

Minimal Subsampled Rank-1 Lattices for Multivariate Approximation with Optimal Convergence Rate

Felix Bartel* Alexander D. Gilbert* Frances Y. Kuo*
Ian H. Sloan*

June 27, 2025

In this paper we show error bounds for randomly subsampled rank-1 lattices. We pay particular attention to the ratio of the size of the subset to the size of the initial lattice, which is decisive for the computational complexity. In the special case of Korobov spaces, we achieve the *optimal polynomial sampling complexity* whilst having the *smallest initial lattice possible*. We further characterize the frequency index set for which a given lattice is reconstructing by using the reciprocal of the worst-case error achieved using the lattice in question. This connects existing approaches used in proving error bounds for lattices. We make detailed comments on the implementation and test different algorithms using the subsampled lattice in numerical experiments.

MSC2020: 41A25, 94A20

1 Introduction

This paper mainly deals with the question of approximating functions from partial rank-1 lattices. Rank-1 lattices have their origin as a Quasi-Monte Carlo (QMC) method and have been thoroughly studied in various settings ever since, see e.g., [39, 21, 22, 36, 6, 7]. The lattice point sets are given by $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k \in \{0, \dots, n-1\}}$, where $\mathbf{z} \in \mathbb{Z}^d$ is known as the generating vector and it controls the quality of the point set, and $\bmod 1$ denotes a component-wise fractional part of a vector. The study of partial or subsampled rank-1 lattices has only begun recently, cf. [1]. Such point sets have the form

$$\mathbf{X}_J = \left\{ \frac{k}{n}\mathbf{z} \bmod 1 \right\}_{k \in J} \quad \text{with} \quad J \subseteq \{0, \dots, n-1\}.$$

*School of Mathematics and Statistics, University of New South Wales, Sydney NSW 2052, Australia.
E-Mail: {f.bartel, alexander.gilbert, f.kuo, i.sloan}@unsw.edu.au

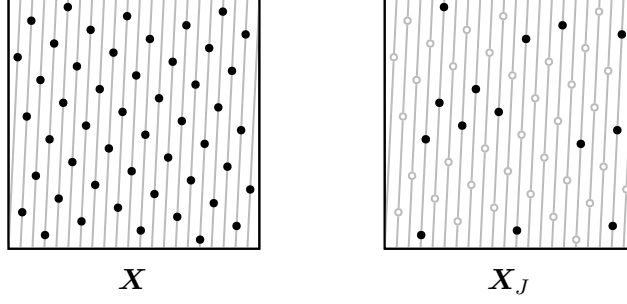


Figure 1: Left: rank-1 lattice with generating vector $\mathbf{z} = (1, 21)$ and lattice size $n = 55$.
Right: subsampled rank-1 lattice with $|J| = \lceil \sqrt{n} \log \sqrt{n} \rceil = 15$.

An example is depicted in Figure 1. The structure of lattice point sets allows for the application of fast Fourier methods when approximating functions whilst the subsampling improves the approximation properties, cf. [2]. We investigate the worst-case setting, where the point set and algorithm are fixed and are used to approximate every function from a given function class H . The *worst-case error* for a general algorithm A measured in L_2 -norm is defined by

$$e^{\text{wor-app}}(A) := \sup_{\|f\|_H \leq 1} \|f - Af\|_{L_2},$$

where $\|\cdot\|_H$ denotes the norm in H .

For full lattices $\mathbf{X} = \mathbf{X}_{\{0, \dots, n-1\}}$ there are two main approaches to control the worst-case error:

- (i) Based on the function space H and a lattice size n , one computes a generating vector $\mathbf{z}$ directly minimizing (a computable upper bound on) the worst-case error. It was shown in e.g., [33, 6] that, for certain Korobov spaces with smoothness parameter $\alpha > 1/2$, a component-by-component (CBC) construction achieves a decay rate

$$e^{\text{wor-app}}(A_{\mathcal{A}}^{\mathbf{X}}) \lesssim n^{-\alpha/2+\varepsilon}, \quad (1.1)$$

where $\varepsilon > 0$, $A_{\mathcal{A}}^{\mathbf{X}}$ is the “classical lattice algorithm” (see (2.1) below), and $A \lesssim B$ means there exists a constant $C > 0$ independent of n such that $A \leq CB$. When α is an integer it is the number of available square-integrable derivatives in each coordinate.

- (ii) Another approach was taken in e.g., [21, 22] where one first constructs a suitable frequency index set in the Fourier domain and then uses a CBC construction to end up with a “reconstructing lattice” (see precise definition in (3.1) below). This enables one to exactly reconstruct every trigonometric polynomial supported on the given frequencies. The “aliasing”, i.e., the error arising from frequencies outside of the index set, is then controlled via an error bound. The resulting decay rate is the same as in (1.1).

In this paper we connect both approaches by finding a frequency index set for which a lattice is reconstructing in Theorem 3.2 below. These frequencies solely depend on the worst-case error achievable using any algorithm and the lattice in question. In that way, lattices constructed according the first approach can be investigated with respect to their reconstructing property, and tools from the second approach become available.

Regardless of the approach, there is a lower bound on the error rate using a full lattice of $n^{-\alpha/2}$, which is half the polynomial optimal rate of decay $n^{-\alpha}$ when there is no restriction for the point set used, cf. [3]. This seems to be an intrinsic problem with redundant information in the full lattice and motivates only using a subsampled lattice. This was already investigated in the paper [1].

For unweighted Korobov spaces known as periodic Sobolev spaces with dominating mixed smoothness $\alpha > 1/2$, a combination of random and deterministic subsampling was used in [1, Corollary 1.2], yielding the bound

$$e^{\text{wor-app}}(S_B^{\mathbf{X}_J}) \lesssim |J|^{-\alpha} (\log |J|)^{(d-1)\alpha + \frac{1}{2}} \quad \text{with} \quad n \sim |J|^{\frac{2}{1-1/(2\alpha)}}, \quad (1.2)$$

where $S_B^{\mathbf{X}_J}$ is the “least squares approximation” to be defined precisely in (4.1) below and $A \sim B$ means $A \lesssim B$ and $B \lesssim A$ holds simultaneously.

The method in [1] achieves the optimal polynomial order $|J|^{-\alpha+\varepsilon}$ up to $\varepsilon > 0$ in terms of sampling complexity. However, the computational complexity is determined by n . Since an algorithm using a subsampled lattice is inherently still using the lattice of size n , the lower bound from [3] still applies, and achieving the rate $|J|^{-\alpha+\varepsilon}$ implies $n^{-\alpha/2} \lesssim |J|^{-\alpha+\varepsilon}$, yielding the lower bound

$$n \gtrsim |J|^{2-\frac{2\varepsilon}{\alpha}}.$$

With respect to the size of $|J|$, (1.2) is off by a polynomial order depending on the smoothness α .

In this paper, in Theorem 5.3 below, we show that we can go arbitrarily close to the squared number of points independently of the smoothness, i.e., we have

$$e^{\text{wor-app}}(S_B^{\mathbf{X}_J}) \lesssim \left(\frac{|J|}{\log |J|}\right)^{-\alpha+\varepsilon} \quad \text{with} \quad \left(\frac{|J|}{\log |J|}\right)^{2\sqrt{1-\varepsilon/\alpha}} \lesssim n \lesssim \left(\frac{|J|}{\log |J|}\right)^{\frac{2}{\sqrt{1-\varepsilon/\alpha}}},$$

for $0 < \varepsilon < \alpha$. This result is optimal in the sense that it achieves the best polynomial sampling complexity whilst having the smallest initial lattice size.

This paper is organized as follows. We start by introducing the classical lattice algorithm and the kernel method in Section 2. Then we investigate the projection property of approximation algorithms in Section 3, where we also prove a connection between the worst-case error and the reconstructing property of lattices in Theorem 3.2, which appears to be a new result. In Section 4 we show general error bounds for the randomly subsampled lattice, which we apply to the Korobov space setting in Section 5. After some comments on different implementations in Section 6, we present some numerical experiments in Section 7 and give concluding remarks in Section 8.

Notation. In this paper $\mathbb{T} = \mathbb{R}/\mathbb{N}$ is the 1-periodic torus; $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product; $\mathbf{x} \bmod 1$ is understood as the component-wise fractional part of a vector $\mathbf{x} \in \mathbb{R}^d$; we write $a \equiv_n b$ if there exists $k \in \mathbb{N}$ such that $a = b + kn$ and $a \not\equiv_n b$ if not; $\|\cdot\|_2$ denotes the spectral norm of a matrix; vectors are understood as column vectors.

2 Classical lattice algorithm versus kernel method

A *rank-1 lattice* point set $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1}$ is defined by a *generating vector* $\mathbf{z} \in \mathbb{Z}^d$ and the *lattice size* $n \in \mathbb{N}$. Rank-1 lattices were first studied as a Quasi-Monte Carlo (QMC) method to compute integrals, see e.g., [39]

$$\int_{\mathbb{T}^d} f(\mathbf{x}) \, d\mathbf{x} \approx \frac{1}{n} \sum_{k=0}^{n-1} f\left(\frac{k}{n}\mathbf{z} \bmod 1\right),$$

which corresponds to approximating the Fourier coefficient $\widehat{f}_{\mathbf{0}}$ of a function with the Fourier series

$$f = \sum_{\mathbf{h} \in \mathbb{Z}^d} \widehat{f}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle), \quad \text{where} \quad \widehat{f}_{\mathbf{h}} = \int_{\mathbb{T}^d} f(\mathbf{x}) \exp(-2\pi i \langle \mathbf{h}, \mathbf{x} \rangle) \, d\mathbf{x}.$$

Doing this for multiple frequencies $\mathbf{h}$ from a prescribed frequency index set $\mathcal{A} \subseteq \mathbb{Z}^d$, one may construct the trigonometric polynomial approximation to f which we refer to as the *classical lattice algorithm*

$$A_{\mathcal{A}}^{\mathbf{X}} f := \sum_{\mathbf{h} \in \mathcal{A}} \underbrace{\left(\frac{1}{n} \sum_{k=0}^{n-1} f\left(\frac{k}{n}\mathbf{z} \bmod 1\right) \exp(-2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle) \right)}_{=: \widehat{g}_{\mathbf{h}}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle), \quad (2.1)$$

see e.g., [33, 22, 21, 36, 6, 7]. To get a feeling for this method, we have a look at the Fourier coefficients of the approximation $A_{\mathcal{A}}^{\mathbf{X}} f =: g = \sum_{\mathbf{h} \in \mathcal{A}} \widehat{g}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle)$ in terms of the Fourier coefficients of f . By the *character property* $\frac{1}{n} \sum_{k=0}^{n-1} \exp(2\pi i m \frac{k}{n}) = \delta_{m \equiv_n 0}$, we have for $\mathbf{h} \in \mathcal{A}$

$$\begin{aligned} \widehat{g}_{\mathbf{h}} &= \frac{1}{n} \sum_{k=0}^{n-1} \left(\sum_{\ell \in \mathbb{Z}^d} \widehat{f}_{\ell} \exp(2\pi i \frac{k}{n} \langle \ell, \mathbf{z} \rangle) \right) \exp(-2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle) \\ &= \sum_{\ell \in \mathbb{Z}^d} \widehat{f}_{\ell} \frac{1}{n} \sum_{k=0}^{n-1} \exp(2\pi i \frac{k}{n} \langle \ell - \mathbf{h}, \mathbf{z} \rangle) \\ &= \sum_{\substack{\ell \in \mathbb{Z}^d \\ \langle \ell, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}, \mathbf{z} \rangle}} \widehat{f}_{\ell}. \end{aligned} \quad (2.2)$$

Thus, the classical lattice approximation has an aliasing effect where multiple frequencies of the target function are mapped to a single frequency of the approximation. In order

to control this error we have to demand some prior smoothness assumptions on the function f . This is usually done by enforcing a decay in the Fourier coefficients as follows

$$f \in H = \left\{ f: \mathbb{T}^d \rightarrow \mathbb{C} : \|f\|_H^2 = \sum_{\mathbf{h} \in \mathbb{Z}^d} r(\mathbf{h}) |\widehat{f}_{\mathbf{h}}|^2 < \infty \right\}, \quad (2.3)$$

where $r(\mathbf{h}) \rightarrow \infty$ for $\|\mathbf{h}\|_2 \rightarrow \infty$. Specific choices of the function $r(\cdot)$ lead to e.g., the Korobov spaces in Section 5. Obviously there are good and bad choices for the frequency index set $\mathcal{A} \subseteq \mathbb{Z}^d$ and the generating vector $\mathbf{z} \in \mathbb{Z}^d$. The following theorem gives an error bound for the classical lattice algorithm and follows [6, Section 2.4].

Theorem 2.1. *Let H be the function space defined in (2.3). Further, let $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ a lattice, and $\mathcal{A} = \{\mathbf{h} \in \mathbb{Z}^d : r(\mathbf{h}) \leq M\}$ for some radius $M > 0$. Then for the classical lattice algorithm defined in (2.1), we have*

$$e^{\text{wor-app}}(A_{\mathcal{A}}^{\mathbf{X}})^2 < \frac{1}{M} + M\mathcal{S}_n(\mathbf{z}), \quad (2.4)$$

with

$$\mathcal{S}_n(\mathbf{z}) = \sum_{\mathbf{h} \in \mathbb{Z}^d} \frac{1}{r(\mathbf{h})} \sum_{\substack{\boldsymbol{\ell} \in \mathbb{Z}^d \setminus \{\mathbf{0}\} \\ \langle \boldsymbol{\ell}, \mathbf{z} \rangle \equiv_n 0}} \frac{1}{r(\mathbf{h} + \boldsymbol{\ell})}. \quad (2.5)$$

In particular, with the optimal choice $M = 1/\sqrt{\mathcal{S}_n(\mathbf{z})}$, which minimizes (2.4), we have

$$e^{\text{wor-app}}(A_{\mathcal{A}}^{\mathbf{X}})^2 < \frac{2}{M} = 2\sqrt{\mathcal{S}_n(\mathbf{z})}. \quad (2.6)$$

We have stated (2.4) and (2.6) as strict inequalities. This can be seen from the derivation of the quantity $\mathcal{S}_n(\mathbf{z})$, see e.g., [6, Formula (2.7)], where infinitely many positive terms with index $\mathbf{h} \notin \mathcal{A}$ were added.

For weighted Korobov spaces (see Section 5 below) and a given lattice size $n \in \mathbb{N}$, component-by-component (CBC) constructions for generating vectors $\mathbf{z}$ are known, controlling $\mathcal{S}_n(\mathbf{z})$ as in Theorem 5.1 below, which is [30, Theorem 3.3]. Fast CBC implementations are given in [6, 14].

