

Digital Straightness

Azriel Rosenfeld¹

*Center for Automation Research, University of Maryland
College Park, Maryland, USA*

Reinhard Klette²

*CITR, University of Auckland, Tamaki Campus
Auckland, New Zealand*

Abstract

A digital arc is called ‘straight’ if it is the digitization of a straight line segment. Since the concept of digital straightness was introduced in the mid-1970’s, dozens of papers on the subject have appeared; many characterizations of digital straight lines have been formulated, and many algorithms for determining whether a digital arc is straight have been defined. This paper reviews the literature on digital straightness and discusses its relationship to other concepts of geometry, the theory of words, and number theory.

1 Introduction

The computer representation of lines and curves has been an active subject of research for nearly half a century [3,4,5,14]. Related work even earlier on the theory of words [1] (specifically, on mechanical or Sturmian words) remained unnoticed in the pattern recognition community. This paper reviews the subject of digital straightness with respect to its interactions with other disciplines (many of which are listed in [44]), as well as its role within the pattern recognition literature itself.

We consider the digitization of rays

$$\gamma_{\alpha,\beta} = \{(x, \alpha x + \beta) : 0 \leq x < +\infty\}$$

in the set $\mathbb{N}^2 = \{(i, j) : i, j \in \mathbb{N}\}$ of all *grid points* with non-negative integer coordinates in the plane. As a simplification we assume that $0 \leq \alpha \leq 1$; this

¹ Email: ar@cfar.umd.edu

² Email: r.klette@auckland.ac.nz

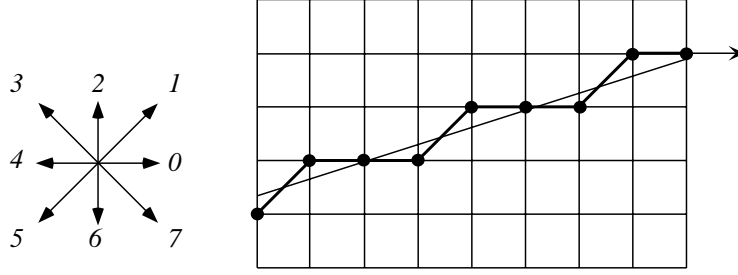


Fig. 1. Segment of a digital ray, defined by grid-intersection digitization (as calculated by the Bresenham algorithm [6]).

is possible due to the symmetry of the grid. Such a ray generates a sequence of intersection points $\rho_0, \rho_1, \rho_2, \dots$ of $\gamma_{\alpha,\beta}$ with the vertical *grid lines* at $n \geq 0$. Let $(n, I_n) \in \mathbb{Z}^2$ be the grid point nearest to ρ_n . (If there are two nearest grid points, we take the upper one.) Formally,

$$I_{\alpha,\beta} = \{(n, I_n) : n \geq 0 \wedge I_n = \lfloor \alpha n + \beta + 0.5 \rfloor\},$$

and $i_{\alpha,\beta} = i_{\alpha,\beta}(0)i_{\alpha,\beta}(1)i_{\alpha,\beta}(2)\dots$ is a *digital ray* with *slope* α and *intercept* β , where differences between successive I_n 's define *chain codes*:

$$i_{\alpha,\beta}(n) = I_{n+1} - I_n = \begin{cases} 0, & \text{if } I_n = I_{n+1} \\ 1, & \text{if } I_n = I_{n+1} - 1 \end{cases}, \text{ for } n \geq 0.$$

Code 0 is interpreted as a horizontal grid increment and 1 specifies a diagonal increment in the grid \mathbb{N}^2 ; see Fig. 1.

A finite or infinite 8-arc³ is *irreducible* iff its set of grid points does not remain 8-connected after removing a point which is not an end point.

Theorem 1.1 (Rosenfeld 1974) *A digital ray is an irreducible 8-arc.*

Proof. A ray $\gamma_{\alpha,\beta}$, with $0 \leq \alpha \leq 1$, intersects grid lines $x = n$, once each. Its intercepts with any two successive lines $x = n$ and $x = n+1$ differ vertically by α ; hence the digitizations of these successive intercepts also differ vertically by ≤ 1 . Thus the successive grid points of the digital ray are 8-neighbors. Removing the grid point at any $x = n$ would leave grid points at $x = n-1$ and $x = n+1$ disconnected with respect to the 8-neighborhood. \square

The digital ray $i_{\alpha,\beta}$ is generated by ray $\gamma_{\alpha,\beta}$. If $\beta - \beta'$ is an integer then $i_{\alpha,\beta} = i_{\alpha,\beta'}$. Thus we may assume that intercepts are limited to $0 \leq \beta \leq 1$ without loss of generality. Evidently $i_{0,\beta} = 000\dots$ and $i_{1,\beta} = 111\dots$

³ If $p = (x, y)$ is a grid point, an *8-neighbor* of p is any grid point $q = (i, j)$ with $d_\infty(p, q) = \max\{|x - i|, |y - j|\} = 1$. An *8-arc* is a finite or infinite sequence of grid points such that any point is an 8-neighbor of its predecessor in the sequence.

Theorem 1.2 (Bruckstein 1991) *For irrational α , $I_{\alpha,\beta}$ uniquely determines both α and β . For rational α , $I_{\alpha,\beta}$ uniquely determines α , and β is determined up to an interval.*

Proof. For arbitrary $\alpha, \alpha', \beta, \beta'$, $I_{\alpha,\beta} = I_{\alpha',\beta'}$ implies $\alpha = \alpha'$ since otherwise the vertical distance between $\alpha x + \beta$ and $\alpha' x + \beta'$ would become unbounded as x goes to infinity, i.e. the I_n -values would differ starting at some large enough n .

If α is irrational then the set of all vertical intercepts of $\alpha x + \beta$ modulo 1, $x \geq 0$, is dense in $[0, 1]$. Therefore, for every $\varepsilon > 0$ there exist n_0 and m_0 such that

$$\begin{aligned} \alpha n_0 + \beta - \lfloor \alpha n_0 + \beta \rfloor &< \varepsilon, \\ \alpha m_0 + \beta - \lfloor \alpha m_0 + \beta \rfloor &> 1 - \varepsilon, \end{aligned}$$

and changing β by ε would result in a change in $I_{\alpha,\beta}$. Therefore, for irrational α , $I_{\alpha,\beta}$ also uniquely determines β .

If α is rational then the set of all vertical intercepts of $\alpha x + \beta$ modulo 1, $x \geq 0$, is finite, i.e. β is determined only up to an interval, and the length of the interval depends upon α . \square

This theorem states that a digital ray $i_{\alpha,\beta}$ always determines α uniquely. A digital ray is *rational* if it has a rational slope, and it is *irrational* if its slope is irrational. For a specification of the intercepts β see the discussion of [29] in Section 5.

We use the alphabet $A = \{0, 1, \dots, 7\}$ (or a subset of it) and a geometric interpretation of its elements as indicated in Fig. 1. Digital rays are (right) infinite words over 0, 1. We recall a few basic definitions from the theory of words [35, 59]. A (finite) *word* over A is a finite sequence of elements of A . The *length* $|u|$ of the word $u = a_1 a_2 \dots a_n$, $a_i \in A$, is the number n of *letters* a_i in u . The *empty word* ε has length zero. The set of all words defined on alphabet A is denoted by A^* . A word v is a *factor* of a word u iff there exist words v_1, v_2 such that $u = v_1 v v_2$. A word v is a *subword* of a word u iff $v = a_1 a_2 \dots a_n$, $a_i \in A$, and there exist $v_0, v_1, \dots, v_n \in A^*$ such that $u = v_0 a_1 v_1 a_2 \dots a_n v_n$.

Let $X \subset A^*$. The set of all *infinite words* $w = u_0 u_1 u_2 \dots$, with $u_i \in X - \{\varepsilon\}$, is denoted by X^ω . For $v \in A^*$ and $w \in A^\omega$ the concatenation vw is well defined, v is a *prefix* of vw and w is a *suffix* of vw . A finite word v is a *factor* of an infinite word w if $w = uvw_1$.

An integer $k \geq 1$ is a *period* of a word $u = a_1 a_2 \dots a_n$, $a_i \in A$, if $a_i = a_{i+k}$ for $i = 1, \dots, n - k$. The smallest period of u is called *the* period of u . An infinite word $w \in A^\omega$ is *periodic* if it is of the form $w = v^\omega$, for some non-empty word $v \in A^*$. A word $w \in A^\omega$ is *eventually periodic* if it is of the form $w = uv^\omega$, for $u \in A^*$ and a non-empty word $v \in A^*$. A word $w \in A^\omega$ is *aperiodic* if it is not eventually periodic.

It has been known since [13] that grid-intersection digitization of rays $\gamma_{\alpha,\beta}$ produces periodic digital rays if the slope α is rational, and aperiodic finite sequences if it is irrational:

“When a slope is an irreducible rational fraction, the string is periodic, and the length of a period is the denominator of the fraction. For example, one period of the string for the straight line with slope $2/5$ can be expressed as 01010, 00101, 10010, 01001, or 10100. Which of these periods is chosen is not important, because the bounds of the period can be placed anywhere.”

Theorem 1.3 (Brons 1974) *Rational digital rays are periodic and irrational digital rays are aperiodic.*

If v is the shortest word such that $w = v^\omega$ then v is called the *basic segment* of w and $|v|$ is *the* period of w . [13] specifies an algorithm for calculating the basic segment of any rational digital ray, for $\beta = 0$. For example, the slope $2/5$ does not specify a basic segment uniquely, but a rational slope α together with an intercept β do. [26] specifies an algorithm for calculating the basic segment of an arbitrary rational digital ray, using α and β as inputs.

This paper is structured as follows: Section 2 reviews alternative definitions of digital rays or digital straight lines, and specifies digitized lines by distances between tangential lines. Self-similarity studies in the context of pattern recognition are reviewed in Section 3, and in the context of the theory of words in Section 4. Number-theoretical results are briefly listed in Section 5. A review of recognition algorithms for digital straight segments is presented in Section 6. Section 7 concludes the paper.

