TILBURG ◆ UNIVERSITY

Maximin Latin Hypercube Designs in Two Dimensions

van Dam, E.R.; Husslage, B.G.M.; den Hertog, D.; Melissen, H.

Document version:
Publisher's PDF, also known as Version of record

Publication date:
2005

Link to publication

Citation for published version (APA):
van Dam, E. R., Husslage, B. G. M., den Hertog, D., & Melissen, H. (2005). *Maximin Latin Hypercube Designs in Two Dimensions*. (CentER Discussion Paper; Vol. 2005-8). Operations research.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright, please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 04. Dec. 2020

CentER

Discussion Paper

No. 2005–08

MAXIMIN LATIN HYPERCUBE DESIGNS IN TWO DIMENSIONS

By Edwin van Dam, Bart Husslage, Dick den Hertog, Hans Melissen

January 2005

ISSN 0924-7815

Tilburg University

Maximin Latin hypercube designs in two dimensions

Edwin van Dam[1,*] • Bart Husslage[1,†] • Dick den Hertog[1] • Hans Melissen[2]

[1] *Department of Econometrics and Operations Research, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands*

[2] *Faculty of Information Technology and Systems, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands*

Edwin.vanDam@uvt.nl • B.G.M.Husslage@uvt.nl • D.denHertog@uvt.nl • J.B.M.Melissen@its.tudelft.nl

Abstract

The problem of finding a maximin Latin hypercube design in two dimensions can be described as positioning n non-attacking rooks on an $n \times n$ chessboard such that the minimal distance between pairs of rooks is maximized. Maximin Latin hypercube designs are important for the approximation and optimization of black box functions. In this paper general formulas are derived for maximin Latin hypercube designs for general n, when the distance measure is ℓ^∞ or ℓ^1. Furthermore, for the distance measure ℓ^2 we obtain maximin Latin hypercube designs for $n \leq 70$ and approximate maximin Latin hypercube designs for other values of n. We show that the reduction in the maximin distance caused by imposing the Latin hypercube design structure is small. This justifies the use of maximin Latin hypercube designs instead of unrestricted designs.

Keywords: Branch-and-bound, circle packing, Latin hypercube design, mixed integer programming, non-collapsing, space-filling.
JEL Classification: C90.

1 Introduction

The problem of finding a maximin Latin hypercube design (LHD) in two dimensions can be easiest described as a rook problem. This problem aims to position n rooks on an $n \times n$ chessboard, such that they do not attack each other, and such that the separation distance (i.e. the minimal distance between pairs of rooks) is maximized. More formally, a maximin LHD can be defined as a set of points $(x_i, y_i) \in \{0, \ldots, n-1\}^2$, $i = 0, \ldots, n-1$, such that $x_i \neq x_j$ and $y_i \neq y_j$, $i \neq j$, and such that the separation distance $\min_{i \neq j} d((x_i, y_i), (x_j, y_j))$ is maximal, where d is a certain distance measure. In this paper we derive explicit descriptions of maximin LHDs and general formulas for the maximin LHD distance when the distance measure is ℓ^∞ or ℓ^1. Furthermore, for the ℓ^2-distance measure we obtain maximin LHDs for $n \leq 70$ by using a branch-and-bound method, and approximate maximin LHDs for higher values of n.

Our main motivation for investigating this subject is the fact that maximin LHDs are extremely useful in the field of black box optimization. Suppose that our aim is to approximate and minimize a black box function on a box-constrained domain. By nature, a black box function is not given explicitly, however, we may perform function evaluations. As evaluations of the black box function often involve time-consuming computer simulations, the function is sometimes replaced by an approximating model, based on evaluations in some points. See, e.g. Montgomery [13], Sacks et al [19], [20], Myers [15], Jones et al [10], Booker et al [2], and Den Hertog & Stehouwer [8]. We call such a set of evaluation points a *design*. As is recognized by several authors, such a design for computer experiments should at least satisfy the following two criteria (see Johnson et al [9] and Morris & Mitchell [14]). First of all, the design should be *space-filling* in some sense. When no details on the functional behavior of the response parameters are

*The research of E.R. van Dam has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences.

†The research of B.G.M. Husslage is funded by the SamenwerkingsOrgaan Brabantse Universiteiten (SOBU).

available, it is important to be able to obtain information from the entire design space. Therefore, design points should be "evenly spread" over the entire region. Secondly, the design should be *non-collapsing*. When one of the design parameters has (almost) no influence on the black box function value, two design points that differ only in this parameter will "collapse", i.e. they can be considered as the same point that is evaluated twice. For deterministic black box functions this is not a desirable situation. Therefore, two design points should not share any coordinate values when it is not known a priori which dimensions are important.

There is also a connection between maximin designs and location theory and circle packing. The maximin design problem has already been defined and studied in location theory (see Drezner [5]). In this area of research, the problem is usually referred to as the *continuous multiple facility location problem* or *p-dispersion problem*. Facilities are placed in the plane such that the minimal distance to any other facility is maximal. This problem can be solved by techniques based on Voronoi diagrams (cf Drezner [5]). The resulting solution is certainly space-filling, but not necessarily non-collapsing. We do not see how to adapt these techniques such that the solution fulfills the non-collapsingness criterion as well. There is also much literature on packing and covering with circles. The problem of finding the maximal common radius of n circles which can be packed into a square is equivalent to the maximin design problem. Melissen [12] gives a comprehensive overview of the historical developments and state-of-the-art research in this field. For the ℓ^2-distance measure optimal solutions are known for $n \leq 20$, and many good approximating solutions have been found for $n \geq 21$; see Specht's Packomania website [22]. Baer [1] solved the maximum ℓ^∞-circle packing problem in a d-dimensional unit cube. The ℓ^1-circle packing problem in a square has been solved for many values of n; see Fejes Tóth [6] and Florian [7].

Other space-filling designs, like minimax, IMSE, and maximum entropy designs, are also used. For a good survey of these designs see the book of Santner et al [21]. In this book it is also shown that maximin LHDs generally speaking yield the best approximations. Only a few papers consider maximin designs, e.g. Trosset [24], Dimnaku et al [4], Locatelli & Raber [11], and Stinstra et al [23]. These papers describe heuristics to find approximate maximin designs. Morris & Mitchell [14] are one of the few who consider maximin LHDs.

Designs that are optimized for the space-fillingness criterion often turn out to be highly collapsing. We therefore concentrate on maximin LHDs. In this paper we derive maximin LHDs in two dimensions for the ℓ^∞ and ℓ^1-distance measure and we show that the maximal separation distances are $\lfloor\sqrt{n}\rfloor$ and $\lfloor\sqrt{2n+2}\rfloor$, respectively. By comparing these results with the circle packing results mentioned above, we show that the non-collapsingness restriction reduces the optimal value only slightly, and the reduction converges to zero as $n \to \infty$. For the ℓ^2-measure we were not able to derive such general results, however, using a branch-and-bound technique we were able to find maximin LHDs for $n \leq 70$. For $n \geq 71$ we find periodic and adapted periodic LHDs as approximations for the ℓ^2-maximin LHDs. We also analyze the trade-off between the space-fillingness and the non-collapsingness criterion by relaxing the LHD restriction to $|x_i - x_j| \geq \alpha$ and $|y_i - y_j| \geq \alpha$, $i \neq j$, where $0 \leq \alpha \leq 1$. Note that $\alpha = 0$ corresponds to an unrestricted maximin design, while $\alpha = 1$ leads to a maximin LHD. We show how these maximin quasi-LHDs can be formulated as mixed integer programming problems.

This paper is organized as follows. In Sections 2, 3, and 4, we treat the ℓ^∞, ℓ^1, and ℓ^2-case, respectively. In Section 5 we analyze the trade-off between the space-fillingness and the non-collapsingness criterion. This paper ends with some conclusions in Section 6.