Next we comment on the kernel method, which is often used in practice and benefits from the lattice structure, cf. [20, Section 5]. For kernel methods we need to assume that H is a reproducing kernel Hilbert space, i.e., function evaluations are continuous, which is a common and natural assumption. By the Moore–Aronszajn theorem H has a corresponding reproducing kernel $K: \mathbb{T}^d \times \mathbb{T}^d \rightarrow \mathbb{C}$, which is symmetric and positive definite. The *kernel approximation* of f based on the lattice $\mathbf{X}$ is defined by

$$A_{\text{ker}}^{\mathbf{X}} f := \sum_{k=0}^{n-1} a_k K(\cdot, \frac{k}{n}\mathbf{z} \bmod 1) \quad \text{with} \quad \mathbf{a} = [a_k]_{k \in \{0, \dots, n-1\}} = \mathbf{K}^{-1} \mathbf{f}, \quad (2.7)$$

where $\mathbf{K} = [K(\frac{k}{n}\mathbf{z} \bmod 1, \frac{k'}{n}\mathbf{z} \bmod 1)]_{k, k' \in \{0, \dots, n-1\}}$ is the kernel matrix, and the vector $\mathbf{f} = [f(\frac{k}{n}\mathbf{z} \bmod 1)]_{k \in \{0, \dots, n-1\}}$ consists of the function evaluations on the lattice $\mathbf{X}$. Since

$K(\mathbf{x}, \mathbf{y}) = K(\mathbf{x} - \mathbf{y}, \mathbf{0})$, we have that $\mathbf{K}$ is a circulant matrix

$$\begin{aligned}\mathbf{K} &= \left[K\left(\frac{k}{n}\mathbf{z} \bmod 1, \frac{k'}{n}\mathbf{z} \bmod 1\right) \right]_{k, k' \in \{0, \dots, n-1\}} \\ &= \left[K\left(\frac{k-k'}{n}\mathbf{z} \bmod 1, \mathbf{0}\right) \right]_{k, k' \in \{0, \dots, n-1\}}.\end{aligned}$$

Thus, it is possible to diagonalize it by a discrete Fourier transform (DFT) matrix $\mathbf{F} = [\exp(2\pi i k k' / n)]_{k, k' \in \{0, \dots, n-1\}}$

$$\mathbf{K} = \frac{1}{n} \mathbf{F} \text{diag}(\mathbf{F}^* \mathbf{c}) \mathbf{F}^* \quad \text{and} \quad \mathbf{K}^{-1} = \frac{1}{n} \mathbf{F} \text{diag}(\mathbf{F}^* \mathbf{c})^{-1} \mathbf{F}^*,$$

with $\mathbf{c} = [K(\frac{k}{n}\mathbf{z} \bmod 1, \mathbf{0})]_{k \in \{0, \dots, n-1\}}$. The kernel approximation with lattices is therefore computable using three fast Fourier transforms in $\mathcal{O}(n \log n)$ arithmetic operations.

In terms of the error it is known that for any given set of points, the kernel approximation (2.7) is optimal among all (linear or nonlinear) algorithms using function values at the same point set in the worst-case setting with respect to any error norm, cf. [20]. In particular, we have that the worst-case error of the kernel approximation is bounded by the worst-case error of the classical lattice algorithm (2.1)

$$e^{\text{wor-app}}(A_{\text{ker}}^{\mathbf{X}}) \leq e^{\text{wor-app}}(A_{\mathcal{A}}^{\mathbf{X}}).$$

Thus, the bound from Theorem 2.1 applies to the kernel method as well.

3 Projection property of approximation algorithms

A projection is a linear, idempotent map, meaning that the projection is equal to its composition with itself. In this section we derive a condition under which the classical lattice algorithm $A_{\mathcal{A}}^{\mathbf{X}}$ defined in (2.1) is a projection.

Clearly we have linearity for $A_{\mathcal{A}}^{\mathbf{X}}$ but the idempotence is not always present. For instance, if there are two aliasing frequencies, i.e., $\mathbf{h}, \mathbf{h}' \in \mathcal{A}$ with $\langle \mathbf{h}, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}', \mathbf{z} \rangle$, we can take $f := \exp(2\pi i \langle \mathbf{h}, \cdot \rangle) + \exp(2\pi i \langle \mathbf{h}', \cdot \rangle)$. We then have $f \in \text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{A}}$ yet by (2.2)

$$A_{\mathcal{A}}^{\mathbf{X}} f = 2f \neq f.$$

Thus, to ensure idempotence it is necessary that there are no aliasing frequencies in the index set, that is, no indices have the same inner product with the generating vector $\mathbf{z}$ modulo n .

A lattice $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ is said to have the *reconstructing property* on $\mathcal{B} \subseteq \mathbb{Z}^d$ if

$$\langle \mathbf{h}, \mathbf{z} \rangle \not\equiv_n \langle \mathbf{h}', \mathbf{z} \rangle \quad \text{for all} \quad \mathbf{h}, \mathbf{h}' \in \mathcal{B}. \quad (3.1)$$

This is also sufficient for the idempotence as the following theorem shows. For ease of readability, we will write $\mathcal{B} \subseteq \mathbb{Z}^d$ when we have the reconstructing property and $\mathcal{A} \subseteq \mathbb{Z}^d$ for a generic frequency index set.

Theorem 3.1. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a nonempty frequency index set for which the lattice $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ has the reconstructing property (3.1). Then the classical lattice algorithm $A_{\mathcal{B}}^{\mathbf{X}}$ defined in (2.1) is a projection onto $\text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$.*

Proof. By definition we have that $A_{\mathcal{B}}^{\mathbf{X}}$ is linear and maps onto the linear space $\text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$. Combining the aliasing formula (2.2) with the reconstructing property of $\mathbf{X}$ on $\mathcal{B}$ as in (3.1), we obtain $\hat{g}_{\mathbf{h}} = \hat{f}_{\mathbf{h}}$ for all $\mathbf{h} \in \mathcal{B}$ and confirm the idempotence. $\blacksquare$

There are CBC constructions that compute a generating vector $\mathbf{z}$ for a given frequency index set $\mathcal{B}$ and $n \in \mathbb{N}$ large enough that the reconstructing property holds, see e.g., [24, 25]. We investigate the opposite question: given a lattice, what is the set of frequencies on which we have the reconstructing property, or equivalently what is the space on which the lattice algorithm is a projection.

Theorem 3.2. *Let $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ be a lattice. Further, let H be the function space from (2.3) and $A^{\mathbf{X}}$ a (linear or non-linear) approximation algorithm using samples of $\mathbf{X}$. Then $\mathbf{X}$ is a lattice with reconstructing property (3.1) for the frequency index set*

$$\left\{ \mathbf{h} \in \mathbb{Z}^d : r(\mathbf{h}) < (e^{\text{wor-app}}(A^{\mathbf{X}}))^{-2} \right\}. \quad (3.2)$$

In particular, the above holds for the best possible worst-case error achievable by any (linear or non-linear) algorithm using points from the lattice $\mathbf{X}$.

Proof. The proof uses ideas from [3, Theorem 3.2], which showed a lower bound on the worst-case error using samples from lattices in function spaces of dominating mixed smoothness.

We will show that for any two aliasing frequencies, at least one must lie outside of the proposed frequency index set.

Let $\mathbf{h}, \mathbf{h}' \in \mathbb{Z}^d$ with $\mathbf{h} \neq \mathbf{h}'$, $\langle \mathbf{h}, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}', \mathbf{z} \rangle$, and $r(\mathbf{h}) \leq r(\mathbf{h}')$. We define the fooling function

$$f := \frac{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle) - \exp(2\pi i \langle \mathbf{h}', \cdot \rangle)}{\sqrt{r(\mathbf{h}) + r(\mathbf{h}')}},$$

so that

$$\|f\|_H = \sqrt{\frac{r(\mathbf{h}) + r(\mathbf{h}')}{r(\mathbf{h}) + r(\mathbf{h}')}} = 1 \quad \text{and} \quad \|f\|_{L_2} = \sqrt{\frac{2}{r(\mathbf{h}) + r(\mathbf{h}')}}.$$

Because of the aliasing frequencies, this function evaluates to zero on the lattice $\mathbf{X}$, i.e.,

$$f(\frac{k}{n}\mathbf{z} \bmod 1) = \frac{\exp(2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle) - \exp(2\pi i \frac{k}{n} \langle \mathbf{h}', \mathbf{z} \rangle)}{\sqrt{r(\mathbf{h}) + r(\mathbf{h}')}} = 0 \quad \text{for } k \in \{0, \dots, n-1\}.$$

If $A^{\mathbf{X}}$ were to be linear, we would have $A^{\mathbf{X}}(f) \equiv 0$ and $\|f\|_{L_2} = \|f - A^{\mathbf{X}}(f)\|_{L_2} \leq e^{\text{wor-app}}(A^{\mathbf{X}})$. Since we also include non-linear algorithms we only have $A^{\mathbf{X}}(f) =$

$A^{\mathbf{X}}(-f)$ and need an intermediate step for the same result:

$$\begin{aligned}
\|f\|_{L_2} &= \frac{1}{2} \|f - A^{\mathbf{X}}(f) + f + A^{\mathbf{X}}(-f)\|_{L_2} \\
&\leq \frac{1}{2} \left(\|f - A^{\mathbf{X}}(f)\|_{L_2} + \|(-f) - A^{\mathbf{X}}(-f)\|_{L_2} \right) \\
&\leq \max \left\{ \|f - A^{\mathbf{X}}(f)\|_{L_2}, \|(-f) - A^{\mathbf{X}}(-f)\|_{L_2} \right\} \\
&\leq e^{\text{wor-app}}(A^{\mathbf{X}}).
\end{aligned}$$

Thus

$$\frac{1}{r(\mathbf{h}')} \leq \frac{2}{r(\mathbf{h}) + r(\mathbf{h}')} = \|f\|_{L_2}^2 \leq e^{\text{wor-app}}(A^{\mathbf{X}})^2.$$

Since this holds for any two aliasing frequencies, at least one of the two aliasing frequencies must lie outside of the proposed frequency index set (3.2), which is equivalent to the reconstructing property. $\blacksquare$

Applying this to the classical lattice algorithm, we obtain the following result.

Corollary 3.3. *Let $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ be a lattice and $\mathcal{S}_n(\mathbf{z})$ be defined as in (2.5). Then $\mathbf{X}$ has the reconstructing property on*

$$\left\{ \mathbf{h} \in \mathbb{Z}^d : r(\mathbf{h}) \leq \frac{1}{2\sqrt{\mathcal{S}_n(\mathbf{z})}} \right\}. \quad (3.3)$$

Proof. Using the classical lattice algorithm (2.1) in Theorem 3.2, we obtain the reconstructing property on

$$\left\{ \mathbf{h} \in \mathbb{Z}^d : r(\mathbf{h}) < (e^{\text{wor-app}}(A_{\mathcal{A}}^{\mathbf{X}}))^{-2} \right\}.$$

Now we plug in the bound (2.6), which only makes the frequency index set smaller, not losing the reconstructing property. We have $\leq$ instead of $<$ in (3.3) since equality is not attained in (2.1). $\blacksquare$

Note, the frequency index set (3.3) has half of the radius of the frequency index set used in the classical lattice algorithm in Theorem 2.1, where we have $M = 1/\sqrt{\mathcal{S}_n(\mathbf{z})}$. With the latter choice we do not have the guarantee for the reconstructing property. This difference could be an artifact of the proof, however.

4 Error bounds for subsampled rank-1 lattices

In this section we investigate what happens to the projection property as well as to error bounds if we use only a subset of the lattice points $\mathbf{X}_J = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k \in J}$ for $J \subseteq \{0, \dots, n-1\}$. Instead of using the classical lattice algorithm, we will use a *least squares approximation*

$$S_B^{\mathbf{X}_J} f := \arg \min_{g \in \text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}} \sum_{k \in J} \left| g(\frac{k}{n}\mathbf{z} \bmod 1) - f(\frac{k}{n}\mathbf{z} \bmod 1) \right|^2. \quad (4.1)$$

Note that it is necessary for $\mathbf{X}$ to have the reconstructing property on $\mathcal{B}$ as otherwise there would be functions with aliasing frequencies having the same function values along the lattice and subsequently also along the subsampled lattice. This would result in the non-uniqueness of the least squares approximation. Furthermore, bad choices in J would not work as well. For instance, we need at least $|J| \geq |\mathcal{B}|$, otherwise we could construct a non-zero function that evaluates to zero on $\mathbf{X}_J$, again, yielding non-uniqueness.

Throughout this paper we will use multisets for J , meaning that elements may repeat. For the cardinality $|J|$, elements are counted according to their repeated appearance, e.g. $|\{1, 1\}| = 2$. Note that for the actual number of points used, or equivalently, the number of function evaluations used, it is not necessary to count by repeated appearance but it always provides an upper bound.

Lemma 4.1. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a nonempty frequency index set for which the lattice $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ has the reconstructing property. For $J \subseteq \{0, \dots, n-1\}$ a multiset with $|J| \geq |\mathcal{B}|$, suppose that*

$$\mathbf{L}_{J,\mathcal{B}} := \left[\exp(2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle) \right]_{k \in J, \mathbf{h} \in \mathcal{B}} \in \mathbb{C}^{|J| \times |\mathcal{B}|} \quad (4.2)$$

has full rank $|\mathcal{B}|$, and let $\mathbf{f}_J := [f(\frac{k}{n}\mathbf{z} \bmod 1)]_{k \in J}$ for $f: \mathbb{T}^d \rightarrow \mathbb{C}$. Then for the least squares approximation $S_{\mathcal{B}}^{\mathbf{X}_J} f = \sum_{\mathbf{h} \in \mathcal{B}} \hat{g}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle)$ defined in (4.1), we have the Fourier coefficients

$$\hat{\mathbf{g}}_{\mathcal{B}} := [\hat{g}_{\mathbf{h}}]_{\mathbf{h} \in \mathcal{B}} = (\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}})^{-1} \mathbf{L}_{J,\mathcal{B}}^* \mathbf{f}_J.$$

Proof. The definition of the least squares approximation (4.1) can be rewritten in matrix-vector form using the wanted Fourier coefficients

$$\arg \min_{\hat{\mathbf{g}}_{\mathcal{B}} \in \mathbb{C}^{|\mathcal{B}|}} \|\mathbf{L}_{J,\mathcal{B}} \hat{\mathbf{g}}_{\mathcal{B}} - \mathbf{f}_J\|_2^2.$$

Computing this minimum, one uses the normal equation

$$\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}} \hat{\mathbf{g}}_{\mathcal{B}} = \mathbf{L}_{J,\mathcal{B}}^* \mathbf{f}_J.$$

By assumption the left-hand side matrix $\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}}$ has full rank and can therefore be inverted, yielding the assertion. $\blacksquare$

Note, that the full rank condition already implies that $\mathbf{X}$ has to have the reconstructing property on $\mathcal{B}$ and $|J| \geq |\mathcal{B}|$. Next we show that given full rank of $\mathbf{L}_{J,\mathcal{B}}$, the least squares approximation still maintains the projection property.

Lemma 4.2. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a nonempty frequency index set for which the lattice $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ has the reconstructing property. For $J \subseteq \{0, \dots, n-1\}$ a multiset with $|J| \geq |\mathcal{B}|$, suppose $\mathbf{L}_{J,\mathcal{B}}$ from (4.2) has full rank $|\mathcal{B}|$. Then the least squares approximation $S_{\mathcal{B}}^{\mathbf{X}_J}$ defined in (4.1), with the subsampled lattice $\mathbf{X}_J$ is a projection onto $\text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$.*

Proof. By definition we have that $S_{\mathcal{B}}^{\mathbf{X}_J}$ maps linearly into $\text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$. To show the idempotence, we use the Fourier representation of a function supported on $\mathcal{B}$

$$f = \sum_{\mathbf{h} \in \mathcal{B}} \widehat{f}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle) \in \text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$$

and collect the Fourier coefficients in the vector $\widehat{\mathbf{f}}_{\mathcal{B}} = [\widehat{f}_{\mathbf{h}}]_{\mathbf{h} \in \mathcal{B}}$. From the reconstructing property, there holds $\mathbf{f}_J = \mathbf{L}_{J,\mathcal{B}} \widehat{\mathbf{f}}_{\mathcal{B}}$. By Lemma 4.1 we have for the Fourier coefficients of the least squares approximation

$$\widehat{\mathbf{g}}_{\mathcal{B}} = (\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}})^{-1} \mathbf{L}_{J,\mathcal{B}}^* \mathbf{f}_J = (\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}})^{-1} \mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}} \widehat{\mathbf{f}}_{\mathcal{B}} = \widehat{\mathbf{f}}_{\mathcal{B}},$$

where the inverse exists because of the assumptions on $\mathbf{L}_{J,\mathcal{B}}$ having full rank. The Fourier coefficients of the least squares approximation coincide with those of the function f . Thus, they are the same functions, which proves the assertion. $\blacksquare$

Before commenting on a specific choice in J , we relate the worst-case error of the least squares approximation to the spectral properties of certain matrices.

