Lattice Methods for Numerical Integration

Alexander Temper

July 14, 2021

Abstract

Our main focus will be to review investigations of lattice methods of approximating the definite integral

$$I(f) := \int_U f(\mathbf{x})\, d\mathbf{x} \tag{1}$$

of **smooth, periodic functions** over the **unit-cube** $U := [0,1)^d$ as presented in a paper by Sloan and Kachoyan [4] and get the reader familiar with the motivations and basic terminology of numerical integration.

Contents

1 Integration: analytical and numeric

The first questions the reader might bear in mind could be '*What is numerical integration?*' and '*Why should I care about numerical integration?*'. The following section aims to give a concise answer.

1.1 Analytical integration

Quite often, the general concept of integration is strongly associated with the rather specific idea of finding the area under a (smooth) curve, by using the **fundamental theorem of calculus**. When contemplating it a little, we might notice that it is actually a somewhat unintuitive but astonishing result: we compute this area without computing the 'height' of *any* point within the defined boundaries of the integrals. Even more impressive is the fact that this method is exact, i.e., its **error** is 0. When integrating analytically, the information is extracted from the global property of the **antiderivative**.

The reader likely knows how to compute simple integrals analytically, such as

$$\int_0^\pi \cos(x)\, dx, \tag{2}$$

however, to find an antiderivative can be rather difficult. As an exercise to the reader, consider the integral

$$\int_0^1 e^{-x^2}\, dx. \tag{3}$$

It turns out that the antiderivative of many functions is burdensome or infeasible to find. However, many integrals can still be approximated fairly well; this is the subject of numerical integration.

1.2 Numerical integration

Numerical integration aims to approximate a definite integral, by making use of one of a vast variety of algorithms, most often by averaging some function values between the boundaries. Often, we call numerical integration methods which act on univariate functions *quadratures*, and on multivariate functions *cubatures*, since, geometrically speaking, the integral can be interpreted as the sum of infinitesimal areas of squares, or respectively volumes of cubes, under the target function.

1.2.1 The general procedure

Generally, numerical integration methods follow the form

$$\phi(\mathbf{f}(x_1), \dots, \mathbf{f}(x_N))$$

where ϕ is some function which performs on some sampled points $x_1, ...x_N$ of the domain in the function. This can be a rather complicated function, or simple as in taking the median or the arithmetic mean of their respective function values. In most cases, however, it is enough to consider quadrature rules of the form

$$\frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i) w_i,$$

where the *weights* $w_1, ...w_N$, represent how much 'importance' we attribute to each point. In the case that all weights are equal to one, then the approximated integral would simply be the arithmetic mean of the function at the sampled points.

Next, we will consider certain 'metrics', which we can use to evaluate an algorithm.

1.2.2 Errors

If we want to investigate certain numerical algorithms, we might want to consider some metric with which we want to quantify its quality. The most natural is an algorithms potential **error**, which is the difference of the algorithms approximated value and the true integral. In most instances, the error is a defined value; however, if we desire to make more general statements, we often reason about **error-bounds**, which hold in multiple cases.

Generally, error bounds are fundamentally what can make numerical analysis rigorous. We might not be able to find precise values/optima/function-approximations, but sometimes we can **prove** that an algorithm, given certain conditions, can find solutions of certain qualities.

1.2.3 Complexity

Studying those algorithms usually also means reasoning about their *Complexity*, i.e.,

- their run-time and
- their memory-requirements,

where we can, for the first two, differentiate between the **average**-, **best**- and **worst**-case. For mathematical arguments, the worst-case is the most interesting, as it enables us to make certain statements which hold definitely (under certain assumptions). Nevertheless, we might also be interested in the other two cases, especially for applications.

1.2.4 Reasons for integrating numerically

There are, besides others, a few specific reasons why we study cubatures.

- Most notably, the difficulty of finding closed form solutions for the antiderivative of some/many functions; certain functions, such as discontinuous ones, do not even have an antiderivative.
- We might not know the function explicitly, only some of its values. In machine learning contexts, we can think of the **empirical risk** [3] as a numerical approximation of the **generalization error,** both of which can be viewed as integrals.
- Simple automated programs might be better off integrating numerically; many of the (simpler) algorithms are straight-forward to implement, and often we do not need high precision.

1.2.5 Numerical integration methods besides lattice-methods

Methods to integrate numerically have been known for a long time now, hence there is also a vast variety of them. Generally, we can differentiate between **deterministic** and **random** methods, where the former often scale poorly to higher dimensions, and the latter mostly provide not deterministic but **probabilistic** error bounds. See Weinzierl [5, p. 4-12] for an exposition.

Monte Carlo methods A simple yet powerful technique to find an integral is to (pseudo-)randomly pick points within the boundaries of the integral. As mentioned in [5, p. 9], such methods can scale rather well with dimensionality. Their performance can often be improved by tweaking the randomness, by, for example, using non-uniform distributions or intentionally reducing the variance of the random points; such methods are called *Quasi-Monte Carlo methods*.

