



Deterministic Constructions of High-Dimensional Sets with Small Dispersion

Mario Ullrich^{1,2} · Jan Vybíral³

Received: 15 March 2020 / Accepted: 28 January 2022 / Published online: 24 February 2022
© The Author(s) 2022

Abstract

The dispersion of a point set $P \subset [0, 1]^d$ is the volume of the largest box with sides parallel to the coordinate axes, which does not intersect P . It was observed only recently that, for any $\varepsilon > 0$, certain randomized constructions provide point sets with dispersion smaller than ε and number of elements growing only logarithmically in d . Based on deep results from coding theory, we present explicit, deterministic algorithms to construct such point sets in time that is only polynomial in d . Note that, however, the running-time will be super-exponential in ε^{-1} . Our construction is based on the apparently new insight that low-dispersion point sets can be deduced from solutions of certain k -restriction problems, which are well-known in coding theory.

Keywords Dispersion · High dimensions · Explicit construction · Error-correcting codes

We would like to express our gratitude to the Erwin Schrödinger International Institute for Mathematics and Physics for its hospitality during the programme on “Tractability of High Dimensional Problems and Discrepancy”, where some part of this research was carried out. We also gratefully acknowledge the support of the Oberwolfach Research Institute for Mathematics, where initial discussion were held during the workshop “Perspectives in High-Dimensional Probability and Convexity”. We also thank Michael Gnewuch, Daniel Král’ and Hemant Tyagi for fruitful discussions. Moreover, we thank the referee for several constructive comments, including reference [36] which led to an improvement of the main result. JV was supported by the grant P201/18/00580S of the Grant Agency of the Czech Republic and by the European Regional Development Fund-Project “Center for Advanced Applied Science” (No. CZ.02.1.01/0.0/0.0/16_019/0000778).

✉ Mario Ullrich
mario.ullrich@jku.at

Jan Vybíral
jan.vybiral@fjfi.cvut.cz

¹ Institut für Analysis, Johannes Kepler Universität Linz, Altenbergerstr. 69, 4040 Linz, Austria

² Moscow Center for Fundamental and Applied Mathematics, Lomonosov Moscow State University, Moscow, Russia

³ Department of Mathematics, Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University, Trojanova 13, 12000 Praha, Czech Republic

1 Introduction and Results

For $d \in \mathbb{N}$ and a point set $P \subset [0, 1]^d$ we define the *dispersion of P* by

$$\text{disp}(P) := \sup_{B: B \cap P = \emptyset} |B|,$$

where the supremum is over all axis-parallel boxes $B = I_1 \times \cdots \times I_d$ with intervals $I_\ell \subset [0, 1]$, and $|B|$ denotes the (Lebesgue) volume of B . As we are interested in point sets which make the above quantity as small as possible, we additionally define, for $n, d \in \mathbb{N}$, the *n -th-minimal dispersion*

$$\text{disp}(n, d) := \inf_{\substack{P \subset [0, 1]^d: \\ \#P = n}} \text{disp}(P),$$

where $\#P$ denotes the cardinality of the set P . For $\varepsilon \in (0, 1)$, we also introduce

$$N(\varepsilon, d) := \min \left\{ n : \text{disp}(n, d) \leq \varepsilon \right\}.$$

i.e., the minimal cardinality of a point set $P \subset [0, 1]^d$ that has dispersion smaller than ε .

Besides the fact that the above geometric quantities are interesting in its own right, they also attracted attention in the numerical analysis community, especially when it comes to very high dimensional applications. The reason is that bounds on the (minimal) dispersion lead to bounds on worst-case errors, and hence the complexity, for some numerical problems, including optimization and approximation in various settings, see [5, 24, 31, 34, 38, 42, 43, 46]. This is a similar situation as for the much more studied *discrepancy*, which corresponds to certain numerical integration problems, see e.g. [9, 10, 13, 32, 33, 35].

Moreover, bounding the dispersion is clearly also related to the problem of *finding the largest empty box*. In dimension two, this is the *Maximum Empty Rectangle Problem*, which is one of the oldest problems in computational geometry. For the state of the art and further references we refer to [8, 14–17, 28].

Regarding bounds on the inverse of the minimal dispersion, we have

$$\frac{\log_2(d)}{8\varepsilon} \leq N(\varepsilon, d) \leq \frac{C^d}{\varepsilon}$$

for some $C < \infty$ and all $\varepsilon < 1/8$, see [1], where the upper bound is attained by certain digital nets. See also [15, 38] for a related upper bound. Although these bounds show the correct dependence on ε , they are rather bad with respect to the dimension d . This gap was narrowed in the past years by several authors, see e.g. [23, 25, 39, 41, 45], including the important work of Sosnovec who proved that the logarithmic dependence

on d is optimal. With this respect, the best bound at present is

$$\frac{\log_2(d)}{8\varepsilon} \leq N(\varepsilon, d) \leq 2^7 \log_2(d) \frac{(1 + \log_2(\varepsilon^{-1}))^2}{\varepsilon^2}, \tag{1.1}$$

see [45] and Theorem 4.1 below. Note that the logarithmic dependence is special for the cube, as it is known that, for the same problem on the torus, we have a lower bound linear in d , see [44]. Depending on the relation between d and ε^{-1} , (1.1) can sometimes be improved. We refer to [7, 26] for recent results in this direction as well as for further references. Let us stress, that we are mainly interested in the regime, where d is (much) larger than ε^{-1} and where the logarithmic dependence on d is crucial.

The main drawback of most of these results is that they only show the existence of point sets with small dispersion. Often, such point sets are at the core of approximation algorithms and are the only non-constructive ingredient in the corresponding proofs, see e.g. [5, 34, 42, 43] and references therein (in particular, [34,Thm. 11] and [43,Thm. 2.2]). Hence, explicit, deterministic constructions for point sets lead directly to “constructive” algorithms for certain high-dimensional approximation problems.

For example, the bound from [34,Thm. 11] on the minimal number of function evaluations needed for uniform recovery of certain (tensor-product) functions is based on a random construction, and is therefore only an existence result. If we plug in our deterministic construction, and note that the “dispersion part” of the upper bound is insignificant compared to the “approximation part”, when it comes to high dimensions, we see that we can achieve basically the same bound, but in a purely constructive way.

In this paper, we consider explicit constructions of point sets with small dispersion, i.e., deterministic algorithms, which for given integer $d \geq 2$ and positive parameter $\varepsilon > 0$ construct a point set $P \subset [0, 1]^d$ with dispersion at most ε . The quality of the algorithm is measured by the cardinality of the output set P and by its running time, i.e., by the number of algebraic operations on real numbers executed by the algorithm. Concerning such explicit constructions, we are only aware of the above mentioned digital nets, which lead to a bad d -dependence, and sparse grids, which satisfy the upper bound $N(\varepsilon, d) \leq (2d)^{\log_2(1/\varepsilon)}$, see [23]. It is already clear from the number of points, that both do not lead to constructions of point sets with small dispersion that can be carried out in time that is polynomial in d .

Moreover, as the existence proofs are based on random points on a finite grid, one could use the *naïve algorithm*: try each of the possible configurations, calculate its dispersion (if possible) and output a set that satisfies the requested bound. The running-time of this algorithm is (in the worst case) clearly at least exponential in d .