2 Alternative Definitions and Tangential Lines

An alternative way of defining a digital ray is as the boundary of either the upper or lower dichotomy of \mathbb{N}^2 defined by a ray separation. Formally, let

$$\begin{aligned} U_{\alpha,\beta} &= \{(n, U_n) : n \geq 0 \wedge U_n = \lceil \alpha n + \beta \rceil\} \quad \text{and} \\ L_{\alpha,\beta} &= \{(n, L_n) : n \geq 0 \wedge L_n = \lfloor \alpha n + \beta \rfloor\} \end{aligned}$$

and $u_{\alpha,\beta}(n) = U_{n+1} - U_n$ and $l_{\alpha,\beta}(n) = L_{n+1} - L_n$ for $n \geq 0$. The chain code sequence $u_{\alpha,\beta}$ is the *upper digital ray*, and the chain code sequence $l_{\alpha,\beta}$ is the *lower digital ray* generated by $\gamma_{\alpha,\beta}$. The slope specifies rational and irrational lower or upper digital straight lines, which are always irreducible 8-arcs in \mathbb{N}^2 .

$L_{\alpha,\beta} = I_{\alpha,\beta-0.5}$, i.e. any lower digital ray is also a digital ray, and vice versa. If $\alpha n + \beta$ is not an integer then $U_n = L_n + 1$. Otherwise, $U_n = L_n$; the digital rays $u_{\alpha,\beta}$ and $l_{\alpha,\beta}$ will differ in this case, and $\gamma_{\alpha,\beta}$ has an *integral point* at n . If $\gamma_{\alpha,\beta}$ has no integral points, then $u_{\alpha,\beta} = i_{\alpha,\beta-0.5} = l_{\alpha,\beta}$. If $\gamma_{\alpha,\beta}$ has integral points and α is rational then there exists β' such that $U_{\alpha,\beta} = I_{\alpha,\beta'}$. Finally, if $\gamma_{\alpha,\beta}$ has integral points and α is irrational, then $U_{\alpha,\beta}$ and $L_{\alpha,\beta}$ differ by subsequences of length two only. For practical purposes, the classes of digital rays, upper digital rays, and lower digital rays coincide.

Arithmetic geometry, as established in [49], specifies digital hyperplanes by double Diophantine inequalities, allowing a uniform approach to studying n -dimensional digital hyperplanes. In the two-dimensional case, let a, b be

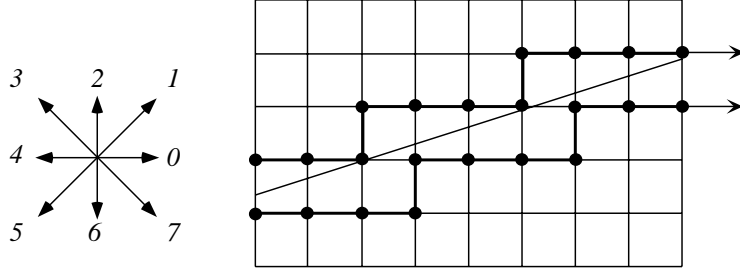


Fig. 2. Segments of lower and upper digital 4-rays, which follow boundaries of the upper and lower dichotomies, which are linearly separated by a ray.

relatively prime integers, i.e. $\gcd(a, b) = 1$, let c, d be integers, and let

$$D_{a,b,c,d} = \{(i, j) \in \mathbb{Z}^2 : c \leq ai - bj < c + d\}.$$

The set $D_{a,b,c,d}$ is called a *digital bar* with slope a/b , lower bound c and *arithmetical width* d .

Theorem 2.1 (Reveillés 1991) *Any set of grid points $D_{a,b,c,\max\{|a|,|b|\}}$ coincides with a set of grid points assigned to a digital straight line, and conversely, for any digital straight line there are parameters a, b, c such that the set of grid points assigned to this digital straight line coincides with $D_{a,b,c,\max\{|a|,|b|\}}$.*

This theorem also means that $d = \max\{|a|, |b|\}$ specifies an irreducible 8-arc.⁴ Due to our general assumption of considering only lines with slope $0 \leq a/b \leq 1$, we have $0 \leq a \leq b$ and $d = b$. All grid points in $D_{a,b,c,b}$ are between two lines $ax - by = c$ and $ax - by = c + b - 1$, i.e. $y = \alpha x + \beta$ and $y = \alpha x + \beta - (1 - \frac{1}{b})$, for $\alpha = a/b$ and $\beta = -c/b$. These two lines define a *pair of tangential lines* with respect to the given set $D_{a,b,c,b}$ of grid points (this proves Corollary 2.3).

$I_{\alpha,\beta}$, $U_{\alpha,\beta}$ and $L_{\alpha,\beta}$ can also be used to introduce *digital 4-rays* [44], which are 4-arcs:

$$i_{\alpha,\beta}^{\circ}(n) = \begin{cases} 0, & \text{if } I_n = I_{n+1} \\ 02, & \text{if } I_n = I_{n+1} - 1 \end{cases}$$

where a horizontal increment in \mathbb{N}^2 is encoded by 0, and 2 specifies a vertical increment. Analogously we define *upper digital 4-rays* $u_{\alpha,\beta}^{\circ}(n)$ and *lower digital 4-rays* $l_{\alpha,\beta}^{\circ}(n)$, all for $n \geq 0$. See Fig. 2 for an illustration of upper and lower digital 4-rays. We still have $i_{0,\beta}^{\circ} = 000\dots$, but $i_{1,\beta}^{\circ} = 020202\dots$. Again, the classes of digital 4-rays, upper digital 4-rays, and lower digital 4-rays coincide for practical purposes.

Digital 4-rays are actually just images under a morphism defined on digital rays. A *morphism* or *substitution* $\varphi : A^* \rightarrow B^*$ is a function with $\varphi(xy) = \varphi(x)\varphi(y)$, for all $x, y \in A^*$. A morphism is uniquely determined by its values for all letters in the alphabet. A morphism is *nonerasing* if a letter is always mapped into a nonempty word. A nonerasing morphism $\varphi : A^* \rightarrow B^*$ defines a

⁴ Digital straight lines are called *naive lines* in arithmetic geometry.

function, also called a morphism, from A^ω to B^ω by $\varphi(a(0)a(1)\dots a(n)\dots) = \varphi(a(0))\varphi(a(1))\dots\varphi(a(n))\dots$. Digital 4-rays may also be defined by specifying a morphism on A^\star

$$\varphi : \begin{array}{l} 0 \mapsto 0 \\ 1 \mapsto 02 \end{array}$$

mapping digital rays into digital 4-rays.⁵ The theory of words studies morphisms on infinite words.

Definition 2.2 A *digital straight segment* (DSS for short) is a geometrically interpreted non-empty factor of a digital ray, and a *digital 4-straight segment* (4-DSS for short) is a non-empty factor of a digital 4-ray, using the appropriate geometric interpretation of its chain code.

A digital straight segment u connects two grid points $p = (m_p, n_p), q = (m_q, n_q) \in \mathbb{N}^2$, $m_p < m_q$, iff the geometric interpretation of $u = u(1)\dots u(m_q - m_p + 1)$ specifies a sequence of horizontal and diagonal steps which leads from p to q . For an 8-arc $u = u(1)u(2)\dots u(n)$ of length n let $G(u) = \{p_0, p_1, \dots, p_{n-1}\}$ be the *assigned set of grid points* such that $p_0 = (0, 0)$ and u connects p_0 with p_{n-1} via a sequence of horizontal and diagonal steps which passes through p_1, \dots, p_{n-2} . An early algorithm for generating a digital straight segment connecting two arbitrary grid points p and q was published in [11]. Theorem 2.1 implies:

Corollary 2.3 A word $u \in \{0, 1\}^\star$ is a DSS iff the set $G(u)$ of assigned grid points lies on or between two parallel lines having a distance less than 1, measured in the y -axis direction.⁶

The geometric characterization of digital 4-straight segments has been discussed in [42], based on results on the ‘nearest support below or above’ of a digital straight segment in [30]. There are two possible diagonals in grid squares; see Fig. 3. The *main diagonal* for a pair of parallel lines is the one which maximizes the dot product with the normal to the lines. The *main diagonal distance* between two parallel lines is measured in the direction of the main diagonal. The following theorem specifies an unproven statement in [42]:

Theorem 2.4 A finite 4-arc $u \in \{0, 2\}^\star$ is a digital 4-straight segment iff its assigned set of grid points $G(u)$ is between or on a pair of parallel lines having a main diagonal distance of less than $\sqrt{2}$.

Proof. Let μ be a mapping from $\{0, 1, 2\}^\star$ into $\{0, 1, 2\}^\star$ defined by replacing any factor 02 by 1. Following the definition of digital 4-rays, a word

⁵ As another example, rule **X** studied in [44] for digital 4-rays is actually a morphism

$$\varphi_{\mathbf{X}} : \begin{array}{l} 0 \mapsto 2 \\ 2 \mapsto 0 \end{array}$$

⁶ This is already shown in [15] using the chord property of Theorem 3.2; see also [30, 36].

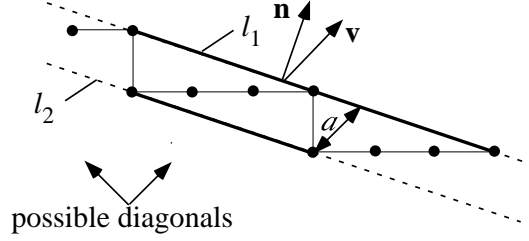


Fig. 3. Two parallel lines l_1 and l_2 contain a 4-arc between them; $a < \sqrt{2}$ is the main diagonal distance between the lines. Vector \mathbf{n} is the normal to l_1 , and \mathbf{v} is the unit vector along the main diagonal.

$u \in \{0, 1, 2\}^*$ is a 4-DSS iff $\mu(u)$ is a DSS. We also use Corollary 2.3 which characterizes DSSs by distance 1 (in the y -direction) between a pair of tangential lines. The main diagonal for 4-arcs $u \in \{0, 2\}^*$ makes angle 135° with the x -axis.

Assume a pair γ_1, γ_2 of parallel lines having main diagonal distance less than $\sqrt{2}$. Consider a finite 4-arc $u \in \{0, 2\}^*$ with an assigned set of grid points $G(u)$ between or on this pair of parallel lines. If the slope α of these lines is either 0 or 1, then the 4-arc is either $u = 0^n$ or $u = (02)^n$, i.e. a 4-straight segment. Now assume $0 < \alpha < 1$. The word $\mu(u)$ allows the lower line (say γ_2) to move into line ζ , closer to γ_1 by a parallel shift, such that γ_1, ζ are a pair of tangential lines for $G(\mu(u))$, and the distance between these two lines in the y -direction is less than 1, i.e. $\mu(u)$ is a DSS and u is a 4-DSS.