Theorem 4.3. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a nonempty frequency index set for which the lattice $\mathbf{X} = \{\frac{k}{n} \mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ has the reconstructing property. For $J \subseteq \{0, \dots, n-1\}$ a multiset with $|J| \geq |\mathcal{B}|$, suppose $\mathbf{L}_{J,\mathcal{B}}$ in (4.2) has full rank, and define*

$$\Phi_{J,\mathcal{B}} := \left[\frac{\exp(2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle)}{\sqrt{r(\mathbf{h})}} \right]_{k \in J, \mathbf{h} \notin \mathcal{B}}. \quad (4.3)$$

Then the worst-case error of the least squares approximation defined in (4.1) with respect to the function space defined in (2.3) satisfies the bound

$$e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 \leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \frac{\|\Phi_{J,\mathcal{B}}\|_2^2}{\sigma_{\min}^2(\mathbf{L}_{J,\mathcal{B}})},$$

where $\sigma_{\min}(\mathbf{L}_{J,\mathcal{B}})$ denotes the smallest singular value of $\mathbf{L}_{J,\mathcal{B}}$.

Proof. For $f \in H$ we write $f - S_{\mathcal{B}}^{\mathbf{X}_J} f = (f - P_{\mathcal{B}} f) + (P_{\mathcal{B}} f - S_{\mathcal{B}}^{\mathbf{X}_J} f)$, where $P_{\mathcal{B}} f = \sum_{\mathbf{h} \in \mathcal{B}} \widehat{f}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle)$ is the L_2 -projection of f onto $\text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$. This decomposition is orthogonal because by definition $S_{\mathcal{B}}^{\mathbf{X}_J} f = P_{\mathcal{B}} S_{\mathcal{B}}^{\mathbf{X}_J} f$. Thus,

$$\|S_{\mathcal{B}}^{\mathbf{X}_J} f - f\|_{L_2}^2 = \|f - P_{\mathcal{B}} f\|_{L_2}^2 + \|P_{\mathcal{B}} f - S_{\mathcal{B}}^{\mathbf{X}_J} f\|_{L_2}^2. \quad (4.4)$$

For the first summand of (4.4) we use

$$\|f - P_{\mathcal{B}} f\|_{L_2}^2 = \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} r(\mathbf{h}) |\widehat{f}_{\mathbf{h}}|^2 \leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} \|f\|_H^2.$$

Now we estimate the second summand of (4.4). By Lemma 4.2, $S_{\mathcal{B}}^{\mathbf{X}_J}$ is a projection onto $\text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}$ and we obtain using Lemma 4.1

$$\begin{aligned} \|P_{\mathcal{B}}f - S_{\mathcal{B}}^{\mathbf{X}_J}f\|_{L_2}^2 &= \|S_{\mathcal{B}}^{\mathbf{X}_J}P_{\mathcal{B}}f - S_{\mathcal{B}}^{\mathbf{X}_J}f\|_{L_2}^2 = \|S_{\mathcal{B}}^{\mathbf{X}_J}(P_{\mathcal{B}}f - f)\|_{L_2}^2 \\ &\leq \left\| (\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}})^{-1} \mathbf{L}_{J,\mathcal{B}}^* \left[(P_{\mathcal{B}}f - f)\left(\frac{k}{n}\mathbf{z} \bmod 1\right) \right]_{k \in J} \right\|_2^2 \\ &\leq \|(\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}})^{-1} \mathbf{L}_{J,\mathcal{B}}^*\|_2^2 \sum_{k \in J} \left| (P_{\mathcal{B}}f - f)\left(\frac{k}{n}\mathbf{z} \bmod 1\right) \right|^2. \end{aligned}$$

By [23, Proposition 3.1], we have $\|(\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}})^{-1} \mathbf{L}_{J,\mathcal{B}}^*\|_2 = 1/\sigma_{\min}(\mathbf{L}_{J,\mathcal{B}})$. On the other hand, we have

$$\begin{aligned} \sum_{k \in J} \left| (f - P_{\mathcal{B}}f)\left(\frac{k}{n}\mathbf{z} \bmod 1\right) \right|^2 &= \sum_{k \in J} \left| \sum_{\mathbf{h} \notin \mathcal{B}} \hat{f}_{\mathbf{h}} \exp(2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle) \right|^2 \\ &= \sum_{k \in J} \left| \sum_{\mathbf{h} \notin \mathcal{B}} \sqrt{r(\mathbf{h})} \hat{f}_{\mathbf{h}} \frac{1}{\sqrt{r(\mathbf{h})}} \exp(2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle) \right|^2 \\ &= \|\Phi_{J,\mathcal{B}}[\sqrt{r(\mathbf{h})} \hat{f}_{\mathbf{h}}]_{\mathbf{h} \notin \mathcal{B}}\|_2^2 \\ &\leq \|\Phi_{J,\mathcal{B}}\|_2^2 \|f\|_H^2. \end{aligned}$$

Combining the two estimates completes the proof ■

We have seen in Lemma 4.2 that it is possible to reconstruct trigonometric polynomials from values at a subsampled lattice $\mathbf{X}_J = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k \in J} \subseteq \mathbb{T}^d$ for $J \subseteq \{0, \dots, n-1\}$ given that $\mathbf{L}_{J,\mathcal{B}}$ has full rank. The following lemma quantifies this for a randomly selected subset J .

Lemma 4.4. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a nonempty frequency index set for which the lattice $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ has the reconstructing property. Further, let $t > 0$ and $J \subseteq \{0, \dots, n-1\}$ be a multiset of uniformly i.i.d. drawn integers with*

$$|J| \geq 12 |\mathcal{B}| (\log |\mathcal{B}| + t). \quad (4.5)$$

For $\mathbf{L}_{J,\mathcal{B}}$ defined as in (4.2) we have the following two inequalities, each holding with probability exceeding $1 - \exp(-t)$,

$$\frac{|J|}{2} \leq \sigma_{\min}^2(\mathbf{L}_{J,\mathcal{B}}) \quad \text{and} \quad \sigma_{\max}^2(\mathbf{L}_{J,\mathcal{B}}) \leq \frac{3|J|}{2}. \quad (4.6)$$

For $|J| \geq 3$, $\Phi_{J,\mathcal{B}}$ as in (4.3), and $\Phi_{\mathcal{B}} = \Phi_{\{0, \dots, n-1\}, \mathcal{B}}$, we have with probability exceeding $1 - 2^{3/4} \exp(-t)$ that

$$\|\Phi_{J,\mathcal{B}}\|_2^2 \leq 21 (\log |J| + t) \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + 2 \frac{|J|}{n} \|\Phi_{\mathcal{B}}\|_2^2. \quad (4.7)$$

Proof. By [1, Lemma 2.2] the reconstructing property of $\mathbf{X}$ corresponds to an exact L_2 -Marcinkiewicz–Zygmund inequality, i.e.,

$$\|f\|_{L_2}^2 = \frac{1}{n} \sum_{k=0}^{n-1} |f(\frac{k}{n}\mathbf{z} \bmod 1)|^2 \quad \text{for all } f \in \text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}}.$$

With the logarithmic oversampling assumption (4.5) on J , we are able to use the technique of [1, Theorem 3.1], which yields that the subsampled points fulfill an L_2 -Marcinkiewicz–Zygmund inequality with constants $1/2$ and $3/2$, i.e.,

$$\frac{1}{2} \|f\|_{L_2}^2 \leq \frac{1}{|J|} \sum_{k \in J} |f(\frac{k}{n}\mathbf{z} \bmod 1)|^2 \leq \frac{3}{2} \|f\|_{L_2}^2 \quad \text{for all } f \in \text{span}\{\exp(2\pi i \langle \mathbf{h}, \cdot \rangle)\}_{\mathbf{h} \in \mathcal{B}},$$

where from their proof each bound holds with probability $1 - \exp(-t)$, respectively. This random technique uses a matrix Chernoff concentration inequality [42, Theorem 1.1] with one of the first applications in the sampling context in [5, Theorem 2.1]. Using [1, Lemma 2.2] again converts this to the desired bounds on the singular values in (4.6).

For $k \in \{0, \dots, n-1\}$, we define the sequences

$$\mathbf{y}^k := \left[\frac{\exp(-2\pi i \frac{k}{n} \langle \mathbf{h}, \mathbf{z} \rangle)}{\sqrt{r(\mathbf{h})}} \right]_{\mathbf{h} \notin \mathcal{B}},$$

with $\|\mathbf{y}^k\|_2^2 \leq \sum_{\mathbf{h} \notin \mathcal{B}} 1/r(\mathbf{h})$, so that

$$\frac{1}{|J|} \sum_{k \in J} \mathbf{y}^k (\mathbf{y}^k)^* = \frac{1}{|J|} \sum_{k \in J} \left[\frac{\exp(2\pi i \frac{k}{n} \langle \mathbf{h}' - \mathbf{h}, \mathbf{z} \rangle)}{\sqrt{r(\mathbf{h})r(\mathbf{h}')}} \right]_{\mathbf{h}, \mathbf{h}' \notin \mathcal{B}} = \frac{1}{|J|} \Phi_{J, \mathcal{B}}^* \Phi_{J, \mathcal{B}}.$$

The set J gives us a random set of sequences $\{\mathbf{y}^k\}_{k \in J}$ for which we know

$$\mathbb{E}_k(\mathbf{y}^k (\mathbf{y}^k)^*) = \frac{1}{n} \sum_{k=0}^{n-1} \left[\frac{\exp(2\pi i \frac{k}{n} \langle \mathbf{h}' - \mathbf{h}, \mathbf{z} \rangle)}{\sqrt{r(\mathbf{h})r(\mathbf{h}')}} \right]_{\mathbf{h}, \mathbf{h}' \notin \mathcal{B}} = \frac{1}{n} \Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}},$$

where the expectation is taken with respect to the discrete random variable k uniformly sampled from $\{0, \dots, n-1\}$. Applying [34, Proposition 3.8] on these random sequences (with n there replaced by $|J| \geq 3$) yields with probability exceeding $1 - 2^{3/4}|J|^{1-\beta}$ for $\beta > 1$ that

$$\left\| \frac{1}{|J|} \Phi_{J, \mathcal{B}}^* \Phi_{J, \mathcal{B}} - \frac{1}{n} \Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}} \right\|_2 \leq \max \left\{ \frac{21\beta \log |J|}{|J|} \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})}, \frac{1}{n} \|\Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}}\|_2 \right\}.$$

Setting $\beta = (\log |J| + t)/\log |J|$ for $t > 0$, this gives with probability exceeding $1 - 2^{3/4} \exp(-t)$ that

$$\left\| \frac{1}{|J|} \Phi_{J, \mathcal{B}}^* \Phi_{J, \mathcal{B}} - \frac{1}{n} \Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}} \right\|_2 \leq \max \left\{ \frac{21(\log |J| + t)}{|J|} \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})}, \frac{1}{n} \|\Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}}\|_2 \right\},$$

from which the assertion (4.7) follows by basic arithmetic operations. ■

Using these spectral bounds in the general error bound for least squares approximation yields the following theorem.

Theorem 4.5. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a nonempty frequency index set for which the lattice $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ has the reconstructing property. Further, let $t \geq 4$ and $J \subseteq \{0, \dots, n-1\}$ be a multiset of uniformly i.i.d. drawn integers with*

$$|J| := \lceil 12 |\mathcal{B}| (\log |\mathcal{B}| + t) \rceil.$$

Then the least squares approximation defined in (4.1) using samples on the subsampled lattice $\mathbf{X}_J$ satisfies with probability exceeding $1 - 3 \exp(-t)$ that

$$e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 \leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \frac{42 (\log |J| + t)}{|J|} \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \frac{4}{n} \|\Phi_{\mathcal{B}}\|_2^2,$$

with respect to the function space defined in (2.3) and $\Phi_{\mathcal{B}} = \Phi_{\{0, \dots, n-1\}, \mathcal{B}}$ as in (4.3). Additionally we have

$$\frac{6 (\log |J| + t)}{|J|} \leq \frac{1}{|\mathcal{B}|} \leq \frac{13 (\log |J| + t)}{|J|}. \quad (4.8)$$

Proof. We use the bound of the smallest singular value of $\mathbf{L}_{J, \mathcal{B}}$ and the bound on the spectral norm of $\Phi_{J, \mathcal{B}}$ of Lemma 4.4. By Boole's inequality, the overall probability exceeds $1 - \exp(-t) - 2^{3/4} \exp(-t) \geq 1 - 3 \exp(-t)$, which is the claimed probability. Plugging in the spectral bounds into Theorem 4.3, we obtain

$$e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 \leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \frac{2}{|J|} \left(21 (\log |J| + t) \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + 2 \frac{|J|}{n} \|\Phi_{\mathcal{B}}\|_2^2 \right).$$

It remains to estimate the scaling factor for the sum.

We first show that for $b \geq 1$ and $t \geq 4$ we have

$$12b (\log b + t) \leq b^2 \exp(t). \quad (4.9)$$

Since $\log b \leq b - 1$ this follows from the stronger assertion

$$12(b - 1 + t) \leq b \exp(t) \quad \Leftrightarrow \quad (\exp(t) - 12)b + 12 - 12t \geq 0.$$

The left-hand side is non-decreasing in b since $t \geq 4 \geq \log(12)$. With $b \geq 1$, it remains to check the non-negativity for $b = 1$, which is satisfied for $t \geq 4$ showing (4.9).

We are ready to prove an upper bound on $(\log j + t)/j$ in terms of $1/b$ when $j := \lceil 12b (\log b + t) \rceil \geq 12b (\log b + t)$. Since $(\log j + t)/j$ is increasing with decreasing j , it suffices to show an estimate for $\ell \leq j$ such that

$$12b (\log b + t) \leq \ell \leq b^2 \exp(t).$$

Such a value of $\ell \in \mathbb{R}$ exists due to (4.9). Then it follows that

$$\frac{\log j + t}{j} \leq \frac{\log \ell + t}{\ell} \leq \frac{2 (\log b + t)}{12b (\log b + t)} = \frac{1}{6b}.$$

Finally we prove an upper bound on $1/b$ in terms of $(\log j + t)/j$. Since $b \geq 1$ and $t \geq 4$, we have $12b(\log b + t) \geq \frac{48}{49} \lceil 12b(\log b + t) \rceil$. Thus

$$\frac{1}{b} \leq \frac{49}{48} \frac{12(\log b + t)}{\lceil 12b(\log b + t) \rceil} \leq \frac{13(\log j + t)}{j},$$

where we used $j > 12b \geq b$ for $\log b < \log j$ and $49 \cdot 12/48 < 13$. This completes the proof. $\blacksquare$

In Theorem 4.5 the spectral norm of an infinite-dimensional matrix $\|\Phi_{\mathcal{B}}\|_2^2$ occurs. We have

$$\|\Phi_{\mathcal{B}}\|_2^2 = \|\Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}}\|_2 = \|\Phi_{\mathcal{B}} \Phi_{\mathcal{B}}^*\|_2,$$

where the first matrix product is infinite-dimensional, whereas the second matrix-product is finite-dimensional (may be used for computing the spectral norm numerically). In the following lemma we use the former to show that this quantity can be upper bounded in terms of the known quantity $\mathcal{S}_n(\mathbf{z})$ from (2.5). Note that the reconstructing property is not needed in this lemma.

