RAIRO-Theor. Inf. Appl. 51 (2017) 169–180
https://doi.org/10.1051/ita/2017016

RAIRO - Theoretical Informatics and Applications
www.rairo-ita.org

TRANSLATION OF A DIGITAL LINE INTO ANOTHER ACCORDING TO VARIOUS DIGITIZATION PROCESSES[☆]

Jean-Pierre Borel[*]

Abstract. We introduce unusual methods for the digitization process of a line. A square pixel of the computer screen is blackened when the line crosses a special part of this pixel, called the *active pixel*. The shape of this active pixel is discussed, in the following sense: can we obtain the new Freeman Code of the line, using of a mechanical transformation of the initial Freeman Code, which is the classical Cutting Sequence? Our choice is to limit mechanical transformations to the existence of a given transducer, so that everytime we put in (a power of) the classical Freeman Code of a line, the output recovers the new Freeman Code. Then we prove that such a transducer exists if and only if the active pixel is a polygon with rational vertices and big enough. The same result can be proved if we introduce several grey levels in the representation of the line. Then we get some antialising effects.

Mathematics Subject Classification. 68R15, 68Q68

Received December 1, 2017. Accepted December 11, 2017.

1. Introduction: Drawing lines on a screen

Drawing a curve, a line, or more practically a finite segment on a computer screen amounts to blacken or enlight some pixels, for example those crossed by the curve in question. Each pixel $\mathfrak{P}$ can be viewed as a *small* square on the screen, translated copy of the unit square $[0,1] \times [0,1]$ (see Sect. 2.1 later), these pixels being organized following the lattice of integer points in the plane.

We look at this curve in a dynamic way, so we can describe the sequence of blackened pixel as follows:

– blacken the first pixel, which contains the starting point of the curve;
– move to the next pixel, by some integer translation, and blacken the new pixel;
– and so on till the end of the curve.

In this way, the curve is encoded by its starting point and the sequences of translations, which is a finite word on the infinite alphabet of integer vectors.

The most natural way is to darken all the pixels crossed by the line. It corresponds to the classical *Cutting Sequence*. For various reasons, another choice has been made, following a proposition due to Bresenham [6]. The main objective is to avoid three consecutive connected black pixels in the drawing. Thus, unlike Cutting

[☆] Research partially supported by Région Limousin.

Keywords and phrases: Digital lines, digitization processes, Freeman codes, cutting sequences.

Mathis-XLim, UMR 6172, Université de Limoges – CNRS, 123 avenue Albert Thomas, 87060 Limoges Cedex, France.

[*] Corresponding author: borel@unilim.fr

Sequence, only one pixel must be blackened in each column (when the slope is between 0 and 1). Many strategies are possible, see [8, 10, 13] for three of them, and algorithms, such as the surprising one given in [14].

2. The cutting sequences

2.1. Segments and half lines

We consider *lines* which are either segments or half-lines in the plane, starting from the *origin* $(0,0)$ (see Convs. A and B below) and ending (for segments) to another *integer* point. This is the first of the two reasonnable choices below.

Convention A. The integer points are on the node of the grid.

Convention B. The integer points are in the center of the squares of the grid.

We will limit ourselves to positive slopes, the negative ones being handled in a similar manner. As mentioned before, the drawing can be described by its *Freeman Code*, namely the word describing the sequence of successive blackened pixels. Each letter corresponds to an elementary geometric translation. When we use Convention B, the Freeman Code is *Freeman Code to avoid confusion. Convention A is the simplest one, due to the self-similarity properties (Sect. 6), whereas Convention B is more realistic for an effective drawing. This last convention is used for the classical Bresenham algorithm (Fig. 1).

2.2. Possible movements

Here we meet the classic question of the neighboring pixels, the 4-*neighborhood* case corresponding to the horizontal or vertical movements N (for *north*), S, E and W, as it is the case for The Freeman Code when it was introduced in [9] or 8-*neighborhood* case as the corresponding diagonal movements NE NW SE SW are also allowed, as in [16]. Our hypothesis of positive slopes implies that among these possibilities only E and N are useful, and NE for the diagonal movement, in these two classical methods. However, we shall use below the five letters a, b, c, d, e for the five movements E, N, NE, ENE, NNE respectively, as shown in Figure 4.

Similar approaches can be found in [7] or [17], and also in [4] for higher dimensional situations. In each case, the geometrical object is described by a coding word. It corresponds to the *discrete lines* in the sense of Réveilles, see [15].

2.3. Cutting sequences

Therefore this amounts to looking at the billiard words, or *Cutting Sequences*, associated with half-lines or segments. These words may be finite (segments) or infinite (half-lines). They have been widely studied, the reader will find overviews in [1] or [3] as examples. Some connections between these notions and the drawing of lines have been studied, see [2] and others authors.

The question of overlapping pixels is crucial. Many choices can be made to avoid overlapping or not.

