem defined as follows. Given a basis B as input, find a basis B' of $\mathcal{L}(B)$ such that $Q(B') \leq \gamma \cdot Q(\mathcal{L}(B))$.

In the case where $\gamma = 1$, we simply refer to the problem as the Basis Reduction Problem with respect to Q.

1.1 Summary of results

The main contribution of this paper is to present a general technique for computing bases which solve the Basis Reduction Problem with respect to any basis quality function Q which meets certain conditions. We use this technique to obtain exact algorithms for computing optimal bases with respect to the orthogonality defect δ and the Gram-Schmidt decay η, as well as the closely related Babai basis quality function Q_{Babai}. We also use this technique to obtain an algorithm for computing $(1+\varepsilon)$-approximately optimal bases with respect to the Seysen condition number S.

To the best of our knowledge, this paper is the first work which considers (nearly) exact algorithms for any of these problems. We summarize these results below.

Theorem 1.3 (Orthogonality defect minimization algorithm)**.** *There exists an algorithm which, on input a basis B of a lattice $\mathcal{L}$ of rank n, outputs a basis B' of $\mathcal{L}$ which satisfies $\delta(B') = \delta(\mathcal{L})$, and which runs in $\delta(n)^{O(n^3)} \leq n^{O(n^4)}$ time and polynomial space.*

Theorem 1.4 (Gram-Schmidt decay minimization algorithm)**.** *There exists an algorithm which, on input a basis B of a lattice $\mathcal{L}$ of rank n, outputs a basis B' of $\mathcal{L}$ which satisfies $\eta(B') = \eta(\mathcal{L})$, and which runs in $(\eta(n) + \sqrt{n})^{O(n^3)} \leq n^{O(n^3 \log n)}$ time and polynomial space.*

Theorem 1.4 also holds when η is replaced by the closely related Babai basis quality function Q_{Babai} (see Section 5.2).

Theorem 1.5 (Seysen condition number minimization approximation scheme)**.** *There exists an algorithm which, on input a basis B of a lattice $\mathcal{L}$ of rank n and a number $\varepsilon \in (0,1)$, outputs a basis B' of $\mathcal{L}$ which satisfies $S(B') \leq (1+\varepsilon) \cdot S(\mathcal{L})$, and which runs in $(n \cdot S(n)/\varepsilon)^{O(n^3)} \leq (n/\varepsilon)^{O(n^3 \log n)}$ time and polynomial space.*

While the dependence on n in the runtimes of these algorithms is high, the algorithms all run in polynomial time on lattices of fixed rank n. Indeed, achieving this is a key goal in designing lattice algorithms, and in particular it is important to avoid dependence on the ratio $\lambda_n(\mathcal{L})/\lambda_1(\mathcal{L})$ of the largest and smallest successive minima of $\mathcal{L}$ in the runtime. In other words, our main goal in this paper is to design fixed-parameter tractable algorithms, where the parameter is the rank n of the lattice. All of our algorithms achieve this, and are also space efficient.

We also emphasize our technique of "breaking a lattice into pieces according to gaps in its successive minima" as a primary conceptual contribution. As this paper shows, this technique works for solving a number of different basis reduction problems, and it may be of independent interest. Indeed, other techniques for breaking a lattice into pieces according to its geometry have been applied very successfully (see the discussion in Section 1.3).

1.2 Techniques

We give an outline of the ideas used in our algorithm while deferring definitions and formal statements. Let $\lambda_1(\mathcal{L}), \ldots, \lambda_n(\mathcal{L})$ denote the successive minima of a lattice $\mathcal{L}$ of rank n (also setting $\lambda_{\max} = \lambda_n$), and let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \in \mathcal{L}$ denote linearly independent vectors which achieve the successive minima of $\mathcal{L}$ (i.e. satisfy $\|\boldsymbol{v}_i\| = \lambda_i(\mathcal{L})$). Let $V_k := \text{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$ and let $\pi_k(\boldsymbol{x})$ denote the projection of $\boldsymbol{x} \in \mathbb{R}^n$ onto $V_k^{\perp}$. Let $\|\boldsymbol{x}\|$ denote the Euclidean norm of a vector $\boldsymbol{x}$, and let $\|B\| := \max_{i \in [n]} \|\boldsymbol{b}_i\|$ denote the maximum norm of a column of a matrix $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.

Let B be a basis of a lattice $\mathcal{L}$ of rank n. The first step behind our algorithm for solving the Basis Reduction Problem with respect to a basis quality function Q is to prove a characterization

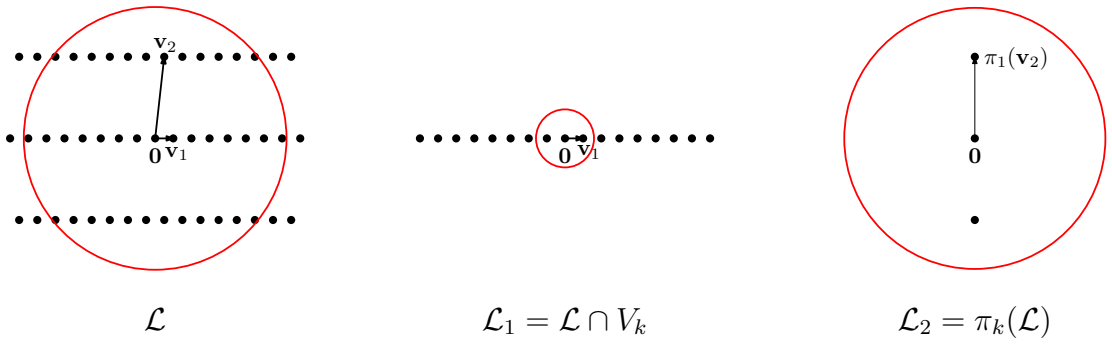

Figure 1: A visual depiction of Theorem 3.4 and Algorithm 1 for a hypothetical basis quality function Q. A lattice $\mathcal{L}$ of rank $n = 2$ with $\|\boldsymbol{v}_{k+1}\|/\|\boldsymbol{v}_k\| = \lambda_{k+1}/\lambda_k > g(2)$ for $k = 1$ (left), together with $\mathcal{L}_1 := \mathcal{L} \cap V_k$ (center), and $\mathcal{L}_2 := \pi_k(\mathcal{L})$ (right). Theorem 3.4, Condition 1 asserts that an optimal basis of B of $\mathcal{L}$ is contained in $\mathcal{B}_2^n(\mathbf{0}, r(Q(n)) \cdot \lambda_{\max}(\mathcal{L}))$ (the Euclidean ball centered at the origin with radius $r(Q(n)) \cdot \lambda_{\max}(\mathcal{L})$), but this ball (shown in red) contains many extraneous vectors of $\mathcal{L}$ as well. Theorem 3.4 further asserts that B has blocks B_1 (a basis of $\mathcal{L}_1$) contained in $\mathcal{L}_1 \cap \mathcal{B}_2^n(\mathbf{0}, r(Q(n)) \cdot \lambda_{\max}(\mathcal{L}_1))$ and B_2 (a basis of $\mathcal{L}_2$) contained in $\mathcal{L}_2 \cap \mathcal{B}(0, r(Q(n)) \cdot \lambda_{\max}(\pi_k(\mathcal{L})))$, each of which contains far fewer extraneous vectors.

which says that if $Q(B) \leq t$ for some $t \geq 0$ then $\|B\| \leq r(t) \cdot \lambda_n(\mathcal{L})$ for some function $r_n(t)$. Suppose that such a characterization holds (which it does for all of the functions Q that we consider), and that $Q(n)$ is bounded (i.e., an optimal basis B with respect to Q always has $Q(B)$ bounded by a function of n). Then we can compute an optimal basis B with respect to Q by enumerating all sequences of vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathcal{L}$ such that $\|\boldsymbol{b}_i\| \leq r(Q(n)) \cdot \lambda_n(\mathcal{L})$ for each $i \in [n]$, and outputting the sequence $B := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ for which $Q(B)$ is minimal among all such sequences which are bases of $\mathcal{L}$.

There is an algorithm which enumerates all lattice vectors in a Euclidean ball of radius $t \cdot \lambda_1(\mathcal{L})$ in $(t \cdot n)^{O(n)}$ time. Using this algorithm, enumerating all lattice vectors in a ball of radius $r(Q(n)) \cdot \lambda_n(\mathcal{L})$ takes $(n \cdot r(Q(n)) \cdot \lambda_n(\mathcal{L})/\lambda_1(\mathcal{L}))^{O(n)}$ time, and enumerating all sequences of n such vectors takes $(n \cdot r(Q(n)) \cdot \lambda_n(\mathcal{L})/\lambda_1(\mathcal{L}))^{O(n^2)}$ time. Unfortunately, the ratio $\lambda_n(\mathcal{L})/\lambda_1(\mathcal{L})$ may be unbounded as a function of n, and therefore exponential in the bit length of the input even for lattices of fixed rank. Therefore this algorithm does not achieve our goal of being fixed-rank tractable.

If $\mathcal{L} = \mathcal{L}_1 \oplus \mathcal{L}_2$ then for many basis quality measures Q we can build an optimal basis by finding separate, orthogonal bases B_1 for $\mathcal{L}_1$ and B_2 for $\mathcal{L}_2$, and concatenating B_1 and B_2 to form a basis of B. The key idea for handling the problem of unbounded $\lambda_n(\mathcal{L})/\lambda_1(\mathcal{L})$ is to "break $\mathcal{L}$ into nearly orthogonal pieces" according to large multiplicative gaps in its successive minima. More precisely, the idea is that when $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L})$ is large for some $k \in [n-1]$, and if a basis quality function Q meets some conditions, then we can find bases B_1, B_2 for $\mathcal{L}_1 := \mathcal{L} \cap V_k, \mathcal{L}_2 := \pi_k(\mathcal{L})$ respectively, and use them as blocks in a basis B of $\mathcal{L}$. Intuitively, "$\mathcal{L} \approx (\mathcal{L} \cap V_k) \oplus \pi_k(\mathcal{L})$" when there is such a gap. See Figure 1 for a visual summary of this technique.

Our algorithm starts by computing the indices $1 \leq k_1 < \cdots < k_{m-1} < n$ which correspond to large gaps in the successive minima, i.e., indices k_i such that $\lambda_{k_i+1}(\mathcal{L})/\lambda_{k_i}(\mathcal{L}) > g(n)$ for some threshold $g(n)$. Then for $i = 1, \ldots, m$, it enumerates bases B_i with low $Q(B_i)$ using the basis enumeration technique described above for each "lattice piece" $\mathcal{L}_i := \pi_{k_{i-1}}(\mathcal{L}) \cap V_{k_i}$ (where $k_0 = 0$

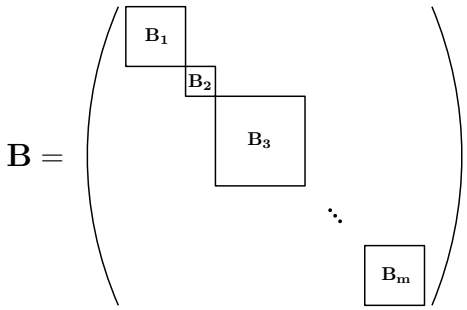

Figure 2: An example of a block projection $B_1, \ldots, B_m$ for an upper triangular matrix B.

and $k_m = n$). Finally, the algorithm lifts these blocks $B_1, \ldots, B_m$ to an optimal basis B of the original lattice $\mathcal{L}$. (These blocks form a *block projection* of B. See Figure 2 and Definition 3.1.) The key advantage of this block enumeration approach is that, unlike for $\mathcal{L}$, the ratios $\lambda_{\max}(\mathcal{L}_i)/\lambda_1(\mathcal{L}_i)$ are bounded by a function of n.

The full formalization of this technique appears in Section 3. This algorithm is presented as Algorithm 1, and the technical conditions for this approach to work appear in the paper's main theorem, Theorem 3.4. In addition to an upper bound on $\|B\|$ for an optimal basis B, Theorem 3.4 requires three further conditions which are roughly as follows: (1) that an optimal basis is such that spans of its basis blocks and spans of vectors achieving its successive minima "agree" (i.e. $\text{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k_i}) = V_{k_i}$ for all $i \in [m]$), (2) that the quality $Q(B)$ of a basis B is lower bounded by the quality $Q(B_i)$ of each of its blocks B_i, and (3) that there is a relatively efficient way to lift blocks $B_1, \ldots, B_m$ of an optimal basis to an optimal basis.

1.3 Related work

At a high level, our algorithm works by enumerating blocks for a basis, and then building a basis out of these blocks. Both of these techniques appear in the basis reduction literature. However, most other algorithms use blocks which are all of the same size, and do not depend on the geometry of the lattice. Our blocks may be of different size, and the way we pick how to form blocks crucially depends on the geometry of the lattice.

The classic enumeration-based algorithms of Kannan [Kan87] for computing HKZ-reduced bases and Helfrich [Hel85] for computing Minkowski-reduced bases work by "repeatedly enumerating the next Gram-Schmidt vector" of a basis. Our algorithm extends this idea by enumerating not just Gram-Schmidt vectors, but potentially larger basis blocks. It then uses these blocks to build a basis of the whole lattice. A number of other basis reduction techniques such as block Korkine-Zolotareff-reduction (BKZ-reduction) [Sch87] and slide-reduction [GN08] similarly work by computing well-reduced blocks and then using them to build a basis of the whole lattice. Helfrich's algorithm is the most similar to ours of any previous algorithm in that it uses repeated enumeration and lifting. It also runs in $2^{O(n^3)}$-time, which is comparable to the running time of our algorithms, showing that hard basis reduction problems may require high runtimes.

The key technique in this paper is to "break a lattice into pieces according to large gaps in its successive minima," which seems natural and should have further applications. Similar ideas have appeared in other work. In particular, an algorithm by Haviv and Regev [HR14] for determining

whether two lattices are isomorphic inspired our algorithm. Their algorithm works by splitting each lattice $\mathcal{L}$ into the sublattice $\mathcal{L} \cap V_k$ and the projected lattice $\pi_k(\mathcal{L})$ whenever there is *any* gap in the successive minima ($\lambda_{k+1}(\mathcal{L}) > \lambda_k(\mathcal{L})$); our algorithm only does so when there is a large gap ($\lambda_{k+1}(\mathcal{L}) \gg \lambda_k(\mathcal{L})$).

