Low Discrepancy Allocation of Two-Dimensional Data*

Richard Anstee[1], János Demetrovics[2], Gyula O.H. Katona[3], Attila Sali[3] **

[1] Mathematics Department, University of British Columbia
Vancouver, Canada V6T 1Y4
[2] Computer and Automatization Institute of HAS
Budapest, Lágymányosi u. 11 H-1111 Hungary
[3] Alfréd Rényi Institute of Mathematics, Hungarian Academy of Sciences
Budapest P.O.B. 127 H-1364 HUNGARY

Abstract. Fast browsing and retrieval of geographically referenced information requires the allocation of data on different storage devices for concurrent retrieval. By dividing the two-dimensional space into tiles, a system can allow users to specify regions of interest using a query rectangle and then retrieving information related to tiles covered by the query. Suppose that there are m I/O devices. A tile is labeled by i if the data corresponding to this area is stored in the ith I/O device. A labeling is efficient if the difference of the numbers of occurrences of distinct labels in any given rectangle is small. Except for some simple cases this discrepancy exceeds 1. In the present paper constructions are given to make this discrepancy small relative to m. The constructions use latin squares and a lower bound is given, which shows that the constructions are best possible under certain conditions.

1 Introduction

Todays information systems often use the two dimensional screen as a tool for retrieval of detailed data that is associated with a specific part of the screen. A standard example is a geographic database, where first a low resolution map is displayed on the screen and then the user specifies a part of the map that is to be displayed in higher resolution. Another application is when pictures of famous historical monuments or sightseeing spots of an area are to be displayed. Efficient support of such queries is quite important for image databases in particular, and for browsing geographically referenced information in general. In the Alexandria Digital Library project [7] a large satellite image is divided into tiles and each tile is decomposed using wavelet decomposition [8]. A wavelet decomposition of

* The work of the second and third authors was partially supported by the Hungarian National Foundation for Scientific Research grant number T016389, and European Communities (Cooperation in Science and Technology with Central and Eastern European Countries) contract number CIPACT930113

** This research was done while the third author visited The University of British Columbia on a grant of the first author.

an image results in a lower resolution image of the original one together with higher order coefficients that can be used to retrieve higher resolution versions of the same image. Similar approaches are common to other systems for browsing large image databases [4, 5]. A user would usually browse the lower resolution images fast and then specify areas to be displayed in higher resolution. This requires the retrieval of the higher resolution components for the various tiles that overlap with the specific region. For a more detailed review of the current state of art the reader is referred to [1].

In the present paper the model introduced in [1] is analised further. It is assumed that data is associated with the tiles of a two-dimensional grid. The data corresponding to individual tiles is usually large, so it is preferable to store them on parallel I/O devices in such a way, that for a given query, retrieval from these parallel devices can occur concurrently. The ideal situation, when information related to each individual tile could be stored on a distinct I/O device and hence data for any query could be retreived concurrently is not realizable in general, because the number of tiles is usually much larger than the number of I/O devices available. Thus, the only hope is to "spread out" data as evenly as possible. In the following, a measure of optimality of data allocation is defined as smallest possible discrepancy in the number of access requests for different I/O devices for any rectangular set of tiles. Upper bounds for this discrepancy are derived that give simple, but efficient allocation methods. These upper bounds are shown to be best assymptotycally for certain types of data allocation. This could be viewed as a generalization of *strict optimality* of [1].

2 The General Model

Let $\mathcal{R}$ be an $n_1 \times n_2$ array, whose elements (i, j) where $0 \leq i \leq n_1 - 1$ and $0 \leq j \leq n_2 - 1$ are called *tiles*. Each tile is supposed to contain detailed information on the area it covers. For example, if the array is a low resolution image of a geographic region, the higher resolution wavelet coefficients may be associated with the individual tiles. Given two tiles (i_1, j_1) and (i_2, j_2), where $i_1 \leq i_2$ and $j_1 \leq j_2$, two dimensional query is defined by

$$\mathcal{R}[(i_1, j_1), (i_2, j_2)] = \{(i, j) \colon i_1 \leq i \leq i_2 \text{ and } j_1 \leq j \leq j_2\}.$$