Under the name *Small Hitting Sets*, a more involved construction from [25] leads to point sets of size $\text{poly}(\frac{d \log(d)}{\varepsilon})$, and the running-time of the presented algorithm is polynomial in d and ε^{-1} . (Note that the additional parameter in [25] has to be set to $m = d/\varepsilon$, see [25,p. 217].) Hence, there is a *fully polynomial-time algorithm* for the dispersion, if the parameters are d and ε^{-1} .

Reconsidering the bound (1.1), one may think, however, that $\log(d)$ (instead of d) would be an appropriate parameter for the problem complexity. Here, we show, using deep results from the theory of error-correcting codes, that point sets with dispersion

at most $\varepsilon > 0$ and size of the order $\log(d) \cdot \text{poly}(1/\varepsilon)$ can be constructed in time that is polynomial in d . Unfortunately, we do not have a good control on the dependence on ε^{-1} , and it remains an open problem to find a construction that is also polynomial in ε^{-1} .

Our two main results are derandomized versions of the results from [41, 45]. Both use different approaches and lead to somewhat different results. The first one, as discussed in Sect. 3, leads to point sets of size $\mathcal{O}_\varepsilon(\log d)$ that can be constructed in time linear in the size of the output, i.e., in time $\mathcal{O}_\varepsilon(d \log d)$, see Algorithm 1 and Theorem 3.3 in Sect. 3.3. Here, and in the following, $\mathcal{O}_\varepsilon(\log d)$ means that the implied constant depends on ε in an unspecified way. A second, and much more involved, construction will be given by Algorithm 2 in Sect. 4.4. The corresponding result reads as follows.

Theorem 4.4 *There is an absolute constant $C < \infty$ such that, for all $\varepsilon \in (0, \frac{1}{4}]$ and $d \geq 2$, Algorithm 2 constructs a set $P \subset [0, 1]^d$ with $\text{disp}(P) \leq \varepsilon$ and*

$$\#P \leq C \left(\frac{1 + \log_2(\varepsilon^{-1})}{\varepsilon} \right)^4 \log(d^*),$$

where $d^* := \max(d, 2/\varepsilon)$. The running-time of Algorithm 2 is $\mathcal{O}_\varepsilon(d \log(d))$.

Note that, in contrast to Algorithm 1, the point set that is constructed by Algorithm 2 has size that is polynomial in ε^{-1} . However, as the proof shows, its computational cost is at least exponential in ε^{-1} .

Our general approach is as following. We start with a detailed inspection of the random constructions of point sets with small dispersion. This will allow us to clearly separate the setting of the construction and its randomized part. It will turn out then, which properties are crucial for each of the approaches and what exactly is the role of randomness. Afterwards, we replace the randomized part by a deterministic one.

Remark 1.1 Regarding the discrepancy of point sets, it is known that the decision problem, if a given point set has *discrepancy* smaller than ε , is known to be NP-hard, see [20] or [12, Section 3.3], and the same is true for the dispersion, if the dimension d is part of the input, cf. [6]. For upper bounds on the cost of constructing points with small discrepancy and further literature, see [11, 12, 18, 19]. However, all known algorithms for this problem so far have running-time exponential in d , or do not lead to a point set of size polynomial in d and ε^{-1} . We hope that the results of this paper will lead to some progress also for this problem.

2 Basics from Coding Theory

Our deterministic constructions of point sets with small dispersion will be essentially obtained by certain “derandomization” of recent proofs from [41, 45]. As we will rely on rather deep tools from coding theory, we summarize in this section the necessary definitions and results for later use.

2.1 Universal Sets

First, we introduce the concept of (n, k) -universal sets, which is also known in coding theory under the name of t -independent set problem. It has its roots in testing of logical circuits [40].

Definition 2.1 ((n, k) -universal sets) Let $1 \leq k \leq n$ be positive integers. We say that $T \subset \{0, 1\}^n$ is an (n, k) -universal set, if for every index set $S \subset \{1, \dots, n\}$ with $\#S = k$, the projection of T on S contains all possible 2^k configurations.

Naturally one is interested in (randomized and deterministic) constructions of small (n, k) -universal sets. The straightforward randomized construction provides the existence of an (n, k) -universal set of size $\lceil k2^k \log(n) \rceil$. On the other hand, [22] gives a lower bound on the size of an (n, k) -universal set of the order $\Omega(2^k \log(n))$. There exist several deterministic constructions of (n, k) -universal sets in the literature (cf. [2, 3, 29]) and we shall rely on the results given in [30].

Theorem 2.2 [30, Theorem 6] *There is a deterministic construction of an (n, k) -universal set of size $2^k k^{O(\log(k))} \log(n) = 2^{k+O(\log^2(k))} \log(n)$, which can be listed in linear time of the length of the output.*

Although the notion of an (n, k) -universal set is not very flexible and comes from a different area of mathematics, we will see in Sect. 3, that there is indeed a link to sets with small dispersion. Reusing the known results from coding theory, it will already allow us to obtain our first deterministic construction of a point set with cardinality of order $\log(d)$. However, in this approach we have only very limited control of the dependence on ε .

We also need the following natural generalization of (n, k) -universal sets. We could not find this concept in the literature, but we assume that this and the proceeding lemma are known.

Definition 2.3 ((n, k, b) -universal sets) Let $1 \leq k \leq n$ and $b \geq 2$ be positive integers. We say that $T \subset \{0, 1, \dots, b - 1\}^n$ is an (n, k, b) -universal set, if for every index set $S \subset \{1, \dots, n\}$ with $\#S = k$, the projection of T on S contains all possible b^k configurations.

If $b = 2$, Definitions 2.1 and 2.3 coincide, i.e. $(n, k, 2)$ -universal sets are just the usual (n, k) -universal sets. We use the following two observations to transfer the known results about (n, k) -universal sets to our setting.

Lemma 2.4 *Let $1 \leq k \leq n$ and $b \geq 2$ be positive integers.*

- (i) *Let $T \subset \{0, 1, \dots, b\}^n$ be an $(n, k, b + 1)$ -universal set. Then there is an (n, k, b) -universal set $T' \subset \{0, 1, \dots, b - 1\}^n$ of at most the same size.*
- (ii) *Let $m \in \mathbb{N}$ and $T \subset \{0, 1\}^{mn}$ be an (mn, mk) -universal set. Then there is an $(n, k, 2^m)$ -universal set of the same size.*

Proof The proof is quite straightforward. To show (i), just replace all occurrences of b among the coordinates of T by zero. For the proof of the second part it is enough to interpret each $x \in T \subset \{0, 1\}^{mn}$ as a digital representation of $\tilde{x} \in \{0, 1, \dots, 2^m - 1\}^n$. □

The direct random construction yields the existence of an (n, k, b) -universal set of the size $\lceil kb^k \log(ebn/k) \rceil$. Using Theorem 2.2, we can easily obtain a deterministic construction of an (n, k, b) -universal set of only a slightly larger size.

Theorem 2.5 *There is a deterministic construction of an $(n, k, 2^m - 1)$ -universal set of size $2^{mk+O(\log^2(mk))} \log(n)$, which can be listed in linear time of the length of the output.*

Proof By Theorem 2.2, there is a construction of an (mn, mk) -universal set with at most $2^{mk+O(\log^2(mk))} \log(mn)$ elements. The result then follows from Lemma 2.4. \square

2.2 k -Restriction Problems

For the derandomization of the analysis of [45] we need a more flexible notion of the so-called k -restriction problems, see [30, Section 2.2]. Solutions to these problems will be one of the building blocks of our deterministic construction of sets with small dispersion whose size is polynomial in $1/\varepsilon$ and, still, logarithmic in d .