Convention 1. $\mathfrak{P}$ is the closed unit square.

Convention 2. $\mathfrak{P}$ is the closed unit square, except the N and E sides.

Convention 3. $\widetilde{\mathfrak{P}}$ is the closed unit square except the two NW and SE vertices

Convention 1 avoids any overlapping, it implies the introduction of the three translations or letters for the Cutting Sequences: a for E, b for N and c for NE as said above. Only the lines of rational slope use the letter c, and we encode segments by adding a final letter c at the usual Cutting Sequence, due to Convention 2. Unlike Convention 2, Convention 3 keeps the diagonal symmetry.

In the sequel we choose Conventions A and 2. Then $\mathfrak{P} = [0,1[\times[0,1[$. The main theorem can be adapted to each convention.

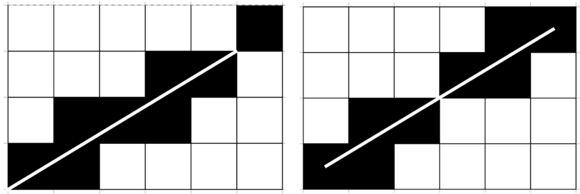

FIGURE 1. The digitized segments in Convention A (*left panel*) and Convention B (*right panel*) associated to the rational number $\frac{3}{5}$.

As an example, the usual Cutting Sequence of the segment joining the two points $(0, 0)$ and $(5, 3)$ is equal to *abaaba*, which is encoded here by *abaabac*. Its representation consists in eight pixels, the first one corresponding to the starting point and the following seven are associated with the seven movements indicated by the seven letters of the Cutting Sequence. If we use Convention B, the *Cutting Sequence is equal to *abacaba* and the digitized segment also has eight pixels. Convention A has some drawbacks, including the asymmetry of the drawing. On the other hand, we get simple formulas: if u is the word encoding the rational number $\frac{p}{q}$, u^n encodes $\frac{np}{nq}$ and u^∞ encodes of the half-line with slope $\frac{p}{q}$.

Notice that the Cutting Sequence encoding the rational number $\frac{p}{q}$ can be easily and rapidly computed, starting with the initial word $a^{q-1}b^{p-1}c$, (see *e.g.* [11] or [5]).

3. A new coding method for lines: The active pixel

3.1. The active pixel

Our choice is to introduce an *active pixel* $\mathfrak{A}$, which is the same (translated) part of any pixel $\mathfrak{P}$, and to use a *positive principle*, *i.e.*, the whole square pixel is blackened whenever the half-line (or the segment) crosses its active part. Hence less pixels are blackened, and the sequence of blackened pixels can be described in two ways:

- either by its Freeman Code, using a new alphabet which corresponds to all the possible movements from a blackened pixel to the next one;
- or by encoding the sequence of grey levels for each letter of the CS. It is a sequence denoted Grey-Level Code on the alphabet whose letters are the two grey levels (black and white), which indicates for any pixel in the Cutting Sequence the corresponding level of grey. We consider here only two levels 0 for white and 1 for black. Nevertheless our results also work with several levels of grey, if we use a new kind of active pixel with many parts, each one corresponding to a level of grey, except white. It produces results which can be compared to those of antialiasing methods.

More precisely, $\mathfrak{A}$ is the intersection of some variation $\mathfrak{P}$ of the square pixel, see Conventions 1-2-3 above, and of a closed connected set which is denoted also by $\mathfrak{A}$ for simplicity reason. In the sequel we choose Convention A and 2 as said before.

3.2. Diameters

Definition 3.1. The diameter seen under the angle θ of the active pixel is the length of its orthogonal projection on the line of polar angle $\theta + \frac{\pi}{2}$. It is denoted by δ_θ.

When $\mathfrak{A}$ is a disk, δ_θ is constant; when $\mathfrak{A} = \mathfrak{P}$ is the whole square $\delta_\theta = \sin\theta + \cos\theta$ with $0 \le \theta \le \frac{\pi}{2}$, then:

$$1 = \delta_0 = \delta_{\frac{\pi}{2}} \le \delta_\theta \le \delta_{\frac{\pi}{4}} = \sqrt{2}.$$

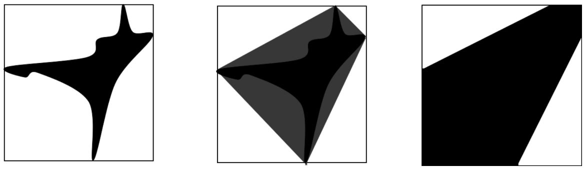

FIGURE 2. An example of active pixel. *Left panel* Active pixel, *Center panel* Convex hull and *Right panel* Positive convex hull.

