MATHEMATICS OF COMPUTATION
Volume 88, Number 315, January 2019, Pages 369–387
http://dx.doi.org/10.1090/mcom/3320
Article electronically published on April 5, 2018

FINDING WELL APPROXIMATING LATTICES FOR A FINITE SET OF POINTS

A. HAJDU, L. HAJDU, AND R. TIJDEMAN

ABSTRACT. In this paper we address the task of finding well approximating lattices for a given finite set A of points in $\mathbb{R}^n$ motivated by practical texture analytic problems. More precisely, we search for $\boldsymbol{o}, \boldsymbol{d_1}, \ldots, \boldsymbol{d_n} \in \mathbb{R}^n$ such that $\boldsymbol{a} - \boldsymbol{o}$ is close to $\Lambda = \boldsymbol{d_1}\mathbb{Z} + \cdots + \boldsymbol{d_n}\mathbb{Z}$ for every $\boldsymbol{a} \in A$. First we deal with the one-dimensional case, where we show that in a sense the results are almost the best possible. These results easily extend to the multi-dimensional case where the directions of the axes are given, too. Thereafter we treat the general multi-dimensional case. Our method relies on the LLL algorithm. Finally, we apply the least squares algorithm to optimize the results. We give several examples to illustrate our approach.

1. INTRODUCTION

In this paper we are concerned with the problem of finding a well fitting grid (lattice) to a given finite set of points in $\mathbb{R}^n$. As our research roots in practical problems related to image processing, we start with sketching the motivating background.

Checking the spatial regularity of elements in nature is a highly investigated texture analytic task to detect abnormalities or to classify different patterns. As perhaps the most vivid field, we can highlight the 2D domain problems corresponding to digital image analysis with various applications considering segmentation, recognition and classification [8]. In general, the spectral and statistical frameworks are highly developed to support these efforts. However, the majority of these approaches consider some kind of assumption on the pattern structure. An emblematic example is the usage of the so-called co-occurrence matrices to extract Haralick features [4], where the method requests a position operator in terms of a spatial vector to give a robust statistical description of textures accordingly. As higher dimensional problems, we can mention the analysis of the regularity of 3D crystal structures [9]. Our motivation behind the introduction of the new theoretical models is to support these pattern recognition activities by introducing efficient (polynomial-time) algorithms in arbitrary dimensions, which are capable of discovering the regularity of the patterns automatically. That is, our approach may be considered as a preliminary step for pattern analysis techniques, which is currently

Received by the editor April 9, 2016, and, in revised form, January 11, 2017, and August 6, 2017.

2010 *Mathematics Subject Classification.* Primary 11J13, 11H06.

Key words and phrases. Diophantine approximation, lattice, LLL algorithm, least squares algorithm.

This research was supported in part by the OTKA grants K100339, NK101680, K115479 and the projects TÁMOP-4.2.2.C-11/1/KONV-2012-0001, EFOP-3.6.2-16-2017-00015 and VKSZ_14-1-2015-0072, SCOPIA: Development of diagnostic tools based on endoscope technology supported by the European Union, co-financed by the European Social Fund.

missing. On the other hand, the proposed algorithm can be also considered as a substantive measure to describe the regularity of point sets in any dimensions.

Now we formulate the problem we are studying in a precise way: given a finite set A of points in $\mathbb{R}^n$ which do not fit in an affine hyperplane, find $\boldsymbol{o}, \boldsymbol{d_1}, \ldots, \boldsymbol{d_n} \in \mathbb{R}^n$ such that the distance of $\boldsymbol{a} - \boldsymbol{o}$ to the lattice $\Lambda := \boldsymbol{d_1}\mathbb{Z} + \cdots + \boldsymbol{d_n}\mathbb{Z}$ is relatively small for every $\boldsymbol{a} \in A$. We shall call Λ a *well approximating lattice*, ignoring that a shift is made from the origin to $\boldsymbol{o}$. As usual, for $U \subset \mathbb{R}^n$, $\boldsymbol{v} \in \mathbb{R}^n$ and $r \in \mathbb{R}$ we write $U + \boldsymbol{v}$ for $\{\boldsymbol{u} + \boldsymbol{v} : \boldsymbol{u} \in U\}$ and rU for $\{r\boldsymbol{u} : \boldsymbol{u} \in U\}$. Vectors will always be denoted by boldface letters.

Let $A = \{\boldsymbol{a_1}, \ldots, \boldsymbol{a_k}\} \subset \mathbb{R}^n$ be given. If $k \leq n+1$, there is an optimal solution for the problem. In the sequel we assume $k > n+1$. Of course, we can make the distances arbitrarily small by choosing $\boldsymbol{d_1}, \ldots, \boldsymbol{d_n}$ extremely small. Therefore we need a measure which enables us to compare the quality of solutions in a fair way. To do so we introduce the maximum norm $N_{\Lambda,\boldsymbol{o}}(A)$ and the square norm $N^{(2)}_{\Lambda,\boldsymbol{o}}(A)$ by

$$N_{\Lambda,\boldsymbol{o}}(A) := \max_{\boldsymbol{a} \in A} \frac{|\boldsymbol{a} - \boldsymbol{o} - \Lambda|}{\Delta} \left(\frac{\operatorname{diam} A}{\Delta}\right)^{\frac{n}{k-n-1}}$$

and

$$N^{(2)}_{\Lambda,\boldsymbol{o}}(A) := \frac{\sqrt{\sum_{\boldsymbol{a} \in A} |\boldsymbol{a} - \boldsymbol{o} - \Lambda|^2}}{\Delta} \left(\frac{\operatorname{diam} A}{\Delta}\right)^{\frac{n}{k-n-1}},$$

where Δ is the nth root of the lattice determinant of Λ, and $\operatorname{diam}(A)$ is the diameter of A. We set

$$N(A) = \inf_{\Lambda,\boldsymbol{o}} N_{\Lambda,\boldsymbol{o}}(A) \quad \text{and} \quad N^{(2)}(A) = \inf_{\Lambda,\boldsymbol{o}} N^{(2)}_{\Lambda,\boldsymbol{o}}(A).$$

In Section 2, after Theorem 2.4, we explain why the above choice of the norms is appropriate in case $n = 1$.

We note that the problem in dimension one is close to that of finding approximate greatest common divisors of integers; see the papers [5] and [3]. However, our problem is different, because we approximate by real lattices and, moreover, allow the origin to shift to $\boldsymbol{o}$. On the other hand, in [5] and [3] algorithms are given to provide all solutions satisfying some condition, whereas we shall provide solutions without claiming completeness or optimality.

In Section 2, we deal with the one-dimensional case. Theorems 2.1 and 2.2 provide upper bounds for the above norms $N_{\Lambda,\boldsymbol{o}}(A)$, $N^{(2)}_{\Lambda,\boldsymbol{o}}(A)$, $N(A)$, $N^{(2)}(A)$. Our main tools are simultaneous Diophantine approximation and (the theory of) the LLL algorithm [6]. Theorem 2.3 yields that the bounds for $N(A)$ are rather sharp. Theorem 2.4 shows that homogeneous simultaneous Diophantine approximation appears in a natural way in the study of the problem. Finally, Theorem 2.7 tells that if a very well approximating lattice Λ exists, the LLL algorithm should find a well approximating lattice. In Section 3, we extend the algorithmic method of Section 2 to the multi-dimensional case by applying it to each coordinate axis. In Section 4 we generalize the method of Section 2 to the multi-dimensional case in a simultaneous way. In particular, we prove that our strategy provides a good approximation if a very good approximation exists (Theorem 4.1). This result can play a significant role in applications like the ones mentioned in the beginning of the paper. After that, perhaps as our most important contribution, we also give a formal algorithm to generate well approximating lattices. Finally, in Section 5, we use the least squares algorithm to optimize the numerical results with respect to the $N^{(2)}(A)$-norm. We illustrate the various methods by examples.

2. The one-dimensional case

Let $n = 1$ and $A = \{a_1, \dots, a_k\} \subset \mathbb{R}$ be given. Then we have, for o, d in $\mathbb{R}$, the maximum norm

$$N_{d,o}(A) = \frac{\max_{i=1,\dots,k} |a_i - o - d\mathbb{Z}|}{d} \left(\frac{\max_{i<j} |a_i - a_j|}{d} \right)^{\frac{1}{k-2}}$$

and the square norm

$$N^{(2)}_{d,o}(A) = \frac{\sqrt{\sum_{i=1}^k |a_i - o - d\mathbb{Z}|^2}}{d} \left(\frac{\max_{i<j} |a_i - a_j|}{d} \right)^{\frac{1}{k-2}}.$$

We give bounds for

$$N(A) = \inf_{d,o} N_{d,o}(A), \quad N^{(2)}(A) = \inf_{d,o} N^{(2)}_{d,o}(A)$$

taken over all $d \in \mathbb{R}$ with $0 < d \le \operatorname{diam}(A)$ and $o \in \mathbb{R}$, and construct pairs d, o which provide good simultaneous approximations. The upper bound on d is to avoid large d's, which would result in the trivial $N(A) = 0$.

Theorem 2.1. *For any finite subset A of $\mathbb{R}$ we have $N(A) < 1$ and $N^{(2)}(A) < \sqrt{k-2}$.*

It is important to efficiently construct well approximating lattices. In this direction we prove the following result.