This represents a rectangle, whose opposite corners are (i_1, j_1) and (i_2, j_2) and area is $(i_2 - i_1 + 1)\,(j_2 - j_1 + 1)$, the number of tiles contained in the rectangle. To each tile (i, j) in $\mathcal{R}$ is assigned a number $f(i, j)$ from the set $\{1, 2, \ldots, m\}$. The number $f(i, j)$ refers to one of m available I/O devices on which the information related to the given tile is stored. f is called an *m-assignment* for $\mathcal{R}$. The *degree* of k $(1 \leq k \leq m)$ with respect to rectangle $\mathcal{R}[(i_1, j_1), (i_2, j_2)]$ is

$$d_{i_1, j_1, i_2, j_2}(k) = |\{(i, j) \in \mathcal{R}[(i_1, j_1), (i_2, j_2)] \colon f(i, j) = k\}|,$$

that is the number of ocurrences of k as assignments to tiles in the rectangle. An m-assignment is called *d-discrepancy assignment* iff for any given rectangle

$\mathcal{R}[(i_1, j_1), (i_2, j_2)]$

$$\delta(i_1, j_1, i_2, j_2) = \max_k d_{i_1,j_1,i_2,j_2}(k) - \min_k d_{i_1,j_1,i_2,j_2}(k) \leq d$$

holds. $\delta(i_1, j_1, i_2, j_2)$ is called the *discrepancy* of the rectangle $\mathcal{R}[(i_1, j_1), (i_2, j_2)]$. Clearly, d-discrepancy m-assignments with small d are sought for efficient retrieval of data using as many I/O devices concurrently, as possible. The *optimality* $d(m)$ of m is the minimum d, such that a d-discrepancy m-assignment exists for arbitrary n_1 and n_2. 1-discrepancy m-assignments were called *strictly optimal* in [1] and the following theorem was proved.

Theorem 2.1 (Abdel-Ghaffar,El Abbadi'97). *A 1-discrepancy m-assignment exists for an $n_1 \times n_2$ array $\mathcal{R}$ iff one of the following conditions holds:*
- $\min\{n_1, n_2\} \leq 2$,
- $m \in \{1, 2, 3, 5\}$,
- $m \geq n_1 n_2 - 2$,
- $m = n_1 n_2 - 4$ *and* $\min\{n_1, n_2\} = 3$,
- $m = 8$ *and* $n_1 = n_2 = 4$.

Corollary 2.1. $d(m) \geq 2$ *if* $m \notin \{1, 2, 3, 5\}$.

Theorem 2.1 shows that strict optimality can be achieved only in very restricted cases, hence it is natural to ask, how good an assignment can be in general, i.e., good upper bounds for $d(m)$ are of interest. In the rest of the paper d-discrepancy assignments are given, where d is of order of magnitude $\log m$ It is shown to be best possible apart from a multiplicative constant, if the assignment is of *latin square* type.

Because $d(m)$ is defined as the lowest discrepancy that can be achieved for arbitrary n_1 and n_2, we will consider m assignments for an $\infty \times \infty$ array, like an infinite checkerboard covering the plane. In other words, an m-assignment is a map $f: \mathbb{Z} \times \mathbb{Z} \longrightarrow \{1, 2, \ldots, m\}$.

The proof of Theorem 2.1 and most of the previous results use *modular assignments*, i.e. maps of type $f(i,j) = \alpha i + \beta j \bmod m$. Our methods are different: good assignments are constructed for pm provided good ones exist for some m via *blow up* technique. This results in $d(m) = O(\log m)$ for $m = p^t$. Then using special transversals in latin squares the construction is extended for all values of m.

3 The Blow Up

The following construction is crucial in the proofs. Let M be an m-assignment, i.e., an $\infty \times \infty$ array, whose rows and columns are indexed by $\mathbb{Z}$. Furthermore, let $A(p)$ be an $\infty \times p$ array, whose rows are indexed by $\mathbb{Z}$ and each row is a permutation of $\{1, 2, \ldots, p\}$. The *blow up* $A(pM)$ of M by $A(p)$ is defined as follows. In each column of M the i entries are replaced by $i \times$ rows of $A(p)$, i.e., each 1×1 entry i becomes a $1 \times p$ block, a permutation of $\{(i, 1), (i, 2), \ldots, (i, p)\}$. Each i

entry of the given column is mapped this way to a row of $A(p)$, different entries to different rows, and consecutive entries to consecutive rows. For example, if

$$A(p) = \begin{matrix} \vdots \\ 1\,2\,3\;\;4\;\;5\,6\dots p \\ 4\,5\,6\dots p\,1\;\;2\;\;3 \\ \vdots \end{matrix}$$

then the substitution of the 1×1 blocks of i-entries are as follows (* denotes entries different from i)

$$\begin{matrix} \vdots & & \vdots \\ * & & \\ i & \mapsto & (i,1)\;(i,2)\;\dots\;(i,p) \\ * & & \\ i & \mapsto & (i,4)\;(i,5)\;\dots\;(i,3) \\ * & & \\ \vdots & & \vdots \end{matrix}$$

This substitution is performed for each different entries i $(1 \leq i \leq m)$, independently of each other, thus replacing each column of M with p columns, whose entries are from $\{1, 2, \dots, m\} \times \{1, 2, \dots p\}$.

