CentER

Discussion Paper

No. 2007–16

BOUNDS FOR MAXIMIN LATIN HYPERCUBE DESIGNS

By Edwin R. van Dam, Gijs Rennen, Bart Husslage

February 2007

ISSN 0924-7815

Tilburg University

Bounds for maximin Latin hypercube designs

Edwin R. van Dam • Gijs Rennen • Bart Husslage

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

Edwin.vanDam@uvt.nl • *G.Rennen@uvt.nl* • *B.G.M.Husslage@uvt.nl*

Abstract

Latin hypercube designs (LHDs) play an important role when approximating computer simulation models. To obtain good space-filling properties, the maximin criterion is frequently used. Unfortunately, constructing maximin LHDs can be quite time-consuming when the number of dimensions and design points increase. In these cases, we can use approximate maximin LHDs. In this paper, we construct bounds for the separation distance of certain classes of maximin LHDs. These bounds are useful for assessing the quality of approximate maximin LHDs. Until now only upper bounds are known for the separation distance of certain classes of unrestricted maximin designs, i.e. for maximin designs without a Latin hypercube structure. The separation distance of maximin LHDs also satisfies these "unrestricted" bounds. By using some of the special properties of LHDs, we are able to find new and tighter bounds for maximin LHDs. Within the different methods used to determine the upper bounds, a variety of combinatorial optimization techniques are employed. Mixed Integer Programming, the Travelling Salesman Problem, and the Graph Covering Problem are among the formulations used to obtain the bounds. Besides these bounds, also a construction method is described for generating LHDs that meet Baer's bound for the ℓ^∞ distance measure for certain values of n.

Keywords: Latin hypercube design, maximin, space-filling, mixed integer programming, travelling salesman problem, graph covering.

1 Introduction

Latin hypercube designs form a class of designs that are often used for finding approximations of deterministic computer simulation models on a box-constrained domain. This type of simulation model is often used in engineering, logistics, and finance to analyze and optimize the design of products or processes (see Driessen (2006) and Stinstra (2006)). The reason for approximating these models is that a computer simulation run is usually quite time-consuming to perform. This makes the model impractical when it comes to obtaining insight in the underlying process or in optimizing its parameters. A common approach to overcome this problem is to determine a meta-model that approximates the relation between the input and output parameters of the computer simulation model. Such a meta-model is based on the information obtained from a limited number of simulation runs. See e.g. Montgomery (1984), Sacks et al. (1989a), Sacks et al. (1989b), Jones et al. (1998), Myers (1999), Booker et al. (1999), and Den Hertog and Stehouwer (2002). The quality of the meta-model depends, among others, on the choice of the simulation runs. Each simulation run can be represented by a vector containing the values of the input parameters. When the simulation model has k input parameters, the simulation runs are therefore treated as points in the k-dimensional space. A set of simulation runs is called a *design* and the number of design points is denoted by n. As designs can be scaled to any box-constrained domain, the designs in this paper are without loss of generality constructed on a hypercube.

As is recognized by several authors, a design should at least satisfy the following two criteria (see Johnson et al. (1990) and Morris and Mitchell (1995)). Firstly, the design should be *space-filling*. This means that the whole design space should be well-represented by the design points. To accomplish this, we consider the maximin criterion, which states that the points should be chosen such that the minimal distance between any two points is maximal. This minimal distance is called the *separation distance* of the design. The maximin criterion is defined for different distance measures. In this paper, we use the ℓ^∞, ℓ^1, and ℓ^2-measure. Other criteria for space-filling designs are minimax, integrated mean squared error, and maximum entropy designs. A good survey of these designs can be found in the book of Santner et al. (2003). Secondly, the design should be *non-collapsing*. When a parameter has (almost) no influence on the output, then two design points that only differ in this parameter can be considered as the same point. As each point is time-consuming to evaluate, this situation should be avoided. Therefore, non-collapsingness requires that for each parameter the values in the design points should be distinct.

Latin hypercube designs (LHDs) are a particular class of non-collapsing designs. For LHDs on the $[0, n-1]^k$ hypercube, the values of the input parameters are chosen from the set $\{0, 1, \ldots, n-1\}$ and for each input parameter each value in this set is chosen exactly once. More formally, we can describe a k-dimensional LHD of n design points as a set of n points $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ik})$ with $\{x_{ij} | i = 1, 2, \ldots, n\} = \{0, 1, \ldots, n-1\}$ for all j. In Santner et al. (2003), it is shown that maximin LHDs generally yield good approximations.

Finding maximin LHDs can be time-consuming for larger values of k and n. Therefore, most results in this field concern approximate maximin LHDs, although exact maximin LHDs are found for some cases. For the distance measures ℓ^∞ and ℓ^1 for instance, Van Dam et al. (2007) derive general formulas for two-dimensional maximin LHDs. Furthermore, they obtain two-dimensional ℓ^2-maximin LHDs for $n \leq 70$ by using a branch-and-bound algorithm.

For approximate maximin LHDs more results are available. In Van Dam et al. (2007), approximate two-dimensional ℓ^2-maximin LHDs are constructed for up to 1000 points by optimizing a periodic structure. In Husslage et al. (2006), this is extended to more dimensions. Morris and Mitchell (1995) use a simulated annealing approach to obtain approximate ℓ^1- and ℓ^2-maximin LHDs for up to five dimensions and up to 12 points and a few larger values. Jin et al. (2005) describe an enhanced stochastic evolutionary algorithm for finding approximate LHDs. The maximin distance criterion is one of the criteria which they consider. Ye et al. (2000) use an exchange algorithm to obtain approximate maximin symmetric LHDs. The symmetry property is imposed to reduce the computational effort.

In this paper, we construct bounds for the separation distance of certain classes of maximin LHDs. These bounds are useful for assessing the quality of approximate maximin LHDs by comparing their separation distances with the corresponding upper bounds. Until now only upper bounds are known for the separation distance of certain classes of unrestricted maximin designs, i.e. for maximin designs without a Latin hypercube structure. Oler (1961), for instance, gives an upper bound for two-dimensional unrestricted ℓ^2-maximin designs. Furthermore, Baer (1992) gives an upper bound for the separation distance of unrestricted ℓ^∞-maximin designs. The separation distance of maximin LHDs also satisfies the "unrestricted" bounds. By using some of the special properties of LHDs, we are able to find new and tighter bounds for maximin LHDs. Table 1 gives an overview of the classes of maximin LHDs treated in each section of this paper. For these classes, different methods are used to determine the upper bounds. Within the methods, a variety of combinatorial optimization techniques are employed. Mixed Integer Programming, the Travelling Salesman Problem, and the Graph Covering Problem are among the formulations used to obtain the bounds. Besides these bounds, also a construction method is described for generating LHDs that meet Baer's bound for the ℓ^∞ distance measure for certain values of n.

This paper is organized as follows. In Section 2, two methods are described that give bounds for ℓ^2-maximin LHDs. The first method is based on the average squared ℓ^2-distance, which is useful when k is relatively large compared to n. The second method gives a bound for two-dimensional LHDs by partitioning the hypercube into smaller parts. Section 3 describes bounds for ℓ^∞-maximin LHDs. Using graph covering, a bound is obtained for k-dimensional maximin LHDs. Furthermore, a method is described to construct LHDs meeting Baer's bound. Also a specific bound is given for three-dimensional maximin LHDs. The bound is found by projecting

	ℓ^2	ℓ^∞	ℓ^1
$k = 2$	Section 2.2		
$k = 3$		Section 3.3	
k large relative to n	Section 2.1	Section 3.1	Section 4
$n \approx m^k$ for $k, m \in \mathbb{N}$		Section 3.2	

Table 1: Overview of the classes of maximin LHDs treated in this paper.

the three-dimensional hypercube onto two dimensions, and then partitioning it into strips. In Section 4, a bound for ℓ^1-maximin LHDs, which is based on the average ℓ^1-distance, is obtained. This method is similar to the first method for the ℓ^2-distance. Finally, Section 5 gives some final remarks and conclusions.

2 Upper bounds for the ℓ^2-distance

2.1 Bounding by the average

We obtain a bound for the separation distance of an LHD from the fact that the minimal squared distance is at most the average squared distance between points of an LHD.

Proposition 1. *Let D be an LHD of n points and dimension k. Then the separation ℓ^2-distance d satisfies*

$$d^2 \leq \left\lfloor \frac{n(n+1)k}{6} \right\rfloor .$$

Proof. Let $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ik})$. The average squared distance among the points of D is

$$\frac{1}{\binom{n}{2}} \sum_{i>j} \sum_{h} (x_{ih} - x_{jh})^2 = \frac{1}{\binom{n}{2}} \sum_{h} \sum_{i>j} (x_{ih} - x_{jh})^2 = \frac{1}{\binom{n}{2}} \sum_{h} \sum_{i'>j'} (i' - j')^2 = kn(n+1)/6. \tag{1}$$

Since the squared separation distance is integer and at most equal to the average squared distance, rounding (1) finishes the proof. □

For fixed k, the separation distance of n points in a k-dimensional cube of side $n-1$ is at most order $n^{\frac{k-1}{k}}$ (this can be seen by comparing the total volume of n pairwise disjoint balls of diameter d to the total volume of the cube). It follows that the bound in Proposition 1 is not of the right order to be tight if k is fixed and n grows.