Theorem 2.2. *For any finite subset A of $\mathbb{R}$ one can find $d \in \mathbb{R}_{>0}$ with $N_{d,a_1}(A) < 2^{(k-1)/4}$ and $N^{(2)}_{d,a_1}(A) < 2^{(k-1)/4}\sqrt{k-2}$ in polynomial time.*

Theorem 2.1 is in some sense the best possible. This is demonstrated by the following theorem, showing that for any k there exist sets $A = \{a_1, \dots, a_k\}$ with norms bounded away from zero.

Theorem 2.3. *There exist arbitrarily large finite subsets A of $\mathbb{R}$ such that $N(A) > c_1$, where c_1 is a positive number depending only on A.*

The next result shows that existence of a very good inhomogeneous approximating lattice implies the existence of a quite good homogeneous simultaneous diophantine approximation.

Theorem 2.4. *Let $A = \{a_1, \dots, a_k\}$ be a set of real numbers with $k > 2$ and $a_1 < \dots < a_k$. Suppose that for some $d, o \in \mathbb{R}$ with $0 < d \le a_k - a_1$ we have $N_{d,o}(A) < c_2$ where c_2 is a positive real number. Then there exists a positive integer q such that*

$$\max_{1\le i\le k} ||q\alpha_i|| < 6c_2 q^{-1/(k-2)} \quad \text{and} \quad \left| \frac{a_k - a_1}{d} - q \right| < 3c_2 q^{-1/(k-2)},$$

where $\alpha_i = (a_i - a_1)/(a_k - a_1)$ $(i = 2, \dots, k-1)$, and $||.||$ denotes the distance to the nearest integer.

With these theorems it is possible to explain why $N(A)$ is a fair norm, at least in dimension one. Let $A = \{a_1, \ldots, a_k\}$ with $a_1 < \cdots < a_k$. In the first place we should compensate for scaling. If we multiply all a_i by a number $\alpha > 0$, then all the distances are also multiplied by α and so, by dividing by d we neutralize scaling. We further have to compensate for the value of d, which is close to $(a_k - a_1)/q$. According to Theorem 2.4 the expected distances from $q\alpha_i$ to the nearest lattice point is of the order $q^{-1/(k-2)}$. Therefore we multiply by $((a_k - a_1)/d)^{1/(k-2)}$ for compensation.

Remark. Since $\max_{i<j} |a_i - a_j|/d$ is close to q and $\max_{i=1,\ldots,k} |a_i - o - d\mathbb{Z}|/d$ generically approximates 0.5, the expected value of $N_{d,o}(A)$ for a random q is $(0.5 - o_k(1))q^{1/(k-2)}$.

We shall present some examples using the LLL procedure of Maple 15. Here we have to choose some parameters. Without loss of generality we may assume $a_1 < a_2 < \cdots < a_k$. First we subtract a_1 from every element of A. Then we scale all the obtained numbers by dividing them by the largest distance between the numbers $a_i - a_1$, i.e., by $a_k - a_1$. Further we choose an ε of order $d_0^{1/(k-2)}$ where d_0 is the size of the desired d. It may be good to try various values of ε and to compare the different outcomes. In all our examples, unless it is stated differently, we compute with 10-digit precision, but give the data in 6-digit precision.

Example 2.5. We choose $k = 6$ and

$$A = \{a_1 = 0.814258,\ a_2 = 1.294837,\ a_3 = 2.237840,\ a_4 = 2.764132, a_5 = 4.295116,\ a_6 = 7.733842\}.$$

By subtracting a_1 and dividing by $a_6 - a_1 = \max_{i<j} |a_j - a_i|$ we get the normalized numbers

$$(a_1^N, a_2^N, a_3^N, a_4^N, a_5^N, a_6^N) = (0, 0.069452, 0.205732, 0.281791, 0.503044, 1).$$

We apply the LLL algorithm[1] with $\varepsilon = 10^{-3}$ to the matrix

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0.069452 & 0.205732 & 0.281791 & 0.503044 & 0.001000 \end{pmatrix}$$

to get the matrix

$$B = \begin{pmatrix} 0.027672 & 0.119748 & 0.054931 & -0.042622 & -0.014000 \\ 0.098213 & -0.049067 & -0.085426 & -0.101183 & 0.131000 \\ 0.151379 & -0.060477 & -0.131268 & -0.063214 & -0.185000 \\ 0.277808 & -0.177071 & 0.127163 & 0.012178 & 0.004000 \\ -0.1117778 & -0.104443 & 0.202424 & -0.267907 & -0.088000 \end{pmatrix}.$$

Subsequently, we compute the integer matrix

$$S := BT^{-1} = \begin{pmatrix} 1 & 3 & 4 & 7 & -14 \\ -9 & -27 & -37 & -66 & 131 \\ 13 & 38 & 52 & 93 & -185 \\ 0 & -1 & -1 & -2 & 4 \\ 6 & 18 & 25 & 44 & -88 \end{pmatrix}.$$

[1]Here and elsewhere, we always use the implementation of LLL in Maple 15.

We conclude that $(p_1, p_2, \ldots, p_6) = (0, 1, 3, 4, 7, 14)$ with $q = 14$ yields a good approximation. The second row of S gives other good values: $(p'_1, p'_2, \ldots, p'_6) = (0, 9, 27, 37, 66, 131)$ with $q' = 131$. We put $o = a_1$ and choose

$$d = (a_6 - a_1)/q = 0.494256 \text{ and } d' = (a_6 - a_1)/q' = 0.052821,$$

as lattice bases, respectively. Thus, the points $(a_1, \ldots, a_6)$ are close to $o + d \cdot (0, 1, 3, 4, 7, 14)$ and to $o + d' \cdot (0, 9, 27, 37, 66, 131)$. The errors of approximation of A are given by

$$N_{d,o}(A) = \frac{0.059186}{0.494256} \sqrt[4]{\frac{6.919584}{0.494256}} = 0.231632$$

and

$$N_{d',o}(A) = \frac{0.005345}{0.052821} \sqrt[4]{\frac{6.919584}{0.052821}} = 0.342315.$$

Hence

$$N(A) \leq \min(0.231632, 0.342315) = 0.231632$$

which is less than 1 and $2^{(k-1)/4} = 2.378414$, the bounds of Theorems 2.1 and 2.2, respectively. In a similar way we find

$$N^{(2)}_{d,o}(A) = 0.273141, \quad N^{(2)}_{d',o}(A) = 0.581947.$$

If we start with $\varepsilon = 10^{-2}$ in place of $\varepsilon = 10^{-3}$ and we follow the same procedure, then we obtain again $q = 14$, yielding the same error values.

Example 2.6. In the second example we choose

$$A = \{a_1 = 0,\ a_2 = \sqrt{3} = 1.732051,\ a_3 = \sqrt{5} = 2.236068,$$
$$a_4 = \sqrt{7} = 2.645751,\ a_5 = \sqrt{11} = 3.316625,\ a_6 = \sqrt{13} = 3.605551\}.$$

The largest distance is $a_6 - a_1 = \sqrt{13}$. Dividing by this number gives the normalized numbers

$$(a_1^N, a_2^N, a_3^N, a_4^N, a_5^N, a_6^N) = (0, 0.480384, 0.620174, 0.733799, 0.919866, 1).$$

Applying the LLL algorithm to the matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0.480384 & 0.620174 & 0.733799 & 0.919866 & 0.001000 \end{pmatrix}$$

we get a matrix B with first row

$$\begin{pmatrix} -0.057669 & -0.026051 & -0.069908 & 0.020068 & -0.150000 \end{pmatrix}.$$

Subsequently, we compute the integer matrix $S = BT^{-1}$ and obtain

$$\begin{pmatrix} 72 & 93 & 110 & 138 & -150 \end{pmatrix}$$

as its first row. We conclude that $q = 150$ together with

$$(p_1, p_2, \ldots, p_6) = (0, 72, 93, 110, 138, 150)$$

yields a good approximation. We get

$$o = a_1 = 0, d = \sqrt{13}/q = 0.024037.$$

Thus the points $(a_1, \dots, a_6)$ are close to $d \cdot (0, 72, 93, 110, 138, 150)$. In this way we obtain

$$N(A) \le N_{d,o}(A) = 0.244652, \quad N^{(2)}(A) \le N_{d,o}^2(A) = 0.337388.$$

The upper bounds are again less than the bounds from Theorems 2.1 and 2.2.
If we start with $\varepsilon = 10^{-2}$ in place of $\varepsilon = 10^{-3}$ and we follow the same procedure, we obtain $q = 8$, $N_{d,o}(A) = 0.603645, N_{d,o}^{(2)}(A) = 0.696969$. □

The next theorem shows that a well approximating lattice with similarly sized d cannot be much better than the lattice we found by the LLL algorithm.