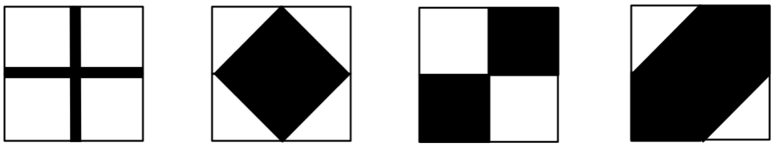

FIGURE 3. Four equivalent active pixels.

3.3. Equivalent active pixels

Remark that the Freeman Code is associated with the *positive convex hull* of $\mathfrak{A}$, as we only consider half-lines or segments with positive slope. The positive convex hull is the intersection of all closed half-planes containing $\mathfrak{A}$, whose border line has a positive slope. In practice, that is to add two rectangular corners SW and NE to the usual convex hull as shown below (Fig. 2).

Two active pixels are called *equivalent* whenever their positive convex hull is the same. As we look at segments or lines with positive slopes, the Freeman Code or *Freeman Code are the same for equivalent pixels. In the same way, equivalent pixels have the same diameters for $0 \leq \theta \leq \frac{\pi}{2}$.

Looking at the shapes in Figure 3, the second one corresponds to the well-known diamond (see [12]) and the fourth one is its positive convex hull. It corresponds to the classical Rosenfeld digitization [16], and to the Bresenham algorithm [6]. For this active pixel, $\delta_\theta = \max(\sin\theta, \cos\theta)$ and:

$$\frac{1}{\sqrt{2}} = \delta_{\frac{\pi}{4}} \leq \delta_\theta \leq \delta_0 = \delta_{\frac{\pi}{2}} = 1$$

for $0 \leq \theta \leq \frac{\pi}{2}$.

4. The finiteness of the alphabet

4.1. Characterization

We are interested in finding a quick and automatic way to compute the Freeman Code or Grey Level Code, starting from the Cutting Sequence. We will express this as a transducer. This implies that the alphabet for the output word of any lines is built on a fixed finite alphabet, which is obviously the case for the Grey Level Code, but not for the Freeman Code.

Proposition 4.1. *The Freeman Code is a word on a fixed finite alphabet if and only if the diameters δ_θ of the positive convex hull of the active pixel $\mathfrak{A}$ satisfy:*

- $\delta_{-\frac{\pi}{4}} = \sqrt{2}$, i.e., *the active pixel touches the four sides of the unit square;*
- $\delta_{\frac{\pi}{4}} \geq \frac{\sqrt{2}}{2}$, i.e., *this diameter is greater or equal to the half of the length of the diagonals of the unit square.*

Then the Freeman Code can be written on a five-letter alphabet $\{a, b, c, d, e\}$, corresponding to the elementary moves E, N, NE, ENE, NNE respectively (see Fig. 4).

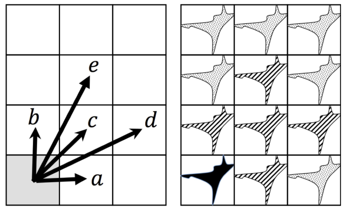

FIGURE 4. The five elementary movements corresponding to the five visible active pixels.

Notice that the first condition simply means that both points $(0,0)$ and $(1,1)$ belong to the positive convex hull, which is equivalent to say that $\mathfrak{A}$ reaches the four sides of the unit square.

Notice also that five-letter alphabet breaks the connexity and induces some discontinuities for the drawing, so it seems to have no practical interest. However, black pixels are not contiguous in general, if we look at drawing of lines with many levels of grey: such a phenomenon appears using antialiasing methods.

4.2. Some examples

For the whole square active pixel we only need the three letters a, b, c, and c is not used for any half-line with irrational slope. For a rational segment from $(0,0)$ to (Dq, Dp) with p and q coprime integers, the letter c appears D times, whenever the segment crosses an integer point except the origin (Conv. A), and it only appears $D-1$ times if pq is odd (Conv. B).

Likewise for the classical diamond active pixel, we only need the three letters a, b, c corresponding to the classical 8-*neighborhood*. The same is true when the pixel is the disk of diameter one. In these two cases, we use only letters a and c when the slope is less than 1 and only b and c otherwise.

If we consider the pixel given in Figure 4, we need the five letters:

Definition 4.2. A *visible active pixel* is an active pixel which can be reached starting from some point in the initial pixel in black, by a segment that does not cross another active pixel path before.

In the example given in Figure 4, the visible active pixels are striped, others simply greyed out.

4.3. Proof of Proposition 4.1

We have to look at the visible active pixels. We denote by O the origin, and for any point A, A_f is the image of A by the translation associated with the generic letter f. So the unit square at the origin can be written $OO_bO_cO_a$ as seen in Figure 5.