Interpolation methods Interpolation methods aim to first **approximate the function** f by some other, easier to integrate function $\tilde{f}$, and then solve

$$\int_U \tilde{f}(\mathbf{x})\, dx \tag{4}$$

analytically. As an example, an algorithm, in English called '*Simpson's rule*' and in German called '*Keplersche Fassregel*', approximates the f by a quadratic interpolant [5, p. 5], of which the antiderivative is easy to find.

Model-based methods Some cubatures make use of parametrization, such as weights, to approximate the integral. Most notable is the family of methods called *Gaussian quadratures*, which broadly speaking assume that

$$\int_a^b w(x) f(x)\, dx = \sum_{j=1}^{n} w_j f(x_j) + \text{some Error}, \tag{5}$$

where w is some weighting function which we pick based on what we believe might constitute to being 'important points'.

There are many more of such integration methods, some of which were the central topic of papers presented at prestigious ML-conferences; see Gunter et al. [1]. Nevertheless, this seminar work will focus on the basics of lattice integration.

2 Lattices

Lattices (dt. *Gitter*) are mathematical structures and find use in many other domains, such as, such as cryptography, general mathematics, (quantum) physics, chemistry and others. We define as follows:

Definition 1. *A **lattice** is an infinite set $\mathcal{L}$ of points in $\mathbb{R}^d$ with the following properties:*

1. *If two points, $\mathbf{x}, \mathbf{y} \in \mathcal{L}$, then so is $\mathbf{x} \pm \mathbf{y}$.*
2. *$\mathcal{L}$ contains (exactly) d linearly independent points.*
3. *A lattice is not dense.*

Lemma 1. *One can generate a lattice by picking d-dimensional independent points $\mathbf{v}_1, ..., \mathbf{v}_d$ and produce all their integer linear combinations, i.e.,*

$$\mathcal{L} = \left\{ \sum_{i=1}^{d} a_i \mathbf{v}_i \,|\, a_i \in \mathbb{Z} \right\}. \tag{6}$$

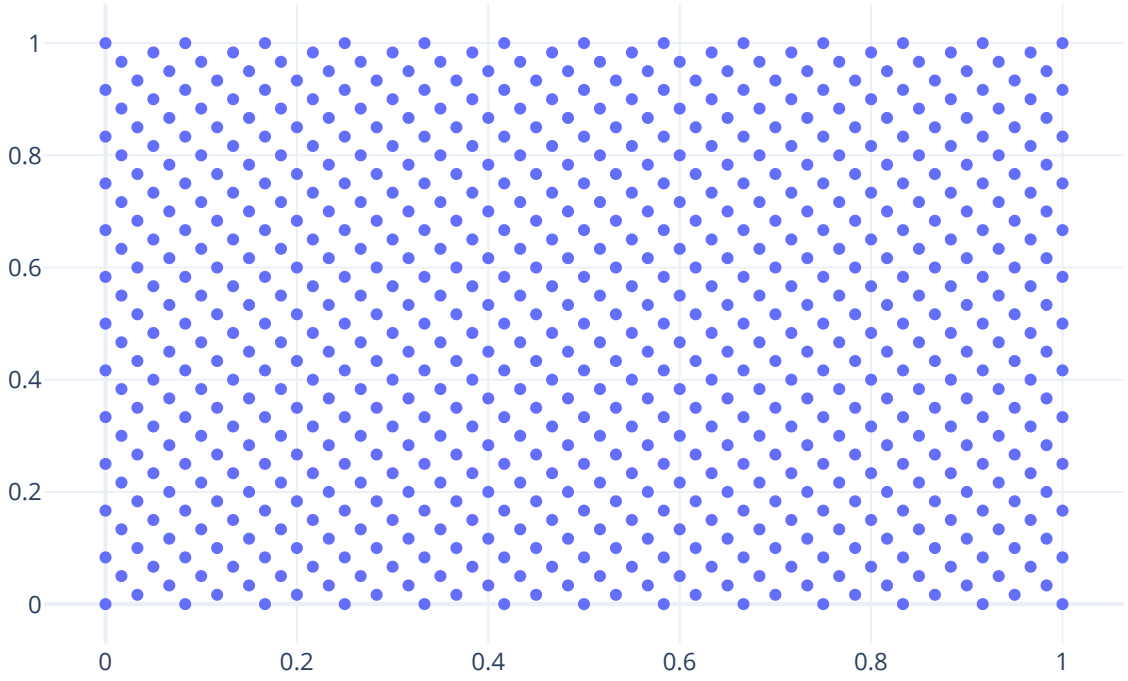

Figure 1: A lattice generated by the points $(0.2, 0.1)^T \cdot \frac{1}{6}$ and $(0.1, -0.2)^T \cdot \frac{1}{6}$. Plotted with [2].