Another way to break a lattice into pieces is related to lattice stability. A lattice $\mathcal{L}$ is called *stable* if $\det(\mathcal{L}) = 1$ and $\det(\mathcal{L}') \geq 1$ for all sublattices $\mathcal{L}' \subseteq \mathcal{L}$. The technique in this paper is analogous to the technique of "decomposing a lattice into stable lattices" which has been applied in a number of works, including with great success by Regev and Stephens-Davidowitz in their recent proof of a reverse Minkowski theorem [RS17]. Namely, this technique decomposes a lattice $\mathcal{L}$ by defining a chain of sublattices $\{\mathbf{0}\} = \mathcal{L}_0 \subset \mathcal{L}_1 \subset \cdots \subset \mathcal{L}_m = \mathcal{L}$ of $\mathcal{L}$ called the *canonical filtration* of $\mathcal{L}$ so that the quotient lattices $\mathcal{L}_i/\mathcal{L}_{i-1}$ (defined as the projection of $\mathcal{L}_i$ onto $\mathrm{span}(\mathcal{L}_{i-1})^\perp$) are scalings of stable lattices. In Section 7.1 we discuss the relationship between gaps in the successive minima and lattice stability.

The literature about lattice basis reduction is vast. A good survey of the aspects discussed in this paper is Chapter 7 in the book by Micciancio and Goldwasser [MG02]. This chapter discusses the problem of minimizing a number of basis quality functions including the orthogonality defect δ (which they call the Quasi-Orthogonal Basis Problem), $\sigma(B) := (\sum_{i=1}^n \|\tilde{\boldsymbol{b}}_i\|^2)^{1/2}$ (the Shortest Diagonal Problem), and $M(B) := \max_{i \in [n]} \|\boldsymbol{b}_i\|$ (the Shortest Basis Problem). It also mentions LLL-reduction, HKZ-reduction, Minkowski-reduction, and various relationships between all of these notions of reduction and other lattice problems. It notably does not discuss the Gram-Schmidt decay of a basis. We do not consider the problems of computing bases which minimize σ or M, and note that they are qualitatively different from the basis quality functions that we do consider in that $\sigma(n)$ and $M(n)$ are unbounded. Nevertheless, it is likely that techniques similar to ours would yield algorithms for these problems.

Schnorr [Sch87] also introduced the quantities $\alpha_n := \max \|\boldsymbol{b}_1\|/\|\tilde{\boldsymbol{b}}_n\|$ and $\beta_n := \max(\prod_{i=1}^n \|\tilde{\boldsymbol{b}}_i\| / \prod_{i=n+1}^{2n} \|\tilde{\boldsymbol{b}}_i\|)^{1/n}$, where the maxima are taken over all HKZ-reduced bases B of rank n and HKZ-reduced bases B' of rank $2n$, respectively. Without the "max" over HKZ-reduced bases, $\alpha(B) := \|\boldsymbol{b}_1\|/\|\tilde{\boldsymbol{b}}_n\|$ and $\beta(B) := (\prod_{i=1}^n \|\tilde{\boldsymbol{b}}_i\| / \prod_{i=n+1}^{2n} \|\tilde{\boldsymbol{b}}_i\|)^{1/n}$ are basis quality functions which are related to the Gram-Schmidt decay of the basis. In particular, $\eta(B) \geq \alpha(B)$ for every basis B. (Ajtai [Ajt08] proved a lower bound on α which therefore implies a lower bound on η; see Section 7.2.3.)

1.4 Paper organization

In Section 2 we give relevant definitions and present relevant background material about linear algebra and lattices. In Section 3 we present the main (meta) algorithm and theorems about its correctness and runtime. In the following three sections we show how to instantiate the meta algorithm in Section 3 to give algorithms for computing (nearly) optimal bases with respect to four basis quality functions of interest: the orthogonality defect δ in Section 4, the Gram-Schmidt decay η and the closely related Babai basis quality function Q_{Babai} in Section 5, and the Seysen condition number S in Section 6. In Section 7 we give motivating discussion and examples. Namely, we compare notions of basis reduction and also compare our main technique of "splitting a lattice according to large gaps in its successive minima" to the notion of lattice stability.

1.5 Open questions

A clear open problem is to get more efficient algorithms for any of the basis reduction problems considered in this paper. The main focus of our paper was to give algorithms which run in polynomial time for lattices of fixed rank n. However, the dependence on n (roughly $(Q(n))^{O(n^3)}$ in each case) is almost undoubtedly suboptimal. One could try to improve the runtime of our algorithms by improving our enumeration scheme, or by finding another general technique for basis reduction. A strength of our technique is that it works for a number of different basis quality functions, but analyzing each quality function Q separately may yield faster algorithms.

In addition to algorithms, it would also be interesting to study the complexity of the basis reduction problems presented in this paper. There are natural decision versions of all of these problems (given a basis B and a quality threshold $t \geq 0$ as input, decide whether $Q(\mathcal{L}(B)) \leq t$), and proving NP-hardness for any of them would be interesting. To the best of our knowledge, this is open even for the exact versions of these problems.

An important structural problem is to improve the lower and upper bounds on $Q(n)$ for any of the basis quality functions Q presented in this paper (see Theorem 2.10). It is of particular interest whether $\eta(n) = \text{poly}(n)$ (and hence $Q_{\text{Babai}}(n) = \text{poly}(n)$), since that would imply that Babai's nearest-plane algorithm solves instances of γ-CVPP on lattices of rank n with $\gamma = \gamma(n) = \text{poly}(n)$ given the right basis as preprocessing. Dadush et al. [DRS14] state this as an open question of interest. (The present paper arguably makes progress towards this goal by showing how to compute a Babai-optimal basis for any *given* lattice, but doesn't improve the bound on its guaranteed quality.)

Finally, an important conceptual direction is to study applications of our technique of "breaking a lattice into pieces according to gaps in its successive minima." Hopefully this technique and other notions of "breaking a lattice into pieces of different scale," such as those related to lattice stability, have applications to problems besides basis reduction.

1.6 Acknowledgements

I thank Daniel Dadush and Noah Stephens-Davidowitz for many insightful comments about this paper.

2 Preliminaries

2.1 Linear algebra

The Euclidean norm of a vector $\boldsymbol{x} \in \mathbb{R}^n$ is $\|\boldsymbol{x}\| := \left(\sum_{i=1}^n x_i^2\right)^{1/2}$. For a matrix $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$, let $\|B\| := \max_{i \in [n]} \|\boldsymbol{b}_i\|$. For $r > 0$ and $\boldsymbol{t} \in \mathbb{R}^n$, let $\mathcal{B}_2^n(\boldsymbol{t}, r) := \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{t}\| \leq r\}$ denote the closed Euclidean n-dimensional ball of radius r centered at $\boldsymbol{t}$. Let $\text{GL}_n(\mathbb{R})$ denote the set of invertible $n \times n$ matrices.

Let $\pi_S(\boldsymbol{x})$ denote the projection of a vector $\boldsymbol{x}$ onto a linear subspace S. Given a full-rank matrix $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ and $1 \leq k \leq n$, let $\pi_k^{(B)}(\boldsymbol{x})$ denote projection of $\boldsymbol{x}$ onto the orthogonal complement of $\text{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$. I.e., $\pi_k^{(B)}(\boldsymbol{x}) := \pi_{\text{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)^\perp}(\boldsymbol{x})$. Let $\pi_0^{(B)}$ denote the identity mapping.

Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ be linearly independent vectors. We define their *Gram-Schmidt orthogonalization* (or *Gram-Schmidt vectors*) $\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n$ as

$$\begin{aligned} \tilde{\boldsymbol{b}}_1 &:= \boldsymbol{b}_1 \ , \\ \tilde{\boldsymbol{b}}_i &:= \boldsymbol{b}_i - \sum_{j=1}^{i-1} \mu_{i,j} \tilde{\boldsymbol{b}}_j \qquad\qquad \text{for } i \leq 2 \leq n \ , \end{aligned}$$

where $\mu_{i,j} := \langle \boldsymbol{b}_i, \tilde{\boldsymbol{b}}_j \rangle / \langle \tilde{\boldsymbol{b}}_j, \tilde{\boldsymbol{b}}_j \rangle$. I.e. for all $i \in [n]$, $\tilde{\boldsymbol{b}}_i := \pi_{i-1}^{(B)}(\boldsymbol{b}_i)$, where $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.

A useful way to organize the information in the Gram-Schmidt orthogonalization is in terms of the *QR-decomposition* of a matrix. The QR-decomposition decomposes a matrix $B \in \mathrm{GL}_n(\mathbb{R})$ into $B = QR$, where $Q = (\boldsymbol{q}_1, \ldots, \boldsymbol{q}_n)$ is an orthogonal matrix with $\boldsymbol{q}_i := \tilde{\boldsymbol{b}}_i / \|\tilde{\boldsymbol{b}}_i\|$ and R is an upper triangular matrix with $R_{i,i} = \|\tilde{\boldsymbol{b}}_i\|$ and $R_{i,j} = \mu_{j,i}$ for $i < j$. (An *orthogonal matrix* $O \in \mathrm{GL}_n(\mathbb{R})$ is a matrix such that $O^T O = I_n$, where I_n is the $n \times n$ identity matrix.)

2.2 Lattice definitions and facts

A matrix $U \in \mathbb{Z}^{n \times n}$ is *unimodular* if $\det(U) = \pm 1$. Two bases $B, B' \in \mathrm{GL}_n(\mathbb{R})$ generate the same lattice (satisfy $\mathcal{L}(B) = \mathcal{L}(B')$) if and only if $B' = BU$ for some unimodular matrix U. The *determinant* of a lattice $\mathcal{L}$ is defined as $\det(\mathcal{L}) := |\det(B)|$ for some basis B of $\mathcal{L}$. Because all bases of $\mathcal{L}$ are equivalent up to right multiplication by unimodular matrices, $\det(\mathcal{L})$ is well-defined. The *rank* of a lattice is the column rank of a basis that generates it. A lattice $\mathcal{L}'$ is called a *sublattice* of $\mathcal{L}$ if $\mathcal{L}' \subseteq \mathcal{L}$.

Given a lattice $\mathcal{L}$ of rank n, for $1 \leq i \leq n$ define the *ith successive minimum* of $\mathcal{L}$ as

$$\lambda_i(\mathcal{L}) := \min\{r > 0 : \dim(\mathrm{span}(\mathcal{L} \cap \mathcal{B}(\boldsymbol{0}, r))) \geq i\} \ .$$

In other words $\lambda_1(\mathcal{L})$ is the length of a shortest non-zero vector $\boldsymbol{v}_1 \in \mathcal{L}$, $\lambda_2(\mathcal{L})$ is the length of a shortest vector $\boldsymbol{v}_2 \in \mathcal{L}$ which is linearly independent of $\boldsymbol{v}_1$, and so on. For convenience, we also define $\lambda_0(\mathcal{L}) := 0$, and we will sometimes write $\lambda_{\max}(\mathcal{L})$ to denote $\lambda_n(\mathcal{L})$, where n is the rank of $\mathcal{L}$. Let *vectors that achieve the successive minima* of a lattice $\mathcal{L}$ of rank n denote linearly independent vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \in \mathcal{L}$ such that for all $i \in [n]$, $\|\boldsymbol{v}_i\| = \lambda_i(\mathcal{L})$. When the underlying lattice is clear from context, we will use V_k to denote $\mathrm{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$ and π_k to denote projection onto $V_k^{\perp}$.

Minkowski's Second Theorem relates the successive minima of a lattice to its determinant. See, e.g., [MG02].

Theorem 2.1 (Minkowski's Second Theorem)**.** *Given a lattice $\mathcal{L}$ of rank n, $\prod_{i=1}^{n} \lambda_i(\mathcal{L}) \leq n^{n/2} \cdot \det(\mathcal{L})$.*

Let $\mathrm{dist}(\boldsymbol{t}, \mathcal{L}) := \min_{\boldsymbol{x} \in \mathcal{L}} \|\boldsymbol{x} - \boldsymbol{t}\|$ denote the distance of a vector $\boldsymbol{t}$ to a lattice $\mathcal{L}$. The *covering radius* of a lattice $\mu(\mathcal{L}) := \max_{\boldsymbol{x} \in \mathrm{span}(\mathcal{L})} \mathrm{dist}(\boldsymbol{x}, \mathcal{L})$ denotes the distance of the farthest point $\boldsymbol{x} \in \mathrm{span}(\mathcal{L})$ from $\mathcal{L}$. The following well-known bound relates the covering radius and successive minima of a lattice. See, e.g., [MG02].

Theorem 2.2. *For every lattice $\mathcal{L}$ of rank n, $\mu(\mathcal{L}) \leq \frac{\sqrt{n}}{2} \cdot \lambda_n(\mathcal{L})$.*

Every lattice $\mathcal{L}$ has an associated *dual lattice* $\mathcal{L}^*$ defined as

$$\mathcal{L}^* := \{\boldsymbol{x} \in \mathrm{span}(\mathcal{L}) : \forall \boldsymbol{y} \in \mathcal{L},\ \langle \boldsymbol{x}, \boldsymbol{y} \rangle \in \mathbb{Z}\} \ .$$

Given a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$, we define the *dual basis* of B as $B^* := (B^{-1})^T = (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*)$. It holds that if B is a basis of $\mathcal{L}$, then the dual basis B^* of B generates the dual lattice $\mathcal{L}^*$ of $\mathcal{L}$.

So-called *transference theorems* relate the parameters of a lattice to the parameters of its dual lattice. In seminal work, Banaszczyk [Ban93] proved the following relationship between the successive minima of a lattice and its dual.

Theorem 2.3 (Banaszczyk's transference theorem [Ban93])**.** *For any lattice $\mathcal{L}$ of rank n and any $1 \leq k \leq n$,*

$$1 \leq \lambda_k(\mathcal{L})\lambda_{n-k+1}(\mathcal{L}^*) \leq n \ .$$

Lastly, we define the important notion of a size-reduced basis.

Definition 2.4. Call a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ with Gram-Schmidt vectors $\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n$ *size-reduced* if $|\mu_{i,j}| \leq \frac{1}{2}$ for all $1 \leq j < i \leq n$, where $\mu_{i,j} := \langle \boldsymbol{b}_i, \tilde{\boldsymbol{b}}_j \rangle / \langle \tilde{\boldsymbol{b}}_j, \tilde{\boldsymbol{b}}_j \rangle$.

Any basis B can be efficiently "size-reduced," i.e. converted to a size-reduced basis B' of the same lattice which has the same Gram-Schmidt vectors.

2.3 Lattice problems

The two most important computational problems on lattices are the Shortest Vector Problem (SVP) and the Closest Vector Problem (CVP).

Definition 2.5. For $\gamma \geq 1$, the γ-approximate Shortest Vector Problem (γ-SVP) is the search problem defined as follows. Given a basis B of a lattice $\mathcal{L}$ as input, output a non-zero vector $\boldsymbol{x} \in \mathcal{L} \setminus \{\boldsymbol{0}\}$ such that $\|\boldsymbol{x}\| \leq \gamma \cdot \lambda_1(\mathcal{L})$.