Theorem 2.7. *Let $a_1 = 0 < a_2 < \dots < a_{k-1} < a_k = 1$, and let L be the lattice generated by the $k-1$ vectors*

$$(1, 0, 0, \dots, 0), \dots, (0, \dots, 0, 1, 0), (a_2, \dots, a_{k-1}, \varepsilon)$$

in $\mathbb{R}^{k-1}$, where ε is a positive real number. Let $\boldsymbol{b_1}$ be the shortest vector of an LLL-reduced basis of L. Then for every $d' > \frac{\sqrt{k}\ 2^{k/2-1}\varepsilon}{|\boldsymbol{b_1}|}$ and every choice of integers $p'_1 = 0, p'_2, \dots, p'_k = q'$ we have

$$\max_{i=1,\dots,k} |a_i - p'_i d'| \ge \frac{2^{1-k/2}}{q'\sqrt{k}} |\boldsymbol{b_1}|,$$

whereas the LLL algorithm finds integers $p_1 = 0, p_2, \dots, p_{k-1}$, $p_k = q = 1/d$ such that

$$\max_{i=1,\dots,k} |a_i - p_i d| \le \frac{|\boldsymbol{b_1}|}{q}.$$

For the proofs of the theorems we first turn to Theorem 2.2.

Proof of Theorem 2.2. Let $A = \{a_1, \dots, a_k\} \subset \mathbb{R}$ $(k \ge 3)$. Without loss of generality we may assume that $a_1 < \dots < a_k$. Let $o = a_1$, and $\alpha_i = (a_i - a_1)/(a_k - a_1)$ for $i = 2, \dots, k-1$. Then, by Lemma 2.3 of [1], for any $\varepsilon \in (0, 1)$ integers $p_2, \dots, p_{k-1}$ and q can be found in polynomial time such that

$$|p_i - q\alpha_i| \le \varepsilon \text{ for } i = 2, \dots, k-1,$$

$$1 \le q \le 2^{(k-2)(k-1)/4} \varepsilon^{-k+2}.$$

We note that the same assertion follows already from Proposition 1.39 of [6] in case of $A \subset \mathbb{Q}$. Put $d = (a_k - a_1)/q$. For $i = 2, \dots, k-1$ we have

$$|p_i - q\alpha_i| \le \varepsilon \le 2^{(k-1)/4} q^{-1/(k-2)}. \tag{1}$$

Since

$$|p_i - q\alpha_i| = |p_i d - (a_i - o)|/d \quad (i = 2, \dots, k-1), \tag{2}$$

we obtain by (1) and (2), putting $p_1 = 0, p_k = q$,

$$N(A) \le N_{d,o}(A) = \frac{\max_{i=1,\dots,k} |p_i d - (a_i - o)|}{d} \left(\frac{a_k - a_1}{d} \right)^{\frac{1}{k-2}} \le 2^{(k-1)/4}.$$

For the upper estimate for $N^{(2)}(A)$ we use the bound 0 for $i = 1, k$ and the bound $2^{(k-1)/4}$ for $i = 2, \dots, k-1$. □

Proof of Theorem 2.1. The proof goes along the same lines as that of Theorem 2.2. The only difference is that in place of Lemma 2.3 of [1], we use a theorem of Dirichlet (see Schmidt [7], Chapter II) in (1), guaranteeing the existence of an integer q such that

$$||q\alpha_i|| < q^{-1/(k-2)} \quad (i = 2, \dots, k-1),$$

where $||.||$ denotes the distance to the nearest integer. Note that in place of Dirichlet's theorem one could also apply Theorem 1.1 of [1]. $\square$

Proof of Theorem 2.3. Let $0 = a_1 < a_2 < \cdots < a_{k-1} < a_k = 1$ be k numbers contained in a real algebraic number field of degree $k-1$ such that $a_2, \dots, a_{k-1}, 1$ are linearly independent over $\mathbb{Q}$. Then, by Theorem III on p. 79 of [2], there exists a positive real number γ (depending only on $a_2, \dots, a_{k-1}$) such that for all positive integers q we have

$$\max_{2 \le i \le k-1} ||qa_i|| \ge \gamma q^{-1/(k-2)}. \tag{3}$$

Suppose that for some real numbers o, d, ε with $0 < d \le 1, 0 < \varepsilon < 1/4$ we have

$$\max_{1 \le i \le k} |a_i - p_i d - o| < \varepsilon d, \tag{4}$$

where the p_i are integers. Then

$$\max_{1 \le i \le k} |(a_i - a_1) - (p_i - p_1)d| < 2\varepsilon d. \tag{5}$$

Set $t = 1/d$. Then we have, by $a_1 = 0, a_k = 1$,

$$\max_{2 \le i \le k-1} ||ta_i|| < 2\varepsilon \quad \text{and} \quad ||t|| < 2\varepsilon. \tag{6}$$

Let q be an integer with $|t - q| \le 1/2$. Then for $i = 2, \dots, k-1$ we obtain, by (3) and (6),

$$||ta_i|| \ge ||qa_i|| - |t-q|a_i = ||qa_i|| - ||t||a_i \ge \gamma q^{-1/(k-2)} - 2\varepsilon.$$

Observe that $t/q \ge 1/2$. By choosing $\varepsilon = \gamma/(4q^{1/(k-2)})$, the above inequality yields

$$\max_{2 \le i < k} ||ta_i|| \ge 2\varepsilon$$

which contradicts (6). Hence for the chosen value of ε inequality (4) does not hold. It follows that

$$N_{d,o}(A) = \frac{\max_{i=1,\dots,k} |a_i - o - d\mathbb{Z}|}{d^{1+1/(k-2)}} \ge \varepsilon t^{1/(k-2)} \ge \frac{\gamma}{4}\left(\frac{t}{q}\right)^{1/(k-2)} \ge \frac{\gamma}{8}. \qquad \square$$

Proof of Theorem 2.4. Let o and d be real numbers with $0 < d \le a_k - a_1$. Suppose that $N_{d,o}(A) < c_2$, where c_2 is an arbitrary positive real. Then we have

$$d^{-1} \max_{1 \le i \le k} |a_i - p_i d - o| < \varepsilon := c_2 \left(\frac{d}{a_k - a_1}\right)^{1/(k-2)}.$$

Hence

$$d^{-1} \max_{1 \le i \le k} |a_i - a_1 - (p_i - p_1)d| < 2\varepsilon.$$

Put $t = (a_k - a_1)/d$ and $\alpha_i = (a_i - a_1)/(a_k - a_1)$ for $i = 1, \dots, k$. Then

$$\max_{2 \le i \le k-1} ||t\alpha_i|| < 2\varepsilon \quad \text{and} \quad ||t|| < 2\varepsilon. \tag{7}$$

Thus choosing integers q such that $|t-q| = ||t||$ and p_i such that $|t\alpha_i - p_i| = ||t\alpha_i||$, by (7) we get for all $i = 2, \dots, k-1$

$$|q\alpha_i - p_i| < |t\alpha_i - p_i| + |t-q|\alpha_i < 4\varepsilon.$$

Hence by the definition of ε and t, noting that by $t \geq 1$ we have $q > 0$ and $q/t \leq 4/3$, we derive

$$\max_{2\leq i\leq k-1} ||q\alpha_i|| < 4\varepsilon = 4c_2 t^{-1/(k-2)} = 4c_2 \frac{(q/t)^{1/(k-2)}}{q^{1/(k-2)}} < \frac{16}{3} c_2 q^{-1/(k-2)}.$$

Similarly, by (7),

$$\left| \frac{a_k - a_1}{d} - q \right| = |t-q| < 2\varepsilon < \frac{8}{3} c_2 q^{-1/(k-2)}. \qquad \square$$

Proof of Theorem 2.7. According to [6, Theorem 1.11] we have, for every lattice point $\boldsymbol{x} \in L$,

$$|\boldsymbol{b_1}|^2 \leq 2^{k-2} |\boldsymbol{x}|^2.$$

Clearly, any lattice point $\boldsymbol{x} \in L$ can be written as

$$\boldsymbol{x} = \left(p'_2 - q'a_2, \dots, p'_{k-1} - q'a_{k-1}, -q'\varepsilon\right),$$

where $p'_2, \dots, p'_{k-1}, p'_k = q'$ are integers. Hence

$$\max\left(\max_{i=2,\dots,k-1} |p'_i - q'a_i|, |q'\varepsilon| \right)^2 \geq \frac{2^{2-k}}{k} |\boldsymbol{b_1}|^2.$$

So provided that $|q'\varepsilon| < \frac{2^{1-k/2}}{\sqrt{k}} |\boldsymbol{b_1}|$, we have

$$\max_{i=2,\dots,k-1} |p'_i - q'a_i| \geq \frac{2^{1-k/2}}{\sqrt{k}} \, |\boldsymbol{b_1}|.$$

Put $d' = 1/q'$. Then, provided that $d' > \frac{2^{k/2-1}\sqrt{k}\,\varepsilon}{|\boldsymbol{b_1}|}$, we have

$$\max_{i=2,\dots,k-1} |a_i - p'_i d'| \geq \frac{2^{1-k/2}}{q'\sqrt{k}} |\boldsymbol{b_1}|.$$

For the second part, observe that it follows from the algorithm in [6] that for some integers $p_2, \dots, p_{k-1}, q$ we have

$$\sum_{i=2}^{k-1} (p_i - qa_i)^2 + (q\varepsilon)^2 = |\boldsymbol{b_1}|^2.$$

Hence $|p_i - qa_i| \leq |\boldsymbol{b_1}|$ which implies $|a_i - p_i/q| \leq |\boldsymbol{b_1}|/q$ for $i = 1, \dots, k$ in view of $a_1 = 0 = p_1, a_k = 1, q = p_k$. $\square$

3. Lattices with the basis vectors in given directions

For any given finite set $A \subset \mathbb{R}^n$ which do not fit into an affine hyperplane we generate vectors $\boldsymbol{d} = (d_1, \dots, d_n)$, $\boldsymbol{o} = (o_1, \dots, o_n) \in \mathbb{R}^n$ such that every element of $A - \boldsymbol{o}$ is close to the lattice $\Lambda = (d_1\mathbb{Z}, \dots, d_n\mathbb{Z})$. That is, we approximate A with a rectangular lattice. By a linear transformation one can transfer any other prescribed set of lattice basis vectors to this case.

