A Competitive Strategy for Distance-Aware Online Shape Allocation

Sándor P. Fekete, Nils Schweer, and Jan-Marc Reinhardt

Department of Computer Science, TU Braunschweig, Germany
{s.fekete,n.schweer,j-m.reinhardt}@tu-bs.de

Abstract. We consider the following online allocation problem: Given a unit square S, and a sequence of numbers $n_i \in \{0, 1\}$ with $\sum_{j=0}^{i} n_j \leq 1$; at each step i, select a region C_i of previously unassigned area n_i in S. The objective is to make these regions compact in a distance-aware sense: minimize the maximum (normalized) average Manhattan distance between points from the same set C_i. Related location problems have received a considerable amount of attention; in particular, the problem of determining the "optimal shape of a city", i.e., allocating a *single* n_i has been studied, both in a continuous and a discrete setting. We present an online strategy, based on an analysis of space-filling curves; for continuous shapes, we prove a factor of 1.8092, and 1.7848 for discrete point sets.

Keywords: Clustering, average distance, online problems, optimal shape of a city, space-filling curves, competitive analysis.

1 Introduction

Many optimization problems deal with allocating point sets to a given environment. Frequently, the problem is to find compact allocations, placing points from the same set closely together. One well-established measure is the average L_1 distance between points. A practical example occurs in the context of grid computing, where one needs to assign a sequence of jobs i, each requiring an (appropriately normalized) number n_i of processors, to a subset C_i of nodes of a large square grid, such that the average communication delay between nodes of the same job is minimized; this delay corresponds to the number of grid hops [10], so the task amounts to finding subsets with a small average L_1, i.e., *Manhattan* distance. Karp et al. [7] studied the same problem in the context of memory allocation.

Even in an offline setting without occupied nodes, finding an optimal allocation for one set of size n_i is not an easy task; as shown in Fig. 1, the results are typically "round" shapes. If a whole sequence of sets has to be allocated, packing such shapes onto the grid will produce gaps, causing later sets to become disconnected, and thus leads to extremely bad average distances. Even restricting the shapes to be rectangular is not a remedy, as the resulting problem of deciding whether a set of squares (which are minimal with respect to L_1 average distance

S.K. Ghosh and T. Tokuyama (Eds.): WALCOM 2013, LNCS 7748, pp. 41–52, 2013.

among all rectangles) can be packed into a given square container is NP-hard [9]; moreover, disconnected allocations may still occur.

In this paper, we give a first algorithmic analysis for the *online* problem. Using an allocation scheme based on a space-filling curve, we establish competitive factors of 1.8092 and 1.7848 for minimizing the maximum average Manhattan distance within an allocated set, and non-trivial lower bounds for these factors.

Related Work

Compact location problems have received a considerable amount of attention. Krumke et al. [8] have considered the *offline* problem of choosing a set of n vertices in a weighted graph, such that the average distance is minimized. They showed that the problem is NP-hard (even to approximate); for the scenario in which distances satisfy the triangle inequality, they gave algorithms that achieve asymptotic approximation factors of 2. For points in two-dimensional space and Manhattan distances, Bender et al. [2] gave a simple 1.75-approximation algorithm, and a polynomial-time approximation scheme for any fixed dimension.

The problem of finding the "optimal shape of a city", i.e., a shape of given area that minimizes the average Manhattan distance, was first considered by Karp, McKellar, and Wong [7]; independently, Bender, Bender, Demaine, and Fekete [1] showed that this shape can be characterized by a differential equation for which no closed form is known. For the case of a finite set of n points that needs to be allocated to a grid, Demaine et al. [5] showed that there is an $O(n^{7.5})$ dynamic-programming algorithm, which allowed them to compute all optimal shapes up to $n = 80$. Note that all these results are strictly offline, even though the original motivation (register or processor allocation) is online.

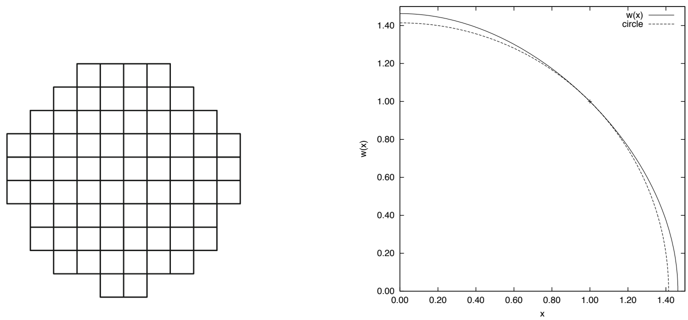

Fig. 1. Finding optimal individual shapes. (Left) An optimal shape composed of n=72 grid cells, according to [5]. (Right) The optimal limit curve $w(x)$, according to [2].