Let us recall that the discrepancy of the (possibly infinite) array M, $\delta(M)$ is defined as the supremum of $\delta(M')$ for finite subrectangles M' of M.

Theorem 3.1. *Let M be an $\infty \times \infty$ m-assignment array of discrepancy $\delta(M)$. Suppose that $A(p)$ is a $\infty \times p$ array whose rows are permutations of $\{1, 2, \dots, p\}$ of discrepancy $\delta(A(p))$. Then*

$$\delta(A(pM)) \leq \delta(M) + 6\,\delta(A(p)).$$

Proof (of Theorem 3.1). Consider a rectangle R in $A(pM)$. It can be decomposed into three parts A, B and C, where B consists of complete blown up columns, A is the "left chunk", and C is the "right chunk", i.e, they consist of only a part of a blow up of one column, respectively, see Figure 1. Let A', B' and C' denote corresponding entries of M. Notice, that A' and C' are single columns. Let $\#_X\alpha$ denote the number of α-entries in the block X of an array. Then the following are immediate facts.

(i) $|\#_{A'}i - \#_{A'}j|$, $|\#_{B'}i - \#_{B'}j|$, $|\#_{C'}i - \#_{C'}j| \leq \delta(M)$
(ii) $|\#_{A'}i + \#_{B'}i - \#_{A'}j - \#_{B'}j|$, $|\#_{B'}i + \#_{C'}i - \#_{B'}j - \#_{C'}j| \leq \delta(M)$
(iii) $|\#_{A'}i + \#_{B'}i + \#_{C'}i - \#_{A'}j - \#_{B'}j - \#_{C'}j| \leq \delta(M)$
(iv) $\#_{B'}i = \#_B(i,k)$ for $k = 1, 2, \dots, p$
(v) $|\#_{ABC}(i,k) - \#_{ABC}(j,l)| \leq |\#_{ABC}(i,k) - \#_{ABC}(i,l)| + |\#_{ABC}(i,l) - \#_{ABC}(j,l)|$

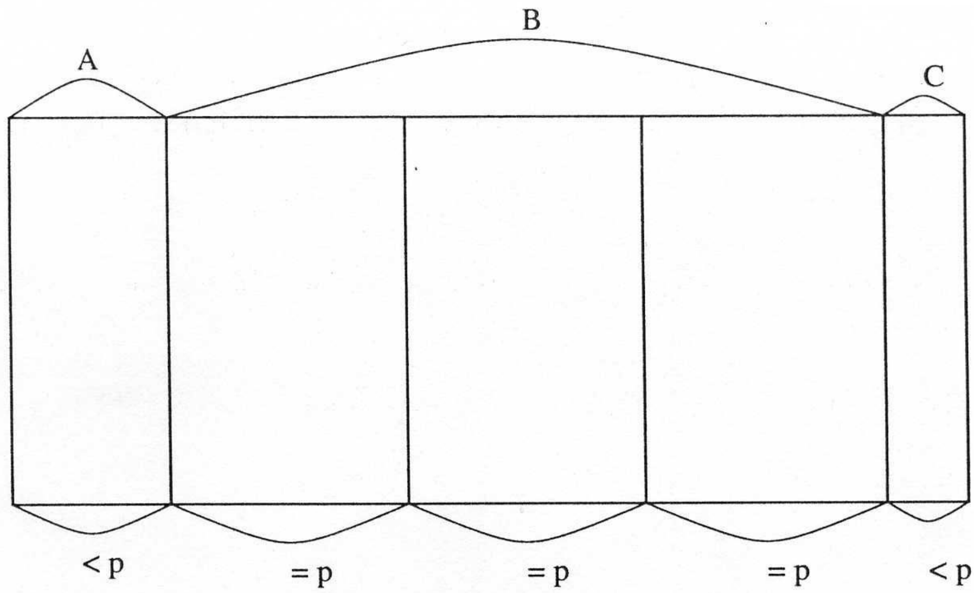

Figure 1.