Note that if an LHD would have separation distance close to the bound in Proposition 1, then the separation distance and average distance are about the same, i.e. all points are at approximately the same distance from each other. Supported by the fact that the maximal number of equidistant points in a k-dimensional space is $k+1$, we are led to believe that the bound in Proposition 1 can only be close to tight if k is large with respect to n. Note that when n is fixed the upper bound is linear in k (besides the rounding). The following lemma also provides a lower bound that is linear in k, which shows that the bound is of the right order if n is fixed and k grows.

Lemma 1. *Let $d_{\max}(n,k)$ be the maximin ℓ^2-distance of an LHD of n points and dimension k. Then $d_{\max}(n, k_1 + k_2)^2 \geq d_{\max}(n, k_1)^2 + d_{\max}(n, k_2)^2$.*

Proof. Let $D_1 = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and $D_2 = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ be maximin LHDs in dimensions k_1 and k_2, respectively. Let $\mathbf{z}_i$ be the concatenation of $\mathbf{x}_i$ and $\mathbf{y}_i$, for $i = 1, \ldots, n$, then one obtains an LHD $D = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$ of n points in dimension $k_1 + k_2$ with squared separation distance at least $d_{\max}(n, k_1)^2 + d_{\max}(n, k_2)^2$. □

To show the strength of the bound in Proposition 1, we determine the maximin distance for LHDs on at most 5 points in any dimension. For this purpose, we first formulate the maximin problem as an integer programming problem.

Let $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ik})$, be an LHD. For each $j = 1, \ldots, k$ the map π sending i to $x_{ij} + 1$ is a permutation of $\{1, 2, \ldots, n\}$. Thus the maximin distance is the solution of the following problem:

$$\begin{array}{ll} \max & d \\ \text{s.t.} & \sum_{\pi \in S_n} k_\pi (\pi(i) - \pi(j))^2 \geq d^2, \quad \forall\, i > j \\ & \sum_{\pi \in S_n} k_\pi = k \\ & k_\pi \in \mathbb{N}_0, \quad \forall\, \pi \in S_n \end{array} \tag{2}$$

where S_n is the set of permutations of $\{1, 2, \ldots, n\}$. Note that for any j, replacing x_{ij} by $n-1-x_{ij}$ for all i does not change the separation distance of the design. Thus we may restrict the set S_n to its first half when ordered lexicographically. This reduces the number of variables in the program to $n!/2$. Note also that we may assume that $k_{\pi^*} \geq 1$ for an arbitrary permutation π^*, since we may reorder the points of the design as we wish.

Consider now the cases $n = 3, 4$, and 5 (for $n = 2$ the bound is trivially attained).

Proposition 2. *For $n = 3$, the maximin ℓ^2-distance satisfies $d_{\max}(3, k)^2 = k + 3\lfloor \frac{k}{3} \rfloor$.*

Proof. The stated result follows from solving the above integer programming problem (2) by hand (the number of variables is 3). □

For $n = 4$ we have that $d^2 \leq \lfloor \frac{10k}{3} \rfloor$. By solving the integer program (2) by computer for $k \leq 19$ we obtain the following table.

dimension k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
$d_{\max}(4, k)^2$	1	5	6	12	14	20	21	26	28	33	35	40	41	46	48	53	55	60	62
upper bound	3	6	10	13	16	20	23	26	30	33	36	40	43	46	50	53	56	60	63

Table 2: Squared maximin ℓ^2-distance for LHDs on 4 points.

Proposition 3. *For $n = 4$, the maximin ℓ^2-distance satisfies $d_{\max}(4, k)^2 = \lfloor \frac{10k}{3} \rfloor - 1$ if $k \equiv 1$ or 5 (mod 6), $d_{\max}(4, k)^2 = \frac{10k}{3} - 2$ if $k \equiv 3$ (mod 6), and $d_{\max}(4, k)^2 = \lfloor \frac{10k}{3} \rfloor$ if k is even, except for the cases $k \leq 5$, $k = 7$, $k = 13$. For these exceptions, see Table 2.*

Proof. By recursively applying Lemma 1 (always with $k_2 = 6$, and starting with $k_1 = 6, 8$, and 10) one obtains maximin LHDs for all even dimensions at least 6 meeting the upper bound.

For the odd dimensions the upper bound $\lfloor \frac{10k}{3} \rfloor$ cannot be attained. Even worse, for odd k divisible by 3, $d^2 = \frac{10k}{3} - 1$ cannot be attained. Suppose on the contrary that one of the above values is attained, then the minimal squared distance is at least $\frac{10k-3}{3}$. Fix the point that has the smallest average squared distance to the remaining points. Then this average squared distance equals $\frac{10k-e}{3}$, where e equals 0, 1, 2, or 3. Now let k_0 be the number of coordinates where the fixed point is 0 or 3, and let $k_1 = k - k_0$ be the number of coordinates where it is 1 or 2. It follows that the average squared distance of this point to the other points equals $\frac{1^2+2^2+3^2}{3} k_0 + \frac{(-1)^2+1^2+2^2}{3} k_1 = \frac{14}{3} k_0 + 2k_1 = \frac{10k-e}{3}$. It now follows that $k_0 = \frac{k}{2} - \frac{e}{8}$, and hence k should be even (and $e = 0$). Thus, if the upper bound is attained, then k cannot be odd, and for odd k divisible by 3, the gap with the upper bound is at least 2.

Now by recursively applying Lemma 1 (always with $k_2 = 6$, and starting with $k_1 = 9, 11$, and 19) one obtains maximin LHDs for all odd $k \geq 21$. □

For $n = 5$ we have that $d^2 \leq 5k$. By solving the integer program (2) by computer for $k \leq 14$ we obtain the following table.

dimension k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$d_{\max}(5,k)^2$	1	5	11	15	24	27	32	40	43	50	54	60	64	70
upper bound	5	10	15	20	25	30	35	40	45	50	55	60	65	70

Table 3: Squared maximin ℓ^2-distance for LHDs on 5 points.

Proposition 4. *For $n = 5$, the maximin ℓ^2-distance satisfies $d_{\max}(5,k)^2 = 5k - 1$ if k is odd, and $d_{\max}(5,k)^2 = 5k$ if k is even, except for the cases $k \leq 4$, $k = 6$, $k = 7$, $k = 9$. For these exceptions, see Table 3.*

Proof. We claim that the bound $5k$ can only be attained for even k. Indeed, if this bound is attained, then all points of the design are at equal distance. Fix a point, let k_0 be the number of coordinates where this point is 0 or 4, let k_1 be the number of coordinates where it is 1 or 3, and let k_2 be the number of coordinates where it is 2. It follows that the average squared distance of this point to the other points equals $\frac{30}{4}k_0 + \frac{15}{4}k_1 + \frac{10}{4}k_2 = 5k$. Since $k_0 + k_1 + k_2 = k$, it follows that $3k_1 + 4k_2 = 2k$, and hence k_1 is always even. We claim that this implies that the distance between any two points must be even, and hence that k must be even. To prove the claim, consider two points, and let k_{ee} and k_{oo} be the number of coordinates where both points are even and odd, respectively. Also, let k_{eo} and k_{oe} be the number of coordinates where one point is even and the other one is odd, and the other way around, respectively. From the above it follows that both $k_{oe} + k_{oo}$ and $k_{eo} + k_{oo}$ are even (k_1 is even), and hence $k_{oe} + k_{eo}$ is even. But then the distance between the two points is even, which proves the claim. Thus we may conclude that the bound $d^2 \leq 5k$ cannot be attained for odd k.

Besides the maximin designs obtained from integer programming, we obtain maximin designs in even dimensions by recursively applying Lemma 1 with k_1 and k_2 both even and at least 8. Then maximin LHDs for other odd dimensions are obtained by applying Lemma 1 with $k_1 = 5$ and k_2 even and at least 8. □

Also for $n = 6$ we computed the integer programming problem (2) for some small values of k, see the table below. For some values of k only a lower bound was obtained.

dimension k	1	2	3	4	5	6	7	8	9	10	11	12	13
$d_{\max}(6,k)^2$	1	5	14	22	32	40	≥ 47	≥ 53	≥ 61	≥ 67	≥ 74	≥ 82	≥ 89
upper bound	7	14	21	28	35	42	49	56	63	70	77	84	91

Table 4: Squared maximin ℓ^2-distance for LHDs on 6 points.

Note that by simulated annealing better designs have been found for $k = 8, 10$, cf. Husslage (2006). More specifically, $d_{\max}(6,8)^2 \geq 54$ and $d_{\max}(6,10)^2 \geq 68$.