Space-filling curves for processor allocation with our objective function have been used before, see Leung et al. [10]; however, no algorithmic results and no competitive factor was proven. Wattenberg [15] proposed an allocation scheme for purposes of minimizing the maximum *Euclidean diameter* of an allocated shape; this is a different measure than the one established by [10]. Like other authors before (in particular, Niedermeier et al. [11] and Gotsman and Lindenbaum [6]), he considered *c-locality*: for a sequence $1, \ldots, i, \ldots, j, \ldots$ of points on a line, a space-filling mapping $h(.)$ will guarantee $L_2(h(i), h(j)) < c\sqrt{|j-i|}$, for a constant c that is $\sqrt{6} \approx 2.449$ for the Hilbert curve, and 2 for the so-called H-curve. One can use c-locality for establishing a constant competitive factor for our problems; however, given that their focus is on bounding the worst-case distance ratio for an embedding instead of the average distance, it should come as no surprise that the resulting values are significantly worse than ours. On a different note, de Berg, Speckmann, and van der Weele [4] consider treemaps with bounded aspect ratio. Other related work includes Dai and Su [3].

Our Results

We give a first competitive analysis for the online shape allocation problem within a given bounding box, with the objective of minimizing the maximum average Manhattan distance. In particular, we give the following results.

- We show that for the case of continuous shapes (in which numbers n_i correspond to area), a strategy based on a space-filling Hilbert curve achieves a competitive ratio of 1.8092.
- For the case of discrete point sets (in which numbers indicate the number of points that have to be chosen from an appropriate $N \times N$ orthogonal grid), we prove a competitive factor of 1.7848.
- We sketch how these factors may be further improved, but point out that a Hilbert-based strategy is no better than a competitive factor of 1.3504, even with an improved analysis.
- We establish a lower bound of 1.144866 for *any* online strategy in the case of discrete point sets, and argue the existence of a lower bound for the continuous case.

The rest of this paper is organized as follows. In Section 2, we give some basic definitions and fundamental facts. In Section 3, we provide a brief description of an allocation scheme based on a space-filling curve. Section 4 gives a mathematical study for the case of continuous allocations, proving that the analysis can be reduced to a limited number of shapes, and establishes a competitive factor of 1.8092. Section 5 sketches a similar analysis for the case of discrete allocations; as a result, we prove a competitive factor of 1.7848. Section 6 discusses lower bounds for online strategies. Final conclusions are presented in Section 7.

2 Preliminaries

We examine the problem of selecting shapes from a square, such that the maximum average L_1-distance of the shapes is minimized. We first formulate the problem more precisely. This covers both the continuous and the discrete case; the former arises as the limiting case of the latter, while the latter needs to be considered for allocations within a grid of limited size.

Definition 1. *A* city *is a (continuous) shape in the plane with fixed area. For a city C of area n, we call*

$$c(C) = \frac{1}{2} \iiiint_{(x,y),(u,v)\in C} (|x-u| + |y-v|)\, \mathrm{d}v\, \mathrm{d}u\, \mathrm{d}y\, \mathrm{d}x \tag{1}$$

the total Manhattan distance between all pairs of points in C *and*

$$\phi(C) = \frac{2\, c(C)}{n^{5/2}} \tag{2}$$

the ϕ-value *or* average distance *of C. An n-*town *T is a subset of n points in the integer grid. Its* normalized average Manhattan distance *is*

$$\phi(T) = \frac{2c(T)}{n^{5/2}} = \frac{\sum_{s\in T}\sum_{t\in T} \|s-t\|_1}{n^{5/2}} \tag{3}$$

The normalization with $n^{2.5}$ yields a dimensionless measure that remains unchanged under scaling (so it depends only on the shape, not on the size), and makes the continuous and the discrete case comparable; see [1].