2 ℓ^∞-maximin LHDs

The problem of arranging n points in the square $[0, n-1]^2$ to maximize the minimal ℓ^∞-distance among the pairs of points has been completely solved by Baer [1]. The corresponding maximin distance equals $d = (n-1)/\lfloor\sqrt{n-1}\rfloor$ and is attained, for example, by choosing n points from the set $\{id \mid i = 0, \ldots, \lfloor\sqrt{n-1}\rfloor\}^2$. This design is of course highly collapsing, and although in general there is some freedom to change the design to decrease the "collapsingness" (without decreasing the distance), only in the cases where $n-1$ is a square it is possible to obtain a maximin Latin hypercube design. This follows implicitly from the following where the maximin distance among the Latin hypercube designs is obtained: it equals $\lfloor\sqrt{n}\rfloor$.

This maximin distance can be attained, for example, by using the following construction.

Construction 1 *Let n and d be positive integers such that $n \geq d^2$. Let the sequence (t_j) be defined by $t_0 = 0$ and $t_{j+1} = t_j + \lfloor \frac{n+j}{d} \rfloor$, $j = 0, \dots, d-1$. Then $D = \{(id - j - 1, t_j + i - 1) | j = 0, \dots, d-1; i = 1, \dots, t_{j+1} - t_j\}$ is a Latin hypercube design of n points with separation ℓ^∞-distance d.*

Proof. First note that D indeed consists of $t_d = \sum_{j=0}^{d-1} \lfloor \frac{n+j}{d} \rfloor = n$ points. Since all first coordinates of the points in D are distinct elements of $\{0, \dots, n-1\}$, as are all second coordinates, it follows that D is a Latin hypercube design. From facts such as $t_{j+1} - t_j \geq d$ we find that the separation distance is d. □

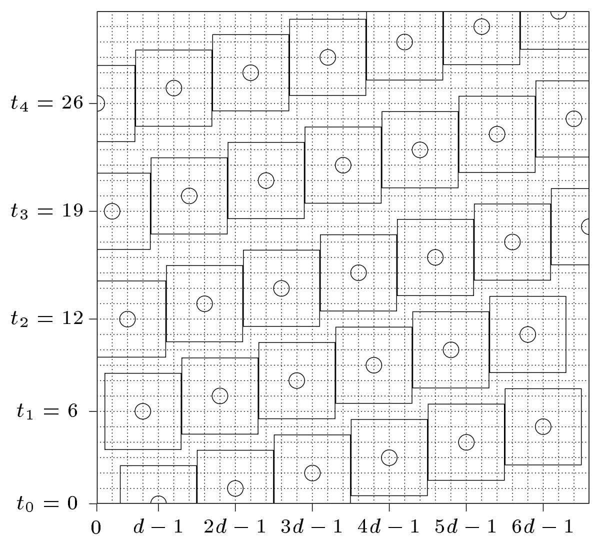


Figure 1: An ℓ^∞-maximin LHD of 33 points; $d = 5$.

This construction (see Figure 1 for an example) shows that Latin hypercube designs of n points with separation distance $\lfloor \sqrt{n} \rfloor$ exist. The following proposition shows that this is optimal.

Proposition 1 *Let $n \geq 2$. An ℓ^∞-maximin Latin hypercube design of n points in two dimensions has separation distance $\lfloor \sqrt{n} \rfloor$.*

Proof. Consider a Latin hypercube design of n points in two dimensions, as subset of $\{0, \dots, n-1\}^2$, with separation distance d. Consider the point $(d-1, y_{d-1})$ of the design. Without loss of generality we may assume that $y_{d-1} \leq \frac{n-1}{2}$. First note that $y_{d-1} + d - 1 \leq n - 1$ because of this assumption and the easily proven fact that $d - 1 \leq \frac{n-1}{2}$. Now, the d points with second coordinates $y_{d-1}, \dots, y_{d-1} + d - 1$ must all have first coordinates in $\{d-1, \dots, n-1\}$ and these coordinates must all be at least d apart. This shows that $n - d \geq (d-1)d$, and hence $d \leq \lfloor \sqrt{n} \rfloor$. This bound and the above construction show that a maximin Latin hypercube design of n points has separation distance $d = \lfloor \sqrt{n} \rfloor$. □

It is easy to see that the difference between the maximin distance for unrestricted designs and the maximin distance for Latin hypercube designs is less than two; hence, the relative difference tends to zero. For example, the reduction in the maximin distance due to the Latin hypercube constraints is less than 10% for $n \geq 324$, and less than 1% for $n \geq 39,204$. See also Figure 2 where the two maximin distances are displayed as a function of the number of points.

3 ℓ^1-maximin LHDs

For the ℓ^1-distance measure the situation is more complicated than for the ℓ^∞-distance measure. Fejes Tóth [6] showed that the maximin distance for unrestricted designs is at most $1 + \sqrt{2n-1}$, with equality if and only if the number of points n is the sum of two consecutive squares. The unique design giving equality for $n = k^2 + (k+1)^2$ is the set $\{i(n-1)/k \mid i = 0, \dots, k\}^2 \cup \{(2i+1)(n-1)/2k \mid i = 0, \dots, k-1\}^2$,

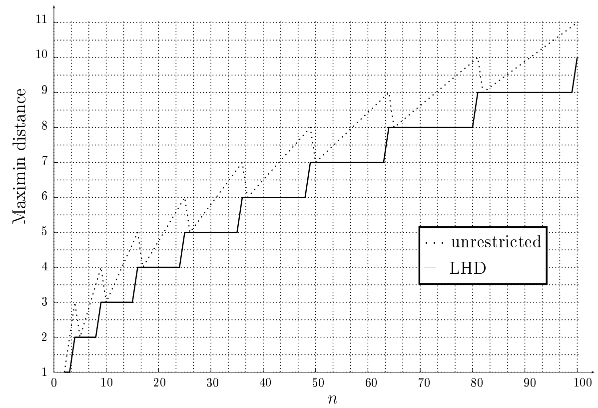


Figure 2: Maximin ℓ^∞-distances for unrestricted designs and for LHDs.

which is highly collapsing. Also for some other values of n the maximin distance has been determined, cf Florian [7]. Typically, the corresponding optimal designs are highly collapsing too; only the cases $n = 2$, 4, and 7, seem to be exceptions: for these cases there is an optimal design which is a Latin hypercube design. For most (approximately "3 out of 4") values of n, however, the maximin distance for unrestricted designs has not been determined yet. For Latin hypercube designs, we will now determine the maximin distance explicitly, for all n: it equals $\lfloor\sqrt{2n+2}\rfloor$. This bound is for example attained by the design in the following constructions, which distinguish between d even and d odd. Particular examples of these constructions are given in Figure 3 ($d = 8$) and Figure 4 ($d = 7$).

Construction 2 *Let n and d be positive integers, d even, such that $n \geq \frac{1}{2}d^2 - 1$. Let the sequence (t_j) be defined by $t_0 = 0$ and $t_{j+1} = t_j + \lfloor\frac{n+\frac{j}{2}+\frac{1}{2}(1-(-1)^j)(\frac{1}{2}d-\frac{1}{2})}{d-1}\rfloor$, $j = 0, \ldots, d-2$. Then $D = \{(i(d-1) - \frac{j}{2} - \frac{1}{2}(1-(-1)^j)(\frac{1}{2}d - \frac{1}{2}) - 1, t_j + i - 1) | j = 0, \ldots, d-2; i = 1, \ldots, t_{j+1} - t_j\}$ is a Latin hypercube design of n points with separation ℓ^1-distance d.*

Proof. Also here D indeed consists of $t_{d-1} = n$ points (although it is more tedious to check here). Checking that D is a Latin hypercube design with separation distance d is tedious, but routine. Important here are the facts that $t_{j+1} - t_j \geq \frac{1}{2}d$ for even j, and $t_{j+1} - t_j \geq \frac{1}{2}d + 1$ for odd j. $\square$