Definition 2.6. For $\gamma \geq 1$, the γ-approximate Closest Vector Problem (γ-CVP) is the search problem defined as follows. Given a basis B of a lattice $\mathcal{L}$ and a vector $\boldsymbol{t}$ as input, output a vector $\boldsymbol{x} \in \mathcal{L}$ such that $\|\boldsymbol{x} - \boldsymbol{t}\| \leq \gamma \cdot \mathrm{dist}(\boldsymbol{t}, \mathcal{L})$.

We also define a variant of CVP which uses preprocessing. This is the variant of CVP which Babai's algorithm solves.

Definition 2.7. For $\gamma \geq 1$, the γ-approximate Closest Vector Problem with Preprocessing (γ-CVPP) is the problem of finding a preprocessing function P and an algorithm $\mathcal{A}$ which work as follows. Given a basis B of a lattice $\mathcal{L}$ as input, P outputs a new representation of $\mathcal{L}$. Given $P(\mathcal{L})$ and a vector $\boldsymbol{t}$ as input, $\mathcal{A}$ computes a vector $\boldsymbol{x} \in \mathcal{L}$ such that $\|\boldsymbol{x} - \boldsymbol{t}\| \leq \gamma \cdot \mathrm{dist}(\boldsymbol{t}, \mathcal{L})$.

The Successive Minima Problem (SMP) is the problem of finding vectors which achieve the successive minima of a lattice. It is a natural generalization of SVP.

Definition 2.8. For $\gamma \geq 1$, the γ-approximate Successive Minima Problem (γ-SMP) is the search problem defined as follows. Given a basis B of a lattice $\mathcal{L}$ of rank n as input, output linearly independent vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \in \mathcal{L}$ such that for all $i \in [n]$, $\|\boldsymbol{v}_i\| \leq \gamma \cdot \lambda_i(\mathcal{L})$.

In the case where $\gamma = 1$, we simply refer to the above problems as SVP, CVP, CVPP, and SMP, respectively.

2.3.1 Basis quality functions of interest

This paper gives algorithms for finding optimal bases with respect to four specific basis quality functions. These algorithms serve as examples of its main technique. We define these basis quality functions below.

Definition 2.9. Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a lattice basis, let $\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n$ be its Gram-Schmidt orthogonalization, and let $B^* = (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*)$ be its dual basis. We define four basis quality functions of interest as follows:

1. The *orthogonality defect* $\delta(B) := \prod_{i=1}^n \|\boldsymbol{b}_i\| / \|\tilde{\boldsymbol{b}}_i\| = (\prod_{i=1}^n \|\boldsymbol{b}_i\|) / \det(\mathcal{L})$.
2. The *multiplicative Gram-Schmidt decay* $\eta(B) := \max_{1 \le i \le j \le n} \|\tilde{\boldsymbol{b}}_i\| / \|\tilde{\boldsymbol{b}}_j\|$.
3. The *Babai basis quality function*

$$Q_{\text{Babai}}(B) := \left(1 + \max_{i \in [n]} \frac{\sum_{j=1}^i \|\tilde{\boldsymbol{b}}_j\|^2}{\|\tilde{\boldsymbol{b}}_i\|^2}\right)^{1/2}. \tag{4}$$

4. The *Seysen condition number* $S(B) := \max_{i \in [n]} \|\boldsymbol{b}_i\| \|\boldsymbol{b}_i^*\|$.

We next give the best known upper bound on $Q(n)$ for each $Q \in \{\delta, \eta, Q_{\text{Babai}}, S\}$ defined above.

Theorem 2.10. *The following bounds hold:*

1. $\delta(n) \le n^n$ *([LLS90]).*
2. $\eta(n) \le n^{O(\log n)}$ *([LLS90]).*
3. $Q_{Babai}(n) \le n^{O(\log n)}$ *(since $Q_{Babai}(B) \le O(\sqrt{n}) \cdot \eta(B)$ for bases B of rank n).*
4. $S(n) \le n^{O(\log n)}$ *([Sey93]).*

The upper bounds in Theorem 1.1 appear in the runtimes stated in the theorems in Section 1.1. All of these upper bounds use HKZ-bases (modified HKZ-bases in the case of $S(n)$), but nevertheless HKZ-bases are not always optimal (see Section 7.2.2). As mentioned in Section 1.5, improving the upper bounds (and finding matching lower bounds) for each of the above basis quality functions is an interesting open problem.

2.4 Enumeration of short lattice vectors

The following theorem, which gives a low-space algorithm for enumerating short vectors in a lattice, is a straightforward corollary of Kannan's Algorithm [Kan87]. See, e.g., [HR14, Corollary 2.16].

Theorem 2.11. *There exists an algorithm which, on input a basis B of a lattice $\mathcal{L}$ of rank n and $t \ge 1$, enumerates all vectors $\boldsymbol{x} \in \mathcal{L}$ such that $\|\boldsymbol{x}\| \le t \cdot \lambda_1(\mathcal{L})$ using $(t \cdot n)^{O(n)}$ time and polynomial space.*

Corollary 2.12. *There exists an algorithm which, on input a basis B of a lattice $\mathcal{L}$ of rank n and $t \geq 1$, enumerates all bases B' of $\mathcal{L}$ such that $\|B'\| \leq t \cdot \lambda_1(\mathcal{L})$ using $(t \cdot n)^{O(n^2)}$ time and polynomial space.*

Proof. The algorithm works as follows. Enumerate all sequences of vectors $\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n \in \mathcal{L}$ with $\|\boldsymbol{b}'_i\| \leq t \cdot \lambda_1(\mathcal{L})$ for each $i \in [n]$ in $(t \cdot n)^{O(n^2)}$ time using Theorem 2.11. For each such $B' := (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n)$, check whether B' is a basis of $\mathcal{L}$ (this can be done efficiently by checking whether $B^{-1}B'$ is unimodular), and if so output it. □

Minimizing some basis quality measures depends on a basis having short Gram-Schmidt vectors. It will therefore be useful to enumerate bases with short Gram-Schmidt vectors. However, in general infinitely many bases of the same lattice have the same Gram-Schmidt vectors, so it is impossible to enumerate all bases with Gram-Schmidt vectors shorter than a give length.

On the other hand, if we define equivalence classes of bases, where two bases are equivalent if and only if they generate the same lattice and have the same Gram-Schmidt vectors, then we can enumerate a representative from each equivalence class of bases with all Gram-Schmidt vectors shorter than a given length. For basis quality measures which *only* depend on the Gram-Schmidt vectors, such as the Gram-Schmidt decay η, this will allow us to enumerate an optimal basis.

Definition 2.13. Call two bases $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ and $B' = (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n)$ *Gram-Schmidt equivalent* if $\mathcal{L}(B) = \mathcal{L}(B')$ and $\forall i \in [n]$, $\tilde{\boldsymbol{b}}_i = \tilde{\boldsymbol{b}}'_i$.

Lemma 2.14. *For every basis B of a lattice of rank n there exists a basis B' which is Gram-Schmidt equivalent to B and which satisfies $\|B'\| \leq \sqrt{n} \cdot \max_{i \in [n]} \|\tilde{\boldsymbol{b}}_i\|$.*

Proof. Let B' denote B after being size-reduced. Then $\tilde{\boldsymbol{b}}'_i = \tilde{\boldsymbol{b}}_i$ and $\|\boldsymbol{b}'_i\| \leq (\|\tilde{\boldsymbol{b}}_i\|^2 + \frac{1}{4}\sum_{j=1}^{i-1} \|\tilde{\boldsymbol{b}}_j\|^2)^{1/2} \leq \sqrt{i} \cdot \|\tilde{\boldsymbol{b}}_i\|$ for all $i \in [n]$. It follows that $\|B'\| \leq \sqrt{n} \cdot \max_{i \in [n]} \|\tilde{\boldsymbol{b}}_i\|$. □

Combining Lemma 2.14 with Corollary 2.12 will allow us to enumerate (up to Gram-Schmidt equivalence) all bases B of a lattice with $\max_{i \in [n]} \|\tilde{\boldsymbol{b}}_i\|$ less than a given length t.

2.5 Properties of successive minima

A *lattice subspace* of a lattice $\mathcal{L}$ is a linear subspace spanned by vectors of $\mathcal{L}$. The following fact says that if S is a lattice subspace then the intersection of S with $\mathcal{L}$ and the projection of $\mathcal{L}$ onto $S^\perp$ are also lattices. See, e.g., [Dad12, Lemma 2.4.1].

Fact 2.15. *Let $\mathcal{L}$ be a lattice, and let S be a lattice subspace of $\mathcal{L}$. Then $\mathcal{L} \cap S$ and $\pi_{S^\perp}(\mathcal{L})$ are lattices.*

In particular, Fact 2.15 asserts that $\mathcal{L} \cap V_k$ and $\pi_k(\mathcal{L}) = \pi_{V_k^\perp}(\mathcal{L})$ are lattices, where $V_k = \operatorname{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$ is the span of vectors achieving the first k successive minima of $\mathcal{L}$. The next fact relates a basis B whose first k vectors have the same span as a lattice subspace S to bases of $\mathcal{L} \cap S$ and $\pi_{S^\perp}(\mathcal{L})$.

Fact 2.16. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis of a lattice $\mathcal{L}$, and assume that B satisfies $\operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k) = S$ for some lattice subspace S of $\mathcal{L}$. Then $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ is a basis of $\mathcal{L} \cap S$, and $(\pi_{S^\perp}(\boldsymbol{b}_{k+1}), \ldots, \pi_{S^\perp}(\boldsymbol{b}_n))$ is a basis of $\pi_{S^\perp}(\mathcal{L})$.*

We next relate the successive minima of a lattice to the successive minima of $\mathcal{L} \cap V_k$ and $\pi_k(\mathcal{L})$. These relationships are folklore.

Lemma 2.17. *Let $\mathcal{L}$ be a lattice of rank $n \geq 2$, and let $k \in \{0, \ldots, n-1\}$. Then:*

1. *If $k \geq 1$ then for every $j \in [k]$, $\lambda_j(\mathcal{L} \cap V_k) = \lambda_j(\mathcal{L})$.*
2. *For every $j \in [n-k]$,*

$$\lambda_{k+j}(\mathcal{L}) - \frac{\sqrt{k}}{2} \cdot \lambda_k(\mathcal{L}) \leq \lambda_j(\pi_k(\mathcal{L})) \leq \lambda_{k+j}(\mathcal{L}) \ . \tag{5}$$

Proof. The statements are clear when $k = 0$ (recalling that we defined π_0 to be the identity map and defined $\lambda_0 = 0$), so assume that $k \geq 1$. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ be vectors achieving the successive minima of $\mathcal{L}$.

For every $j \in [k]$, we have that $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_j \in \mathcal{L} \cap V_k$ so $\lambda_j(\mathcal{L} \cap V_k) \leq \lambda_j(\mathcal{L})$. On the other hand, $\mathcal{L} \cap V_k \subseteq \mathcal{L}$, so $\lambda_j(\mathcal{L} \cap V_k) \geq \lambda_j(\mathcal{L})$. This proves item 1.

We have that $\pi_k(\boldsymbol{v}_{k+1}), \ldots, \pi_k(\boldsymbol{v}_{k+j}) \in \pi_k(\mathcal{L})$ are linearly independent by the linear independence of $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$. Therefore $\lambda_j(\pi_k(\mathcal{L})) \leq \max_{\ell \in [j]} \|\pi_k(\boldsymbol{v}_{k+\ell})\| \leq \lambda_{k+j}(\mathcal{L})$, proving the upper bound in item 2.

Let $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-k} \in \pi_k(\mathcal{L})$ be vectors achieving the successive minima of $\pi_k(\mathcal{L})$, and let $j \in [n-k]$. By the triangle inequality and the definition of the covering radius, there exist liftings $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j \in \mathcal{L}$ of $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_j$ such that $\pi_k(\boldsymbol{x}_\ell) = \boldsymbol{u}_\ell$, and $\|\boldsymbol{x}_\ell\| \leq \|\boldsymbol{u}_\ell\| + \mu(\mathcal{L} \cap V_k)$ for every $\ell \in [j]$. By the linear independence of $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-k}$, we therefore have that $\lambda_{k+j}(\mathcal{L}) \leq \max_{\ell \in [j]} \|\boldsymbol{x}_\ell\| \leq \max_{\ell \in [j]} \|\boldsymbol{u}_\ell\| + \mu(\mathcal{L} \cap V_k) = \lambda_j(\pi_k(\mathcal{L})) + \mu(\mathcal{L} \cap V_k)$. Finally, using Theorem 2.2 and item 1, $\mu(\mathcal{L} \cap V_k) \leq \frac{\sqrt{k}}{2} \lambda_k(\mathcal{L} \cap V_k) = \frac{\sqrt{k}}{2} \lambda_k(\mathcal{L})$. We then have $\lambda_{k+j}(\mathcal{L}) \leq \lambda_j(\pi_k(\mathcal{L})) + \frac{\sqrt{k}}{2} \lambda_k(\mathcal{L})$. Subtracting $\frac{\sqrt{k}}{2} \lambda_k(\mathcal{L})$ from both sides proves the lower bound in item 2. □

3 Main algorithm

A key definition for our approach is a *block projection*, which we define as follows.

Definition 3.1. Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \in \mathbb{R}^{n \times n}$ be a full-rank matrix. A sequence of matrices $B_1, \ldots, B_m$ is a *block projection* of B if there exist $0 = k_0 < k_1 < \cdots < k_m = n$ such that for all $i \in [m]$, $B_i = (\pi^{(B)}_{k_{(i-1)}}(\boldsymbol{b}_{k_{(i-1)}+1}), \ldots, \pi^{(B)}_{k_{(i-1)}}(\boldsymbol{b}_{k_i}))$.

A block projection of an upper triangular matrix B corresponds to a block diagonal matrix obtained by selecting disjoint blocks centered on the main diagonal of B (see Figure 2). In general there are many different block decompositions of the same matrix (corresponding to the number of blocks m and indices $k_1, \ldots, k_{m-1}$), and many bases of the same lattice share a given block decomposition. The Gram-Schmidt vectors $\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n$ of a basis B represent the important special case of a block projection where $m = n$ and $k_i = i$ for $i \in [n]$.

We will use the following equivalence between the subspaces spanned by vectors achieving the successive minima of a lattice (with a sufficiently large gap in its successive minima), and vectors achieving the successive minima of a projection of the lattice.

