Approximating Euclidean circles by neighbourhood sequences in a hexagonal grid

Benedek Nagy [a] and Robin Strand [b]

[a] *Faculty of Informatics, University of Debrecen, Hungary H-4010 Debrecen P.O.Box 12.*

[b] *Centre for Image Analysis, Uppsala University, Sweden*

Abstract

In this paper the nodes of the hexagonal grid are used as points. There are 3 types of neighbours on this grid, therefore neighbourhood sequences contain values 1, 2, 3. The grid is coordinatized by three coordinates in a symmetric way. Digital circles are classified based on digital distances using neighbourhood sequences. They can be triangle, hexagon, enneagon and dodecagon. Their corners and side-lengths are computed, such as their perimeters and areas. The radius of a digital disc is usually not well-defined, i.e., the same disc can have various radii according to the neighbourhood sequence used . Therefore the non-compactness ratio is used to measure the quality of approximation of the Euclidean circles. The best approximating neighbourhood sequence is presented. It is shown that the approximation can be improved using two neighbourhood sequences in parallel. Comparisons to other approximations are also shown.

Key words: Digital geometry, neighbourhood sequences, approximation, digital circles, formal systems, finite automata

1 Introduction

Circle approximations are interesting and important problems of digital geometry. The approximation can be done by using digital distance functions. One of the main problems of classical digital distances, such as the city block and chessboard distances, is that they are not rotational independent [6]. A proposed solution is the neighbourhood sequences: one can vary the allowed

Email addresses: `nbenedek@inf.unideb.hu` (Benedek Nagy), `robin@cb.uu.se` (Robin Strand).

Preprint submitted to Elsevier Science *26 October 2010*

type of the neighbourhood in a path step by step [14,3]. Some properties of distances based on neighbourhood sequences are detailed in [4]. The literature on this topic using $\mathbb{Z}^n$ is rich [7,5], however nowadays, in many applications it is worth to consider other grids than the square one [1]. It is well-known that there are three tessellations of the plane if it is required that all tessellas (pixels) be congruent to one and only one polygon: the triangular tessellation, the square tessellation and the hexagonal tessellation. Moreover the triangular and hexagonal tessellations are duals of each other in graph theoretic sense. (The square grid is self-dual in this sense.) The triangular and hexagonal grids have some nice properties and they are regular, therefore it is not too hard to handle them. In this paper the nodes of hexagonal grid are used as points therefore our grid is isometric with the triangular tessellation. Some of the earlier results given on the triangular grid (where the triangles are used as points) can also be used [8]. The geometry of the hexagonal grid with a symmetric coordinate system is described in [9,11]. In [12,10] the neighbourhood sequences were also used in the hexagonal grid. We note here that the possible extensions of the hexagonal and triangular grids to 3 dimension are the face-centered cubic, the body-centered cubic and the diamond grids. The theory of neighbourhood sequences is also developed to these grids [15,13]. In this paper we will analyse some properties of the distances based on neighbourhood sequences on the hexagonal grid, namely the approximation of Euclidean circles, i.e., disks by digital ones.

Here, instead of the Euclidean geometry, we consider the structure of the hexagonal grid. We use distance functions defined as shortest paths instead of the Euclidean distance. The concept of shortest paths is important in several image processing algorithms such as graph based approaches and algorithms for computing the distance transform etc. For example, the constrained distance transform is much more efficient to compute using the concept of shortest paths compared to when the Euclidean distance is used. Shortest paths can be computed by a standard Dijkstra-like algorithm [13], which is not possible with the Euclidean distance [2]. We note here that there is another way to get less rotational dependent distance functions. The so-called weighted distances give also octagons as digital discs. In [16] it is shown that the best approximating digital discs are dodecagons in the square grid using complex distance functions mixing neighbourhood sequences by weighted distances. We show that similar good results can be obtained on the hexagonal grid in a simpler way, by using only neighbourhood sequences without weights.

One usually used measure of the approximation of the Euclidean distance and circle is the non-compactness ratio (or isoperimetric quotient, that is the ratio perimeter squared divided by the area). It is somehow also a measure of rotational independence of the distance function. It is minimal for circles defined by the Euclidean distance.

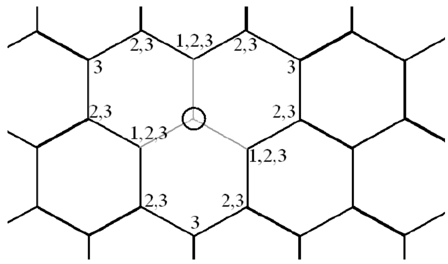


Fig. 1. Types of neighbours in the hexagonal grid

The structure of the paper is as follows. In Section 2 we give our notation, and provide some important properties of the concepts introduced. In Section 3 we characterize the digital discs (some parts are analogous with the results presented in [8]), moreover we compute some geometric details. For small circles the best approximations are presented and a greedy algorithm is shown which provides a good and fast approximation for larger circles. Since we are working on a discrete plane (i.e., in a digital grid) with distances based on neighbourhood sequences, the value of the distances can only be integer, the development of digital disks can be seen as a process on discrete time scale (step by step). In this way finite automata are appropriate choices to present some of the results. Some properties of the obtained disks are also presented, in which the hexagonal grid differs from the square grid. For instance the approximation can be better using the intersection of two discs with the same radius. In the last section we summarize our results.

2 Definitions, Preliminaries

In this section we present some definitions, notation and facts, mostly from the literature mentioned earlier. We also give some useful details and properties of neighbourhood sequences on the hexagonal grid that we use later.

There are usually three types of neighbours defined, as Figure 1 shows, among the nodes of the hexagonal grid: a node and its 12 neighbours are shown. Only the 1-neighbours are directly connected by a side, the other 2- and 3-neighbours are at the positions of shorter and longer diagonals, respectively. These relations are symmetric. In addition, all 1-neighbours of a grid-point are its 2-neighbours and all 2-neighbours are 3-neighbours, as well.

The coordinate values of the grid were introduced as shown in Figure 2. The coordinate axes meet at a grid-point called Origin O having triplet (0,0,0). The axes are at the directions of grid-edges starting at the Origin. The coordinate values of each point can be computed as the sum of steps on the grid-edges taken into direction of the edges. A step by direction of axis x increases the first coordinate value by 1, while a step to inverse direction decreases the first

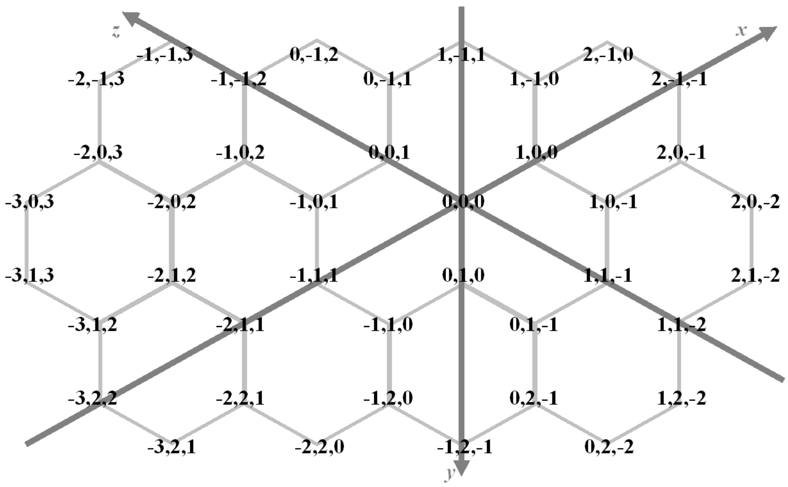


Fig. 2. Coordinate values of nodes

coordinate. Similarly steps on the edges parallel to axis y and z modify the second and third coordinate values, respectively.

By the presented coordinate system every node has a unique triplet which exactly shows the place of the node. We can write the neighbourhood relations in the following formal form.

The points $P(p(1), p(2), p(3))$ and $Q(q(1), q(2), q(3))$ of the hexagonal grid are m-neighbours ($m = 1, 2, 3$), if the following two conditions hold:

(1) $|p(i) - q(i)| \leq 1$, for $i = 1, 2, 3$,
(2) $|p(1) - q(1)| + |p(2) - q(2)| + |p(3) - q(3)| \leq m$.

The hexagonal grid contains exactly those triplets which have sum of coordinate value 0 or 1. We call the points with 0-sum value even (their connections have shape **Y** in the figure); the points with 1-sum are odd (opposite shape).

The formal definition above with the presented coordinate values (Figure 2) gives exactly the neighbourhood relations shown in Figure 1. In [11] the isometric transformations of the grid are described using the coordinate system above. It is very helpful to know the following facts about the coordinate system therefore we recall it.

Proposition 1 *The mirror images of a point with respect to the coordinate axes have the same coordinate values in a permuted way: the images of* $P(x, y, z)$ *are* $P_x(x, z, y)$, $P_y(z, y, x)$ *and* $P_z(y, x, z)$ *with respect to axes* x, y *and* z, *respectively. Moreover* $P_{yz} = P_{zx} = P_{xy}(y, z, x)$ *and* $P_{zy} = P_{xz} = P_{yx}(z, x, y)$ *the mirror images after two operations.*
A point is on an axis if and only if two of its coordinate values are the same. The third value corresponds to the axis in which the point is located.