Construction 3 *Let n and d be positive integers, d odd, such that $n \geq \frac{1}{2}d^2 - \frac{1}{2}$. Let the sequence (s_j) be defined by $s_0 = 0$ and $s_{j+1} = s_j + \lfloor\frac{n+\frac{j}{2}+\frac{1}{2}(1-(-1)^j)(\frac{1}{2}d)}{d}\rfloor$, $j = 0, \ldots, d-1$. Then $D = \{(id - \frac{j}{2} - \frac{1}{2}(1 - (-1)^j)(\frac{1}{2}d) - 1, s_j + i - 1) | j = 0, \ldots, d-1; i = 1, \ldots, s_{j+1} - s_j\}$ is a Latin hypercube design of n points with separation ℓ^1-distance d.*

Proof. The proof is similar as before. One can check that D has $s_d = n$ points and separation distance d by using that $s_{j+1} - s_j \geq \frac{1}{2}(d-1)$ for even j, and $s_{j+1} - s_j \geq \frac{1}{2}(d+1)$ for odd j. $\square$

As before, the above constructions can be used to construct optimal designs:

Proposition 2 *Let $n \geq 2$. An ℓ^1-maximin Latin hypercube design of n points in two dimensions has separation distance $\lfloor\sqrt{2n+2}\rfloor$.*

Proof. We shall prove that $n \geq \frac{1}{2}d^2 - 1$ for any Latin hypercube design of n points with separation distance d. For $d \leq 3$ this is obvious, so we may assume that $d \geq 4$.

Consider the Latin hypercube design as a subset of $\{0, \ldots, n-1\}^2$ embedded in $\mathbb{R}^2$, together with the ℓ^1-circles (diamonds) with radius $\frac{1}{2}d$ centered at the n design points. As the interiors of these circles are disjoint, they cover a total area of $n \cdot \frac{1}{2}d^2$. We shall next find a bound on this total area that implies the bound for n in terms of d.

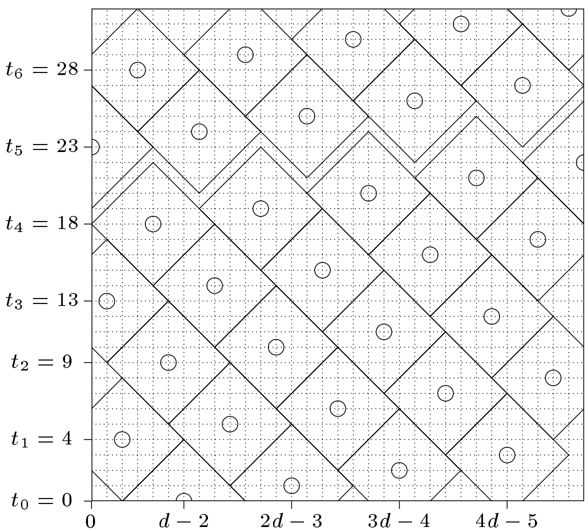


Figure 3: An ℓ^1-maximin LHD of 33 points; $d = 8$.

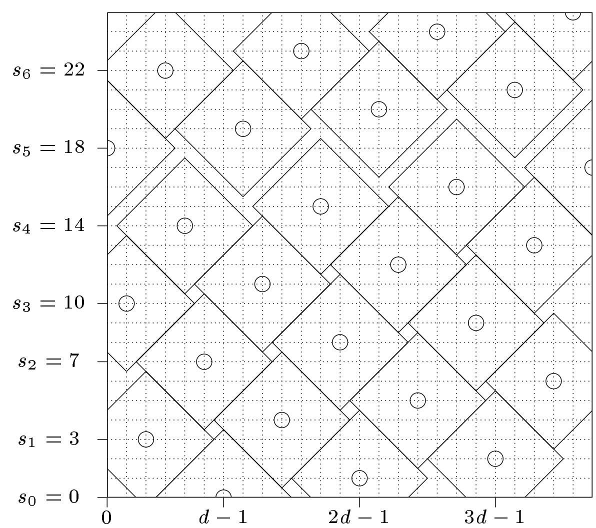


Figure 4: An ℓ^1-maximin LHD of 26 points; $d = 7$.

First, let d be even. The total covered area below the line $y = \frac{1}{2}d - 2$ is equal to

$$\frac{1}{4}d^3 - \frac{3}{4}d^2 + 1.$$

This can be seen by observing that the area below the line $y = \frac{1}{2}d - 2$ that is covered by the two circles centered at the design points with second coordinates i and $d - 4 - i$ equals $\frac{1}{2}d^2$, for $i = 0, \ldots, \frac{1}{2}d - 3$. What remains is to account for the areas covered by the circles that are centered at the design points with second coordinates $\frac{1}{2}d - 2$ and $d - 3$, which are $\frac{1}{4}d^2$ and 1, respectively. The sum of these areas gives the expression above. It thus follows that the total covered area outside the square $[\frac{1}{2}d - 2, n - \frac{1}{2}d + 1]^2$ is at most $d^3 - 3d^2 + 4$, and therefore we find that

$$n \cdot \frac{1}{2}d^2 \leq d^3 - 3d^2 + 4 + (n - d + 3)^2.$$

This implies that $n^2 - n(2d - 6 + \frac{1}{2}d^2) + d^3 - 2d^2 - 6d + 13 \geq 0$, so

$$\begin{aligned} n &\geq d - 3 + \frac{1}{4}d^2 + \frac{1}{4}\sqrt{d^4 - 8d^3 + 24d^2 - 64} \\ &> d - 3 + \frac{1}{4}d^2 + \frac{1}{4}\sqrt{d^4 - 8d^3 + 24d^2 - 32d + 16} = \frac{1}{2}d^2 - 2, \end{aligned}$$

which proves that $n \geq \frac{1}{2}d^2 - 1$. Note that we used that $d \geq 4$, and that the case where $n \leq d - 3 + \frac{1}{4}d^2 - \frac{1}{4}\sqrt{d^4 - 8d^3 + 24d^2 - 64} < 2d - 4$ is easily excluded.

Next, let d be odd. Similar as in the above case we first find that the total covered area below the line $y = \frac{1}{2}(d - 5)$ equals

$$\frac{1}{4}d^3 - d^2 + \frac{5}{2}.$$

As before, this can be seen by observing that the area below the line $y = \frac{1}{2}(d - 5)$ that is covered by the two circles centered at the design points with second coordinates i and $d - 5 - i$ is equal to $\frac{1}{2}d^2$, for $i = 0, \ldots, \frac{1}{2}(d-5) - 1$. The areas covered by the circles that are centered at the design points with second coordinates $\frac{1}{2}(d - 5)$, $d - 4$, and $d - 3$, are $\frac{1}{4}d^2$, $\frac{9}{4}$, and $\frac{1}{4}$, respectively. The sum of these areas gives the expression above. It follows that the total covered area outside the square $[\frac{1}{2}(d - 5), n - \frac{1}{2}d + \frac{3}{2}]^2$ is at most $d^3 - 4d^2 + 10$.

In order to derive a useful inequality we have to look more carefully at the covered area inside the above mentioned square. We claim that each design point (x, y) has the property that the interior of at least one of the two ℓ^1-circles with radius $\frac{1}{2}$ centered at $(x-\frac{1}{2}, y+\frac{1}{2}d)$ and $(x+\frac{1}{2}, y+\frac{1}{2}d)$ is not covered, and we call such an uncovered circle a hole (such holes can clearly be identified in Figure 4). Indeed, a circle that covers any of these two mentioned small circles also covers the circle with radius $\frac{1}{2}$ around $(x, y+\frac{1}{2}(d+1))$. Since the two small circles clearly cannot be covered by the same circle, this proves the claim. We note now that the interiors of all holes are disjoint and, moreover, all holes lie above the line $y=\frac{1}{2}(d-5)$. Since there are $d-2$ design points with holes above the line $y=n-\frac{1}{2}d+\frac{3}{2}$, there are at least $n-d+3-(d-2)=n-2d+5$ holes (among those coming from design points with first coordinates $\frac{1}{2}(d-5)+1,\ldots,n-\frac{1}{2}d+\frac{1}{2}$) that lie entirely inside the square $[\frac{1}{2}(d-5), n-\frac{1}{2}d+\frac{3}{2}]^2$. We thus obtain that

$$n\cdot\frac{1}{2}d^2 \le d^3-4d^2+10+(n-d+4)^2-\frac{1}{2}(n-2d+5),$$

which implies that $n^2-n(2d-\frac{15}{2}+\frac{1}{2}d^2)+d^3-3d^2-7d+\frac{47}{2}\ge 0$. Therefore,

$$\begin{aligned} n &\ge d-\frac{15}{4}+\frac{1}{4}d^2+\frac{1}{4}\sqrt{d^4-8d^3+34d^2-8d-151} \\ &> d-\frac{15}{4}+\frac{1}{4}d^2+\frac{1}{4}\sqrt{d^4-8d^3+34d^2-72d+81}=\frac{1}{2}d^2-\frac{3}{2}, \end{aligned}$$

which implies that $n\ge\frac{1}{2}d^2-1$.