If the closed set $\mathfrak{A}$ does not intersect the segment O_bO_c then all the translated active pixels $\mathfrak{A} + \overrightarrow{(n,1)}$ are visible, and the same is true if $\mathfrak{A}$ does not intersect the segment OO_a. If the closed set $\mathfrak{A}$ does not intersect the segment O_aO_c then all the translated active pixels $\mathfrak{A} + \overrightarrow{(1,n)}$ are visible, and the same is true if $\mathfrak{A}$ does not intersect the segment OO_b. Thus the finiteness of the alphabet implies that $\mathfrak{A}$ intersects the four sides of the unit square. It means that the positive convex hull contains both O and O_c or equivalently that $\delta_{-\frac{\pi}{4}} = \sqrt{2}$.

In the same way, if $\delta_{\frac{\pi}{4}} < \frac{\sqrt{2}}{2}$, then all the translated active pixels $\mathfrak{A} + \overrightarrow{(n,n+1)}$ are visible. Thus the conditions given in Proposition 4.1 are necessary for finiteness.

Suppose now that these two conditions are satisfied. It implies that the active pixel at the origin contains the following points:

- O and O_c;
- some points A and B such that the distance between their orthogonal projections on the direction $\theta = \frac{\pi}{4}$ is equal to $\frac{1}{\sqrt{2}}$.

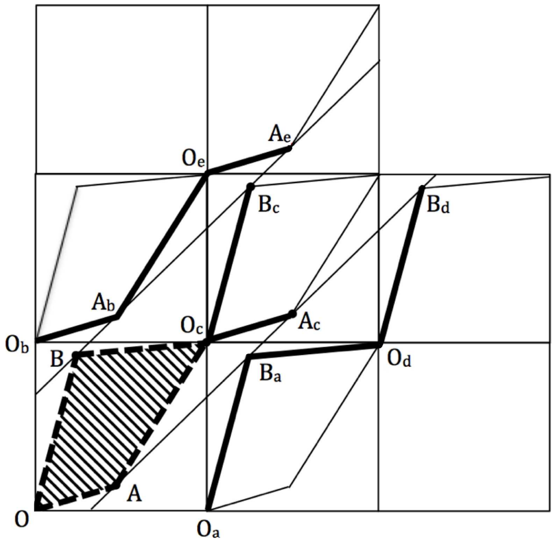

FIGURE 5. The possible moves from the initial pixel.

Thus the active pixel contains the area OBO_cA.

Using the translations corresponding to the five letters a, b, c, d, e we see that the three points A_b, B_c, A_e are on the same line whose slope is equal to 1, and the same holds for the three points B_a, A_c, B_d. It implies that any half-line starting anywhere into the unit square at the origin crosses one of the eight thick segments (see Fig. 5) composing $O_bA_bO_eA_e$, $B_cO_cA_c$ and $O_aB_aO_dB_d$. Then this half-line necessarily crosses one of the five active pixels $\mathfrak{A}+a$, $\mathfrak{A}+b$, $\mathfrak{A}+c$, $\mathfrak{A}+d$ or $\mathfrak{A}+e$. This proves Proposition 4.1. □

5. THE MAIN RESULT

We consider the two following hypothesis:

Hypothesis 5.1. [Hform] $\mathfrak{A}$ is a polygon with rational vertices.

Then we denote by D a common denominator of the coordinates of these vertices.

Hypothesis 5.2. [Hsize] $\delta_{-\frac{\pi}{4}} = \sqrt{2}$ and $\delta_{\frac{\pi}{4}} \geq \frac{\sqrt{2}}{2}$, *i.e.*, the alphabet is finite, according to Proposition 4.1.

Then we get a characterization of our active pixels for which there exists a transducer.

Theorem 5.3. *The Grey Level Code is the image of the Cutting Sequence by some transducer $\mathcal{T}$ if and only if [Hform] is satisfied.*

Theorem 5.4. *The Freeman Code is the image of the Cutting Sequence by some transducer $\mathfrak{T}$ if and only if [Hform] and [Hsize] are both satisfied*

In these two theorems $\mathcal{T} = \mathcal{T}_{\mathfrak{A}}$ is a letter-to-letter deterministic transducer which depends only on the chosen pixel $\mathfrak{A}$, and the input words in $\mathcal{T}$ are:

- for any half-line its Cutting Sequence u;
- for any segment the word u^D, where u is the Cutting Sequence of the segment.

Both proofs are very close, we shall see how the proof of Theorem 5.4 works. It will be done in Section 7 (sufficient condition) and Section 8 (necessary condition) (Figs. 9–11).

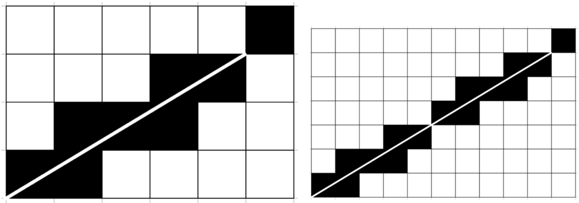

FIGURE 6. The digitized segments in Convention A for $\frac{3}{5}$, initial grid (*left*) and small grid with $D = 2$ (*right*).