Proof. For Property 1, assume $\mathbf{x}, \mathbf{y} \in \mathcal{L}$, generated by a_i and b_i respectively. To get $\mathbf{x} \pm \mathbf{y}$, we only need to subtract or add the corresponding a_i from the b_i. For Property 2, we have that all points are linear integer combinations of the $\mathbf{v}_i$. □

Definition 2. *Let $\mathcal{L}$ be a lattice. If the columns $\mathbf{v}_i$ of a matrix $\mathbf{M}$ can generate $\mathcal{L}$ by means of Lemma 1, then we call $\mathbf{M}$ a **generating matrix** of $\mathcal{L}$.*

Definition 3. *An **integration lattice** is a lattice which contains as a sub-lattice $\mathbb{Z}^d$, i.e.*

$$\mathcal{L} \text{ is an integration lattice} \iff \mathbb{Z}^d \subset \mathcal{L}. \tag{7}$$

Next, we shall discuss another important concept, that being the *dual lattices*.

2.1 The dual of a lattice

A key concept in the study of lattices is that of the *dual lattice*. Generally, *duality* in mathematics usually refers to some kind of 1:1 relation between two structures.

Definition 4. *Let $\mathcal{L}$ be a lattice in dimension d. Its corresponding* dual $\mathcal{L}^\perp$ *is another lattice with the property that any point $\mathbf{m} \in \mathcal{L}^\perp$ and any point $\mathbf{x} \in \mathcal{L}$ have an **integer inner-product**. More formally, this means*

$$\mathcal{L}^\perp := \left\{ \mathbf{m} \in \mathbb{R}^d \,|\, \mathbf{m} \cdot \mathbf{x} \in \mathbb{Z},\, \forall \mathbf{x} \in \mathcal{L} \right\}. \tag{8}$$

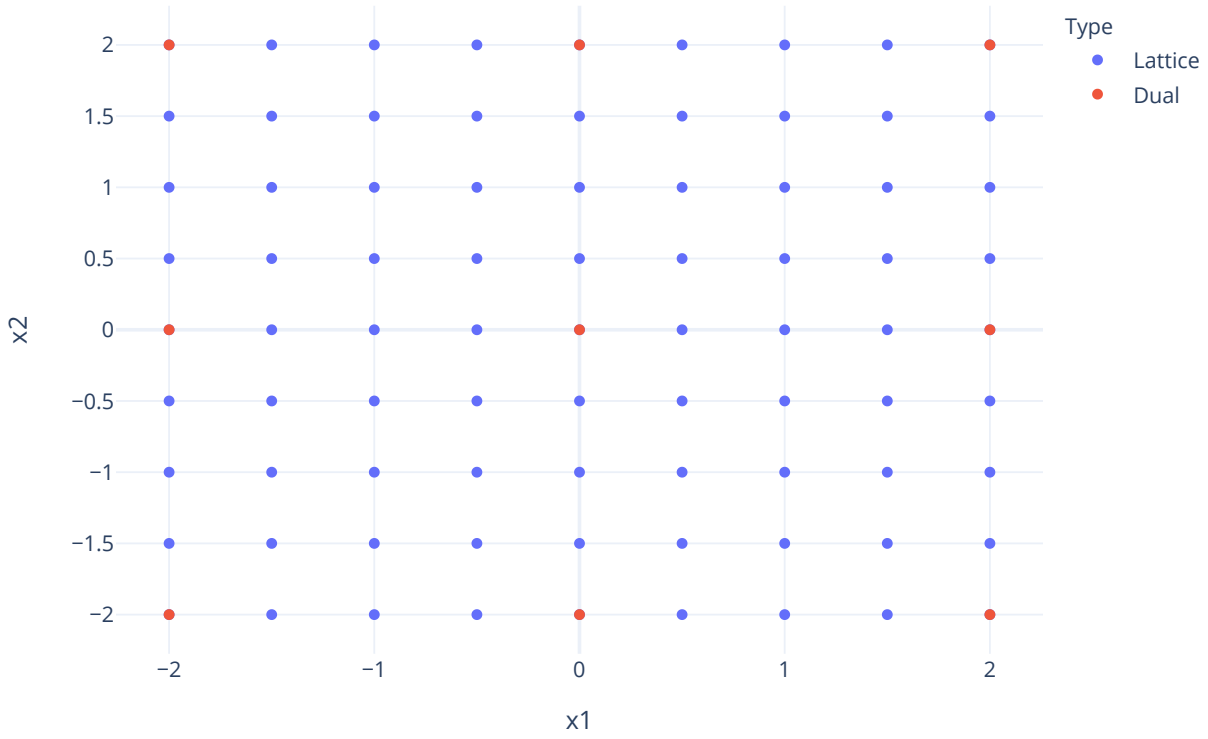

Figure 2: A rectangular lattice and its dual.