Lemma 3.2. *Let $\mathcal{L}$ be a lattice of rank n, and let $k = n$ or let $1 < k < n$ and be such that $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > \sqrt{k}/2 + 1$. Then for every $1 \le i < k$, $\mathrm{span}(\tilde{\boldsymbol{v}}_{i+1}, \ldots, \tilde{\boldsymbol{v}}_k) = \mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i})$, where $\tilde{\boldsymbol{v}}_1, \ldots, \tilde{\boldsymbol{v}}_n$ is the Gram-Schmidt orthogonalization of vectors achieving the successive minima of $\mathcal{L}$, and $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-1}$ are vectors achieving the successive minima of $\pi_i(\mathcal{L})$.*

Proof. The case when $k = n$ is clear. By definition $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i} \in V_i^{\perp}$ and $\tilde{\boldsymbol{v}}_{i+1}, \ldots, \tilde{\boldsymbol{v}}_k \in V_k \cap V_i^{\perp}$, so it suffices to show that $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i} \in V_k$. Indeed, then $\mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i}) \subseteq \mathrm{span}(\tilde{\boldsymbol{v}}_{i+1}, \ldots, \tilde{\boldsymbol{v}}_k) = V_k \cap V_i^{\perp}$, which implies that $\mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i}) = \mathrm{span}(\tilde{\boldsymbol{v}}_{i+1}, \ldots, \tilde{\boldsymbol{v}}_k)$ since the subspaces have the same dimension.

Suppose for contradiction that $\mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i}) \not\subseteq \mathrm{span}(\tilde{\boldsymbol{v}}_{i+1}, \ldots, \tilde{\boldsymbol{v}}_k)$. Then $\boldsymbol{u}_j \notin V_k$ for some $j \in [k-i]$. Using the triangle inequality and the definition of the covering radius, there exists a lifting $\boldsymbol{x} \in \mathcal{L} \setminus V_k$ of $\boldsymbol{u}_j$ such that $\pi_i(\boldsymbol{x}) = \boldsymbol{u}_j$, and $\|\boldsymbol{x}\| \le \|\boldsymbol{u}_j\| + \mu(\mathcal{L} \cap V_i)$. Using Theorem 2.2, Lemma 2.17 item 1, and the upper bound in Equation (5), $\|\boldsymbol{u}_j\| + \mu(\mathcal{L} \cap V_i) \le \|\boldsymbol{u}_j\| + \frac{\sqrt{i}}{2} \cdot \lambda_i(\mathcal{L} \cap V_i) \le \lambda_{i+j}(\mathcal{L}) + \frac{\sqrt{i}}{2} \cdot \lambda_i(\mathcal{L}) \le (\frac{\sqrt{k}}{2} + 1) \cdot \lambda_k(\mathcal{L})$. But because $\boldsymbol{x} \notin V_k$, this implies that $\lambda_{k+1}(\mathcal{L}) \le \|\boldsymbol{x}\| \le (\frac{\sqrt{k}}{2} + 1) \cdot \lambda_k(\mathcal{L})$, which is a contradiction. □

We next prove a result which will let us upper bound the ratios $\lambda_{\max}(\mathcal{L}_i)/\lambda_1(\mathcal{L}_i)$ for the lattices $\mathcal{L}_i$ in Algorithm 1.

Lemma 3.3. *Let $\mathcal{L}$ be a lattice of rank n, let $0 \le i < k \le n$, and let $g(n) \ge 10\sqrt{n}$. Assume that $\lambda_{j+1}(\mathcal{L})/\lambda_j(\mathcal{L}) \le g(n)$ for all $i < j < k$, and that:*

1. *Either $i = 0$, or $i > 0$ and $\lambda_{i+1}(\mathcal{L})/\lambda_i(\mathcal{L}) > g(n)$.*
2. *Either $k = n$, or $k < n$ and $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > g(n)$.*

Then $\mathcal{L}' := \pi_i(\mathcal{L}) \cap V_k$ is a lattice of rank $k - i$, and $\lambda_{\max}(\mathcal{L}')/\lambda_1(\mathcal{L}') \le 2 \cdot (g(n))^{k-i-1}$.

Proof. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ denote vectors which achieve the successive minima of $\mathcal{L}$, and let $\tilde{\boldsymbol{v}}_1, \ldots, \tilde{\boldsymbol{v}}_n$ be their Gram-Schmidt orthogonalization. Because $V_i = \mathrm{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$ is a lattice subspace of $\mathcal{L}$, it holds that $\pi_i(\mathcal{L}) = \pi_{V_i^{\perp}}(\mathcal{L})$ is a lattice by Fact 2.15. Let $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-i}$ be vectors which achieve the successive minima of $\pi_i(\mathcal{L})$. Then by Lemma 3.2, $\mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i}) = \mathrm{span}(\tilde{\boldsymbol{v}}_{i+1}, \ldots, \tilde{\boldsymbol{v}}_n)$. Since $\mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i})$ is a lattice subspace of $\pi_i(\mathcal{L})$, it holds that $\mathcal{L}' = \pi_i(\mathcal{L}) \cap V_k = \pi_i(\mathcal{L}) \cap \mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-i})$ is a lattice again by Fact 2.15.

We next upper bound the ratio of the maximum and minimum successive minima of $\mathcal{L}'$.

$$
\begin{aligned}
\frac{\lambda_{\max}(\mathcal{L}')}{\lambda_1(\mathcal{L}')} &= \frac{\lambda_{k-i}(\pi_i(\mathcal{L}) \cap \mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-1}))}{\lambda_1(\pi_i(\mathcal{L}) \cap \mathrm{span}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-1}))} \\
&= \frac{\lambda_{k-i}(\pi_i(\mathcal{L}))}{\lambda_1(\pi_i(\mathcal{L}))} && \text{(Lemma 2.17, Item 1)} \\
&\le \frac{\lambda_k(\mathcal{L})}{\lambda_{i+1}(\mathcal{L}) - \frac{\sqrt{i}}{2} \cdot \lambda_i(\mathcal{L})} && \text{(Lemma 2.17, Item 2)} \\
&\le \frac{\lambda_k(\mathcal{L})}{(1 - \frac{\sqrt{i}}{2 \cdot g(n)}) \cdot \lambda_{i+1}(\mathcal{L})} && (i = \lambda_i = 0, \text{ or } i > 0 \text{ and } \frac{\lambda_{i+1}(\mathcal{L})}{\lambda_i(\mathcal{L})} > g(n)) \\
&\le 2 \cdot \frac{\lambda_k(\mathcal{L})}{\lambda_{i+1}(\mathcal{L})} && (g(n) \ge 10\sqrt{n}) \\
&\le 2 \cdot (g(n))^{k-i-1} \ . && (\lambda_{j+1}(\mathcal{L})/\lambda_j(\mathcal{L}) \le g(n) \text{ for } i < j < k)
\end{aligned}
$$

Algorithm 1: BlockEnumOpt(B)

Input: A basis B for a lattice $\mathcal{L}$ of rank n. The algorithm is also parameterized by a basis quality function Q, numbers $s, R > 0$ and a procedure Lift that will be set in the analysis.

Output: A basis B' that satisfies $Q(B') = Q(\mathcal{L})$ when the conditions for Theorems 3.4 or Theorem 3.6 are met. A basis B' that achieves $S(B') \leq (1+\varepsilon) \cdot S(\mathcal{L})$ when $Q = S$ and the algorithm is parameterized as in Section 6.2.

Compute vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ which achieve the successive minima of $\mathcal{L}$.
Let $1 \leq k_1 < \cdots < k_{m-1} < n$ denote the indices such that $\lambda_{k_i+1}(\mathcal{L})/\lambda_{k_i}(\mathcal{L}) > s$.
Let $k_0 = 0$ and let $k_m = n$.
foreach *sequence* $B_1, \ldots, B_m$ *where each* B_i *is a basis of* $\mathcal{L}_i := \pi_{k_{i-1}}(\mathcal{L}) \cap V_{k_i}$ *that satisfies* $\|B_i\| \leq R \cdot \lambda_{\max}(\mathcal{L}_i)$ **do**
  $B' \leftarrow \text{Lift}(B, (B_1, \ldots, B_m))$.
  Compute $Q(B')$.
end
Output the basis B' computed above for which $Q(B')$ is minimum.

The first inequality additionally uses the fact that either $i = \lambda_i = 0$, or that $\lambda_{i+1}(\mathcal{L}) > g(n) \cdot \lambda_i(\mathcal{L}) \geq 10\sqrt{n} \cdot \lambda_i(\mathcal{L})$ so that the denominator of the right-hand side is positive. □

3.1 The main algorithm

We next present the main algorithm and theorem.

Theorem 3.4. *Let $Q(B)$ be a polynomial-time computable basis quality function with $Q(n) \leq 2^{\text{poly}(n)}$, and suppose that there exist monotone functions $g(n) \geq 10\sqrt{n}$ and $r_n(t)$ such that the following conditions hold for every lattice $\mathcal{L}$ of rank n:*

1. *(Short optimal bases): Given a basis B of $\mathcal{L}$ and a number $t > 0$, if $Q(B) \leq t$ then $\|B\| \leq r_n(t) \cdot \lambda_{\max}(\mathcal{L})$.*
2. *(Blocks follow the successive minima): Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ be vectors that achieve the successive minima of $\mathcal{L}$. Then there exists a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ of $\mathcal{L}$ that satisfies $Q(B) = Q(\mathcal{L})$, and is such that for all $k \in [n-1]$ with $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > g(n)$, $\text{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k) = \text{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$.*
3. *(Block lower bound): Let B be a basis of $\mathcal{L}$. Then for all block decompositions $B_1, \ldots, B_m$ of B, $Q(B) \geq \max_{i \in [m]} Q(B_i)$.*
4. *(Block lifting): There exists an algorithm, Lift_Q, which runs in $n^{O(n)}$ time and which, on input a basis B of $\mathcal{L}$ and a block decomposition $B_1, \ldots, B_m$ of a (possibly different) basis of $\mathcal{L}$, outputs a basis B' of $\mathcal{L}$ with minimum $Q(B')$ over all bases of $\mathcal{L}$ which have $B_1, \ldots, B_m$ as a block decomposition.*

Set $s = s(n) := g(n)$, $R = R(n) := r_n(Q(n))$, and Lift *to be* Lift_Q *in* BlockEnumOpt. *Then on input a basis B of a lattice $\mathcal{L}$ of rank n* BlockEnumOpt *computes a basis B' of $\mathcal{L}$ with $Q(B') = Q(\mathcal{L})$, and runs in $(r_n(Q(n)) \cdot g(n)^{n-1})^{O(n^2)}$ time and polynomial space.*

Proof. We start by proving correctness. It suffices to show that one of the sequences of blocks $B_1, \ldots, B_m$ considered in Algorithm 1 is a block projection of a basis A of $\mathcal{L}$ which is optimal with respect to Q. Indeed, then by Condition 4 $\textsc{Lift}_Q(B, (B_1, \ldots, B_m))$ outputs a (possibly different) optimal basis B' of $\mathcal{L}$.

By Condition 2, there exists a basis $A = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ of $\mathcal{L}$ which satisfies $Q(A) = Q(\mathcal{L})$ and $\operatorname{span}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k) = V_k$ for all $k \in [n-1]$ with $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > g(n)$. Let $k_0 < \cdots < k_m$ be as defined in Algorithm 1. We will show that $A_i := (\pi_{k_{(i-1)}}(\boldsymbol{a}_{k_{(i-1)}+1}), \ldots, \pi_{k_{(i-1)}}(\boldsymbol{a}_{k_i}))$ is a basis of $\mathcal{L}_i := \pi_{k_{(i-1)}}(\mathcal{L}) \cap V_{k_i}$ which satisfies $\|A_i\| \leq r_n(Q(n)) \cdot \lambda_{\max}(\mathcal{L}_i)$.

Using Fact 2.16, it holds that for $i \in [m]$, $(\pi_{k_{(i-1)}}(\boldsymbol{a}_{k_{(i-1)}+1}), \ldots, \pi_{k_{(i-1)}}(\boldsymbol{a}_n))$ is a basis of $\pi_{k_{(i-1)}}(\mathcal{L})$ since $V_{k_{(i-1)}}$ is a lattice subspace of $\mathcal{L}$, and therefore, using Fact 2.16 again, A_i is a basis of $\mathcal{L}_i$ since $\operatorname{span}(\pi_{k_{(i-1)}}(\mathcal{L})) \cap V_{k_i} = \pi_{k_{(i-1)}}(\mathcal{L}) \cap \operatorname{span}(\pi_{k_{(i-1)}}(\boldsymbol{v}_{k_{(i-1)}+1}), \ldots, \pi_{k_{(i-1)}}(\boldsymbol{v}_{k_i}))$ is a lattice subspace of $\pi_{k_{(i-1)}}(\mathcal{L})$. Finally, $Q(A_i) \leq Q(A) \leq Q(n)$ for all $i \in [m]$ by Condition 3 and the optimality of A, which implies that $\|A_i\| \leq r_n(Q(n)) \cdot \lambda_{\max}(\mathcal{L}_i)$ by Condition 1, as needed.

We next analyze the time and space complexity of Algorithm 1. Computing the indices $0 = k_0 < \cdots < k_m = n$ and subspaces V_{k_i} amounts to computing the successive minima of $\mathcal{L}$. This can be done in $n^{O(n)}$ time and polynomial space by reducing the problem of computing successive minima to CVP ([Mic08] gives an efficient reduction between these problems), and then running Kannan's Algorithm [Kan87] for solving CVP.

By Corollary 2.12, enumerating all bases B_i of $\mathcal{L}_i$ with $\|B_i\| \leq r_n(Q(n)) \cdot \lambda_{\max}(\mathcal{L}_i)$ for $i \in [m]$ takes $(r_n(Q(n)) \cdot n \cdot \lambda_{\max}(\mathcal{L}_i)/\lambda_1(\mathcal{L}_i))^{O((k_i - k_{(i-1)})^2)}$ time, and so enumerating all sequences $B_1, \ldots, B_m$ of the form in Algorithm 1 can be done in at most

$$\begin{aligned}\prod_{i=1}^{m}(r_n(Q(n)) \cdot n \cdot \lambda_{\max}(\mathcal{L}_i)/\lambda_1(\mathcal{L}_i))^{O((k_i-k_{(i-1)})^2)} &\leq \prod_{i=1}^{m}(r_n(Q(n)) \cdot g(n)^{n-1})^{O((k_i-k_{(i-1)})^2)} \\ &= (r_n(Q(n)) \cdot g(n)^{n-1})^{\sum_{i=1}^{m} O((k_i-k_{(i-1)})^2)} \\ &\leq (r_n(Q(n)) \cdot g(n)^{n-1})^{O(n^2)}\end{aligned} \tag{6}$$

time, where the first inequality follows by Lemma 3.3.

The right-hand side of Equation (6) also upper bounds the number of iterations of the "for loop" in Algorithm 1. Each loop iteration takes $n^{O(n)}$ time to execute $\textsc{Lift}$ and $\operatorname{poly}(n)$ time to compute $Q(B')$. The overall time spent on the for loop in Algorithm 1 is then at most

$$(r_n(Q(n)) \cdot g(n)^{n-1})^{O(n^2)} \cdot (n^{O(n)} + \operatorname{poly}(n)) = (r_n(Q(n)) \cdot g(n)^{n-1})^{O(n^2)},$$

which dominates the runtime of Algorithm 1. The enumeration algorithm also uses space which is polynomial in the length of the input, i.e., in the lattice rank n and the length of the input basis $\|B\|$. □

3.2 Permutation invariance

We next describe another useful notion of equivalence for some basis quality functions. Call a basis quality function Q *permutation invariant* if $Q(B) = Q(B')$ for any bases $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ and $B' = (\boldsymbol{b}_{\pi(1)}, \ldots, \boldsymbol{b}_{\pi(n)})$ where π is a permutation of $[n]$. The orthogonality defect δ and Seysen condition number S are permutation invariant.