Now assume a 4-arc $u \in \{0, 2\}^*$ such that the minimum diagonal distance in direction 135° between a pair of parallel lines is greater than or equal to $\sqrt{2}$, i.e. u contains at least one subword 22. Then $\mu(u)$ is not a DSS, and u not a 4-DSS. \square

The two parallel lines at minimum diagonal distance specify a *pair of tangential lines* with respect to a given digital 4-straight segment. Note that a finite 4-arc is also a finite 8-arc, but being between a pair of parallel lines having a main diagonal distance of less than $\sqrt{2}$ does not mean that this 4-arc is also a digital straight segment because it is not an irreducible 8-arc. A pair of tangential lines with respect to a set $D_{a,b,c,b}$ of grid points has intercepts which differ by $0 < 1 - \frac{1}{b} < 1$, i.e. this pair of parallel lines also has a main diagonal distance of less than $\sqrt{2}$.

Finally we briefly review another option for specifying digital straightness. Besides sequences of grid points we may also consider sequences of grid squares for defining digitized rays or straight lines [17]. Assume a uniform mosaic in the Euclidean plane defined by square isothetic closed cells C having grid points $p \in \mathbb{Z}^2$ as their center points and edges of length 1. A family of cells is *edge connected* iff the set of center points of these cells is 4-connected.⁷

⁷ To be precise, these cells should be called *2-cells*, because their vertices are called *0-cells* and their edges are *1-cells* in the theory of cellular complexes [58].

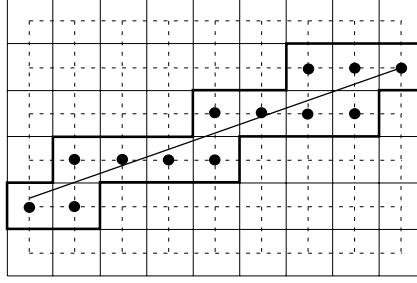


Fig. 4. A cellular straight segment.

Definition 2.5 A *cellular straight line* is a family F of cells C defined by a straight line γ : every cell in F has a non-empty intersection with γ , and γ is contained in the union $\bigcup F$ of all cells contained in F . A *cellular straight segment* is defined by a straight line segment γ in the same way.

See Fig. 4 for an illustration of cellular straightness. The distance between a pair of parallel lines is measured in the direction of the normal to the lines. Let S be a bounded set in the plane and θ a direction with $0 \leq \theta < 2\pi$. The width $w_\theta(S)$ is defined to be the minimum distance between a pair of parallel lines such that S is completely between them, and θ is the direction of the normal to them. Let $R_{2 \times 2}$ be a square formed by four cells.

Theorem 2.6 (Fam and Sklansky 1977) *An edge-connected family F of cells is cellularly straight iff there exists a direction θ with $w_\theta(\bigcup F) \leq w_\theta(R_{2 \times 2})$.*

The width $w_\theta(\bigcup F)$ as specified in this theorem is related to a *pair of tangential lines* with respect to the given family of cells. Altogether we have stated three theorems specifying pairs of tangential lines for digital straight segments, digital 4-segments, and cellular straight segments.

3 Self-Similarity Studies in Pattern Recognition

Self-similarity properties of digital rays or digital straight segments have been studied in pattern recognition with a major focus on geometric characterizations and efficient algorithms. *Chain code sequences* are finite or (right or two-sided) infinite words over $A = \{0, 1, \dots, 7\}$, and the interpretations of the elements in A are the directions to the eight neighbors of a grid point (in a systematic, e.g. clockwise, order). An initial formulation of necessary conditions for self-similarity of *digital straight lines* (defined by generalizing the concept of digital rays to two-sided infinite words) is given in [9]:

“To summarize, we thus have the following three specific properties which all chains of straight lines must possess [4]:

- (F1) at most two types of elements can be present, and these can differ only by unity, modulo eight;
- (F2) one of the two element values always occurs singly;

(F3) successive occurrences of the element occurring singly are as uniformly spaced as possible.”

These properties (actually listed as (1), (2) and (3) in the historic source) were illustrated by examples and based on heuristic insights. The imprecise criterion (F3) is not suitable for a formal proof, as stated, e.g., in [18].

[13] proposed grammars for chain code generation of rational digital rays based on criteria (F1), (F2) and (F3). A publication in the same year, [14], provided a first formal characterization of digital straight lines which also allowed a further specification of property (F3).

Definition 3.1 A set G of grid points satisfies the *chord property* iff for any two different points p and q of G , and any point r on the (real) line segment pq between p and q , there exists a grid point $t \in G$ such that $d_\infty(r, t) = \max(|x_r - x_t|, |y_r - y_t|) < 1$.

Theorem 3.2 (Rosenfeld 1974) *A finite irreducible 8-arc $u \in \{0, 1\}^*$ is a digital straight segment iff its assigned set of grid points $G(u)$ satisfies the chord property.*

Proof. First we show that $G(u)$ satisfies the chord property if u is a digital straight segment (Theorem 1 in [14]). Let p, q be points of $G(u)$. The line segment pq intersects grid lines $x = n$ that lie between p and q . Thus for any point $r = (x, y)$ of pq , we have $|n - x| \leq \frac{1}{2}$ for some point $(n, m) \in G(u)$. It suffices to show that whenever pq crosses a line $x = n$, the point $t = (n, m)$ of $G(u)$ on that line lies at vertical distance $|y - m| < 1$ above or below the crossing point $r = (n, y)$.

Let u be a nonempty factor of a digitization of ray $\gamma_{\alpha, \beta}$, i.e. neither p or q can be $> \frac{1}{2}$ vertically above or $\geq \frac{1}{2}$ vertically below $\gamma_{\alpha, \beta}$. Let $r = (n, y)$ be $a_r \geq 0$ vertically above $\gamma_{\alpha, \beta}$ (or $b_r \geq 0$ vertically below $\gamma_{\alpha, \beta}$). It follows that $0 \leq a_r \leq \frac{1}{2}$ (or $0 \leq b_r < \frac{1}{2}$). If r is above t , then $\gamma_{\alpha, \beta}$ intersects $x = n$ at a vertical distance $0 \leq a_t < \frac{1}{2}$ above (or at) t , and we have $y - m \leq a_r + a_t < 1$. If r is below t , then $\gamma_{\alpha, \beta}$ intersects $x = n$ at a vertical distance $0 \leq b_t \leq \frac{1}{2}$ below (or at) t , and we have $m - y \leq b_r + b_t < 1$.

Now we show that u is a digital straight segment if $G(u)$ satisfies the chord property. The following proof is due to [33], which uses the *Transversal Theorem* by Santaló [2]:

Consider a finite family \mathcal{F} of parallel straight segments in the plane \mathbb{R}^2 . If every three segments in \mathcal{F} have a common transversal, then there is a transversal common to all the segments in \mathcal{F} .

A *transversal* of a straight segment σ in \mathbb{R}^2 is a straight line in \mathbb{R}^2 which intersects σ but is not incident with σ .

Assume that the 8-arc u connects grid point (n, y_0) with grid point $(n + m, y_m)$, with $m > 0$ and $y_m - y_0 \leq m$. In case $y_m - y_0 = m$ we have a diagonal, and the chord property implies that $G(u)$ contains exactly all grid points along this diagonal, i.e. u is a digital straight segment.

Assume $y_m - y_0 \leq m - 1$ from now on. Let T_i , $0 \leq i \leq m$, be the set of all grid points in $G(u)$ on grid line $x = i$. The chord property implies that $T_i \neq \emptyset$ for $0 \leq i \leq m$, and that for any i , $0 \leq i \leq m$, there are two integers l_i and u_i such that T_i is the set of all grid points $(n + i, y)$ with $l_i \leq y \leq u_i$. We assign a (real) straight segment $L(p)$ to any grid point $p = (x, y)$:

$$L(p) = \{(x, v) : y - 0.5 < v \leq y + 0.5\}.$$

Let L_i be the union of all straight segments $L(p)$ of all grid points p in T_i , for $0 \leq i \leq m$. We have

$$L_i = \{(n + i, v) : l_i - 0.5 < v \leq u_i + 0.5\},$$

and these straight segments form a family $\mathcal{F} = \{L_0, \dots, L_m\}$ which satisfies the precondition of the Transversal Theorem:

Clearly L_0, \dots, L_m are parallel straight segments. Consider three segments L_i, L_j, L_k with $0 \leq i < j < k \leq m$. Consider two grid points $p \in L_i$ and $q \in L_k$. The straight segment pq intersects the grid line $x = j$ in a point $r = (j, y_r)$. By the chord property, there is a grid point $t = (x_t, y_t) \in G(u)$ such that $d_\infty(r, t) < 1$, i.e. t is also on the grid line $x = j$, i.e. $x_t = j$. Let s be the midpoint of the straight segment rt , and let $\varepsilon = |y_t - y_r|/2$. Consider a straight line γ parallel to pq and passing through point s . Then γ intersects the grid line $x = i$ at $x_p + \varepsilon$ or $x_p - \varepsilon$, and $x = k$ at $x_q + \varepsilon$ or $x_q - \varepsilon$. Because $\varepsilon < 0.5$ it follows that γ intersects $L(p)$, $L(t)$ and $L(q)$, i.e. it intersects L_i , L_j , and L_k .

By the Transversal Theorem it follows that there is a straight line γ intersecting all the segments L_i , with $0 \leq i \leq m$. It remains to show that such a line generates all the grid points in $G(u)$ following the grid-intersection digitization definition.

Each set T_i contains a grid point p_i such that γ intersects $L(p_i)$. We have $p_0 = (n, y_0)$ and $p_m = (n + m, y_m)$. Let q_0 and q_m be the intersection points of γ with $L(p_0)$ and $L(p_m)$, respectively, i.e. $q_0 = (n, y_0 + \lambda)$ and $q_m = (n + m, y_m + \mu)$, with $-0.5 < \lambda, \mu \leq 0.5$. The horizontal distance between q_0 and q_m is m , and the vertical distance is $|y_0 + \lambda - y_m - \mu| \leq |y_0 - y_m| + |\lambda - \mu| \leq m - 1 + |\lambda - \mu| < m$. The straight segment q_0q_m forms an angle smaller than 45° with the horizontal line, i.e. its grid-intersection digitization is specified by intersections with the vertical grid lines $x = n + i$, $0 \leq i \leq m$.