Lemma 2. *Let $\mathcal{L}$ be a lattice. We have that*

1. $(\mathcal{L}^\perp)^\perp = \mathcal{L}$
2. $\mathcal{L}$ *is an integration lattice* $\implies \mathcal{L}^\perp \subset \mathbb{Z}^d$.

Proof. The first statement obviously holds, the second one holds because integration lattices contain $\mathbb{Z}^d$, and thus all unit vectors $\mathbf{e}_i$. Since for any $\mathbf{m} \in \mathcal{L}^\perp, \mathbf{e}_i \cdot \mathbf{m} \in \mathbb{Z}$, all of $\mathbf{m}$'s components must be integers. □

Lemma 3. *Assume $\mathcal{L} \subset \mathbb{R}^d$ is generated by a matrix $\mathbf{M}$. A matrix which can generate its corresponding dual $\mathcal{L}^\perp$ is $(\mathbf{M}^{-1})^T$.*

Proof. Assume that $\mathbf{m} \in \mathcal{L}^\perp$. Further, assume that $\mathbf{M}$ and has dimension $d \times d$ and consists of the columns $\mathbf{a}_i$, where $1 \leq i \leq d$. Further, let $\mathbf{M}^{-1}$ consisting of the rows $\mathbf{b}_j$ be the inverse of $\mathbf{M}$, where $1 \leq j \leq d$. Since $\mathbf{M}\mathbf{M}^{-1} = \mathbf{I}$, we know that

$$\mathbf{b}_j \cdot \mathbf{a}_i = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

which shows us that all the $\mathbf{b}_j \in \mathcal{L}^\perp$. Further, note that for any $0 \leq i \leq d$,

$$\mathbf{a}_i \cdot \mathbf{m} \in \mathbb{Z}, \tag{10}$$

by the definition of a dual lattice. This we will use in the following:

$$\begin{aligned} \mathbf{m} &= (\mathbf{M}\mathbf{M}^{-1})\mathbf{m} && (11)\\ &= (\mathbf{M}\mathbf{M}^{-1})^T\mathbf{m} && (12)\\ &= (\mathbf{M}^{-1})^T\mathbf{M}^T\mathbf{m} && (13)\\ &= \sum_i^d \mathbf{b}_i(\mathbf{a}_i \cdot \mathbf{m}), && (14) \end{aligned}$$

which is an integer combination of the vectors $\mathbf{b}_j$. Hence, we get that $(\mathbf{M}^{-1})^T$ is a generating matrix for $\mathcal{L}^\perp$. □

2.2 Lattice rules

Definition 5. *Let $\mathcal{L}$ be an integration lattice and let*

$$\mathcal{L}_U := \mathcal{L} \cap U \tag{15}$$

be the set of points which lie in both the d-dimensional unit-cube U and $\mathcal{L}$. A ***lattice rule*** *is defined as*

$$I_{\mathcal{L}}(f) = \frac{1}{N} \sum_{\mathbf{x} \in \mathcal{L}_U} f(\mathbf{x}), \tag{16}$$

where $N := |\mathcal{L}_U|$.

As we will see later, the error of a lattice rule can be reduced by making the lattice more granular, which can be done by dividing the generator matrix $\mathbf{M}$ by some natural number $n \in \mathbb{N}$. However, in higher dimensions this rapidly increases $|\mathcal{L}_U|$: this can be attributed to the *curse of dimensionality*. Moreover, a 'good' generator for given N is not necessarily good for other N.

Theorem 1. *The N points in the unit-cube U of a lattice $\mathcal{L} \in \mathbb{R}^d$ can be generated by an integer vector $\mathbf{p} \in \mathbb{Z}^d$, which has some component p which is relatively prime to N, by the rule*

$$\mathcal{L}_U = \left\{ \left\{ j \frac{\mathbf{p}}{N} \right\} \mid j \in \{1, \ldots, N\} \right\}, \tag{17}$$

where $\{x\}$ denotes the fractional part of x. For example, $\{5.2\} = 0.2$.

Not only does this method give control over the number of sampling points, but also has fewer parameters than generating a lattice with a generating matrix. We omit the proof.