Definition 2.6 (k -restriction problems) Let b, k, n, M be positive integers and let $\mathcal{C} = \{C_1, \dots, C_M : C_i \subset \{0, 1, \dots, b-1\}^k\}$ be invariant under the permutations of the index set $\{1, \dots, k\}$, i.e., for every $i \in \{1, \dots, M\}$ and every permutation π on $\{1, \dots, k\}$, there is $j \in \{1, \dots, M\}$, such that $C_j = \{x \circ \pi : x \in C_i\}$. We say that $T = \{x^1, \dots, x^N\} \subset \{0, 1, \dots, b-1\}^n$ satisfies the k -restriction problem with respect to \mathcal{C} , if for every $S \subset \{1, \dots, n\}$ with $\#S = k$ and for every $j \in \{1, \dots, M\}$, there exists $x^\ell \in T$ with $x^\ell|_S \in C_j$.

Definitions 2.1 and 2.3 are indeed special cases of Definition 2.6. To show this, let us choose $M = b^k$ and let C_1, \dots, C_M be all the different singleton subsets of $\{0, \dots, b-1\}^k$. Then, T satisfies the k -restriction problem with respect to \mathcal{C} if for every index set $S \subset \{1, \dots, n\}$ and every possible $v \in \{0, \dots, b-1\}^k$ there exists an $x \in T$ with $x|_S = v$, i.e. if the restriction of T to every index set S with k elements attains all b^k possible values.

An important parameter of the k -restriction problems is the minimal size of each of the restriction sets C_j , i.e.

$$c = c(\mathcal{C}) := \min_{1 \leq j \leq M} \#C_j. \quad (2.1)$$

Random constructions of sets satisfying the k -restriction problem with parameters (b, k, n, M) and $\mathcal{C} = \{C_1, \dots, C_M\}$ are based on a simple union bound. Indeed, let $1 \leq j \leq M$ and $S \subset \{1, \dots, n\}$ with $\#S = k$ be fixed. The probability, that a randomly chosen vector $v \in \{0, 1, \dots, b-1\}^n$ satisfies C_j on S is at least

$$\mathbb{P}(v \text{ satisfies } C_j \text{ at } S) = \frac{\#C_j}{b^k} \geq \frac{c}{b^k}.$$

If we choose N random vectors v_1, \dots, v_N independently, the probability that none of them satisfies C_j at S is at most

$$\mathbb{P}(\text{no } v_1, \dots, v_N \text{ satisfies } C_j \text{ at } S) = \left(1 - \frac{\#C_j}{b^k}\right)^N \leq \left(1 - \frac{c}{b^k}\right)^N.$$

Finally, the probability that there is a set $S \subset \{1, \dots, n\}$ with $\#S = k$ and $1 \leq j \leq M$, such that no v_1, \dots, v_N satisfies C_j at S is, by the union bound, at most

$$\binom{n}{k} \cdot M \cdot \left(1 - \frac{c}{b^k}\right)^N.$$

This expression is smaller than one if

$$N > \frac{\log(n^k M)}{\log\left(\frac{b^k}{b^k - c}\right)}.$$

This means that there exist solutions to a k -restriction problem with parameters (b, k, n, M) of size N whenever

$$N \geq \left\lceil \frac{b^k}{c} \log(n^k M) \right\rceil,$$

where c is from (2.1). Theorem 1 of [30] states that there is a deterministic algorithm that outputs such a solution of size equaling the union bound. The main idea of its proof is that the random sampling can be replaced by an extensive search through a k -wise independent probability space with n random variables with values in $\{1, \dots, b\}$.

Theorem 2.7 [30, Theorem 1] *For any k -restriction problem with parameters (b, k, n, M) with $b \leq n$, there is a deterministic algorithm that outputs a collection obeying the k -restrictions, with the size of the collection equaling*

$$\left\lceil \frac{b^k}{c} \log(n^k M) \right\rceil,$$

where c is from (2.1). The time taken to output the collection is

$$\mathcal{O}\left(\frac{b^k}{c} \left(\frac{en^2}{k}\right)^k M \mathcal{T}\right),$$

where \mathcal{T} is the time complexity of the membership oracle, i.e., a procedure which, for given $v \in \{1, \dots, b\}^n$, $S \subset \{1, \dots, n\}$ with $\#S = k$ and $j \in \{1, \dots, M\}$, outputs if the restriction of v on S belongs to C_j .

In what follows it can be executed in $\mathcal{T} = O(k)$ time. Also note that the formulation of Theorem 2.7 follows from Theorem 1 in [30] by using the bound on the size

of k -wise independent probability space with n random variables as described in [30,Section 2.1].

2.3 Splitters

The last ingredient of our derandomization procedure are splitters. They played a central role in [30] as the basic building blocks of all deterministic constructions given there. Essentially, they allow to split a large problem into smaller problems which can then be treated by the extensive search of Theorem 2.7.

Definition 2.8 ((n, k, ℓ) -splitter) Let n, k, ℓ be positive integers. An (n, k, ℓ) -splitter H is a family of functions from $\{1, \dots, n\}$ to $\{1, \dots, \ell\}$, such that for every $S \subset \{1, \dots, n\}$ with $\#S = k$ there is $h \in H$, which splits S perfectly. It means that the sets $h^{-1}(\{j\}) \cap S$ are of the same size for all $j \in \{1, \dots, \ell\}$ (or as similar as possible if $\ell \nmid k$).

Similarly to [4, 30], we will rely on (n, k, k^2) -splitters. By Definition 2.8, $A(n, k)$ is an (n, k, k^2) -splitter, if it is a collection of mappings $a : \{1, \dots, n\} \rightarrow \{1, \dots, k^2\}$ such that for every $S \subset \{1, \dots, n\}$ with $\#S = k$, there is an $a \in A(n, k)$, which is injective on S . Moreover, note that every (n, k, ℓ) -splitter with $k \leq \ell \leq k^2$ is also a (n, k, k^2) -splitter.

Small (n, k, ℓ) -splitters with $\ell \geq \eta k^2$ for $\eta > \frac{1}{2}$ can be obtained from asymptotically good error correcting codes, see [4, Lemma 3]. Indeed, let $c_1, \dots, c_n \in \{1, \dots, \ell\}^L$ denote the codewords of an error correcting code of size n over the alphabet $\{1, \dots, \ell\}$ and normalized Hamming distance at least $1 - \frac{2\eta}{\ell}$. This is, c_i and c_j ($i \neq j$) can be equal on at most $2\eta L/\ell$ coordinates. If now $S \subset \{1, \dots, n\}$ with $\#S = k$, then there are $k(k-1)/2$ pairs (i, j) with $i \neq j \in S$. As $\frac{2\eta L}{\ell} \cdot \frac{k(k-1)}{2} < L$, there must be a coordinate j , where all the $c_1(j), \dots, c_L(j)$ are different. Finally, if we consider the mappings $h_j : i \rightarrow c_i(j), i = 1, \dots, n$, we observe that $H = \{h_j : j = 1, \dots, L\}$ is an (n, k, ℓ) -splitter, and hence an (n, k, k^2) -splitter, of size L .