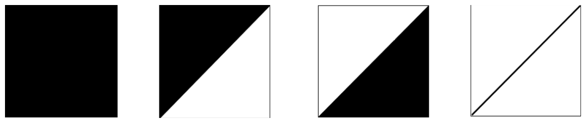

FIGURE 7. The four shapes: whole square, NW or SE triangle, diagonal.

6. The self-similarity properties for cutting sequences

6.1. Small grids

Let D be a given positive integer, and consider the D-grid, which is the grid of small squares of size $\frac{1}{D}$. The original size squares split in D^2 small squares, whose vertices are rational points with denominator D. Then any line has a new Freeman Code in the D-grid, in the 8-neighbourhood classical sense, which is called its D-Freeman Code or D-*Freeman Code depending on Convention A or B (Fig. 6).

6.2. Self-similarity

We denote by u and v respectively the Freeman Code or *Freeman Code of some segment or half-line starting from the origin.

Proposition 6.1. *(1) Using Convention A*
- *for any half-line, u is also its D-Freeman Code;*
- *for any segment, u^D is its D-Freeman Code.*

(2) Using Convention B
- *for any half-line, v is also its D-Freeman Code when D is odd, and u is its D-*Freeman Code when D is even;*
- *for any segment, v^D is its D-*Freeman Code when D is odd, and u^D is its D-*Freeman Code when D is even.*

These properties are easy consequences of the self-similarity of half-lines with ratio $\frac{1}{D}$, simply remark that a given segment can be cut in D sub-segments homothetic in the same ratio to the original one, and that centers of squares are also centers of small squares when D is odd, but become vertices when D is even. ☐

7. Proof of the main result: How to build the transducer?

The sufficient condition is shown by a hand-made construction of the transducer in each case. It is based on two results.

7.1. D-digital polygons

A part of the unit square is called *positive D-digital polygon* if it is a connected set and a union of parts of small (*i.e.*, in the small D-grid) squares, each one having one of the four possible forms given in Figure 7:

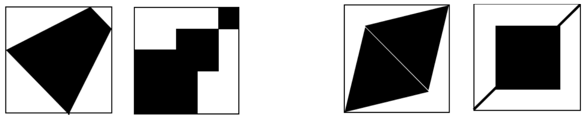

FIGURE 8. Two polygonal active pixels and their equivalent positive 5-digital polygons.

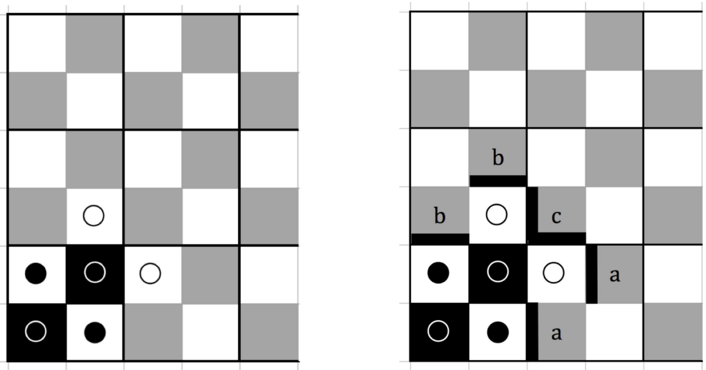

FIGURE 9. The six states and the output of a new letter.

These shapes put in the small grid are called the *elementary forms*.

Proposition 7.1. *Let $\mathfrak{A}$ be a polygonal active pixel whose vertices have rational coordinates, let D be a common denominator of their coordinates. Then $\mathfrak{A}$ is equivalent to a positive D-digital polygon.*

It can be easily seen that for any polygonal active pixel $\mathfrak{A}$ whose vertices have rational coordinates with denominator D, its positive convex hull $\tilde{\mathfrak{A}}$ is equivalent to the union of all elementary forms it contains. □

Proposition 7.2. *Let $\mathfrak{A}$ be a positive D-digital polygon. Then there exists a transducer $\mathcal{T}$ with the properties of Theorems 5.3 and 5.4.*

We prove the Proposition in the diamond case, the construction of the transducer $\mathcal{T}$ is the same in any case. The active pixels 3 and 4 in Figure 3 are positive 2-digital polygons, and the first one gives the simplest construction.

The states of the transducer $\mathcal{T}$ are those small squares in the 2-grid either in the original unit square, or visible from this square, *i.e.*,:

- are empty (or partially empty) for the corresponding active pixel;
- are reachable by some segment that starts from the original square and does not intersect another active pixel.

In the diamond case we get four states representing the four small squares in the unit square (the SW square is the initial state), and two others representing the other small squares that can be achieved. As soon as one crosses a thick black line in the right part of Figure 9, *i.e.*, one encounters a new active pixel, it generates a new letter in the Freeman Code.