Call a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *sorted* if $\|\boldsymbol{b}_1\| \leq \cdots \leq \|\boldsymbol{b}_n\|$. For permutation invariant basis quality functions we can consider sorted bases without loss of generality. This is useful in the next

proposition which says that, if the quality of a basis suitably depends on how well the lengths of its basis vectors approximate the successive minima (i.e. on $\|\boldsymbol{b}_i\|/\lambda_i$), then Q satisfies conditions (1) and (2) in Theorem 3.4.

Proposition 3.5. *Let Q be a permutation invariant basis quality function, and suppose that there exists a function r such that, for every sorted basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ of a lattice $\mathcal{L}$ of rank n, it holds that if $Q(B) \leq t$ then $\|\boldsymbol{b}_i\| \leq Q(B)$ for all $i \in [n]$.*

1. *(Short optimal bases): Given a basis B of $\mathcal{L}$ and a number $t > 0$, if $Q(B) \leq t$ then $\|B\| \leq r_n(t) \cdot \lambda_{\max}(\mathcal{L})$.*
2. *(Blocks follow the successive minima): Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ be vectors which achieve the successive minima of $\mathcal{L}$. Then there exists a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ of $\mathcal{L}$ which satisfies $Q(B) = Q(\mathcal{L})$, and is such that for all $k \in [n-1]$ with $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > r_n(Q(n))$, $\operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k) = \operatorname{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$.*

Proof. Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis of a lattice $\mathcal{L}$ of rank n, and let $B' = (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n)$ be sorted and such that there exists a permutation π such that for every $i \in [n]$, $\boldsymbol{b}_i = \boldsymbol{b}'_{\pi(i)}$.

Suppose that $\|B\| \geq r_n(t) \cdot \lambda_{\max}(\mathcal{L})$ for some $t > 0$. Then $\|\boldsymbol{b}'_n\| = \|B\| \geq r_n(t) \cdot \lambda_n(\mathcal{L})$, which implies by assumption that $Q(B) = Q(B') \geq t$. Therefore, Q satisfies condition (1).

Suppose now that B is an optimal basis with respect to Q. Then $Q(B') = Q(B) = Q(\mathcal{L}) \leq Q(n)$. Then $\|\boldsymbol{b}'_i\| \leq r_n(Q(n)) \cdot \lambda_i(\mathcal{L})$ for all $i \in [n]$, and so, if $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > r_n(Q(n))$ for some $k \in [n-1]$, it must hold that $\operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k) = V_k$. Therefore, Q satisfies condition (2). □

3.3 A variant of the main theorem

We also present a variant of Theorem 3.4 which replaces the condition that "good bases are short" with the condition that "good bases have short Gram-Schmidt vectors" together with a condition that Q only depends on the Gram-Schmidt vectors. Note that this second condition is crucial since we cannot in general enumerate all bases with Gram-Schmidt vectors shorter than some threshold (there are in general infinitely many bases of a lattice with the same Gram-Schmidt vectors). This variant will be useful for handling the Gram-Schmidt decay η and Babai basis quality function Q_{Babai} in Section 5.

Theorem 3.6. *Theorem 3.4 holds if we instead set $t = t(n) := \sqrt{n} \cdot r_n(Q(n))$ in Algorithm 1 and replace Condition 1 with Conditions 1'* (a) *and* (b) *below:*

1'. a. *(Short Gram-Schmidt vectors): Given a basis B of $\mathcal{L}$ with Gram-Schmidt vectors $\tilde{\boldsymbol{b}}_1, \ldots, \tilde{\boldsymbol{b}}_n$ and a number $t > 0$, if $Q(B) \leq t$ then $\max_{i \in [n]} \|\tilde{\boldsymbol{b}}_i\| \leq r_n(t) \cdot \lambda_{\max}(\mathcal{L})$.*

   b. *(Dependence only on Gram-Schmidt vectors): If two bases B and B' are Gram-Schmidt equivalent (as defined in Definition 2.13) then $Q(B) = Q(B')$.*

Proof. By Condition 1'(b), it suffices to show that $B' = \textsc{Lift}(B, (B_1, \ldots, B_m))$ for one of the sequences of blocks $B_1, \ldots, B_m$ considered in Algorithm 1 is Gram-Schmidt equivalent to a basis A of $\mathcal{L}$ with $Q(A) = Q(\mathcal{L})$. To do this, it in turn suffices to show that there exists a block projection $A_1, \ldots, A_m$ of A such that A_i and B_i are Gram-Schmidt equivalent for all $i \in [m]$.

Let $k_0 < \cdots < k_m$ be as defined in Algorithm 1. By the same reasoning as in the proof of Theorem 3.4, there exists a block projection $A_1, \ldots, A_m$ of A such that A_i is a basis of $\mathcal{L}_i := \pi_{k_{i-1}}(\mathcal{L}) \cap V_{k_i}$. Furthermore, we again have that $Q(A_i) \leq Q(A) \leq Q(n)$ for all $i \in [m]$ by Condition 3 and the optimality of A, which by Condition 1'(a) implies that

$$\max_{k_{(i-1)}+1 \leq j \leq k_i} \|\tilde{\boldsymbol{a}}_j\| \leq r_n(Q(n)) \cdot \lambda_{\max}(\mathcal{L}_i) \tag{7}$$

where $\tilde{\boldsymbol{a}}_{k_{(i-1)}+1}, \ldots, \tilde{\boldsymbol{a}}_{k_i}$ are the Gram-Schmidt vectors of A_i. Finally, by Lemma 2.14 and Equation (7) it holds that for each $i \in [m]$ there is a basis B_i of $\mathcal{L}_i$ with

$$\|B_i\| \leq \sqrt{k_i - k_{i-1}} \cdot \max_{k_{(i-1)}+1 \leq j \leq k_i} \|\tilde{\boldsymbol{a}}_j\| \leq \sqrt{n} \cdot r_n(Q(n)) \cdot \lambda_{\max}(\mathcal{L}_i) \ .$$

The runtime analysis is the same since the added $\sqrt{n}$ factor in t gets absorbed into the $O(\cdot)$ in the runtime exponent. □

Finally, we note that Condition 1'(b) in Theorem 3.6 implies Condition 4 in Theorem 3.4. If only the Gram-Schmidt vectors are relevant, then *any* lifting of the vectors in the blocks $B_1, \ldots, B_m$ to vectors in the full lattice will result in a basis $B' = (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n)$ of the same quality. I.e., all that matters for lifting the jth vector of the ith block $(B_i)_j$ to a basis vector $\boldsymbol{b}'_{k_i+j}$ of B' is that $\pi_{k_i}(\boldsymbol{b}'_{k_i+j}) = (B_i)_j$. Such liftings are easy to compute by writing $(B_i)_j = \sum_{i=1}^{n} a_i \pi_{k_i}(\boldsymbol{b}_i)$ with $a_1, \ldots, a_n \in \mathbb{Z}$, and then setting $\boldsymbol{b}'_{k_i+j} := \sum_{i=1}^{n} a_i \boldsymbol{b}_i$, where $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is the input basis.

4 Orthogonality defect minimization

In this section we show how to apply the results in Section 3 to get an algorithm for computing optimal bases with respect to the orthogonality defect δ. We start by showing a block lower bound for δ.

Lemma 4.1. *Let $B_1, \ldots, B_m$ be a block projection of a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$. Then $\delta(B) \geq \max_{i \in [m]} \delta(B_i)$.*

Proof. By the definition of a block projection, there exist $0 = k_0 < k_1 < \cdots < k_m = n$ such that for every $i \in [m]$, $B_i = (\pi^{(B)}_{k_{(i-1)}}(\boldsymbol{b}_{k_{(i-1)}+1}), \ldots, \pi^{(B)}_{k_{(i-1)}}(\boldsymbol{b}_{k_i}))$. Then for every $i \in [m]$, $\delta(B) = \prod_{j=1}^{n} \|\boldsymbol{b}_j\| / \|\tilde{\boldsymbol{b}}_j\| \geq \prod_{j=k_{(i-1)}+1}^{k_i} \|\boldsymbol{b}_j\| / \|\tilde{\boldsymbol{b}}_j\| \geq \prod_{j=k_{(i-1)}+1}^{k_i} \|\pi^{(B)}_{k_{(i-1)}}(\boldsymbol{b}_j)\| / \|\tilde{\boldsymbol{b}}_j\| = \delta(B_i)$, as needed. □

Given a lattice $\mathcal{L}$ and a lattice subspace S of $\mathcal{L}$, we define the "CVP-lifting" of a vector $\boldsymbol{y} \in \pi_{S^\perp}(\mathcal{L})$ to be a shortest vector $\boldsymbol{y}' \in \mathcal{L}$ with the same projection onto $S^\perp$ as $\boldsymbol{y}$. The idea of CVP-lifting appears in Helfrich's algorithm for computing Minkowski-reduced bases [Hel85], where she calls it "correctly 'deprojecting'" a vector.

Definition 4.2 (CVP-lifting)**.** Let $\mathcal{L}$ be a lattice and let S be a lattice subspace of $\mathcal{L}$. Given a vector $\boldsymbol{y} \in \pi_{S^\perp}(\mathcal{L})$, define CVP-Lift$(\mathcal{L}, S, \boldsymbol{y}) := \arg\min_{\boldsymbol{y}' \in \mathcal{L}} \{\|\boldsymbol{y}'\| : \pi_{S^\perp}(\boldsymbol{y}') = \boldsymbol{y}\}$.

Computing CVP-Lift$(\mathcal{L}, S, \boldsymbol{y})$ amounts to solving an instance of CVP on $\mathcal{L} \cap S$. More specifically, computing CVP-Lift$(\mathcal{L}, S, \boldsymbol{y})$ can be done as follows. Let $k = \dim(S)$ and let

$B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis of $\mathcal{L}$ where $B_1 = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ is a basis of $\mathcal{L} \cap S$. (Such a basis B always exists and can be computed efficiently; see [Mic08, Lemma 1].) Then computing CVP-Lift$(\mathcal{L}, S, \boldsymbol{y})$ amounts to solving the CVP instance $(B_1, \sum_{i=k+1}^{n} a_i \boldsymbol{b}_i)$, where $a_{k+1}, \ldots, a_n$ are defined by the relation $\boldsymbol{y} = \sum_{i=k+1}^{n} a_i \cdot \pi_k(\boldsymbol{b}_i)$.

Using CVP-Lifting, we then get an algorithm for lifting block projections to bases with the lowest possible orthogonality defect.

Lemma 4.3. *There exists an algorithm,* $\textsc{Lift}_\delta$*, which runs in* $n^{O(n)}$ *time and polynomial space and which, on input a basis* B *of* $\mathcal{L}$ *and a block projection* $B_1, \ldots, B_m$ *of a (possibly different) basis of* $\mathcal{L}$*, outputs a basis* B' *of* $\mathcal{L}$ *with minimal* $Q(B')$ *over all bases of* $\mathcal{L}$ *which have* $B_1, \ldots, B_m$ *as a block projection.*

Proof. Let $0 = k_0 < k_1 < \cdots < k_m = n$ be the indices such that for $i \in [m]$, $B_i = (\pi_{k_{i-1}}(\boldsymbol{b}_{k_{i-1}+1}), \ldots, \pi_{k_{i-1}}(\boldsymbol{b}_{k_i}))$. Let $(B_i)_j$ denote the jth basis vector in B_i. $\textsc{Lift}_\delta$ then computes $B' = (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n)$ as follows. For $1 \leq j \leq k_1$, it sets $\boldsymbol{b}'_j := (B_1)_j$, and for $i > 1$ and $1 \leq j \leq k_i - k_{i-1}$ it sets $\boldsymbol{b}'_j := \text{CVP-Lift}(\mathcal{L}, \text{span}(B_1, \ldots, B_{i-1}) \cap \mathcal{L}, (B_i)_j)$.

It is straightforward to check that B' is a basis of $\mathcal{L}$ and that it satisfies $\delta(B') \leq \delta(B'')$ for all bases B'' of $\mathcal{L}$ which have $B_1, \ldots, B_m$ as a block projection. Furthermore, $\textsc{Lift}_\delta$ performs fewer than n CVP-liftings, each of which corresponds to solving an instance of CVP on a lattice of rank less than n. These operations dominate the time complexity $\textsc{Lift}_\delta$, which is then at most $n \cdot n^{O(n)} = n^{O(n)}$ by using Kannan's Algorithm [Kan87] to solve CVP. Similarly, because Kannan's Algorithm uses polynomial space, $\textsc{Lift}_\delta$ also uses at most polynomial space.

□

We conclude with a proof that the orthogonality defect δ meets the conditions for a basis quality function specified in Theorem 3.4.

Proof of Theorem 1.3. From its definition as $\delta(B) := (\prod_{i=1}^{n} \|\boldsymbol{b}_i\|)/\det(\mathcal{L})$, it is clear that δ is a permutation invariant basis quality function. Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a sorted basis of a lattice $\mathcal{L}$. Then $\delta(B) = (\prod_{i=1}^{n} \|\boldsymbol{b}_i\|)/\det(\mathcal{L}) \geq \prod_{i=1}^{n} \|\boldsymbol{b}_i\|/\lambda_i(\mathcal{L}) \geq \max_{i \in [n]} \|\boldsymbol{b}_i\|/\lambda_i(\mathcal{L})$. Therefore, if $\|\boldsymbol{b}_i\| \geq t \cdot \lambda_i(\mathcal{L})$ for some $i \in [n]$ it holds that $\delta(B) \geq t$. So, by Proposition 3.5, δ satisfies conditions (1) and (2) in Theorem 3.4 with $r_n(t) = t$ and $g(n) = r_n(\delta(n)) = \delta(n)$.

Furthermore, δ satisfies Theorem 3.4, condition (3) by Lemma 4.1 and Theorem 3.4, condition (4) by Lemma 4.3.

□

5 Gram-Schmidt decay minimization

In this section we show that the multiplicative Gram-Schmidt decay η and the closely related Babai basis quality function Q_{Babai} meet the conditions of Theorem 3.6.

5.1 Minimizing η

We first show that since the last Gram-Schmidt vector $\|\tilde{\boldsymbol{b}}_n\|$ of a basis B cannot be too large, neither can any of the other Gram-Schmidt vectors $\|\tilde{\boldsymbol{b}}_i\|$ unless $\eta(B)$ is large.