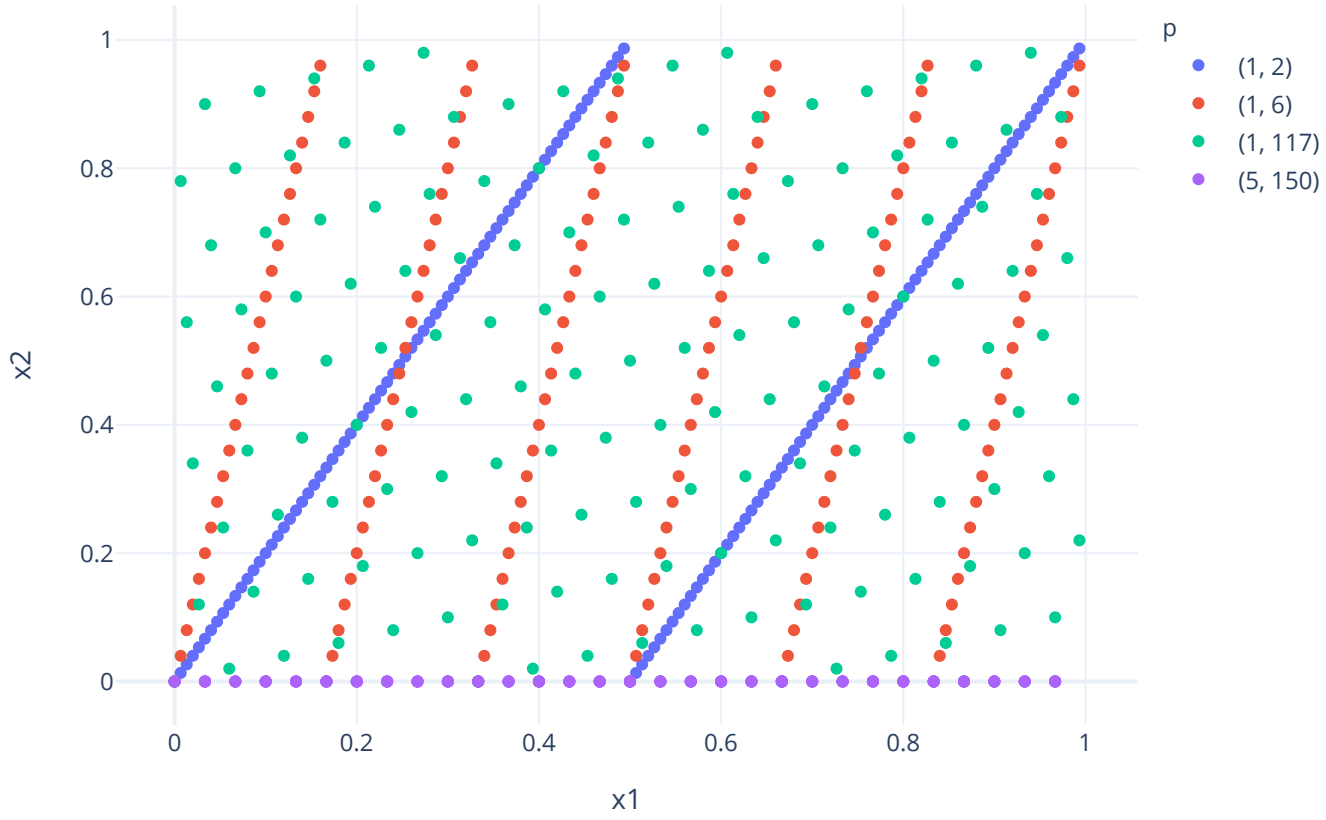

Figure 3: Lattices with $N = 150$ generated by different $\mathbf{p}$. Notice how for the red and the green lattices, the p_2 corresponds to the number of 'diagonal columns'. Further, as can be seen with the purple lattice, if none of $\mathbf{p}$'s components are relatively prime to $\mathbf{N}$, then the resulting points are not unique.

3 Analysis and error bounds

Now that we have gotten familiar with some important terms, let us continue with an error bound for lattice integration. At the heart of the proofs presented lies an analysis of the dual lattice $\mathcal{L}^\perp$.

3.1 An exponential sum

Theorem 2. *Let $\mathcal{L}$ be an integration lattice, and let $\mathbf{x}_1, ..., \mathbf{x}_N$ be the points which are both in the $\mathcal{L}$ and in the unit-cube $[0,1)^d$. Then, for any integer vector $\mathbf{m} \in \mathbb{Z}$, it holds that*

$$\frac{1}{N}\sum_{i=1}^{N} \exp(2\pi i \mathbf{m}\cdot\mathbf{x}_i) = \begin{cases} 1 & \text{if } \mathbf{m} \in \mathcal{L}^\perp, \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Proof. If we assume that $\mathbf{m} \in \mathcal{L}^\perp$, then we get that

$$\frac{1}{N}\sum_{i=1}^{N} \exp(2\pi i \mathbf{m}\cdot\mathbf{x}_i) = \frac{1}{N}\sum_{i=1}^{N} 1 = 1, \tag{19}$$

using the identity $\exp(2\pi i k) = 1$, for arbitrary $k \in \mathbb{Z}$. The more involved part of the proof deals with the case $\mathbf{m} \notin \mathcal{L}^\perp$. For this, we define a new operator T_i called 'Translation' for each point $\mathbf{x}_i \in \mathcal{L}$ which can be applied to periodic functions p such that

$$T_j p(\mathbf{x}) = p(\mathbf{x} + \mathbf{x}_j) \tag{20}$$

Using that

$$\mathbf{x}_j + \mathbf{x}_i := \mathbf{x}_l \in \mathcal{L} \tag{21}$$

we see that

$$T_j T_i p(\mathbf{x}) = p(\mathbf{x} + \mathbf{x}_j + \mathbf{x}_i) = p(\mathbf{x} + \mathbf{x}_l) = T_l p(\mathbf{x}) \tag{22}$$

which shows that there is a 1:1 correspondence between the set of the T_i and the set of the points $\mathbf{x}_i$. Note that if we 'cross the boundary' with a translation, we just translate to the other side of the period-window.