Problem 2. *In the continuous setting, we are given a sequence $n_1, n_2, \ldots, n_k \in \mathbb{R}^+$ with $\sum_{i=1}^{k} n_i \leq 1$. Cities $C_1, C_2, \ldots, C_k$ of size $n_1, n_2, \ldots, n_k$ are to be chosen from the unit square, such that $\max_{1\leq i\leq k} \phi(C_i)$ is minimized.*

In the discrete setting, we are given a sequence $n_1, n_2, \ldots, n_k \in \mathbb{N}^+$ with $\sum_{i=1}^{k} n_i \leq N^2$. Towns $C_1, C_2, \ldots, C_k$ of size $n_1, n_2, \ldots, n_k$ are to be chosen from the $N \times N$ grid, such that $\max_{1\leq i\leq k} \phi(C_i)$ is minimized.

Although it has not been formally proven, the offline problem is conjectured to be NP-hard, see [13]; if we restrict city shapes to be rectangles, there is an immediate reduction from deciding whether a set of squares can be packed into a larger square [9]. (A special case arises from considering integers, which corresponds to choosing grid locations.) Our approximation works online, i.e., we choose the cities in a specified order, and no changes can be made to previously allocated cities; clearly, this implies approximation factors for the corresponding offline problems.

There are lower bounds for $\max_{1\leq i\leq k} \phi(C_i)$ that generally cannot be achieved by any algorithm. One important result is the following theorem.

Theorem 3. *Let C be any city. Then $\phi(C) \geq 0.650245$.*

A proof can be found in [1]. For $n_1 = 1$ any algorithm must select the whole unit square, thus $2/3$, the ϕ-value of a square, is a lower bound for the achievable ϕ-value. We will discuss better lower bounds in the conclusions.

3 An Allocation Strategy

While long and narrow shapes tend to have large ϕ-values, shapes that fill large parts of an enclosing rectangle with similar width and height usually have better average distances; however, one has to make sure that early choices with small average distance do not leave narrow pieces with high average distance, or even disconnected pieces, making the normalized ϕ-values potentially unbounded.

Our approach uses the recursive Hilbert family of curves in order to yield a provably constant competitive factor. That family is based on a recursive construction scheme and becomes space filling for infinite repetition of said scheme [12]. For a finite number r of repetitions, the curve traverses all points of the used grid. For $1 \leq r \leq 3$, the curve is shown in Fig. 2. Thus, the Hilbert curve provides an order for the cells of the grid, which is then used for allocation, as illustrated in Fig. 3. More formal details of the recursive definition of the Hilbert family (e.g. with text-rewriting rules, such as the ones in [14]) go beyond the scope of this extended abstract.

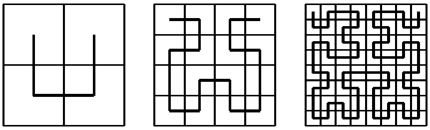

Fig. 2. Hilbert curve with $1 \leq r \leq 3$

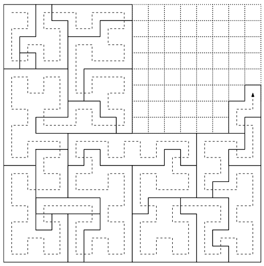

Fig. 3. A sample allocation according to our strategy

More technically, the unit square is recursively subdivided into a grid consisting of $2^r \times 2^r$ grid cells, for an appropriate refinement level $r > 0$, as shown in Fig. 2. For the sake of concise presentation within this short abstract, we assume that every input n_i is an integral multiple of $c = 4^{-R}$, for an appropriately large $R > 0$. (We will mention in the Conclusions how this assumption can be removed, based on Lemma 6.) Similar to the recursive structure of quad-trees, the actual subdivision can be performed in a self-refining manner, whenever a grid cell is not completely filled. This means that during the course of the online allocation, we may use different refinement levels in different parts of the layout; however, this will not affect the overall analysis, as further refinement of the grid does not change the quality of existing shapes.

Definition 4. *For a given refinement level r, an r-*pixel *P is a grid square of size $2^{-r} \times 2^{-r}$. For a given allocated shape C_i, a pixel is* full *if $P \subseteq C_i$; it is* fractional*, if $P \cap C_i \neq \emptyset$ and $P \not\subseteq C_i$.*

Now the description of the algorithm is simple: for every input n_i, choose the next set of $n_i/2^R$ R-pixels traversed by the Hilbert curve as the city C_i, starting in the upper left corner of the grid. For an illustration, see Fig. 3.