Lemma 4.6. *Let $\mathcal{B} \subseteq \mathbb{Z}^d$ be a frequency index set and $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ a lattice. Further let $\Phi_{\mathcal{B}} := \Phi_{\{0, \dots, n-1\}, \mathcal{B}}$ be as in (4.3) and $\mathcal{S}_n(\mathbf{z})$ as in (2.5). Then*

$$\frac{1}{n} \|\Phi_{\mathcal{B}}\|_2^2 \leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \sqrt{\mathcal{S}_n(\mathbf{z})}.$$

Proof. By the definition of $\Phi_{\mathcal{B}}$, we have

$$\frac{1}{n} \|\Phi_{\mathcal{B}}\|_2^2 = \left\| \frac{1}{n} \Phi_{\mathcal{B}}^* \Phi_{\mathcal{B}} \right\|_2 = \left\| \left[\frac{1}{n} \sum_{k=0}^{n-1} \frac{\exp(2\pi i \frac{k}{n} \langle \mathbf{h}' - \mathbf{h}, \mathbf{z} \rangle)}{\sqrt{r(\mathbf{h}) r(\mathbf{h}')}} \right]_{\mathbf{h}, \mathbf{h}' \notin \mathcal{B}} \right\|_2.$$

By the character property $\frac{1}{n} \sum_{k=0}^{n-1} \exp(2\pi i m \frac{k}{n}) = \delta_{m \equiv 0}$ this evaluates to

$$\frac{1}{n} \|\Phi_{\mathcal{B}}\|_2^2 = \left\| \left[\frac{\delta_{\langle \mathbf{h}, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}', \mathbf{z} \rangle}}{\sqrt{r(\mathbf{h}) r(\mathbf{h}')}} \right]_{\mathbf{h}, \mathbf{h}' \notin \mathcal{B}} \right\|_2.$$

Now we split this matrix into its diagonal and off-diagonal terms

$$\frac{1}{n} \|\Phi_{\mathcal{B}}\|_2^2 \leq \left\| \text{diag} \left(\frac{1}{r(\mathbf{h})} \right)_{\mathbf{h} \notin \mathcal{B}} \right\|_2 + \left\| \left[\frac{\delta_{\mathbf{h} \neq \mathbf{h}' \delta_{\langle \mathbf{h}, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}', \mathbf{z} \rangle}}{\sqrt{r(\mathbf{h}) r(\mathbf{h}')}} \right]_{\mathbf{h}, \mathbf{h}' \notin \mathcal{B}} \right\|_2.$$

The spectral norm of the diagonal matrix is known and for the off-diagonal matrix we estimate it by the Frobenius norm

$$\begin{aligned} \frac{1}{n} \|\Phi_{\mathcal{B}}\|_2^2 &\leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \left\| \left[\frac{\delta_{\mathbf{h} \neq \mathbf{h}' \delta_{\langle \mathbf{h}, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}', \mathbf{z} \rangle}}{\sqrt{r(\mathbf{h}) r(\mathbf{h}')}} \right]_{\mathbf{h}, \mathbf{h}' \in \mathbb{Z}^d} \right\|_F \\ &= \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \left(\sum_{\mathbf{h} \in \mathbb{Z}^d} \sum_{\substack{\mathbf{h}' \in \mathbb{Z}^d \\ \mathbf{h}' \neq \mathbf{h}}} \frac{\delta_{\langle \mathbf{h}, \mathbf{z} \rangle \equiv_n \langle \mathbf{h}', \mathbf{z} \rangle}}{r(\mathbf{h}) r(\mathbf{h}')} \right)^{1/2} \\ &= \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \left(\sum_{\mathbf{h} \in \mathbb{Z}^d} \frac{1}{r(\mathbf{h})} \sum_{\substack{\ell \in \mathbb{Z}^d \setminus \{\mathbf{0}\} \\ \langle \ell, \mathbf{z} \rangle \equiv_n 0}} \frac{1}{r(\mathbf{h} + \ell)} \right)^{1/2}, \end{aligned}$$

where the second term is exactly $\sqrt{\mathcal{S}_n(\mathbf{z})}$ as claimed. $\blacksquare$

Now we fix the frequency set $\mathcal{B}$ to be as in Corollary 3.3, where we have guaranteed reconstructing property, allowing us to omit it in the assumptions.

Theorem 4.7. *Let $\mathbf{X} = \{\frac{k}{n}\mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ be a lattice and $\mathcal{S}_n(\mathbf{z})$ as in (2.5). Further, let H be the function space defined in (2.3) and fix*

$$\mathcal{B} := \left\{ \mathbf{h} \in \mathbb{Z}^d : r(\mathbf{h}) \leq \frac{1}{2\sqrt{\mathcal{S}_n(\mathbf{z})}} \right\}.$$

For $t \geq 4$, let $J \subseteq \{0, \dots, n-1\}$ be a multiset of uniformly i.i.d. drawn integers with

$$|J| := \lceil 12 |\mathcal{B}| (\log |\mathcal{B}| + t) \rceil.$$

Then the least squares approximation defined in (4.1) using samples on the subsampled lattice $\mathbf{X}_J$ satisfies with probability exceeding $1 - 3\exp(-t)$ that

$$e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 \leq 14\sqrt{\mathcal{S}_n(\mathbf{z})} + \frac{7}{|\mathcal{B}|} \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})}.$$

Proof. By Corollary 3.3 we know that $\mathbf{X}$ has the reconstructing property for $\mathcal{B}$. Thus, we can apply Theorem 4.5, where we plug in Lemma 4.6 for estimating $\|\Phi_{\mathcal{B}}\|_2^2$ and obtain with the desired probability

$$e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 \leq \sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \frac{7}{|\mathcal{B}|} \sum_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + 4 \left(\sup_{\mathbf{h} \notin \mathcal{B}} \frac{1}{r(\mathbf{h})} + \sqrt{\mathcal{S}_n(\mathbf{z})} \right).$$

By the definition of $\mathcal{B}$, we have $\sup_{\mathbf{h} \notin \mathcal{B}} 1/r(\mathbf{h}) \leq 2\sqrt{\mathcal{S}_n(\mathbf{z})}$, which, when plugged in the above estimate, yields the assertion. $\blacksquare$

Remark 4.8. *Following the same proof, it is easily seen, that Theorem 4.7 still holds if we replace $\mathcal{S}_n(\mathbf{z})$ by any upper estimate, even though it changes the definition of $\mathcal{B}$ and with that the underlying method.*

Note that the occurrence of the quantity $1/|\mathcal{B}| \sum_{\mathbf{h} \notin \mathcal{B}} 1/r(\mathbf{h})$ in Theorem 4.7 is natural, since there exist function spaces where this is a lower bound on the sampling numbers with $|\mathcal{B}|$ samples, cf. [18, Theorem 2]. Comparing the full lattice algorithm Theorem 2.1 with Theorem 4.7, both bounds depend on the term $\sqrt{\mathcal{S}_n(\mathbf{z})}$. The difference is in the number of used samples being $|J|$ in the new algorithm instead of n , with $|J|$ usually far less than n , as we will see in Section 5.

5 Application to weighted Korobov spaces

In this section we apply the general theory to weighted Korobov spaces to obtain double the main rate of convergence for the subsampled lattice in comparison to the full lattice.

For dimension $d \in \mathbb{N}$, smoothness $\alpha > 1/2$, and weight parameters $\gamma = \{\gamma_{\mathbf{u}}\}_{\mathbf{u} \subseteq \mathbb{N}}$, the *weighted Korobov space* is defined by

$$H_{d,\alpha,\gamma} := \left\{ f: \mathbb{T}^d \rightarrow \mathbb{C} : \|f\|_{d,\alpha,\gamma}^2 = \sum_{\mathbf{h} \in \mathbb{Z}^d} r_{d,\alpha,\gamma}(\mathbf{h}) |\widehat{f}_{\mathbf{h}}|^2 < \infty \right\}, \quad (5.1)$$

with

$$r_{d,\alpha,\gamma}(\mathbf{h}) := \gamma_{\text{supp}(\mathbf{h})}^{-1} \prod_{j \in \text{supp}(\mathbf{h})} |h_j|^{2\alpha} \quad \text{and} \quad \text{supp}(\mathbf{h}) := \left\{ j \in \{1, \dots, d\} : h_j \neq 0 \right\}. \quad (5.2)$$

Here, the weight $\gamma_{\mathbf{u}}$ controls the importance of the part of the function f which only depends on the variables $\mathbf{x}_{\mathbf{u}} = [x_j]_{j \in \mathbf{u}}$. We fix $\gamma_{\emptyset} := 1$ such that $\|1\|_{d,\alpha,\gamma} = \|1\|_{L_2}$. For integer $\alpha \in \mathbb{N}$ it is known that

$$\|f\|_{d,\alpha,\gamma}^2 = \sum_{\mathbf{u} \subseteq \{1, \dots, d\}} \frac{1}{(2\pi)^{\alpha|\mathbf{u}|}} \frac{1}{\gamma_{\mathbf{u}}} \int_{[0,1]^{|\mathbf{u}|}} \left| \int_{[0,1]^{d-|\mathbf{u}|}} \left(\prod_{j \in \mathbf{u}} \frac{\partial}{\partial x_j} \right)^{\alpha} f(\mathbf{x}) \, d\mathbf{x}_{\{1, \dots, d\} \setminus \mathbf{u}} \right|^2 d\mathbf{x}_{\mathbf{u}}.$$

Korobov spaces describe a wide class of functions, with various forms of weights γ used throughout literature, such as

- $\gamma_{\mathbf{u}} = 1$ for all $\mathbf{u} \subseteq \mathbb{N}$: these spaces coincide with classical (unweighted) periodic Sobolev spaces with dominating mixed smoothness, see e.g., [13];
- $\gamma_{\mathbf{u}} = \prod_{j \in \mathbf{u}} \gamma_j$ with a positive sequence $(\gamma_j)_{j \geq 1}$: these are called product weights, see e.g., [40, 44];
- $\gamma_{\mathbf{u}} = \Gamma_{|\mathbf{u}|}$ for a sequence $(\Gamma_{\ell})_{\ell \geq 0}$: they are called order dependent weights, see e.g., [11];
- combining product and order dependent weights yield POD weights $\gamma_{\mathbf{u}} = \Gamma_{|\mathbf{u}|} \prod_{j \in \mathbf{u}} \gamma_j$, see e.g., [32, 15, 31, 16];
- including the smoothness parameter yields smoothness driven product and order dependent (SPOD) weights $\gamma_{\mathbf{u}} = \sum_{\nu_{\mathbf{u}} \in \{1, \dots, \alpha\}^{|\mathbf{u}|}} \Gamma_{\|\nu_{\mathbf{u}}\|_1} \prod_{j \in \mathbf{u}} \gamma_{j, \nu_j}$, see e.g., [9, 10, 19].

For the classical periodic Sobolev spaces of dominating mixed smoothness, it is known that the worst-case error behaves asymptotically as $n^{-\alpha}(\log n)^{\alpha(d-1)}$ when the approximation is based on n measurements from arbitrary linear functionals. This includes function evaluations as well as Fourier coefficients, see e.g., [38] or [13, Section 4.1]. It was finally shown in [12] that the same rate of convergence can be attained when the approximation was limited to use function evaluations. This result is a non-constructive existence result for the points used in the approximation.

On the other hand, it was shown in [3, Proposition 3.3] that for an approximation based on a full lattice the worst-case error can be bounded from below by $n^{-\alpha/2}$. Thus, using the full lattice halves the polynomial rate of convergence.

It was already shown in [1], that the optimal polynomial rate can be recovered using a subset of the initial lattice given that the initial lattice is large enough. In particular, in [1] a lattice of size $n \gtrsim |J|^{2/(1-1/(2\alpha))}$ is necessary. We will show in Theorem 5.3 that the optimal polynomial rate $(|J|/\log |J|)^{-\alpha+\varepsilon}$ can be recovered using only an initial lattice of size $n \lesssim (|J|/\log |J|)^{2/\sqrt{1-\varepsilon/\alpha}}$ for any $0 < \varepsilon < \alpha$.

In order to apply our general result on subsampled lattices, we first present known results for the full lattice. Note that the parameter α in [30] is 2α in this paper.

Theorem 5.1 ([30, Theorem 3.3]). *Given $n \geq 2$, $d \in \mathbb{N}$, $\alpha > 1/2$ and weights $\gamma = \{\gamma_u\}_{u \subseteq \mathbb{N}}$, the generating vector $\mathbf{z}$ obtained from the CBC construction following [30, Algorithm 3.2] satisfies*

$$\frac{2\zeta(2\alpha)\gamma_{\{1\}}}{n^{2\alpha}} \leq \mathcal{S}_n(\mathbf{z}) \leq \left(\frac{\tau_\lambda C_\lambda^2}{\varphi(n)} \right)^{1/\lambda} \quad \text{for all } \frac{1}{2\alpha} < \lambda \leq 1,$$

where $\varphi(n) = |\{z \in \{1, \dots, n\} : \gcd(z, n) = 1\}|$ is the Euler totient function (e.g., $\varphi(n) = n-1$ for n prime), $\zeta(x) = \sum_{h=1}^{\infty} h^{-x}$ is the Riemann zeta function, $\tau_\lambda = 2^{4\alpha\lambda+1} + 1$, and

$$C_\lambda = C_{d,\alpha,\gamma,\lambda} := \sum_{u \subseteq \{1, \dots, d\}} \max\{1, |u|\} \gamma_u^\lambda (2\zeta(2\alpha\lambda))^{|u|}. \quad (5.3)$$

Proof. The trivial lower bound is obtained by keeping only the $\mathbf{h} = \mathbf{0}$ and $\ell = [\ell_1, 0, \dots, 0]$ terms in (2.5) together with $z_1 = 1$. A CBC upper bound for $\mathcal{S}_n(\mathbf{z})$ was first proved in [8] for product weights, in [6] for general weights, both for prime n , and in [30] for composite n . ■

Applying the estimate on $\mathcal{S}_n(\mathbf{z})$ from Theorem 5.1 to Theorem 2.1 achieves the order of convergence $n^{-\alpha/2+\varepsilon}$ for the full lattice, which is half the optimal order.

Lemma 5.2 ([30, Lemmata 5.2 and 5.3]). *Let $M \geq 1$, $\alpha > 1/2$, and*

$$\mathcal{A}_{d,\alpha,\gamma,M} := \left\{ \mathbf{h} \in \mathbb{Z}^d : r_{d,\alpha,\gamma}(\mathbf{h}) \leq M \right\}.$$

Then we have for $1/(2\alpha) < \lambda < 1$ and $C_\lambda = C_{d,\alpha,\gamma,\lambda}$ from (5.3) that

$$(\gamma_{\{1\}} M)^{1/(2\alpha)} \leq |\mathcal{A}_{d,\alpha,\gamma,M}| \leq C_\lambda M^\lambda$$

and

$$\frac{1}{|\mathcal{A}_{d,\alpha,\gamma,M}|} \sum_{\mathbf{h} \notin \mathcal{A}_{d,\alpha,\gamma,M}} \frac{1}{r_{d,\alpha,\gamma}(\mathbf{h})} \leq \left(\frac{C_\lambda^{1/\lambda}}{\gamma_{\{1\}}^{1/(2\alpha\lambda)}} \cdot \frac{\lambda}{1-\lambda} \right) \left(\frac{1}{M} \right)^{-\frac{1}{2\alpha\lambda}}.$$

In the original [30, Lemma 5.2] the constant C_λ equals $\sum_{u \subseteq \{1, \dots, d\}} \gamma_u^\lambda (2\zeta(2\alpha\lambda))^{|u|}$. Here we used an upper estimate (5.3) to simplify our presentation.