2.2 Bounding by non-overlapping circles in two dimensions

2.2.1 Methods to determine upper bounds

To find a bound on the ℓ^2-maximin distance for two-dimensional LHDs, we first look at the more general class of unrestricted designs. An upper bound for the ℓ^2-maximin distance of unrestricted designs, derived with Oler's theorem (Oler 1961), is:

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

For LHDs, the value of d^2 is always the sum of two squared integers. We can use this property to define a slightly stronger upper bound. The Oler bound for LHDs is obtained by rounding down

$$\left(1 + \sqrt{1 + (n-1)\frac{2}{\sqrt{3}}}\right)^2$$

to the nearest integer that can be written as the sum of two squared integers.

To determine a bound more tailored to the special characteristics of two-dimensional ℓ^2-maximin LHDs, we use the following properties. A two-dimensional LHD of n points can be represented by a sequence y which is a permutation of the set $\{0, 1, \ldots, n-1\}$. The points of the LHD are then given by $\{(x, y_x)|x = 0, \ldots, n-1\}$. We can depict a two-dimensional ℓ^2-maximin LHD with separation distance d by n non-overlapping circles with diameter d and their centers given by $\{(x, y_x)|x = 0, \ldots, n-1\}$. We will call circles consecutive if they have consecutive x-values.

The general idea for the new bound is the following. First, determine for each d how much distance along the y-axis is at least needed to place $\lceil d \rceil$ consecutive non-overlapping circles with diameter d on the $\{0, 1, \ldots, \lceil d \rceil - 1\} \times \mathbb{N}_+$-grid. With this information, we can determine a lower bound for the distance along the y-axis necessary to place n non-collapsing points with separation distance d. The second step is to determine d_n, which denotes the minimal d for which this distance is larger than $n - 1$. By taking the largest sum of two squares that is strictly smaller than d_n^2, we have found an upper bound on the squared separation distance of two-dimensional ℓ^2-maximin LHDs of n points. In the remainder of this section, we describe these two steps in more detail.

For the first step, fix $x \in \{0, 1, \ldots, n - \lceil d \rceil\}$ and consider a subset of $\lceil d \rceil$ circles with consecutive x-values $x, \ldots, x + \lceil d \rceil - 1$ with y-values $y_x, \ldots, y_{x+\lceil d \rceil - 1}$. The distance along the x-axis between any of these circles is less than d. This implies that the y-value of any circle in this set influences the y-value of any other circle in the set, due to the non-overlapping criterion.

The first step is thus to determine the minimal distance along the y-axis necessary to place $\lceil d \rceil$ consecutive non-overlapping circles with diameter d. This minimal distance is independent of the fixed value x and equal to $Y(d)$ in the following problem:

$$\begin{array}{rll} Y(d) = & \min & (\max\{y_1, \ldots, y_{\lceil d \rceil}\} - \min\{y_1, \ldots, y_{\lceil d \rceil}\}) \\ & \text{s.t.} & \|(k, y_k) - (l, y_l)\| \geq d \quad \forall\, k, l \in \{1, \ldots, \lceil d \rceil\}, k \neq l \\ & & y \in \mathbb{N}_+^{\lceil d \rceil}, \end{array} \tag{3}$$

where $y_1, \ldots, y_{\lceil d \rceil}$ represent the y-values of $\lceil d \rceil$ consecutive circles.

For every $k, l \in \{1, \ldots, \lceil d \rceil\}$, $k \neq l$, we can calculate the minimal required difference between y_k and y_l. When we take, without loss of generality, $y_k \leq y_l$, applying Pythagoras' theorem gives that:

$$y_l - y_k \geq \left\lceil \sqrt{d^2 - (l-k)^2} \right\rceil \geq 1.$$

In this result, we can round up because y_l and y_k must be integer. Furthermore, the last inequality holds because $(l - k)^2 < d^2$. This inequality implies that the points in the set are also non-collapsing. Thus, adding non-collapsingness constraints will not influence the value of $Y(d)$.

A drawback of solving Problem (3) is that it is very time-consuming for larger values of d. Therefore, instead of solving Problem (3), we propose to solve the following problem:

$$\begin{array}{rll} \widetilde{Y}(d) = & \min & y_{\sigma(\lceil d \rceil)} - y_{\sigma(1)} \\ & \text{s.t.} & \left\|(\sigma(i+1), y_{\sigma(i+1)}) - (\sigma(i), y_{\sigma(i)})\right\| \geq d \quad \forall\, i \in \{1, \ldots, \lceil d \rceil - 1\} \\ & & y_{\sigma(1)} < y_{\sigma(2)} < \ldots < y_{\sigma(\lceil d \rceil)} \\ & & \sigma \in S_{\lceil d \rceil} \\ & & y \in \mathbb{N}_+^{\lceil d \rceil}, \end{array} \tag{4}$$

where $S_{\lceil d \rceil}$ is the set of all permutations of $\{1, \ldots, \lceil d \rceil\}$. For $\widetilde{Y}(d)$ the following holds:

Lemma 2. *For any d, we have $\widetilde{Y}(d) \leq Y(d)$.*

Proof. The difference between Problems (3) and (4) is only in the constraints. Problem (4) only requires non-overlappingness of circles with consecutive y-values, whereas Problem (3) requires that all circles are non-overlapping. As the constraints of Problem (4) are thus a subset of the constraints of Problem (3), $\widetilde{Y}(d)$ is at most $Y(d)$. □

Take for example $d = \sqrt{65}$. By total enumeration, we find that $Y(\sqrt{65}) = 49$ and $\widetilde{Y}(\sqrt{65}) = 46$. Figures 1 and 2 show two settings for y that attain these values. As can be seen, the solution to Problem (3) gives a solution where all circles are non-overlapping. Problem (4), on the other hand, results in overlapping of circles with non-consecutive y-values.

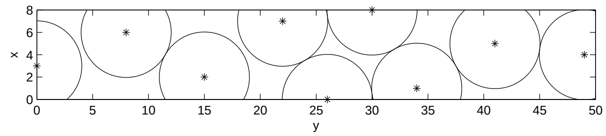

Figure 1: Setting for y that attains $Y(\sqrt{65}) = 49$.

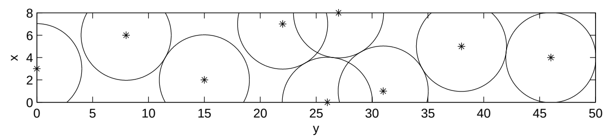

Figure 2: Setting for y that attains $\widetilde{Y}(\sqrt{65}) = 46$.

The following lemma shows that Problem (4) can be reformulated.

Lemma 3. *$\widetilde{Y}(d)$ in Problem (4) is equal to:*

$$\widetilde{Y}(d) = \min_{\sigma} \sum_{i=1}^{\lceil d \rceil - 1} \left\lceil \sqrt{d^2 - (\sigma(i+1) - \sigma(i))^2} \right\rceil \tag{5}$$

Proof. Clearly, $y_{\sigma(\lceil d \rceil)} - y_{\sigma(1)} = \sum_{i=1}^{\lceil d \rceil - 1} (y_{\sigma(i+1)} - y_{\sigma(i)})$. For a given permutation σ, we must choose y such that $y_{\sigma(i+1)} - y_{\sigma(i)}$ is minimized for each $i \in \{1, \ldots, \lceil d \rceil - 1\}$ and satisfies the constraints. This way, we also minimize the sum of these terms. Applying Pythagoras' theorem gives that the minimal difference between $y_{\sigma(i+1)}$ and $y_{\sigma(i)}$ that satisfies the constraints is:

$$\left\lceil \sqrt{d^2 - (\sigma(i+1) - \sigma(i))^2} \right\rceil .$$

We can round up because $y_{\sigma(i+1)}$ and $y_{\sigma(i)}$ must be integer. Using this result, we can rewrite $\widetilde{Y}(d)$ as stated in (5). □

Determining $Y(d)$ and $\widetilde{Y}(d)$ can be done in a number of ways:

- Total enumeration. This can be done within reasonable time for $d \leq 10$. For larger values of d, computation time becomes very large, as the number of permutations is $\lceil d \rceil !$. We can use this method to determine both $Y(d)$ and $\widetilde{Y}(d)$.
- Mixed Integer Program (MIP). We can rewrite Problem (3) as a MIP as follows:

$$\begin{array}{rlll} Y(d) = & \min & y_{max} & \\ & \text{s.t.} & y_{max} \geq y_i & \forall\, i \in \{1, \ldots, \lceil d \rceil\} \\ & & (y_l - y_k) + M x_{kl} \geq \left\lceil \sqrt{d^2 - (l-k)^2} \right\rceil & \forall\, k, l \in \{1, \ldots, \lceil d \rceil\}, k \neq l \\ & & (y_k - y_l) + M(1 - x_{kl}) \geq \left\lceil \sqrt{d^2 - (l-k)^2} \right\rceil & \forall\, k, l \in \{1, \ldots, \lceil d \rceil\}, k \neq l \\ & & y \in \mathbb{R}_+^{\lceil d \rceil}, x_{kl} \in \{0, 1\} & \forall\, k, l \in \{1, \ldots, \lceil d \rceil\}, \end{array} \tag{6}$$

  with $M = 2\lceil d \rceil$. Note that we do not have to require $y \in \mathbb{N}_+^{\lceil d \rceil}$, as the constraints will enforce this. We can also rewrite Problem (4) as a MIP problem, but we will omit this as the next method is more suitable.