Now

(vi) $|\#_{ABC}(i,k) - \#_{ABC}(i,l)| \le |\#_A(i,k) - \#_A(i,l)| + |\#_B(i,k) - \#_B(i,l)| + |\#_C(i,k) - \#_C(i,l)| \le \delta(A(p)) + 0 + \delta(A(p))$

by definition of $\delta(A(p))$ and by (iv). Also, $|\#_{ABC}(i,l) - \#_{ABC}(j,l)| \le |\#_{AC}(i,l) - \#_{AC}(j,l)| + |\#_{B'}i - \#_{B'}j|$ by (iv). Assume A has a columns $(0 \le a < p)$ and C has c columns $(0 \le c < p)$. Then

$$\frac{a}{p}\,\#_{A'}i - \delta(A(p)) \le \#_A(i,l) \le \frac{a}{p}\,\#_{A'}i - \delta(A(p))$$

is obtained using the fact that the expected number of (i,l)'s in A is $\frac{1}{p} \times$ size of the array $= \frac{a\,\#_{A'}i}{p}$. Thus,

$$\begin{aligned}\frac{a}{p}\,(\#_{A'}i - \#_{A'}j) + \frac{c}{p}\,(\#_{C'}i - \#_{C'}j) - 4\,\delta(A(p)) \le& \\ &\le \#_{AC}(i,l) - \#_{AC}(j,l) \le \\ &\le \frac{a}{p}\,(\#_{A'}i - \#_{A'}j) + \frac{c}{p}\,(\#_{C'}i - \#_{C'}j) + 4\,\delta(A(p))\end{aligned}$$

Again, using (iv) the following is obtained.

(vii) $\frac{a}{p}\,(\#_{A'}i - \#_{A'}j) + \frac{c}{p}\,(\#_{C'}i - \#_{C'}j) + (\#_{B'}i - \#_{B'}j) - 4\,\delta(A(p)) \le \#_{ABC}(i,l) - \#_{ABC}(j,l)$.

If $\#_{A'}i - \#_{A'}j \ge 0$ and $\#_{C'}i - \#_{C'}j \ge 0$ then from (vii) $(\#_{B'}i - \#_{B'}j) - 4\,\delta(A(p)) \le \#_{ABC}(i,l) - \#_{ABC}(j,l)$ follows, which in turn, using (i) implies that

$$-\delta(M) - 4\,\delta(A(p)) \le \#_{ABC}(i,l) - \#_{ABC}(j,l).$$

For $\#_{A'}i - \#_{A'}j < 0$ and $\#_{C'}i - \#_{C'}j < 0$ we get $\#_{A'}i - \#_{A'}j \le \frac{a}{p}\,(\#_{A'}i - \#_{A'}j)$ and $\#_{C'}i - \#_{C'}j \le \frac{c}{p}\,(\#_{C'}i - \#_{C'}j)$ so inequality (vi) becomes $\#_{A'B'C'}i - \#_{A'B'C'}j - 4\,\delta(A(p)) \le \#_{ABC}(i,l) - \#_{ABC}(j,l)$, which again results in

$$-\delta(M) - 4\,\delta(A(p)) \le \#_{ABC}(i,l) - \#_{ABC}(j,l)$$

using (i). By similar arguments for the remaining two cases using $A'B'$ or $B'C'$, respectively,

$$\text{(viii)}\quad -\delta(M) - 4\,\delta(A(p)) \le \#_{ABC}(i,l) - \#_{ABC}(j,l) \le \delta(M) + 4\,\delta(A(p))$$

is obtained. Combining (v), (vi), and (viii) we deduce

$$|\#_{ABC}(i,k) - \#_{ABC}(j,l)| \le \delta(M) + 6\,\delta(A(p))$$

which proves the result. □

Corollary 3.1. *If the prime factorization of m is $m = p_1^{e_1}\, p_2^{e_2} \ldots p_k^{e_k}$ then*

$$d(m) \le \sum_{i=1}^{k} 6\, e_i\, d(A(p_i)).$$

In particular, for $m = 2^k$ $d(m) = O(\log m)$.

Theorem 3.1 suggests finding $A(p)$ arrays for all prime p of low discrepancy. Clearly, for small p one can do that. For arbitrary p the modular assignment $f(i,j) = s\,i + j \bmod p$ where $s = \lfloor\sqrt{p}\rfloor$ gives $\delta(A(p)) = O(\sqrt{p})$ (To be strict, we should replace symbol 0 with p in this case). We only sketch the simple proof of this observation, because in the next section a better upper bound is proved by a different method.