We now present the main result of this section which bounds the worst-case error for a subsampled lattice achieving optimal polynomial rate using the smallest initial lattice size.

Theorem 5.3. Let H be the Korobov function space defined in (5.1) with $\alpha > 1/2$ and weights $\gamma = (\gamma_u)_{j \in u}$. Let $\mathbf{X} = \{\frac{k}{n} \mathbf{z} \bmod 1\}_{k=0}^{n-1} \subseteq \mathbb{T}^d$ be a lattice with $n \geq 7$ sufficiently large so that $\mathcal{S}_n(\mathbf{z}) = \mathcal{S}_{d,\alpha,\gamma,n}(\mathbf{z}) \leq 1/4$, and let

$$\mathcal{B} := \mathcal{B}_{d,\alpha,\gamma,n} := \left\{ \mathbf{h} \in \mathbb{Z}^d : r_{d,\alpha,\gamma}(\mathbf{h}) \leq \frac{1}{2\sqrt{\mathcal{S}_n(\mathbf{z})}} \right\}.$$

For $t \geq 4$, let $J \subseteq \{0, \dots, n-1\}$ be a multiset of uniformly i.i.d. drawn integers with

$$|J| := \lceil 12 |\mathcal{B}| (\log |\mathcal{B}| + t) \rceil.$$

Then the least squares approximation defined in (4.1) using samples on the subsampled lattice $\mathbf{X}_J$ satisfies with probability exceeding $1 - 3 \exp(-t)$ that

$$e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 \leq 7 \left(1 + \frac{C_\lambda^{1/\lambda}}{\gamma_{\{1\}}^{1/(2\alpha\lambda)}} \cdot \frac{\lambda}{1-\lambda} \right) (13 C_\lambda)^{\frac{1}{2\alpha\lambda^2}} \left(\frac{\log |J| + t}{|J|} \right)^{\frac{1}{2\alpha\lambda^2}} \quad (5.4)$$

for all $1/(2\alpha) < \lambda < 1$, with $C_\lambda = C_{d,\alpha,\gamma,\lambda}$ from (5.3). Moreover, if $\mathbf{z}$ is constructed according to [30, Algorithm 3.2], we have (with $\varphi(n)$ again denoting the Euler totient function)

$$(\varphi(n))^{\frac{1}{4\alpha\lambda}} \log \varphi(n) \lesssim |J| \lesssim n^{\alpha\lambda} \log n, \quad (5.5)$$

and hence

$$\left(\frac{|J|}{\log |J|} \right)^{\frac{1}{\alpha\lambda}} \lesssim n \quad \text{and} \quad \varphi(n) \lesssim \left(\frac{|J|}{\log |J|} \right)^{4\alpha\lambda} \quad (5.6)$$

Proof. For the index set $\mathcal{B}$ with radius $M := 1/(2\sqrt{\mathcal{S}_n(\mathbf{z})}) \geq 1$, Lemma 5.2 gives for all $1/(2\alpha) < \lambda < 1$,

$$(\gamma_{\{1\}} M)^{1/(2\alpha)} \leq |\mathcal{B}| \leq C_\lambda M^\lambda \quad \Leftrightarrow \quad \frac{\gamma_{\{1\}}}{|\mathcal{B}|^{2\alpha}} \leq \frac{1}{M} \leq \left(\frac{C_\lambda}{|\mathcal{B}|} \right)^{1/\lambda}.$$

Theorem 4.7 and Lemma 5.2 then yield

$$\begin{aligned} e^{\text{wor-app}}(S_{\mathcal{B}}^{\mathbf{X}_J})^2 &\leq \frac{7}{M} + 7 \left(\frac{C_\lambda^{1/\lambda}}{\gamma_{\{1\}}^{1/(2\alpha\lambda)}} \cdot \frac{\lambda}{1-\lambda} \right) \left(\frac{1}{M} \right)^{\frac{1}{2\alpha\lambda}} \\ &\leq 7 \left(1 + \frac{C_\lambda^{1/\lambda}}{\gamma_{\{1\}}^{1/(2\alpha\lambda)}} \cdot \frac{\lambda}{1-\lambda} \right) \left(\frac{C_\lambda}{|\mathcal{B}|} \right)^{\frac{1}{2\alpha\lambda^2}}, \end{aligned}$$

from which the bound (5.4) is obtained using the upper bound on $1/|\mathcal{B}|$ in (4.8).

Next we relate $|\mathcal{B}|$ to the bounds on $M = 1/(2\sqrt{\mathcal{S}_n(\mathbf{z})})$ from Theorem 5.1, to obtain

$$\begin{aligned} \left(\frac{\varphi(n)}{\tau_\lambda C_\lambda^2} \right)^{\frac{1}{4\alpha\lambda}} \left(\frac{\gamma_{\{1\}}}{2} \right)^{\frac{1}{2\alpha}} &\leq \left(\frac{\gamma_{\{1\}}}{2\sqrt{\mathcal{S}_n(\mathbf{z})}} \right)^{\frac{1}{2\alpha}} \leq |\mathcal{B}| \leq C_\lambda \left(\frac{1}{2\sqrt{\mathcal{S}_n(\mathbf{z})}} \right)^\lambda \\ &\leq \frac{C_\lambda n^{\alpha\lambda}}{(8\zeta(2\alpha)\gamma_{\{1\}})^{\lambda/2}}. \end{aligned}$$

Since $12 |\mathcal{B}| (\log |\mathcal{B}| + t) \leq |J| \leq 13 |\mathcal{B}| (\log |\mathcal{B}| + t)$, we obtain (5.5) and (5.6). ■

In particular, for n prime, we have $(|J|/\log |J|)^{1/(\alpha\lambda)} \lesssim n \lesssim (|J|/\log |J|)^{4\alpha\lambda}$. Substituting $1/(4\alpha\lambda^2) = \alpha - \varepsilon$ for $0 < \varepsilon < \alpha$, we obtain the convergence rate $(|J|/\log |J|)^{-\alpha+\varepsilon}$ with $(|J|/\log |J|)^{2\sqrt{1-\varepsilon/\alpha}} \lesssim n \lesssim (|J|/\log |J|)^{2/\sqrt{1-\varepsilon/\alpha}}$, where the exponent is arbitrarily close to 2.

Note that for small n it is possible for the theoretically defined value of $|J|$ to be bigger than n . Additionally, the constants in the asymptotic bounds (5.5) and (5.6) are quite loose. Later in our numerical experiments we will choose instead $|J| = \lceil \sqrt{n} \log(n) \rceil$.

6 Implementation and computational cost

The goal of subsampling from a lattice is to make use of the underlying structure, where fast and memory-efficient algorithms are applicable. Without that structure uniformly generated random points would suffice for a similar error bound, cf. [29, 27].

Both the least squares approximation and the kernel method involve solving a system of equations. For the full lattice closed form solutions are known, but for the subsampled lattice it seems that an analytical inverse is not available. Direct solvers would have a cubic runtime with respect to the number of points $|J|$. Instead, we will apply iterative methods: the least squares (LSQR) [35] and the conjugate gradient (CG) [17] methods, where only matrix-vector products are needed. The following result bounds the number of necessary iterations depending on the desired accuracy.

Theorem 6.1 ([17, Theorem 3.1.1]). *Let $\mathbf{A}$ be a Hermitian positive definite matrix with condition number $\kappa_2(\mathbf{A}) = \lambda_{\max}(\mathbf{A})/\lambda_{\min}(\mathbf{A})$. Further, let $\mathbf{x}^{(i)}$ be the i -th iterate of the conjugate gradient (CG) method applied to the system of equations $\mathbf{A}\tilde{\mathbf{x}} = \mathbf{b}$. Then it holds*

$$\frac{\|\mathbf{x}^{(i)} - \tilde{\mathbf{x}}\|_2}{\sqrt{\kappa_2(\mathbf{A})}\|\tilde{\mathbf{x}}\|_2} \leq \sqrt{\frac{(\mathbf{x}^{(i)} - \tilde{\mathbf{x}})^* \mathbf{A} (\mathbf{x}^{(i)} - \tilde{\mathbf{x}})}{(\tilde{\mathbf{x}})^* \mathbf{A} \tilde{\mathbf{x}}}} \leq 2 \left(\frac{\sqrt{\kappa_2(\mathbf{A})} - 1}{\sqrt{\kappa_2(\mathbf{A})} + 1} \right)^i \leq 2 \exp \left(\frac{-2i}{\sqrt{\kappa_2(\mathbf{A})}} \right).$$

Proof. The first inequality follows from $\lambda_{\min}(\mathbf{B})\|\mathbf{x}\|_2 \leq \|\mathbf{B}\mathbf{x}\|_2 \leq \lambda_{\max}(\mathbf{B})\|\mathbf{x}\|_2$ for any Hermitian positive definite matrix $\mathbf{B}$ and vector $\mathbf{x}$. The second inequality is the statement of [17, Theorem 3.1.1]. For the last inequality, we need that for $x \geq 1$

$$\frac{x-1}{x+1} \leq \exp \left(-\frac{2}{x} \right),$$

or equivalently

$$g(x) := \log \left(\frac{x-1}{x+1} \right) + \frac{2}{x} \leq 0.$$

For x approaching 1 the statement is true. For $x > 1$, we have $g'(x) > 0$. Together with $\lim_{x \rightarrow \infty} g(x) = 0$ this gives the assertion. $\blacksquare$

The LSQR method is analytically equivalent to applying CG to the normal equation but in a numerically more stable manner, cf. [35]. Thus the iteration bound in Theorem 6.1 holds for LSQR with the minimal and maximal eigenvalue replaced by the squared singular numbers.

6.1 Fast least squares approximation with subsampled lattices

In this section we discuss how to compute the least squares approximation for a subsampled lattice making use of the Fast Fourier Transform (FFT). Let $\mathcal{B} := \{\mathbf{h} \in \mathbb{Z}^d : r(\mathbf{h}) \leq M\}$ and

$$S_{\mathcal{B}}^{\mathbf{X}_J} f = \sum_{\mathbf{h} \in \mathcal{B}} \hat{g}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \cdot \rangle)$$

be defined in (4.1). The approximation is computed once the Fourier coefficients $\hat{g}_{\mathcal{B}} = (\hat{g}_{\mathbf{h}})_{\mathbf{h} \in \mathcal{B}}$ representing the approximation are obtained. Essential is the set of frequencies $\mathcal{B}$. Because of the downward-closed structure, this can be set up incrementally in dimension. For the step from dimension j to $j+1$, we have frequencies supported in the first j dimensions and for every one of them we seek all h_{j+1} such that $r_{d,\alpha,\gamma}([h_1, \dots, h_j, h_{j+1}, 0, \dots, 0]) \leq M$. This has a computational cost of $\mathcal{O}(d|\mathcal{B}|)$. We then compute the coefficients by solving the system of equations in Lemma 4.1 via LSQR.

By (4.6) in Lemma 4.4, we have, with $|J|$ sufficiently large and high probability, that $\kappa_2(\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}}) = \lambda_{\max}(\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}}) / \lambda_{\min}(\mathbf{L}_{J,\mathcal{B}}^* \mathbf{L}_{J,\mathcal{B}}) \leq 3$. Fixing the desired accuracy to machine precision $\mathbf{eps} = 10^{-16}$, we obtain by Theorem 6.1 a fixed number of iterations for LSQR. Each iteration uses one matrix-vector product with $\mathbf{L}_{J,\mathcal{B}}$ and one with $\mathbf{L}_{J,\mathcal{B}}^*$. For the full lattice $J = \{1, \dots, n\}$ this matrix-vector product simplifies to a one-dimensional FFT with a computational cost of

$$\mathcal{O}(n \log n + |\mathcal{B}|) \text{ flops and } \mathcal{O}(n) \text{ memory requirements,}$$

which is known as the Lattice Fast Fourier Transform (LFFT), cf. [26]. The multiplication with the matrix corresponding to the subsampled lattice can be done using the full lattice by

$$\mathbf{L}_{J,\mathcal{B}} = \mathbf{P} \mathbf{L}_{\mathcal{B}} \quad \text{with} \quad \mathbf{P} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & & & & & \vdots & \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \end{pmatrix} \begin{matrix} 1 \\ 2 \\ \vdots \\ |J| \end{matrix} \in \{0,1\}^{|J| \times n}, \quad (6.1)$$

$\begin{matrix} j_1 & & j_2 & & & & j_{|J|} \end{matrix}$

which has thus the same computational cost. Note, we essentially have an FFT where we only use parts of the result. This is known as “pruned FFT”, which are partial FFTs with cost $\mathcal{O}(n \log |J|)$. As current FFT implementations are highly tuned this gain is not observed in practice.

With $|J| \sim \sqrt{n} \log n$ and $|\mathcal{B}| \sim \sqrt{n}$ as in Theorem 5.3 we obtain the computational cost in the upper two rows of Table 1 when using the LFFT compared to the naive matrix-vector product. The iteration count does not appear here as it is bounded by a general constant independent of the number of points.

Note, when using the full lattice an iterative solver is not needed, since we have by the reconstructing property $(\mathbf{L}_{\mathcal{B}}^* \mathbf{L}_{\mathcal{B}})^{-1} = (1/n) \mathbf{I}$. This can be also viewed as LSQR converging with one iteration because $\kappa_2(\mathbf{L}_{\mathcal{B}}^* \mathbf{L}_{\mathcal{B}}) = 1$ in Theorem 6.1. Thus, using the full lattice has computational cost $\mathcal{O}(n \log n)$ as well.

Subsampled method	Flops	Memory
least squares approx. (naive)	$\mathcal{O}(J \mathcal{B}) = \mathcal{O}(n \log n)$	$\mathcal{O}(J \mathcal{B}) = \mathcal{O}(n \log n)$
least squares approx. using LFFT	$\mathcal{O}(n \log n)$	$\mathcal{O}(n)$
kernel method (naive)	$\mathcal{O}(r J ^2) = \mathcal{O}(r n \log n)$	$\mathcal{O}(J ^2) = \mathcal{O}(n \log n)$
kernel method using FFT	$\mathcal{O}(r n \log n)$	$\mathcal{O}(n)$

Table 1: Computational cost for the least squares approximation $S_B^{\mathbf{X}_J}$ and the kernel method $A_{\text{ker}}^{\mathbf{X}_J}$ with a subsampled lattice of size $|J| = \lceil \sqrt{n} \log n \rceil$ utilizing the FFT compared to the naive matrix-vector product. Here r is the number of iterations depending on the condition number of the kernel matrix.