The existence of such explicit codes for $k = 1$ is trivial and they exist by [36] with $L = \mathcal{O}(k^2 \log n)$ for $k \geq 2$. For this, let q be a prime with $\frac{3k^2}{4} \leq q \leq k^2$. Such a q exists by classical estimates on the prime counting function (cf. [37]) for $k \geq 23$ and for $2 \leq k \leq 22$ its existence can be checked directly. From Theorem 2 of [36] we obtain that one can construct a (linear) error correcting code over the alphabet $\{1, \dots, q\}$ of size n , length L and normalized Hamming distance at least $\delta \in [0, 1]$, whenever

$$L \geq (1 - H_q(\delta))^{-1} \log_q(n),$$

where $H_q(\delta) := -\delta \log_q(\delta) - (1 - \delta) \log_q(1 - \delta) + \delta \log_q(q - 1)$ is the q -ary entropy function. (Such a code is called an $[L, \log_q(n), \delta L]_q$ -LC in [36].) The running-time of the proposed algorithm is $\mathcal{O}(L \cdot n)$.

Setting $\delta = 1 - \frac{3}{2q}$, we obtain by simple computations that $1 - H_q(\delta) \geq \frac{1}{10q \log(q)}$, and hence that we can choose $L = \mathcal{O}(q \log n) = \mathcal{O}(k^2 \log n)$.

The following lemma summarizes the discussion above and shows that (n, k, k^2) -splitters of relatively small size can be constructed explicitly in polynomial time.

Lemma 2.9 (cf. [4, 36]) *There is an explicit (n, k, k^2) -splitter of size*

$$\mathcal{O}(k^2 \log(n))$$

that can be constructed in time $\mathcal{O}(k^2 n \log n)$.

The (n, k, k^2) -splitters can be used whenever n is “very large” compared to k . Roughly speaking, and in the context of the present paper, we will transform a k -restriction problem of size n to a k^2 -restriction problem of size k^2 , which can then be solved using the results of Sect. 2.2. This will lead to construction algorithms with an apparently optimal dependence of their running time on the original problem size n . This approach was already used to prove [30, Theorem 6], see Theorem 2.5.

Remark 2.10 In a previous version of this manuscript we used the explicit construction of [3], which leads to the bound $L = \mathcal{O}(k^4 \log n)$. As above, we obtained (n, k, k^2) -splitters that can be constructed in polynomial time, see also [30, Thm. 3(iv)]. The used code is a two-fold concatenation code that combines the Wozencraft ensemble, Justesen codes and expander codes, which in turn rely on famous deterministic constructions of expander graphs [27]. For the details we refer to Sects. 3 and 4 of [3].

3 Derandomization of Sosnovec’s Proof

First, we consider the construction of Sosnovec [41], which gives logarithmic dependence of $N(\varepsilon, d)$ on d but involves no special control of its dependence on ε^{-1} . His main theorem was the following.

Theorem 3.1 [41, Theorem 2] *For every $\varepsilon \in (0, \frac{1}{4})$, there exists a constant $c_\varepsilon > 0$, such that for every $d \geq 2$ there is a point set $P \subset [0, 1]^d$ with $\text{disp}(P) \leq \varepsilon$ and*

$$\#P \leq c_\varepsilon \log(d).$$

We will see that this result can be essentially derandomized using results from coding theory, while losing only a negligible factor. The drawback of this approach is the extremely bad dependence of c_ε on ε . We sketch the main ideas of Sosnovec’s proof.

3.1 Sosnovec’s Proof: The Setting

For an integer $m \geq 2$ with $2^{-m} \leq \varepsilon < 2^{-m+1}$, we define

$$\mathbb{M}_m = \left\{ \frac{1}{2^m}, \dots, \frac{2^m - 1}{2^m} \right\}.$$

The point set constructed will be a subset of \mathbb{M}_m^d . Furthermore, we define

$$\Omega_m := \left\{ \mathbb{B} = I_1 \times \cdots \times I_d \subset [0, 1]^d : \text{vol}(\mathbb{B}) > \frac{1}{2^m} \right\} \tag{3.1}$$

to be the set of all boxes with sides parallel to the coordinate axes and volume larger than 2^{-m} .

Let $\mathbb{B} = I_1 \times \cdots \times I_d \in \Omega_m$. The key observation of [41] is that the number of indices $j \in \{1, \dots, d\}$ with $\mathbb{M}_m \not\subset I_j$ is bounded from above by $m2^m$, a quantity independent on d . To be more specific, if we denote

$$\mathcal{A}(\mathbb{B}) = \{j \in \{1, \dots, d\} : \mathbb{M}_m \not\subset I_j\},$$

Then $\#\mathcal{A}(\mathbb{B}) \leq A_m := \min(m2^m, d)$ for every $\mathbb{B} \in \Omega_m$. We will refer to $\mathcal{A}(\mathbb{B})$ as the set of “active indices” of \mathbb{B} . If $\mathcal{A}(\mathbb{B})$ is not of the full possible size, we enlarge it by adding any of the other indices to obtain a set with cardinality equal to A_m . Therefore, we can associate to each $\mathbb{B} \in \Omega_m$ (possibly in a non-unique way) a set \mathcal{A} with $\#\mathcal{A} = A_m$ and a vector $z \in \mathbb{M}_m^{A_m}$ such that any $x \in \mathbb{M}_m^d$ with $x|_{\mathcal{A}} = z$ lies in \mathbb{B} .

Vice versa, if we have a point set $P = \{x^1, \dots, x^N\} \subset \mathbb{M}_m^d$, such that for every $\mathcal{A} \subset \{1, \dots, d\}$ with $\#\mathcal{A} = A_m$ and to every $z \in \mathbb{M}_m^{A_m}$, there is some $x^j \in P$ with $x^j|_{\mathcal{A}} = z$, then, by what we just said, P intersects every $\mathbb{B} \in \Omega_m$. Therefore, the dispersion of P can not be larger than 2^{-m} , i.e., $\text{disp}(P) \leq 2^{-m}$ and hence $N(2^{-m}, d) \leq N$.

To simplify the combinatorial part later on, we multiply all coordinates by 2^m , which results to vectors with integer components. This motivates the following definition.

Definition 3.2 Let $m \geq 2$. We say that $T = \{x^1, \dots, x^N\} \subset \{1, 2, \dots, 2^m - 1\}^d$ satisfies the condition (S) of the order m if for every $\mathcal{A} \subset \{1, \dots, d\}$ with $\#\mathcal{A} = A_m := \min(m2^m, d)$, the set of restrictions $x^1|_{\mathcal{A}}, \dots, x^N|_{\mathcal{A}}$ contains all $(2^m - 1)^{A_m}$ possible values.

By what we said above, anytime $m \geq 2$ and $T = \{x^1, \dots, x^N\}$ satisfies the condition (S) of the order m , then $P \subset \mathbb{M}_m^d$ with $P = 2^{-m} \cdot T$ satisfies $\text{disp}(P) \leq 2^{-m}$. The proof of Theorem 3.1 is therefore finished, once we find a set T with $\#T \leq c_m \log(d)$ satisfying the condition (S) of order m .