A point with its (iterated) mirror images represent at most six points. Exactly six points are addressed in the case when there are not equal coordinate values, i.e. the point is not on an axis.

The hexagonal grid is two dimensional. To help compute Euclidean measures of polygons we show how the hexagonal coordinate values (x, y, z) can be represented by the traditional Cartesian coordinate frame. The following proposition can be proved by a straightforward coordinate-geometrical computation.

Proposition 2 *Let the axes of the hexagonal coordinatization be given (as it is shown in Figure 2). Let the Origin of the Cartesian coordinate system be also the point* $(0, 0, 0)$*, and let Cartesian* x' *axis be the same as the hexagonal* x*. Then the Cartesian axis* y' *is orthogonal to axis* x'*, and so to* x*; and the direction of* y' *is growing between the opposite direction of axis* y *and the direction of* z*. Let a point* $P(x, y, z)$ *be given with hexagonal coordinate values. Then its Cartesian coordinate values are:* $(x', y') = \left(x - \frac{y+z}{2}, \frac{\sqrt{3}(z-y)}{2}\right)$.

In this paper, we are dealing with neighbourhood sequences in the hexagonal grid. The sequence $B = (b(i))_{i=1}^{\infty}$, where $b(i) \in \{1, 2, 3\}$ for all $i \in \mathbb{N}$, is called a neighbourhood sequence (on the hexagonal grid). When we need only the initial part up to the l-th element, then we briefly write $B_l = (b(1), b(2), \ldots, b(l))$.

A movement is called a $b(i)$-step when we move from a point P to a point Q and they are $b(i)$-neighbours. Let P, Q be two points and B be a neighbourhood sequence. The point-sequence $P = P_0, P_1, \ldots, P_k = Q$, in which we move from P_{i-1} to P_i by a $b(i)$-step $(1 \leq i \leq k)$, is called a B-path from P to Q of length k. The B-distance $d(P, Q; B)$ from P to Q is defined as the length of the shortest B-path(s).

In this paper we are approximating the Euclidean circle/disk by digital ones. The digital circles/disks are based on digital distances. In this paper we consider digital disks based on distances generated by neighbourhood sequences. Now, we formally define the digital disks that we are going to use.

Definition 3 *Let the digital* B*-disk with radius* k *be the region occupied by* B *after the first* k *steps:*

$$C_{B_k} = \{P : d(O, P; B) \leq k\}.$$

It contains all the points of the grid having B*-distance at most* k *from the Origin.*

In our further analysis we focus on the regions that the neighbourhood sequences occupy and cover after some steps; we consider only the finite initial parts of neighbourhood sequences instead of the whole infinite sequences. Usually several initial parts (of various neighbourhood sequences) define the same region. We call two such initial parts region-equivalent (or simply equivalent) if they occupy the same region.

Informally we can write that the sequence part $(..., 2, 3, ...)$ is region-equivalent to the part $(..., 3, 2, ...)$, and $(..., 1, 2, ...)$ is region-equivalent to $(..., 2, 1, ...)$. The sequence part $(..., 2, ...)$ is region-equivalent to $(..., 1, 1, ...)$. The sequence part $(..., 3, 3, ...)$ is region-equivalent to the part $(..., 3, 2, ...)$. For this purpose the following lemma plays an important role. It can be proven by simple combinatorics by listing all points that can be reached by appropriate steps.

Lemma 4 *Let the initial part $(b(1), ..., b(k))$ of a neighbourhood sequence be given. Concerning the digital disk generated by this initial part (in k steps) the following equivalences are fulfilled.*

(1) If $b(i) = 2, b(i+1) = 3$ for a value i $(1 \leq i \leq k-1)$, then the basic part built up by the values $b'(j) = \begin{cases} 3 & \text{if } j = i, \\ 2 & \text{if } j = i+1, \\ b(j) & \text{otherwise} \end{cases}$
is region-equivalent to the original one.
If $b(i) = 2, b(i+1) = 1$ for a value i $(1 \leq i \leq k-1)$, then the basic part built up by the values $b'(j) = \begin{cases} 1 & \text{if } j = i, \\ 2 & \text{if } j = i+1, \\ b(j) & \text{otherwise} \end{cases}$
is region-equivalent to the original one.

(2) If $b(i) = 2$ for a value i $(1 \leq i \leq k)$, then the basic part $(b'(1), ..., b'(k+1))$ with values $b'(j) = \begin{cases} b(j) & \text{if } j < i, \\ 1 & \text{if } j = i, \\ 1 & \text{if } j = i+1, \\ b(j-1) & \text{if } j > i+1 \end{cases}$
is region-equivalent to the original one.

(3) If $b(i) = 3, b(i+1) = 3$ for a value i $(1 \leq i \leq k-1)$, then the basic part built up by the values $b'(j) = \begin{cases} 2 & \text{if } j = i+1, \\ b(j) & \text{otherwise} \end{cases}$
is equivalent to the original one.

Since the region-equivalence is a symmetric relation they also work in the opposite direction. Moreover all the equivalences can be given by finitely many *applications of the above three equivalences.*

The first statement of Lemma 4 allows to move the values 2 within (the used initial part of) the sequence, while using the second statement one can replace every occurrence of two consecutive 1's to a value 2 or vice-versa. It is very important that by this fact one can modify the number of steps without changing the generated digital disk, i.e. the radius of the disk is not well-defined. By the definition of our disks it is not surprising that different sequences can yield the same disk. This is also the case in $\mathbb{Z}^n$, where the elements of the neighbourhood sequence used can freely be permuted without changing the digital object obtained. However in $\mathbb{Z}^n$ all the appropriate neighbourhood sequences need the same number of steps to define the same region. Here we have an entirely different case. We will detail this phenomenon later on. The third statement of the lemma allows to replace values 3 by value 2 in some places; it is interesting because it changes a value present in the neighbourhood sequence.

We already defined the region-equivalence among the initial parts of neighbourhood sequences. This relation will be crucial in our paper. There was another equivalence relation defined among the neighbourhood sequences that we also will use. The two concepts have some connections.

Opposite to the square grid (and generally to $\mathbb{Z}^n$) in the hexagonal grid there are neighbourhood sequences that generate the same distance functions. B and B' generate the same distance if the B-distance of any point pair is the same as their B'-distance. For instance the neighbourhood sequence containing only 3's ($B = (3, 3, 3, 3, ...)$) generate the same distance function as any neighbourhood sequences starting with a 3 and not containing any 1's. This is due to the property of the grid, that one cannot go further from a point by two strict 3-steps than a 3-step and a 2-step. The part $(..., 3, 3, ...)$ is not only region equivalent to the part $(..., 3, 2, ...)$, but 'stepwise' region-equivalent, i.e., the distance function is the same. The neighbourhood sequences generating the same distance function form equivalence classes. The next definition and lemma can be found in [12] and they describe these equivalence classes.

Definition 5 *Let $B = (b(i))_{i=1}^{\infty}$ and $B' = (b'(i))_{i=1}^{\infty}$ be two neighbourhood sequences. B' is the* minimal equivalent neighbourhood sequence *of B, if the following conditions hold:*

(1) $d(P, Q; B) = d(P, Q; B')$ for all point pairs P, Q, and
(2) for each neighbourhood sequence $B'' = (b''(i))_{i=1}^{\infty}$, if $d(P, Q; B) = d(P, Q; B'')$ for all point pairs P, Q, then $b'(i) \leq b''(i)$ for all i.

Lemma 6 *The minimal equivalent neighbourhood sequence B' of B is uniquely determined, and it is given by*

- $b'(i) = b(i)$, *if* $b(i) < 3$,
- $b'(i) = 3$, *if* $b(i) = 3$ *and there is no* $j < i$ *such that* $b'(j) = 3$,
- $b'(i) = 3$, *if* $b(i) = 3$ *and there is some* $b'(l) = 3$ *with* $l < i$, *and* $\sum_{k=j+1}^{i-1} b'(k)$ *is odd, where* $j = \max\{l | l < i, b'(l) = 3\}$,
- $b'(i) = 2$, *otherwise.*

Now we sketch an alternative proof based on Lemma 4.

PROOF. The generated distance function is the same for two neighbourhood sequences if and only if the same sequence of digital disks can be obtained. The third statement of Lemma 4 allows to replace values 3 by value 2 in places immediately after a value 3. Combining it with the first two statements (a value 2 can be replaced by 11 an vice versa, moreover a value 2 can freely move through the sequence and back) one can change any value $b(i) = 3$ to 2 for which the subsum $\sum_{k=j+1}^{i-1} b(k)$ of the neighbourhood sequence after the previous value 3 ($= b(j)$) is even. (By convention the empty sum is 0, in case of $i = j + 1$ meaning two consecutive 3's.) □

The minimal equivalent neighbourhood sequence is elementwise minimal among the neighbourhood sequences generating the same distance function. Every equivalent class of neighbourhood sequences can and will be represented by the minimal equivalent neighbourhood sequence of the class.