The following lemma is a consequence of the recursive structure of the Hilbert family. We use it in the following section for deriving upper bounds.

Lemma 5. *Let C be a city generated by our strategy with area at most $n \leq l\,4^j\,4^{-R}$ for $j \in \{0, 1, \ldots, R\}, l \in \mathbb{N}$. Then at any refinement level r, C contains at most two fractional r-pixels.*

4 Analysis

For the analysis of our allocation scheme we will first make use of Lemma 5. As noted in the following Lemma 6, filling in the two fractional pixels of an allocated shape yields an estimate for the total distance at a coarser refinement level. In a second step, this will be used to derive global bounds by computing the worst-case bounds for shapes of at most refinement level 3. This reduces the task of providing a general upper bound on the competitive factor to considering a finite number of shapes of limited size. (As discussed in the Conclusions, carrying out the computations on a lower or higher refinement level gives looser or tighter results.)

In the following, W_n denotes the worst case among all cities of n pixels that can be produced by our Hilbert strategy; because of the normalized nature of ϕ, this is independent on the size of the pixels, and only the shape matters.

Lemma 6. *Let C be a city generated by our strategy with area at most $n \leq l\,4^r\,4^{-R}$ for $r \in \{0, 1, \ldots, R\}, l \in \mathbb{N}$. Then we have $c(C) \leq c(W_{l+1})$, where W_{l+1} is a worst case among all cities produced by our allocation scheme that consists of $(l+1)$ r-pixels.*

Proof. By Lemma 5, we know that only the first and the last pixel of C may be fractional. Therefore C cannot intersect more than $l+1$ r-pixels. By replacing the two fractional pixels by full pixels, we get a city W that consists of $l+1$ full r-pixels, and $c(C) \leq c(W)$. By definition, $c(W) \leq c(W_{l+1})$, and the claim holds. □

Therefore, we can give upper bounds for the worst case by considering the values of W_n at some moderate refinement level. The W_n can be found by enumeration; as described in the full version of the paper, a speed-up can be achieved by making use of the recursive construction of the W_n. We determined the shapes and ϕ-values of the W_n for $n \leq 65$; by Lemma 6, this suffices to provide upper bounds for all cities with area up to $64 * 2^{-r}$, i.e., these computational results give an estimate for the round-up error using refinement level 3. The full table of average distances can be found in the full version of the paper; the worst cases among the examined ones are W_{56} and W_{14}, which have the same shape, shown in Fig. 4.

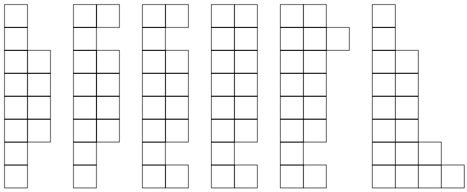

Fig. 4. Worst cases W_n for $12 \leq n \leq 17$

Theorem 7. *A Hilbert strategy guarantees* $\max_{1 \leq n \leq k} \phi(C_n) \leq 1.1764$.

Proof. Consider a city C of size n generated by our strategy. If n is sufficiently small, i.e., smaller than an $R - r$-pixel, $r \geq 0$, C consists of at most 4^r cells and its average distance can be bounded by the worst case for that particular number of cells. In the case that C has a larger, more refined shape, an analysis of a finite number of shapes is still sufficient:

We know that $n > 4^r c$ and we can assume that $n \leq 4^{r+1} c$ (or else we use the analysis on the less refined $(R - (r+1))$-pixels). Thus, there must be an l such that $l 4^r c < n \leq (l+1) 4^r c$ with $l = 1, \ldots, 3$. Yet, we can get closer to n, as we know that an $(R - r)$-pixel consists of 4^r cells. We get the inequality $l 4^{r-k} < n \leq (l+1) 4^{r-k} c$, $k \leq r$, $l = 4^k, \ldots, 4^{k+1} - 1$.