The grid-intersection digitization of q_0q_m generates the sequence of grid points p_0, p_1, \dots, p_m , and all these grid points lie in the given set $G(u)$ because γ is a transversal of all segments L_i , $0 \leq i \leq m$. Because u is an irreducible 8-arc it follows that $G(u)$ only contains the points p_0, p_1, \dots, p_m . \square

There are infinitely many irreducible two-sided infinite 8-arcs that satisfy the chord property without being digital straight lines, for example $0^\omega 10^\omega$, or (in general) ‘sparse’ occurrences of 1’s in 0^ω . The above theorem was used in

[14] to derive the following necessary conditions for (the chain code sequences of) digital straight segments:⁸

- (R1) “The runs have at most two directions, differing by 45° , and for one of these directions, the run length must be 1.
- (R2) The runs can have only two lengths, which are consecutive integers.
- (R3) One of the runs can occur only once at a time.
- (R4) ..., for the run length that occurs in runs, these runs can themselves have only two lengths, which are consecutive integers; and so on.”

These properties (actually listed as 1), 2), 3) and 4) in the historic source) still do not allow a formulation of sufficient conditions for the characterization of a digital straight segment, but they specify (F3) by a recursive argument on run lengths.

The property of *evenness* (i.e. ‘on a digital straight segment the digital slope must be the same everywhere’), as discussed in [32], is equivalent to the chord property (see Section 4 regarding balanced words which specify evenness).

It was later proved [15] that point sequences generated by the Brons algorithm possess the chord property, and [16,18] that the formal language L of digital straight segments is context-sensitive. This implies that linear-bounded or cellular automata may be specified for the recognition of digital straight segments using ‘string rewriting rules’. A result in the theory of words [46] says that the complement $\{0,1\}^* \setminus L$ of the set of all digital straight segments is a context-free language.

Criteria (F1–F3) are defined in a precise way in [20] following the recursion idea in (R1–R4). To prepare for this definition, we first introduce the following concepts:

Let $s = (s(i))_{i \in I}$ be a finite or infinite word over \mathbb{N} , for an index interval $I \subseteq \mathbb{Z}$. A letter (number) k is *singular in s* iff it appears in s , and for all $i \in I$, if $s(i) = k$ then $s(i-1) \neq k$ and $s(i+1) \neq k$, if $i-1$ and $i+1$ are in I . A letter k is *nonsingular in s* iff it appears in s and is not singular in s . A word s is *reducible* iff it contains no singular letter, or any factor of s containing only nonsingular letters is of finite length. Assume s to be reducible, and let $R(s)$ be

- (1) the length of s , if s is finite and contains no singular letter, or
- (2) the word that results from s by replacing all factors of nonsingular letters in s , which are between two singular letters in s , by their run lengths, and by deleting all other letters in s , or
- (3) the letter a if $s = a^\omega$.

A recursive application of this *reduction operation* R produces a sequence of words: $s_0 = s$, and $s_{n+1} = R(s_n)$, for all or just a finite sequence of $n \in \mathbb{N}$.

⁸ A *run* is a maximum-length factor a^n , for $a \in A$.

The definition as used in [20] is as follows (formulation following [39]):

Definition 3.3 A chain code sequence c of a two-sided infinite 8-arc satisfies the *DSL property* iff $c_0 = c$ and $c_{n+1} = R(c_n)$ are reducible words, for $n \in \mathbb{N}$; and any sequence c_n , $n \geq 0$, satisfies the following two conditions:

- (L1) There are at most two different letters a and b in c_n , and if there are two, then $|a - b| = 1$ (counting modulo 8 in the case of c_0).
- (L2) If there are two different letters in c_n , then at least one of them is singular in c_n .

Following this definition for the case of digital straight lines, it was possible [20] to derive a definition of a digital straight segment property that allowed the formulation of a necessary and sufficient condition for such chain code sequences. Possible finite words of nonsingular letters at both ends of a finite word require special attention. Let $l(s)$ and $r(s)$ denote the run lengths of nonsingular letters to the left of the first singular letter, or to the right of the last singular letter, respectively, for a finite word s . The following definition is a citation from [39]:

Definition 3.4 A finite chain code sequence c satisfies the *DSS property* iff $c_0 = c$ satisfies conditions (L1) and (L2), and any nonempty sequence $c_n = R(c_{n-1})$, for $n \geq 1$, satisfies (L1) and (L2) and the following two conditions:

- (S1) If c_n contains only one letter a , or two different letters a and $a + 1$, then $l(c_{n-1}) \leq a + 1$ and $r(c_{n-1}) \leq a + 1$.
- (S2) If c_n contains two different letters a and $a + 1$, and a is nonsingular in c_n , then if $l(c_{n-1}) = a + 1$ then c_n starts with a , and if $r(c_{n-1}) = a + 1$ then c_n ends with a .

[26] proves that an algorithm which accepts exactly those 8-arcs satisfying the DSS property recognizes just the chain code sequences of all finite, irreducible 8-arcs that have the chord property. This concluded in 1982 the process of specifying Freeman's informal constraints (F1-F3), providing an important set of constraints for the design of efficient DSS recognition procedures. We cite (without proof at this stage, but see the continued-fraction discussion later on):

Theorem 3.5 (Wu 1982) *A finite 8-arc is a digital straight segment iff its chain code sequence satisfies the DSS property.*

Note that [26] does not contain a theorem but statements about an algorithm specified by a flow-chart. However, it is easily seen that this algorithm is actually an implementation of the DSS property as cited above, i.e. [26] actually contains a proof of Theorem 3.5, covering the generation of straight lines having rational or irrational slopes.

[26] also considers the case of infinite code sequences and shows that any finite factor of a two-sided infinite chain code sequence c satisfies the DSS prop-

erty iff there is exactly one straight line with slope α and intercept β defining c by grid-intersection digitization. Based on this result, [39] concluded:

Theorem 3.6 (Hübler 1989) *A two-sided infinite 8-arc is a digital straight line iff its chain code sequence satisfies the DSL property.*

Wu's proof of Theorem 3.5 shows the equivalence of the chord property and the DSS property for irreducible finite 8-arcs; this proof is based on number theory and consists of many case discussions. Researchers therefore tried to find shorter, 'more elegant' proofs of Wu's theorem.

Material for a concise proof of Wu's theorem based on properties of Farey series was published in 1984 in [29], again in the form of an algorithm⁹. Proofs of Wu's theorem based on continued fractions were published in 1991 in two independent papers [44,51]; see also [53]. The use of continued fractions for modelling digital rays was already discussed in 1974 in [13].

Assume a rational digital straight line with slope a_0/a_1 , with integers $a_0 > a_1 > 1$. The rational number a_0/a_1 can be represented as a finite continued fraction,

$$\frac{a_0}{a_1} = [q_1, q_2, \dots, q_n] = \cfrac{1}{q_1 + \cfrac{1}{q_2 + \cfrac{1}{\ddots + \cfrac{1}{q_{n-1} + \cfrac{1}{q_n}}}}} \quad ,$$

with integer coefficients $q_i > 0$, for $1 \leq i \leq n$. The Euclidean algorithm can be used to derive such continued fractions:

$$\begin{aligned} \frac{a_0}{a_1} &= q_1 + \frac{a_2}{a_1} \quad \text{with} \quad 0 < \frac{a_2}{a_1} < 1 \, , \\ \frac{a_1}{a_2} &= q_2 + \frac{a_3}{a_2} \quad \text{with} \quad 0 < \frac{a_3}{a_2} < 1 \, , \\ &\dots\dots\dots , \\ \frac{a_{n-2}}{a_{n-1}} &= q_{n-1} + \frac{a_n}{a_{n-1}} \quad \text{with} \quad 0 < \frac{a_n}{a_{n-1}} < 1 \, , \\ \frac{a_{n-1}}{a_n} &= q_n \quad \text{with} \quad a_{n+1} = 0 \, . \end{aligned}$$

Irrational numbers can be represented by infinite continued fractions.

Continued fractions are used in [44,51] to characterize digital straight lines. Related results in number theory [40] have been of use in these studies. We review the related definitions and results as given by K. Voss in [53].

We consider a straight line passing (w.l.o.g.) through the origin and having rational slope a/b , with $\gcd(a, b) = 1$. The *characteristic triangle* of such a line is given by the vertices $(0, 0), (a, 0), (0, b)$. The *concatenation* $T_1 \otimes T_2$ of two characteristic triangles T_1 and T_2 is a characteristic triangle defined by

⁹ The DSL property is called 'linearity conditions' in this article. See also our discussion of [29] in Section 5.

the slope a/b with $a = \frac{1}{c}(a_1 + a_2)$ and $b = \frac{1}{c}(b_1 + b_2)$, for an integer c such that $\gcd(a, b) = 1$.