Let $A = \{\boldsymbol{a_1}, \dots, \boldsymbol{a_k}\}$ with $\boldsymbol{a_i} = (a_{i1}, \dots, a_{in}) \in \mathbb{R}^n$ for $i = 1, \dots, k$. As norms we use

$$N_{\boldsymbol{d},\boldsymbol{o}}(A) := \frac{\max_{\boldsymbol{a}\in A} |\boldsymbol{a} - \boldsymbol{o} - \Lambda|}{\Delta} \left(\frac{\text{diam } A}{\Delta}\right)^{\frac{n}{k-n-1}}$$

and

$$N^{(2)}_{\boldsymbol{d},\boldsymbol{o}}(A) := \frac{\sqrt{\sum_{\boldsymbol{a}\in A} |\boldsymbol{a} - \boldsymbol{o} - \Lambda|^2}}{\Delta} \left(\frac{\text{diam } A}{\Delta}\right)^{\frac{n}{k-n-1}},$$

where $\Delta := \left(\prod_{i=1}^n |d_i|\right)^{1/n}$ is the nth root of the lattice determinant. Again we fix $\boldsymbol{o} = \boldsymbol{a_1}$.

We illustrate by two examples how the results of Section 2 can be used in this case.

Example 3.1. We combine Examples 2.5 and 2.6. We choose $n = 2$, $k = 6$ and

$$A = \{\boldsymbol{a_1} = (0.814258, 0),\ \boldsymbol{a_2} = (1.294837, \sqrt{3}),\ \boldsymbol{a_3} = (2.237840, \sqrt{5}),$$

$$\boldsymbol{a_4} = (2.764132, \sqrt{7}),\ \boldsymbol{a_5} = (4.295116, \sqrt{11}),\ \boldsymbol{a_6} = (7.733842, \sqrt{13})\}.$$

We put $\varepsilon = 10^{-3}$. Recall

$$\sqrt{3} = 1.732051,\ \sqrt{5} = 2.236068,\ \sqrt{7} = 2.645751,$$

$$\sqrt{11} = 3.316625,\ \sqrt{13} = 3.605551.$$

From Examples 2.5 and 2.6 we obtain $\boldsymbol{d} = (0.494256, 0.024037)$ and the approximating points become

$$(0.814258, 0),\ (1.308514, 1.732051),\ (2.297026, 2.236068),$$

$$(2.791282, 2.645751),\ (4.274050, 3.316625),\ (7.733842, 3.605551).$$

Using $\Delta = \sqrt{0.494256 \cdot 0.024037} = 0.108997$ and $\text{diam}(A) = |\boldsymbol{a_6} - \boldsymbol{a_1}| = 7.802605$, the values $N_{\boldsymbol{d},\boldsymbol{o}}(A) = 9.362160$, $N^{(2)}_{\boldsymbol{d},\boldsymbol{o}}(A) = 11.045325$ follow. □

In Example 3.1 both coordinates are in increasing order. Of course, this need not be the case. In the next example we permute the second coordinates.

Example 3.2. We choose $n = 2$, $k = 6$, and

$$A = \{\boldsymbol{a_1} = (0.814258, \sqrt{5}),\ \boldsymbol{a_2} = (1.294837, 0),\ \boldsymbol{a_3} = (2.237840, \sqrt{13}),$$

$$\boldsymbol{a_4} = (2.764132, \sqrt{3}),\ \boldsymbol{a_5} = (4.295116, \sqrt{11}),\ \boldsymbol{a_6} = (7.733842, \sqrt{7})\}.$$

We take $\varepsilon = 10^{-3}$. From Examples 2.5 and 2.6 we obtain that

$$\boldsymbol{d} = (0.494256, 0.024037).$$

The approximating lattice remains the same. Therefore the approximating points are obtained from Example 3.1 by making the corresponding permutation. Using $\Delta = 0.108997$ and $\text{diam}(A) = |\boldsymbol{a_6} - \boldsymbol{a_2}| = 6.961378$, we obtain the values $N(A) \le N_{\boldsymbol{d},\boldsymbol{o}}(A) = 8.676052$, $N^{(2)}(A) \le N^{(2)}_{\boldsymbol{d},\boldsymbol{o}}(A) = 10.236439$. That the values are smaller than the corresponding values in Example 3.1 is mainly due to the smaller diameter. □

4. Approximating with general lattices

In this section we present a method for finding well approximating general lattices. First we prove that our strategy provides a good approximation if a very good approximation exists. We generalize the method of Section 2 and illustrate how it works through some examples.

Theorem 4.1. *If A is a finite set of k points in $\mathbb{R}^n$ with $k > n$ such that they do not fit into an affine hyperplane and there exist a lattice Λ, a point $\boldsymbol{o} \in \mathbb{R}^n$ and an $\varepsilon > 0$ such that $|\boldsymbol{a} - \boldsymbol{o} - \Lambda| < \varepsilon < 1/2$ for all $\boldsymbol{a} \in A$, then there exist an affine (inhomogeneous linear) transformation $V : \mathbb{R}^n \to \mathbb{R}^n$ and $\boldsymbol{a_1}, \dots, \boldsymbol{a_{n+1}} \in A$ such that V maps the lattice point in Λ nearest to $\boldsymbol{a_i} - \boldsymbol{o}$ to $\boldsymbol{a_i} - \boldsymbol{o}$ itself for $i = 1, \dots, n+1$ and $|\boldsymbol{a} - \boldsymbol{o} - V(\Lambda)| < 2^n\varepsilon$ for all $\boldsymbol{a} \in A$.*

We shall use the following lemma.

Lemma 4.2. *Let a rectangular box*

$$B = \{0 \le x_1 \le b_1, |x_2| \le b_2, \dots, |x_n| \le b_n : b_1, b_2, \dots, b_n \in \mathbb{R}_{>0}\}$$

be given in $\mathbb{R}^n$. Set $\boldsymbol{a_1''} = (b_1, 0, \dots, 0)$ and for $i = 2, \dots, n$ let $\boldsymbol{a_i''}$ be a point of the form $(b_{1i}, \dots, b_{ii}, 0, \dots, 0)$ with

$$0 \le b_{1i} \le b_1, |b_{ji}| \le b_j \text{ for } j = 2, \dots, i-1, b_{ii} = b_i. \tag{8}$$

Then every point $\boldsymbol{x} \in B$ can be written as $\lambda_1 \boldsymbol{a_1''} + \dots + \lambda_n \boldsymbol{a_n''}$ with $|\lambda_i| \le 2^{n-i}$ for $i = 1, \dots, n$.

Proof. By induction on n. For $n = 1$ the assertion is obvious. Suppose the statement is true for n. Then set

$$B_{n+1} = \{0 \le x_1 \le b_1, |x_2| \le b_2, \dots, |x_{n+1}| \le b_{n+1} : b_1, \dots, b_{n+1} \in \mathbb{R}_{>0}\}.$$

We identify the tuple $(x_1, \dots, x_n) \in B_n$ with $(x_1, \dots, x_n, 0) \in B_{n+1}$ and write $\boldsymbol{b_{n+1}} = (0, \dots, 0, b_{n+1})$. Any point $\boldsymbol{x} \in B_{n+1}$ can be written as $\boldsymbol{x'} + \mu \boldsymbol{b_{n+1}}$ with $\boldsymbol{x'} \in B_n$ and $-1 \le \mu \le 1$. By the induction hypothesis $\boldsymbol{x'}$ can be written as $\lambda_1 \boldsymbol{a_1''} + \dots + \lambda_n \boldsymbol{a_n''}$ with $|\lambda_i| \le 2^{n-i}$ for $i = 1, \dots, n$. By the same hypothesis $\boldsymbol{a_{n+1}''}$ can be written as $\mu_1 \boldsymbol{a_1''} + \dots + \mu_n \boldsymbol{a_n''} + \boldsymbol{b_{n+1}}$ with $|\mu_i| \le 2^{n-i}$ for $i = 1, \dots, n$. Hence

$$\begin{aligned} \boldsymbol{x} &= \lambda_1 \boldsymbol{a_1''} + \dots + \lambda_n \boldsymbol{a_n''} + \mu(\boldsymbol{a_{n+1}''} - \mu_1 \boldsymbol{a_1''} \dots - \mu_n \boldsymbol{a_n''}) \\ &= (\lambda_1 - \mu\mu_1)\boldsymbol{a_1''} + \dots + (\lambda_n - \mu\mu_n)\boldsymbol{a_n''} + \mu \boldsymbol{a_{n+1}''}. \end{aligned}$$