Hence, a city of arbitrary size n corresponds to at most $(l+1)$ sub-squares of a certain size (depending on the precision of the analysis), i.e., a city of size at most $(l+1) 4^{r-k} c$. Now we can use Lemma 5 to bound the average distance of the city, yielding

$$\phi(C) \leq \frac{2\, c(W)}{(l\, 4^{r-k} c)^{5/2}} = \frac{\phi(W_{l+2})((l+2)\, 4^{r-k} c)^{5/2}}{(l\, 4^{r-k} c)^{5/2}} \tag{4}$$

$$= \phi(W_{l+2}) \left(1 + \frac{2}{l}\right)^{5/2} =: \Phi(W_l). \tag{5}$$

The resulting bound is $\max(\{\phi(W_i) : 1 \leq i \leq 4^r\} \cup \{\Phi(W_l) : 4^k \leq l \leq 4^{k+1} - 1\})$. Note that the number of shapes considered is at most 4^{k+1}.

We conducted the calculations for $k = 2$; as it turns out, the maximum is attained for $\Phi(W_{16}) = 1.1764$. See the full version of the paper for details. □

Corollary 8. *Our strategy achieves a competitive factor of 1.8092.*

Proof. According to Theorem 3, no algorithm can guarantee a better ϕ-value than 0.650245. Our strategy yields an upper bound of 1.1764. This results in a factor of $1.1764/0.650245 \approx 1.8092$. □

5 Discrete Point Sets

Our above analysis relies on continuous weight distributions, which imply the lower bound on ϕ-values stated in Theorem 1. This does not include the discrete scenario, in which each value n_i indicates the number of integer grid points that have to be chosen from an appropriate $N \times N$-grid. As discussed in the paper [5], considering discrete weight distributions may allow lower average distances; e.g., a single point yields a ϕ-value of 0. As a consequence, *towns* (subsets of the integer grid) have lower average distances than cities of the equivalent total weight. However, we still get a competitive ratio for the case of online towns.

Theorem 9. *For n-towns, a Hilbert-curve strategy guarantees a competitive factor of at most 1.7848 for the ϕ-value.*

Proof. Lemma 5 still holds, so analogously to Theorem 7, we consider the values up to $n = 64$, and show that the worst case is attained for $n = 16$, which yields an upper bound of 1.123. See the full version for detailed numbers.

For a lower bound, the general value of 0.650245 for ϕ-values cannot be applied, as discrete point sets may have lower average distance. Instead, we verify that the ratio $\rho(n)$ of achieved ϕ to optimal ϕ, is less than 1.7848. This is the same as $c(T_n)/c_{town}(n)$ for $n \leq 64$; see the full version of the paper. For $65 \leq n \leq 80$, the optimal values in [5] allow us to verify that $\phi \geq 0.6292$; see the full version of the paper.

Thus, we have to establish a lower bound for ϕ for $n \geq 81$. We make use of equation (5), p. 89 of [5]; see Fig. 5: for a given number n of grid points, the difference between the optimal total Manhattan distance $c_{city}(n)$ for a city consisting of n unit squares and the optimal total distance $c_{town}(n)$ for a town consisting of n grid points is equal to $\Lambda(n) := \frac{1}{6}\left(\sum_i c_i^2 + \sum_j r_j^2\right)$, where c_i is the number of grid points in column i, and r_j is the number of grid points in row j. Because $\frac{2c_{city}(n)}{n^{2.5}}$ is bounded from below by $\psi = 0.650245$, we get a lower bound of $\psi - \frac{2\Lambda(n)}{n^{2.5}} \leq \frac{2c_{town}(n)}{n^{2.5}}$ for the ϕ-value of an n-town.

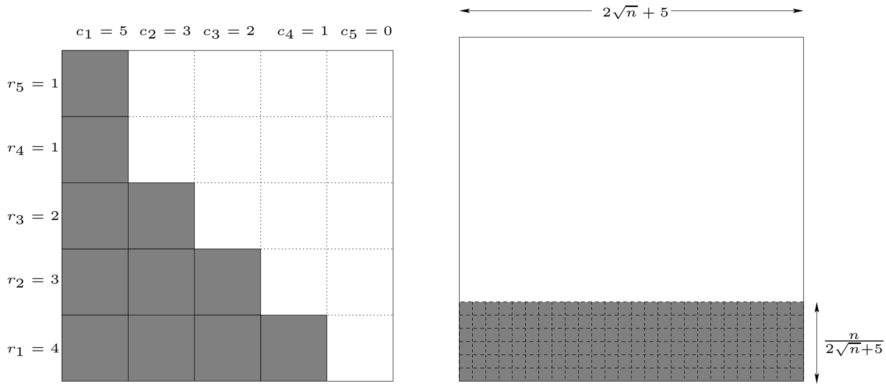

Fig. 5. Establishing a lower bound for ϕ: Defining $\Lambda(n)$; an arrangement that maximizes $\Lambda(n)$