Let $A(p)$ given by the above f. Because s and p are relative primes, each consecutive p entries in a column are permutations of $\{0, 1, \ldots, p-1\}$. Thus, calculating the discrepancy it is enough to consider only rectangles with at most p rows. $A(p)$ is tiled with "L" shaped tiles (see Figure 2)

s^2	$\ldots$	$p-1$		
s^2-s	s^2-s+1	s^2-s+2	$\ldots$	s^2-1
$\vdots$				$\vdots$
s	$s+1$	$s+2$	$\ldots$	$2s-1$
0	1	2	$\ldots$	$s-1$

Each such tile contains every different entry exactly once. A rectangle of at most p rows cuts $O(\sqrt{p})$ such tiles, each cut tile adds at most one to the discrepancy of the rectangle.

This construction and Corollary 3.1 gives $d(m) = O(\sqrt{m})$ for all m.

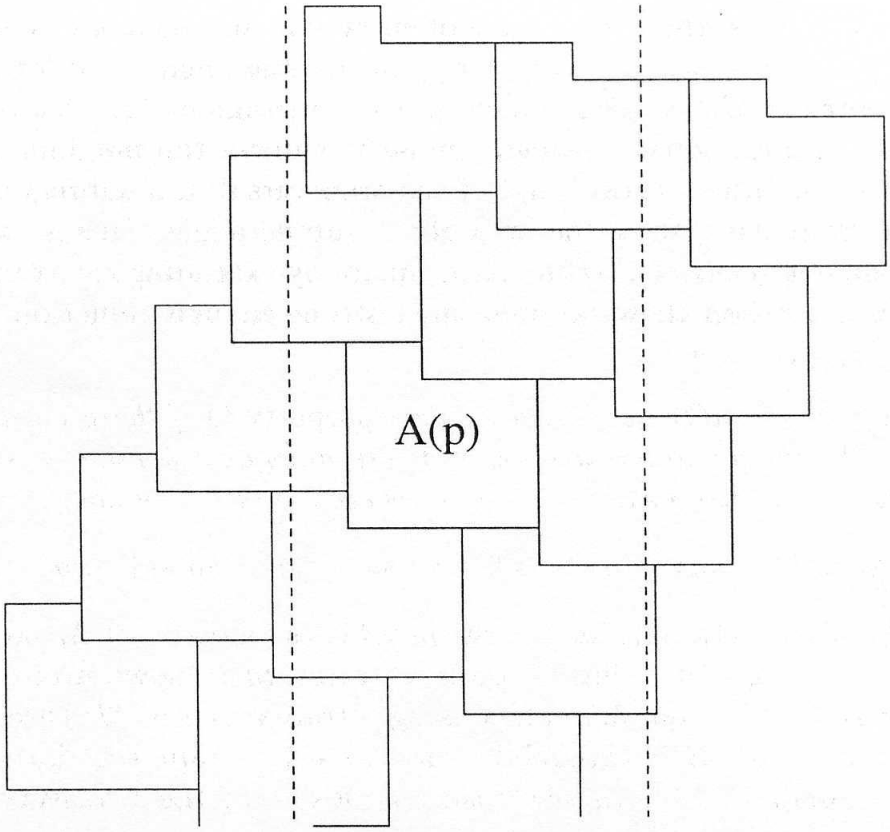

Figure 2.

4 Latin Square Type Assignments

In the previous section a construction was given that allows "multiplication", i.e. if good assignments for m and p are given, then a good one can be generated for $p\,m$. In this section we show how to "add". To this end, we consider assignments of *latin square type*. Let us recall that a latin square is an $m \times m$ array consisting of m different symbols, such that each symbol occurs in each row and column exactly once. An m-assignment is called latin square type, if

$$f(i,j) = f(i+m,j) = f(i,j+m) \qquad (*)$$

and the array $M = [f(i,j)]_{\substack{i=1,2,\ldots,m\\ j=1,2,\ldots,m}}$ forms a latin square. In this case M is called the *generator* of the assignment. The discrepancy of such an assignment, denoted by $\delta(M)$, is clearly attained by a rectangle whose number of rows and columns, respectively, is not larger than m.

A *transversal* of an $m \times m$ latin square $M = [f(i,j)]_{\substack{i=1,2,\ldots,m\\ j=1,2,\ldots,m}}$ is a set of pairwise distinct entries $\{f(i_1,j_1), f(i_2,j_2), \ldots, f(i_m,j_m)\}$ such that in each row and column there is exactly one of them. In other words, $\{i_1, i_2, \ldots, i_m\}$ and $\{j_1, j_2, \ldots, j_m\}$ are both permutations of $\{1, 2, \ldots, m\}$.