Before we detail the approximation of Euclidean circles, i.e., disks, we would like to present some further important properties of the neighbourhood sequences on the hexagonal grid.

Based on the defined equivalence of initial parts of neighbourhood sequences (occupance of the same region) we present a finite class of them that is sufficient to generate all possible digital disks. This classification procedure is the following.

In the next definition we write the initial parts of the neighbourhood sequences as strings over the alphabet $\{1, 2, 3\}$, i.e., we do not care about the separations, and parentheses can be used as technical symbols in the description. We will also use this string form of initial parts later.

Definition 7 *An initial part of a neighbourhood sequence is a basic part, if it can be described by one of the following forms:*

(A)	(B)	(C)	(D)	(E)
1^l	$(13)^n 2^m$	$1(31)^n 2^m$	$3(13)^{n-1} 2^m$	$(31)^n 2^m$

where $n, m \in \mathbb{N}$ *and* w^j *is the repetition of the string* w j *times as usual (allowing the empty string* w^0*). In cases* (B)–(E) *we suppose that* $n \geq 1$.

The following proposition shows that it is sufficient to deal with the above defined classes of basic parts in our approximation procedure.

Proposition 8 *Let* B *be a neighbourhood sequence on the hexagonal grid. Every finite initial part of* B *has an equivalent basic part.*

PROOF. If the initial part is also a basic part, then there is nothing to prove, therefore we deal with only possible initial parts that are not basic parts.
If the initial part does not contain 3, then by using equivalence (2) of Lemma 4 for every value 2 one can obtain an initial part of the form (A).
If there is at least one 3 in the initial part, then by equivalence (2) one can replace each consecutive pair of 1's by a 2. Then by equivalence (1) of Lemma 4 one can move all values 2 to the end of the initial part. (If necessary these two steps can be repeated.) Finally there is no consecutive occurrence of 1's, and all 2's in the end of the initial part. Now the second element of every consecutive 3's can be replaced by a 2 with equivalence (3) and it can be moved to the end of the initial part by equivalence (1). The result will not contain consecutive 1's and consecutive 3's and all the values 2 will be in the end. This fact can be fulfilled in four different ways. All 4 possibilities are presented as type (B), (C), (D) and (E). $\square$

It is easy to show that by our assumption ($n \geq 1$) the five types of basic parts presented are pairwise disjoint sets.

Moreover, the equivalences presented in Lemma 4 are enough to get any initial sequence from any other one from the same equivalence class. Formally, the next theorem can be obtained.

Theorem 9 *The associative calculus* $W(\{1, 2, 3\}, \{23 \leftrightarrow 32, 21 \leftrightarrow 12, 33 \leftrightarrow 32, 11 \leftrightarrow 2\})$ *describes the equivalence classes of the initial parts written in string from.*

PROOF. The statement follows from the fact (Proposition 8) that every initial part can be transformed to an equivalent basic part. $\square$

Since the digital disks are finite sets of points we will use the convex hull of them in our purpose. In this way we work in our approximations with real Euclidean objects (with perimeter and area), namely with digital discs. We note here that the generated discs obtained by the union of triangles on the triangular grid are also real Euclidean objects (they are polygons), but they are usually not convex (they can have 'hilly' and 'sawtooth' sides, see [8]). Therefore it is not convenient to use them in our approximations. Moreover they are represented on the dual grid of the hexagonal, and so their characterization and classes are slightly different from our present characterization and classes. In this way to continue the research started in [8] we interchanged the grid used to its hexagonal representation.

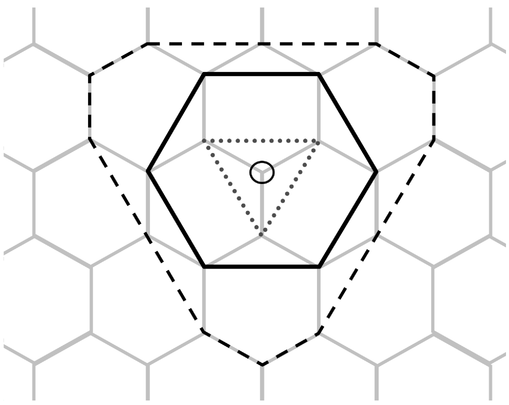
There are several ways to compare digital disks (or other objects) to the Euclidean disk. Since the radius of our digital discs is not always well-defined we need a measure which is independent of the radius.

A standard geometric measure of the approximations of Euclidean circles or disks is the *noncompactness ratio*. It is computed as $\frac{\text{perimeter}^2}{\text{area}}$ for any object in the plane. In Euclidean geometry, according to the isoperimetric inequality, this ratio is never less than 4π, and is minimal for the circle (disk). So, the noncompactness ratio of the Euclidean circle is $4\pi \approx 12.566$; it is 16 for squares; and for regular hexagon, octagon, enneagon and dodecagon it is approximately 13.856, 13.255, 13.103 and 12.862, respectively.

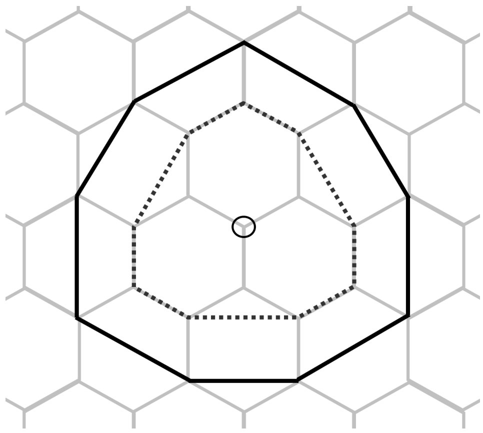
3 Approximating the Euclidean disc

3.1 Digital discs on the hexagonal grid

First we present some examples for the digital disks in Figure 3.



(a)



(b)

Fig. 3. Examples of digital disks: (a) Digital disks by (1) (dotted line, radius 1), (1,1) (solid line, radius 2) and (1,3) (dashed line, radius 3). (b) Digital disks by (3) (dotted line, radius 1) and (3,1) (solid line, radius 2).

In the figures we will use the convex hull as the occupied polygon of the digital disk. These polygons have sides and corners in the usual sense. Since they are convex, the sets of the coordinate triplets of corners describe them. (We use the term corner only for angles less than π.)

As we already mentioned, usually a digital disc can be generated in several ways. The disk $C_{(2)}$ is the same as the disk $C_{(1,1)}$ (see Figure 3 (a), where $C_{(1)}$ is also shown by dotted line), so the radius of this disc is not well-defined, i.e., it is not unique, in the sense that having only the disc one may not be sure about its radius. Another interesting fact can also be observed in Figure 3: opposite to the grids $\mathbb{Z}^n$ the permutation of the used elements of a neighbourhood sequence may generate different discs. The elements 3 and 1 are not freely permutable in the sequence. The disks generated by (3,1) and (1,3) are incomparable. This fact is related to the existence of non-symmetric distances on the hexagonal grid [12]. All possible permutations that generate the same disc was already discussed in Lemma 4.

In the next part we present algorithms (and state-diagrams in the form of automata) computing the geometrical data of the digital disks, such as coordinate values of corners, sidelengths and type of the polygon.

Based on technical calculations, the next incremental algorithm gives the corners of polygons occupied by a neighbourhood sequence step by step. Not only the coordinate values of the corner points but their various types are also indicated. The polygon has angle $\frac{\pi}{3}$ at corners type α, $\frac{2\pi}{3}$ at the corners type β and γ, and $\frac{5\pi}{6}$ at corners type δ. The indices will have the following meaning: e means that the corner is on an even point, o means that it is on an odd point. Every corner type γ is located on a coordinate axis, it can be y and z for the computed points. (Corners type β are not on any coordinate axis.) One of the sides of corners type δ is perpendicular to a coordinate axis. This is the second index of these corners.

Similar algorithm can work for the infinity if one wants to continue the process beyond any finite value of k.

Figure 4 shows the state-transition diagram of the possible corners according to Algorithm 1. There are some places on the figure where two edges start simultaneously (with element 3 of the neighbourhood sequence). In these steps two new corners of different types are obtained. These cases are exactly those ones in which the points P_1 and P_2 start to develop in different ways. (Note that we use the region $z \geq x \geq y$ in our computation, all other parts can be obtained by permuting the coordinate values.) One can easily extend this machine (with tapes) to a more complex one that computes also the coordinate values.

The coordinates of the vertices can be computed step by step based on the

Algorithm 1: Computing corners of the digital disc C_{B_k}.

Input: neighbourhood sequence B, radius $k > 0$.

Output: The points P_1, P_2 and all the points that are obtained as coordinate permutation of any of them as the corners of the digital disc C_{B_k}.