Here we used that $d\ge 4$; the case $n\le d-\frac{15}{4}+\frac{1}{4}d^2-\frac{1}{4}\sqrt{d^4-8d^3+34d^2-8d-151}<2d-6$ is easily excluded. We have thus proved the inequality $n\ge\frac{1}{2}d^2-1$ for all d, and hence that $d\le\lfloor\sqrt{2n+2}\rfloor$. The above constructions show that equality can be attained. □

The difference between the maximin distance for unrestricted designs and the maximin distance for Latin hypercube designs is again less than two. The reduction in the maximin distance due to the Latin hypercube constraints is less than 10% for $n\ge 144$, and less than 1% for $n\ge 19,404$. See also Figure 5 where the maximin distance for Latin hypercube designs and the upper bound/exact value for the maximin distance for unrestricted designs are displayed as a function of the number of points.

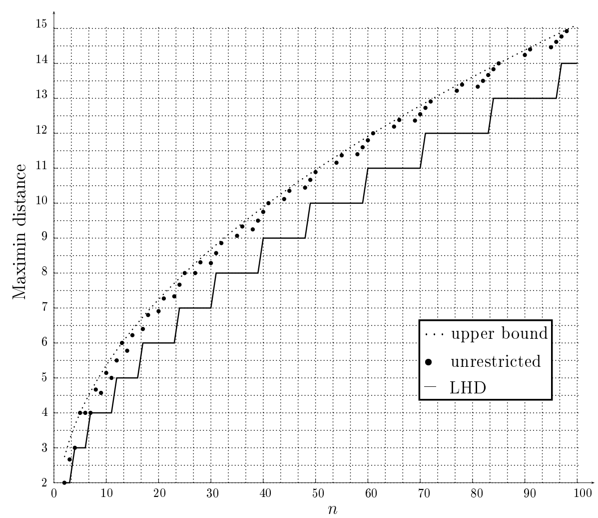


Figure 5: Maximin ℓ^1-distances for unrestricted designs and for LHDs, and general upper bound.

4 ℓ^2-maximin LHDs

So far we have considered maximin designs for the ℓ^∞ and ℓ^1-distance measures. For many real-world applications, however, the ℓ^2-distance measure remains the first choice. Unfortunately, for the Euclidean measure the situation is much more complicated than for the other two measures. There is no known infinite class of optimal designs in the unrestricted situation, as was the case, for instance, for the ℓ^1-measure, let alone a complete solution like for the ℓ^∞-measure. Optimal designs are only known for up to 20 points (cf Melissen [12]). Many of the designs require dedicated optimality proofs and some of the larger cases were even proven by computer-assisted proof techniques, e.g. see Peikert et al [17]. As there are no general results for maximin designs in the ℓ^2-measure, this is still a field of research where world records can be broken, e.g. see Casado et al [3]. A list of the best-known circle packings in a square (and also in a circle and in a rectangle) is on a website maintained by Specht [22]. So far, the list contains many very good (and probably close to optimal) designs for up to 300 points, and a few larger numbers. The optimal designs may be devoid of any symmetry or nice structure (for instance, for 10 or 13 points), there can be multiple optimal solutions (e.g. for 17 points), and there are even optimal designs that have points that are not fixed, but that can move around a little (for instance, for 7, 11, and 13 points). This supports the believe that a complete solution for all points is not likely to be found ever. To a lesser extent the same seems to be the case for the problem of finding ℓ^2-maximin LHDs, although the (adapted) periodic solutions we found may turn out to be optimal.

4.1 Maximin LHDs

To find maximin Latin hypercube designs for the ℓ^2-distance measure (for small n), we designed a branch-and-bound algorithm. This algorithm searches for Latin hypercube designs of n points with separation distance at least d, for given n and d, by examining all designs $\{(x, y_x) | x = 0, \ldots, n-1\}$, represented by the sequence $(y_0, y_1, \ldots, y_{n-1}) \in \{0, 1, \ldots, n-1\}^n$, while checking whether they are non-collapsing and have separation distance at least d.

As a first approach one could use the search tree where the root has n branches giving the value of y_0, and each corresponding node further branches into n parts giving the value of y_1, et cetera, until we are at the end nodes giving the value of y_{n-1}. One can cut branches from the node corresponding to the partial design $(y_0, y_1, \ldots, y_t)$ if points are already collapsing or are separated by distance less than d. In this way, we found maximin LHDs for n up to 40.

A disadvantage of above approach is that it does not use the fact that useless partial designs occur as part of other partial designs (for example, $(0, 3, 4)$ is part of $(9, 12, 15, 0, 3, 4)$) in different parts of the tree, and hence are not cut off by just one cut. Note also in this respect that it is beneficial to cut the tree at small depth. To (partly) solve this disadvantage, we use a different tree. For this, we first fix the value $y_x = y \neq \frac{n-1}{2}$, where the index x will be determined later, and will depend on the particular end node in the tree. Because of symmetry, we will assume that x is at most $f = \lfloor \frac{n}{2} \rfloor - 1$. This will be the root of the tree, and it branches into n parts, giving the value of y_{x+1}. The corresponding nodes further branch into n parts, giving the value of y_{x+2}, et cetera, up to the nodes giving the value of y_{x+n-1} (and at these end nodes we take $x = 0$). Moreover, for $t = 0, \ldots, f-1$, the nodes corresponding to the value of $y_{x+n-1-f+t}$ (roughly speaking: when over "half" of the points in the design are chosen) have n additional branches giving the value of y_{x-1} (we now start extending the partial design on the other side of x), and these branch further corresponding to the values of y_{x-2}, et cetera, up to the values of y_{x-f+t}. At these end nodes we have a design $(y_0, y_1, \ldots, y_{n-1})$ by taking $x = f - t$. With the branch-and-bound algorithm based on this tree we managed to find optimal designs, or prove optimality of some designs found by hand, for $n \leq 70$ by taking $y = \lfloor \sqrt{d^2 - 2} \rfloor$ (but this value does not seem to be crucial). For the instance $(n, d) = (69, \sqrt{80})$ we took $y = \frac{n-1}{2}$. This has the advantage that because of symmetry only the cases $y_{x+1} < y$ (i.e. only half the tree) have to be searched, however, the disadvantage is that also the value $x = \frac{n-1}{2}$ must be considered (this is implemented by letting $f = \frac{n-1}{2}$). In this particular case this was no disadvantage since all cutting turned out to be performed far before half of the points in the design were chosen.

Using these branch-and-bound techniques we were able to find maximin LHDs for up to 70 points. These maximin LHDs, which were also found by our heuristic, can be derived from Table 1 (see Section 4.2). Unlike the situation without LHD constraints many of the optimal designs exhibit some nice regularity in that the designs turn out to be either periodic arrangements or slightly adapted periodic arrangements. As an example, see the ℓ^2-maximin LHDs in Figures 6 and 7.

4.2 Heuristics

Due to increasing computational effort the applicability of the branch-and-bound algorithm that we presented is restricted to the smaller designs that we have found. To extend the range of designs we have tried several heuristics to find good designs.