Lemma 5.1. *Let* $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *be a basis of a lattice* $\mathcal{L}$*. Then* $\|\tilde{\boldsymbol{b}}_n\| \leq (n-1)^{(n-1)/2} \cdot \lambda_n(\mathcal{L})$.

Proof. Let $\mathcal{L}' = \mathcal{L}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1})$. We then have that

$$\|\tilde{\boldsymbol{b}}_n\| = \frac{\det(\mathcal{L})}{\det(\mathcal{L}')} \leq \frac{\det(\mathcal{L}) \cdot (n-1)^{(n-1)/2}}{\prod_{i=1}^{n-1} \lambda_i(\mathcal{L}')} \leq \frac{\det(\mathcal{L}) \cdot (n-1)^{(n-1)/2}}{\prod_{i=1}^{n-1} \lambda_i(\mathcal{L})} \leq (n-1)^{(n-1)/2} \cdot \lambda_n(\mathcal{L}) \ .$$

The first inequality uses Minkowski's Second Theorem (Theorem 2.1), the second inequality uses the fact that $\lambda_i(\mathcal{L}) \leq \lambda_i(\mathcal{L}')$ for all $i \in [n-1]$, and the third inequality uses the fact that $\det(\mathcal{L}) \leq \prod_{i=1}^{n} \lambda_i(\mathcal{L})$. □

By the definition of η we get the following corollary.

Corollary 5.2. *For every basis B of $\mathcal{L}$, if $\max_{i\in[n]} \|\tilde{\boldsymbol{b}}_i\| \geq t \cdot (n-1)^{(n-1)/2} \cdot \lambda_n(\mathcal{L})$ then $\eta(B) \geq t$.*

We next show that bases with low $\eta(B)$ must have vectors which roughly follow the span of vectors which achieve successive minima.

Lemma 5.3. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis of a lattice $\mathcal{L}$ with $\eta(B) < \frac{\lambda_{k+1}(\mathcal{L})}{\lambda_k(\mathcal{L})} - \frac{\sqrt{k}}{2}$ for some $k \in [n-1]$. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ be vectors that achieve the successive minima of $\mathcal{L}$, and let $V_k := \operatorname{span}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$. Then $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in V_k$.*

Proof. Suppose not. Let $i_1 \in [n]$ be the minimum index such that $\boldsymbol{b}_{i_1} \notin V_k$. Then $i_1 \leq k$ and $\pi_k(\boldsymbol{b}_{i_1}) = \tilde{\boldsymbol{b}}_{i_1}$, where π_k denotes projection onto $V_k^{\perp}$. By Theorem 2.2, there exists a vector $\boldsymbol{y} \in \mathcal{L} \cap V_k$ such that $\|\pi_{V_k}(\boldsymbol{b}_{i_1} - \boldsymbol{y})\| \leq \frac{\sqrt{k}}{2} \cdot \lambda_k(\mathcal{L})$. Because $\boldsymbol{b}_{i_1} - \boldsymbol{y} \in \mathcal{L} \setminus V_k$, we then have by the triangle inequality that $\lambda_{k+1}(\mathcal{L}) \leq \|\boldsymbol{b}_{i_1} - \boldsymbol{y}\| \leq \frac{\sqrt{k}}{2} \cdot \lambda_k(\mathcal{L}) + \|\tilde{\boldsymbol{b}}_{i_1}\|$, and therefore that $\|\tilde{\boldsymbol{b}}_{i_1}\| \geq \lambda_{k+1}(\mathcal{L}) - \frac{\sqrt{k}}{2} \cdot \lambda_k(\mathcal{L})$.

There must also exist some $i_2 \leq k$ such that $\boldsymbol{v}_{i_2} \notin \operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$. We can write $\boldsymbol{v}_{i_2} = B\boldsymbol{a}$ for some $\boldsymbol{a} \in \mathbb{Z}^n$. Let i_3 be the maximum non-zero coordinate in $\boldsymbol{a}$. Then $\lambda_{i_2}(\mathcal{L}) = \|B\boldsymbol{a}\| \geq \|a_{i_3}\tilde{\boldsymbol{b}}_{i_3}\| \geq \|\tilde{\boldsymbol{b}}_{i_3}\|$. Furthermore, $i_3 > k$ since $\boldsymbol{v}_{i_2} \notin \operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$.

Combining bounds, we get that

$$\eta(B) \geq \frac{\|\tilde{\boldsymbol{b}}_{i_1}\|}{\|\tilde{\boldsymbol{b}}_{i_3}\|} \geq \frac{\lambda_{k+1}(\mathcal{L}) - \frac{\sqrt{k}}{2} \cdot \lambda_k(\mathcal{L})}{\lambda_{i_2}(\mathcal{L})} \geq \frac{\lambda_{k+1}(\mathcal{L})}{\lambda_k(\mathcal{L})} - \frac{\sqrt{k}}{2},$$

which is a contradiction. □

This immediately implies the following corollary.

Corollary 5.4. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis which achieves $\eta(B) = \eta(\mathcal{L})$, and let $g(n) := \eta(n) + \sqrt{n}/2$. Then for all $k \in [n-1]$ with $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) > g(n)$, $\operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k) = V_k$.*

We also prove a block lower bound condition for η.

Lemma 5.5. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis of $\mathcal{L}$ of rank n. Then for all block projections $B_1, \ldots, B_m$ of B, $\eta(B) \geq \max_{i\in[m]} \eta(B_i)$.*

Proof. By definition of a block projection, there exist $0 = k_0 < \cdots < k_m = n$ such that for all $i \in [m]$, $B_i = (\pi_{k_{(i-1)}}(\boldsymbol{b}_{k_{(i-1)}+1}), \ldots, \pi_{k_{(i-1)}}(\boldsymbol{b}_{k_i}))$. Then for each $i \in [m]$,

$$\eta(B) = \max_{1\leq j\leq \ell\leq n} \|\tilde{\boldsymbol{b}}_j\|/\|\tilde{\boldsymbol{b}}_\ell\| \geq \max_{k_{(i-1)}+1\leq j\leq \ell\leq k_i} \|\tilde{\boldsymbol{b}}_j\|/\|\tilde{\boldsymbol{b}}_\ell\| = \eta(B_i)$$

as needed. □

We conclude by showing that η meets the conditions of Theorem 3.6.

Proof of Theorem 1.4. Let $g(n) = \eta(n) + \sqrt{n}$ and let $r_n(t) = t \cdot n^n$. The claim follows by combining Corollary 5.2, Corollary 5.4, and Lemma 5.5 with the fact that for Gram-Schmidt equivalent bases B and B', $\eta(B) = \eta(B')$ to meet the conditions of Theorem 3.6. □

5.2 Minimizing Q_{Babai}

Babai's nearest-plane algorithm [Bab86] uses a basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ as preprocessing to solve γ-CVP for

$$\gamma \leq \left(1 + \max_{i \in [n]} \frac{\sum_{j=1}^{i} \|\tilde{\boldsymbol{b}}_j\|^2}{\|\tilde{\boldsymbol{b}}_i\|^2}\right)^{1/2} . \tag{8}$$

Motivated by this, we define $Q_{\text{Babai}}(B)$ to be the expression on the right-hand side of Equation (8). Often Babai's algorithm is presented with the assumption that an LLL-reduced basis is used as its preprocessing. We refer the reader to the lecture notes of Stephens-Davidowitz [Ste] which prove the more general bound in Equation (8).

We note the relationship between $Q_{\text{Babai}}(B)$ and the Gram-Schmidt decay $\eta(B)$:

$$\eta(B) \leq Q_{\text{Babai}}(B) \leq \sqrt{n+1} \cdot \eta(B) . \tag{9}$$

As Equation (9) shows, $\eta(B)$ approximates $Q_{\text{Babai}}(B)$ up to a factor of $O(\sqrt{n})$. This allows us to leverage results for η from Section 5.1 to get an algorithm for computing bases which are optimal with respect to Q_{Babai}. We also need to prove a separate block lower bound result for Q_{Babai}.

Lemma 5.6. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis of $\mathcal{L}$ of rank n. Then for all block projections $B_1, \ldots, B_m$ of B, $Q_{Babai}(B) \geq \max_{i \in [m]} Q_{Babai}(B_i)$.*

Proof. By definition of a block projection, there exist $0 = k_0 < \cdots < k_m = n$ such that for all $i \in [m]$, $B_i = (\pi_{k_{(i-1)}}(\boldsymbol{b}_{k_{(i-1)}+1}), \ldots, \pi_{k_{(i-1)}}(\boldsymbol{b}_{k_i}))$. Then for each $i \in [m]$,

$$Q_{\text{Babai}}(B) = \left(1 + \max_{\ell \in [n]} \frac{\sum_{j=1}^{\ell} \|\tilde{\boldsymbol{b}}_j\|^2}{\|\tilde{\boldsymbol{b}}_\ell\|^2}\right)^{1/2} \geq \left(1 + \max_{k_{i-1}+1 \leq \ell \leq k_i} \frac{\sum_{j=k_{i-1}+1}^{\ell} \|\tilde{\boldsymbol{b}}_j\|^2}{\|\tilde{\boldsymbol{b}}_\ell\|^2}\right)^{1/2} = Q_{\text{Babai}}(B_i)$$

as needed. □

Theorem 5.7. *There exists an algorithm which, on input a basis B of a lattice $\mathcal{L}$ of rank n, outputs a basis B' of $\mathcal{L}$ which satisfies $Q_{Babai}(B') = Q_{Babai}(\mathcal{L})$, and which runs in $(\sqrt{n} \cdot \eta(n))^{O(n^3)} \leq n^{O(n^3 \log n)}$ time and polynomial space.*

Proof. We show that Q_{Babai} meets the conditions of Theorem 3.6 with $g(n) := (\sqrt{n}+1) \cdot \eta(n) + \sqrt{n}/2$ and $r_n(t) := t \cdot n^n$.

By Equation (9), $\eta(B) \leq Q_{\text{Babai}}(B)$ for all bases B. Therefore, by Corollary 5.2 it holds that if $Q_{\text{Babai}}(B) \leq t$ then $\max_{i \in [n]} \|\tilde{\boldsymbol{b}}_i\| \leq r_n(t) \cdot \lambda_n(\mathcal{L})$, so Q_{Babai} meets Theorem 3.6 Condition 1'(a). Condition 1'(b) (and therefore also Theorem 3.4 Condition 4) holds since if B and B' are Gram-Schmidt equivalent bases then $Q_{\text{Babai}}(B) = Q_{\text{Babai}}(B')$.

Moreover, as a consequence of Equation (9), $Q_{\text{Babai}}(n) \leq \sqrt{n+1} \cdot \eta(n)$. So, a basis $B' = (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n)$ satisfying $Q_{\text{Babai}}(B') = Q_{\text{Babai}}(\mathcal{L})$ must satisfy $\eta(B') \leq \sqrt{n+1} \cdot \eta(n)$, and so by Lemma 5.3 if $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) \geq g(n)$ then B' must satisfy $\text{span}(\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k) = V_k$. Therefore Q_{Babai} satisfies Theorem 3.4 Condition 2.

Finally, Theorem 3.4 Condition 3 holds by Lemma 5.6. □

6 Seysen condition number minimization

In this section we show how to use our technique to get an approximation scheme for computing bases which are $(1+\varepsilon)$-approximately optimal with respect to the Seysen condition number S.

In previous work, Seysen [Sey93] and Bennett, Dadush, and Stephens-Davidowitz [BDS16] showed that the basis vectors in an optimal basis B with respect to S must follow the successive minima of $\mathcal{L}$, and that B^* must follow the successive minima of $\mathcal{L}^*$.

Lemma 6.1 ([Sey93, Theorem 8], [BDS16, Lemma 21])**.** *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a sorted basis of $\mathcal{L}$. Then for all $k \in [n]$,*

1. $S(B) \geq \|\boldsymbol{b}_k\| / \lambda_k(\mathcal{L})$.
2. $S(B) \geq \|\boldsymbol{b}_k^*\| / \lambda_{n-k}(\mathcal{L}^*)$.

Lemma 6.1 will allow us to leverage Proposition 3.5. We next show a block lower bound for S.

Lemma 6.2. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis with block projection $B_1, \ldots, B_m$. Then $S(B) \geq \max_{i \in [m]} S(B_i)$.*

Proof. Let $B^* = (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*)$ denote the dual basis of B. If $B_1, \ldots, B_m$ is a block projection of B then there exists a basis B_2' such that B_1, B_2' is a block projection of B, and $B_2, \ldots, B_m$ is a block projection of B_2'.

Let $k \in [n-1]$, and let $B_1 = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k), B_2' = (\pi_k^{(B)}(\boldsymbol{b}_{k+1}), \ldots, \pi_k^{(B)}(\boldsymbol{b}_n))$ be a block projection of B. We have that

$$S(B) \geq \max_{1 \leq i \leq k} \|\boldsymbol{b}_i\| \|\boldsymbol{b}_i^*\| \geq \max_{1 \leq i \leq k} \|\boldsymbol{b}_i\| \|\pi_{\operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)}(\boldsymbol{b}_i^*)\| = S(B_1) \ ,$$

and similarly

$$S(B) \geq \max_{k+1 \leq i \leq n} \|\boldsymbol{b}_i\| \|\boldsymbol{b}_i^*\| \geq \max_{k+1 \leq i \leq n} \|\pi_k^{(B)}(\boldsymbol{b}_i)\| \|\boldsymbol{b}_i^*\| = S(B_2) \ ,$$

The claim follows by induction on B_2'. □

6.1 Seysen lifting

The lemmas in the previous section meet the first three conditions of Theorem 3.4. However, it is unclear how to satisfy the lifting condition, condition 4. Indeed, the crux of applying our technique to computing well-conditioned bases with respect to S is determining how to lift a block projection $B_1, \ldots, B_m$.

In this section we show how to lift block projections which meet certain conditions to a *nearly* optimal basis B' with $S(B') \leq (1+\varepsilon) \cdot S(\mathcal{L}(B'))$ by using a basis reduction technique introduced by Seysen [Sey93] which *simultaneously* reduces a primal basis B and its dual basis B^*. Size-reduction greedily reduces a primal basis B, but may lead to a dual basis B^* with very long vectors (see Proposition 7.6). Seysen's new simultaneous reduction technique was his key insight.

The proposition below is a generalization of [Sey93, Proposition 5], and its proof closely follows Seysen's proof. Let $\lfloor X \rceil$ denote the matrix obtained by rounding each entry in a real-valued matrix X to the nearest integer. Let $\|X\|_\infty$ denote the largest magnitude of an entry in such a matrix X.