- Travelling Salesman Problem (TSP). It is possible to rewrite Problem (5) as a TSP problem. Take a complete graph $K_{\lceil d \rceil + 1}$ and label the vertices $0, 1, \ldots, \lceil d \rceil$. Define the weights of the edges as follows:

$$\begin{aligned} w_{0,i} &= 0 && \forall\ i = 1, \ldots, \lceil d \rceil \\ w_{i,j} &= \left\lceil \sqrt{d^2 - (j-i)^2} \right\rceil && \forall\ i, j = 1, \ldots, \lceil d \rceil. \end{aligned}$$

  A shortest tour in this graph now corresponds to a permutation that minimizes Problem (5).

We can thus determine a lower bound for the minimal distance along the y-axis, necessary to place $\lceil d \rceil$ consecutive non-overlapping circles with diameter d. Step 2 is now to use $Y(d)$ or $\widetilde{Y}(d)$ to find an upper bound for the maximin distance of a two-dimensional LHD. When we base our upper bound on $Y(d)$, define d_n as follows:

$$\begin{aligned} d_n = \quad & \min \quad d \\ & \text{s.t.} \quad Y(d) + \left\lfloor \tfrac{n}{\lceil d \rceil} \right\rfloor - 1 > n - 1 \\ & \qquad\quad d^2 \in \mathbb{N}_+. \end{aligned} \tag{7}$$

Proposition 5. *Let d_n^{*2} be the largest sum of two squares that is strictly smaller than d_n^2. Then the value d_n^* is an upper bound for the separation distance of a two-dimensional LHD of n points.*

Proof. First determine for a given number of points n whether an LHD with ℓ^2-distance d can possibly exist. For given n and d, we do this as follows. The maximal number of mutually disjoint subsets of $\lceil d \rceil$ consecutive circles is given by $\left\lfloor \frac{n}{\lceil d \rceil} \right\rfloor$. For each subset, $Y(d)$ is a lower bound for the distance along the y-axis necessary to place the points in the subset. Due to the non-collapsingness criterion, the y-value of the point in a subset with the smallest y-value must be different for each subset. Therefore, we need at least $Y(d) + \left\lfloor \frac{n}{\lceil d \rceil} \right\rfloor - 1$ distance along the y-axis to construct an LHD of n points with ℓ^2-distance d. By solving Problem (7), we find the minimal d for which this minimal required distance is larger than $n-1$, i.e. for which $Y(d) + \left\lfloor \frac{n}{\lceil d \rceil} \right\rfloor - 1 > n - 1$ holds. We thus know that d_n is the minimal d for which our method shows that no LHD with maximin distance d_n exists. By taking d_n^{*2} equal to the largest sum of two squares that is strictly smaller than d_n^2, we have an upper bound for the maximin distance of a two-dimensional LHD. □

We can also determine an upper bound by replacing $Y(d)$ in Problem (7) with $\widetilde{Y}(d)$ or any other lower bound on $Y(d)$. By doing so, we will find an upper bound that is at most as good as the bound based on $Y(d)$.

As we expect that the separation distance of two-dimensional maximin LHDs is non-decreasing in n, we would like the upper bound to have this same property. The following lemma shows that the upper bound d_n^* indeed has this property.

Lemma 4. *The upper bound d_n^* is non-decreasing in n.*

Proof. To show that the bound is non-decreasing in n, we use the fact that d_n is the smallest d for which $Y(d) + \left\lfloor \frac{n}{\lceil d \rceil} \right\rfloor - 1 > n - 1$ holds. As $d_n^* < d_n$, we thus know that:

$$Y(d_n^*) + \left\lfloor \frac{n}{\lceil d_n^* \rceil} \right\rfloor - 1 \leq n - 1,$$

which implies that:

$$Y(d_n^*) + \left\lfloor \frac{n+1}{\lceil d_n^* \rceil} \right\rfloor - 1 \leq Y(d_n^*) + \left\lfloor \frac{n}{\lceil d_n^* \rceil} \right\rfloor \leq n.$$

This means that d_n^* does not satisfy the constraint of Problem (5) for $n+1$. Therefore, $d_{n+1} > d_n^*$. Recall that d_{n+1}^{*2} is obtained by rounding down $d_{n+1}^2$to the largest sum of two squares that is strictly smaller than d_{n+1}^2. As d_n^{*2} is a sum of two squares that is strictly smaller than d_{n+1}^2, we can conclude that $d_{n+1}^* \geq d_n^*$. Hence, the upper bound d_n^* based on $Y(d)$ is non-decreasing in n. $\square$

The same holds for the upper bound based on $\widetilde{Y}(d)$ or any other lower bound on $Y(d)$.

2.2.2 Numerical results

The MIP formulation in (4) was used to determine $Y(d)$ for $d^2 = 2, \ldots, 144$. To solve the MIP, we implemented it in AIMMS (Bisschop and Entriken 1993) and used the CPLEX 9.1 solver. All calculations were done on a PC with a 2.40-GHz Pentium IV processor. Small values of d^2 required less than a second to solve but the largest d^2 required two days.

The TSP formulation was used to determine $\widetilde{Y}(d)$ for $d^2 = 2, \ldots, 665$. The TSP problem is symmetric, which enabled us to use the algorithm described by Volgenant and Jonker (1982). Their exact algorithm is based on the 1-tree relaxation in a branch-and-bound algorithm. We used the implementation provided and described in Volgenant (1990). Most cases were solved in less than a minute, but a few of the larger cases required a few hours to solve.

With the obtained values for $Y(d)$ and $\widetilde{Y}(d)$, we determined upper bounds for $n = 2, \ldots, 114$ and $n = 2, \ldots, 529$, respectively. All bounds can be found in the appendix. For $n = 2, \ldots, 70$, Van Dam et al. (2007) determined optimal maximin designs using branch-and-bound techniques. In Table 5 a comparison is made between the upper bounds and the d- and d^2-values of these optimal maximin LHDs.

	Average % above optimal d	Average % above optimal d^2	Number of tight cases
Oler bound	22.24	50.26	0
Bound based on $Y(d)$	5.77	12.04	12
Bound based on $\widetilde{Y}(d)$	6.44	14.47	12

Table 5: Comparison between bounds and optimal maximin LHDs for $n = 2, \ldots, 70$, as given in Van Dam et al. (2007).

The table shows that the new bounds are a considerable improvement when compared to the Oler bound for smaller values of n. By definition, the bound based on $Y(d)$ is always at least as good as the bound based on $\widetilde{Y}(d)$. For $n = 2, \ldots, 70$, the bound based on $Y(d)$ is tighter for 13 values of n.

As we may not have optimal maximin LHDs for $n > 70$, we compare the upper bounds with the approximate maximin LHDs in Van Dam et al. (2007). These designs are found by considering periodic designs. Table 6 shows that the new bounds are still better than the Oler bound, but the differences are smaller.

	Average % above best known d	Average % above best known d^2
Oler bound	18.49	41.16
Bound based on $Y(d)$	5.89	12.26
Bound based on $\widetilde{Y}(d)$	6.51	13.58

Table 6: Comparison between bounds and approximate maximin LHDs for $n = 2, \ldots, 114$, as given in Van Dam et al. (2007).

For $n = 115, \ldots, 529$, we can only compare the Oler bound and the bound based on $\widetilde{Y}(d)$. In Table 7 the comparison is made for different intervals of n. We see that the Oler bound becomes relatively better as n increases.

size n	$[2, 100]$	$[101, 200]$	$[201, 300]$	$[301, 400]$	$[401, 500]$	$[501, 529]$
Oler bound	22.40	10.45	7.75	6.46	5.74	5.66
Bound based on $\widetilde{Y}(d)$	6.65	6.31	5.85	5.72	5.70	5.95

Table 7: Average % above d of approximate maximin LHDs as given in Van Dam et al. (2007).

The bound based on $\widetilde{Y}(d)$ is at least as good as the Oler bound for $n = 2, \ldots, 410$. For $n = 411, \ldots, 415$, sometimes one bound is better and sometimes the other. For values of $n \geq 416$, the Oler bound is at least as good as the $\widetilde{Y}(d)$ based bound. This has two reasons. Firstly, the LHD becomes more similar to an unrestricted design as n increases. Since the original Oler bound is intended for unrestricted designs, it is to be expected that the Oler bound becomes better as n increases. Secondly, the definition of $\widetilde{Y}(d)$ allows certain circles to overlap. When d becomes larger, this will occur more and more frequently. The bound based on $\widetilde{Y}(d)$ is thus weaker for large n. However, in practice LHDs are used for relatively small values of n (several dozens) which makes this drawback less relevant.

3 Upper bounds for the ℓ^∞-distance

3.1 Bounding by graph covering

For the ℓ^∞-distance we obtain a bound for the separation distance of an LHD as follows.