If $b(1) = 1$ **then** let $P_1 = (0,0,1)_\alpha$ and $P_2 = (0,0,1)_\alpha$.
If $b(1) = 2$ **then** let $P_1 = (0,-1,1)_{\beta_e}$ and $P_2 = (0,-1,1)_{\beta_e}$.
If $b(1) = 3$ **then** let $P_1 = (0,-1,1)_{\delta_{e,z}}$ and $P_2 = (1,-1,1)_{\gamma_{o,y}}$.
for $j = 2$ *to* k **do**
- **If** $b(j) = 1$ **for** $i = 1,2$ **do**
  - **If** $P_i = (x,y,z)_\alpha$ **then** let $P_i' = (x,y-1,z)_{\beta_e}$.
  - **If** $P_i = (x,y,z)_{\beta_e}$ **then** let $P_i' = (x,y,z+1)_{\beta_o}$.
  - **If** $P_i = (x,y,z)_{\beta_o}$ **then** let $P_i' = (x,y-1,z)_{\beta_e}$.
  - **If** $P_i = (x,y,z)_{\gamma_{o,y}}$ **then** let $P_i' = (x,y-1,z)_{\gamma_{e,y}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{e,y}}$ **then** let $P_i' = (x,y,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{o,z}}$ **then** let $P_i' = (x,y-1,z)_{\delta_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{e,z}}$ **then** let $P_i' = (x,y,z+1)_{\gamma_{o,z}}$.
  - **If** $P_i = (x,y,z)_{\delta_{o,y}}$ **then** let $P_i' = (x,y-1,z)_{\delta_{e,y}}$.
  - **If** $P_i = (x,y,z)_{\delta_{e,y}}$ **then** let $P_i' = (x,y,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\delta_{o,z}}$ **then** let $P_i' = (x,y-1,z)_{\delta_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\delta_{e,z}}$ **then** let $P_i' = (x,y,z+1)_{\delta_{o,z}}$.
- **If** $b(j) = 2$ **for** $i = 1,2$ **do**
  - **If** $P_i = (x,y,z)_\alpha$ **then** let $P_i' = (x,y-1,z+1)_{\beta_o}$.
  - **If** $P_i = (x,y,z)_{\beta_e}$ **then** let $P_i' = (x,y-1,z+1)_{\beta_e}$.
  - **If** $P_i = (x,y,z)_{\beta_o}$ **then** let $P_i' = (x,y-1,z+1)_{\beta_o}$.
  - **If** $P_i = (x,y,z)_{\gamma_{o,y}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{e,y}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{e,y}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{o,z}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{o,z}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{e,z}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\delta_{o,y}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\delta_{e,y}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{e,y}}$.
  - **If** $P_i = (x,y,z)_{\delta_{o,z}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{o,z}}$.
  - **If** $P_i = (x,y,z)_{\delta_{e,z}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{e,z}}$.
- **If** $b(j) = 3$ **for** $i = 1,2$ **do**
  - **If** $P_1 = (x,y,z)_\alpha$ **then** let $P_1' = (x-1,y-1,z+1)_{\gamma_{e,z}}$.
  - **If** $P_2 = (x,y,z)_\alpha$ **then** let $P_2' = (x,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_1 = (x,y,z)_{\beta_e}$ **then** let $P_1' = (x,y-1,z+1)_{\delta_{e,z}}$.
  - **If** $P_2 = (x,y,z)_{\beta_e}$ **then** let $P_2' = (x+1,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_1 = (x,y,z)_{\beta_o}$ **then** let $P_1' = (x,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_2 = (x,y,z)_{\beta_o}$ **then** let $P_2' = (x,y-1,z)_{\delta_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{o,y}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{e,y}}$ **then** let $P_i' = (x+1,y-1,z+1)_{\gamma_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{o,z}}$ **then** let $P_i' = (x-1,y-1,z+1)_{\gamma_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\gamma_{e,z}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\delta_{o,y}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\delta_{e,y}}$ **then** let $P_i' = (x+1,y-1,z+1)_{\delta_{o,y}}$.
  - **If** $P_i = (x,y,z)_{\delta_{o,z}}$ **then** let $P_i' = (x-1,y-1,z+1)_{\delta_{e,z}}$.
  - **If** $P_i = (x,y,z)_{\delta_{e,z}}$ **then** let $P_i' = (x,y-1,z+1)_{\delta_{e,z}}$.
- Let $P_1 = P_1'$ and $P_2 = P_2'$.

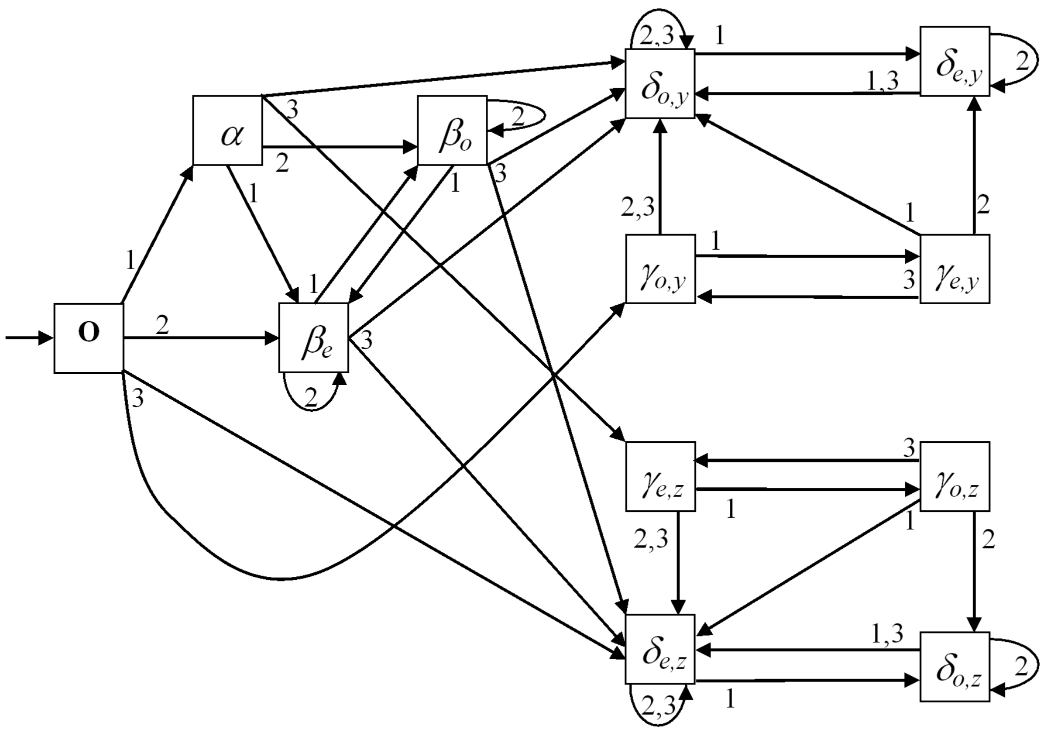


Fig. 4. State transition of types of corners of digital disks.

neighbourhood sequence with incremental (linear) algorithm. Now we give simple formulae to compute the coordinate values of the vertices of the digital discs. Based on Proposition 8 we do not need to follow the neighbourhood sequence step by step, it is enough to know the basic part that generates the desired polygon. The formula depends on the type of the disc, i.e., on the type of the basic part the disc can be obtained with.

Theorem 10 *The corners of the (convex hull of) digital disks are the following.*

- *type* (A) $\begin{cases} l = 1: & \alpha(0,0,1) & \\ l = 2k: & \beta(0,-k,k); & (k > 0) \\ l = 2k+1: & \beta(0,-k,k+1); & (k > 0) \end{cases}$

- *type* (B) $\begin{cases} m = 0: & \gamma(-n,-n,2n), & \delta(n-1,-2n+1,n+1) \\ m > 0: & \delta(-n,-n-m,2n+m), & \delta(n-1,-2n-m+1,n+m+1) \end{cases}$

- *type* (C) $\begin{cases} m = 0: & \gamma(-n,-n,2n+1), & \delta(n-1,-2n,n+1) \\ m > 0: & \delta(-n,-n-m,2n+m+1), & \delta(n-1,-2n-m,n+m+1) \end{cases}$

- *type* (D) $\begin{cases} m=0: & \gamma(n,-2n+1,n), \quad \delta(-n+1,-n,2n-1) \\ m>0: & \delta(n,-2n-m+1,n+m),\ \delta(-n+1,-n-m,2n+m-1) \end{cases}$
- *type* (E) $\begin{cases} m=0: & \gamma(n,-2n,n), \quad \delta(-n+1,-n,2n) \\ m>0: & \delta(n,-2n-m,n+m),\ \delta(-n+1,-n-m,2n+m) \end{cases}$

and all of their permutations.

PROOF. We omit the technical calculations using Algorithm 1 for the basic parts. □

Based on the various corners of the polygons their shapes can be described.