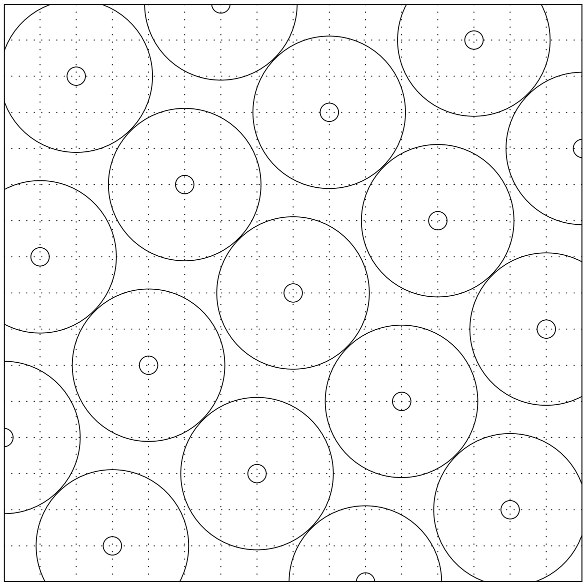


Figure 6: An ℓ^2-maximin LHD of 17 points; $d^2 = 18$.

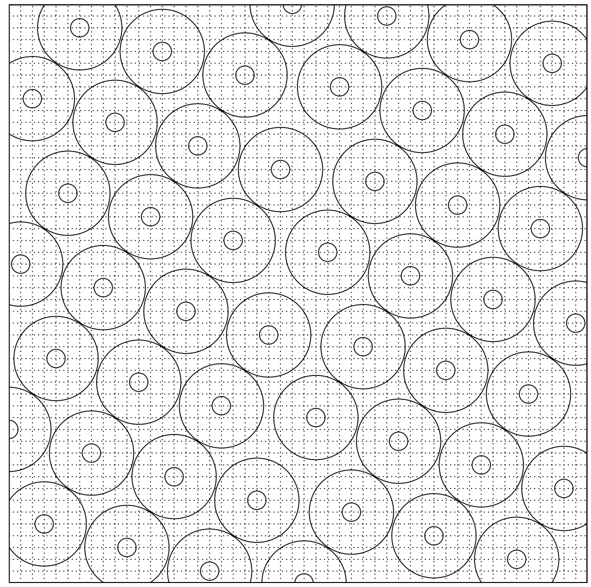


Figure 7: An ℓ^2-maximin LHD of 50 points; $d^2 = 52$.

One option, for instance, is to consider the ℓ^∞ and ℓ^1-maximin Latin hypercube designs. If we pick the best of the two, with respect to the ℓ^2-measure, we end up with some good designs.

We have tried simulated annealing to improve these ℓ^∞ and ℓ^1-designs. However, when starting with these designs the algorithm was not able to turn up better ones. When starting from a random design the algorithm consumed excessive amounts of computation time without turning up solutions that were at least as good as the ℓ^∞ or ℓ^1-designs.

A third approach that presented itself, when noticing the nice, periodic structure of many of the maximin LHDs that were found by the branch-and-bound algorithm, is to look for periodic designs. This turned out to be very successful.

For given n, we started with choosing a period p such that $\gcd(n+1,p) = 1$ and constructed an LHD with points (x, y_x), where $y_x = (x+1)p \bmod (n+1) - 1$, for $x = 0, \ldots, n-1$. This heuristic often resulted in maximin LHDs and, otherwise, good designs.

To improve our results we then considered the more general sequence $z_x = (s + xp) \bmod n$ (note that we changed the modulus), for all periods $p = 1, \ldots, \lfloor \frac{n}{2} \rfloor$, and different starting points $s = 0, \ldots, \lfloor \frac{n}{2} \rfloor$. Note, however, that the resulting sequence z might no longer be one-to-one, i.e. some values may occur more than once, and hence the resulting design $\{(x, z_x) | x = 0, \ldots, n-1\}$ might not be an LHD. Now, let $k > 0$ be the smallest value for which $z_k = z_0$; it then follows that $k = \frac{n}{\gcd(n,p)}$. When $k < n$ a way to construct a one-to-one sequence of length n, and hence an LHD, is by shifting parts of the sequence by, say, q, and repeating this when necessary. To formulate this more explicitly, we obtain an LHD represented by $y_x = (s + xp + \beta q) \bmod n$, for $x = \beta k, \ldots, (\beta+1)k - 1$ and $\beta = 0, \ldots, \gcd(n,p) - 1$. For n up to 200 we tested all "shifts" q in the range $[1-p, p-1]$ and all starting points $s = 0, \ldots, \lfloor \frac{n}{2} \rfloor$ and it turned out that taking q equal to either $1-p$ or -1, and s equal to $p-1$, yielded the best designs. Additional tests indicated that the value $q = 1$ should also be considered. Therefore, the final heuristic considered only $q \in \{1-p, -1, 1\}$ and $s = p - 1$.

Combining both periodic heuristics we found results for n up to 1000; the obtained LHDs for $n \leq 70$ are optimal. The LHDs, with their corresponding minimal distances, are depicted in Table 1. In this table the tuple (p, q, m) defines an LHD as follows. If $m = n + 1$ we get the design points (x, y_x), where

$$y_x = (x+1)p \bmod (n+1) - 1, \text{ for } x = 0, \ldots, n-1,$$

whereas we have

$$y_x = ((x+1)p - 1 + \beta q) \bmod n, \text{ for } x = \beta k, \ldots, (\beta+1)k - 1, \text{ and } \beta = 0, \ldots, \gcd(n,p) - 1,$$

when $m = n$, where $k = \frac{n}{\gcd(n,p)}$.

n	d^2	p	q	m	n	d^2	p	q	m
2	2	1	–	3	374	425	118	–	375
4	5	2	–	5	388	442	21	–	389
7	8	3	–	8	395	450	139	–	396
9	10	3	–	10	408	461	22	−21	408
12	13	5	–	13	415	466	79	–	416
14	17	4	–	15	422	481	96	−95	422
17	18	5	–	18	429	482	22	−1	429
21	20	5	–	22	430	485	22	−21	430
22	25	5	–	23	433	490	59	–	434
23	26	5	–	24	448	509	61	–	449
28	29	12	–	29	462	530	141	–	463
31	32	7	–	32	470	533	193	–	471
33	34	13	–	34	474	545	62	−61	474
34	37	6	–	35	488	549	64	–	489
38	41	7	–	39	492	565	86	–	493
44	50	19	–	45	509	578	89	–	510
50	52	14	−13	50	520	586	136	1	520
52	58	8	–	53	534	593	64	–	535
58	61	9	–	59	537	610	25	–	538
60	65	8	–	61	550	613	154	–	551
65	68	25	–	66	552	629	199	–	553
67	74	9	–	68	559	640	67	–	560
75	80	9	–	76	575	650	155	–	576
76	85	34	–	77	582	661	93	–	583
83	90	25	–	84	586	673	26	−25	586
86	97	10	−9	86	600	674	168	–	601
90	98	27	–	91	607	680	27	–	608
93	100	11	–	94	613	692	71	–	614
95	101	10	−1	95	626	722	265	–	627
100	109	30	–	101	634	725	27	–	635
102	113	28	−27	102	641	738	119	–	642
104	117	11	–	105	658	745	28	–	659
111	128	41	–	112	666	746	119	–	667
121	130	51	–	122	672	761	100	–	673
126	145	12	–	127	678	765	130	−129	678
136	149	13	–	137	679	778	101	–	680
146	157	56	−55	146	686	785	28	–	687
148	160	34	−1	148	694	793	124	–	695
149	170	13	–	150	706	808	288	−287	706
156	178	36	–	157	710	809	76	–	711
162	180	14	−13	162	717	818	249	–	718
166	181	36	–	167	730	820	78	−77	730
170	185	52	−51	170	732	829	76	–	733
171	194	37	–	172	738	850	192	–	739
176	197	14	−13	176	756	853	209	–	757
180	202	39	–	181	758	865	340	−339	758
184	205	66	−65	184	761	866	79	–	762
187	208	15	–	188	766	872	30	−29	766
194	212	52	−51	194	776	882	295	–	777
200	218	16	–	201	777	884	107	–	778
202	226	15	–	203	783	898	183	–	784
208	241	56	–	209	795	901	30	−1	795
216	245	16	–	217	800	909	187	–	801
225	250	99	–	226	808	914	287	–	809
232	257	16	–	233	814	925	169	–	815
240	269	71	–	241	821	932	31	–	822
246	277	17	–	247	828	949	266	–	829
253	290	45	–	254	840	954	298	−297	840
260	292	46	−45	260	843	962	175	–	844
267	296	79	–	268	850	977	205	–	851
268	305	63	–	269	866	981	196	–	867
279	306	18	−1	279	875	986	137	–	876
280	320	18	−17	280	880	1009	32	–	881
291	328	81	–	292	888	1013	115	–	889
298	338	116	–	299	896	1025	116	–	897
306	346	113	–	307	914	1037	194	–	915
313	356	19	–	314	919	1042	119	–	920
324	360	51	−1	324	922	1060	268	−267	922
326	365	120	−119	326	940	1073	33	–	941
330	370	20	–	331	957	1076	145	–	958
335	386	71	–	336	962	1090	204	−1	962
350	401	20	–	351	970	1105	147	–	971
358	409	54	–	359	985	1124	277	–	986
367	410	21	–	368	998	1129	258	−257	998