Now, let us consider an average translation function defined by

$$\bar{p} = \frac{1}{N}\sum_{j=1}^{N} T_j p. \tag{23}$$

We find that this average translation function does not change when being subjugated to any translation operator T_k, since

$$T_k \bar{p} = \frac{1}{N}\sum_{j=1}^{N} (T_j T_k) p = \frac{1}{N}\sum_{j=1}^{N} T_l p = \frac{1}{N}\sum_{l=1}^{N} T_l p. = T_l \bar{p}. \tag{24}$$

More formally, we say that $\bar{p}$ is *invariant to Translation.* Now we shall put those observations to use as follows: We define a new function,

$$\phi_{\mathbf{m}}(\mathbf{x}) = \exp(2\pi i \mathbf{m}\cdot\mathbf{x}), \qquad \mathbf{m} \in \mathbb{Z}^d, \mathbf{x} \in \mathbb{R}^d. \tag{25}$$

It is periodic, so we can translate it as defined above by the points in $\mathcal{L}$. We further have

$$T_j \phi_{\mathbf{m}}(x) = \exp(2\pi i \mathbf{m}\cdot(\mathbf{x} + \mathbf{x}_j)) \tag{26}$$

$$= \exp(2\pi i \mathbf{m}\cdot\mathbf{x}_j)\phi_{\mathbf{m}}(\mathbf{x}). \tag{27}$$

Here, recall that we assume $\mathbf{m} \notin \mathcal{L}^\perp$. Hence, there are j, i for which

$$T_j \phi_{\mathbf{m}} \neq T_i \phi_{\mathbf{m}}, \tag{28}$$

which means that if $\mathbf{m} \notin \mathcal{L}^\perp$, then $\phi_{\mathbf{m}}$ is **not** invariant. Now finally, let us recall our notion of the average translation to finish the proof:

$$\bar{\phi}_{\mathbf{m}} = \frac{1}{N}\sum_{i=1}^{N} T_i \phi_{\mathbf{m}} = \left(\frac{1}{N}\sum_{i=1}^{N} \exp(2\pi i \mathbf{m}\cdot\mathbf{x}_i)\right)\phi_{\mathbf{m}}; \tag{29}$$

Since $\phi_{\mathbf{m}}$ is not invariant, yet $\bar{\phi}_{\mathbf{m}}$ is, the term in the parentheses on the right must be zero. This concludes the proof. □

3.2 Error analysis

We now want to investigate the error of a given lattice rule; let us start with the error of an individual function.

Definition 6. *Let $\mathcal{L}$ be an integration lattice and f be a function, and let $I_{\mathcal{L}}(f)$ be the lattice integral as defined in Definition 5 and let $I(f)$ be the true integral of f over the unit cube U as defined in equation (1). We define the* ***integration error*** *of $\mathcal{L}$ as*

$$\mathcal{E}_{\mathcal{L}}(f) := I_{\mathcal{L}}(f) - I(f). \tag{30}$$

Theorem 3. *Let $\mathcal{L}$ be an integration lattice and $f : \mathbb{R}^d \to \mathbb{R}$ be a function. If we assume that f has an absolutely convergent Fourier series representation*

$$f(\mathbf{x}) = \sum_{\mathbf{m}\in\mathbb{Z}^d} \hat{f}(\mathbf{m}) \exp(2\pi i \mathbf{m}\cdot\mathbf{x}) \tag{31}$$

where

$$\hat{f}(\mathbf{m}) = \int_U \exp(-2\pi i \mathbf{m}\cdot\mathbf{x}) f(\mathbf{x})\, d\mathbf{x} \tag{32}$$

is the Fourier coefficient corresponding to the integer vector $\mathbf{m}$, then

$$\mathcal{E}_{\mathcal{L}}(f) = \sum_{\mathbf{m}\in\mathcal{L}^\perp,\, \mathbf{m}\neq\mathbf{0}} \hat{f}(\mathbf{m}). \tag{33}$$