It follows that for $i = 1, \dots, n$ the coefficient of $\boldsymbol{a_i''}$ satisfies

$$|\lambda_i - \mu\mu_i| \le 2^{n-i} + 2^{n-i} = 2^{n+1-i}. \qquad \square$$

Proof of Theorem 4.1. Suppose that a lattice $\Lambda \subset \mathbb{R}^n$, $\boldsymbol{o} \in \mathbb{R}^n$ and $\varepsilon > 0$ are such that $|\boldsymbol{a} - \boldsymbol{o} - \Lambda| < \varepsilon < 1/2$ for all $\boldsymbol{a} \in A$. Let $\boldsymbol{a'}$ denote the lattice point of Λ nearest to $\boldsymbol{a} - \boldsymbol{o}$ for $\boldsymbol{a} \in A$ and let A' be the set of such lattice points $\boldsymbol{a'}$. Then, for $\boldsymbol{a} \in A$, $|\boldsymbol{a} - \boldsymbol{o} - \boldsymbol{a'}| < \varepsilon$. Without loss of generality we assume that $\boldsymbol{a_1'}, \boldsymbol{a_2'}$ are such that $|\boldsymbol{a_2'} - \boldsymbol{a_1'}|$ equals the diameter b_1 of A', $\boldsymbol{a_3'} \in A'$ is such that the distance b_2 from $\boldsymbol{a_3'}$ to the line through $\boldsymbol{a_1'}$ and $\boldsymbol{a_2'}$ is maximal, $\boldsymbol{a_4'}$ is such that the distance b_3 from $\boldsymbol{a_4'}$ to the plane through $\boldsymbol{a_1'}, \boldsymbol{a_2'}, \boldsymbol{a_3'}$ is maximal, and so on up to b_n. Let $\boldsymbol{a_{n+2}'}, \dots, \boldsymbol{a_k'}$ be the remaining points of A'. Label the elements of A accordingly. Hence, for $i = 1, \dots, k$ we have $|\boldsymbol{a_i} - \boldsymbol{o} - \boldsymbol{a_i'}| < \varepsilon$. Set

$$B = \{0 \le x_1 \le b_1, |x_2| \le b_2, \dots, |x_n| \le b_n\}.$$

Consider the affine transformation U for which $U\boldsymbol{a_1'} = (0,\dots,0)$ and $U\boldsymbol{a_{i+1}'} = (b_{1i}, b_{2i}, \dots, b_{i-1,i}, b_i, 0\dots,0)$ $(i = 1,\dots,n)$. Then by the choice of $\boldsymbol{a_1'},\dots,\boldsymbol{a_{n+1}'}$ we have that $U\boldsymbol{a_i'} \in B$ for $i = 1,\dots,k$, and also that $U\boldsymbol{a_1},\dots,U\boldsymbol{a_{n+1}}$ satisfy the conditions of Lemma 4.2. Hence every $U\boldsymbol{a_i'}$ can be written as $q_{1i}U\boldsymbol{a_2'} + q_{2i}U\boldsymbol{a_3'} + \cdots + q_{ni}U\boldsymbol{a_{n+1}'}$ with $|q_{ji}| \le 2^{n-j}$ for $j = 1,\dots,n$. Therefore every $\boldsymbol{a_i'}$ can be written as $\boldsymbol{a_i'} = q_{1i}\boldsymbol{a_2'} + q_{2i}\boldsymbol{a_3'} + \cdots + q_{ni}\boldsymbol{a_{n+1}'}$ with $|q_{ji}| \le 2^{n-j}$ for $j = 1,\dots,n$.

Let V be the affine transformation which maps $\boldsymbol{a_i'}$ to $\boldsymbol{a_i} - \boldsymbol{o}$ for $i = 1,\dots,n+1$. Then $|\boldsymbol{a_i'} - V(\boldsymbol{a_i'})| < \varepsilon$ for $i = 1,\dots,n+1$. Let $\Lambda' = V(\Lambda)$. Then for $i = 1,\dots,k$ we have $V(\boldsymbol{a_i'}) \in \Lambda'$ and

$$|\boldsymbol{a_i} - \boldsymbol{o} - V(\boldsymbol{a_i'})| \le |\boldsymbol{a_i} - \boldsymbol{o} - \boldsymbol{a_i'}| + |\boldsymbol{a_i'} - V(\boldsymbol{a_i'})|$$

$$\le \varepsilon + \sum_{j=1}^{n} |q_{ji}||\boldsymbol{a_{j+1}'} - V(\boldsymbol{a_{j+1}'})| \le \varepsilon + \varepsilon \sum_{j=1}^{n} 2^{n-j} = 2^n\varepsilon. \qquad \square$$

Theorem 4.1 shows that the original inhomogeneous problem is not far from a homogeneous problem. We follow this idea in our treatment enclosed in the following algorithmic form.

Algorithm. Finding well approximating lattices in the general case.

Input. Point set A to approximate.

Step 1. Normalize A by a shift and linear transformation W.

Details and background. Let $A = \{\boldsymbol{a_1},\dots,\boldsymbol{a_k}\} \subset \mathbb{R}^n$, and write $\boldsymbol{a_i} = (a_{1i},\dots,a_{ni})$ for $i = 1,\dots,k$. We apply first a normalization as we did in Section 2. We choose $n+1$ points of A, say $\boldsymbol{a_1},\dots,\boldsymbol{a_{n+1}}$ as in the proof of Theorem 4.1. We first apply a shift which moves $\boldsymbol{a_1}$ to the origin $\boldsymbol{a_1^N} = \boldsymbol{o}$ and subsequently a linear transformation W which moves $\boldsymbol{a_{i+1}} - \boldsymbol{a_1}$ to the unit vector $\boldsymbol{a_{i+1}^N} := \boldsymbol{e_i}$ for $i = 1,\dots,n$. By this shift and transformation the $k-n-1$ remaining points of A, $\boldsymbol{a_{n+2}},\dots,\boldsymbol{a_k}$, say, move to points $\boldsymbol{a_{n+2}^N},\dots,\boldsymbol{a_k^N}$, respectively.

Step 2. Compose a matrix T from the normalized elements and apply LLL to obtain a matrix B.

Details and background. Consider the $(k-1)\times(k-1)$ matrix

$$T := \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ a_{1(n+2)}^N & a_{1(n+3)}^N & \cdots & a_{1k}^N & \varepsilon & 0 & \cdots & 0 \\ a_{2(n+2)}^N & a_{2(n+3)}^N & \cdots & a_{2k}^N & 0 & \varepsilon & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n(n+2)}^N & a_{n(n+3)}^N & \cdots & a_{nk}^N & 0 & 0 & \dots & \varepsilon \end{pmatrix}$$

with a small ε, to be specified later in an appropriate way. Let $B = (b_{ij})_{i,j=1,\dots,k-1}$ denote the $(k-1)\times(k-1)$ matrix obtained by applying the LLL algorithm to the rows of T as in [6]. We expect the resulting entries b_{ij} of B to be relatively small.

Step 3. Calculate the unimodular transformation matrix S such that $B = ST$.

Details and background. There exists a unimodular $(k-1)\times(k-1)$ transformation matrix S such that $B = ST$ holds. We set

$$S := \begin{pmatrix} p_{11} & \cdots & p_{1(k-n-1)} & q_{11} & \cdots & q_{1n} \\ p_{21} & \cdots & p_{2(k-n-1)} & q_{21} & \cdots & q_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{(k-1)1} & \cdots & p_{(k-1)(k-n-1)} & q_{(k-1)1} & \cdots & q_{(k-1)n} \end{pmatrix}.$$

Note that all the p_{ij} $(i = 1,\dots,k-1; j = 1,\dots,k-n-1)$ and the q_{ij} $(i = 1,\dots,k-1; j = 1,\dots,n)$ are integers. For $i = 1,\dots,k-1$ and $j = 1,\dots,k-n-1$ we have

$$b_{ij} = p_{ij} + q_{i1}a^N_{1(n+1+j)} + q_{i2}a^N_{2(n+1+j)} + \cdots + q_{in}a^N_{n(n+1+j)}.$$

Since the b_{ij} are "small", it means that (for the above choice of the indices i,j)

$$-p_{ij} \approx q_{i1}a^N_{1(n+1+j)} + q_{i2}a^N_{2(n+1+j)} + \cdots + q_{in}a^N_{n(n+1+j)}.$$

Step 4. Compose an invertible matrix Q from the entries of the last n columns of S and use the transformation $-W^{-1}Q^{-1}$ and inverse shift for approximation.