Table 1: (Maximin) ℓ^2-distance LHDs on break points.

Table 1 only gives designs for which n is a "break point", i.e. the values of n for which $d_n > d_i$, for all $i < n$. Designs for intermediate values of n may have a minimal distance that is smaller than the minimal distance of their preceding break point. For these n, however, better designs can easily be derived. Every

LHD is defined by its sequence of y_x-values, which can be split up into several increasing subsequences. For example, the ℓ^2-maximin LHD of 17 points in Figure 6 consists of the sequences $(4, 9, 14)$, $(1, 6, 11, 16)$, $(3, 8, 13)$, $(0, 5, 10, 15)$, and $(2, 7, 12)$. Each of these sequences can be augmented by extra points, starting with the sequence with the smallest end value (i.e. 12 in above example), while retaining the minimal distance. Hence, a given periodic LHD of n points can be extended to an LHD of $n' > n$ points with the same minimal distance. Figure 8 shows how to extend an ℓ^2-maximin LHD of 17 points, with $d^2 = 18$, to ℓ^2-maximin LHDs of 18, 19, and 20 points, all with d^2 equal to 18. The LHD of 17 points could also be extended further to LHDs of $n' \geq 21$ points with $d^2 = 18$, however, Table 1 shows that this is no longer optimal.

Figure 9 displays the best found ℓ^2-distances d for unrestricted designs and Latin hypercube designs for up to 300 points. The upper bound depicted in this figure can easily be derived when applying Oler's theorem [16] to the square $[0, n-1]^2$, resulting in:

$$d \leq 1 + \sqrt{1 + (n-1)\frac{2}{\sqrt{3}}}.$$

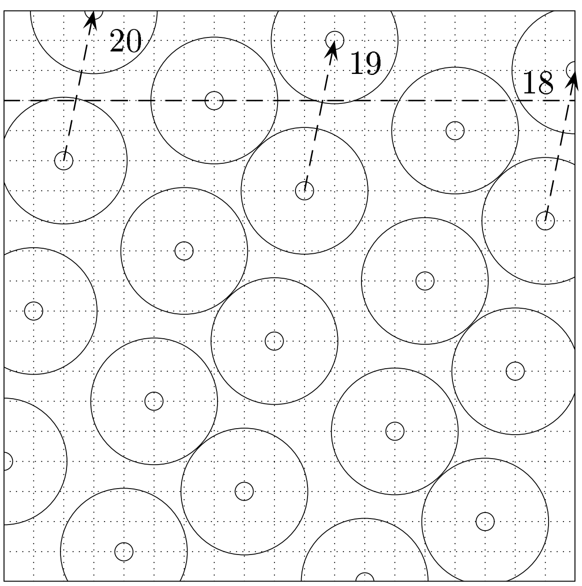


Figure 8: LHD constructions for 18, 19, and 20 points, based on an ℓ^2-maximin LHD of 17 points.

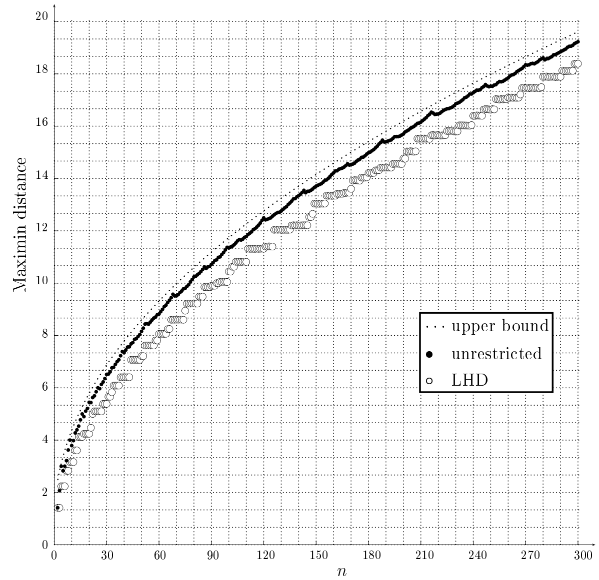


Figure 9: (Maximin) ℓ^2-distances d for unrestricted designs and LHDs, and general upper bound.

5 Quasi non-collapsing designs

In previous sections we have looked at the maximin distance of unrestricted designs and of Latin hypercube designs, with respect to the ℓ^∞, ℓ^1, and ℓ^2-distance measures. For unrestricted designs we were interested in finding design points in the square $[0, n-1]^2$ with maximal separation distance. To obtain Latin hypercube designs we also required that projections of the design points along any of the coordinate axes resulted in a one-dimensional equidistant design. This extra restriction drastically reduced the number of possible designs, however, it was shown that the effect on the maximin distance was small.

Instead of requiring the coordinates of a design to be equally distributed over the interval $[0, n-1]$, we will now require the coordinates to be separated by at least some distance $\alpha \in [0, 1]$. Note that $\alpha = 0$ results in an unrestricted (possibly collapsing) design, whereas $\alpha = 1$ yields a (non-collapsing) Latin hypercube design. Therefore, we will call a design with $\alpha \in [0, 1]$ *quasi non-collapsing*. It is interesting to investigate how the maximin distance is affected by the choice of α. For a given value of $\alpha \in [0, 1]$ we can find the corresponding maximin distance by solving the following optimization problem:

$$\begin{array}{lll} \max \min\limits_{i \neq j} & d((x_i, y_i), (x_j, y_j)) & \\ \text{s.t.} & \alpha \le |x_i - x_j| & i,j = 0, \dots, n-1; i \neq j \\ & \alpha \le |y_i - y_j| & i,j = 0, \dots, n-1; i \neq j \\ & 0 \le x_i \le n-1 & i = 0, \dots, n-1 \\ & 0 \le y_i \le n-1 & i = 0, \dots, n-1. \end{array} \quad (1)$$

Here, $d((x_i, y_i), (x_j, y_j))$ denotes the distance between the design points (x_i, y_i) and (x_j, y_j). In the next sections we show how to compute the maximin distance (and the corresponding design), with respect to the ℓ^∞, ℓ^1, and ℓ^2-distance measures, for every value of $\alpha \in [0, 1]$.