Theorem 11 *The shape of the digital disk generated by the neighbourhood sequence B in k steps is a triangle if and only if it is $C_{(1)}$. The shape is a hexagon if and only if there is no element 3 in the initial part of B up to the k-th element. (Except the previous case, in which $C_{(1)}$ is especially a triangle.) The shape is an enneagon if and only if there is neither any element 2 nor any consecutive 1,1 or 3,3 occurring in the initial part B_k. (Except $C_{(1)}$.) In every other case the digital disk is a dodecagon.*

PROOF. In the first case the triangle has 3 α corners. According to Algorithm 1 only other types of corners can occur at the other digital disks. With other type of corners it is impossible to get a triangle.

Now let us consider the other digital disks. It is easy to check by Algorithm 1 that starting with an element 2 we get a hexagon with six β corners. Moreover it is clear that using 1-step(s) and/or 2-step(s) from corners type α and from corners type β the new corners will be type β as well. With only corners type β there must be six of them to make a polygon. Therefore without a 3-step the result is a hexagon with corners type β.

It is shown in Figure 3 (b) that $C_{(3)}$ is an enneagon, it has 3 type γ and 6 type δ corners. The corner γ $(1,-1,1)$ is odd and it is on the axis y. Therefore with a 1-step it grows to a $\gamma_{e,y}$ which is even and on the same axis. (With a 2-step or a 3-step the corner would be divided to two δ vertices that are symmetric pairs to axis y, and a side connect them orthogonally to this axis.) From $\gamma_{e,y}$ with a 3-step $\gamma_{o,y}$ is resulted; it is odd on axis y. (From $\gamma_{e,y}$ with a 1-step or a 2-step we would obtain two δ corners.) Since $\gamma_{o,y}$ is exactly the same type as γ $(1,-1,1)$ the cycle is repeating till only 1-steps and 3-steps turn by turn. Observe in Figure 4 that corner type γ can be obtained from α, but not from β. Therefore there is no way to get γ vertices from $C_{(2)}$ and so

(A) ($l = 1$)	(A) ($l \geq 1$)	(B)–(E) ($m = 0$)	(B)–(E) ($m \geq 1$)
triangle	hexagon	enneagon	dodecagon

Table 1
The possible types of polygons depending on the used basic part

from any hexagon. From $C_{(1)}$ one can obtain an enneagon in one way, namely to get $C_{(1,3)}$. The obtained corner γ $(-1,-1,2)$ even and it is on the axis z. One can check that there is a similar computing cycle for this γ to keep it with only 1-steps and 3-steps by turns. (Leaving this computing cycle two corners type δ are obtained instead of the type γ corner.)

Finally, in all other cases the polygon has only δ vertices. When 12 corners type δ are in a digital disk, then it never happen that they change to another type (see Algorithm 1 and Figure 4) and twelve of them are needed for a polygon. Therefore the last statement is proved. □

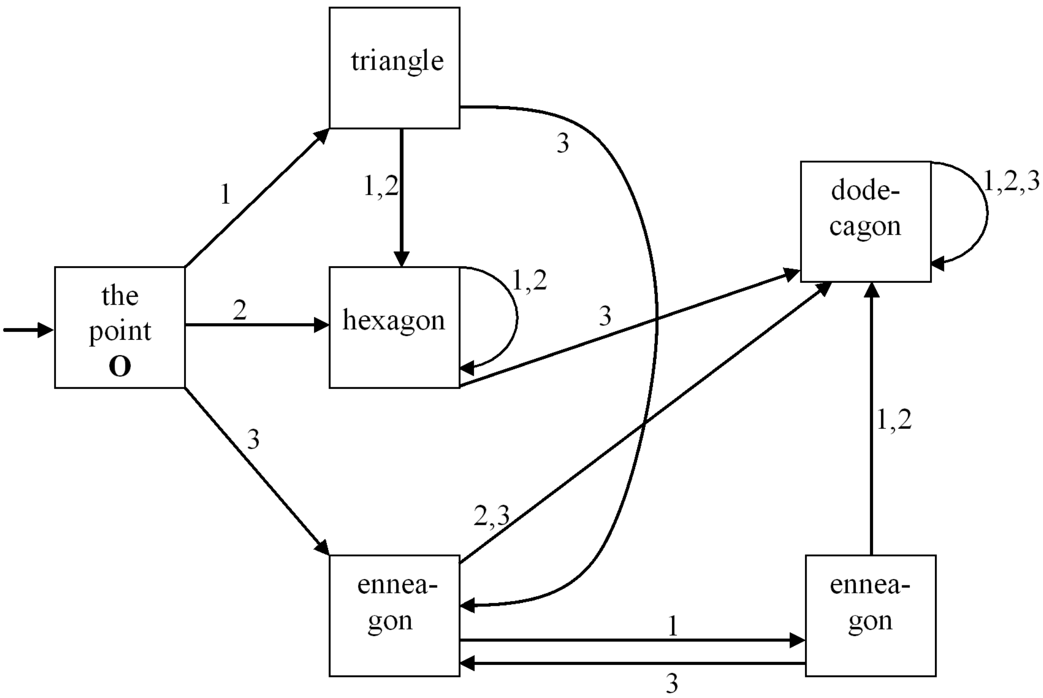


Fig. 5. State transition of possible types of digital disks.

Figure 5 also shows the resulted polygons. Various polygons can be obtained as digital disks on the hexagonal grid. Since all digital disks can be obtained by using only basic parts, we show a characterization in this way. Table 1 summarizes the types of the possible polygons that can be produced by using various basic parts.

Figure 6 shows an automaton that computes the type of the basic sequence of the obtained disc. Moreover it computes also the parameters l, m, n for the various types. Remember that l is the number of 1's in type (A), in other types m and n are the number of 2's and 3's, respectively.

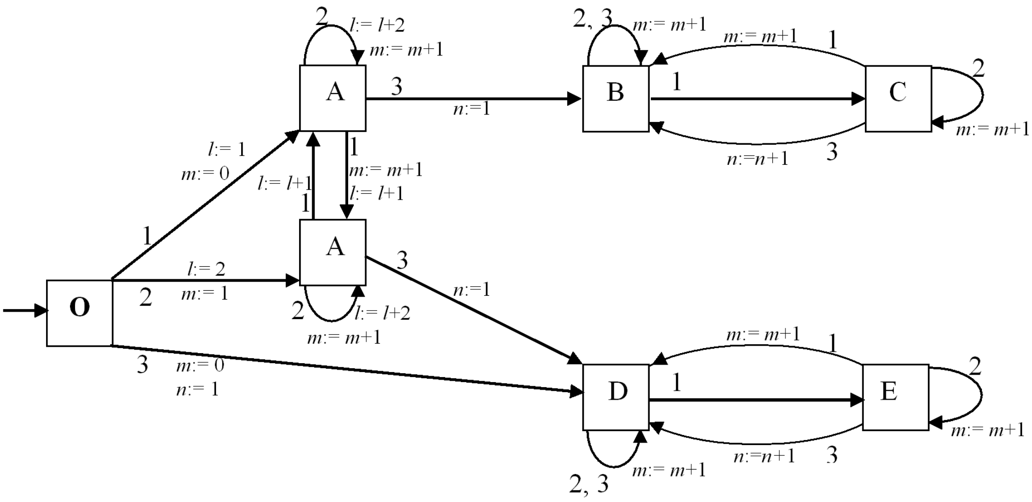


Fig. 6. State transition of types of basic parts of neighbourhood sequences extended by computation of parameters.

Now the side-lengths of the polygons are computed. There can be three different side-lengths in an obtained dodecagon. There can be two types of sides (three-three sides) that are orthogonal to one of the coordinate axes. (At triangle and at enneagons one type of these sides has length 0, therefore these cases are degenerate cases of the hexagons and dodecagons.) Moreover there can be (and there are at enneagon and dodecagon) six sides that are parallel to one of the axes. Let the length of the sides orthogonal to an axis be a_1 and a_2 with $a_1 \geq a_2$. See Figure 7. These sides form the triangle and the hexagons (using basic part type (A)). At enneagons and dodecagons these sides are between two type δ corners such that they are symmetric images of each other and so the side is orthogonal to an axis. Let the length of the sides parallel to an axis be a_3. These sides can be only between corners type γ and δ, and between two corners type δ if the side do not meet with any of the axes. The last case corresponds to sides between points P_1 and P_2 of Algorithm 1.