Proof. First, observe that

$$\begin{aligned}
&I_{\mathcal{L}}(f) && && (34)\\
&= \frac{1}{N}\sum_{\mathbf{x}\in\mathcal{L}_U} f(\mathbf{x}) && \text{(Expand definition.)} && (35)\\
&= \frac{1}{N}\sum_{\mathbf{x}\in\mathcal{L}_U}\sum_{\mathbf{m}\in\mathbb{Z}^d} \hat{f}(\mathbf{m})\exp(2\pi i\mathbf{m}\cdot\mathbf{x}) && \text{(Write as Fourier series.)} && (36)\\
&= \frac{1}{N}\sum_{\mathbf{m}\in\mathbb{Z}^d}\sum_{\mathbf{x}\in\mathcal{L}_U} \hat{f}(\mathbf{m})\exp(2\pi i\mathbf{m}\cdot\mathbf{x}) && \text{(Change order of summation.)} && (37)\\
&= \frac{1}{N}\sum_{\mathbf{m}\in\mathbb{Z}^d} \hat{f}(\mathbf{m}) \sum_{\mathbf{x}\in\mathcal{L}_U} \exp(2\pi i\mathbf{m}\cdot\mathbf{x}) && \text{(Extract Fourier coefficient.)} && (38)\\
&= \sum_{\mathbf{m}\in\mathbb{Z}^d} \hat{f}(\mathbf{m}) \left(\frac{1}{N}\sum_{\mathbf{x}\in\mathcal{L}_U} \exp(2\pi i\mathbf{m}\cdot\mathbf{x})\right) && \text{(Move fraction inside.)} && (39)\\
&= \sum_{\mathbf{m}\in\mathcal{L}^\perp} \hat{f}(\mathbf{m}). && \text{(Use Theorem 2 and Lemma 2.)} && (40)\\
& && && (41)
\end{aligned}$$

Now given that

$$\hat{f}(\mathbf{0}) = \int_U \exp(-2\pi i \mathbf{0} \cdot \mathbf{x}) f(\mathbf{x})\, d\mathbf{x} = \int_U f(\mathbf{x})\, d\mathbf{x} = I(f), \tag{42}$$

we find that

$$\mathcal{E}_{\mathcal{L}}(x) = \left(\sum_{\mathbf{m} \in \mathcal{L}^\perp} \hat{f}(\mathbf{m}) \right) - \hat{f}(\mathbf{0}) = \sum_{\mathbf{m} \in \mathcal{L}^\perp,\, \mathbf{m} \neq \mathbf{0}} \hat{f}(\mathbf{m}). \tag{43}$$

□

3.2.1 Korobov classes

It is remarkable that the error of a lattice can be computed *exactly* for a given function, if we know its Fourier coefficients. However, we usually do not know those values and, moreover, do not consider one specific function, but would like to guarantee small-errors on whole function classes. Hence, let us discuss one (historically) important example.

Definition 7. *Let f be a function with an absolutely convergent Fourier series representation as used in Theorem 3. Further, let us define a class of functions $E_d^\alpha(c)$, where $\alpha > 1$, $c > 0$, which we call a* ***Korobov class****. We say that f is a member of $E_d^\alpha(c)$ if all of f's Fourier coefficients satisfy*

$$\left| \hat{f}(\mathbf{m}) \right| \leq \frac{c}{\left(\prod_{i=1}^d \bar{m}_i \right)^\alpha}, \tag{44}$$

where

$$\bar{m} = \begin{cases} |m| & \text{if } |m| \geq 1 \text{ or} \\ 1 & \text{otherwise.} \end{cases} \tag{45}$$

3.2.2 An upper bound for Korobov classes

Theorem 4. *If $f \in E_d^\alpha(c)$, then*

$$|\mathcal{E}_{\mathcal{L}}(f)| \leq c \sum_{\mathbf{m} \in \mathcal{L}^\perp, \mathbf{m} \neq \mathbf{0}}{}' \frac{1}{\left(\prod_{i=1}^d \bar{m}_i \right)^\alpha}. \tag{46}$$

Proof.

$$|\mathcal{E}_{\mathcal{L}}| = \left| \sum_{\mathbf{m} \in \mathcal{L}^\perp, \mathbf{m} \neq \mathbf{0}}{}' \hat{f}(\mathbf{m}) \right| \tag{47}$$

$$\leq \sum_{\mathbf{m} \in \mathcal{L}^\perp, \mathbf{m} \neq \mathbf{0}}{}' \left| \hat{f}(\mathbf{m}) \right| \qquad \text{(Use triangle inequality.)} \tag{48}$$

$$= c \sum_{\mathbf{m} \in \mathcal{L}^\perp, \mathbf{m} \neq \mathbf{0}}{}' \frac{1}{\left(\prod_{i=1}^d \bar{m}_i \right)^\alpha} \qquad \text{(Use equation (44).)} \tag{49}$$

□

This bound shows that it remains to find '*good*' lattices, i.e., lattices which make the latter series small.