3.2 Sosnovec’s Proof: Randomized Construction

The rest of the proof in [41] can now be understood as a randomized construction of a small set satisfying the condition (S). Indeed, there are $\binom{d}{A_m}$ subsets S of $\{1, \dots, d\}$ with A_m elements. We now fix one such set S and one vector $z \in \{1, \dots, 2^m - 1\}^{A_m}$. The probability that a point $x \in \{1, \dots, 2^m - 1\}^d$ chosen at random (from the uniform distribution) fulfills $x|_S = z$ is $(2^m - 1)^{-A_m}$. Therefore, the probability that none of the N randomly chosen points x^1, \dots, x^N fulfills this restriction is $[1 - (2^m - 1)^{-A_m}]^N$. Finally, the probability that there is a set $S \subset \{1, \dots, d\}$ with $\#S = A_m$ and a vector

$z \in \{1, \dots, 2^m - 1\}^{A_m}$ such that no $\{x^1, \dots, x^N\}$ satisfies $x^j|_S = z$ is, by the union bound, at most

$$\binom{d}{A_m} (2^m - 1)^{A_m} [1 - (2^m - 1)^{-A_m}]^N.$$

By simple calculus, if N is of the order $c_m \log_2(d)$ for c_m large enough, this expression is smaller than one. Hence, with positive probability, the randomly chosen point set $\{x^1, \dots, x^N\}$ satisfies the condition (S).

3.3 Derandomization Using Universal Sets

Definition 3.2 resembles very much the concept of (n, k) -universal sets, see Sect. 2.1. In particular, it is easy to see that every $(d, A_m, 2^m - 1)$ -universal set -after adding 1 to each coordinate- satisfies condition (S) of the order m .

Therefore, we can use Theorem 2.5 to replace the random arguments of the last section by a deterministic algorithm. We glue all the components together in the form of an algorithm.

Algorithm 1

1. For $\varepsilon \in (0, \frac{1}{4}]$ and $d \geq 2$, choose a positive integer m with $2^{-m} \leq \varepsilon < 2^{-m+1}$ and set $A_m := \min(m2^m, d)$;
2. Generate an (md, mA_m) -universal set as in [30, Theorem 6];
3. Interpret these vectors as digital decompositions to obtain an $(d, A_m, 2^m)$ -universal set;
4. Replace $2^m - 1$ by 0 in all coordinates, increase all the coordinates by one (this set satisfies (S)), divide all the coordinates by 2^m and output the point set.

It remains to consider the running-time of this algorithm.

Theorem 3.3 *Let $\varepsilon \in (0, \frac{1}{4}]$ and $d \geq 2$. Then there is a positive constant $c_\varepsilon > 0$, such that Algorithm 1 constructs a set $P \subset [0, 1]^d$ with $\text{disp}(P) \leq \varepsilon$ and*

$$\#P \leq c_\varepsilon \log_2(d).$$

The running time of Algorithm 1 is linear in the length of the output.

Proof The first part of the theorem is proven by what we said above. Concerning the running-time, we obtain from Theorem 2.5, that the $(d, A_m, 2^m - 1)$ -universal set, which we generate in steps (2)–(4) of Algorithm 1, can be constructed in linear time and with size

$$N = 2^{mA_m + O(\log^2(mA_m))} \log(d) \leq 2^{m^2 2^m + O(m^2)} \log_2(d).$$

The remaining operations can be done in a linear time, without enlarging the point set. \square

As expected, the dependence of the size of P on $2^m \approx \varepsilon^{-1}$ is rather bad (as it was in [41]), but there is indeed only a logarithmic dependence on d .

4 Improving the Dependence in ε^{-1}

The main aim of [45] was to refine the analysis of [41] and to achieve a better dependence of $N(\varepsilon, d)$ on ε , without sacrificing the logarithmic dependence on d . The main theorem of [45] was the following.

Theorem 4.1 *Let $d \geq 2$ be a natural number and let $\varepsilon \in (0, 1/2)$. Then there exists a point set $P \subset [0, 1]^d$ with $\text{disp}(P) \leq \varepsilon$ and*

$$\#P \leq 2^7 \log_2(d) \frac{(1 + \log_2(\varepsilon^{-1}))^2}{\varepsilon^2}.$$

Also this result can be derandomized using results from coding theory. By doing this, we will lose some power of $1/\varepsilon$ in the size of the point set. However, it will still be of order $\log(d)$.

4.1 Enhanced Analysis of the Random Construction

The main novelty of [45] was a more careful splitting of Ω_m (see (3.1)) into subgroups. To be more specific (and using the notation of [21]), for $s = (s_1, \dots, s_d) \in \{1, \dots, 2^m - 1\}^d$ and $p = (p_1, \dots, p_d) \in \mathbb{M}_m^d$, we denoted $\Omega_m(s, p)$ to be those cubes from Ω_m , which have I_ℓ approximately of the length $\frac{s_\ell}{2^m}$ and its left point around p_ℓ for all $\ell = 1, \dots, d$, i.e.

$$\Omega_m(s, p) := \left\{ \mathbb{B} = I_1 \times \dots \times I_d \in \Omega_m : \forall \ell \in \{1, \dots, d\} : \frac{s_\ell}{2^m} < \text{vol}(I_\ell) \leq \frac{s_\ell + 1}{2^m} \right. \\ \left. \text{and } \inf I_\ell \in \left[p_\ell - \frac{1}{2^m}, p_\ell \right) \right\}.$$

We denote by \mathbb{I}_m the pairs (s, p) , for which $\Omega_m(s, p)$ is non-empty. It is easy to see that their number is bounded from above by

$$\#\mathbb{I}_m \leq \exp\left(m2^m \log(2^{m+3}d)\right), \tag{4.1}$$

see [21, Eq. (2.1)]. If $(s, p) \in \mathbb{I}_m$, we further set

$$\mathbb{B}_m(s, p) := \bigcap_{\mathbb{B} \in \Omega_m(s, p)} \mathbb{B} = \prod_{\ell=1}^d \left[p_\ell, p_\ell + \frac{s_\ell - 1}{2^m} \right]. \tag{4.2}$$

The advantage of dividing of Ω_m into the groups $\Omega_m(s, p)$ is the surprisingly good control of the probability that a randomly chosen point $z \in \mathbb{M}_m^d$ lies in the intersection of all the cubes from $\Omega_m(s, p)$. It is actually, up to a constant, of the same order as the volume of each of the cubes in Ω_m , i.e. of 2^{-m} .