Details and background. Take indices $i_1,\dots,i_n$ with $1 \le i_1 < \cdots < i_n \le k-1$ such that

$$Q := \begin{pmatrix} q_{i_1 1} & \cdots & q_{i_1 n} \\ \vdots & \vdots & \vdots \\ q_{i_n 1} & \cdots & q_{i_n n} \end{pmatrix}$$

is invertible. Then, recalling $\boldsymbol{a_j^N} = W(\boldsymbol{a_j} - \boldsymbol{o})$ for all $j = 1,\dots,k$, we find that

$$-W^{-1}Q^{-1}\begin{pmatrix} -q_{i_1(j-1)} \\ \vdots \\ -q_{i_n(j-1)} \end{pmatrix} + \begin{pmatrix} o_1 \\ \vdots \\ o_n \end{pmatrix} = \begin{pmatrix} a_{1j} \\ \vdots \\ a_{nj} \end{pmatrix}$$

for $j = 2,\dots,n+1$, and

$$-W^{-1}Q^{-1}\begin{pmatrix} p_{1j} \\ \vdots \\ p_{nj} \end{pmatrix} + \begin{pmatrix} o_1 \\ \vdots \\ o_n \end{pmatrix} \approx \begin{pmatrix} a_{1j} \\ \vdots \\ a_{nj} \end{pmatrix}$$

for $j = n+2,\dots,k$. This means that writing $\boldsymbol{d_i}$ for the ith column of $-W^{-1}Q^{-1}$, we get that the point $\boldsymbol{a_j}$ is just the shifted lattice point

$$\boldsymbol{o} - q_{i_1(j-1)}\boldsymbol{d_1} - q_{i_2(j-1)}\boldsymbol{d_2} - \cdots - q_{i_n(j-1)}\boldsymbol{d_n}$$

for $j = 2,\dots,n+1$, and it is close to

$$\boldsymbol{o} + p_{1j}\boldsymbol{d_1} + p_{2j}\boldsymbol{d_2} + \cdots + p_{nj}\boldsymbol{d_n}$$

for $j = n+2,\dots,k$. Recall that we also have $\boldsymbol{a_1} = \boldsymbol{o}$.

Output. The shifted origin $\boldsymbol{o}$, and the basis vectors $\boldsymbol{d_1},\dots,\boldsymbol{d_n}$ of the approximating lattice.

The resulting affine lattice $\boldsymbol{o} + \boldsymbol{d_1}\mathbb{Z} + \cdots + \boldsymbol{d_n}\mathbb{Z}$ is a well approximating lattice. How well the approximation is, depends on the skewness of the lattice transformation caused by multiplying by $(QW)^{-1}$. Until this multiplication is made, according to Theorem 4.1, the approximation is at most 2^n times the best approximation. If the result we get is not satisfactory, then it is worth to make another choice for W and Q.

We illustrate the method by some examples. In the examples we use the same norms as in Section 3, but Δ is no longer equal to $\left(\prod_{i=1}^{n} |\boldsymbol{d_i}|\right)^{1/n}$, but it still equals the absolute value of the nth root of the lattice determinant of the computed vectors $\boldsymbol{d}$. From the construction it follows that $|\det(\boldsymbol{d_1}, \dots, \boldsymbol{d_n})| \cdot |\det(QW)| = 1$. Hence $\Delta = (|\det(QW)|)^{-1/n}$.

In our first example we give a detailed description of our method.

Example 4.3. We work with the same values as in Example 3.1. We choose $n = 2$, $k = 6$ and

$$A = \{\boldsymbol{a_1} = (0.814258, 0),\ \boldsymbol{a_2} = (1.294837, \sqrt{3}),\ \boldsymbol{a_3} = (2.237840, \sqrt{5}),$$

$$\boldsymbol{a_4} = (2.764132, \sqrt{7}),\ \boldsymbol{a_5} = (4.295116, \sqrt{11}),\ \boldsymbol{a_6} = (7.733842, \sqrt{13})\}.$$

We choose an affine transformation where $\boldsymbol{a_1}$ goes to $\boldsymbol{a_1^N} = (0, 0)$, $\boldsymbol{a_6}$ goes to $\boldsymbol{a_6^N} = (1, 0)$ (these points yield the diameter), and $\boldsymbol{a_4}$ (the furthest point from the line through $\boldsymbol{a_1}$ and $\boldsymbol{a_6}$) goes to $\boldsymbol{a_4^N} = (0, 1)$. Then we get $\boldsymbol{a_2^N} = (-0.186731, 0.909125)$, $\boldsymbol{a_3^N} = (-0.052638, 0.916888)$, $\boldsymbol{a_5^N} = (0.243190, 0.922154)$. In fact, this transformation is given by $W(\boldsymbol{a} - \boldsymbol{o})$, with $\boldsymbol{o} = \boldsymbol{a_1}$ and

$$W = \begin{pmatrix} 0.234612 & -0.172905 \\ -0.319722 & 0.613595 \end{pmatrix}.$$

Choosing $\varepsilon = 10^{-3}$, we apply the LLL algorithm to the matrix

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -0.186731 & -0.052638 & 0.243190 & 0.001000 & 0 \\ 0.909125 & 0.916888 & 0.922154 & 0 & 0.001000 \end{pmatrix}.$$

Observe that we spared the columns corresponding to the vectors $\boldsymbol{a_1^N}$, $\boldsymbol{a_6^N}$ and $\boldsymbol{a_4^N}$, hence T is of type $(k-1) \times (k-1) = 5 \times 5$. We get the matrix

$$B = \begin{pmatrix} -0.027998 & 0.011434 & 0.018210 & -0.013000 & -0.028000 \\ -0.028385 & 0.024585 & -0.031923 & 0.006000 & 0.032000 \\ 0.031921 & -0.035755 & -0.002273 & -0.061000 & 0.015000 \\ -0.027917 & -0.019289 & -0.061607 & 0.024000 & -0.027000 \\ -0.034503 & -0.020212 & 0.029431 & 0.026000 & 0.068000 \end{pmatrix}.$$

Subsequently, we compute the integer matrix

$$S := BT^{-1} = \begin{pmatrix} 23 & 25 & 29 & -13 & -28 \\ -28 & -29 & -31 & 6 & 32 \\ -25 & -17 & 1 & -61 & 15 \\ 29 & 26 & 19 & 24 & -27 \\ -57 & -61 & -69 & 26 & 68 \end{pmatrix}.$$

We take the 2×2 matrix

$$Q = \begin{pmatrix} -13 & -28 \\ 6 & 32 \end{pmatrix}$$

in the upper-right corner of S. Then a basis of a well approximating lattice to the points to $\boldsymbol{a_1^N}, \ldots, \boldsymbol{a_6^N}$ is given by the column vectors of the matrix

$$-Q^{-1} = \begin{pmatrix} 0.129032 & 0.112903 \\ -0.024194 & -0.052419 \end{pmatrix}.$$

That is, we have

$$\boldsymbol{d_1'} = (0.129032, -0.024194), \quad \boldsymbol{d_2'} = (0.112903, -0.052419).$$

Recalling our choice for using the indices $1, 6, 4$ at the normalization step, as approximating points to $\boldsymbol{a_1^N}, \ldots, \boldsymbol{a_6^N}$ we obtain

$$\begin{pmatrix} 0 & 0 \\ 23 & -28 \\ 25 & -29 \\ 28 & -32 \\ 29 & -31 \\ 13 & -6 \end{pmatrix} \begin{pmatrix} \boldsymbol{d_1'} \\ \boldsymbol{d_2'} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ -0.193548 & 0.911290 \\ -0.048387 & 0.915323 \\ 0 & 1 \\ 0.241935 & 0.923387 \\ 1 & 0 \end{pmatrix}.$$

Here the entries in the second, third, and fifth rows of the matrix on the left-hand side come from the first three entries of the first two rows of S (since these are the values p_{ij} for $i = 1, 2$ and $j = 1, 2, 3$). The zeroes in the first row are clear, since $\boldsymbol{a_1^N} = (0, 0)$. Finally, the entries in the sixth and fourth line come from the identity

$$(-Q) \cdot (-Q^{-1}) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = (\boldsymbol{a_6^N}, \boldsymbol{a_4^N}).$$

We also get that besides $\boldsymbol{o} = \boldsymbol{a_1}$, a basis of a well approximating lattice to the original points $\boldsymbol{a_1}, \ldots, \boldsymbol{a_6}$ is given by the column vectors of

$$-W^{-1}Q^{-1} = \begin{pmatrix} 0.845675 & 0.679032 \\ 0.401223 & 0.268390 \end{pmatrix},$$

that is, by

$$\boldsymbol{d_1} = (0.845675, 0.401223), \quad \boldsymbol{d_2} = (0.679032, 0.268390).$$

Further, we get the following approximating points to $\boldsymbol{a_1}, \ldots, \boldsymbol{a_6}$:

$$(0.814258, 0), (1.251885, 1.713199), (2.264203, 2.247254),$$

$$(2.764132, 2.645751), (4.288839, 3.315363), (7.733842, 3.605551).$$

Using that $\Delta = 0.213242$, $\mathrm{diam}(A) = 7.802605$ we obtain

$$N_{\Lambda,\boldsymbol{o}}(A) = 2.424424, \quad N^{(2)}_{\Lambda,\boldsymbol{o}}(A) = 2.859764.$$

These values may be compared with 9.362160 and 11.045325, respectively, from Example 3.1. The greater flexibility leads to better upper bounds.