Proposition 6.3 (Seysen reduction)**.** *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a basis and let $1 \le k \le n$. Let $C := (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_k) = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ and let $D := (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{n-k}) = (\pi_k^{(B)}(\boldsymbol{b}_{k+1}), \ldots, \pi_k^{(B)}(\boldsymbol{b}_n))$. Then there exists a unimodular matrix $T = T(B,k)$ such that $S(BT) \le \max\{S(C), S(D)\} + \frac{n}{2}\|C\|\|D^*\|$.*

Proof. Let $B = QB'$ be the QR-decomposition of B. Then B' has the form

$$B' = \begin{pmatrix} C' & Z' \\ 0 & D' \end{pmatrix}$$

with $C' \in \mathrm{GL}_k(\mathbb{R})$ satisfying $\|C'\| = \|C\|$, and $D' \in \mathrm{GL}_{n-k}(\mathbb{R})$ satisfying $\|(D')^*\| = \|D^*\|$. Let

$$T = T(B,k) := \begin{pmatrix} I_k & -\lfloor (C')^{-1} Z' \rceil \\ 0 & I_{n-k} \end{pmatrix} \tag{10}$$

It then holds that

$$B'T = \begin{pmatrix} C' & C'W \\ 0 & D' \end{pmatrix}, \qquad (B'T)^* = \begin{pmatrix} (C')^* & 0 \\ -(D')^* W^T & (D')^* \end{pmatrix},$$

where $W := (C')^{-1}Z' - \lfloor (C')^{-1}Z' \rceil$. Using the fact that $\|W\|_\infty \le \frac{1}{2}$ we have that

$$\|C'W\| \le k/2 \cdot \|C'\| = k/2 \cdot \|C\|$$

and

$$\|-(D')^* W^T\| \le (n-k)/2 \cdot \|(D')^*\| = (n-k)/2 \cdot \|D^*\| \ .$$

Therefore,

$$\begin{aligned} S(BT) &= S(B'T) \\ &\le \max\{\max_{i \in [k]} \|\boldsymbol{c}_i\| \cdot (\|\boldsymbol{c}_i^*\| + (n-k)/2 \cdot \|D^*\|), \max_{i \in [n-k]} (\|\boldsymbol{d}_i\| + (k/2) \cdot \|C\|) \cdot \|\boldsymbol{d}_i^*\|\} \\ &\le \max\{S(C), S(D)\} + (n/2) \cdot \|C\|\|D^*\| \end{aligned}$$

as claimed. □

It follows easily from the definition of S that for any basis B,

$$\|B\| \le S(B)/\lambda_1(\mathcal{L}(B)^*) \tag{11}$$

We will use this fact in the following lemma, which shows that if C is a basis of $\mathcal{L} \cap V_k$ and D is a basis of $\pi_k(\mathcal{L})$ for a lattice $\mathcal{L}$ where $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L})$ is large then the "additive error term" $\frac{n}{2}\|C\|\|D^*\|$ in the conclusion of Proposition 6.3 is small.

Lemma 6.4. *Let B be a basis of a lattice $\mathcal{L}$ of rank n. Assume that $C = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ is a basis of $\mathcal{L} \cap V_k$, and assume that $D = (\pi_k^{(B)}(\boldsymbol{b}_{k+1}), \ldots, \pi_k^{(B)}(\boldsymbol{b}_n))$ is a basis of $\pi_k(\mathcal{L})$ for some index $k \in [n-1]$ which satisfies $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) \ge t$ with $t \ge 10\sqrt{n}$. Then $\|C\|\|D^*\| \le 2 \cdot S(C)S(D) \cdot \lambda_k(\mathcal{L})/\lambda_{k+1}(\mathcal{L})$.*

Proof. We have that

$$\|C\|\|D^*\| \leq \frac{S(C)S(D)}{\lambda_1(\mathcal{L}(C)^*)\lambda_1(\mathcal{L}(D))} \leq S(C)S(D) \cdot \frac{\lambda_k(\mathcal{L} \cap V_k)}{\lambda_1(\pi_k(\mathcal{L}))} \tag{12}$$

where the first inequality holds by applying Equation (11) twice and noting that $S(D) = S(D^*)$, and the second inequality is by the lower bound in Theorem 2.3.

Furthermore, $\lambda_k(\mathcal{L} \cap V_k) = \lambda_k(\mathcal{L})$ by Lemma 2.17 item 1 and $\lambda_1(\pi_k(\mathcal{L})) \geq \lambda_{k+1}(\mathcal{L}) - \frac{\sqrt{k}}{2} \cdot \lambda_k(\mathcal{L})$ by Lemma 2.17 item 2. Combining these bounds and using the assumption that $t \geq 10\sqrt{n}$, we get that $\|C\|\|D^*\| \leq 2 \cdot S(C)S(D) \cdot \lambda_k(\mathcal{L})/\lambda_{k+1}(\mathcal{L})$, as claimed. □

Following this discussion, we define a lifting algorithm $\textsc{Lift}_S$ that combines blocks in a block projection one at a time to build a larger basis. It works as follows. On input a block projection $B_1, \ldots, B_m$, $\textsc{Lift}_S$ computes bases B'_i for $i = 1, \ldots, m$. It sets $B'_1 := B_1$. For $i = 2, \ldots, m$, $\textsc{Lift}_S$ first lifts the vectors in B_i to arbitrary vectors in $\mathcal{L}$ with the same projection onto $\text{span}(B'_{i-1})^\perp$ to form A_i. It then sets $B'_i := (B'_{i-1}, A_i) \cdot T((B'_{i-1}, A_i), \text{rank}(B'_{i-1}))$, where T is the matrix defined in Equation (10). Finally, $\textsc{Lift}_S$ outputs $B' = B'_m$.

6.2 An approximation scheme for S

We conclude with a proof of Theorem 1.5. Because S meets the first three conditions of Theorem 3.4, the proof is similar to the proof of Theorem 3.4. We focus on analyzing the quality of the "approximately optimal lifting" involved in $\textsc{Lift}_S$.

Proof of Theorem 1.5. Set $s = s(n, \varepsilon) := 2n^2 \cdot S(n)^2/\varepsilon$, $R = R(n) := S(n)$, and $\textsc{Lift} := \textsc{Lift}_S$ in Algorithm 1.

Because S is a permutation invariant basis quality function, we can use Lemma 6.1, item 1 to invoke Proposition 3.5, and so S meets Theorem 3.4 conditions 1 and 2. Furthermore, by Lemma 6.2, S meets Theorem 3.4 condition 3.

Therefore by the same reasoning as in the proof of Theorem 3.4, one of the sequences of bases $B_1, \ldots, B_m$ enumerated by Algorithm 1 is a block projection of an basis $\hat{B}$ that satisfies $S(\hat{B}) = S(\mathcal{L})$. Again using Lemma 6.2, we have that $\max_{i \in [n]} S(B_i) \leq S(\hat{B}) = S(\mathcal{L}) \leq S(n)$. We prove the theorem by showing that the basis $B' = B'_m$ output by $\textsc{Lift}_S(B, (B_1, \ldots, B_m))$ satisfies $S(B') \leq (1 + \varepsilon) \cdot S(\mathcal{L})$.

To do this we prove by induction that $S(B'_i) \leq S(\mathcal{L}) + i/n \cdot \varepsilon$ for all $i \in [m]$. We have that $S(B'_1) = S(B_1) \leq S(\hat{B}) = S(\mathcal{L})$. Suppose for $i > 1$ that $S(B'_{i-1}) \leq S(\mathcal{L}) + (i-1)/n \cdot \varepsilon$.

By Proposition 6.3, $S(B'_i) \leq \max\{S(B'_{i-1}), S(B_i)\} + \frac{n}{2}\|B'_{i-1}\|\|B_i^*\|$, and by Lemma 6.4, the definition of $s(n, \varepsilon)$, and the assumption that $\varepsilon \in (0, 1)$,

$$\begin{aligned}
\frac{n}{2}\|B'_{i-1}\|\|B_i^*\| &\leq n \cdot S(B'_{i-1}) \cdot S(B_i) \cdot \lambda_k(\mathcal{L})/\lambda_{k+1}(\mathcal{L}) \\
&\leq n \cdot (S(\mathcal{L}) + (i-1)/n \cdot \varepsilon) \cdot S(\mathcal{L}) \cdot \lambda_k(\mathcal{L})/\lambda_{k+1}(\mathcal{L}) \\
&\leq 2n \cdot S(\mathcal{L})^2 \cdot \lambda_k(\mathcal{L})/\lambda_{k+1}(\mathcal{L}) \\
&\leq 2n \cdot S(n)^2 \cdot s(n, \varepsilon)^{-1} \\
&\leq \varepsilon/n
\end{aligned}$$

So $S(B'_i) \leq S(\mathcal{L}) + (i-1)/n \cdot \varepsilon + \varepsilon/n \leq S(\mathcal{L}) + i/n \cdot \varepsilon$, as claimed. It then follows that $B' = B'_m$ satisfies $S(B') \leq S(\mathcal{L}) + m/n \cdot \varepsilon \leq (1 + \varepsilon) \cdot S(\mathcal{L})$, which proves the theorem. □

7 Motivating discussion and examples

This section discusses the relationship between lattice stability and gaps in the successive minima (Section 7.1), discusses the (non-)optimality of HKZ-bases for several of the basis quality functions described in this paper (Section 7.2), and gives some relationships between some of these functions (Section 7.3). Specifically, we motivate the techniques described in this paper by showing that gaps in the successive minima are not equivalent to lattice stability, and by showing that HKZ-reduced bases – which are a greedy way of formalizing what it means to be a "shortest possible lattice basis" – need not be minimizers of the basis quality functions that we consider.

7.1 Stability versus gaps in the successive minima

Recall that a lattice $\mathcal{L}$ is called *stable* if $\det(\mathcal{L}) = 1$ and $\det(\mathcal{L}') \geq 1$ for all sublattices $\mathcal{L}' \subseteq \mathcal{L}$. In this section, we compare the notion of stability with gaps in the successive minima.

Our first result gives a connection between stability and multiplicative gaps in the successive minima. Specifically, it shows that if $\mathcal{L}$ is stable then the "average gap" is relatively small: $(\prod_{k=1}^{n-1} \lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}))^{1/(n-1)} = (\lambda_n(\mathcal{L})/\lambda_1(\mathcal{L}))^{1/(n-1)} \leq O(\sqrt{n})$. However, it only gives a very loose upper bound on the "maximum gap" $\max_{k\in[n-1]} \lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L})$.

Lemma 7.1. *Let $\mathcal{L}$ be a stable lattice of rank n. Then $\lambda_n(\mathcal{L})/\lambda_1(\mathcal{L}) \leq n^{n/2}$.*

Proof. Since $\mathcal{L}$ is stable, $\det(\mathcal{L}) = 1$ and $\lambda_1(\mathcal{L}) \geq 1$. So, by Minkowski's Second Theorem (Theorem 2.1), $\prod_{i=1}^{n} \lambda_i(\mathcal{L}) \leq n^{n/2}$. Since $\lambda_1(\mathcal{L}) \geq 1$ it follows that $\lambda_n(\mathcal{L}) \leq n^{n/2}$, and hence that $\lambda_n(\mathcal{L})/\lambda_1(\mathcal{L}) \leq n^{n/2}$. □

Our second result gives a family of stable lattices with polynomial gaps in their successive minima, therefore separating the notions of stability and gaps in the successive minima. We obtain such a family of examples by taking the direct sum of the integers with a "random lattice" in the sense of [Sie45]. Such random lattices are almost surely both stable ([SW14]) and nearly tight for Minkowski's Theorems ([Sie45]), implying Fact 7.2 below.

Fact 7.2. *For every $n \geq 1$, there exists a lattice $\mathcal{L}$ of rank n such that $\mathcal{L}$ is stable and satisfies $\lambda_i(\mathcal{L}) = \Theta(\sqrt{n})$ for all $i \in [n]$.*

Using Fact 7.2, we construct examples of lattices which have a gap in their successive minima but are nonetheless stable. Our examples are related to examples of Regev et al. [RSW17] which disproved a conjecture about the covering radii of a certain family of lattices. Define the direct sum of lattices $\mathcal{L}_1, \mathcal{L}_2$ as $\mathcal{L}_1 \oplus \mathcal{L}_2 := \{(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x} \in \mathcal{L}_1, \boldsymbol{y} \in \mathcal{L}_2\}$. $\mathcal{L}_1 \oplus \mathcal{L}_2$ is also a lattice.

Lemma 7.3. *For every $n \geq 2$ there exists a lattice $\mathcal{L}$ of rank n which is stable, and which has $\lambda_{k+1}(\mathcal{L})/\lambda_k(\mathcal{L}) \geq \Omega(\sqrt{n})$ for some $k \in [n-1]$.*

Proof. Let $\mathcal{L}_1$ be a stable lattice of rank $n-1$ with $\lambda_i(\mathcal{L}) = \Theta(\sqrt{n})$ for all $i \in [n-1]$, and let $\mathcal{L} := \mathbb{Z} \oplus \mathcal{L}_1$. (Fact 7.2 ensures the existence of such a lattice $\mathcal{L}_1$.) Then $\mathcal{L}$ is also stable, and satisfies $\lambda_1(\mathcal{L}) = 1$ and $\lambda_2(\mathcal{L}) \geq \Omega(\sqrt{n})$. □

7.2 (Non-)optimality of HKZ-reduced bases

HKZ-reduced bases (or simply HKZ-bases) give a greedy way of formalizing what it means to be a shortest possible lattice basis. As we show in this section, there are HKZ-reduced bases B that do not minimize either δ or S, so this greedy formalization is not always optimal. As important motivation for this paper, it follows that minimizing these basis quality functions is not as straightforward as computing an HKZ basis.

7.2.1 HKZ-reduced bases

We define HKZ-reduced bases as follows.

Definition 7.4. A basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is *HKZ-reduced* if it satisfies the following conditions:

1. $\boldsymbol{b}_1$ is a shortest non-zero vector of $\mathcal{L}(B)$,
2. B is size-reduced,
3. If $n > 1$ then $(\pi_1^{(B)}(\boldsymbol{b}_2), \ldots, \pi_1^{(B)}(\boldsymbol{b}_n))$ is an HKZ-reduced basis of $\pi_1(\mathcal{L})$.

7.2.2 Exponential δ and S gaps for HKZ-reduced bases

In this section we give an example of a family of bases $\{A^{(n)}\}_{n=1}^{\infty}$ (defined in Equation (13) below) that are poorly conditioned in the sense of Seysen (this essentially appeared already in [BDS16, Section 5.2]), and which have poor orthogonality defect compared to other bases of $\mathcal{L}(A^{(n)})$. In fact, we show an exponential gap between $\delta(A^{(n)})$ (resp. $S(A^{(n)})$) and $\delta(\mathcal{L}(A^{(n)}))$ (resp. $S(\mathcal{L}(A^{(n)}))$).