Lemma 4.2 ([45, Lemma 3] and [21, Lemma 2.1]) *Let $m \in \mathbb{N}$, $s \in \{0, 1, \dots, 2^m - 1\}^d$ and $p \in \{1/2^m, \dots, 1 - 1/2^m\}^d$ be such that $(s, p) \in \mathbb{I}_m$. Let z be uniformly distributed in \mathbb{M}_m^d . Then*

$$\mathbb{P}(z \in \mathbb{B}_m(s, p)) \geq \frac{1}{2^{m+4}}.$$

The aim of [45] was to combine (4.1) with Lemma 4.2 and the union bound. Indeed, the probability that a randomly chosen point $z \in \mathbb{M}_m^d$ avoids $\mathbb{B}_m(s, p)$ is at most $1 - 2^{-m-4}$. Therefore, the probability that a set $P = \{x^1, \dots, x^N\} \subset \mathbb{M}_m^d$ of N randomly and independently generated points does not intersect $\mathbb{B}_m(s, p)$ is at most $(1 - 2^{-m-4})^N$, i.e.

$$\mathbb{P}(\forall \ell \in \{1, \dots, N\} : x^\ell \notin \mathbb{B}_m(s, p)) \leq (1 - 2^{-m-4})^N.$$

By the union bound over all $(s, p) \in \mathbb{I}_m$, we get further

$$\mathbb{P}(\exists (s, p) \in \mathbb{I}_m : \forall \ell \in \{1, \dots, N\} : x^\ell \notin \mathbb{B}_m(s, p)) \leq \#\mathbb{I}_m \cdot (1 - 2^{-m-4})^N.$$

As $\mathbb{B}_m(s, p)$ was defined in (4.2) as the intersection of all cubes from $\Omega_m(s, p)$, finding a point $x^\ell \in \mathbb{B}_m(s, p)$ means that the same point may be found in all cubes in $\Omega_m(s, p)$. We conclude that if N is large enough to ensure that

$$\exp\left(m2^m \log(2^{m+3}d)\right)(1 - 2^{-m-4})^N < 1,$$

i.e., $N > m2^{2m+4} \log(2^{m+3}d)$, then the randomly generated $P = \{x^1, \dots, x^N\} \subset \mathbb{M}_m^d$ intersects every $\mathbb{B} \in \Omega_m$ with positive probability. Hence, there exists P with $\#P \leq N$ such that $\text{disp}(P) \leq 2^{-m}$. This is essentially the result of [45].

4.2 Connection to k -Restriction Problems

By what we said above, if a point set $P \subset [0, 1]^d$ intersects $\mathbb{B}_m(s, p)$ for all $(s, p) \in \mathbb{I}_m$, then $\text{disp}(P) \leq 2^{-m}$. The randomized construction in [45], which we now want to replace by a deterministic one, was restricted in its choice of points to \mathbb{M}_m^d . Therefore we define

$$\mathbb{C}_m(s, p) = 2^m [\mathbb{B}_m(s, p) \cap \mathbb{M}_m^d], \quad (s, p) \in \mathbb{I}_m.$$

Definition 4.3 Let $m \geq 2$. We say that $T \subset \{1, 2, \dots, 2^m - 1\}^d$ satisfies the condition (S') of the order m if, for every $(s, p) \in \mathbb{I}_m$ it intersects $\mathbb{C}_m(s, p)$.

The rest of [45] then provides a randomized construction of a small set T , which satisfies the condition (S') of order m . This task has two things in common with k -restriction problems. First, the system $\{\mathbb{C}_m(s, p) : (s, p) \in \mathbb{I}_m\}$ is invariant under permutations of $\{1, \dots, d\}$. Indeed, if π is a permutation on $\{1, \dots, d\}$, then $\{x \circ \pi : x \in \mathbb{C}_m(s, p)\} = \mathbb{C}_m(s \circ \pi, p \circ \pi)$. And second, the number of active coordinates is, for every $(s, p) \in \mathbb{I}_m$, bounded from above by $A_m := \min(m2^m, d)$.

To build the connection between the condition (S') and the k -restriction problems, we choose the quadruplet of parameters (b, k, n, m) , see Definition 2.6, as $(2^m - 1, A_m, d, M)$, where m is an arbitrary positive integer. The system \mathcal{C} collects the sets $\mathbb{C}_m(s, p)$ for those $(s, p) \in \mathbb{I}_m$ which have the corresponding active coordinates in $\{1, \dots, A_m\}$. Finally, M is the cardinality of \mathcal{C} .

More formally, let $(s, p) \in \mathbb{I}_m$ with $s_j = 2^m - 1$ for $j > A_m$. Then, we set

$$C_m(s, p) = \mathbb{C}_m(s, p) - 1$$

and define

$$\mathcal{C} = \left\{ C_m(s, p) : (s, p) \in \mathbb{I}_m \text{ with } s_j = 2^m - 1 \text{ for } j > A_m \right\}. \tag{4.3}$$

It follows directly from the definition of \mathbb{I}_m that if $(s, p) \in \mathbb{I}_m$ and $s_j = 2^m - 1$ for some $j \in \{1, \dots, d\}$, then $p_j = 2^{-m}$ and $\mathbb{M}_m \subset [p_j, p_j + \frac{s_j - 1}{2^m}]$. Therefore, every $C_m(s, p) \in \mathcal{C}$ is fully determined by its restriction to $\{1, \dots, A_m\}$ and (with slight abuse of notation) we shall denote by \mathcal{C} also the collection of $C_m(s, p)$ restricted to $\{1, \dots, A_m\}$.

With this in mind, we observe that a set T satisfies the condition (S') of order m if, and only if, the set $T - 1$ satisfies the k -restriction problem with respect to \mathcal{C} .

The parameter M , which is just the cardinality of \mathcal{C} , can be estimated from above in a way similar to (4.1), but note that we do not have to choose the subset of active indices anymore. Each $C_m(s, p) \in \mathcal{C}$ is characterized by $s \in \{0, 1, \dots, 2^m - 1\}^{A_m}$ and $p \in \{1/2^m, \dots, (2^m - 1)/2^m\}^{A_m}$. Therefore,

$$M \leq 2^{2^m A_m}. \tag{4.4}$$

The second important parameter of a k -restriction problem is the minimal size $c := c(\mathcal{C})$ of each of the restriction sets $C_m(s, p)$, see (2.1). A lower bound on c follows directly from Lemma 4.2 and we obtain

$$c \geq \mathbb{P}(z \in \mathbb{B}_m(s, p)) \cdot \#\mathbb{M}_m^{A_m} \geq 2^{-m-4} (2^m - 1)^{A_m}. \tag{4.5}$$

With the choice of parameters as given above, we have $c/b^k \geq 2^{-m-4}$.

4.3 A First Attempt for Derandomization

Using the arguments of the last section, one could use the construction from Theorem 2.7 directly to solve the corresponding k -restriction problem with parameters

$(2^m - 1, A_m, d, 2^{2mA_m})$, whenever $d > 2^m$. This leads to a point set $P \subset \mathbb{M}_m^d$ with $\text{disp}(P) \leq 2^{-m}$ and

$$\#P \leq \left\lceil 2^{m+4} \log(d^{A_m} 2^{2mA_m}) \right\rceil = \mathcal{O}(m^2 2^{2m} \log(d)).$$

Note that this bound matches the union bound from Sect. 4.1. However, the running-time of the algorithm, as given by Theorem 2.7, is

$$\mathcal{O}\left(2^m d^{2A_m} 2^{2mA_m} \mathcal{T}\right) = \mathcal{O}\left(2^{m(1+m2^{m+1})} d^{2m^2 2^m} \mathcal{T}\right),$$

where \mathcal{T} is the time complexity of the membership oracle, which can be assumed of the order $\mathcal{O}(A_m) = \mathcal{O}(m2^m)$ in this case.

4.4 Derandomization Using Splitters

We now describe how we can improve the construction of an explicit solution to the desired k -restriction problem with parameters (b, k, n, m) equal to $(2^m - 1, A_m, d, 2^{2mA_m})$ and the set system \mathcal{C} defined by (4.3). We use the approach of [30] to obtain solutions of the k -restriction problem which are “small” in size and running-time of the corresponding algorithm. “Small” means here, that the dependence on the original problem dimension d is as small as possible.