Proposition 6. *Let D be an LHD of n points and dimension k. Then the separation ℓ^∞-distance d satisfies*

$$k(n-d)(n-d+1) \geq n(n-1).$$

Proof. In each coordinate $(n-d)(n-d+1)/2$ pairs of points have distance at least d. Since each pair of points must be separated by a distance at least d in at least one of the coordinates, it follows that $k(n-d)(n-d+1)/2 \geq n(n-1)/2$. $\square$

As for the bound for the ℓ^2-case in Section 2.1, this bound does not seem to be of the right order if k is fixed. For example, if $k = 2$, then the inequality in the proposition is satisfied if $n \geq 4d$. However, the maximin distance satisfies $d = \lfloor \sqrt{n} \rfloor$, cf. Van Dam et al. (2007).

Rather than finding the maximin distance given n and k, it seems more convenient here to find the smallest $k = k_{\min}(n, d)$ for which an LHD of size n and dimension k with separation distance d exists. The proof of Proposition 6 suggests to formulate the problem as a graph covering problem. Consider the complete graph on n vertices (representing the points of the design). Each edge of this graph (representing a pair of points) must be covered by one of k subgraphs of a particular form. For each coordinate this graph has as edges those pairs of points that are at distance at least d in this coordinate. These subgraphs are all isomorphic copies of the graph that can be described as follows: the vertices are the points $0, 1, \ldots, n-1$, and two points are adjacent if their absolute difference is at least d. Thus the problem can now be reformulated as to find the minimal number of copies of a graph $G(n, d)$ that cover all edges of the complete graph K_n. Such graph covering problems are not studied much. However, graph partitioning problems, where the complete graph must be partitioned into (the right number of) copies of a given graph, are, cf. Heinrich (1996). Of course, if such a partitioning exists, then it is a minimal covering.

The graphs $G(n, d)$ that are of interest to us depend only on the difference between n and d, so it makes sense to fix this difference. To start off easy, let $d = n - 1$ (which is extremal). The graph $G(n, d)$ now consists of a single edge (and some isolated vertices that we may discard), and it is clear that we can cover (partition) the edges of the complete graph K_n by $\binom{n}{2}$ copies. Thus the above bound is tight, and we have the following proposition.

Proposition 7. *For $d = n - 1$, the smallest $k = k_{\min}(n, d)$ for which an LHD of size n and dimension k with separation ℓ^∞-distance d exists, satisfies $k_{\min}(n, n-1) = \binom{n}{2}$.*

For $d = n-2$, we have the following.

Proposition 8. *For $d = n-2$, the smallest $k = k_{\min}(n,d)$ for which an LHD of size n and dimension k with separation ℓ^∞-distance d exists, satisfies $k_{\min}(n, n-2) = \lceil \frac{n(n-1)}{6} \rceil$.*

Proof. Let $d = n-2$, then the graph $G(n,d)$ is a path P_4 of 4 vertices (again we may discard the isolated vertices). Bermond and Sotteau (1976) showed that if $n(n-1)$ is a multiple of 6, then K_n can be partitioned into copies of P_4. It is straightforward to extend their result to minimal coverings, that is, K_n can be covered by $k_{\min}(n, n-2) = \lceil \frac{n(n-1)}{6} \rceil$ copies of P_4. Thus for LHDs of size n and separation distance $d = n-2$ we need precisely this many dimensions. $\square$

For $d = n-3$ we were able to show the following. We omit the proof, which is similar (but more technical) to the proof of the case $d = n-2$.

Proposition 9. *For $d = n-3$, the smallest $k = k_{\min}(n,d)$ for which an LHD of size n and dimension k with separation ℓ^∞-distance d exists, satisfies $k_{\min}(n, n-3) = \lceil \frac{n(n-1)}{12} \rceil$.*

For smaller d the situation becomes more complicated. For large n we have the following: if $e = n-d$ is fixed, then by a result of Wilson (1976) there is a function $N(e)$ such that a partition of K_n into copies of $G(n, n-e)$ exists if $n(n-1)$ is a multiple of $(n-d)(n-d+1)$ and $n > N(e)$. Thus in those cases $k_{\min}(n, d = n-e) = \frac{n(n-1)}{e(e+1)}$.

For $n \leq 10$ and all d it is possible to construct LHDs with dimension k meeting the lower bound $\lceil \frac{n(n-1)}{(n-d)(n-d+1)} \rceil$, except in the cases $n = 8, d = 3$ and $n = 10, d = 5$. For the first exception, $k = 2$ cannot be attained, since in two dimensions we have $n \geq d^2$, cf. Van Dam et al. (2007). For the second exception, $k = 3$ cannot be attained, as was found by a complete search by computer (see also Section 5.1).

3.2 Attaining Baer's bound

Baer (1992) showed that the maximin ℓ^∞-distance d for unrestricted designs of n points on $[0, n-1]^k$ equals $\frac{n-1}{\lfloor (n-1)^{1/k} \rfloor}$. For $n = m^k + 1$, this maximin distance equals m^{k-1}. In this section we shall give a construction of maximin LHDs of $n = m^k$ points with separation distance m^{k-1}, and show that n cannot be smaller to achieve this separation distance. First, we need the following lemma.

Lemma 5. *For the ℓ^∞-distance, the maximin distance d for LHDs of n points and dimension k is a non-decreasing function of n.*

Proof. Consider an LHD D of size n and separation distance d. Let the point of D with first coordinate $n-d$ have remaining coordinates $x_2, x_3, \ldots, x_k$. Now construct D' from D by increasing by one all coordinates that are at least $x_2, x_3, \ldots, x_k$, respectively, and adding the point $(n, x_2, x_3, \ldots, x_k)$. Then D' is an LHD of size $n+1$ with the same separation distance d as D, which proves the lemma. $\square$

From this lemma and the above observation it follows that an LHD of $n = m^k$ points and dimension k has separation distance at most m^{k-1}. We shall now give a construction of an LHD attaining that upper bound.

Construction 1. *Let $m \geq 2$ and $k \geq 1$ be integers, and let $n = m^k$. For $\mathbf{a} = (a_1, a_2, \ldots, a_k) \in \{0, 1, \ldots, m-1\}^k$ and $j = 1 \ldots, k$, let $\mathbf{x}(\mathbf{a}) = (x_1, x_2, \ldots, x_k)$, where*

$$x_j = \sum_{i=k-j}^{k-1} a_{i+1-k+j} m^i + m^{k-j} - 1 - \sum_{i=0}^{k-j-1} a_{k-i} m^i,$$

and let D be the design $D = \{\mathbf{x}(\mathbf{a}) \mid \mathbf{a} \in \{0, 1, \ldots, m-1\}^k\}$.

Examples of this construction are given in the following tables.

a_1	0	1	0	1	0	1	0	1
a_2	0	0	1	1	0	0	1	1
a_3	0	0	0	0	1	1	1	1
x_1	3	7	1	5	2	6	0	4
x_2	1	3	5	7	0	2	4	6
x_3	0	1	2	3	4	5	6	7

Table 8: Construction 1 for $m = 2$ and $k = 3$.

a_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
a_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
a_3	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
a_4	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x_1	7	15	3	11	5	13	1	9	6	14	2	10	4	12	0	8
x_2	3	7	11	15	1	5	9	13	2	6	10	14	0	4	8	12
x_3	1	3	5	7	9	11	13	15	0	2	4	6	8	10	12	14
x_4	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 9: Construction 1 for $m = 2$ and $k = 4$.

Proposition 10. *The design D from Construction 1 is an LHD of $n = m^k$ points and dimension k with maximin ℓ^∞-distance $d = m^{k-1}$.*