Theorem 12 *The lengths of different sides (for notation see Figure 7) are the following.*

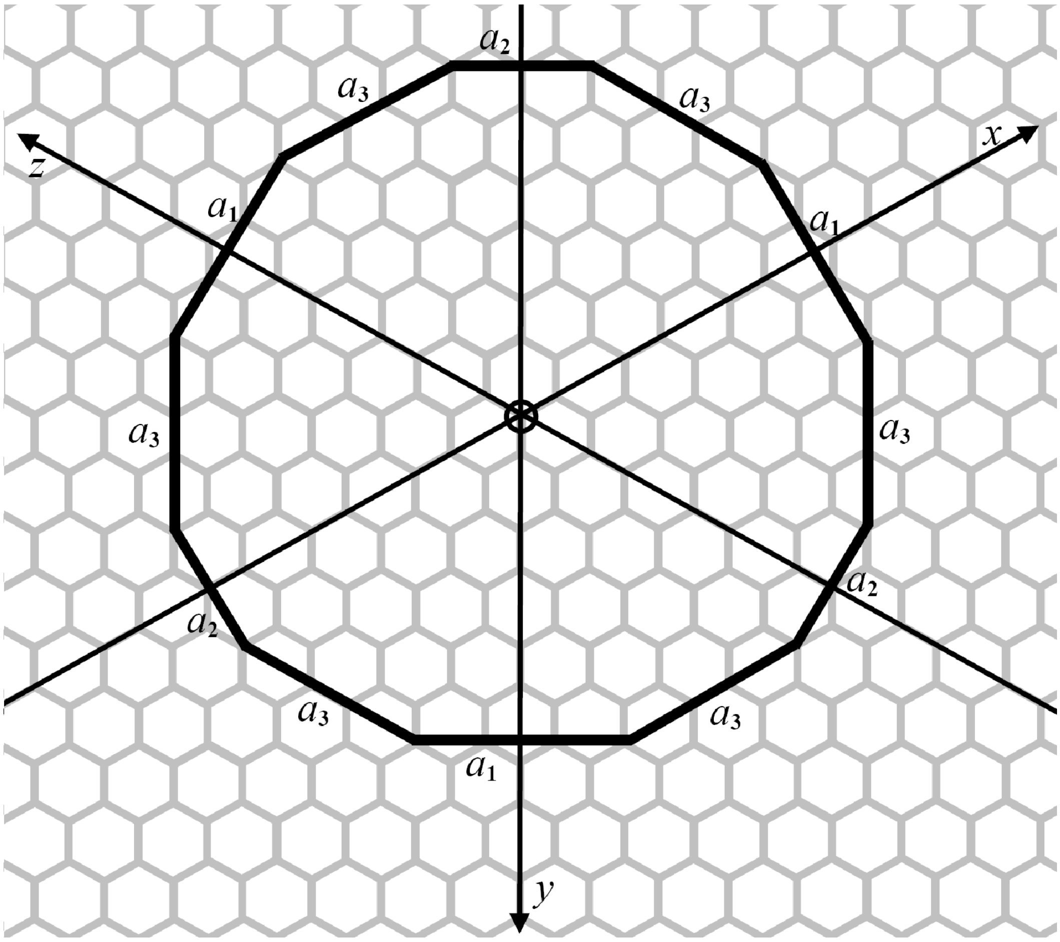


Fig. 7. The side-lengths of the polygons: $a_1 \geq a_2$ are the lengths of sides orthogonal to a coordinate axis; a_3 is the length of sides parallel to a coordinate axis.

type	case	length of a_1	length of a_2	length of a_3
(A)	$l = 2k \ (k > 0)$	$k\sqrt{3}$	$k\sqrt{3}$	0
	$l = 2k + 1 \ (k \geq 0)$	$(k+1)\sqrt{3}$	$k\sqrt{3}$	0
(B)		$(m+2)\sqrt{3}$	$m\sqrt{3}$	$3n - 2$
(C)		$(m+2)\sqrt{3}$	$m\sqrt{3}$	$3n - 1$
(D)		$(m+1)\sqrt{3}$	$m\sqrt{3}$	$3n - 2$
(E)		$(m+1)\sqrt{3}$	$m\sqrt{3}$	$3n - 1$

PROOF. The calculation is based on the coordinate values of the corners given in Theorem 10 and on Proposition 2. □

The diagram-automaton of Figure 6 can easily be modified to compute the side-lengths of the polygon in a similar way as the parameters l, m, n are computed. A part of this modified automaton can be seen in Figure 8.

Now we are ready to give the necessary measures for the non-compactness ratio.

Theorem 13 *The perimeter (L) and area (A) of various digital disks are computed in the following way (for notation see Figure 7):*

$$L = 3a_1 + 3a_2 + 6a_3$$

$$A = \sqrt{3}\left(\frac{a_1^2}{4} + \frac{a_2^2}{4} + \frac{3a_3^2}{2} + a_1a_2\right) + 3a_1a_3 + 3a_2a_3.$$

PROOF. The computation is based on (coordinate) geometry. □

So one needs only to substitute the values of the side-lengths that given by the function of the values 2 and 3 for various classes of basic types (from Theorem 12).

We note here that the automata of types of polygons, corners and side-lengths (Figures 5,4,6) can be used independently as generator devices, but one can join them to a large automaton with several tapes (input tape: elements of neighbourhood sequence, output tapes: types, corners etc.).

3.2 The best approximating neighbourhood sequence

Since there are several neighbourhood sequences that generate the same sequence of discs, the minimal equivalent neighbourhood sequence representing the class is used in the approximation.

For very small digital disks the following ones are the best approximating ones: For $k = 1$ there are three possibilities; they are shown in the following table:

polygon	$C_{(1)}$	$C_{(2)}$	$C_{(3)}$
shape	triangle	hexagon	enneagon
non-compactness ratio	20.785	13.856	13.785

Observing all possible initial parts of neighbourhood sequences with 2,3,4 and 5 elements, the best approximating discs are the following ones:

length of the initial part	best initial part	non-compactness ratio
2	(3,2)=(2,3)=(3,3)	13.287
3	(3,2,1)=(3,1,2)=...	13.020
4	(3,2,1,3)=(3,1,3,2)=...	12.972
5	(3,2,1,3,2)=(3,1,3,2,2)=...	12.906

The type of all these discs is dodecagon. We can state that up to the first 5 steps the initial part (3,2,1,3,2) is the best approximating one, there is no neighbourhood sequence that gives better result for any of the radii 1,...,5. (One can check this fact by enumerating and check all possibilities.) In the table the basic part generating the disc is also shown among possible initial parts generating the same disc.

In other side, in the limit $k = \infty$ the regular dodecagon can be obtained by any of the types (B)–(E). If the ratio $\frac{a_1}{a_3}$ goes to 1, then the regular dodecagon is obtained in the limit.

Based on these facts we propose a neighbourhood sequence that can be obtained by a greedy algorithm which gives a very good approximation.

Instead of computing all possibilities at every step we use the next heuristics. The more sides (and corners) of the polygon the better the approximation can be. The non-compactness ratio is lower (i.e., better) for polygons closer to the regular one. Therefore we propose a fast algorithm that tries to keep the sidelengths of the obtained dodecagon as close to each other as possible. The side-lengths of a_1 and a_2 are developing in the same way (see Theorem 12). The difference of their length is fixed as $2\sqrt{3}$ at discs generated by basic parts type (B) and (C), while it is fixed as $\sqrt{3}$ at type (D) and (E). Therefore we use the latter types in our approximation. Figure 8, in form of an automaton, shows how the sidelengths are changing. The figure shows only the modified part part of Figure 6 which is needed in our analysis, dealing with neighbourhood sequences start with 3.

The algorithm chooses the next value of the neighbourhood sequence such that the length a_3 is the closest to the average of a_1 and a_2 among the possibilities.

The best approximating neighbourhood sequence up to the first 20 elements is: B_{opt} =(3,2,1,3,2,1,2,3,2,1,2,3,2,1,3,2,1,2,3,2). The noncompactness values of the generated digital disks can be seen in Figure 9. One can check that Algorithm 2 gives exactly the best choices for these steps, i.e., the next value of the neighbourhood sequence corresponds to the disc having the minimal non-compactness ratio among the possible next discs. We note here that it is impossible to have the sequence of discs that contain only the best approx-

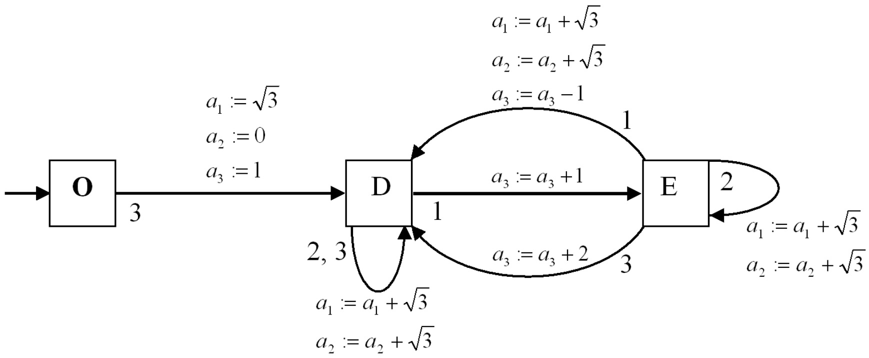


Fig. 8. State transition of types of basic parts starting with a 3, extended by computation of side-lengths.

Algorithm 2: Computing neighbourhood sequence for approximating the Euclidean distance.