This leaves the task of providing an upper bound for $2\Lambda(n)/n^{2.5}$. According to Lemma 5 of [5], the bounding box of an optimal n-town does not exceed $2\sqrt{n}+5$. Therefore, we have $c_i \leq 2\sqrt{n}+5$; as $\sum_i c_i = n$ and the function $\sum_i c_i^2$ is superlinear in the c_i, we conclude that $\sum_i c_i^2$ is maximized by subdividing n into $\frac{n}{2\sqrt{n}+5}$ columns of $2\sqrt{n}+5$ points each, so $\sum_i c_i^2 \leq n(2\sqrt{n}+5)$. Analogously, we have $\sum_j r_j^2 \leq n(2\sqrt{n}+5)$, so $2\Lambda(n)/n^{2.5} \leq \frac{2}{3}(\frac{2}{n} + \frac{5}{n^{1.5}})$. For $n \geq 81$, this implies $2\Lambda(n)/n^{2.5} \leq \frac{4}{243} + \frac{10}{2187} = 0.0210333...$ or $\phi(n) \geq 0.6292$. This yields an overall competitive ratio of not more than 1.123/0.6292, i.e., 1.7848. □

A more refined analysis of $\Lambda(n)$ considers maximizing $\sum_i c_i^2 + \sum_j r_j^2$ all at once, instead of $\sum_i c_i^2$ and $\sum_j r_j^2$ separately, for a maximum value of $n(2\sqrt{n}+5) + \frac{n^2}{2\sqrt{n}+5}$. For $n \geq 81$, this yields $2\Lambda(n)/n^{2.5} \leq \frac{2}{243} + \frac{5}{2187} + \frac{2}{621} = 0.0137373...$ As the resulting competitive ratio of 1.7643 is only very slightly better, we omit further details from this extended abstract. If instead we rely on the unproven conjecture in [5] that $\frac{2c_{town}}{n^{2.5}} \approx \psi - \frac{0.410}{n}$, we get $\phi \geq 0.6451$, which corresponds to experimental evidence; the resulting competitive factor is 1.7406.

6 Lower Bounds

We demonstrate that there are non-trivial lower bounds for a competitive factor. We start by considering the discrete online scenario for towns.

Theorem 10. *No online strategy can guarantee a competitive factor below* $\frac{64}{\sqrt{5^5}} =$ 1.144866....

Proof. Consider a 3×3 square, and let $n_1 = 4$; see Fig. 6. If (a) the strategy allocates a 2×2 square (for a total distance of 8), then $n_2 = 5$, and the resulting L-shape has a total distance of 20 and a ϕ-value of $40/5^{2.5} = 0.715541...$ Allocating (b) the first town with an L-shape of total distance 10 results in $\phi = 20/32 = 0.625$, and the second with a total distance of 16, or $\phi = 32/5^{2.5} = 0.572433...$

If instead, (c) the first town is allocated different from a square, the total distance is at least 10, and $\phi \geq 20/32$; then (d) $n_2 = n_3 = n_4 = n_5 = n_6 = 1$, and an optimal strategy can allocate the first town as a 2x2 square, with $\phi = 0.5$. This bounds the competitive ratio, as claimed. □

For the case of continuous allocations, we claim the following.

Theorem 11. *There is* $\delta > 0$*, such that no online strategy can guarantee a competitive factor* $1+\delta$*.*

Proof. Consider $n_1 = 1/2$, in combination with the two possible scenarios

- **(a)** $n_2 = 1/2$;
- **(b)** $n_2 = n_3 = \ldots = \varepsilon$.