Proof. One can check that for each j, the map sending $\mathbf{a}$ to x_j is a one-to-one map from $\{0, 1, \ldots, m-1\}^k$ to $\{0, 1, \ldots, n-1\}$. Thus D is an LHD on n points. Next, observe the recursive structure of the construction. For fixed m, each point $\mathbf{x}'(a_1, a_2, \ldots, a_{k-1}) = (x'_1, x'_2, \ldots, x'_{k-1})$ of the LHD in dimension $k-1$, and each value a_k determines a point $\mathbf{x}(a_1, a_2, \ldots, a_k) = (x_1, x_2, \ldots, x_k)$ in the LHD in dimension k, where $x_j = m(x'_j + 1) - a_k - 1$, for $j = 1, \ldots, k-1$, and $x_k = \sum_{i=0}^{k-1} a_{i+1} m^i$. For an example, see the constructed designs in Tables 8 and 9. We shall use this recursion now to prove by induction on k that the separation distance is $d = m^{k-1}$. Of course this is trivial for $k = 1$, the basis for induction. Now suppose that the statement is true for $k-1$. Let $\mathbf{a}$ and $\mathbf{b} \in \{0, 1, \ldots, m-1\}^k$, and consider the corresponding design points $\mathbf{x}(\mathbf{a}) = (x_1, x_2, \ldots, x_k)$ and $\mathbf{y}(\mathbf{b}) = (y_1, y_2, \ldots, y_k)$, respectively. If x_k and y_k differ by at least d, then we are done, hence we may assume that they differ by at most $d-1$. Then it follows that a_k and b_k differ by at most one. Since the points $\mathbf{x}' = (\frac{1}{m}(x_j + a_k + 1) - 1)_{j=1,\ldots,k-1}$ and $\mathbf{y}' = (\frac{1}{m}(y_j + b_k + 1) - 1)_{j=1,\ldots,k-1}$ are in the LHD of dimension $k-1$ (as explained above), which by assumption has separation distance m^{k-2}, it follows that if $a_k = b_k$, then $\mathbf{x}$ and $\mathbf{y}$ are at distance at least $m \cdot m^{k-2} = d$. Moreover, if a_k and b_k differ by one, say (without loss of generality) that $a_k = b_k + 1$, then the points $\mathbf{x}'$ and $\mathbf{y}'$ are at distance at least m^{k-2}. If this distance is at least $m^{k-2} + 1$, then $m(\mathbf{x}' + 1)$ and $m(\mathbf{y}' + 1)$ have distance at least $d + m$, hence $\mathbf{x}$ and $\mathbf{y}$ are at distance at least $d + m - 1 \geq d$. If the aforementioned distance between $\mathbf{x}'$ and $\mathbf{y}'$ is however exactly m^{k-2}, then $(a_1, a_2, \ldots, a_{k-1})$ and $(b_1, b_2, \ldots, b_{k-1})$ must differ (by one) in exactly one coordinate, say the t-th one. If $a_t = b_t - 1$, then $y_t - x_t = m^{k-1} - b_k + a_k = d + 1$; otherwise $a_t = b_t + 1$, and then $x_k = y_k + m^{k-1} + m^{t-1} > d$, and so in any case $\mathbf{x}$ and $\mathbf{y}$ have distance at least d. The statement now follows by induction. $\square$

In fact, we can slightly generalize the above result.

Proposition 11. *Let $m \geq 2$, $k \geq 2$, and $t \leq m$ be nonnegative integers, and let $n = m^k + t$. Then the maximin distance d for LHDs of n points and dimension k satisfies $d = m^{k-1}$.*

Proof. It follows from Lemma 5 and Proposition 10 that the maximin distance is at least as stated. From Baer's upper bound $\frac{n-1}{\lfloor (n-1)^{1/k} \rfloor}$ (rounded down) it follows that it is at most as stated. $\square$

Observe also that if D is an LHD with separation distance d, and we remove an arbitrary point $(x_1, x_2, \ldots, x_k)$ from D, and from the remaining points in D we decrease by one all coordinates that are larger than $x_1, x_2, \ldots, x_k$, respectively, then we obtain an LHD of size $n-1$ with separation distance at least $d-1$. Thus the maximin distance cannot increase by more than one as n increases by one. We now show that the above construction is extremal in the sense that we cannot decrease n and still achieve the same maximin distance.

Proposition 12. *Let $m \geq 2$ and $k \geq 2$ be integers, and let $n = m^k - 1$. Then the maximin ℓ^∞-distance d for LHDs of n points and dimension k satisfies $d = m^{k-1} - 1$.*

Proof. By the above observation and Proposition 10 it suffices to prove that an LHD on n points cannot have separation distance $d = m^{k-1}$. Suppose on the contrary that we have such an LHD. Partition the set $I = \{0, 1, \ldots, n-1\}$ into m parts: $I_i = \{id, id+1, \ldots, id+d-1\}, i = 0, \ldots, m-2$ (each of cardinality d), and $I_{m-1} = \{(m-1)d, (m-1)d+1, \ldots, (m-1)d+d-2\}$ (of cardinality $d-1$). Accordingly, partition the set I^k into m^k parts $I_{i_1} \times I_{i_2} \times \cdots \times I_{i_k}$. In each of these parts the points are at mutual distance at most $d-1$, hence each part contains at most one design point. Suppose now that the part $I_{i_1} \times I_{i_2} \times \cdots \times I_{i_k}$ does not contain a design point. Since $I_{i_1} \times I \times \cdots \times I$ then contains $|I_{i_1}|$ points on one hand, and at most $m^{k-1} - 1 = d - 1$ on the other hand, this implies that $i_1 = m-1$. Similarly it follows that $i_2 = \cdots = i_k = m-1$, hence all parts except $I_{m-1} \times I_{m-1} \times \cdots \times I_{m-1}$ contain precisely one point.

Now consider a slightly different partition of I, i.e., into parts $J_i = I_i$, for $i = 0, \ldots, m-3$, $J_{m-2} = I_{m-2} \setminus \{(m-1)d - 1\}$, and $J_{m-1} = \{(m-1)d - 1\} \cup I_{m-1}$. By considering the partition of I^k into parts $J_{i_1} \times I_{i_2} \times \cdots \times I_{i_k}$, it follows that $J_{m-1} \times I_{m-1} \times \cdots \times I_{m-1}$ contains precisely one design point. Similarly $I_{m-1} \times J_{m-1} \times \cdots \times I_{m-1}$ contains precisely one design point. Since $I_{m-1} \times I_{m-1} \times \cdots \times I_{m-1}$ does not contain a design point, these two points must be distinct. However, both are contained in $J_{m-1} \times J_{m-1} \times I_{m-1} \times \cdots \times I_{m-1}$, which contradicts the fact that also this part can contain at most one design point. □

3.3 Bounding by projection and partitioning in three dimensions

Consider a three-dimensional LHD of n points (x_i, y_i, z_i), $i = 1, \ldots, n$, with ℓ^∞-distance d. Now, project all design points for which $z_i \leq d - 1$ onto the (x, y)-plane. Since the z-values of all these design points differ less than d, the differences of the x- or y-values should at least be d for all points, i.e. the projected points form a two-dimensional design with separation distance d. The same holds for any other "layer" within the three-dimensional LHD for which the z-values of the design points differ less than d. By taking the right layer and further analyzing the projected design we obtain the following proposition.

Proposition 13. *For integers $n \geq 3$ and $d \geq 2$, let $N(n, d)$ be given by*

$$N(n,d) = \sum_{i=1}^{\lfloor \frac{n}{d} \rfloor} \left(\left\lfloor \frac{n - \lfloor \frac{n}{d} \rfloor - i + 1}{d} \right\rfloor + 1 \right) + \min \left\{ n - d \lfloor \frac{n}{d} \rfloor, \ \left\lfloor \frac{n - 2\lfloor \frac{n}{d} \rfloor}{d} \right\rfloor + 1 \right\}. \tag{8}$$

The maximal d such that $d \leq N(n, d)$ is an upper bound for the ℓ^∞-maximin distance $d_{\max}$ for a three-dimensional LHD of n points.

Proof. Consider the mutually disjoint "layers" $I_i = \{id, id+1, \ldots, id+d-1\}, i = 0, \ldots, \lfloor \frac{n}{d} \rfloor - 1$, of z-values of the LHD. Among these $\lfloor \frac{n}{d} \rfloor$ layers there must be at least one for which the corresponding projected design (as described above) has all its x-values at most $n - \lfloor \frac{n}{d} \rfloor$ (since all x-values are distinct). The projection of this layer will be onto the $(n - \lfloor \frac{n}{d} \rfloor) \times (n-1)$-grid; see Figure 3.

In Figure 3, one can identify $\lfloor \frac{n}{d} \rfloor$ mutually disjoint strips of size $(n - \lfloor \frac{n}{d} \rfloor) \times (d-1)$. Furthermore, since the differences in y-values within each strip are less than d, the x-values have to differ at least d, and, hence, the first strip contains at most $\lfloor \frac{n - \lfloor \frac{n}{d} \rfloor}{d} \rfloor + 1$ points. Moreover, since all x-values are distinct, the second strip contains at most $\lfloor \frac{n - \lfloor \frac{n}{d} \rfloor - 1}{d} \rfloor + 1$ points, the third strip at most $\lfloor \frac{n - \lfloor \frac{n}{d} \rfloor - 2}{d} \rfloor + 1$ points, et cetera.

When n is not divisible by d, the remaining "partial" strip contains at most $\lfloor \frac{n - 2\lfloor \frac{n}{d} \rfloor}{d} \rfloor + 1$ points, but also at most $n - d\lfloor \frac{n}{d} \rfloor$ points. Note that in case n is divisible by d (and there is no remaining strip), the latter term is equal to 0 and the former term is non-negative (since $d \geq 2$). Thus, $N(n, d)$ is an upper bound for the number of points in the projected design, and the result follows. □

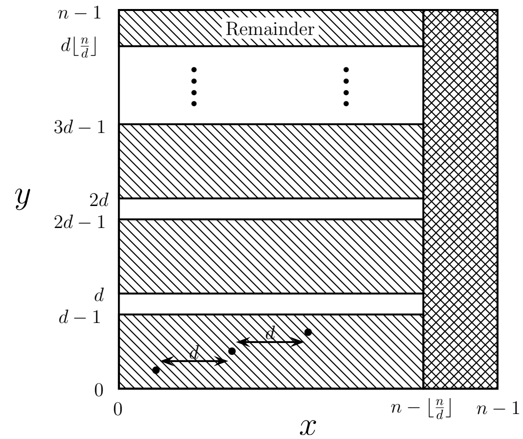

Figure 3: Strips within the projection onto the (x, y)-plane.