The *discrepancy of an entry* in a rectangle $\mathcal{R}$ of an m-assignment is the maximum deviation of the number of occurrences of that entry in any sub-rectangle of $\mathcal{R}$ from the expected value, that is from $\frac{1}{m}$th of the area of the subrectangle. That is, the discrepancy of entry k in subrectangle $\mathcal{R}[(a,b),(c,d)]$ is $|\frac{1}{m}(c-a)(d-b) - d_{a,b,c,d}(k)|$. Clearly, an m-assignment is of low discrepancy iff each entry is of low discrepancy in every rectangle. The *discrepancy of a transversal* in a latin square defined similarly, namely the maximum deviation of the number of entries (positions) of the transversal in a subregtangle of the latin square from the $\frac{1}{m}$th of the area of the subrectangle. This is extended for the m-assignment generated by the latin square by extending the transversal via (*). The next definition allows formulating a strong enough induction hypotheses that can be carried over.

Definition 4.1. *Number m is said to have* property O *if there exists a $m \times m$ latin square M that generates an $c \log m$ discrepancy m-assignment, such that M has a transversal of discrepancy $c_1 \log m$, where c and c_1 are absolute constants.*

Theorem 4.1. *If m has property O, then so do $3m$, $3m+1$, $3m+2$, as well.*

Proof (of Theorem 4.1). The idea of the proof is as follows. Let M be the $m \times m$ latin square showing that m has property O. First, M is blown up to a $3m \times 3m$ latin square B by Theorem 3.1. Then using a transversal of M, three new ones of B are constructed. B is extended to a $(3m+1) \times (3m+1)$ latin square C by putting symbol $3m+1$ to the positions of one of the transversals and to entry $(3m+1, 3m+1)$, while the original entries of the transversal are placed in column and row $3m+1$, respectively. Using that C has two transversals left, one can extend it with one more column and row, and preserve one transversal to carry over the induction.

Let $A(3)$ be the $\infty \times 3$ matrix generated by the latin square

$$L = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix}.$$

Then $\delta(A(3M)) \leq \delta(M)+6$, by Theorem 3.1. Also, it is generated by the $3m \times 3m$ latin square

$$B = \begin{bmatrix} [1\,2\,3] \times M \\ [3\,1\,2] \times M \\ [2\,3\,1] \times M \end{bmatrix}.$$

L has three transversals:

$$\begin{bmatrix} \bigcirc & \triangle & \square \\ \triangle & \square & \bigcirc \\ \square & \bigcirc & \triangle \end{bmatrix},$$

the *circle*, the *triangle* and the *square* transversals. The product of any of these and a transversal of M yields a transversal for B that are also called circle, triangle and square transversals, respectively.

In order to prove the statement for $3m$ we need to show that any of these product transversals has low discrepancy. Consider a subrectangle R of the $3m$-assignment generated by B. We may assume without loss of generality, that R

has at most $3m$ columns. R can be decomposed in the same way as in the proof of Theorem 3.1 into 3 parts: $R = S \cup T \cup V$, where T consists of the fully blown up parts, while S and V consist of 1 or 2 extra columns on the left-hand side and right-hand side, respectively. Because T is a fully blown up part, the product transversal has the same number of entries in T, as the transversal of M has in the subrectangle, which is blown up to T. That is, the density of the product transversal in T is just 1/3rd of that of the original, which is needed exactly. The parts S and V add at most 4 to the discrepancy of the product tranversal, so it has $O(\log 3m)$ discrepancy.

Now, from B a $(3m+1) \times (3m+1)$ latin square C is constructed as follows. Take the square transversal of B. For each entry t of it at position (i, j) we replace it by $3m+1$ and place t's in positions $(3m+1, j)$ and $(i, 3m+1)$. Furthermore, let the $(3m+1, 3m+1)$ entry of C be $3m+1$.

The $3m+1$-assignment generated by C has discrepancy $O(\log(3m+1))$, since each entry has low discrepancy. In order to obtain a transversal of C use the triangle (or circle) transversal of B and add the $3m+1$ entry in lower right (i.e. in position $(3m+1, 3m+1)$).