3.3 An example: analyzing lattices in one dimension

To conclude, let us apply our findings to the simple scenario of one-dimensional lattices, which turn out to be equidistant points, as shown below:

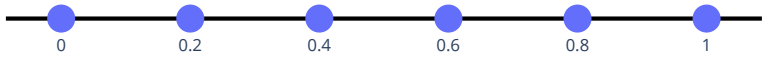

Figure 4: An integration lattice generated by the point $\frac{1}{5}$.

The generating point for our lattice $\mathcal{L}$ is going to be $\frac{1}{n}$, where $n \in \mathbb{N}$. By Lemma 3, we know that we can generate its dual with all integer multiples of $\frac{1}{n^{-1}} = n$, i.e.,

$$\mathcal{L}^{\perp} = \{kn, k \in \mathbb{Z}\}. \tag{50}$$

Now, assume that $f \in E_{\alpha}(c)$. Then we can use Theorem 4 to find an explicit upper bound for integrating f with $\mathcal{L}$:

$$|\mathcal{E}_{\mathcal{L}}(f)| \leq c \sum_{k \in \mathbb{Z}}{}' \frac{1}{(\bar{k}n)^{\alpha}} = \frac{c}{n^{\alpha}} \sum_{k \in \mathbb{Z}}{}' \frac{1}{\bar{k}^{\alpha}}; \tag{51}$$

if we further assume that $\alpha = 2, c = 1$, we get the upper bound

$$|\mathcal{E}_{\mathcal{L}}(f)| \leq \frac{1}{n^2} \sum_{k \in \mathbb{Z}}{}' \frac{1}{\bar{k}^2} = \frac{2}{n^2} \sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{3n^2}. \tag{52}$$

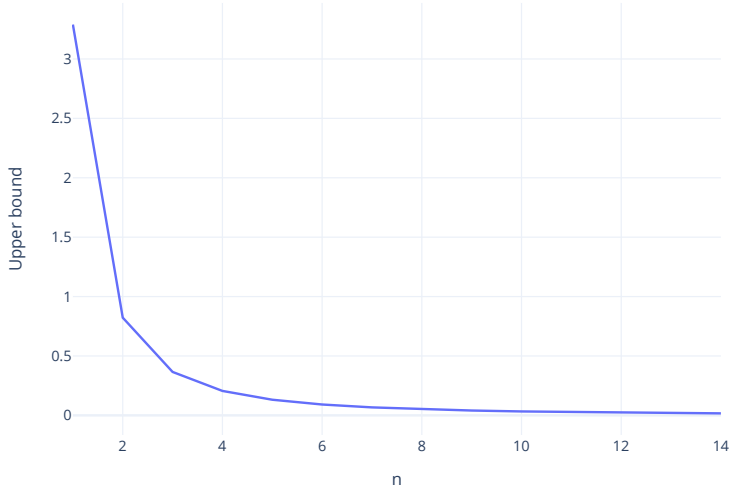

Figure 5: A plot of the upper-bound (52) for various n.

This means that the absolute integration error goes to zero as we send $n \to \infty$, which in turn means that our lattice integral converges to the true integral. This comes to no surprise, as this limit is also known as the *Riemann-integral*.

3.4 Lattices in higher dimensions

Finding '*good*' lattices in higher dimensions is much harder, and there are many open problems from different ares of mathematics related to this problem. Although there is no good solution to the aforementioned problem yet, let us state a last theorem.

Theorem 5. *There exist lattices $\mathcal{L}$, such that for all $f \in \mathrm{E}_d^\alpha(c)$*

$$|\mathcal{E}_\mathcal{L}| \leq cN^{-\alpha} \cdot \log(N)^{(d-1)(\alpha+1)}.$$

With our current knowledge, we cannot prove this statement; we refer to [4]. This shows that, also in higher dimensions, some lattices are capable of achieving a rate of convergence that depends on the dimension only logarithmically, which suggests that aforementioned lattice rules might also be useful in high dimensions. This is the subject of ongoing research, in particular, number theory and numerical analysis.

References

[1] Tom Gunter et al. "Sampling for Inference in Probabilistic Models with Fast Bayesian Quadrature". In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper/2014/file/e94f63f579e05cb49c05c2d050ead9c0-Paper.pdf.

[2] Plotly Technologies Inc. *Collaborative data science*. 2015. URL: https://plot.ly.

[3] Kevin P. Murphy. *Machine learning: a probablistic perspective*. CRC Press, 2012, pp. 205–206. ISBN: 978-0-262-010802-9.

[4] Ian H Sloan and Philip J. Kachoyan. "Lattice methods for multiple integration: Theory, Error Analysis and Examples". In: *Siam J. Numer. Anal.* 24.1 (1987).

[5] Stefan Weinzierl. *Introduction to Monte Carlo methods*. Report. 2000. URL: https://arxiv.org/abs/hep-ph/0006269.