For values of n up to 165, the corresponding upper bounds are provided in Table 10. For many values of n the bound is better than Baer's bound. They also confirm Proposition 12 for $k = 3$ and $m \leq 5$.

$n \leq$	3	5	10	13	15	18	21	30	34	38	41	45	49	53	68
$d_{\max} \leq$	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$n \leq$	73	78	83	87	92	97	102	107	130	136	142	148	154	159	165
$d_{\max} \leq$	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Table 10: Upper bound for ℓ^∞-maximin distance $d_{\max}$ for several n.

4 Upper bounds for the ℓ^1-distance

In this section we apply the ideas of Section 2.1 to the ℓ^1-distance. Bounding by the average gives the following bound.

Proposition 14. *Let D be an LHD of n points and dimension k. Then the separation ℓ^1-distance d satisfies*

$$d \leq \left\lfloor \frac{(n+1)k}{3} \right\rfloor.$$

Proof. Let $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, with $\mathbf{x}_i = (x_{i1}, \ldots, x_{ik})$. The average distance among the points of D is

$$\frac{1}{\binom{n}{2}} \sum_{i>j} \sum_{h} |x_{ih} - x_{jh}| = \frac{1}{\binom{n}{2}} \sum_{h} \sum_{i>j} |x_{ih} - x_{jh}| = \frac{1}{\binom{n}{2}} \sum_{h} \sum_{i'>j'} (i' - j') = k(n+1)/3,$$

and rounding finishes the proof. □

Similar remarks as in the ℓ^2-case apply here. More evidence for the fact that the bound is not of the right order to be tight if k is fixed is given by the case $k = 2$, where the maximin distance is known to be $\lfloor \sqrt{2n+2} \rfloor$, cf. Van Dam et al. (2007). The analogue of Lemma 1 is the following.

Lemma 6. *Let $d_{\max}(n, k)$ be the maximin ℓ^1-distance of an LHD of n points and dimension k. Then $d_{\max}(n, k_1 + k_2) \geq d_{\max}(n, k_1) + d_{\max}(n, k_2)$.*

We can write the maximin distance d as the solution of the following integer programming problem.

$$\begin{array}{ll} \max & d \\ \text{s.t.} & \sum_{\pi \in S_n} k_\pi |\pi(i) - \pi(j)| \geq d, \quad \forall\ i > j \\ & \sum_{\pi \in S_n} k_\pi = k \\ & k_\pi \in \mathbb{N}_0, \quad \forall\ \pi \in S_n \end{array} \tag{9}$$

where S_n is the set of permutations of $\{1, 2, \ldots, n\}$. As before, we may restrict the set S_n to its first half when ordered lexicographically, and we may assume that $k_{\pi^*} \geq 1$ for an arbitrary permutation π^*.

Also here we consider the cases $n = 3, 4$, and 5 to show the strength of the bound in Proposition 14.

Proposition 15. *For $n = 3$, the maximin ℓ^1-distance satisfies $d_{\max}(3, k) = \lfloor \frac{4k}{3} \rfloor$.*

Proof. The stated result follows from solving the above integer programming problem (9) by hand (the number of variables is 3). Alternatively, it also follows by using the upper bound and recursively applying Lemma 6 starting from $d_{\max}(3, 1) = 1$, $d_{\max}(3, 2) = 2$ (both trivial), and $d_{\max}(3, 3) = 4$. The latter is attained by the design $\{(0, 1, 2), (1, 2, 0), (2, 0, 1)\}$. □

Proposition 16. *For $n = 4$, the maximin ℓ^1-distance satisfies $d_{\max}(4, k) = \lfloor \frac{5k}{3} \rfloor - 1$ if $k \equiv 3 \pmod 6$, and $d_{\max}(4, k) = \lfloor \frac{5k}{3} \rfloor$ otherwise.*

Proof. First, we show that the upper bound $\lfloor \frac{5k}{3} \rfloor$ cannot be attained if $k \equiv 3 \pmod 6$. Suppose that k is a multiple of 3, and that an LHD with separation distance $d = 5k/3$ exists. This implies that all points in the design are at equal distance. Fix one point, and let k_0 be the number of coordinates where this point is 0 or 3, and let $k_1 = k - k_0$ be the number of coordinates where it is 1 or 2. It follows that the average distance of this point to the other points equals $2k_0 + \frac{4}{3}k_1 = \frac{5}{3}k$. It now follows that $k_1 = k/2$, hence k should be even. Thus, for $k \equiv 3 \pmod 6$ the bound cannot be attained.

By solving the integer programming problem (9) for $k \leq 6$ by computer and using Lemma 6 (with $k_2 = 6$), we then find that the upper bound $\lfloor \frac{5k}{3} \rfloor$ is attained for all k except for $k \equiv 3$ (mod 6), and that for these exceptions the maximin distance is one less. □

Proposition 17. *For $n = 5$, the maximin ℓ^1-distance satisfies $d_{\max}(5, k) = 2k - 1$ if $k \leq 4$ or $k = 7$, and $d_{\max}(5, k) = 2k$ otherwise.*

Proof. We first show that the bound $2k$ cannot be attained for $k \leq 4$ and $k = 7$. If the bound is attained, then all points of the design are at equal distance. Fix a point, let k_0 be the number of coordinates where this point is 0 or 4, let k_1 be the number of coordinates where it is 1 or 3, and let k_2 be the number of coordinates where it is 2. It follows that the average distance of this point to the other points equals $\frac{10}{4}k_0 + \frac{7}{4}k_1 + \frac{6}{4}k_2 = 2k$. Since $k_0 + k_1 + k_2 = k$, it follows that $k_1 + \frac{4}{3}k_2 = \frac{2}{3}k$. For $2 \leq k \leq 4$ and $k = 7$, there is a unique nonnegative integer solution (k_0, k_1, k_2) to these equations, and so each point has the same number k_2 of coordinates where this point is 2. This implies that the total number of coordinates where a 2 occurs equals $5k_2$ on one hand, and k on the other hand. This gives a contradiction in these cases.

By solving the integer programming problem (9) for $k \leq 9$ by computer, and using Lemma 6 (with $k_2 = 5$ or 6), we then find that the upper bound $2k$ is attained for all k except for $k \leq 4$ and $k = 7$, and that for these exceptions the maximin distance is one less than the given upper bound. □

Also for $n = 6$ and $n = 7$ we computed the integer programming problem (9) for $k \leq 20$, see the tables below. Note that for some values of k only a lower bound was obtained. The tables show that also here the upper bound is attained for many values of k. In particular it follows that for $n = 6$, the upper bound $\lfloor \frac{7k}{3} \rfloor$ is attained for all $k \equiv 0, 1, 2, 5 \pmod 6$, except for $k = 1, 2$ and possibly for $k = 7$. For $n = 7$, the upper bound $\lfloor \frac{8k}{3} \rfloor$ is attained for all $k \equiv 0, 1 \pmod 3$, except for $k = 1$ and 3.

dimension k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$d_{\max}(6, k)$	1	3	6	8	11	14	≥ 15	18	≥ 20	≥ 22	25	28	30	32	≥ 34	≥ 36	39	42	44	46
upper bound	2	4	7	9	11	14	16	18	21	23	25	28	30	32	35	37	39	42	44	46

Table 11: Maximin ℓ^1-distance for LHDs on 6 points.

dimension k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$d_{\max}(7, k)$	1	4	6	10	12	16	18	≥ 20	24	26	≥ 28	32	34	≥ 36	40	42	≥ 44	48	50	≥ 52
upper bound	2	5	8	10	13	16	18	21	24	26	29	32	34	37	40	42	45	48	50	53

Table 12: Maximin ℓ^1-distance for LHDs on 7 points.

5 Final remarks and conclusions

5.1 Final remarks

By a branch-and-bound algorithm we were able to find maximin LHDs in three dimensions for small n and the three distance measures ℓ^2, ℓ^1, and ℓ^∞. The maximin distances are given in the following table.

size n	2	3	4	5	6	7	8	9	10	11	12	13	14	15
squared maximin ℓ^2-distance	3	6	6	11	14	17	21	22	27	30	36	41		
maximin ℓ^1-distance	3	4	4	5	6	6	7	8	8	8	9	10	10	
maximin ℓ^∞-distance	1	2	2	2	3	3	4	4	4	4	5	5	5	6

Table 13: Maximin distances for LHDs in three dimensions.

The corresponding maximin designs, and all other (approximate) maximin LHDs that appeared in this paper can be obtained from the website `http://www.spacefillingdesigns.nl`.