We would like to repeat this construction to go from $3m+1$ to $3m+2$ by using a transversal of C. However, to find a transversal of the resulting latin square D, such a transversal of C is needed that does not include the lower right entry. To this end, let us consider an entry i of the low discrepancy transversal of M. After the blow up, this becomes

$$\begin{matrix} (i,1) & (i,2) & (i,3) \\ \vdots & \vdots & \vdots \\ (i,3) & (i,1) & (i,2) \\ \vdots & \vdots & \vdots \\ (i,2) & (i,3) & (i,1) \end{matrix}$$

in B. It is transformed further in C to

$$\begin{matrix} (i,1) & \underline{(i,2)} & \underline{\underline{3m+1}} & \dots & (i,3) \\ \vdots & \vdots & \vdots & & \vdots \\ \underline{(i,3)} & 3m+1 & (i,2) & \dots & \underline{\underline{(i,2)}} \\ \vdots & \vdots & \vdots & & \vdots \\ 3m+1 & \underline{\underline{(i,3)}} & \underline{(i,1)} & \dots & (i,2) \\ \vdots & \vdots & \vdots & & \vdots \\ \underline{\underline{(i,2)}} & (i,1) & (i,3) & \dots & \underline{3m+1} \end{matrix}$$

Here the single underlined entries are from the triangle transversal. Instead of them, take the doubly underlined entries from this part of C together with the rest of the triangle transversal to obtain $(3m+2) \times (3m+2)$ latin square D in the same way as C was generated from B. This new transversal is also of low discrepancy, because it is a slight perturbation of the triangle transversal, which is of low discrepancy. Hence, D generates a $O(\log(3m+2))$ discrepancy $3m+2$-assignment. The only thing left to finish the proof is to find a good transversal of D. Now, in D we have

$$\begin{array}{cccccc}
(i,1) & \underline{(i,2)} & 3m+2 & \dots & (i,3) & 3m+1 \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
\underline{(i,3)} & 3m+1 & (i,2) & \dots & 3m+2 & (i,1) \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
3m+1 & 3m+2 & \underline{(i,1)} & \dots & (i,2) & (i,3) \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
3m+2 & (i,1) & (i,3) & \dots & \underline{3m+1} & (i,2) \\
(i,2) & (i,3) & 3m+1 & \dots & (i,1) & \underline{3m+2}
\end{array}$$

The underlined entries and the rest of the triangle transversal from the rest of B forms a low discrepancy transversal of D. □

Corollary 4.1.

$$d(m) = O(\log m)$$

for all $m > 0$. □

5 A Lower Bound

In this section we use the following deep result of Schmidt [6] to prove that Theorem 4.1 is best possible for latin square type assignments.

Theorem 5.1. *Let P be an arbitrary set of N points in the unit square $[0,1)^2$. Then there exisits a rectangle $B \subset [0,1)^2$ with sides parallel to the coordinate axes such that*

$$\Big| |P \cap B| - N\,\mathit{area}(B) \Big| > c \log N \qquad (**)$$

where c is an absolute constant.

To prove a lower bound for the discrepancy of an assignment it is enough to consider a finite part of it, in our case, the generating lating square.

Theorem 5.2. *Let M be an $m \times m$ latin square. Then any entry has discrepancy at least $c \log m$, where c is an absolute constant.*

Proof (of Theorem 5.2). Let us partition the unit square into m^2 little squares of side $1/m$. Consider entry t of M and put a point in the center of the little square in the ith row and jth column if the (i,j) entry of M is equal to t. Apply Theorem 5.1 with $N = m$ to find subrectangle B. We may assume without loss of generality, that B's sides coincide with the sides of some little squares, so B corresponds to some $\mathcal{R}[a,b,c,d]$ subrectangle of M. The number of points in B is equal to $d_{a,b,c,d}(t)$, while $N\,\mathrm{area}(B) = m\,\frac{c-a}{m}\,\frac{d-b}{m}$, so inequality (**) states that the deviation of entry t from the expected value in the subrectangle of M corresponding to B is at least $c \log m$. □

6 Conclusions, Open Problems

We have shown that the optimality of every m is $O(\log m)$. However, the lower bound works only for latin square type assignments. Thus, it is natural to ask, whether it holds in general?

In the proof of Theorem 4.1 triple-fold blow-up is used. One might ask why was it neccessary, could not the proof be done using only double blow-up? The reason for the seemingly more complicated induction is that transversals of the 3×3 latin square are essentially used, however, a 2×2 latin square does not have any.