5.1 The ℓ^1-case

For the ℓ^1-distance measure the objective function in (1) reduces to $|x_i - x_j| + |y_i - y_j|$, which results in a non-convex, non-linear program (NLP). We rewrite problem (1) as the following mixed integer program (MIP):

$$\begin{array}{lll} \max & d & \\ \text{s.t.} & d \le x_j - x_i + z_{ij} & i,j = 0, \dots, n-1; i < j \\ & \alpha \le x_{i+1} - x_i & i = 0, \dots, n-2 \\ & \alpha \le z_{ij} & i,j = 0, \dots, n-1; i < j \\ & z_{ij} \le y_i - y_j + 2(n-1)(1-h_{ij}) & i,j = 0, \dots, n-1; i < j \\ & z_{ij} \le y_j - y_i + 2(n-1)h_{ij} & i,j = 0, \dots, n-1; i < j \\ & 0 \le x_i \le n-1 & i = 0, \dots, n-1 \\ & 0 \le y_i \le n-1 & i = 0, \dots, n-1 \\ & 0 \le z_{ij} \le n-1 & i,j = 0, \dots, n-1; i < j \\ & h_{ij} \in \{0, 1\} & i,j = 0, \dots, n-1; i < j. \end{array} \quad (2)$$

Here, $h_{ij} = 1$ if $y_i \ge y_j$, and $h_{ij} = 0$ otherwise, resulting in $z_{ij} \le |y_i - y_j|$. Since d (and hence z_{ij}) is maximized this yields $z_{ij} = |y_i - y_j|$. Solving (2) gives the maximin distance d as function of the quasi non-collapsingness parameter $\alpha \in [0, 1]$. Figure 10 gives two examples of such a function for designs of 10 and 11 points, respectively. The plots are a result of solving (2), using the *XA Mixed Integer Solver*, for 200 equidistant values of $\alpha \in [0, 1]$.

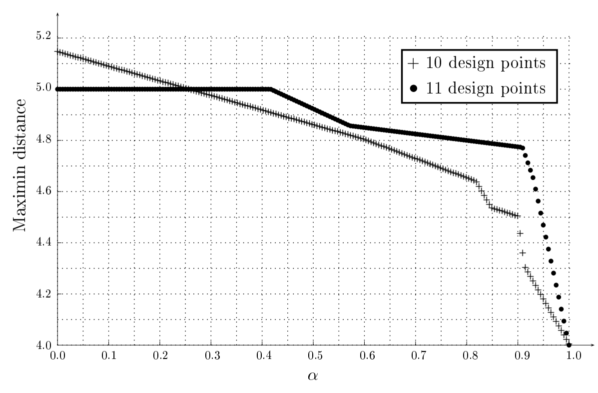
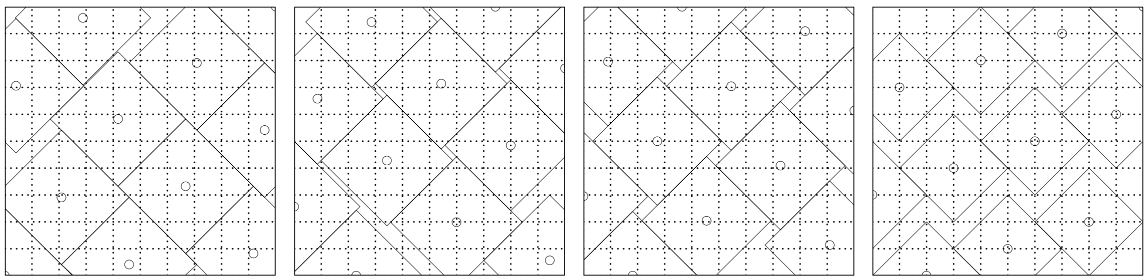


Figure 10: Maximin ℓ^1-distance as function of the quasi non-collapsingness parameter α for 10 and 11 design points.

Both of these plots indicate non-concave, non-increasing, piecewise-linear functions. This behavior can be explained as follows. Fixing all h_{ij} in (2) results in a linear program (LP) with continuous variables only, and α in the right-hand side of the constraints. From the sensitivity analysis of an LP we know that the optimal value as a function of α is a non-increasing, concave, piecewise-linear function. For every realization of the binary variables we get such a function. The maximal d is found by taking the maximum over all these functions, resulting in a non-increasing, piecewise-linear function that is not necessarily concave.

An interesting observation can be made from the designs of 11 points. It is seen that α can be taken up to a value of 0.41 without affecting the unrestricted maximin distance. Furthermore, for α between 0.41 and 0.91 the maximin distance stays within 5% of its unrestricted value; dropping sharply only for values larger than 0.91. Apparently, it is possible to construct a highly non-collapsing design of 11 points, without decreasing the unrestricted maximin distance much. As an example, see Figure 11, which shows four maximin designs corresponding to the points of inflection $\alpha = 0.41$, $\alpha = 0.57$, $\alpha = 0.91$, and $\alpha = 1.00$.



$\alpha = 0.41,\ d = 5.00$ $\alpha = 0.57,\ d = 4.86$ $\alpha = 0.91,\ d = 4.77$ $\alpha = 1.00\ d = 4.00$

Figure 11: Maximin ℓ^1-distance designs of 11 points for $\alpha = 0.41$, $\alpha = 0.57$, $\alpha = 0.91$, and $\alpha = 1.00$.

5.2 The ℓ^∞-case

For the ℓ^∞-distance measure the objective function in (1) reduces to $\max\{|x_i - x_j|, |y_i - y_j|\}$. We can follow the same kind of reasoning as with the ℓ^1-measure and rewrite the optimization problem as an MIP. Unfortunately, extra binary variables have to be included to deal with the maximum-operator in the objective function, which will increase the computation time:

$$
\begin{array}{lll}
\max & d & \\
\text{s.t.} & d \le x_j - x_i + (n-1)(1-k_{ij}) & i,j = 0,\dots,n-1; i<j \\
 & d \le z_{ij} + (n-1)k_{ij} & i,j = 0,\dots,n-1; i<j \\
 & \alpha \le x_{i+1} - x_i & i = 0,\dots,n-2 \\
 & \alpha \le z_{ij} & i,j = 0,\dots,n-1; i<j \\
 & z_{ij} \le y_i - y_j + 2(n-1)(1-h_{ij}) & i,j = 0,\dots,n-1; i<j \\
 & z_{ij} \le y_j - y_i + 2(n-1)h_{ij} & i,j = 0,\dots,n-1; i<j \\
 & 0 \le x_i \le n-1 & i = 0,\dots,n-1 \\
 & 0 \le y_i \le n-1 & i = 0,\dots,n-1 \\
 & 0 \le z_{ij} \le n-1 & i,j = 0,\dots,n-1; i<j \\
 & h_{ij} \in \{0,1\} & i,j = 0,\dots,n-1; i<j \\
 & k_{ij} \in \{0,1\} & i,j = 0,\dots,n-1; i<j.
\end{array}
\tag{3}
$$

The binary variables h_{ij} serve the same purpose as in (2); for the extra binary variables k_{ij} it holds that $k_{ij} = 1$ if $|x_i - x_j| \ge |y_i - y_j|$, and $k_{ij} = 0$ otherwise, resulting in $d \le \max\{|x_i - x_j|, |y_i - y_j|\}$. Like the ℓ^1-distance measure, we can compute the maximin distance d for several values of $\alpha \in [0,1]$. Figure 12 gives two examples, for designs of 6 and 7 points, respectively. The plots are a result of solving (3) for 200 uniformly distributed values of $\alpha \in [0,1]$. Again, it can be argued that the maximin distance, as a function of α, is a non-increasing, piecewise-linear function. Note that this function appears to be linear for designs of 6 points. For 7 points, we can construct highly non-collapsing designs without decreasing the maximin distance more than 15%, by taking $\alpha \le 0.85$.