In two dimensions the ℓ^∞-maximin distance is equal to $\lfloor n^{\frac{1}{2}} \rfloor$, cf. Van Dam et al. (2007). The results in three dimensions suggest that the corresponding ℓ^∞-maximin distance equals $\lfloor n^{\frac{2}{3}} \rfloor$. A natural extension would be that the ℓ^∞-maximin distance in k dimensions equals $d = \lfloor n^{\frac{k-1}{k}} \rfloor$. However, this is not the case in general, since for example for the case $n = 17$ and $k = 23$ the optimal distance is smaller than $\lfloor 17^{\frac{22}{23}} \rfloor = 15$ according to Proposition 6. The expression for d may, however, still provide an upper bound for the maximin distance.

Another interesting point is that we conjecture, but were unable to prove, that the analogue of Lemma 5 holds for the ℓ^2- and ℓ^1-distance measure, i.e. that also for these distance measures the maximin distance is non-decreasing in n.

5.2 Conclusions

We have obtained bounds for the separation distance of LHDs for several distance measures. These bounds are useful to assess the quality of approximate maximin LHDs by comparing their separation distances with the corresponding upper bounds. For the ℓ^2- and ℓ^1-distances we obtain bounds by considering the average distance. These bounds are close to tight when the dimension k is relatively large. For the ℓ^2-distance in two dimensions we obtain a method that produces a bound that is better than Oler's bound if the number of points of the LHD is at most 400. For the ℓ^∞-distance we obtain a bound by looking at it as a graph covering problem. Besides this bound we construct maximin LHDs attaining Baer's bound for infinitely many values of n (the number of points) in all dimensions. Finally, we present a method to obtain a bound for three-dimensional LHDs that is better than Baer's bound for many values of n.

Acknowledgements

The authors thank Dick den Hertog for the many inspiring conversations on the topic of this paper, and Dolf Talman for his comments on an earlier version.

References

Baer, D. (1992). Punktverteilungen in Würfeln beliebiger Dimension bezüglich der Maximumnorm. *Wiss. Z. Pädagog. Hochsch. Erfurt/Mühlhausen, Mathematik-Naturwissenschaften Reihe*, **28**, 87–92.

Bermond, J.C. and D. Sotteau (1976). Graph decompositions and G-designs. In Nash-Williams and Sheehan (Eds.), *Proc. 5th British Combinatorial Conf. 1975*, 53–72. Utilitas Mathematica Publ.

Bisschop, J. and R. Entriken (1993). *AIMMS: The Modeling System*. Haarlem, The Netherlands: Paragon Decision Technology.

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

Dam, E.R. van, B.G.M. Husslage, D. den Hertog, and J.B.M. Melissen (2007). Maximin Latin hypercube designs in two dimensions. *Operations Research*, **55**. (to appear).

Driessen, L.T. (2006). *Simulation-based Optimization for Product and Process Design*. Ph. D. thesis, CentER for Economic Research, Tilburg University.

Heinrich, K. (1996). Graph decompositions and designs. In Colbourn and Dinitz (Eds.), *Handbook of Combinatorial Designs*, 361–365. CRC Press.

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

Husslage, B.G.M. (2006). *Maximin Designs for Computer Experiments*. Ph. D. thesis, CentER for Economic Research, Tilburg University.

Husslage, B.G.M., G. Rennen, E.R. van Dam, and D. den Hertog (2006). Space-filling Latin hypercube designs for computer experiments. *CentER Discussion Paper 2006-18*. Tilburg University.

Jin, R., W. Chen, and A. Sudjianto (2005). An efficient algorithm for constructing optimal design of computer experiments. *Journal of Statistical Planning and Inference*, **134(1)**, 268–287.

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

Jones, D., M. Schonlau, and W.J. Welch (1998). Efficient global optimization of expensive blackbox functions. *Journal of Global Optimization*, **13**, 455–492.

Montgomery, D.C. (1984). *Design and Analysis of Experiments* (Second ed.). New York: John Wiley & Sons.

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

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

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

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

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

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

Stinstra, E. (2006). *The META-model Approach for Simulation-based Design Optimization*. Ph. D. thesis, CentER for Economic Research, Tilburg University.

Volgenant, A. (1990). Symmetric traveling salesman problems. *European Journal of Operational Research*, **49(1)**, 153–154.

Volgenant, A. and R. Jonker (1982). A branch and bound algorithm for the symmetric traveling salesman problem based on the 1-tree relaxation. *European Journal of Operational Research*, **9**, 83–89.

Wilson, R.M. (1976). Decompositions of complete graphs into subgraphs isomorphic to a given graph. In Nash-Williams and Sheehan (Eds.), *Proc. 5th British Combinatorial Conf. 1975*, 647–659. Utilitas Mathematica Publ.

Ye, K.Q., W. Li, and A. Sudjianto (2000). Algorithmic construction of optimal symmetric Latin hypercube designs. *Journal of Statistical Planning and Inference*, **90(1)**, 145–159.

Appendix: Upper bounds on two-dimensional ℓ^2-maximin LHDs

n	Oler	$d_n^{*2}(Y(d))$	$d_n^{*2}(\widetilde{Y}(d))$	d^2
2	5	2	2	2*
3	5	2	2	2*
4	8	5	5	5*
5	10	5	5	5*
6	10	5	5	5*
7	13	8	8	8*
8	13	8	8	8*
9	17	10	10	10*
10	18	13	13	10*
11	20	13	13	10*
12	20	13	13	13*
13	20	13	13	13*
14	25	17	17	17*
15	26	17	17	17*
16	26	18	18	17*
17	29	20	20	18*
18	29	20	20	18*
19	32	25	25	18*
20	32	25	25	18*
21	34	25	25	20*
22	34	26	26	25*
23	37	29	29	26*
24	37	29	29	26*
25	40	29	29	26*
26	41	29	29	26*
27	41	32	32	26*
28	41	34	34	29*
29	45	34	34	29*
30	45	34	34	29*
31	45	34	37	32*
32	45	37	40	32*
33	50	40	40	34*
34	52	41	41	37*
35	53	41	41	37*
36	53	41	41	37*
37	53	45	45	37*
38	53	45	45	41*
39	58	45	45	41*
40	58	45	50	41*
41	61	45	52	41*
42	61	50	52	41*
43	61	52	52	41*
44	65	52	52	50*
45	65	52	53	50*
46	68	53	53	50*
47	68	58	58	50*
48	68	58	58	50*
49	72	58	58	50*
50	73	61	61	52*
51	74	61	61	52*
52	74	61	65	58*
53	74	61	65	58*
54	74	61	65	58*
55	80	65	65	58*
56	80	65	68	58*
57	82	68	68	58*
58	82	68	73	61*

n	Oler	$d_n^{*2}(Y(d))$	$d_n^{*2}(\widetilde{Y}(d))$	d^2
59	85	73	73	61*
60	85	73	73	65*
61	85	74	74	65*
62	89	74	74	65*
63	90	74	74	65*
64	90	74	74	65*
65	90	80	80	68*
66	90	80	80	68*
67	90	80	82	74*
68	97	80	85	74*
69	98	85	85	74*
70	98	85	85	74*
71	100	85	85	74
72	101	85	89	74
73	101	85	89	74
74	104	89	89	74
75	106	89	90	80
76	106	90	90	85
77	106	97	97	85
78	109	97	97	85
79	109	97	97	85
80	109	97	97	85
81	113	100	100	85
82	113	100	101	85
83	116	100	104	90
84	117	100	104	90
85	117	100	106	90
86	117	104	106	97
87	117	106	106	97
88	122	106	106	97
89	122	106	109	97
90	125	109	109	98
91	125	109	109	98
92	125	113	113	98
93	128	113	116	100
94	130	116	116	100
95	130	117	117	100
96	130	117	117	101
97	130	117	117	101
98	130	117	122	101
99	136	117	125	101
100	137	117	125	109
101	137	117	125	109
102	137	125	125	113
103	137	125	125	113
104	137	125	130	117
105	137	128	130	117
106	145	130	130	117
107	146	130	130	117
108	146	130	130	117
109	149	130	136	117
110	149	130	136	117
111	149	136	136	128
112	149	136	137	128
113	153	137	137	128
114	153	137	137	128

n	Oler	$d_n^{*2}(\widetilde{Y}(d))$	d^2
120	162	148	128
130	173	160	145
140	185	173	149
150	200	185	170
160	212	202	178
170	225	208	185
180	234	225	202
190	245	234	208
200	261	250	218
210	274	261	241
220	281	274	245
230	298	290	250
240	306	298	269
250	320	314	277
260	333	325	292
270	346	338	305
280	360	349	320
290	370	365	320
300	377	373	338
310	394	388	346
320	405	401	356
330	416	410	370
340	433	425	386
350	445	442	401
360	457	450	409
370	468	464	410
380	481	477	425
390	493	490	442
400	505	505	450
410	514	514	461
420	522	530	466
430	541	544	485
440	549	549	490
450	565	565	509
460	578	580	509
470	586	592	533
480	601	601	545
490	613	617	549
500	626	629	565
510	637	641	578
520	650	656	586
529	661	661	586

Table 14: Oler bound, bounds based on $Y(d)$ and $\widetilde{Y}(d)$, and d^2 of the best know LHD. When an optimal maximin LHD is known, the corresponding d^2 is marked with *.