The input state is the SW one, and arrows correspond to the elementary movements E, N and NE associated with the letters a, b and c respectively, as shown in Figure 10. The final state of an arrow is:

- either the state achieved by moving from the original one, when it exists. Then there is no output;
- or this achieved state retained by translation in the unit square at the origin. It means that a thick black line is crossed (right part of Fig. 9) and the corresponding letter is the output.

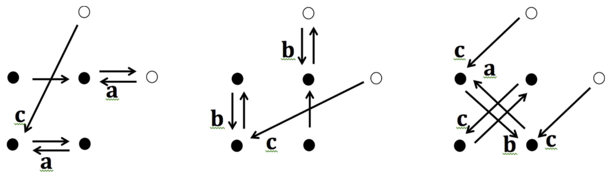

FIGURE 10. The transducer $\mathcal{T}_{diamond}$. *Left panel* Input a, *Center panel* Input b and *Right panel* Input c.

Then we get the following transducer (Fig. 10): letters at the end of the arrows are the outputs (if not there is no corresponding output) and whose initial state is at the SW corner.

This proves Proposition 7.2 in the diamond case. □

In the general case, when the small squares are of a different shape and are visible, they are represented by one state in the transducer (for empty triangle part) or two different states (for each of the two empty triangles in the case of the single diagonal), and the mechanism is similar. However the description of the transducer is a little bit more complicated.

So we get the sufficient part of Theorem 5.3: if the active pixel is equivalent to a polygon with rational vertices, we let D be a common denominator of all these coordinates. The active pixel is equivalent to a D-digital polygon, and we can built the transducer $\mathcal{T}$. □

The construction of the transducer in Theorem 5.3 can be deduced from the preceding case:

– if we look only at half-lines with irrational slopes, the Cutting Sequence uses only the two letters a and b, and it suffices to replace a, b, c, d, e in the output by $1, 1, 01, 001, 011$ respectively;
– when the letter c appears in the Cutting Sequence u, letters a, b, c, d, e in the output have to be replaced by $1, 1, 01, 001, 001$ as before, except for one case: the output letter c must be replaced by 1 when it arises from a letter c in the input whose rank among the c's in the input u or u^D is a multiple of D. So we have to count this rank $\mod D$ of the c's in the input, and this can be done by an adapted articulation of D copies of the former transducer, which gives also a transducer. This new transducer obviously works also in the former case.

8. Proof of the main theorem: The use of geometrical constraints

Consider now a given active pixel $\mathfrak{A}$ and its positive convex hull $\tilde{\mathfrak{A}}$. We need two steps in the proof.

Proposition 8.1. *Suppose that there exists some transducer with the following property: input the Cutting Sequence of any half-line with positive slope, and get its Freeman Code as output. Then there exists some constant N such that any half-line Δ starting form the origin O and tangent in some point T to any active pixel $\mathfrak{A}$ (or its positive convex hull $\tilde{\mathfrak{A}}$) contains an integer point B such that $\overrightarrow{OB} = \lambda\overrightarrow{OT}$, with $1 \le \lambda \le N$.*

This result contains two kinds of information.

– the first one is qualitative: all lines Δ have a rational slope, which means that they pass through infinitely many integer points, as they start at the origin. It can be easily proven that such a property is false if the active pixel is the disk with radius $\frac{1}{2}$. However it is possible to build active pixels $\tilde{\mathfrak{A}}$ with various shapes and satisfying this qualitative property, even without any angular vertex;
– the second one is quantitative: the first integer point met cannot be too far away. It constitutes the major constraint.

The proof of Proposition 8.1 is easy: the transducer turns infinite word in another infinite word, so any sufficiently long word has a non-empty picture. The length ratio between a finite word u and its image is therefore between two absolute positive constant values:

$$0 < C_1 \leq \frac{|\mathcal{T}(u)|}{|u|} \leq C_2 < +\infty.$$

The half-line $\varDelta$ starts form the origin O and is tangent in some point T to some active pixel $\tilde{\mathfrak{A}}$. Let θ be the polar angle of the tangent $\varDelta$, and take two half-lines $\varDelta_+$ et $\varDelta_-$ starting at the origin, and with polar angle $\theta \pm \varepsilon$. As one of these lines is crossing $\tilde{\mathfrak{A}}$ and the other is not, the two FC differ due to $\tilde{\mathfrak{A}}$, hence on a letter whose rank r can be compared to the length OA. Their Cutting Sequence cannot have the same prefix of length $\frac{r}{C_1}$. It implies the existence of an integer point in the angular sector between $\varDelta_+$ and $\varDelta_-$, whose distance at the origin is less than $\frac{r}{C_1}$. As ε decreases to 0 we get the existence of an integer point B, and $\lambda = O(C_1)$ is bounded.