6.2 Fast kernel method with subsampled lattices

We further comment on the kernel method, which we introduced in Section 2 for the full lattice. Since $H_{d,\alpha,\gamma}$ is a reproducing kernel Hilbert space, it has an associated reproducing kernel $K(\cdot, \cdot) = K_{d,\alpha,\gamma}(\cdot, \cdot)$. For integer smoothness $\alpha \in \mathbb{N}$, a simple closed form is known:

$$K(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{u} \subseteq \{1, \dots, d\}} \gamma_{\mathbf{u}} \prod_{j \in \mathbf{u}} \eta_{\alpha}(x_j, x'_j) \quad \text{with} \quad \eta_{\alpha}(x, x') = \frac{(2\pi)^{2\alpha} B_{2\alpha}((x - x') \bmod 1)}{(-1)^{\alpha+1} (2\alpha)!},$$

where $B_{2\alpha}$ is the Bernoulli polynomial of degree 2α .

The *kernel approximation* of f based on the sampling set $\mathbf{X}_J$ is defined analogously to (2.7) by

$$A_{\text{ker}}^{\mathbf{X}_J} f := \sum_{k \in J} a_k K(\cdot, \frac{k}{n} \mathbf{z} \bmod 1) \quad \text{with} \quad \mathbf{a} = (a_k)_{k \in J} = \mathbf{K}_J^{-1} \mathbf{f}_J \quad (6.2)$$

where $\mathbf{K}_J = [K(\frac{k}{n} \mathbf{z} \bmod 1, \frac{k'}{n} \mathbf{z} \bmod 1)]_{k, k' \in J}$ is the so-called kernel matrix and $\mathbf{f}_J = [f(\frac{k}{n} \mathbf{z} \bmod 1)]_{k \in J}$.

As discussed in Section 2, the kernel method is optimal in the worst-case setting for any given set of points. In particular, the bound from Theorem 5.3 applies as well, since $e^{\text{wor-app}}(A_{\text{ker}}^{\mathbf{X}_J}) \leq e^{\text{wor-app}}(S_B^{\mathbf{X}_J})$. Similar to $\mathbf{L}_{J,\mathcal{B}}$, a fast multiplication with the kernel matrix for the full lattice is known, since $\mathbf{K}_{\{0, \dots, n-1\}}$ is a circulant matrix, cf. [20, Table 1], with a computational cost of

$$\mathcal{O}(n \log n) \text{ flops and } \mathcal{O}(n) \text{ memory requirements.}$$

The algorithm for the full lattice can be used for the subsampled kernel matrix as well. We have

$$\mathbf{K}_J = \mathbf{P} \mathbf{K}_{\{0, \dots, n-1\}} \mathbf{P}^*,$$

with $\mathbf{P}$ as in (6.1). The cost for evaluating the kernel will depend on the structure of the weights $\gamma_{\mathbf{u}}$, ranging from linear in d for product weights to quadratic in d for POD and SPOD weights, see [20, Table 1].

In contrast to the least squares matrix $\mathbf{L}_{J,\mathcal{B}}$, the condition number of the kernel matrix $\mathbf{K}_J$ grows in the number of points, resulting in the need for many iterations or the necessity of preconditioning, cf. [43, Section 12.2]. Thus, the number of necessary CG iterations cannot be bounded by a general constant according to Theorem 6.1.

With $|J| \sim \sqrt{n} \log n$ as in Theorem 5.3, we obtain the computational cost in the lower two rows of Table 1 when using the FFT compared to the naive matrix-vector product when using r iterations.

6.3 Comparison to full-lattice methods

From Theorem 5.3 we know that subsampling allows for a better sampling complexity. However, picking the best method in regards to computational cost also depends on whether functions evaluations are expensive. To discuss this balancing, we split the computations into five steps and address their individual computational cost. This gives the following for the least squares approximation and the kernel method with the full and subsampled lattice, which is summarized in Table 2.

- (i) *Point construction.* All methods use the CBC construction from [14] with different cost depending on the type of weights as shown in the first row of Table 2. The additional cost for creating the subsampled indices is negligible.
- (ii) *The sampling of the function to approximate.* When f_{cost} models the cost of a single function evaluation, this has to be multiplied by the number of samples used by the approximation. For the full lattice these are n points and for the subsampled lattice we have $|J| \sim \sqrt{n} \log n$ points, cf. Theorem 5.3.
- (iii) *The function-independent setup.* These computations may depend on a pre-defined function class but can be used for any function thereof. This becomes important when multiple instances need to be approximated.

For the least squares approximation this involves the construction of the frequency index set $\mathcal{B}$ and the computation of the inner products with the generating vector $\mathbf{z}$. As discussed in Section 6.1 this can be done incrementally in dimension with a cost of $d|\mathcal{B}|$, where $|\mathcal{B}| \sim \sqrt{n}$ in our case.

For the kernel method the entries of the kernel matrix $\mathbf{K}_J$ have to be computed, where the cost of one single kernel evaluation depends on the weight parameters, cf. [20, Table 1]. Because we have $K(\frac{k}{n}\mathbf{z} \bmod 1, \frac{k'}{n}\mathbf{z} \bmod 1) = K(\frac{k-k'}{n}\mathbf{z} \bmod 1, \mathbf{0})$, the number of different values equals $|\{(k - k') \bmod n : k, k' \in J\}|$. For the full lattice this evaluates to n . For the randomly subsampled lattice with $|J| \sim \sqrt{n} \log n$ we might have $|\{(k - k') \bmod n : k, k' \in J\}| = n$ as well.

- (iv) *The computation of the coefficients.* At this step the approximation is obtained from the sampled function values using the function-independent setup and encoded in a respective representation. The least squares approximation is described by its Fourier coefficients $[\hat{g}_{\mathbf{h}}]_{\mathbf{h} \in \mathcal{B}}$ and the kernel method by the coefficients $[a_k]_{k \in J}$ (where $J = \{0, \dots, n-1\}$ for the full lattice). These coefficients are defined by the solution

Computation step	$S_{\mathcal{B}}^{\mathbf{X}_J}$	$S_{\mathcal{B}}^{\mathbf{X}}$	$A_{\text{ker}}^{\mathbf{X}_J}$	$A_{\text{ker}}^{\mathbf{X}}$
point construction	$d n \log n + d^i \alpha^j n$			
sampling	$f_{\text{cost}} \sqrt{n} \log n$	$f_{\text{cost}} n$	$f_{\text{cost}} \sqrt{n} \log n$	$f_{\text{cost}} n$
function-indep. setup	$d \sqrt{n}$		$d^i \alpha^j n$	
computing coefficients	$n \log n$		$r n \log n$	$n \log n$
evaluation	$d \sqrt{n}$		$d^i \alpha^j \sqrt{n} \log n$	$d^i \alpha^j n$

Table 2: Computational cost for the least squares approximation $S_{\mathcal{B}}^{\mathbf{X}_J}$ and the kernel method $A_{\text{ker}}^{\mathbf{X}_J}$ with a full and subsampled lattice of size $|J| = \lceil \sqrt{n} \log n \rceil$. Here r is the number of iterations depending on the condition number of the kernel matrix and f_{cost} is the cost of a single function evaluation. We have $(i, j) = (1, 0)$ for product weights, $(i, j) = (2, 0)$ for POD weights, and $(i, j) = (2, 2)$ for SPOD weights.

of the systems of equations in Lemma 4.1 and (2.7), respectively. For the full lattice the involved inverse matrices are known in closed form as discussed in Section 6.1 and Section 2, which results in a computational cost of an FFT for the computation of the coefficients.

For the subsampled methods we use the iterative schemes described in Section 6.1 and Section 6.2. In case of the least squares method, the number of iterations is bounded and by a general constant. For kernel method more iterations are needed when the number of points increases, as the kernel matrix is worse conditioned for an increasing number of points. In Table 2 the number of iterations for the kernel method is denoted by r .

- (v) *The evaluation of the approximation.* This describes the cost of evaluating the approximation in a single unseen data point $\mathbf{x} \in \mathbb{T}^d$. For the least squares approximation this involves evaluating the sum $(S_{\mathcal{B}}^{\mathbf{X}_J} f)(\mathbf{x}) = \sum_{\mathbf{h} \in \mathcal{B}} \hat{g}_{\mathbf{h}} \exp(2\pi i \langle \mathbf{h}, \mathbf{x} \rangle)$ with a cost of $d |\mathcal{B}|$, with $|\mathcal{B}| \sim \sqrt{n}$ in our case.

For the kernel method we need to evaluate $\sum_{k \in J} a_k K(\mathbf{x}, \frac{k}{n} \mathbf{z} \bmod 1)$, which needs $|J|$ evaluations of the kernel with $|J| = \{0, \dots, n-1\}$ for the full lattice.

Note that for a given n all methods have the L_2 worst-case error proportional to $\sqrt[4]{S_n(\mathbf{z})}$, cf. Theorems 2.1 and 5.3.

Apart from user-defined benchmark-functions, the cost of function evaluations f_{cost} is often expensive, like when solving a PDE or taking in-field measurements, see e.g. [32]. Then the approximation serves as a surrogate model and a fast evaluation of the approximation is the priority, which favors the methods using the subsampled lattice. The overload of using an iterative solver is also reasonable, in particular for the least squares approximation, where the number of iterations can be bounded by a general constant.

7 Numerical results

In this section we test the proposed algorithms in order to verify our theoretical results by comparing

- the classical lattice algorithm (2.1) using the full lattice, with frequency index set $\mathcal{A} = \{\mathbf{h} \in \mathbb{Z}^d : r_{d,\alpha,\gamma}(\mathbf{h}) \leq M\}$ and $M = 1/\sqrt{\mathcal{S}_n(\mathbf{z})}$,
- the least squares approximation (4.1) using the full and subsampled lattice where the frequency index set $\mathcal{B} = \{\mathbf{h} \in \mathbb{Z}^d : r_{d,\alpha,\gamma}(\mathbf{h}) \leq M\}$ and $M = 1/(2\sqrt{\mathcal{S}_n(\mathbf{z})})$ is chosen as in Theorem 5.3, and
- the kernel method (2.7) with both the full and subsampled lattice.

With n being the next biggest prime from a power of 2, we compute the generating vector for a lattice using the CBC algorithm from [14] rather than the original algorithm from [6]. The new algorithm has the advantages that it is much simpler to implement, is extensible to any dimension d , and (especially for SPOD weights) is cheaper to run. Empirically it gives similar L_2 -approximation error bounds, but equivalent theoretical error bounds are yet to be proven. For the methods using subsampled points, we generate a random subset of size $|J| = \lceil \sqrt{n} \log n \rceil$, which is of the same order as in Theorem 5.3 but ignores the involved constants. The L_2 -error is then estimated using 50 random shifts of the initial lattice,

$$\int_{\mathbb{T}^d} |(f - A(f))(\mathbf{x})|^2 d\mathbf{x} \approx \frac{1}{50n} \sum_{i=1}^{50} \sum_{k=0}^{n-1} |(f - A(f))((\frac{k}{n}\mathbf{z} + \mathbf{\Delta}_i) \bmod 1)|^2,$$

where $\mathbf{\Delta}_i \in \mathbb{T}^d$ for $i = 1, \dots, 50$ are chosen independently and uniformly random. For the kernel method the evaluation at the shifted lattice can be done by the same fast methods as for the original lattice used in the approximation, cf. [20, Section 5]. The least squares approximation can be evaluated at any lattice fast using the **LFFT**.

The implementation is done in Julia with the **CG** and **LSQR** implementation used from [4]. All the computations were carried out on a high performance computer cluster [37] with eight parallel threads.

7.1 Kink function

For a first numerical experiment we choose the target “kink” function

$$f(\mathbf{x}) = \left(\frac{5^{3/4}15}{4\sqrt{3}}\right)^d \prod_{j=1}^d \max\left\{0, \frac{1}{5} - \left(x_j - \frac{1}{2}\right)^2\right\} \quad (7.1)$$

in dimensions $d = 2$ and $d = 5$. The 2-dimensional example is depicted in Figure 2 (left). This function was considered in [3, 23, 1] and we know that $f \in H_{d,3/2-\varepsilon,\gamma}(\mathbb{T}^d)$ for $\varepsilon > 0$ and $\gamma_{\mathbf{u}} = 1$ for all $\mathbf{u} \subseteq \{1, \dots, d\}$.

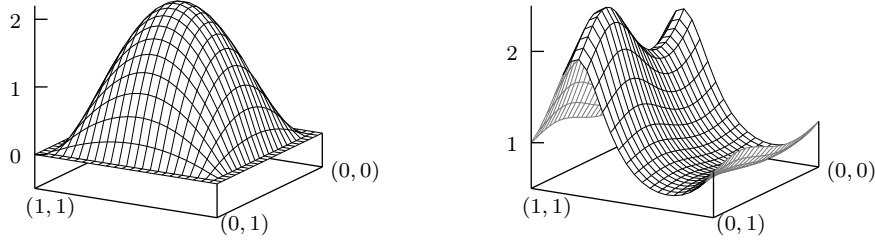


Figure 2: “Kink” function (7.1) (left) and “reciprocal” function (7.2) with $q = 2.5$ (right), both in dimension $d = 2$.

For the approximation we chose product weights with parameters

$$\alpha = 1 \quad \text{and} \quad \gamma_j = \frac{1}{2}.$$

The result for dimensions $d = 2$ and $d = 5$ are depicted in Figure 3.

We start by evaluating the 2-dimensional example, hoping to observe the theoretical convergence rate. All rates displayed in the legend were estimated by fitting the last ten data points.

- : For $(n, \sqrt[4]{\mathcal{S}_n(\mathbf{z})})$ we expect a decay rate arbitrarily close to $1/2$ according to Theorem 5.1. The numerics yield a rate of 0.47 affirming this behavior.
- ▼, ◆: The classical lattice algorithm and the least squares method using the full lattice differ only by a factor of two in the radius of their frequencies. According to Theorem 2.1 they behave at least as well as $(n, \sqrt[4]{\mathcal{S}_n(\mathbf{z})})$ with a rate of $1/2$. In fact, the same rate of 0.7 is observed in both methods and they only differ in the corresponding constant. The better than expected rate can be explained by these methods “picking up” the smoothness. Since the smoothness does not affect the shape of the frequency index set, the method for higher smoothness is the same. In our case we have smoothness $3/2 - \varepsilon$, of which half is “picked up”. That it is only half is intuitively explained by full lattices only achieving half the optimal order of convergence, cf. [3].
- ▲: For the kernel method using the full lattice we know by the discussion at the end of Section 2 that it performs at least as well as the classical lattice algorithm with a rate of $1/2$ in the worst-case setting. It was further shown in [41] that the kernel method (using any set of points) doubles the rate of convergence when the function is doubly as smooth. In our case we do not have quite double the smoothness, but still see this effect numerically with a rate of 1.01. Note, that the experiment is only about the approximation error of one individual function which could perform better than the worst-case error.

- : For the subsampled methods we expect to achieve the same worst-case error but with fewer points. With $(|J|/\log |J|, \sqrt[4]{\mathcal{S}_n(\mathbf{z})})$, we observe a rate of 0.93, which is about double of 0.47 from the full lattice $(n, \sqrt[4]{\mathcal{S}_n(\mathbf{z})})$ (●).
- ◇: According to Theorem 4.7 the least squares approximation using the subsampled lattice has at least the same rate as $(|J|/\log |J|, \sqrt[4]{\mathcal{S}_n(\mathbf{z})})$ (○), which was 0.93. For reasons explained above, the least squares method is able to “pick up” the smoothness, which is $3/2 - \varepsilon$. Numerically we observe a rate of 1.37.
- △: The kernel method performs at least as well as any other method using the same points. Since the least squares method using the subsampled lattice has at least linear decay, this holds for the kernel method using the subsampled points as well. Additionally the doubling of the rate effect from [41] applies again. Numerically we observe the best rate among all methods with 1.47.