The numerical value of a continued fraction can also be expressed in the form of multiples of q_n ,

$$\frac{a_0}{a_1} = [q_1, q_2, \dots, q_n] = \frac{\alpha_n q_n + \beta_n}{\gamma_n q_n + \delta_n},$$

where $\alpha_n, \beta_n, \gamma_n, \delta_n$ are defined by the coefficients q_1, q_2, \dots, q_{n-1} . For $n = 1$ we have $\alpha_1 \delta_1 - \beta_1 \gamma_1 = -1$. For $n \geq 1$ we have

$$[q_1, q_2, \dots, q_{n-1}, q_{n+1}] = \frac{\alpha_{n+1} q_{n+1} + \beta_{n+1}}{\gamma_{n+1} q_{n+1} + \delta_{n+1}} = \frac{\alpha_n (q_n q_{n+1} + 1) + \beta_n q_{n+1}}{\gamma_n (q_n q_{n+1} + 1) + \delta_n q_{n+1}},$$

and thus

$$\frac{\alpha_{n+1} q_{n+1} + \beta_{n+1}}{\gamma_{n+1} q_{n+1} + \delta_{n+1}} = \frac{(\alpha_n q_n + \beta_n) q_{n+1} + \alpha_n}{(\gamma_n q_n + \delta_n) q_{n+1} + \gamma_n}.$$

This means that the slope $\frac{a_0}{a_1} = [q_1, q_2, \dots, q_n]$ of a characteristic triangle can be expressed as

$$\begin{aligned} & \frac{(\alpha_{n-1} q_{n-1} + \beta_{n-1}) q_n + \alpha_{n-1}}{(\gamma_{n-1} q_{n-1} + \delta_{n-1}) q_n + \gamma_{n-1}} \\ &= \frac{(\alpha_{n-1} q_{n-1} + \beta_{n-1}) (q_n - 1) + \alpha_{n-1} (q_{n-1} + 1) + \beta_{n-1}}{(\gamma_{n-1} q_{n-1} + \delta_{n-1}) (q_n - 1) + \gamma_{n-1} (q_{n-1} + 1) + \delta_{n-1}}. \end{aligned}$$

Therefore, the characteristic triangle defined by slope a_0/a_1 is equal to the result of repeated concatenation \otimes of one characteristic triangle with slope $[q_1, q_2, \dots, q_{n-1} + 1]$ and $q_n - 1$ triangles with slope $[q_1, q_2, \dots, q_{n-1}]$, which may be expressed by the formula

$$[q_1, q_2, \dots, q_n] = [q_1, q_2, \dots, q_{n-1} + 1] \otimes (q_n - 1) \cdot [q_1, q_2, \dots, q_{n-1}].$$

This allows us to prove Freeman's conjecture and Rosenfeld's refined hypothesis as follows:

$$\begin{aligned} [q_1, q_2, \dots, q_n] &= (q_{n-1} \cdot [q_1, q_2, \dots, q_{n-2}] \otimes [q_1, q_2, \dots, q_{n-2} + 1]) \\ &\quad \otimes (q_n - 1) \cdot ((q_{n-1} - 1) \cdot [q_1, q_2, \dots, q_{n-2}] \otimes [q_1, q_2, \dots, q_{n-2} + 1]). \end{aligned}$$

The isolated code number is $[q_1, q_2, \dots, q_{n-2} + 1]$, and $[q_1, q_2, \dots, q_{n-2}]$ is the other ('non-isolated') code number of a straight line passing through the origin and having rational slope. The run lengths q_{n-1} and $q_{n-1} - 1$ differ by 1.

Note that this proof covers only digital straight segments that are factors of rational rays, and this is actually the class of all DSSs; see Corollary 4.4.

4 Periodicity Studies in the Theory of Words

Self-similarity studies have a long history in number theory and astronomy. The theory of words [35,59] is a more recent discipline which also contains many interesting results on self-similarity, often with a special focus on irrational straight rays. Rational digital rays are specific periodic infinite words, and irrational digital rays are aperiodic infinite words which are studied under the name of *Sturmian words*.¹⁰ This section gives basic definitions and results as presented in [59]. We also cite a few proofs for purposes of illustration.

Let w be a finite or infinite word over $A = \{0, 1\}$. Let $F(w)$ be the set of all factors of w , and let $F_n(w)$ be the set of all factors of w of length n . The *complexity function* of w is defined by

$$P(w, n) = \text{card}(F_n(w)), \text{ for } n \geq 0.$$

$P(w, 0) = 1$ (the empty word is always a factor), and $P(w, 1)$ is the number of letters appearing in w . For an infinite word w , $P(w, n) \leq P(w, n+1)$ since every factor of length n can be extended to the right by at least one letter. Furthermore, $F_{m+n}(w) \subseteq F_m(w)F_n(w)$ implies $P(w, m+n) \leq P(w, m)P(w, n)$.

Consider an infinite periodic word w with period k . Then $P(w, n) \leq k$, for all $n \geq 0$, i.e. the complexity of a periodic word is limited by its period. The following theorem from [12] shows that the inverse conclusion is true as well, and generalizes these statements to eventually periodic words. Rational digital rays are periodic infinite words as stated in Theorem 1.3. For example, 10^ω is not periodic but is eventually periodic, and it is not a rational straight ray either.

Theorem 4.1 (Coven and Hedlund 1973) *The following conditions on an infinite word w are equivalent:*

- (i) w is eventually periodic,
- (ii) $P(w, n) = P(w, n+1)$ for some $n \geq 0$,
- (iii) $P(w, n) < n + k - 1$ for some $n \geq 1$, where k is the number of letters appearing in w ,
- (iv) $P(w, n)$ is bounded.

Proof. (i) \Rightarrow (iv): Let $w = uv^\omega$. Then $P(w, n) \leq |uv|$, for all $n \geq 0$.

(iv) \Rightarrow (iii): Let $P(w, n) < p$ for all $n \geq 0$. If k is the number of letters appearing in w then $P(w, 1) = k < p$, i.e. $p \geq k+1$. Then $P(w, p-k+1) < p$.

(iii) \Rightarrow (ii): Assume (ii) is not true, i.e. $P(w, m-1) < P(w, m)$, for all $m \geq 0$; then we would have $n+k-1 > P(w, n) \geq P(w, 1) + n - 1 = k+n-1$, for some $n \geq 1$, which is impossible.

¹⁰ Named after the mathematician C.F. Sturm (1803-1855). We follow [59] with respect to the definition of Sturmian words. Some authors also used the name ‘Sturmian words’ for lower digital straight lines; see, for example, [52].

(ii) \Rightarrow (i): Consider the *factor graph* $G_n(w)$ which is a labelled graph with vertex set $F_n(w)$ and edge set $E = \{(bu, a, ua) : a, b \in A \wedge bua \in F_{n+1}(w)\}$. There is at least one edge starting at each vertex in $G_n(w)$ because every factor of length n is a prefix of a factor of length $n + 1$. Since $P(w, n) = P(w, n + 1)$ there is actually exactly one edge leaving each vertex, i.e. any strongly connected component of $G_n(w)$ is a simple circuit. The word w is the label of an infinite path passing through $G_n(w)$, i.e. it will loop through a fixed circuit after some prefix, i.e. its labels are eventually periodic. \square

A sequence $(v_n)_{n \geq 0}$ of finite words over an alphabet A *converges* to an infinite word w if every prefix of w is a prefix of all but a finite number of words v_n . For example, the sequence $0^n 1^n$ converges to 0^ω .

Let $f_0 = 0$, $f_1 = 01$ and $f_{n+1} = f_n f_{n-1}$, for $n \geq 2$. The sequence of lengths $|f_n|$ is the Fibonacci sequence $F_1 = 1, F_2 = 2, F_3 = 3, F_4 = 5, \dots$. The sequence $(f_n)_{n \geq 0}$ converges to the *Fibonacci word*

$$f = 0100101001001010010100100101001001\dots$$

and, for example, 01001 is a prefix of f_n for $n \geq 4$. The Fibonacci word can also be defined by a morphism: for

$$\varphi : \begin{array}{l} 0 \mapsto 01 \\ 1 \mapsto 0 \end{array}$$

we have $f = \varphi^\omega(0)$.

Definition 4.2 A *Sturmian word* is an infinite word $w = a_1 a_2 a_3 \dots$ over a binary alphabet A that has exactly $n + 1$ factors of length n , for every $n \geq 0$.

Any suffix of a Sturmian word is again a Sturmian word. The Fibonacci word is Sturmian. The *Thue-Morse word* $t = \mu^\omega(0) = 0110100110010110\dots$, with

$$\mu : \begin{array}{l} 0 \mapsto 01 \\ 1 \mapsto 10 \end{array}$$

is another example of a Sturmian word.

A Sturmian word w is defined by $P(w, n) = n + 1$, for $n \geq 0$. According to Theorem 4.1, any aperiodic word has complexity $P(w, n) \geq n + 1$, for $n \geq 0$, i.e. Sturmian words have minimal complexity $P(w, n)$ among aperiodic infinite words. The value $P(w, 1) = 2$ shows that w is defined on a binary alphabet, here $A = \{0, 1\}$.

A *right special factor* of an infinite word w is a finite word u such that $u0$ and $u1$ are factors of w . A word w is Sturmian iff it has exactly one right special factor of each length $n \geq 0$. The empty word is always the special right factor of length zero. For the Fibonacci word f we have: 11 is not a factor, so 0 is the only right special factor of length one; 000 and 011 are not factors, so 10 is the only factor of length two; etc.

The *height* $h(w)$ of a word $w \in A^*$ is the number of letters equal to 1 in w . Given two words v and w of the same height, $\delta(v, w) = |h(v) - h(w)|$ is their *balance*. A set $X \subset A^*$ of words is *balanced* iff $|v| = |w|$ implies $\delta(v, w) \leq 1$ for all pairs of words $v, w \in X$.¹¹ An infinite word w is *balanced* if its set of factors is balanced.

The *slope* of a nonempty word w is the number $\pi(w) = h(w)/|w|$. We have

$$\pi(uv) = \frac{|u|}{|uv|}\pi(u) + \frac{|v|}{|uv|}\pi(v) .$$

It is possible to show [59] that an infinite word w is balanced iff, for all non-empty factors u, v of w , we have

$$|\pi(u) - \pi(v)| < \frac{1}{|u|} + \frac{1}{|v|} .$$

This shows that the sequence of slopes is a Cauchy sequence, i.e. a balanced infinite word possesses a uniquely defined slope based on the slopes of its finite prefixes. Let w be an infinite balanced word, and let w_n be the prefix of length n of w , for $n \geq 1$. Then the sequence $(\pi(w_n))_{n \geq 1}$ converges for $n \rightarrow \infty$. For example, for the Fibonacci word f we have $h(f_n) = F_{n-2}$ and $|f_n| = F_n$, and F_{n-2}/F_n converges to $\pi(f) = 1/\tau^2$ with $\tau = (1 + \sqrt{5})/2$.

Digital rays, i.e. infinite words, are defined for rational or irrational slope by using the slope of the generating ray. The following theorem from [1] was actually formulated for *mechanical words*, which is what digital rays are called in the theory of words.