Proposition 8.2. *Let $\tilde{\mathfrak{A}}$ be a positive convex active pixel satisfying the condition given in Proposition 8.1. Then $\tilde{\mathfrak{A}}$ is a convex polygon with rational vertices.*

Proof of Proposition 8.2. Let $\varDelta$ be the tangent half-line starting from the origin, to some given active pixel (we suppose that the active pixel is below, the argument is the same as above). Proposition 8.1 implies that $\varDelta$ has a rational slope. Let B be the first integer point on $\varDelta$. Let T be the nearest from the origin contact point of $\varDelta$ with this active pixel. We can suppose that T is between O and B, otherwise we change the pixel using translation of vector $-\overrightarrow{OB}$.

The lattice of integer points belongs to an infinite discrete set of parallel lines, parallel to OB. Let $C = C_0$ be a given point on the first parallel line below OB which contains integer points, with negative integer coordinates. So $\overrightarrow{OB}, \overrightarrow{OC}$ is a new basis of the integer points lattice.

We introduce the following new points:

- $C_k = C - k\overrightarrow{OB}$ for any $k \geq 1$;
- T_k the first contact point of the active pixel with its tangent half-line starting from C_k;
- M_k the intersection point of the two lines C_kT_k and OB.

So we get:

- the points M_k belong to the segment $[OT]$ and become closer to T as k increases. We denote by M their limit point;
- the slope of the half-line C_kT_k is a rational number according to Proposition 8.1. Hence starting from C_k it passes through a first integer point B_k;
- the point B_k belongs to one of the N first lines parallel to OB and above OB (Prop. 8.1). This is the key-point of the proof.

Thales Theorem implies that vectors $N!\overrightarrow{OM_k}$ have integer coordinates, so the points M_k belong to a discrete grid. But M_k tends to M. Hence we have a stationary sequence and $M_k = M$ for large k, $M_k = T_k$ for these k (the slope of the tangent line decreases as k increases), and $T = M$ is an angular vertex of the active pixel $\tilde{\mathfrak{A}}$, with rational coordinates whose denominators divide $N!$.

So all the points of contact of tangents are on a discrete grid, there is therefore a finite number of such points, and the active pixel $\tilde{\mathfrak{A}}$ is a polygon whose vertices have rational coordinates with the common denominator $N!$. □

We get immediately the necessary condition in Theorem 5.4 by combining Propositions 8.1 and 8.2. □

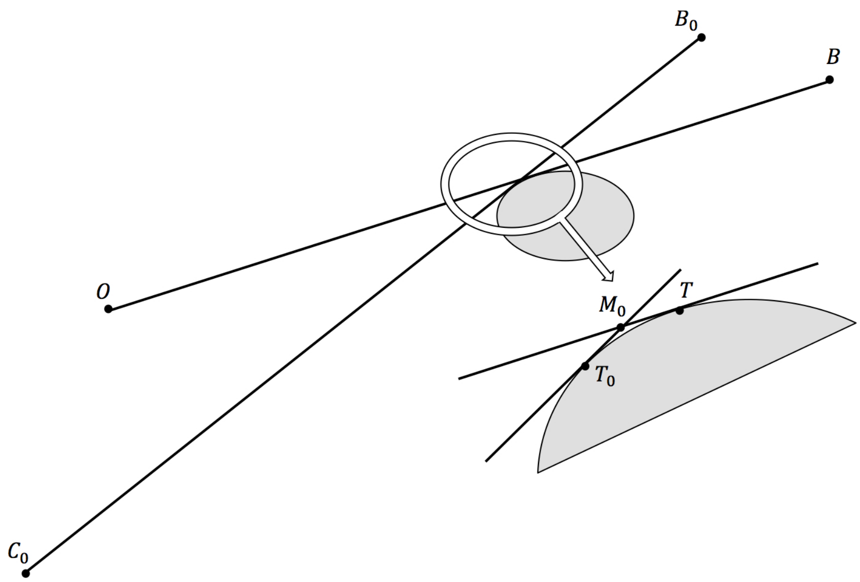

FIGURE 11. The two first tangent lines to an active pixel, with magnification of the intersection.

9. Variations due to conventions

The choice between Conventions 1-2-3 (see Sect. 2.3) does not affect the proof, as it changes the active pixel only on the frontier of the unit square. It may only modify the effective construction of the transducer $\mathcal{T}$.

On the other hand Conventions A-B have an important effect on the main theorems. This is due to the different forms of the self-similarity properties detailed in Section 6. We give here a version of Theorem 5.4 if we choose Convention B, *i.e.*, look at the *Freeman Codes and *Cutting Sequence.

Theorem 9.1. *The *Freeman Code is the image of the *Cutting Sequence by some transducer $\mathcal{T}$ if and only if [Hform] with an odd D and [Hsize] are both satisfied.*

The construction of the transducer $\mathcal{T}$ is similar, using Part.2 of Proposition 6.1 as D is odd. The main change concerns the geometric argument used in Section 8 to prove that the active pixel satisfies [Hform], *i.e.*, is a polygon with rational vertices. In Figure 11, the point O must be translated to the center I of the initial square, whose coordinates are $(\frac{1}{2}, \frac{1}{2})$.