Applying the blow up for $p = 2$ the the obtained assignments are generated by the following latin squares for $m = 2^t$ $t = 1, 2, 3, 4$:

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \\ 2 & 1 & 4 & 3 \\ 4 & 3 & 2 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 6 & 7 & 8 & 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 & 7 & 8 & 5 & 6 \\ 7 & 8 & 5 & 6 & 3 & 4 & 1 & 2 \\ 2 & 1 & 4 & 3 & 6 & 5 & 8 & 7 \\ 6 & 5 & 8 & 7 & 2 & 1 & 4 & 3 \\ 4 & 3 & 2 & 1 & 8 & 7 & 6 & 5 \\ 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 6 & 7 & 8 & 1 & 2 & 3 & 4 & 13 & 14 & 15 & 16 & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & 9 & 10 & 11 & 12 & 5 & 6 & 7 & 8 & 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 & 7 & 8 & 5 & 6 & 11 & 12 & 9 & 10 & 15 & 16 & 13 & 14 \\ 11 & 12 & 9 & 10 & 15 & 16 & 13 & 14 & 3 & 4 & 1 & 2 & 7 & 8 & 5 & 6 \\ 7 & 8 & 5 & 6 & 3 & 4 & 1 & 2 & 15 & 16 & 13 & 14 & 11 & 12 & 9 & 10 \\ 15 & 16 & 13 & 14 & 11 & 12 & 9 & 10 & 7 & 8 & 5 & 6 & 3 & 4 & 1 & 2 \\ 2 & 1 & 4 & 3 & 6 & 5 & 8 & 7 & 10 & 9 & 12 & 11 & 14 & 13 & 16 & 15 \\ 10 & 9 & 12 & 11 & 14 & 13 & 16 & 15 & 2 & 1 & 4 & 3 & 6 & 5 & 8 & 7 \\ 6 & 5 & 8 & 7 & 2 & 1 & 4 & 3 & 14 & 13 & 16 & 15 & 10 & 9 & 12 & 11 \\ 14 & 13 & 16 & 15 & 10 & 9 & 12 & 11 & 6 & 5 & 8 & 7 & 2 & 1 & 4 & 3 \\ 4 & 3 & 2 & 1 & 8 & 7 & 6 & 5 & 12 & 11 & 10 & 9 & 16 & 15 & 14 & 13 \\ 12 & 11 & 10 & 9 & 16 & 15 & 14 & 13 & 4 & 3 & 2 & 1 & 8 & 7 & 6 & 5 \\ 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 16 & 15 & 14 & 13 & 12 & 11 & 10 & 9 \\ 16 & 15 & 14 & 13 & 12 & 11 & 10 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{bmatrix}$$

The discrepancies are 1, 2, 2, 3, respectively. Studying the pattern of these latin squares one can find an explicit, non-recursive method for constructing them, starting from the first row $[1, 2, \ldots, 2^t]$. We strongly believe that for $m = 2^t$ our construction is best possible.

Theorem 5.2 works also for modular assignments, as well. However, there are not known bounds for their performance, in general. The construction for $A(p)$, p prime, gives an $O(\sqrt{m})$ upper bound, for certain m's. The question is, whether the lower, or the upper bound is sharp in that case?

In the present paper we studied low discrepancy allocation of two-dimensional data. It is natural to extend the scope of investigations to higher dimensions. For example, one can have a database of temperature distribution in a three-dimensional body. How well can three- (or higher-) dimensional data distributed?

References

1. K. A. S. Abdel-Ghaffar and A. El Abbadi,*Optimal Allocation of Two Dimensinal Data*, in: Database Theory – ICDT'97, F. Afrati and Ph. Kolaitis (Eds.), *Lecture Notes in Computer Science 1186*, 1997, 409-418.
2. C. Y. Chen and C. C. Chang, On GDM allocation method for partial range queries, *Information Systems*, **17** 1992, 381-394.
3. H. C. Du and J. S. Sobolewski, Disk allocation for cartesian product files on multiple disk systems, *ACM Trans. Database Syst.*, **7** 1982, 82-101.
4. C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic and W. Equitz, Efficient and effective querying by image content, *Journal of Intelligent Information Systems*, **3** 1994, 231-262.
5. V. E. Ogle and M. Stonebraker, Chabot: Retrieval from relational database of images, *Computer* **28**(9) 1995, 40-48.
6. W. M. Schmidt,Irregularities of distribution VII, *Acta Arithm.* **21**, (1972) 45-50.
7. T. R. Smith, A digital library for geographically referenced materials, *Computer*, **29**(5) 1996, 54-60.
8. M. Vitterli and C. Herley, Wavelets and filter banks: Theory and design, *IEEE Transactions on Signal Processing*, **40**(9) 1992, 2207-2232.