Theorem 4.3 (Morse and Hedlund 1940) *Let w be a digital ray with slope α . Then w is balanced of slope α .*

Proof. Let w be a lower digital ray. The height of a factor $u = w(n) \dots w(n+p-1)$ is the number $h(u) = \lfloor \alpha(n+p) + \beta \rfloor - \lfloor \alpha n + \beta \rfloor$; thus

$$\alpha \cdot |u| - 1 < h(u) < \alpha \cdot |u| + 1 , \quad \text{i.e.} \quad \lfloor \alpha \cdot |u| \rfloor \leq h(u) \leq 1 + \lfloor \alpha \cdot |u| \rfloor .$$

This shows that $h(u)$ takes only two consecutive values when u ranges over w factors of fixed length, i.e. w is balanced. Moreover, it follows that

$$\left| \frac{h(u)}{|u|} - \alpha \right| = |\pi(u) - \alpha| < \frac{1}{|u|} ,$$

and thus $\pi(u) \rightarrow \alpha$ for $|u| \rightarrow \infty$, and α is the slope of w as defined for balanced words. \square

Note that the inequality $|\pi(u) - \alpha| < 1/|u|$ also provides a criterion for evaluating the accuracy of an estimated slope based on a finite digital straight

¹¹ [32] calls non-balanced words *uneven* and shows that an infinite 8-arc has the chord property iff it has no uneven finite factors.

segment. An alternative method of evaluating the accuracy of an estimated slope will be discussed at the end of Section 5. This inequality $|\pi(u) - \alpha| < 1/|u|$ also allows us to state that ‘rational digital rays are sufficient for studies in pattern recognition’:

Corollary 4.4 *Any digital straight segment is a factor of a rational digital ray.*

Proof. An interval in $[0, 1)$ of width $1/|u|$, containing an irrational number α , also contains rational numbers α' satisfying $|\pi(u) - \alpha'| < 1/|u|$. \square

We conclude this section by citing the main theorem on irrational digital rays:

Theorem 4.5 (Morse and Hedlund 1940) *The following conditions are equivalent for an infinite word w :*

- (i) w is Sturmian,
- (ii) w is balanced and aperiodic,
- (iii) w is an irrational digital ray.

Note that a balanced infinite word is not always a digital ray when the slope is rational. For example, 01^ω is not a digital ray. It has slope 1, but $l_{1,\beta} = 1^\omega$. Only (purely) periodic infinite balanced words are rational digital rays.

5 Counts and Spirographs

We have already cited several studies in which number theory has contributed to studies on digital straightness. The following theorem from [48] is from the theory of words:

Theorem 5.1 (Mignosi 1991) *The number of balanced words of length n is*

$$1 + \sum_{i=1}^n (n+1-i)\phi(i) ,$$

where ϕ is Euler’s totient function.

A finite word u is balanced iff it is a factor of some irrational digital ray [59]. Due to Corollary 4.4 it follows that any finite balanced word u is also a factor of some rational digital ray, i.e. Theorem 5.1 actually specifies the number of digital straight segments of length n starting at the origin $(0, 0)$. The technical report [52] contains an alternative proof of Theorem 5.1 and also an algorithm for random generation of lower digital straight segments of length n .

[41] considers the same set of segments u of lower digital rays, defined by $0 \leq x \leq n$, $0 \leq \alpha \leq 1$, and $0 \leq \beta < 1$, i.e. the first grid point in the set $G(u)$ of assigned grid points is $(0, 0)$, and $G(u)$ contains exactly $n + 1$ grid points.

In [41] it is shown that the number of such digital straight segments passing through the origin is

$$(1) \quad \frac{1}{\pi^2} \cdot n^3 + \mathcal{O}(n^2 \cdot \log n).$$

The Euler function $\phi(i)$ tends to $6i/\pi^2$, i.e. the formula in Theorem 5.1 may be transformed into Equ. (1). Suggestions about using Farey series for modelling digitized lines were already made in [7,13,16,31]. A *Farey series* $F(n)$ of order $n \geq 1$ is defined as the ascending series of irreducible fractions between 0 and 1 whose denominators do not exceed n , i.e. all rational numbers a_0/a_1 , with $0 \leq a_0 \leq a_1 \leq n$ and a_0 and a_1 relatively prime, sorted in increasing order. For example, for $n = 5$ we have the sequence

$$\frac{0}{1}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{1}{1}$$

In [16] it is shown that digital straight segments of length n , passing through the origin, are in one-one correspondence with the n th Farey series. This is actually already a proof of Equ. (1).

There is an obvious one-one correspondence between the set of digital line segments starting at $(0, 0)$ and the set of linear partitions of an $n \times n$ orthogonal grid, where a *linear partition* of a set S is defined to be any partition of S into sets X and $S \setminus X$ by a line γ such that the sets X and $S \setminus X$ belong to different halfplanes defined by line γ . Of course, any digital straight segment consisting of $n + 1$ points and beginning at $(0, 0)$ defines exactly one linear partition, but there are also further linear partitions of the $n \times n$ grid which do not correspond to digital rays starting at $(0, 0)$.

The number of linear partitions of an $m \times n$ orthogonal grid is considered in [43]. There it is shown that the number of such partitions is equal to

$$(2) \quad \frac{3}{\pi^2} \cdot m^2 \cdot n^2 + \mathcal{O}(m^2 \cdot n \cdot \log n) + \mathcal{O}(m \cdot n^2 \cdot \log \log n)$$

where it is assumed that $m \leq n$. This result can be understood as the ‘capacity’ of a digital picture of size $m \times n$ with respect to digital rays, i.e., it shows how many digital rays can be discriminated by digitization on an $m \times n$ orthogonal grid.

Both asymptotic formulas, for the number of digital straight segments and for the number of linear partitions, can be derived by using well-known formulas for average values of number-theoretical functions and Riemann-Stieltjes integration.

Dorst and Duin [29] developed a theory of *spirographs* for establishing links between digital rays and number theory (Farey series, continued fractions). Figure 5 shows on the right a ray $y = \alpha x$, with $0 < \alpha < 1$, passing through grid point $(0, 0)$ and intersecting grid line $x = 0$ in the interval $[0, 1)$, and a few parallel shifts of this ray. For any grid line $x = n$ there is exactly one grid point (n, y_n) such that ray $y = \alpha x + \beta_n$ passes through (n, y_n) and intersects grid

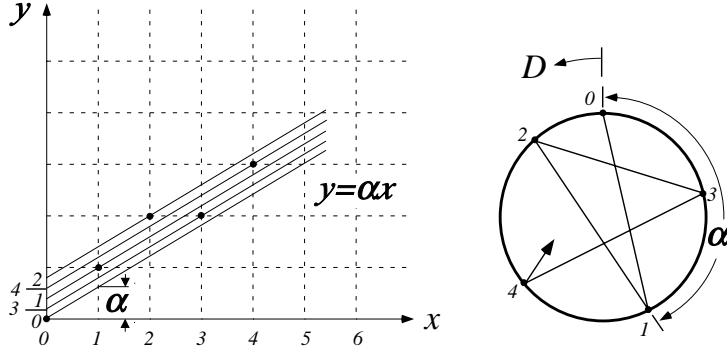


Fig. 5. Five intersection points (left) for grid lines $n = 0, 1, \dots, 4$, mapped into a spirograph (right).

line $x = 0$ in the interval $[0, 1)$. Spirographs¹² are diagrams which visualize and model the distribution of these intersection points in $[0, 1)$. See Fig. 5 on the right: Assume a circle with perimeter 1 and mark a first vertex on this circle representing the intersection point for grid line $x = 0$, i.e. $\beta_0 = 0$. In clockwise orientation, proceed from the first vertex to a second vertex on this circle at radial distance α representing the intersection point for grid line $x = 1$, etc.

Definition 5.2 A *spirograph* $S(\alpha, n)$ is a set of n points on a circle with unit perimeter, marked $0, 1, \dots, n - 1$, and defined by parallel rays with slope α intersecting grid lines $x = 0, x = 1, \dots, x = n - 1$ at grid point positions.

For simplicity we identify these points in $S(\alpha, n)$ with their marks. If α is rational then there is only a finite number of such rays, creating a finite set of intersection points in $[0, 1)$, with a periodic repetition of these intersection points for n to infinity, and thus only a bounded number of marked points on the spirograph, for any n . The *topology* of a spirograph $S(\alpha, n)$ is the order modulo n of the marked points on the circumference of the circle.

The intervals between intersection points in $[0, 1)$, for α rational, specify intervals of intercepts β such that $y = \alpha x + \beta$ leads to the same lower digital ray for all values β within the same interval (see Theorem 1.2).

The distance $D_\alpha(i, j)$ between two points $i, j \in S(\alpha, n)$, $0 \leq i, j < n$, is the length of the arc extending anticlockwise from i to j :

$$D_\alpha(i, j) = (i - j)\alpha - \lfloor (i - j)\alpha \rfloor.$$

The smallest distance D_{right} to the right (clockwise) of point $0 \in S(\alpha, n)$ is $D_{\text{right}} = \min\{D_\alpha(i, 0) : i \neq 0 \wedge i \in S(\alpha, n)\}$. Let $i_{\text{right}} = \min\{k \neq 0 : k \in S(\alpha, n) \wedge D_\alpha(k, 0) = D_{\text{right}}\}$ be the point determining this minimum distance. Similarly, let $D_{\text{left}} = \min\{D_\alpha(0, i) : i \neq 0 \wedge i \in S(\alpha, n) \wedge D_\alpha(0, i) \neq 0\}$ and $i_{\text{left}} = \max\{k \neq 0 : k \in S(\alpha, n) \wedge D_\alpha(0, k) = D_{\text{left}}\}$.

¹² The name is that of a children's toy for drawing curves.

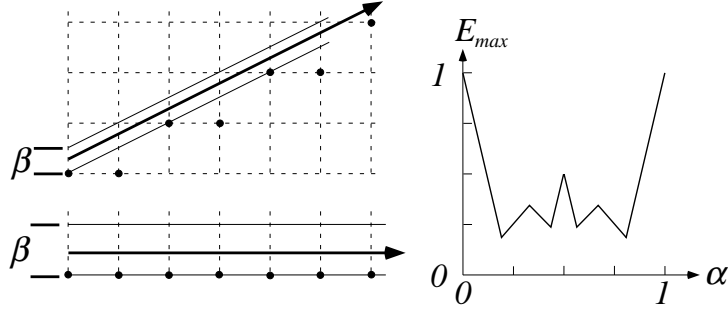


Fig. 6. Right: the maximum error for β is 1 or 0.5, respectively for $\alpha = 0$ or $\alpha = 0.5$. Left: maximum error of β as a function of estimated α -value, for $n = 6$ [29].

Now we are prepared to state a few results from [29] using their theory of spirographs. We select those related to the possible accuracy of estimating the slope and the intercept of a generating ray as a function of the length of the given digital straight segment.