5.3 The ℓ^2-case

For the ℓ^2-distance measure the situation is more complicated than for the ℓ^∞ and ℓ^1-measures. The objective function in (1) reduces to the quadratic function $(x_i - x_j)^2 + (y_i - y_j)^2$ (for sake of convenience we square the ℓ^2-distance). The resulting NLP is in fact a multi-extremal optimization problem, which calls for a global optimizer. We used the Lipschitz Global Optimizer (LGO) of Pintér [18] to compute

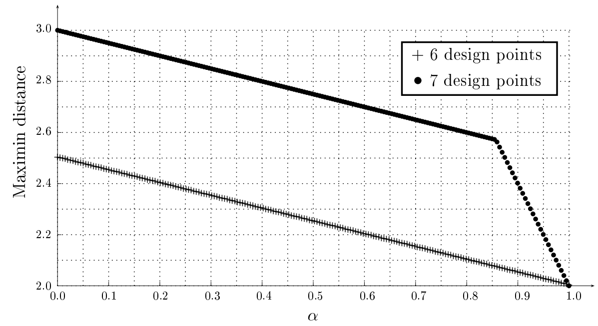


Figure 12: Maximin ℓ^∞-distance as function of the quasi non-collapsingness parameter α for 6 and 7 design points.

the maximin distance as function of the quasi non-collapsingness parameter $\alpha \in [0, 1]$. Within LGO we applied the multi-start global search option, followed by a local search phase, to increase the probability of obtaining a good solution.

Although the obtained distances only give us lower bounds for the (unknown) global maximin distances, we can still extract information about the behavior of the maximin distances from them. As an example, see Figure 13, which shows results for designs of 5 and 6 points, respectively. We solved NLP (1), with objective function $(x_i - x_j)^2 + (y_i - y_j)^2$, for 50 evenly spread values of $\alpha \in [0, 1]$ to obtain these results. Both plots indicate a non-trivial behavior. For 5 design points, a small change in α heavily affects the maximin distance for values of α less than 0.53 and larger than 0.86, whereas this effect is less pronounced when α lies between 0.53 and 0.86. For designs of 6 points, the maximin distance is only heavily affected by large values of α, i.e. $\alpha > 0.80$. This facilitates the construction of highly non-collapsing designs with a maximin distance that does not deviate too much from the unrestricted maximin distance.

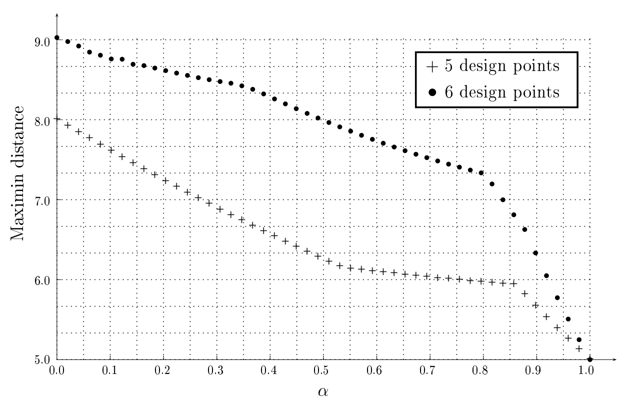


Figure 13: Maximin ℓ^2-distance d^2 as function of the quasi non-collapsingness parameter α for 5 and 6 design points.

6 Conclusions

For the ℓ^∞ and ℓ^1-distance measures it is possible to explicitly describe maximin LHDs. For the ℓ^2-distance measure we have obtained maximin LHDs up to $n = 70$. Using (adapted) periodic LHDs we have found LHDs that are optimal for $n \leq 70$ and that approximate ℓ^2-maximin LHDs for values of n up to 1000. A comparison with unrestricted maximin designs shows that adding the non-collapsingness criterion only slightly reduces the maximin distance. For the ℓ^∞-measure the reduction in maximin distance due to the LHD restriction is less than 10% for $n \geq 324$. For the ℓ^1-measure the reduction in the maximin distance due to the LHD restriction is less than 10% for $n \geq 144$. This justifies the use of maximin LHDs instead of unrestricted maximin designs in practice.

The trade-off between the space-fillingness and the non-collapsingness criterion can even be made more precise. To this end we have introduced maximin quasi-LHDs, which can be obtained by mixed integer programming methods. The resulting trade-off curve can be used in practice to decide on the level of non-collapsingness. Extending the results of this paper to maximin LHDs in higher dimensions is subject of further research.

References

[1] Baer, D. (1992). Punktverteilungen in Würfeln beliebiger Dimension bezüglich der Maximum-norm, *Wiss. Z. Pädagog. Hochsch. Erfurt/Mühlhausen, Math.-Naturwiss. Reihe* 28, 87–92.

[2] Booker, A.J., Dennis, J.E., Frank, P.D., Serafini, D.B., Torczon, V., and Trosset, M.W. (1999). A rigorous framework for optimization of expensive functions by surrogates, *Structural Optimization* 17, 1–13.

[3] Casado, L.G., Garcia, I., Szabó, P.G., and Csendes, T. (2000). Packing equal circles in a square II – New results for up to 100 circles using the TAMSASS-PECS algorithm, *New Trends in Equilibrium Systems*, 1–9.

[4] Dimnaku, A., Kincaid, R., and Trosset, M.W. (2002). Approximate solutions of continuous dispersion problems, *2002 ISOLDE conference*, *http://www.math.wm.edu/˜trosset/*.

[5] Drezner, Z., ed. (1995). *Facility location: A survey of applications and methods*, Springer-Verlag, New York.

[6] Fejes Tóth, L. (1971). Punktverteilungen in einem Quadrat, *Studia Sci. Math. Hung.* 6, 439–442.

[7] Florian, A. (1989). Verteilung von Punkten in einem Quadrat, *Sitzungsber., Abt. II, Österr. Akad. Wiss., Math.-Naturwiss. Kl.* 198, 27–44.

[8] Hertog, D. den, and Stehouwer, H.P. (2002). Optimizing color picture tubes by high-cost nonlinear programming, *European Journal on Operations Research* 140(2), 197–211.

[9] Johnson, M.E., Moore, L.M., and Ylvisaker, D. (1990). Minimax and maximin distance designs, *Journal of Statistical Planning and Inference* 26, 131–148.

[10] Jones, D., Schonlau, M., and Welch, W. (1998). Efficient global optimization of expensive black-box functions, *Journal of Global Optimization* 13, 455–492.

[11] Locatelli, M., and Raber, U. (2002). Packing equal circles in a square: A deterministic global optimization approach, *Discrete Applied Mathematics* 122(1-3), 139–166.

[12] Melissen, J.B.M. (1997). *Packing and Covering with Circles*, Ph.D. Thesis, ISBN 90-393-1500-0, Utrecht.

[13] Montgomery, D.C. (1984). *Design and Analysis of Experiments*, Second Edition, John Wiley & Sons, New York.

[14] Morris, M.D., and Mitchell, T.J. (1995). Exploratory designs for computer experiments, *Journal of Statistical Planning and Inference* 43, 381–402.

[15] Myers, R.H. (1999). Response surface methodology – Current status and future directions, *Journal of Quality Technology* 31, 30–74.

[16] Oler, N. (1961). A finite packing problem, *Canadian Mathematical Bulletin* 4(2), 153–155.

[17] Peikert, R., Würtz, D., Monagan, M., and Groot, C. den (1991). Packing circles in a sphere: A review and new results, *Proceedings of the 15th IFIP Conference on System Modeling and Optimization, Springer Lecture Notes in Control and Information Sciences* 180, 111–124.

[18] Pintér, J.D. (1995). *LGO: A model development and solver system for continuous global optimization*, Pintér Consulting Services Inc., Halifax, Nova Scotia, Canada.

[19] Sacks, J., Schiller, S.B., and Welch, W.J. (1989). Designs for computer experiments, *Technometrics* 31, 41–47.

[20] Sacks, J., Welch, W.J., Mitchell, T.J., and Wynn, H.P. (1989). Design and analysis of computer experiments, *Statistical Science* 4, 409–435.

[21] Santner, Th.J., Williams, B.J., and Notz, W.I. (2003). *The Design and Analysis of Computer Experiments, Springer Series in Statistics*, Springer-Verlag, New York.

[22] Specht, E., *Packomania*, *http://www.packomania.com*.

[23] Stinstra, E., Hertog, D. den, Stehouwer, H.P., and Vestjens, A. (2003). Constrained maximin designs for computer experiments, *Technometrics* 45(4), 340–346.

[24] Trosset, M.W. (1999). Approximate maximin distance designs, *Proceedings of the Section on Physical and Engineering Sciences*, 223–227, Alexandria, VA.