In the heart of the constructions are splitters, see Sect. 2.3. As already indicated in Sect. 2.3, we use a (d, A_m, A_m^2) -splitter, say $A(m, d)$, to map the original d -dimensional problem to a k -restriction problem in dimension A_m^2 , which can then be solved with cost independent of d .

Recall that, by Definition 2.8, $A(m, d)$ is an (d, A_m, A_m^2) -splitter, if it is a collection of mappings $a : \{1, \dots, d\} \rightarrow \{1, \dots, A_m^2\}$ such that for every $S \subset \{1, \dots, d\}$ with $\#S = A_m$, there is an $a \in A(m, d)$, which is injective on S , i.e., $a(S) \subset \{1, \dots, A_m^2\}$ has A_m elements.

Further, let $T(m) \subset \{1, 2, \dots, 2^m - 1\}^{A_m^2}$ be the solution of the k -restriction problem with parameters $(2^m - 1, A_m, A_m^2, 2^{2mA_m})$ with respect to the original system of restrictions \mathcal{C} , see (4.3). This means, that $T(m) = \{\tau_1, \dots, \tau_K\} \subset \{0, 1, \dots, 2^m - 2\}^{A_m^2}$ such that for every $S' \subset \{1, \dots, A_m^2\}$ with $\#S' = A_m$ and any $C \in \mathcal{C}$ there is $\tau \in T(m)$ with $\tau|_{S'} \in C$.

Now we are in the position to define the solution to the restriction problem with parameters $(2^m - 1, A_m, d, 2^{2mA_m})$ and the system \mathcal{C} . Indeed, we define

$$\begin{aligned} T^* &= T(m) \circ A(m, d) \\ &:= \{\tau \circ a : \{1, \dots, d\} \rightarrow \{0, 1, \dots, 2^m - 2\} : \tau \in T(m), a \in A(m, d)\} \end{aligned}$$

to be the set of concatenations of any splitter with any element of the solution to the restriction problem. Here, we switch between the notion of vectors of length d (resp. A_m^2) and mappings from $\{1, \dots, d\}$ (resp. $\{1, \dots, A_m^2\}$) to $\{0, 1, \dots, 2^m - 2\}$. This should not lead to any confusion.

To show that T^* is indeed a solution to our k -restriction problem, let $S \subset \{1, \dots, d\}$ with $\#S = A_m$. Then, there exists $a \in A(m, d)$, such that $S' = a(S) \subset \{1, 2, \dots, A_m^2\}$ has A_m mutually different elements, i.e., $\#S' = A_m$. Now, for every $C \in \mathcal{C}$, there is some $\tau \in T(m)$, such that $C \ni \tau|_{S'} = (\tau \circ a)|_S$. Hence, T^* , which satisfies $\#T^* = \#T(m) \cdot \#A(m, d)$, is a solution to the restriction problem with parameters $(2^m - 1, A_m, d, 2^{2mA_m})$.

We merge all the components together in a form of an algorithm.

Algorithm 2

1. For $\varepsilon \in (0, \frac{1}{4}]$ and $d \geq 2$, choose a positive integer m with $2^{-m} \leq \varepsilon < 2^{-m+1}$ and set $A_m := \min(m2^m, d)$;
2. Generate a (d, A_m, A_m^2) -splitter $A(m, d)$ as in Lemma 2.9;
3. Generate a solution $T(m)$ to the k -restriction problem with parameters $(2^m - 1, A_m, A_m^2, 2^{2mA_m})$ and restrictions \mathcal{C} from (4.3) as in Theorem 2.7;
4. Set $T^* = T(m) \circ A(m, d) \subset \{0, \dots, 2^m - 2\}^d$;
5. Increase all the coordinates by one, then divide them by 2^m ;
6. Output the resulting point set P .

Theorem 4.4 *There is an absolute constant $C < \infty$ such that, for all $\varepsilon \in (0, \frac{1}{4}]$ and $d \geq 2$, Algorithm 2 constructs a set $P \subset [0, 1]^d$ with $\text{disp}(P) \leq \varepsilon$ and*

$$\#P \leq C \left(\frac{1 + \log_2(\varepsilon^{-1})}{\varepsilon} \right)^4 \log(d^*),$$

where $d^* := \max(d, 2/\varepsilon)$.

The running-time of Algorithm 2 is $\mathcal{O}_\varepsilon(d \log(d))$.

Proof In case that $d < \lfloor 2/\varepsilon \rfloor$, replace d by $\lfloor 2/\varepsilon \rfloor$ in all what follows and delete at the end the last $\lfloor 2/\varepsilon \rfloor - d$ coordinates of the constructed point set.

By Lemma 2.9, there is an explicit, deterministic construction of a (d, A_m, A_m^2) -splitter $A(m, d)$ of size $\mathcal{O}(A_m^2 \log(d)) = \mathcal{O}(m^2 2^{2m} \log(d))$. Furthermore, by Theorem 2.7 and since $d \geq \lfloor 2/\varepsilon \rfloor > 2^m - 1$, there is a deterministic solution $T(m)$ to the k -restriction problem with parameters $(2^m - 1, A_m, A_m^2, 2^{2mA_m})$ and restrictions \mathcal{C} from (4.3) of size

$$\#T(m) = \mathcal{O}\left(2^m \log(A_m^{2A_m} 2^{2mA_m})\right) = \mathcal{O}\left(m^2 2^{2m}\right).$$

Here, we also used (4.5). Altogether, we get

$$\#T^* = \#T(m) \cdot \#A(m, d) = \mathcal{O}\left(m^4 2^{4m} \log(d)\right),$$

which implies the result since $m \leq 1 + \log_2(1/\varepsilon)$.

For the running-time, we need $\mathcal{O}(A_m^2 d \log(d))$ for the construction of $A(m, d)$, see Lemma 2.9, and

$$\mathcal{O}(2^m (eA_m)^{A_m} 2^{2m A_m} \mathcal{T}) = \mathcal{O}(2^{4m^2 2^m} \mathcal{T})$$

for the construction of $T(m)$, see Theorem 2.7, where, again, the cost \mathcal{T} of the membership oracle can be assumed to be $\mathcal{O}(A_m)$. Finally, we need $\mathcal{O}(\#T^*)$ time to build up T^* . The remaining steps are less expensive, which implies the result. \square