The corresponding values for $\varepsilon = 10^{-2}$ in place of 10^{-3} are $\boldsymbol{d_1} = (0.988512, 0.515079)$, $\boldsymbol{d_2} = (-5.903922, -4.706066)$, $\Delta = 1.269259$, $\mathrm{diam}(A) = 7.802605$,

$$N_{\Lambda,\boldsymbol{o}}(A) = 1.763342, \quad N^{(2)}_{\Lambda,\boldsymbol{o}}(A) = 2.851124.$$

Example 4.4. We work with the same values as in Example 3.2. We choose $n = 2$, $k = 6$, $\varepsilon = 10^{-3}$ and

$$A = \{\boldsymbol{a_1} = (0.814258, \sqrt{5}),\ \boldsymbol{a_2} = (1.294837, 0),\ \boldsymbol{a_3} = (2.237840, \sqrt{13}),$$
$$\boldsymbol{a_4} = (2.764132, \sqrt{3}),\ \boldsymbol{a_5} = (4.295116, \sqrt{11}),\ \boldsymbol{a_6} = (7.733842, \sqrt{7})\}.$$

We take the affine transformation where $\boldsymbol{a_2}$ goes to $\boldsymbol{a_2^N} = (0, 0)$, $\boldsymbol{a_6}$ goes to $\boldsymbol{a_6^N} = (1, 0)$ (these points give the diameter), and $\boldsymbol{a_3}$ (the furthest point from the line through $\boldsymbol{a_2}$ and $\boldsymbol{a_6}$) goes to $\boldsymbol{a_3^N} = (0, 1)$. The transformation is given by $W(\boldsymbol{a} - \boldsymbol{o})$ with $\boldsymbol{o} = \boldsymbol{a_2}$ and

$$W = \begin{pmatrix} 0.174003 & -0.045509 \\ -0.127683 & 0.310745 \end{pmatrix}.$$

We get $\boldsymbol{a_1^N} = (-0.185384, 0.756208)$, $\boldsymbol{a_4^N} = (0.176838, 0.350621)$, and $\boldsymbol{a_5^N} = (0.371121, 0.647538)$. We apply the LLL algorithm with $\varepsilon = 10^{-3}$ to the matrix

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -0.185384 & 0.176838 & 0.371121 & 0.00100 & 0 \\ 0.756208 & 0.350621 & 0.647538 & 0 & 0.00100 \end{pmatrix}$$

and obtain

$$\begin{pmatrix} -14 & -7 & -13 & 2 & 19 \\ 11 & -3 & -7 & 31 & -7 \end{pmatrix}$$

as the first two rows of S. By inverting the matrix

$$Q = \begin{pmatrix} 2 & 19 \\ 31 & -7 \end{pmatrix},$$

and using the inverse of W we get as a basis

$$\boldsymbol{d_1} = (-0.123227, -0.216074), \quad \boldsymbol{d_2} = (-0.199760, -0.071407)$$

of an approximating lattice to the original points. This results in the following approximating points to $\boldsymbol{a_1}, \ldots, \boldsymbol{a_6}$:

$$(0.822664, 2.239557), (1.294837, 0), (2.237840, 3.605551),$$
$$(2.756708, 1.726735), (4.295111, 3.308802), (7.733842, 2.645751).$$

The errors of approximation of A are given by

$$N_{\Lambda,\boldsymbol{o}}(A) = 0.552388, \quad N^{(2)}_{\Lambda,\boldsymbol{o}}(A) = 0.912265.$$

These values may be compared with 8.676052 and 10.236439, respectively, from Example 3.2.

To see the influence of the choice of the ε we have used our program to compare the results for $\varepsilon = 10^{-i}$ with $i = 2, 3, \ldots, 10$. In Table 1, we give a summary of the results. Notice that the lattices become smaller, but that the norms do not vary too much. To avoid rounding errors the calculations were made with 20-digit precision.

One of the aims of the project is to recognize hidden structures. In the following example we started with linear combinations with integer coefficients of $(\lg 3, \lg 7)$ and $(\lg 5, \lg 8)$ (where $\lg x$ is the logarithm of $x > 0$ to base 10) and wondered whether the algorithm finds the underlying lattice. With $\varepsilon = 10^{-4}$ we obtained the following.

TABLE 1. Basis vectors and $N^{(2)}(A)$ errors for approximating lattices for A for different values of ε.

ε	$\boldsymbol{d_1}$	$\boldsymbol{d_2}$	$N^{(2)}(A)$
10^{-2}	$(0.186184, 2.709474)$	$(0.943003, 3.605551)$	1.106647
10^{-3}	$(1.23227, 2.16074) \cdot 10^{-1}$	$(1.99760, 0.71407) \cdot 10^{-1}$	0.912265
10^{-4}	$(8.8003, 3.4581) \cdot 10^{-2}$	$(0.7410, 6.0680) \cdot 10^{-2}$	0.787361
10^{-5}	$(3.40238, 7.27944) \cdot 10^{-1}$	$(2.29256, 5.05200) \cdot 10^{-1}$	2.181773
10^{-6}	$(2.122, -1.098) \cdot 10^{-3}$	$(2.534, 1.508) \cdot 10^{-3}$	0.903954
10^{-7}	$(9.513, 7.310) \cdot 10^{-4}$	$(4.297, -0.853) \cdot 10^{-4}$	0.778563
10^{-8}	$(4.539, 2.942) \cdot 10^{-4}$	$(2.158, 2.222) \cdot 10^{-4}$	1.545291
10^{-9}	$(0.856, 3.481) \cdot 10^{-5}$	$(3.382, 0.792) \cdot 10^{-5}$	1.110116
10^{-10}	$(1.012, 1.251) \cdot 10^{-5}$	$(0.697, 0.197) \cdot 10^{-5}$	1.314036

Example 4.5. We choose $n = 2$, $k = 6$, $\varepsilon = 10^{-4}$, and

$$A = \{\boldsymbol{a_1} = (0,0),\ \boldsymbol{a_2} = (72.683692, 103.283859),$$
$$\boldsymbol{a_3} = (41.208735, 66.961502),\ \boldsymbol{a_4} = (44.746198, 62.843566),$$
$$\boldsymbol{a_5} = (51.149317, 78.204526),\ \boldsymbol{a_6} = (10.827976, 11.474991)\}.$$

We choose an affine transformation where $\boldsymbol{a_1}$ goes to $\boldsymbol{a_1^N} = (0,0)$, $\boldsymbol{a_2}$ goes to $\boldsymbol{a_2^N} = (1,0)$, $\boldsymbol{a_3}$ goes to $\boldsymbol{a_3^N} = (0,1)$. The transformation is given by $W(\boldsymbol{a} - \boldsymbol{o})$ with $\boldsymbol{o} = \boldsymbol{a_1} = (0,0)$ and

$$W = \begin{pmatrix} 0.109627 & -0.067465 \\ -0.169193 & 0.118995 \end{pmatrix}.$$

We get $\boldsymbol{a_4^N} = (0.665620, -0.088174)$, $\boldsymbol{a_5^N} = (0.331240, 0.656986)$, and $\boldsymbol{a_6^N} = (0.412873, -0.465463)$. We apply the LLL algorithm with $\varepsilon = 10^{-4}$ and find

$$\begin{pmatrix} 27 & -16 & 34 & -35 & 42 \\ 22 & 53 & -11 & -41 & -60 \end{pmatrix}$$

as the first two rows of the basis transformation matrix S. Then with the usual process we obtain

$$\boldsymbol{d_1} = (0.698970, 0.903090), \quad \boldsymbol{d_2} = (1.1760913, 1.748188)$$

as a basis for an approximating lattice for the original points. Here we recognize the approximate values $\lg 5$, $\lg 8$, $\lg 15$, and $\lg 56$. This results in the following approximating points to $\boldsymbol{a_1}, \dots, \boldsymbol{a_6}$:

$$(0,0), (72.683692, 103.283859), (41.208735, 66.961502),$$
$$(44.746198, 62.843566), (51.149316, 78.204526), (10.827977, 11.474991).$$

The errors of approximation of A are given by

$$N(A) \le N_{\Lambda,\boldsymbol{o}}(A) = 0.000086, \quad N^{(2)} \le N^{(2)}_{\Lambda,\boldsymbol{o}}(A) = 0.000125.$$

These small errors indicate that the lattice Λ is actually found. The coefficients derived from the first two rows of S are given by

$$(72.683692, 103.283859) = 35(\lg 5, \lg 8) + 41(\lg 15, \lg 56),$$
$$(41.208735, 66.961502) = -42(\lg 5, \lg 8) + 60(\lg 15, \lg 56),$$
$$(44.746198, 62.843566) = 27(\lg 5, \lg 8) + 22(\lg 15, \lg 56),$$

$$(51.149317, 78.204526) = -16(\lg 5, \lg 8) + 53(\lg 15, \lg 56),$$

$$(10.827976, 11.474991) = 34(\lg 5, \lg 8) - 11(\lg 15, \lg 56).$$

This can be reduced to the values with which we have started:

$$(72.683692, 103.283859) = 41(\lg 3, \lg 7) + 76(\lg 5, \lg 8),$$

and so on. □

5. Fine-tuning $N^{(2)}_{o,\Lambda}(A)$

The results obtained in the previous section can be improved by applying the least squares algorithm in order to find the optimal values of $\boldsymbol{o}$ and the lattice vectors for the values of the q_i's and p_i's selected after applying the LLL algorithm. As we have seen in Section 4, the underlying idea is that an "origin" $\boldsymbol{o}$ and a lattice spanned by $\boldsymbol{d_1}, \dots, \boldsymbol{d_n}$ are chosen in such a way that the point $\boldsymbol{a_j}$ is close to the lattice point

$$\boldsymbol{o} - q_{i_1(j-1)}\boldsymbol{d_1} - q_{i_2(j-1)}\boldsymbol{d_2} - \cdots - q_{i_n(j-1)}\boldsymbol{d_n}$$

for $j = 2, \dots, n+1$, and to

$$\boldsymbol{o} + p_{1j}\boldsymbol{d_1} + p_{2j}\boldsymbol{d_2} + \cdots + p_{nj}\boldsymbol{d_n}$$

for $j = n+2, \dots, k$, respectively.