Fix $n \geq 1$ and define the matrix $A = A^{(n)}$ with entries as follows for $i, j \in [n]$:

$$A_{i,j} = \begin{cases} 0 & \text{if } j < i, \\ 1 & \text{if } j = i, \\ -\frac{1}{2} & \text{if } j > i. \end{cases} \tag{13}$$

I.e., A is the $n \times n$ upper-triangular matrix with diagonal entries equal to 1 and off-diagonal upper triangular entries equal to $-\frac{1}{2}$. It is straightforward to check that A is an HKZ-reduced basis of $\mathcal{L}(A)$. The matrix A previously appeared in [LT08, BDS16] as an example of a poorly conditioned HKZ-reduced basis. Here we show that A is poorly conditioned in the sense of Seysen (this essentially appeared already in [BDS16, Section 5.2]), and also that it has poor orthogonality defect compared to other bases of A.

Proposition 7.5 (Non-optimality of HKZ-reduced bases for orthogonality defect δ)**.** *For every $n \geq 1$, there exists an HKZ-reduced basis A with $\delta(A) \geq 2^{-n} \cdot \sqrt{n!} \geq n^{\Omega(n)}$ and another basis A' of $\mathcal{L}(A)$ with $\delta(A') = (\sqrt{13}/2)^{n-1} \leq 2^{O(n)}$.*

Proof. Define A' with entries as follows for $i, j \in [n]$:

$$A'_{i,j} = \begin{cases} 1 & \text{if } j = i, \\ -\frac{3}{2} & \text{if } j = i+1, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

I.e., A' is the $n \times n$ bi-diagonal basis with entries equal to 1 on the main diagonal, and entries equal to $-\frac{3}{2}$ on the diagonal above.

It is straightforward to check that $\mathcal{L}(A') = \mathcal{L}(A)$, where A is the $n \times n$ basis defined in Equation (13). Moreover, $\delta(A) \geq 2^{-n} \cdot \sqrt{n!} \geq n^{\Omega(n)}$ and $\delta(A') = (\sqrt{13}/2)^{n-1} = 2^{O(n)}$, as needed. □

We next show that $S(A)$ is exponentially larger than $S(\mathcal{L}(A))$. This relies on the main theorem in Seysen's paper [Sey93], which says that $S(n) \leq n^{O(\log n)}$.

Proposition 7.6 (Non-optimality of HKZ-reduced bases for Seysen condition number S)**.** *For every $n \geq 1$, there exists an HKZ-reduced basis A with $S(A) \geq \Omega(1.5^n)$ and another basis A'' of $\mathcal{L}(A)$ with $S(A'') \leq n^{O(\log n)}$.*

Proof. $S(A) \geq \Omega(1.5^n)$ follows by noting that $\|\boldsymbol{a}_i\| \geq 1$ for all $i \in [n]$, and from a computation which shows that the largest entry in A^{-1} is of magnitude $\Omega(1.5^n)$. The existence of such an A'' follows from [Sey93]. □

In fact, because A is upper-triangular and unipotent (has all diagonal entries equal to 1), the basis A'' in Proposition 7.6 can even be computed efficiently using the algorithm described in [Sey93, Proposition 5].

We note that both A' and A'' are Gram-Schmidt equivalent to A. Determining whether there exists a basis A''' that is not Gram-Schmidt equivalent to any HKZ basis, and that satisfies $\delta(A''') < \delta(A)$ or $S(A''') < S(A)$ for all HKZ bases A of $\mathcal{L}(A''')$ is another interesting question.

7.2.3 Gram-Schmidt decay in HKZ-reduced Bases

In this section we briefly summarize previous work on Gram-Schmidt decay in HKZ-bases. Unlike for the orthogonality defect δ and the Seysen condition number S, to the best of our knowledge there is no family of HKZ bases with asymptotically non-optimal multiplicative Gram-Schmidt decay η. In fact, we are not even aware of a single HKZ basis B where $\eta(B) \neq \eta(\mathcal{L}(B))$.

Lagarias et al. [LLS90] showed that for every lattice $\mathcal{L}$ there exists an HKZ-reduced basis B of $\mathcal{L}$ such $\eta(B) \leq n^{O(\log n)}$. Moreover, Ajtai [Ajt08] showed that this analysis is essentially optimal – there exists a family of HKZ bases $\{B^{(n)}\}_{i=1}^{\infty}$ such that $\eta(B) \geq n^{\Omega(\log n)}$ for $B = B^{(n)}$. In fact, Ajtai showed an even stronger lower bound. He shows that $\alpha_n := \sup \|\boldsymbol{b}_1\|^2 / \|\tilde{\boldsymbol{b}}_n\|^2$ is at least $n^{c \log n}$ for some absolute constant $c > 0$ (where the supremum in the definition of α_n is taken over all HKZ-reduced bases of rank n). Note that the Gram-Schmidt decay parameter $\eta(B)$ is the maximum ratio $\|\tilde{\boldsymbol{b}}_i\| / \|\tilde{\boldsymbol{b}}_j\|$ taken over *all* pairs of Gram-Schmidt vectors $\tilde{\boldsymbol{b}}_i, \tilde{\boldsymbol{b}}_j$ with $1 \leq i \leq j \leq n$, whereas α_n only considers the case where $i = 1, j = n$.

These results are summarized in the following theorem.

Theorem 7.7 ([LLS90, Ajt08])**.** *There exist absolute constants $c_1, c_2 > 0$ such that the following statement holds. For every HKZ-reduced basis B of rank $n \geq 1$, $\eta(B) \leq n^{c_1 \cdot \log n}$, and for every $n \geq 1$ there exists an HKZ-reduced basis B' of rank n such that $\eta(B') \geq n^{c_2 \cdot \log n}$.*

As previously mentioned it is an important open question whether whether HKZ bases have optimal Gram-Schmidt decay or whether each lattice has a basis with Gram-Schmidt decay poly(n).

7.3 Relationships between basis quality measures

We conclude by presenting some relationships between the basis quality measures that we consider. As noted in Equation (9), Q_{Babai} and η are closely related. Here we compare S with η and δ. We also refer the reader to [MG02, Theorem 7.10] which presents a number of relationships between basis reduction problems.

Proposition 7.8. *Let $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a sorted basis. Then $S(B) \geq \eta(B)$.*

Proof. We use the fact that for any basis $B \in \mathbb{R}^{n \times n}$,

$$\|\boldsymbol{b}_i^*\| \geq \|\tilde{\boldsymbol{b}}_i\|^{-1} \ . \tag{15}$$

Indeed, $\boldsymbol{b}_i = \tilde{\boldsymbol{b}}_i + \boldsymbol{x}$ for some $\boldsymbol{x} \in \operatorname{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1})$. Equation (15) then follows from the fact that

$$\|\tilde{\boldsymbol{b}}_i\| \|\boldsymbol{b}_i^*\| \geq \langle \tilde{\boldsymbol{b}}_i, \boldsymbol{b}_i^* \rangle = \langle \tilde{\boldsymbol{b}}_i + \boldsymbol{x}, \boldsymbol{b}_i^* \rangle = \langle \boldsymbol{b}_i, \boldsymbol{b}_i^* \rangle = 1 \ .$$

where the inequality holds by the Cauchy-Schwarz inequality, and the equalities hold because $\langle \boldsymbol{b}_i, \boldsymbol{b}_j^* \rangle$ is equal to 1 if $i = j$ and is equal to 0 otherwise.

Therefore,

$$S(B) = \max_j \|\boldsymbol{b}_j\| \|\boldsymbol{b}_j^*\| \geq \max_{i \leq j} \|\boldsymbol{b}_i\| \|\boldsymbol{b}_j^*\| \geq \max_{i \leq j} \|\tilde{\boldsymbol{b}}_i\| \|\boldsymbol{b}_j^*\| \geq \max_{i \leq j} \frac{\|\tilde{\boldsymbol{b}}_i\|}{\|\tilde{\boldsymbol{b}}_j\|} = \eta(B),$$

where the first inequality holds because B is sorted, and the third inequality holds by Equation (15). □

Because S is permutation invariant, there is always a sorted basis B of $\mathcal{L}$ which achieves $S(B) = S(\mathcal{L})$, and we get the following corollary.

Corollary 7.9. *For every lattice $\mathcal{L}$, $S(\mathcal{L}) \geq \eta(\mathcal{L})$ and for every $n \geq 1$, $S(n) \geq \eta(n)$.*

We finish with a simple relationship between $S(B)$ and the normalized orthogonality defect $\delta(B)^{1/n}$.

Lemma 7.10. *For every basis $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$, $S(B) \geq \delta(B)^{1/n}$.*

Proof. We have that $S(B) \geq \left(\prod_{i=1}^n \|\boldsymbol{b}_i\| \|\boldsymbol{b}_i^*\| \right) \geq \left(\prod_{i=1}^n \|\boldsymbol{b}_i\| / \|\tilde{\boldsymbol{b}}_i\| \right) = \delta(B)^{1/n}$, where the second inequality follows by Equation (15). □

References

[AD97] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 284–293, 1997.

[Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, pages 99–108, 1996.

[Ajt08] Miklós Ajtai. Optimal lower bounds for the Korkine-Zolotareff parameters of a lattice and for Schnorr's algorithm for the shortest vector problem. *Theory of Computing*, 4(2):21–51, 2008.

[Bab86] László Babai. On Lovász' lattice reduction and the nearest lattice point problem. *Combinatorica*, 6(1):1–13, 1986.

[Ban93] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(1):625–635, 1993.

[BDS16] Huck Bennett, Daniel Dadush, and Noah Stephens-Davidowitz. On the lattice distortion problem. In *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, pages 9:1–9:17, 2016.

[CDL17] Antonio Campello, Daniel Dadush, and Cong Ling. AWGN-goodness is enough: Capacity-achieving lattice codes based on dithered probabilistic shaping. *CoRR*, abs/1707.06688, 2017. To appear in IEEE Transactions on Information Theory 2018.

[CS98] John Conway and Neil J. A. Sloane. *Sphere Packings, Lattices and Groups*. Springer, 1998.

[Dad12] Daniel Dadush. *Integer Programming, Lattice Algorithms, and Deterministic Volume Estimation*. PhD thesis, Georgia Institute of Technology, 2012.

[DRS14] Daniel Dadush, Oded Regev, and Noah Stephens-Davidowitz. On the closest vector problem with a distance guarantee. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, pages 98–109, 2014.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 169–178, 2009.

[GGH97] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. Public-key cryptosystems from lattice reduction problems. In *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, pages 112–131, 1997.

[GN08] Nicolas Gama and Phong Q. Nguyen. Finding short lattice vectors within Mordell's inequality. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 207–216, 2008.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 197–206, 2008.

[Hel85] Bettina Helfrich. Algorithms to construct Minkowski reduced and Hermite reduced lattice bases. *Theor. Comput. Sci.*, 41:125–139, 1985.

[HL90] Johan Håstad and Jeffrey C. Lagarias. Simultaneously good bases of a lattice and its reciprocal lattice. *Mathematische Annalen*, 287(1):163–174, Mar 1990.

[HR14] Ishay Haviv and Oded Regev. On the lattice isomorphism problem. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 391–404, 2014.

[Kan87] Ravi Kannan. Minkowski's convex body theorem and integer programming. *Math. Oper. Res.*, 12(3):415–440, 1987.

[KZ73] A. Korkine and G. Zolotareff. Sur les formes quadratiques. *Mathematische Annalen*, 6(3):366–389, 1873.

[LB14] Cong Ling and Jean-Claude Belfiore. Achieving AWGN channel capacity with lattice gaussian coding. *IEEE Trans. Information Theory*, 60(10):5918–5929, 2014.

[Len83] Hendrik W. Lenstra Jr. Integer programming with a fixed number of variables. *Math. Oper. Res.*, 8(4):538–548, 1983.

[LLL82] Arjen Lenstra, Hendrik Lenstra Jr., and László Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.

[LLS90] J. C. Lagarias, Hendrik W. Lenstra Jr., and Claus-Peter Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. *Combinatorica*, 10(4):333–348, 1990.

[LT08] Franklin T. Luk and Daniel M. Tracy. An improved lll algorithm. *Linear Algebra and its Applications*, 428(2):441 – 452, 2008.

[Maz10] Grard Maze. Some inequalities related to the Seysen measure of a lattice. *Linear Algebra and its Applications*, 433(8):1659 – 1665, 2010.

[MG02] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems*. Springer Science+Business Media, 2002.

[Mic08] Daniele Micciancio. Efficient reductions among lattice problems. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2008, San Francisco, California, USA, January 20-22, 2008*, pages 84–93, 2008.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.

[RS17] Oded Regev and Noah Stephens-Davidowitz. A reverse Minkowski theorem. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 941–953, 2017.

[RSW17] Oded Regev, Uri Shapira, and Barak Weiss. Counterexamples to a conjecture of woods. *Duke Mathematical Journal*, 166(13):2443–2446, 2017. arXiv:1604.07644.

[Sch87] Claus-Peter Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theor. Comput. Sci.*, 53:201–224, 1987.

[Sey93] Martin Seysen. Simultaneous reduction of a lattice basis and its reciprocal basis. *Combinatorica*, 13(3):363–376, 1993.

[Sey99] Martin Seysen. A measure for the non-orthogonality of a lattice basis. *Combinatorics, Probability and Computing*, 8(3):281291, 1999.

[Sie45] Carl Ludwig Siegel. A mean value theorem in geometry of numbers. *Annals of Mathematics*, 46(2):pp. 340–347, 1945.

[SMH07] Dominik Seethaler, Gerald Matz, and Franz Hlawatsch. Low-complexity MIMO data detection using seysen's lattice reduction algorithm. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2007, Honolulu, Hawaii, USA, April 15-20, 2007*, pages 53–56, 2007.

[Ste] Noah Stephens-Davidowitz. Lattices mini course lecture notes. Available at `http://www.noahsd.com/lattices_mini_course.html`.

[SW14] Uri Shapira and Barak Weiss. A volume estimate for the set of stable lattices. *Comptes Rendus Mathematique*, 352(11):875 – 879, 2014.

[Vaz01] Vijay V. Vazirani. *Approximation algorithms*. Springer, 2001.

[vdWG68] Bartel L. van der Waerden and Herbert Gross. *Studien zur Theorie der quadratischen Formen*. Birkhuser, 1968.

[YH15] Shaoshi Yang and Lajos Hanzo. Fifty years of MIMO detection: The road to large-scale mimos. *IEEE Communications Surveys and Tutorials*, 17(4):1941–1988, 2015.

[ZAM08] Wei Zhang, Felix Arnold, and Xiaoli Ma. An analysis of Seysen's lattice reduction algorithm. *Signal Processing*, 88(10):2573–2577, 2008.

[ZMS10] Wei Zhang, Xiaoli Ma, and Ananthram Swami. Designing low-complexity detectors based on seysen's algorithm. *IEEE Trans. Wireless Communications*, 9(10):3301–3311, 2010.