In scenario (a), an adversary can assign two $(1 \times 1/2)$-rectangles, for a ϕ-value of 0.707...; in scenario (b), an adversary can assign all shapes as squares, for a ϕ-value of 0.666... If the player chooses a square of size $\sqrt{2}/2$ first, the adversary

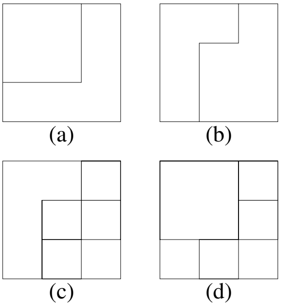

Fig. 6. The four cases considered in Theorem 10; the left column shows the choices by an algorithm, the right the corresponding optimal choices for the ensung sequence

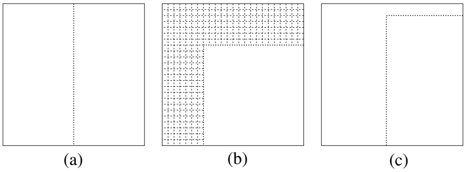

Fig. 7. The scenarios considered in Theorem 11, and a possible choice for the player

can choose scenario (a), causing the second allocation to be in L-shape with ϕ-value $\frac{2}{3}(7-4\sqrt{2}) = 0.895431...$, as opposed to the optimal value of 0.707... If the player chooses a $(1 \times 1/2)$-rectangle first, the adversary chooses scenario (b), for a ratio of 1.06066... The existence of the claimed lower bound follows from continuity, as the ϕ-value changes continuously with continuous deformation of the involved shapes. □

The precise value arising from the scenarios in Theorem 11 is complicated. It can be obtained by computing the optimal intermediate value for the player that allows him to protect against both scenarios at once. For example, optimizing over the family of allocations shown in Figure 7 (c) yields a competitive ratio that is better than 1.06; however, the player may do even better by using curved boundaries. The involved computational effort for the resulting optimization problem promises to be at least as complicated as computing the "optimal shapes of a city", for which no closed-form solution is known, see [7,1].

7 Conclusions

We have established a number of results for the online shape allocation problem. In principle, further improvement could be achieved by replacing the computational results for level 3 (i.e., $n = 16, \ldots, 64$) by level 4 (i.e., $n = 65, \ldots, 256$). (Conversely, a simplified analysis with level 2, i.e., $n = 4, \ldots, 16$; yields a worse factor of 3.6525.) However, the highest known optimal ϕ-values are for $n = 80$, obtained by using the $O(n^{7.5})$ algorithm of [5]. In any case, there is a threshold of 1.3504 for Hilbert-based strategies, which we believe to be tight: this is the ratio between the upper bound of 0.8768 for $n = 14$ (and for $n = 56, 224, \ldots$) and the asymptotic lower bound of 0.650245; because asymptotically, continuous and discrete case converge, this also applies to the discrete case. Other open problems are to raise the lower bound of 1.144866 for the discrete case, and establish definitive values for the continuous case.

As noted in Section 3, we can eliminate the assumption of all n_i being multiples of some 2^{-R}, by making use of Lemma 6, and allocating a small round-off fraction to a fractional pixel maintains the same bounds. However, the formal aspects of describing the resulting allocation scheme become somewhat tedious and would require more space than provided for this short abstract.

The offline problem is interesting in itself: for given n_i, $i = 0, \ldots, m$, allocate disjoint regions of area n_i in a square, such that the maximum average Manhattan distance for each shape is minimized. As mentioned, there is some indication that this is an NP-hard problem; however, even relatively simple instances are prohibitively tricky to solve to optimality, making it hard to give a formal proof. Clearly, our online strategy provides a simple approximation algorithm; however, better factors should be possible by exploiting the a-priori information of knowing all n_i, e.g., by sorting them appropriately.

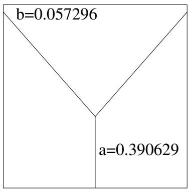

Fig. 8. A possible worst-case scenario for the offline problem

Another interesting open question for the offline scenario is the maximum optimal ϕ-value for any set $n_1, \ldots, n_m$. A simple lower bound is $2/3 = 0.666...$, as that is the average distance of the whole square. A better lower bound is provided by dividing the square into two or three equal-sized parts. For the case $n_1 = n_2 = 1/2$, we can use symmetry and convexity to argue that an optimum can be obtained by a vertical split, yielding $\phi = \sqrt{2}/2 = 0.707$. We believe the global worst case is attained for $n_1 = n_2 = n_3 = 1/3$. Unfortunately, it is no longer possible to exploit only symmetry for arguing global optimality. Figure 8 shows an allocation with $\phi = 0.718736...$ for all three regions[1]. We conjecture that this is the best solution for $n_1 = n_2 = n_3 = 1/3$, as well as the worst case for any optimal partition of the unit square.