The least squares method enables us to optimize $\boldsymbol{o}$, $\boldsymbol{d_1}, \dots, \boldsymbol{d_n}$ with respect to the sum of the Euclidean distances between the points and the approximating lattice points. Notice that, when applying the least squares method, inversion of the matrix Q as in Section 4 is no longer needed. We illustrate this by applying the Maple 15 procedure LeastSquares to some treated examples.

We stress that in fact this method minimizes the numerator of the main term of the norm, i.e., the expression

$$\sqrt{\sum_{\boldsymbol{a} \in A} |\boldsymbol{a} - \boldsymbol{o} - \Lambda|^2}.$$

However, since the change in the basis vectors is minimal, we expect that the norm itself improves. This is supported by the examples below, too.

Example 5.1. This is a continuation of Example 2.6. We started from

$$A = \{a_1 = 0,\ a_2 = \sqrt{3} = 1.732051,\ a_3 = \sqrt{5} = 2.236068,$$

$$a_4 = \sqrt{7} = 2.645751,\ a_5 = \sqrt{11} = 3.316625,\ a_6 = \sqrt{13} = 3.605551\}$$

and found in Example 2.6 the tuple

$$(p_1, \dots, p_6) = (0, 72, 93, 110, 138, 150).$$

We apply the least squares algorithm and find $o = 0.000695$, $d = 0.024035$. For this choice of d and o we have

$$N^{(2)}_{d,o}(A) = 0.276646$$

which is less than 0.337388 found in Example 2.6. □

Example 5.2. This is a continuation of Example 4.4. We started with

$$A = \{\boldsymbol{a_1} = (0.814258, \sqrt{5}),\ \boldsymbol{a_2} = (1.294837, 0),\ \boldsymbol{a_3} = (2.237840, \sqrt{13}),$$
$$\boldsymbol{a_4} = (2.764132, \sqrt{3}),\ \boldsymbol{a_5} = (4.295116, \sqrt{11}),\ \boldsymbol{a_6} = (7.733842, \sqrt{7})\}$$

and obtained

$$\begin{pmatrix} -14 & -7 & -13 & 2 & 19 \\ 11 & -3 & -7 & 31 & -7 \end{pmatrix}$$

as the first two rows of S. We found as a basis of an approximating lattice

$$\boldsymbol{d_1} = (-0.123227, -0.216074), \quad \boldsymbol{d_2} = (-0.199760, -0.071407)$$

and approximating points

$$(0.822664, 2.239557), (1.294837, 0), (2.237840, 3.605551),$$
$$(2.756708, 1.726735), (4.295111, 3.308802), (7.733842, 2.645751)$$

to the original $\boldsymbol{a_1}, \ldots, \boldsymbol{a_6}$. The error of approximation of A was

$$N^{(2)}_{\Lambda,\boldsymbol{o}}(A) = 0.912265.$$

After applying the least squares algorithm to the matrices

$$\begin{pmatrix} 1 & -14 & 11 \\ 1 & 0 & 0 \\ 1 & -19 & 7 \\ 1 & -7 & -3 \\ 1 & -13 & -7 \\ 1 & -2 & -31 \end{pmatrix}, \quad \begin{pmatrix} 0.814258 & 2.236068 \\ 1.294837 & 0 \\ 2.237840 & 3.605551 \\ 2.764132 & 1.732051 \\ 4.295116 & 3.316625 \\ 7.733842 & 2.645751 \end{pmatrix},$$

we obtain $\boldsymbol{o} = (1.295513, 0.000049)$ and $\boldsymbol{d_1} = (-0.123106, -0.216201)$, $\boldsymbol{d_2} = (-0.199832, -0.071509)$ and get the following approximating points to $\boldsymbol{a_1}, \ldots, \boldsymbol{a_6}$:

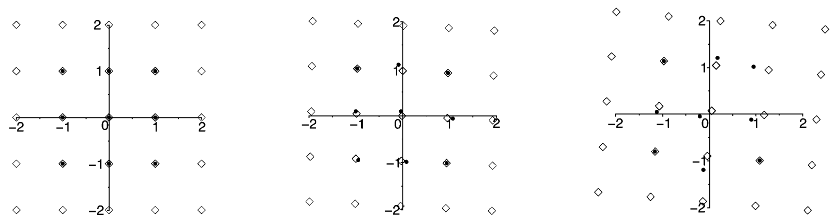
$$(0.820847, 2.240254), (1.295513, 0.000049), (2.235703, 3.607294),$$
$$(2.756749, 1.727982), (4.294710, 3.311223), (7.736504, 2.649244).$$

The new error of approximation of A is

$$N^{(2)}_{\Lambda,\boldsymbol{o}}(A) = 0.830252.$$

This value may be compared with the value 0.912265 from Example 4.4. □

Our final example demonstrates how the proposed techniques can be applied to 2D pattern analysis tasks which primarily motivated our research. Namely, let us consider the point set $A = \{-1, 0, 1\} \times \{-1, 0, 1\} \subset \mathbb{R}^2$ as a regular pattern (square grid) shown by dot symbols in Figure 1(a). Then, to have more irregular patterns, we distort A by adding some random vectors from $[-0.1, 0.1] \times [-0.1, 0.1]$ (see Figure 1(b)), and from $[-0.2, 0.2] \times [-0.2, 0.2]$ (see Figure 1(c)) to each of its points, respectively. Then, we apply our LLL-based algorithm to find well approximating lattices to these three point sets. Our method was efficient in finding well approximating lattices with the best fitting ones shown with diamond symbols in the respective subfigures. As for the practical challenge to measure pattern regularity, it is nicely observable that the best approximating lattice tends to move away with the level of distortion from the uniform square grid. Besides this intuitive shape analysis, the level of irregularity can be precisely measured by the error terms of approximation $N_{\Lambda,\boldsymbol{o}}(A)$ being $0, 0.229660, 0.441007$, respectively, and $N^{(2)}_{\Lambda,\boldsymbol{o}}(A)$ being $0, 0.382416, 0.865091$, respectively, in this example.



(a) (b) (c)

FIGURE 1. Approximation results (diamonds) for the square grid (dots) distorted by (a) 0%, (b) 10%, (c) 20% random error.

Acknowledgment

The authors are grateful to the referees for their work and useful and helpful remarks.

References

[1] W. Bosma and I. Smeets, *Finding simultaneous Diophantine approximations with prescribed quality*, ANTS X—Proceedings of the Tenth Algorithmic Number Theory Symposium, Open Book Ser., vol. 1, Math. Sci. Publ., Berkeley, CA, 2013, pp. 167–185, DOI 10.2140/obs.2013.1.167. MR3207413

[2] J. W. S. Cassels, *An introduction to Diophantine approximation*, Cambridge Tracts in Mathematics and Mathematical Physics, No. 45, Cambridge University Press, New York, 1957. MR0087708

[3] H. Cohn and N. Heninger, *Approximate common divisors via lattices*, ANTS X—Proceedings of the Tenth Algorithmic Number Theory Symposium, Open Book Ser., vol. 1, Math. Sci. Publ., Berkeley, CA, 2013, pp. 271–293, DOI 10.2140/obs.2013.1.271. MR3207418

[4] R. M. Haralick, K. Shanmugam and I. Dinstein, *Textural features for image classification*, IEEE Trans. on Systems, Man, and Cybern., **6** (1973), 610–621.

[5] N. Howgrave-Graham, *Approximate integer common divisors*, Cryptography and lattices (Providence, RI, 2001), Lecture Notes in Comput. Sci., vol. 2146, Springer, Berlin, 2001, pp. 51–66, DOI 10.1007/3-540-44670-2_6. MR1903887

[6] A. K. Lenstra, H. W. Lenstra Jr., and L. Lovász, *Factoring polynomials with rational coefficients*, Math. Ann. **261** (1982), no. 4, 515–534, DOI 10.1007/BF01457454. MR682664

[7] W. M. Schmidt, *Diophantine Approximation*, Lecture Notes in Mathematics, vol. 785, Springer, Berlin, 1980. MR568710

[8] L. G. Shapiro and G. C. Stockman, *Computer Vision*, Prentice Hall, Upper Saddle River, 2001.

[9] W. Steurer and S. Deloudi, *Crystallography of Quasicrystals: Concepts, Methods and Structures*, Springer, Heidelberg, 2009.

Faculty of Informatics, University of Debrecen, P.O. Box 12, H-4010 Debrecen, Hungary

Email address: hajdu.andras@inf.unideb.hu

Institute of Mathematics, University of Debrecen, P.O. Box 12, H-4010 Debrecen, Hungary

Email address: hajdul@science.unideb.hu

Mathematical Institute, Leiden University, Postbus 9512, 2300 RA Leiden, The Netherlands

Email address: tijdeman@math.leidenuniv.nl