Funding Open access funding provided by Johannes Kepler University Linz.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Aistleitner, C., Hinrichs, A., Rudolf, D.: On the size of the largest empty box amidst a point set. *Discrete Appl. Math.* **230**, 146–150 (2017)
2. Alon, N.: Explicit construction of exponential sized families of k -independent sets. *Discrete Math.* **58**(2), 191–193 (1986)
3. Alon, N., Bruck, J., Naor, J., Naor, M., Roth, R.M.: Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs. *IEEE Trans. Inf. Theory* **38**(2), 509–516 (1992)
4. Alon, N., Moshkovitz, D., Safra, S.: Algorithmic construction of sets for k -restrictions. *ACM Trans. Algorithms* **2**, 153–177 (2006)
5. Bachmayr, M., Dahmen, W., DeVore, R., Grasedyck, L.: Approximation of high-dimensional rank one tensors. *Constr. Approx.* **39**(2), 385–395 (2014)
6. Backer, J., Keil, M.: The mono- and bichromatic empty rectangle and square problems in all dimensions. In *Proceedings of the 9th Latin American Symposium on Theoretical Informatics*, pp. 14–25 (2010)
7. Bukh, B., Chao, T.-W.: Empty axis-parallel boxes. In: *International Mathematics Research Notices*, rmb123 (2021) <https://doi.org/10.1093/imrn/rmb123>
8. Chazelle, B., Drysdale, R.L., Lee, D.T.: Computing the largest empty rectangle. *SIAM J. Comput.* **15**(1), 300–315 (1986)
9. Dick, J., Pillichshammer, F.: *Digital Nets and Sequences*. Cambridge University Press, Cambridge (2010)
10. Dick, J., Pillichshammer, F.: Discrepancy theory and quasi-Monte Carlo integration. In: *A Panorama of Discrepancy Theory*, *Lecture Notes in Mathematics* 2107. Springer (2014)
11. Doerr, B., Gnewuch, M., Wahlström, M.: Algorithmic construction of low-discrepancy point sets via dependent randomized rounding. *J. Complex.* **26**, 490–507 (2010)
12. Doerr, C., Gnewuch, M., Wahlström, M.: Calculation of discrepancy measures and applications. In: *A Panorama of Discrepancy Theory*, *Lecture Notes in Mathematics* 2107. Springer (2014)
13. Drmota, M., Tichy, R.F.: *Sequences, discrepancies and applications*. In: *Lecture Notes in Mathematics*, vol. 1651. Springer (1997)
14. Dumitrescu, A., Jiang, M.: Maximal empty boxes amidst random points. *Combin. Probab. Comput.* **22**(4), 477–498 (2013)
15. Dumitrescu, A., Jiang, M.: On the largest empty axis-parallel box amidst n points. *Algorithmica* **66**(2), 225–248 (2013)

16. Dumitrescu, A., Jiang, M.: Perfect vector sets, properly overlapping partitions, and largest empty box (2016) [arXiv:1608.06874v1](https://arxiv.org/abs/1608.06874v1)
17. Dumitrescu, A., Jiang, M.: On the number of maximum empty boxes amidst n points. *Discrete Comput. Geom.* **59**(3), 742–756 (2018)
18. Even, G., Goldreich, O., Luby, M., Nisan, N., Veličković, B.: Approximations of general independent distributions. In: Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing (STOC '92), pp. 10–16. Association for Computing Machinery, New York
19. Gnewuch, M.: Entropy, randomization, derandomization, and discrepancy, Monte Carlo and quasi-Monte Carlo methods 2010. In: Springer Proceedings of the Mathematics Statistics 23. Springer (2012)
20. Gnewuch, M., Srivastav, A., Winzen, C.: Finding optimal volume subintervals with k points and calculating the star discrepancy are NP-hard problems. *J. Complex.* **25**, 115–127 (2009)
21. Hinrichs, A., Prochno, J., Ullrich, M., Vybíral, J.: The minimal k -dispersion of point sets in high-dimensions. *J. Complex.* **51**, 68–78 (2019)
22. Kleitman, D.J., Spencer, J.H.: Families of k -independent sets. *Discrete Math.* **6**, 255–262 (1973)
23. Krieg, D.: On the dispersion of sparse grids. *J. Complex.* **45**, 115–119 (2018)
24. Krieg, D., Rudolf, D.: Recovery algorithms for high-dimensional rank one tensors. *J. Approx. Theory* **237**, 17–29 (2019)
25. Linial, N., Luby, M., Saks, M., Zuckerman, D.: Efficient construction of a small hitting set for combinatorial rectangles in high dimension. *Combinatorica* **17**, 215–234 (1997)
26. Litvak, A.E.: A remark on the minimal dispersion. In: Communications in Contemporary Mathematics, Vol. 23, No. 06. (2021) <https://doi.org/10.1142/S0219199720500601>
27. Lubotzky, A., Phillips, R., Sarnak, P.: Explicit expanders and the Ramanujan conjectures. In: Proceedings of the 18th ACM Symposium Theory of Computing, pp 240–246 (1986)
28. Naamad, A., Lee, D., Hsu, W.-L.: On the maximum empty rectangle problem. *Discrete Appl. Math.* **8**(3), 267–277 (1984)
29. Naor, J., Naor, M.: Small-bias probability spaces: Efficient constructions and applications. *SIAM J. Comp.* **22**(4), 838–856 (1993)
30. Naor, M., Schulman, L.J., Srinivasan, A.: Splitters and near-optimal derandomization. In: IEEE Proceedings of the 36th Annual Symposium, Foundations of Computer Science, pp. 182–191 (1995)
31. Niederreiter, H.: A quasi-Monte Carlo method for the approximate computation of the extreme values of a function. In: Studies in Pure Mathematics, pp. 523–529. Birkhäuser, Basel (1983)
32. Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. Society for Industrial and Applied Mathematics, Philadelphia (1992)
33. Novak, E.: Some results on the complexity of numerical integration. In: Cools, R., Nuyens, D. (eds.) Monte Carlo and Quasi-Monte Carlo Methods. Proceedings in Mathematics and Statistics, vol. 163. Springer, Berlin (2016)
34. Novak, E., Rudolf, D.: Tractability of the approximation of high-dimensional rank one tensors. *Constr. Approx.* **43**(1), 1–13 (2016)
35. Novak, E., Woźniakowski, H.: Tractability of Multivariate Problems. Volume II: Standard Information for Functionals. European Mathematical Society Publishing, House, Zürich (2010)
36. Porat, E., Rothschild, A.: Explicit non-adaptive combinatorial group testing schemes. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfssdóttir, A., Walukiewicz, I. (eds) Automata, Languages and Programming. ICALP 2008. Lecture Notes in Computer Science, vol. 5125. Springer, Berlin
37. Rosser, J.B., Schoenfeld, L.: Approximate formulas for some functions of prime numbers. III. *J. Math.* **6**, 64–97 (1962)
38. Rote, G., Tichy, R.F.: Quasi-Monte Carlo methods and the dispersion of point sequences. *Math. Comput. Model.* **23**(8–9), 9–23 (1996)
39. Rudolf, D.: An upper bound of the minimal dispersion via delta covers. In: Contemporary Computational Mathematics: A Celebration of the 80th Birthday of Ian Sloan. Springer (2018)
40. Seroussi, G., Bshouty, N.H.: Vector sets for exhaustive testing of logic circuits. *IEEE Trans. Inform. Theory* **34**, 513–522 (1988)
41. Sosnovec, J.: A note on the minimal dispersion of point sets in the unit cube. *Eur. J. Combin.* **69**, 255–259 (2018)
42. Temlyakov, V.N.: The Marcinkiewicz-type discretization theorems. *Constr. Approx.* **48**, 337–369 (2018)
43. Temlyakov, V.N.: Universal discretization. *J. Complex.* **47**, 97–109 (2018)

44. Ullrich, M.: A lower bound for the dispersion on the torus. *Math. Comput. Simul.* **143**, 186–190 (2018)
45. Ullrich, M., Vybíral, J.: An upper bound on the minimal dispersion. *J. Complex.* **45**, 120–126 (2018)
46. Yakowitz, S., L'Ecuyer, P., Vázquez-Abad, F.: Global stochastic optimization with low-dispersion point sets. *Oper. Res.* **48**(6), 939–950 (2000)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.