We multiply the coordinates by 2, to get integer points for simplicity reason. Then the new points O and C_k have odd coordinates, and the points B and B_k have even coordinates, as they were integer points before multiplication. Then Thales Theorem implies that the intersection points M_k have rational coordinates with odd (and bounded) denominators, and the same holds for their limit $M = T$ (see Sect. 8). But these points T are exactly the extremal points of the polygon $\mathfrak{A}$, so the property [Hform] is satisfied with an odd number D. □

It implies that the Bresenham Code, *i.e.*, the Freeman Code associated with the diamond pixel and with Convention B, cannot be deduced automatically *via* some transducer from its Cutting Sequence: it corresponds to a polygonal pixel whose vertices have rational coordinates, but with $D = 2$.

The same kind of proof goes for Convention B for the Bresenham Code and Convention A for the Cutting Sequence. Such a choice does not make sense.

10. Some remarks and possible developments

The techniques described here allow to work in other contexts although technical complications appear. This is the case of:

(1) several levels of grey, corresponding to pitted active pixels

$$\mathfrak{A}_n \subset \mathfrak{A}_{n-1} \subset \cdots \subset \mathfrak{A}_1$$

and the grey level of the square pixel is the ratio $\frac{k}{n}$ where k is the maximal index such that $\mathfrak{A}_i$ is crossed by the line we consider. It gives an antialiasing effect for the drawing;

(2) a *negative principle*: we take a *passive pixel* and the square pixel in the Cutting Sequence is blackened when the line does not cross the passive pixel;

(3) active pixels larger than the square pixel $\mathfrak{P}$.

The combination of (1) and (3) provides the most fertile generalizations.

The same work can also be done for dimension greater than 2. But there are infinitely many tangents issued from the origin to a given active pixel, and the qualitative part of Proposition 8.1 *the slope of any tangent line is a rational number* becomes hard to fulfil. Therefore, even if the active pixel is a polyhedron with rational vertices, such a transducer does not exist in general: the active pixel must be equivalent to a positive digital polyhedron, namely the analog in this dimension of positive digital polygons, and Proposition 7.1 becomes false in this context.

References

[1] J.-P. Allouche, J. Shallit, Automatic Sequences: Theory and Applications. Cambridge University Press, Cambridge (2003).
[2] J. Berstel, Tracé de droites, fractions continues et morphismes réels, in M. Lothaire, "Mots", Mélanges offerts á M.P. Schutzenberger, Hermès (1990).
[3] J. Berstel and P. Seébold, Sturmian words, in Algebraic Combinatorics on Words, edited by M. Lothaire. Cambridge University Press (2002).
[4] J.-P. Borel and C. Reutenauer, Palindromic factors of billiard words. *Theor. Comput. Sci.* **340–342** (2005) 334–348.
[5] J.-P. Borel, How to build Billiard Words using Decimations. *RAIRO: ITA* **44** (2010) 59–77.
[6] J.E. Bresenham, Algorithm for computer control of a digital plotter. *IBM Syst. J.* **4** (1965) 25–30.
[7] D. Crisp, W. Moran, A. Pollington and P. Shive, Substitution invariant cutting sequences. *J. Théorie des Nombres Bordeaux* **5** (1993) 123–137.
[8] J.D. Foley and A. Van Dam, Fundamentals of Interactive Computer Graphics. Addison-Wesley, Cambridge (1982).
[9] H. Freeman, On the encoding of arbitrary geometric configuration. *IRE Trans. Electron. Comput.* **10** (1961) 260–268.
[10] G. Hegron, Synthèse d'images : algorithmes élémentaires. Dunod, Paris (1985).
[11] J. Justin and G. Pirillo, Decimations and Sturmian words. *Theor. Inform. Appl.* **31** (1997) 271–290.
[12] J. Koplowitz, On the performance of chain codes for quantization of line drawings. *IEEE Trans Pattern Anal. Machine Intell.* **PAMI-3** (1981) 357–393.
[13] W.M. Newman and R.F. Sproull, Principles of Interactive Computer Graphics. McGraw-Hill (1985).
[14] M.L.V. Pitteway and R.A. Earnshax, in Euclid's Algorithm and Line Drawing, Fundamental Algorithms in Computer Graphics. Springer Verlag (1985) 101–105.
[15] J.-P. Reveilles, *Géométrie discréte, calcul en nombres entiers et algorithmique*. Thesis, Univ. Louis Pasteur – Strasbourg, France (1991).
[16] A. Rosenfeld, Digital straight line segments. *IEEE Trans. Comput.* **32** (1974) 1264–1269.
[17] C. Series, The geometry of Markoff numbers. *Math. Intell.* **7** (1985) 20–29.