The 5-dimensional example shows the same qualitative but not quantitative behavior, as increasing the dimension makes the problem a lot harder when all dimensions are equally important. The dimension enters logarithmically as we have for the “linear width” $a_n \sim n^{-\alpha}(\log n)^{(d-1)\alpha}$, which is a lower bound to the worst-case error using n samples, cf. [13]. One notable difference is a staircase-like error behavior for the classical lattice algorithm as well as the least squares approximation. This is due to the number of frequencies not being continuous with respect to the radius of the frequency index set. Especially for small radii a bunch of frequencies are added at once relative to the total number of frequencies, resulting in a jump.

7.2 Reciprocal function

For a second batch of numerical experiments we choose the $d = 100$ dimensional target “reciprocal” function

$$f(\mathbf{x}) = \frac{1}{a(\mathbf{x})}, \quad a(\mathbf{x}) = 1 + 0.5 \sum_{j=1}^d j^{-q} \sin(2\pi x_j), \quad (7.2)$$

with decay parameter $q = 6$ and $q = 2.5$. The function in two dimensions is depicted in Figure 2 (right). This function was considered in e.g., [28, Section 4.2] and solves the algebraic equation $a(\mathbf{x})f(\mathbf{x}) = 1$, mimicking the features of a partial differential equation with a random coefficient whilst avoiding the complexity of a spatial variable or the need of a finite element solver.

For the approximation we choose POD weights with parameters

$$\alpha \in \{1, 2\} \quad \text{and} \quad \gamma_{\mathbf{u}} = |\mathbf{u}|! \prod_{j \in \mathbf{u}} j^{-q}.$$

Following the same procedure as in Section 7.1, in Figure 4 we depict the results for $q = 6$, with $\alpha = 1$ (top) and $\alpha = 2$ (bottom). Since we deal with an analytic function, these results are not representative for the worst-case error and therefore the doubling

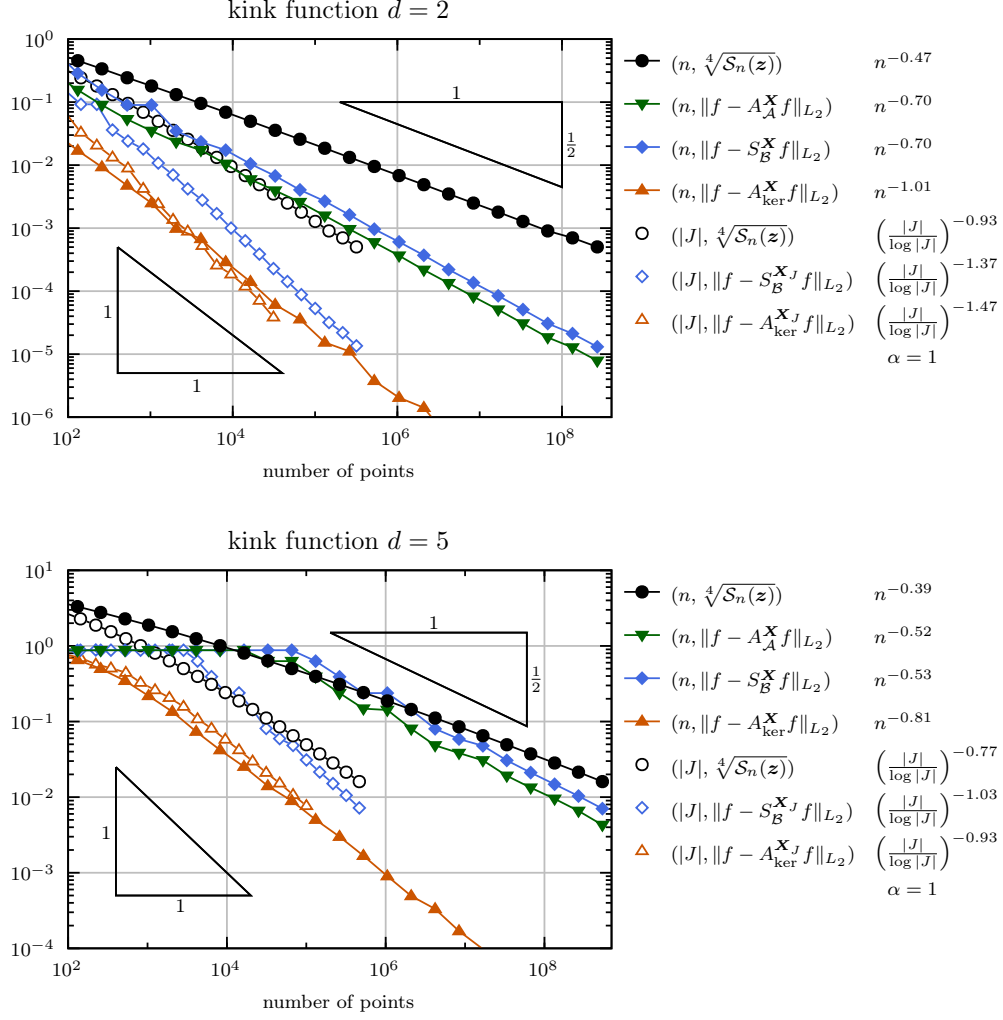


Figure 3: L_2 -error of the approximations of the kink function, with $d = 2$ (top) and $d = 5$ (bottom) with respect to the number of sampling points for the classical lattice algorithm, the least squares method, and the kernel method using the full and subsampled lattice.

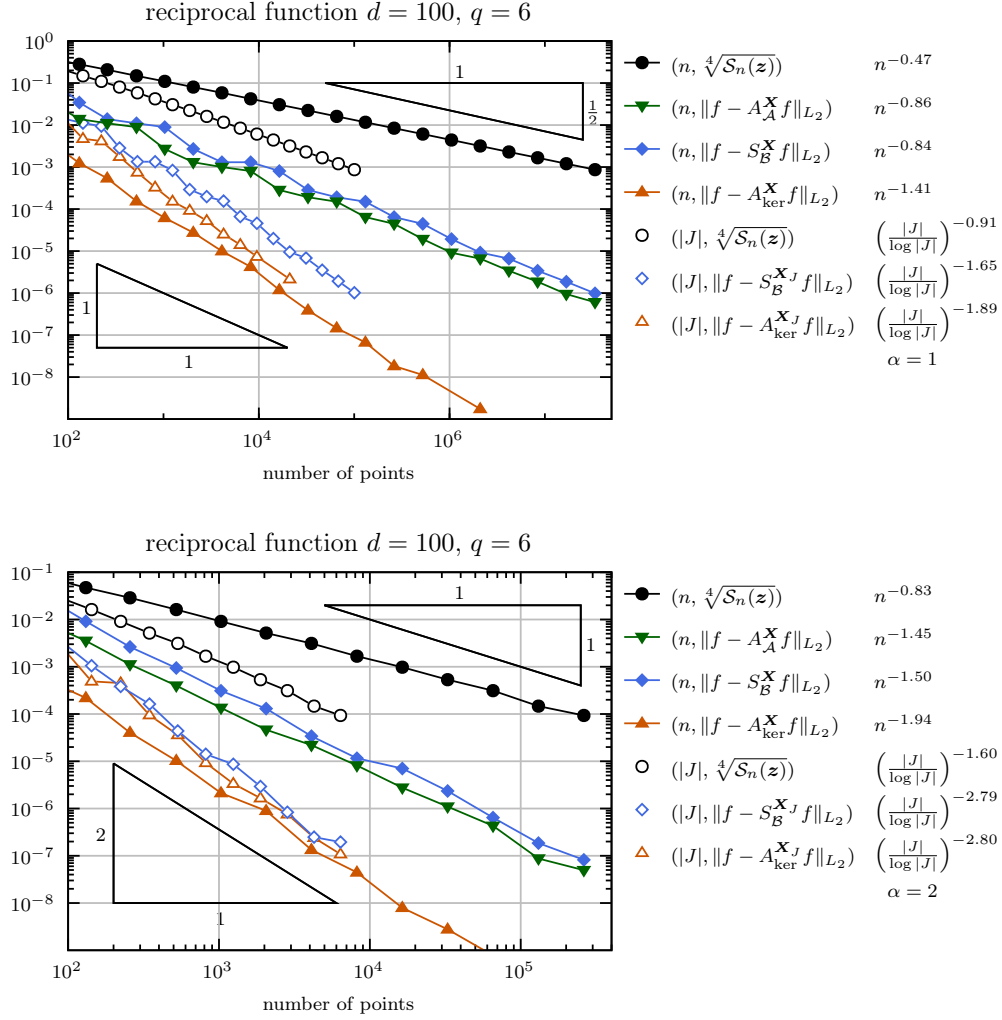


Figure 4: L_2 -error of the approximations of the reciprocal function in $d = 100$ dimensions and decay parameter $q = 6$, with smoothness parameter $\alpha = 1$ (top) and $\alpha = 2$ (bottom), with respect to the number of sampling points for the classical lattice algorithm, the least squares method, and the kernel method using the full and subsampled lattice.

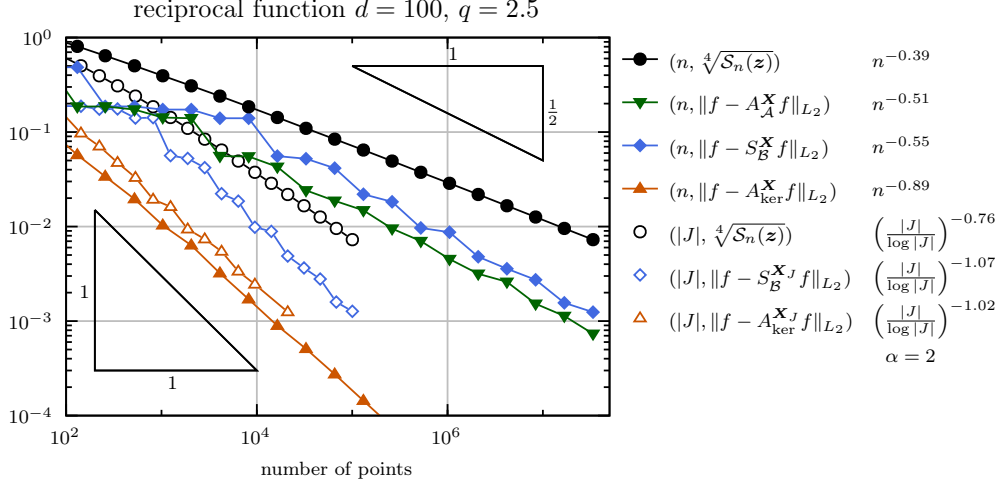


Figure 5: L_2 -error of the approximations of the reciprocal function in $d = 100$ dimensions and with decay parameter $q = 2.5$ and smoothness parameter $\alpha = 2$ with respect to the number of sampling points for the classical lattice algorithm, the least squares method, and the kernel method using the full and subsampled lattice.

of the rate [41] becomes more apparent. The high dimension $d = 100$ diminishes the rates we expected with respect to α . For $\alpha = 1$ (top) the kernel method with both the full as well as for the subsampled lattice perform a lot better than the corresponding least squares approximation, with empirical rates of 1.41 and 1.89 versus 0.84 and 1.65, respectively. For $\alpha = 2$ (bottom), kernel method and least squares approximation with subsampled lattice perform equally well, with empirical rates of 2.79 and 2.8, respectively.

In Figure 4 we plot the corresponding results with the relatively slow decay parameter $q = 2.5$ and $\alpha = 2$. This problem is considerably harder. The empirical rates drop significantly to be just above 1 for both the kernel method and least squares approximation with the subsampled lattice.

The frequencies used in the $q = 6$ example in Figure 4 by the least squares approximation were nonzero only in the first 11 and 23 components for $\alpha = 1$ and $\alpha = 2$, respectively, i.e., the least squares method treated the remaining dimensions as irrelevant, while still achieving errors of 10^{-6} and 10^{-7} . For the $q = 2.5$ case in Figure 5 frequencies with nonzero components were encountered only in the first 52 dimensions. This indicates that the “effective dimension” is low for the $q = 6$ example compared to the $q = 2.5$ example. In this case all the methods for the $q = 2.5$ example behave a lot more like $(n, \mathcal{S}_n(\mathbf{z}))$ for the full lattice and $(|J|, \mathcal{S}_n(\mathbf{z}))$ for the subsampled lattice, in comparison to the $q = 6$ example. It is also an example in which the subsampled least squares error shows a slightly faster empirical rate of 1.07 than the subsampled kernel method empirical rate 1.02.

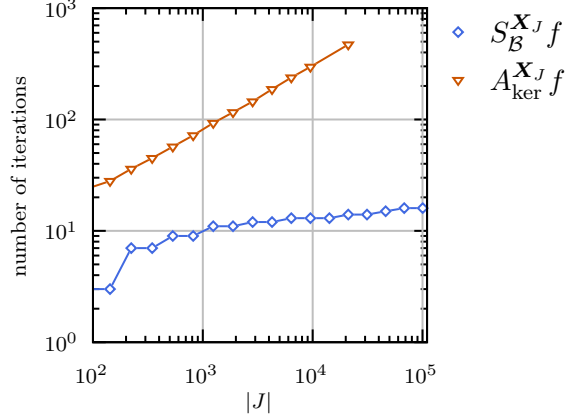


Figure 6: Number of iterations with respect to the number of points for subsampled least squares approximation and kernel method.

7.3 Computational complexity

Since we are interested in the computational complexity, we measure the computation time and counted the number of iterations for the presented methods for approximating the reciprocal function with $q = 2.5$ and $\alpha = 2$. Both the LSQR method for the least squares approximation and the CG method for the kernel method were equipped with a stopping criterion of the relative error being smaller than 10^{-8} . The iteration count of the subsampled methods is depicted in Figure 6. While the least squares approximation never exceeded 16 iterations, the kernel method used up to 471 iterations, growing like $|J|^{0.57}$. This affirms the predicted cost with respect to the number of iterations in Table 1.

For the actual cost, including precomputation and the matrix-vector products, we measure the elapsed time. For the subsampled lattice we used the naive matrix-vector implementation as well as the FFT implementation discussed in Section 6. Further, we modeled two scenarios: function evaluations to have no cost to evaluate and second, needing 0.1 seconds to evaluate. The latter resembles practical examples with function evaluations corresponding to PDE solutions or similar. The results are depicted in Figure 7.

Without considering the cost for function evaluations the kernel method with the full lattice is the fastest when the number of points is large enough. This is again due to making use of the higher smoothness of the function for a faster rate of decay. The cost for all subsampled methods differ only in multiplicative constants for the error regardless of using the FFT or naive matrix-vector products. The high overhead of the kernel method is due to the cost of evaluating the kernel and far more iterations needed for iterative solver, since the kernel matrix is not as well-conditioned as the least squares matrix. In terms of memory, we stored the subsampled matrices for the naive implementation, which required us to stop at an earlier number of points as we were limited to use 512 Gigabytes of memory. This could be overcome by computing the matrix on the fly but would add a larger computational overload.