By definition, $D_{\text{right}} = \alpha i_{\text{right}} - \lfloor \alpha i_{\text{right}} \rfloor$ and $D_{\text{left}} = \alpha i_{\text{left}} - \lfloor \alpha i_{\text{left}} \rfloor$. Therefore the bounds on α that preserve the topology of the spirograph are

$$\frac{\lfloor \alpha i_{\text{right}} \rfloor}{i_{\text{right}}} \leq \alpha < \frac{\lfloor \alpha i_{\text{left}} \rfloor}{i_{\text{left}}},$$

and these bounds for α are the best rational approximations for α with fractions whose denominator does not exceed $n - 1$. The proof of this fact can be based on the property that $\lfloor \alpha i_{\text{right}} \rfloor / i_{\text{right}}$ and $\frac{\lfloor \alpha i_{\text{left}} \rfloor}{i_{\text{left}}}$, with i_{right} and i_{left} obtained from spirograph $S(\alpha, n)$, are two successive fractions in the Farey series $F(n - 1)$.

The intercept estimation problem is illustrated on the right in Fig. 6. Every pair of values of α , $0 \leq \alpha < 1$, and n , $n \geq 1$, allows an interval of β -values, $0 \leq \beta < 1$ of possible intercepts such that the given lower straight line segment of length n is a digitization of ray $\alpha x + \beta$. The width of this interval is defined to be the maximal possible error $E_{\text{max}}(\alpha, n)$. The calculation of error diagrams, see the left of Fig. 6 for an example, is possible based on spirograph studies. The maximum error $E_{\text{max}}(\alpha, n)$ is defined by the maximum arc length in spirograph $S(\alpha, n)$:

Theorem 5.3 (Dorst and Duin 1984) *We have $E_{\text{max}}(\alpha, n) = D_{\text{right}} + D_{\text{left}}$, where the distances D_{right} and D_{left} are calculated in spirograph $S(\alpha, n + 1)$.*

The formula $D_{\text{right}} + D_{\text{left}} = \lceil \alpha i_{\text{left}} \rceil - \lfloor \alpha i_{\text{right}} \rfloor + \alpha(i_{\text{right}} - i_{\text{left}})$, with values from $S(\alpha, n + 1)$, allows a simple calculation of the errors $E_{\text{max}}(\alpha, n)$. If α is a fraction a/b in the Farey series $F(n)$, then $E_{\text{max}}(a/b, n) = 1/b$.

6 Algorithms for DSS Recognition

By now there have been many publications on (efficient) DSS recognition algorithms. The computational problem is as follows: The input is a sequence of chain codes $i(0), i(1), \dots$ with $i(k) \in \{0, 1\}$, $k \geq 0$. An *off-line DSS recognition algorithm* decides for finite words $u \in \{0, 1\}^*$ whether u is a digital straight segment or not. An *on-line DSS recognition algorithm* reads successive chain codes $i(0), i(1), \dots$ and specifies the maximum length $k \geq 0$ such that $i(0), i(1), \dots, i(k)$ is a digital straight segment, and $i(0), i(1), \dots, i(k), i(k+1)$ is not. A recognition algorithm has linear run time behavior (a *linear algorithm* for short), i.e. it runs in $\mathcal{O}(n)$ time, if it performs at most $\mathcal{O}(|u|)$ basic computation steps for any finite input word $u \in \{0, 1\}^*$. Analogous definitions can be given for 4-DSS recognition algorithms. An on-line algorithm is linear if it uses *in the average* a constant number of operations for any incoming chain code symbol. Linear off-line algorithms for DSS recognition based on the DSS property (as defined in Definition 3.4) were published in 1981 in [20] and in 1982 in [26].¹³ A linear off-line algorithm for cellular straight segment recognition, based on convex hull construction, is briefly sketched in [24]. Two linear on-line algorithms for DSS recognition were published in 1982 by Creutzburg, Hübner and Wedler [21,22]; one of them is an on-line version of the off-line algorithm published in [20]. Algorithms for polygonal approximations of digitized curves, not directly related to models of digital straightness (see, for example, [10,19,23]), will not be reviewed here.

The general problem of decomposing a 4- or 8-arc into a sequence of 4-DSSs or DSSs, which includes 4-DSS or DSS recognition as a subproblem, is discussed in, e.g. [42,50,54,57]. Obviously, linear on-line DSS recognition algorithms will support linear decomposition algorithms, but linear off-line algorithms will only allow quadratic run-time behavior.

The design of a DSS recognition algorithm may be based on a unique characterization of digital straight segments, such as

- (C1) the original definition of a DSS based on grid-intersection digitization,
- (C2) a characterization by pairs of tangential lines (special cases: (C2.1a) Theorem 2.1, (C2.1b) Corollary 4, (C2.2) Theorem 2.4, and (C2.3) Theorem 2.6,
- (C3) the equivalence with the chord property, see Theorem 3.2, or
- (C4) the DSS property, see Theorem 3.5,

and further characterizations have also been used for the design of DSS recognition algorithms. Approaches following (C4) are normally called *linguistic techniques*. For an early version of a linguistic DSS recognition algorithm see [16] (however, this was not yet based on the correct DSS property, which became known later).

¹³ [32] discusses a flaw in the Wu algorithm.

$CC_0 = 1101110111011101111011101111011101110$
 $11110111011101110111101110111011110111$
 $s(0) = 0, \quad n(0) = 1, \quad l(0) = 2, \quad r(0) = 3$
 $CC_1 = 33343343343334334$
 $s(1) = 4, \quad n(1) = 3, \quad l(1) = 3, \quad r(1) = 0$
 $CC_2 = 2232$
 $s(2) = 3, \quad n(2) = 2, \quad l(2) = 2, \quad r(2) = 1$
 $CC_3 = \varepsilon$

Fig. 7. Input example for algorithm **CHW_1982a** [21,22].

We review in detail one of the historically first linear on-line algorithms for DSS recognition as published in 1982 in [21,22] which utilizes the DSS property **(C4)**.

Algorithm CHW_1982a

The input sequence is $CC = i(0)i(1)i(2) \dots i(n)$, $i(k) \in A = \{0, 1, \dots, 7\}$ for $0 \leq k \leq n$. Let $CC_0 = CC$, and, if $CC_{k-1} \neq \varepsilon$ (the empty word) then $CC_k = R(CC_{k-1})$ where R denotes the reduction operation specified above for defining the DSL and DSS properties in Section 3. Let $l(k)$ and $r(k)$ be the run lengths of nonsingular letters to the left of the first singular letter in CC_k , or to the right of the last singular letter; see Definition 3.4. Let $s(k)$ be the singular element in CC_k if there is one, otherwise let $s(k) = -1$, and let $n(k)$ be the second element in CC_k if there is one, otherwise let $n(k) = -1$. See Fig. 7 for an example. The input chain code CC_0 is now represented by a *syntactic code*, which is

	s	n	l	r
0	0	1	2	3
1	4	3	3	0
2	3	2	2	1

for the example in Fig. 7. A syntactic code consists of integers in four columns s, n, l, r . The DSS property (see Definition 3.4) specifies constraints on these integers such that the given word $CC = i(0)i(1)i(2) \dots i(n)$ may be classified as being a DSS or not. Before starting to read a word CC , all values in columns s and n are initialized to be -1 , and all values in columns l and r are initialized to be 0. Now assume that the syntactic code has already been calculated for an input sequence of length greater than or equal to zero, and assume that letter d is read as the next chain code of the input sequence. Let $\mathbf{N}(k, a, b)$ be true iff $|a - b| = 1$ for $k \geq 1$, and $|a - b| \pmod{8} = 1$ for $k = 0$. The algorithm uses different tests which follow straightforwardly from the DSS property:

$$\begin{aligned}
 T_1(k, d) : & n(k) = -1 \wedge s(k) = -1 \wedge \\
 & [k > 0 \rightarrow l(k-1) \leq d+1 \wedge r(k-1) \leq d+1] \\
 T_2(k, d) : & n(k) \neq -1 \wedge s(k) = -1 \wedge T_{2.1}(k, d) \wedge T_{2.2}(k, d)
 \end{aligned}$$

$$\begin{aligned}
T_{2.1}(k, d) &: d = n(k) \\
T_{2.2}(k, d) &: \mathbf{N}(k, d, n(k)) \wedge [k > 0 \rightarrow \\
&\quad \{l(k-1) \leq n(k) \vee (l(k-1) = d \wedge l(k) \neq 0)\} \wedge \\
&\quad \{r(k-1) \leq n(k) \vee (r(k-1) = d \wedge r(k) \neq 0)\}] \\
T_3(k, d) &: d = s(k) \wedge r(k) = 0 \wedge \\
&\quad l(k) = 1 \wedge s(k+1) = -1 \wedge n(k+1) \leq 1 \wedge [k > 0 \rightarrow \\
&\quad l(k-1) \leq s(k) \wedge \{r(k-1) \leq s(k) \vee r(k-1) = n(k)\}] \\
T_4(k, d) &: d = n(k) \wedge [s(k+1) = -1 \rightarrow r(k) \leq n(k+1)] \wedge \\
&\quad [s(k+1) \neq -1 \rightarrow r(k) + 1 \leq n(k+1) \vee \\
&\quad \{r(k) + 1 = s(k+1) \wedge r(k+1) \neq 0\}] \\
T_5(k, d) &: d = s(k) \wedge r(k) \neq 0
\end{aligned}$$

The algorithm is specified in Fig. 8. The algorithm ‘inserts’ every new element d into the syntactic code as long as the incoming chain code sequence satisfies the DSS property.