Input: the number of steps k.
Output: $(b(1), ..., b(k))$ as a very good approximation through k steps.
Let $b(1) = 3$. **If** $k = 1$ **then** end.
Let $b(2) = 2$. **If** $k = 2$ **then** end.
Let $b(3) = 1$. **If** $k = 3$ **then** end.
Let $b(4) = 3$. **If** $k = 4$ **then** end.
Let $b(5) = 2$. **If** $k = 5$ **then** end.
Let $m = 2, n = 2$, let type=D, type'=D.
for $j = 6$ *to* k **do**
    if type=D **then**
        if $|\frac{2m+3}{2}\sqrt{3} - 3n + 2| \geq |\frac{2m+1}{2}\sqrt{3} - 3n + 1|$ **then**
            let $b(j) = 1$ and type'=E.
        else
            let $b(j) = 2, \ m = m + 1$
    if type=E **then**
        if $|\frac{2m+3}{2}\sqrt{3} - 3n + 1| \geq |\frac{2m+3}{2}\sqrt{3} - 3n + 2|$ **then**
            if $|\frac{2m+1}{2}\sqrt{3} - 3n - 1| \geq |\frac{2m+3}{2}\sqrt{3} - 3n + 2|$ **then**
                let $b(j) = 1, \ m = m + 1$ and type'=D.
            else
                let $b(j) = 2, \ m = m + 1$
        else
            let $b(j) = 3, \ n = n + 1$, type'=D
    type=type'

---

imating ones. The length of the various sides are growing via various steps, therefore the changes of the ratio of the sides fluctuate around the optimal value. The lengths are increasing with values with an irrational ratio, therefore the fluctuation cannot be periodic. In the given optimal initial part the non-compactness ratio of $C_{B_{opt}\ 10}$ is less than the ratio of $C_{B_{opt}\ 11}$. It can happen,

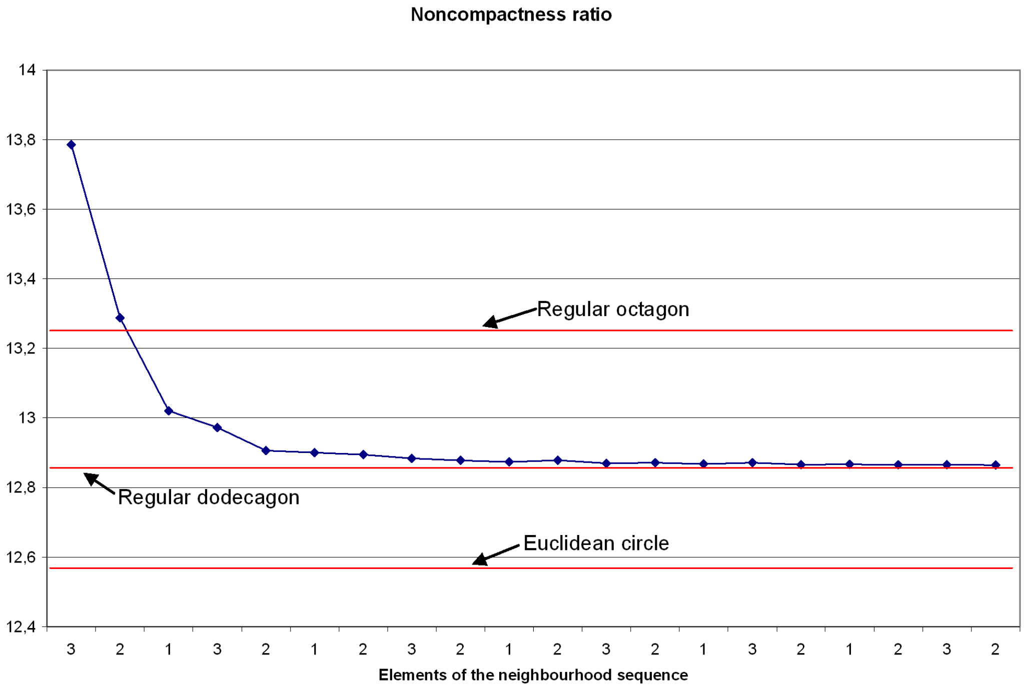


Fig. 9. Non-compactness ratio of the discs obtained by Algorithm 2.

if the approximation in a step is very good (the side-lengths are close to each other). Roughly speaking, the next step must increase the lengths of a_1 and a_2 or (exclusive or) a_3. In this way the next dodecagon is a little bit further from the regular one. Remember now that the radius of these discs is not well defined. It allows us to use the previous disc as a better approximation to the Euclidean with the given radius: replacing any value 2 among the first eleven values of the sequence by two consecutive 1's, $C_{B_{opt}\ 10}$ is obtained as a disc with radius 11 in the previous example. The same trick can be used for discs having radii 13, 15, 17 and 19. Note that the obtained disc with radius 20 has non-compactness ratio 12.86457 that is 1.000234 times more than the ideal (regular dodecagon).

In [4] Das and Chatterji showed that for every initial part of a neighbourhood sequence in the square grid (using Cartesian coordinates and two types of neighbours), the obtained digital disk is always an octagon. These octagons can be degenerated, so the digital disks are squares in the following cases: Using only 1-steps (cityblock) in the initial part of the neighbourhood sequence we get only four sides (the corners will be $(0,k),(k,0),(0,-k),(-k,0)$), while using only 2-steps (chessboard) we get a square with corners $(k,k),(k,-k)$, $(-k,-k)$, $(-k,k)$. In the case when we use both 1-step(s) and 2-step(s) our result is a non-degenerated octagon.

The approximation of large disks can be very close to the regular octagon, and it is achieved in the limit ($r \to \infty$). Since the noncompactness ratio of the regular octagon 13.255, i.e., it is much higher than the value for our dodecagons, it is clear that the approximation is much more effective in the hexagonal grid than in the square grid (see Figure 9). Our approximation was better from the third step (for disk with radius 3) than the best approximation on the square grid that can be obtained in the limit $k \to \infty$.

Since the side-lengths of the obtained dodecagons are increasing in the way as presented in Figure 8 and in Algorithm 2, their ratio tends to 1. The increasing process is close to linear, and the sum of the lengths increasing at least by 3 in any two consecutive steps. It means that every side grows at least by 0.5 average in a step. The maximum difference of side-lengths cannot be larger as a constant, let us say $2+2\sqrt{3}$. In this way we have a very rough approximation: the ratio of side-lengths has deviation less than $\frac{4+4\sqrt{3}}{k}$ that tends to 0.

To get the optimal approximation of a Euclidean circle/disk, we use types (D) and (E). For approximation with radius greater than 2, this gives dodecagons with corners of the same type (δ) and side lengths a_1, a_2, a_3. In Theorem 12, the side-lengths are given as functions of the number of the number of 2's (m) and 3's (n) in the used initial part of the neighbourhood sequence. Since we are dealing only with initial parts of the neighbourhood sequences written as a basic part, see Proposition 8, the number of 1's (l) in the used part of the neighbourhood sequence satisfies $l = n$ or $l = n - 1$ (depending on the type). Asymptotically $a_1 = a_2 = a_3$ is, by Theorem 12, $\sqrt{3}l = m = \sqrt{3}n$. Thus, asymptotically, the number of 1's equals the number of 3's and the number of 2's is a factor $\sqrt{3}$ greater.

We note here that by distances based on neighbourhood sequences using weights on the square grid one can obtain dodecagons. In this way the approximation on the square grid is similarly efficient, but the main difference is that we used pure distances based on neighbourhood sequences without weights. So our approximation method is much simpler, using the hexagonal grid.

3.3 Approximation by two neighbourhood sequences

Prof. A. Kuba proposed the idea to approximate the Euclidean distance by two neighbourhood sequences simultaneously in a private communication during the 4th Hungarian Conference on Image Processing in 2004. Opposite to the square grid (and $\mathbb{Z}^n$) there are digital discs with the same radius that are not comparable (for instance $C_{(3,1)}$ and $C_{(1,3)}$).

Therefore it is worth to consider the combination of two discs (or two sequences of discs generated by two neighbourhood sequences). In this section we present some initial results.

The intersection of two digital discs will be used, since the intersection preserves the convexity [17]. We believe that a good approximation of the Euclidean circle should go by the help of convex polygons.