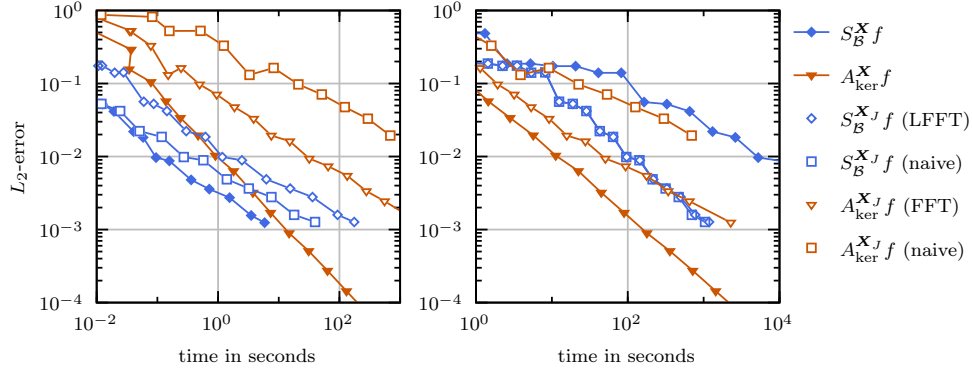


Figure 7: L_2 -error of the numerical experiment with respect to the computation time. Left: function evaluation costs are not accounted for. Right: cost of single function evaluation is 0.1 seconds.

With the addition of a small cost of 0.1 seconds for function evaluations, the graph resembles more the sampling complexity cost in Figure 5. Thus, in the right-hand setting when function evaluations are costly, the overhead of using an iterative solver with the subsampling techniques of this paper is justified. If the function in question is smooth, the kernel method excels due to the doubling of the rate effect [41].

8 Concluding remarks

In this paper we used random subsampling to regain the optimal polynomial order of convergence for lattices while still maintaining the structure to apply fast Fourier algorithms. The focus is on having a small initial lattice size, which determines the computational complexity of the approximation algorithms. We used the least squares approximation and the kernel method. For Korobov spaces we achieve the optimal polynomial order of convergence $(|J|/\log |J|)^{-\alpha+\varepsilon}$ for $0 < \varepsilon < \alpha$, with the initial lattice size $n \lesssim (|J|/\log |J|)^{2/\sqrt{1-\varepsilon/\alpha}}$, which is arbitrarily close to quadratic in terms of the actual subsampled point set size $|J|$. This improves on earlier results in [1], where there is a polynomial gap for the initial lattice size.

Similar to [1] one could use further subsampling techniques from [2] or even the non-constructive techniques from [12] in order to improve on the logarithmic rate in the error bound. Further, in the numerics the constants for the oversampling were ignored while still achieving the theoretical error rates. This indicates possible improvement for the constants.

Acknowledgement. We acknowledge the financial support from the Australian Research Council Discovery Project DP240100769. This research includes computations using the computational cluster Katana supported by Research Technology Services at UNSW Sydney.

References

- [1] F. Bartel, L. Kämmerer, D. Potts, and T. Ullrich. “On the reconstruction of functions from values at subsampled quadrature points”. In: *Mathematics of Computation* 93.346 (Aug. 2023), pp. 785–809. ISSN: 1088-6842. DOI: 10.1090/mcom/3896.
- [2] F. Bartel, M. Schäfer, and T. Ullrich. “Constructive subsampling of finite frames with applications in optimal function recovery”. In: *Applied and Computational Harmonic Analysis* 65 (July 2023), pp. 209–248. ISSN: 1063-5203. DOI: 10.1016/j.acha.2023.02.004.
- [3] G. Byrenheid, L. Kämmerer, T. Ullrich, and T. Volkmer. “Tight error bounds for rank-1 lattice sampling in spaces of hybrid mixed smoothness”. In: *Numer. Math.* 136.4 (2017), pp. 993–1034. ISSN: 0029-599X. DOI: 10.1007/s00211-016-0861-7.
- [4] J. Chen, H. Stoppels, H. Ranocha, J. López, et al. *IterativeSolvers.jl Julia package*. <https://github.com/JuliaLinearAlgebra/IterativeSolvers.jl>. 2025.
- [5] A. Cohen and G. Migliorati. “Optimal weighted least-squares methods”. In: *The SMAI journal of computational mathematics* 3 (Oct. 2017), pp. 181–203. ISSN: 2426-8399. DOI: 10.5802/smai-jcm.24.
- [6] R. Cools, F. Y. Kuo, D. Nuyens, and I. H. Sloan. “Lattice algorithms for multivariate approximation in periodic spaces with general weight parameters”. In: *75 years of mathematics of computation*. Vol. 754. Contemp. Math. Amer. Math. Soc., [Providence], RI, 2020, pp. 93–113. ISBN: 978-1-4704-5163-9. DOI: 10.1090/conm/754/15150.
- [7] J. Dick, P. Kritzer, and F. Pillichshammer. *Lattice rules—numerical integration, approximation, and discrepancy*. Vol. 58. Springer Series in Computational Mathematics. With an appendix by Adrian Ebert. Springer, Cham, 2022, pp. xvi+580. ISBN: 978-3-031-09951-9. DOI: 10.1007/978-3-031-09951-9.
- [8] J. Dick, P. Kritzer, F. Y. Kuo, and I. H. Sloan. “Lattice-Nyström method for Fredholm integral equations of the second kind with convolution type kernels”. In: *J. Complexity* 23.4-6 (2007), pp. 752–772. ISSN: 0885-064X,1090-2708. DOI: 10.1016/j.jco.2007.03.004.
- [9] J. Dick, F. Y. Kuo, Q. T. Le Gia, D. Nuyens, and C. Schwab. “Higher Order QMC Petrov–Galerkin Discretization for Affine Parametric Operator Equations with Random Field Inputs”. In: *SIAM Journal on Numerical Analysis* 52.6 (Jan. 2014), pp. 2676–2702. ISSN: 1095-7170. DOI: 10.1137/130943984.
- [10] J. Dick, F. Y. Kuo, Q. T. Le Gia, and C. Schwab. “Multilevel Higher Order QMC Petrov–Galerkin Discretization for Affine Parametric Operator Equations”. In: *SIAM Journal on Numerical Analysis* 54.4 (Jan. 2016), pp. 2541–2568. ISSN: 1095-7170. DOI: 10.1137/16m1078690.
- [11] J. Dick, I. H. Sloan, X. Wang, and H. Woźniakowski. “Good Lattice Rules in Weighted Korobov Spaces with General Weights”. In: *Numerische Mathematik* 103.1 (Feb. 2006), pp. 63–97. ISSN: 0945-3245. DOI: 10.1007/s00211-005-0674-6.

- [12] M. Dolbeault, D. Krieg, and M. Ullrich. “A sharp upper bound for sampling numbers in L_2 ”. In: *Applied and Computational Harmonic Analysis* 63 (Mar. 2023), pp. 113–134. ISSN: 1063-5203. DOI: 10.1016/j.acha.2022.12.001.
- [13] D. Dũng, V. N. Temlyakov, and T. Ullrich. *Hyperbolic cross approximation*. Advanced Courses in Mathematics. CRM Barcelona. Edited and with a foreword by Sergey Tikhonov. Birkhäuser/Springer, Cham, 2018, pp. xi+218. ISBN: 978-3-319-92240-9. DOI: 10.1007/978-3-319-92240-9.
- [14] A. D. Gilbert and I. H. Sloan. “A new CBC-algorithm”. In: *work in progress* (2025).
- [15] I. G. Graham, F. Y. Kuo, J. A. Nichols, R. Scheichl, C. Schwab, and I. H. Sloan. “Quasi-Monte Carlo finite element methods for elliptic PDEs with lognormal random coefficients”. In: *Numerische Mathematik* 131.2 (Dec. 2014), pp. 329–368. ISSN: 0945-3245. DOI: 10.1007/s00211-014-0689-y.
- [16] I. G. Graham, F. Y. Kuo, D. Nuyens, R. Scheichl, and I. H. Sloan. “Circulant embedding with QMC: analysis for elliptic PDE with lognormal coefficients”. In: *Numerische Mathematik* 140.2 (May 2018), pp. 479–511. ISSN: 0945-3245. DOI: 10.1007/s00211-018-0968-0.
- [17] A. Greenbaum. *Iterative methods for solving linear systems*. Vol. 17. Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997, pp. xiv+220. ISBN: 0-89871-396-X. DOI: 10.1137/1.9781611970937.
- [18] A. Hinrichs, D. Krieg, E. Novak, and J. Vybíral. “Lower bounds for integration and recovery in L_2 ”. In: *J. Complexity* 72 (2022), Paper No. 101662, 15. ISSN: 0885-064X, 1090-2708. DOI: 10.1016/j.jco.2022.101662.
- [19] V. Kaarnioja, F. Y. Kuo, and I. H. Sloan. “Uncertainty Quantification Using Periodic Random Variables”. In: *SIAM Journal on Numerical Analysis* 58.2 (Jan. 2020), pp. 1068–1091. ISSN: 1095-7170. DOI: 10.1137/19m1262796.
- [20] V. Kaarnioja, Y. Kazashi, F. Y. Kuo, F. Nobile, and I. H. Sloan. “Fast approximation by periodic kernel-based lattice-point interpolation with application in uncertainty quantification”. In: *Numerische Mathematik* 150.1 (Nov. 2021), pp. 33–77. ISSN: 0945-3245. DOI: 10.1007/s00211-021-01242-3.
- [21] L. Kämmerer. *High Dimensional Fast Fourier Transform Based on Rank-1 Lattice Sampling*. Dissertation. Universitätsverlag Chemnitz, 2014. ISBN: 978-3-944640-41-9. URL: <http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-157673>.
- [22] L. Kämmerer, D. Potts, and T. Volkmer. “Approximation of multivariate periodic functions by trigonometric polynomials based on rank-1 lattice sampling”. In: *J. Complexity* 31 (2015), pp. 543–576. DOI: 10.1016/j.jco.2015.02.004.
- [23] L. Kämmerer, T. Ullrich, and T. Volkmer. “Worst case recovery guarantees for least squares approximation using random samples”. In: *Constr. Approx.* 54 (2021), pp. 295–352. DOI: 10.1007/s00365-021-09555-0.

- [24] L. Kämmerer. “Reconstructing multivariate trigonometric polynomials from samples along rank-1 lattices”. In: *Approximation Theory XIV: San Antonio 2013*. Springer. 2014, pp. 255–271.
- [25] L. Kämmerer. “A fast probabilistic component-by-component construction of exactly integrating rank-1 lattices and applications”. In: *arXiv:math/2012.14263* (2020).
- [26] L. Kämmerer, S. Kunis, and D. Potts. “Interpolation lattices for hyperbolic cross trigonometric polynomials”. In: *Journal of Complexity* 28.1 (Feb. 2012), pp. 76–92. ISSN: 0885-064X. DOI: 10.1016/j.jco.2011.05.002.
- [27] L. Kämmerer, T. Ullrich, and T. Volkmer. “Worst-case Recovery Guarantees for Least Squares Approximation Using Random Samples”. In: *Constructive Approximation* 54.2 (Aug. 2021), pp. 295–352. ISSN: 1432-0940. DOI: 10.1007/s00365-021-09555-0.
- [28] A. Keller, F. Y. Kuo, D. Nuyens, and I. H. Sloan. “Regularity and Tailored Regularization of Deep Neural Networks, with application to parametric PDEs in uncertainty quantification”. In: *arXiv:math/2502.12496* (2025). DOI: 10.48550/arXiv.2502.12496.
- [29] D. Krieg and M. Ullrich. “Function Values Are Enough for L_2 -Approximation”. In: *Foundations of Computational Mathematics* 21.4 (Dec. 2020), pp. 1141–1151. ISSN: 1615-3383. DOI: 10.1007/s10208-020-09481-w.
- [30] F. Y. Kuo, W. Mo, and D. Nuyens. “Constructing Embedded Lattice-Based Algorithms for Multivariate Function Approximation with a Composite Number of Points”. In: *Constructive Approximation* 61.1 (Apr. 2024), pp. 81–113. ISSN: 1432-0940. DOI: 10.1007/s00365-024-09688-y.
- [31] F. Y. Kuo and D. Nuyens. “Application of Quasi-Monte Carlo Methods to Elliptic PDEs with Random Diffusion Coefficients: A Survey of Analysis and Implementation”. In: *Foundations of Computational Mathematics* 16.6 (Sept. 2016), pp. 1631–1696. ISSN: 1615-3383. DOI: 10.1007/s10208-016-9329-5.
- [32] F. Y. Kuo, C. Schwab, and I. H. Sloan. “Quasi-Monte Carlo Finite Element Methods for a Class of Elliptic Partial Differential Equations with Random Coefficients”. In: *SIAM Journal on Numerical Analysis* 50.6 (Jan. 2012), pp. 3351–3374. ISSN: 1095-7170. DOI: 10.1137/110845537.
- [33] F. Y. Kuo, I. H. Sloan, and H. Woźniakowski. “Lattice Rules for Multivariate Approximation in the Worst Case Setting”. In: *Monte Carlo and Quasi-Monte Carlo Methods 2004*. Springer-Verlag, pp. 289–330. ISBN: 3540255419. DOI: 10.1007/3-540-31186-6_18.
- [34] M. Moeller and T. Ullrich. “ L_2 -norm sampling discretization and recovery of functions from RKHS with finite trace”. In: *Sampl. Theory Signal Process. Data Anal.* 19.2 (2021), p. 13. ISSN: 2730-5716. DOI: 10.1007/s43670-021-00013-3.
- [35] C. C. Paige and M. A. Saunders. “Algorithm 583: LSQR: Sparse Linear Equations and Least Squares Problems”. In: *ACM Transactions on Mathematical Software* 8.2 (June 1982), pp. 195–209. ISSN: 1557-7295. DOI: 10.1145/355993.356000.

- [36] G. Plonka, D. Potts, G. Steidl, and M. Tasche. *Numerical Fourier Analysis*. Applied and Numerical Harmonic Analysis. Birkhäuser, 2018. ISBN: 978-3-030-04305-6. DOI: 10.1007/978-3-030-04306-3.
- [37] PVC (Research Infrastructure), UNSW Sydney. *Katana*. en. 2010. DOI: 10.26190/669X-A286.
- [38] M. Sbornik. “Approximation of functions in the space L_p and C on the torus”. In: *Mat. Sb.* 58 (1962), pp. 397–414.
- [39] I. H. Sloan and S. Joe. *Lattice methods for multiple integration*. Oxford Science Publications. The Clarendon Press, Oxford University Press, New York, 1994, pp. xii+239. ISBN: 0-19-853472-8.
- [40] I. H. Sloan and H. Woźniakowski. “When Are Quasi-Monte Carlo Algorithms Efficient for High Dimensional Integrals?” In: *Journal of Complexity* 14.1 (Mar. 1998), pp. 1–33. ISSN: 0885-064X. DOI: 10.1006/jcom.1997.0463.
- [41] I. H. Sloan and V. Kaarnioja. “Doubling the rate: improved error bounds for orthogonal projection with application to interpolation”. In: *BIT Numerical Mathematics* 65.1 (Jan. 2025). ISSN: 1572-9125. DOI: 10.1007/s10543-024-01049-2.
- [42] J. A. Tropp. “User-Friendly Tail Bounds for Sums of Random Matrices”. In: *Foundations of Computational Mathematics* 12.4 (Aug. 2011), pp. 389–434. ISSN: 1615-3383. DOI: 10.1007/s10208-011-9099-z.
- [43] H. Wendland. *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2004.
- [44] H. Woźniakowski. “Tractability of Multivariate Integration for Weighted Korobov Spaces: My 15 Year Partnership with Ian Sloan”. In: *Monte Carlo and Quasi-Monte Carlo Methods 2008*. Springer Berlin Heidelberg, 2009, pp. 637–653. ISBN: 9783642041075. DOI: 10.1007/978-3-642-04107-5_42.