[1] More precisely, the involved values can be expressed as $a = \frac{1}{108}\left(55 - \frac{791}{\theta} + \theta\right)$ and $\phi = \frac{\left(9602477 - 13416\sqrt{585705}\right)\theta + \left(202679 + 204\sqrt{585705}\right)\psi^2 + 82133\theta^3}{77760\sqrt{3}\theta}$ with $\theta := \left(-16253 + 36\sqrt{585705}\right)^{1/3}$.

Acknowledgments. A short abstract based on preliminary results of this paper appears in the informal, non-competitive Workshop EuroCG. (Standard disclaimer of that workshop: "This is an extended abstract of a presentation given at EuroCG 2011. It has been made public for the benefit of the community only and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear in a conference with formal proceedings and/or in a journal.")

We thank Bettina Speckmann for pointing out references [15] and [4], and other colleagues for helpful hints to improve the presentation of this paper.

References

1. Bender, C.M., Bender, M.A., Demaine, E.D., Fekete, S.P.: What is the optimal shape of a city? J. Physics A: Mathematical and General 37(1), 147–159 (2004)
2. Bender, M.A., Bunde, D.P., Demaine, E.D., Fekete, S.P., Leung, V.J., Meijer, H., Phillips, C.A.: Communication-Aware Processor Allocation for Supercomputers: Finding Point Sets of Small Average Distance. Algorithmica 50(2), 279–298 (2008)
3. Dai, H.K., Su, H.C.: On the Locality Properties of Space-Filling Curves. In: Ibaraki, T., Katoh, N., Ono, H. (eds.) ISAAC 2003. LNCS, vol. 2906, pp. 385–394. Springer, Heidelberg (2003)
4. de Berg, M., Speckmann, B., van der Weele, V.: Treemaps with bounded aspect ratio. CoRR, abs/1012.1749 (2010)
5. Demaine, E.D., Fekete, S.P., Rote, G., Schweer, N., Schymura, D., Zelke, M.: Integer point sets minimizing average pairwise L1 distance: What is the optimal shape of a town? Comp. Geom. 40, 82–94 (2011)
6. Gotsman, C., Lindenbaum, M.: On the metric properties of discrete space-filling curves. IEEE Transactions on Image Processing 5(5), 794–797 (1996)
7. Karp, R.M., McKellar, A.C., Wong, C.K.: Near-Optimal Solutions to a 2-Dimensional Placement Problem. SIAM J. Computing 4(3), 271–286 (1975)
8. Krumke, S., Marathe, M., Noltemeier, H., Radhakrishnan, V., Ravi, S., Rosenkrantz, D.: Compact location problems. Theor. Comput. Sci. 181(2), 379–404 (1997)
9. Leung, J.Y.-T., Tam, T.W., Wing, C.S., Young, G.H., Chin, F.Y.: Packing squares into a square. J. Parallel Distrib. Comput. 10(3), 271–275 (1990)
10. Leung, V.J., Arkin, E.M., Bender, M.A., Bunde, D.P., Johnston, J., Lal, A., Mitchell, J.S.B., Phillips, C.A., Seiden, S.S.: Processor Allocation on Cplant: Achieving General Processor Locality Using One-Dimensional Allocation Strategies. In: Proc. IEEE CLUSTER 2002, pp. 296–304 (2002)
11. Niedermeier, R., Reinhardt, K., Sanders, P.: Towards optimal locality in mesh-indexings. Discrete Applied Mathematics 117(1-3), 211–237 (2002)
12. Sagan, H.: Space-Filling Curves. Springer, New York (1994)
13. Schweer, N.: Algorithms for Packing Problems. PhD thesis, Braunschweig (2010)
14. Siromoney, R., Subramanian, K.: Space-filling Curves and Infinite Graphs. In: Ehrig, H., Nagl, M., Rozenberg, G. (eds.) Graph Grammars 1982. LNCS, vol. 153, pp. 380–391. Springer, Heidelberg (1983)
15. Wattenberg, M.: A note on space-filling visualizations and space-filling curves. In: Proceedings of the IEEE Symposium on Information Visualization, INFOVIS, pp. 181–186 (2005)