Example 14 *Let the approximation for disk* $k = 4$ *be the intersection of digital disks* $C_{(3,1,3,2)}$ *and* $C_{(2,2,2,2)}$. *The corners of the first disc are* $\delta\ (2, -4, 3)$ *and* $\delta\ (-1, -3, 4)$ *(and their mirror images). The corners of the latter disc are* $\beta\ (-4, 0, 4)$ *and its permutations. The intersection is a dodecagon with side-lengths:* $a_1 = 2\sqrt{3}$, $a_2 = 2\sqrt{3}$, $a_3 = 3$ *(see Figure 10 also). Its noncompactness ratio is* 12.866 *that is less than any polygon's can be obtained by four steps with a neighbourhood sequence.*

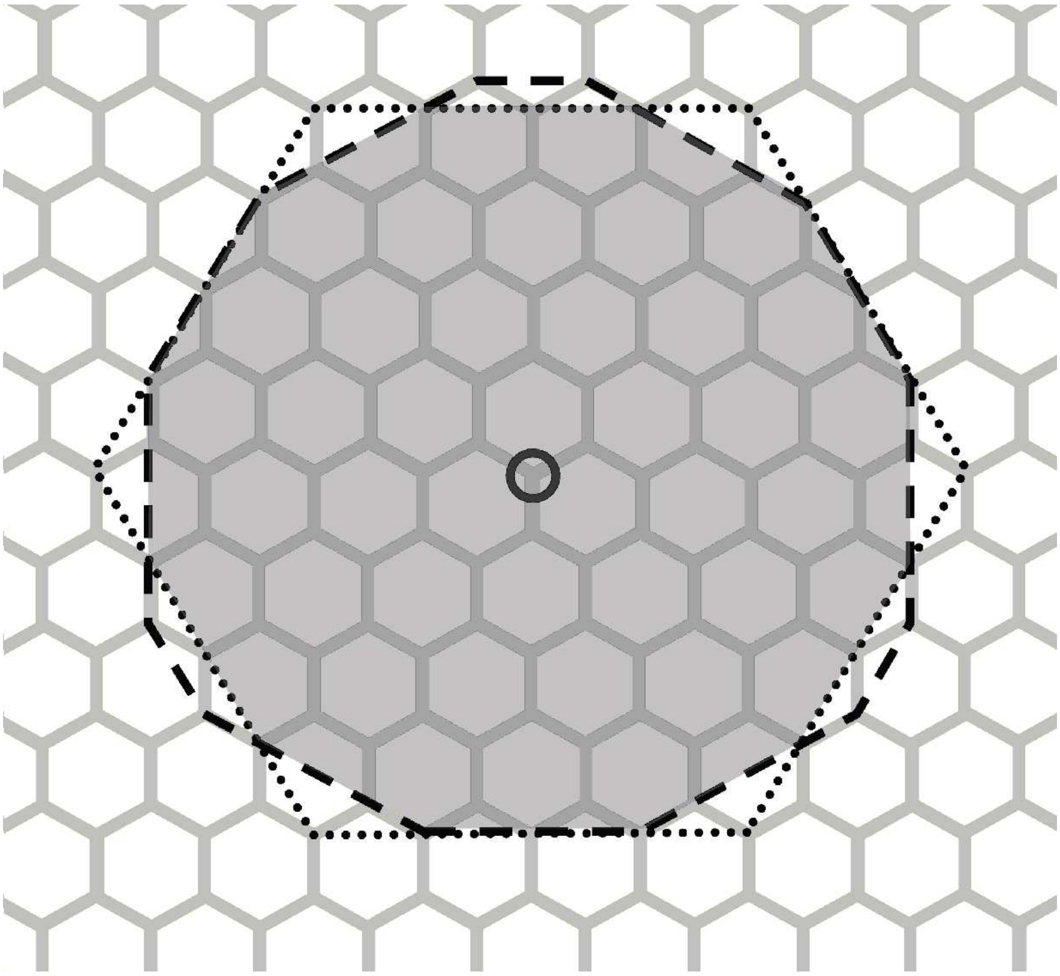


Fig. 10. The intersection of two non-comparable discs gives a better approximation

As the example shows in the hexagonal grid one can obtain better approximation using two neighbourhood sequences, than using only one. However it is not the case in the square grid (nor in $\mathbb{Z}^n$), since for any two digital discs with the same radius at least one of them contains the other one. The digital discs with a given radius form a well-ordered set in the square grid.

Since there are twelve directions (see Figure 7) of the possible edges of the possible polygons the approximation, even using more than one neighbourhood sequences, cannot go beyond the regular dodecagon (in the limit).

4 Conclusions

One can easily handle the hexagonal grid by the recalled coordinate system that uses 3 dependent coordinate values. Since the grid is planar, it can also be described by 2 independent values. For calculation of perimeter, area etc. of some digital objects it is nice to have a connection between the 3-valued description and the Cartesian 2-valued frame. We presented such a connection.

As we gave an alternative proof, the minimal equivalent neighbourhood sequence is a unique representation of the class of the neighbourhood sequences generating the same distance function. Similarly there is a unique basic part for every digital disk as we proved.

The initial parts of the neighbourhood sequences form equivalence classes that are described in a formal way by associative calculus.

Algorithms to compute the corners, types, etc. of the digital disks are presented. The computations are also presented in graphical form using finite automata/transducer approaches. Some properties of the approximation are analysed and a fast greedy algorithm is presented that provides a very good approximation (the best initial part can be obtained by at least length of 20). Example for approximation using two neighbourhood sequences is shown to be more effective than using only one sequence. A topic of further research is to analyse how good the approximation can be in this way. As we showed the approximation result is much more better in the used hexagonal grid (i.e., in the triangular tessellation) than in the square grid using distances based on neighbourhood sequences. By mixing neighbourhood sequences and weighted distances in the square grid [16] the approximation can be as good as in the presented approximation on the hexagonal grid. It is a task of future work to analyse how the value of the approximation would increase using both weights and neighbourhood sequences on the hexagonal grid.

Our results can be used in digital image processing and in the field of networks as well. It can be used in region growing procedures. In grid-structured networks the non-common properties can be useful. Some digital disks can be obtained by more than one radius or the non permutability of the elements of the neighbourhood sequence are exotic properties.

Acknowledgements

The private communication to Professor Attila Kuba about the idea of approximating the Euclidean distance by two neighbourhood sequences simultaneously is gratefully acknowledged. The work of the first author is supported by the TÁMOP 4.2.1./B-09/1/KONV-2010-0007 project. The project is implemented through the New Hungary Development Plan, co-financed by the European Social Fund and the European Regional Development Fund.

References

[1] V.E. Brimkov and R.P. Barneva, "Analytical Honeycomb Geometry for Raster and Volume Graphics," *The Computer Journal*, Oxford University Press, vol. 48 pp. 180–199, 2005.

[2] D. Coeurjolly, S. Miguet and L. Tougne, "2D and 3D visibility in discrete geometry: an application to discrete geodesic paths,", *Pattern Recognition Letters* vol. 25, pp. 561 – 570, 2004.

[3] P.P. Das, P.P. Chakrabarti and B.N. Chatterji, "Distance functions in digital geometry," *Information Sciences* vol. 42, pp. 113–136, 1987.

[4] P.P. Das and B.N. Chatterji, "Octagonal Distances for Digital Pictures," *Information Sciences* vol. 50, pp. 123–150, 1990.

[5] A. Hajdu, L. Hajdu and R. Tijdeman, "General neighbourhood sequences in $\mathbb{Z}^n$," *Discrete Applied Mathematics* vol. 155, pp. 2507–2522, 2007.

[6] R. Klette and A. Rosenfeld, *Digital geometry. Geometric methods for digital picture analysis*, Morgan Kaufmann Publishers, San Francisco, CA; Elsevier Science B.V., Amsterdam, 2004.

[7] B. Nagy, "Distance functions based on neighbourhood sequences," *Publicationes Mathematicae Debrecen*, vol. 63, pp. 483–493, 2003.

[8] B. Nagy, "Characterization of digital circles in triangular grid," *Pattern Recognition Letters* vol. 25, pp. 1231–1242, 2004.

[9] B. Nagy, "A symmetric coordinate system for the hexagonal networks", *Proceedings of Information Society 2004 – Theoretical Computer Science, ACM conference*, Ljubljana, Slovenia, (2004) 193–196.

[10] B. Nagy, "Geometry of neighborhood sequences in hexagonal grid", *Proceedings of Discrete Geometry for Computer Imagery (DGCI 2006)*, Szeged, Hungary, *Lecture Notes in Computer Science (LNCS)* vol. 4245, pp. 53–64, 2006.

[11] B. Nagy, “Isometric transformations of the dual of the hexagonal lattice”, *Proceedings of 6th International Symposium on Image and Signal Processing and Analysis (ISPA 2009)*, Salzburg, Austria, pp. 432–437.

[12] B. Nagy, “Nonmetrical Distances on the Hexagonal Grid Using Neighborhood Sequences,” *Pattern Recognition and Image Analysis* vol. 17, pp. 183–190, 2007.

[13] B. Nagy and R. Strand, “Neighbourhood Sequences in the Diamond Grid”, in: *Image Analysis – From Theory to Applications* (editors: Reneta P. Barneva, Valentin E. Brimkov), Research Publishing, Singapore, Chennai, pp. 187–195, 2008.

[14] A. Rosenfeld and J.L. Pfaltz, “Distance functions on digital pictures,” *Pattern Recognition* vol. 1, pp. 33–61, 1968.

[15] R. Strand and B. Nagy, “Distances Based on Neighbourhood Sequences in Non-Standard Three-Dimensional Grids,” *Discrete Applied Mathematics* vol. 155, pp. 548–557, 2007.

[16] R. Strand, “Weighted distances based on neighbourhood sequences,” *Pattern Recognition Letters* vol. 28, pp. 2029–2036, 2007.

[17] K. Voss, *Discrete Images, Objects, and Functions in* $\mathbb{Z}^n$, Algorithms and Combinatorics; 11, Berlin, Heidelberg, New York: Springer-Verlag, 1993.