Algorithm **CHW_1982a** runs in linear time: $|CC_{k+1}| \leq 1/2 \cdot |CC_k|$, for $k \geq 0$ and any incoming DSS chain code. There is only one loop in this algorithm, in the case that a new element needs to be added to one of the CC_k ’s. Therefore, the run time $t(n)$, for inputs of length $n = |CC_0|$, is on the order

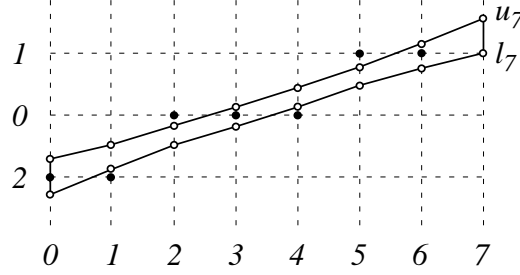
$$\mathcal{O}(|CC_0| + |CC_1| + \dots + |CC_{\log n}|) = \mathcal{O}\left(\sum_{k=0}^{\log_2 n} \frac{n}{2^k}\right) = \mathcal{O}(n) .$$

```

      k = 0
1   if T1(k, d) then goto 10
      if T2(k, d) then goto 20
      if T3(k, d) then goto 30
      if T4(k, d) then goto 40
      if T5(k, d) then goto 50
      goto 100
10  n(k) = d, l(k) = 1, return “yes”
20  if T2.1(k, d) then goto 21
      if T2.2(k, d) then goto 22
      goto 100
21  l(k) = l(k) + 1, return “yes”
22  s(k) = d, return “yes”
30  s(k) = n(k), n(k) = d, l(k) = 0, r(k) = 2
      return “yes”
40  r(k) = r(k) + 1, return “yes”
50  d = r(k), r(k) = 0, k = k + 1, goto 1
100 for m = 0 until k - 2 do r(m) = s(m + 1)
      if k ≠ 0 then r(k - 1) = d
      return “no”

```

Fig. 8. DSS recognition algorithm **CHW_1982a** based on syntactic codes.

Fig. 9. Digitization polygon for $u = 0100100$.

It also follows that the number of relevant integers in the syntactic code is limited by $\mathcal{O}(\log n)$, because the index m of the last non-empty word CC_m satisfies $m \leq \log_2 n$. A stronger inequality is

$$n \geq \left(\frac{1}{2} + \frac{1}{4}\sqrt{2}\right)(1 + \sqrt{2})^m - 2.$$

For example, $n = 2377 \dots 5739$ only requires reduced chain code words CC_k for $k \leq m = 9$. Of course, representing a digital straight segment by the two end points of one of its possible preimages is an even shorter representation. A discussion of the time efficiency of DSS recognition algorithms may also be accompanied by a discussion of their memory requirements.

We conclude this section with brief reviews of some other DSS recognition algorithms (many more have been published which will be not reviewed here due to space limitations, e.g. [25,34,37,38,45,47,55,56]). The intention is a polygonalization of 8-arcs by segmenting them into maximum-length DSS's.

Algorithm CHW_1982b

The second linear on-line algorithm, published in [21,22], uses the possible preimages, see approach **(C1)** above: as long as the union of all possible preimages is non-empty we may continue reading the next chain code element of the given 8-arc.

As in the proof of Theorem 3.2 we consider a family of parallel segments $(x, l_x)(x, u_x)$ of grid lines $x = 0, x = 1, \dots, x = n$ for a given digital straight segment $u \in \{0, 1\}^*$ of length n connecting grid point $p_0 = (0, 0)$ with grid point p_n , passing through grid points p_1, \dots, p_{n-1} . However, this time we assume that $-0.5 \leq l_x \leq u_x < n + 0.5$ specify segments of grid lines $x = 0, x = 1, \dots, x = n$, being the union of all intercepts of these grid lines with possible preimages (i.e. straight line segments) of u with respect to grid intersection digitization, i.e. $x - 0.5 \leq l_x \leq u_x < x + 0.5$. A segment $(x, l_x)(x, u_x)$ may degenerate into a single point, i.e. $l_x = u_x$, and the segment $(x, l_x)(x, u_x)$ must not contain the grid point p_x , for $x = 0, 1, \dots, n$; see Fig. 9 for an example. The point sequence $(0, u_0), (1, u_1), \dots, (n, u_n), (n, l_n), (n - 1, l_{n-1}), \dots, (0, l_0)$ defines the *digitization polygon* of straight line segment u . Because a segment

$(x, l_x)(x, u_x)$ may degenerate into a single point, the digitization polygon need not be simple. Note that the segments $(0, u_0)(n, l_n)$ and $(0, l_0)(n, u_n)$ are contained in this digitization polygon.

Now assume that u is extended by another chain code $a \in \{0, 1\}$. The 8-arc ua is a DSS iff it possesses a digitization polygon. The linear on-line algorithm **CHW_1982b**, specified in detail in [21,22], uses the digitization polygon of u to update this for ua if possible, or returns “no” otherwise if there is no digitization polygon for ua . The algorithm was also published in [36].

Algorithm S_1983

[27] also specifies a linguistic technique (i.e. type **(C4)**) for segmenting an 8-arc into DSSs. As in **CHW_1982a**, the algorithm **S_1983** involves only integer operations following the syntactic rules specified in the DSS property. A parser checks the rules related to one layer k , and (eventually) activates a parser for the next layer $k+1$. Several parsers at different levels may be active simultaneously.

This specifies a different point of view on the approach implemented in **CHW_1982a**, which may support a more obvious implementation of the syntactic rules specified in the DSS property.

The maximum number m of layers is bounded by $4.785 \cdot \log_{10} n + 1.672$, and this maximum is taken on in cases of digital rays having slope a/b where a and b are consecutive Fibonacci numbers [8], but the average depth is less than half of this value [8].

[27] reports on experiments comparing polygons, whose vertices are the *break points* of segmented 8-arcs, with polygonal preimages used to obtain these 8-arcs by grid-intersection digitization (Bresenham algorithm). It states an ambiguity in detecting maximum-length DDSs defined by these break points.

Algorithm AK_1985

[30] has been cited already above with respect to its material on pairs of tangential lines for 8-arcs. It specifies a DSS recognition algorithm which follows approach **(C2.1b)**. Assume an 8-arc $u \in \{0, 1\}^*$ of length n connecting grid point $p_0 = (0, 0)$ with grid point p_n , passing through grid points p_1, \dots, p_{n-1} . *Critical points* form a minimal subset of $G(u) = \{p_0, p_1, \dots, p_n\}$ defining a pair of tangential lines having a minimum distance in the y -axis direction (and $G(u)$ between or on these lines). An 8-arc u is a DSS iff this distance between such a pair of tangential lines is less than 1; see Corollary 4.

W.l.o.g. assume that u possesses four critical points $q_1, q_2, r_1, r_2 \in G(u)$ where $q_1 q_2$ specifies a *nearest support below* and $r_1 r_2$ a *nearest support above*

u . Then u is uniquely specified either by n and q_1, q_2 , or by n and r_1, r_2 . [30] describes a linear off-line (!) algorithm for calculating the nearest support below and/or above. A final test (Corollary 4) decides whether or not u is a DSS.

This algorithm is also used to specify a linear off-line (!) algorithm for calculating the digitization polygon (see algorithm **CHW_1982b**). [30] also discusses the calculation of digitization polyhedra for digital straight segments in three-dimensional space.

Algorithm CHS_1988a

[36] specifies three different linear on-line DSS recognition algorithms. The first one is a slightly improved version of algorithm **CHW_1982b**. The second one also follows approach **(C1)**; however, this time the grid-intersection digitization definition is used to perform DSS recognition based on solving a separability problem for a monotone polygon.

Assume an 8-arc $u \in \{0, 1\}^*$ of length n connecting grid point $p_0 = (0, 0)$ with grid point p_n , passing through grid points p_1, \dots, p_{n-1} . Let $p_k = (k, I_k)$, for $k = 0, 1, \dots, n$. The *weak digitization polygon* of u is defined by vertices $(0, I_0 + 0.5), (1, I_1 + 0.5), \dots, (n, I_n + 0.5), (n, I_n - 0.5), (n - 1, I_{n-1} - 0.5), \dots, (0, I_0 - 0.5)$. The weak digitization polygon of an 8-arc u is monotonic in the x -direction. The separability problem is now as follows: The arc u is a DSS iff the upper polygonal chain $(0, I_0 + 0.5), (1, I_1 + 0.5), \dots, (n, I_n + 0.5)$ of its weak digitization polygon may be separated from its lower polygonal chain $(n, I_n - 0.5), (n - 1, I_{n-1} - 0.5), \dots, (0, I_0 - 0.5)$ by a straight line not intersecting the upper or lower polygonal chain. [36] details a linear on-line algorithm for solving this separability problem for extended 8-arcs ua , $a \in \{0, 1\}$, based on a solution of the separability problem for u . Note that this separability problem may also be stated as a visibility problem (visibility of edge $(0, I_0 - 0.5)(0, I_0 + 0.5)$ from edge $(n, I_n - 0.5)(n, I_n + 0.5)$, or vice versa).

Algorithm CHS_1988b

The third linear on-line DSS recognition algorithm proposed in [36] follows **(C2.1b)**; it is similar to (and independent of the publication of) the linear off-line algorithm **AK_1985**. Algorithm **CHS_1988b** uses the critical points calculated for u to calculate updated critical points for the extended 8-arc ua , $a \in \{0, 1\}$, if possible, and returns “no” otherwise. The algorithm is quite short, allowing a quick implementation. [36] also contains a geometric analysis of possible or impossible locations of critical points. For example, if a critical point of word u is cancelled later on in an extended word uv , it cannot become a critical point again for extensions of uv .

Algorithm K_1990

[42] discusses the recognition of digital 4-straight segments (the boundaries of cellular complexes) following approach **(C2.3)**. It specifies the calculation of a *narrowest strip*, defined by the nearest support below and above, and briefly sketches a linear on-line DSS recognition algorithm (see also the linear off-line algorithm **AK_1985**, and the linear on-line algorithm **CHS_1988b**, both for 8-arcs). Algorithm **K_1990** is reported in greater detail in [57].

Algorithm SD_1991

[50] discusses a linear off-line DSS recognition algorithm following the linguistic approach **(C4)**. It starts with the linear off-line Wu algorithm [26] and corrects the flaw detected in [32]. The paper [50] also contains basic research on digital straightness.

Algorithm DR_1995

A linear on-line DSS recognition algorithm is described in [54] which follows the **(C2.1a)** approach (note: their ‘naive line’ is identical to a digital straight line), i.e. it is based on an updated test of a double Diophantine equation which is basically similar to a test of whether the grid point set $G(u)$ is in a narrowest strip (see algorithm **K_1990**) of arithmetical width $\max\{|a|, |b|\}$.

7 Conclusions

A straight line seems to be a simple object. Our review (which is not yet complete) demonstrates that digital straight lines are actually a very challenging subject, and many interesting results are known to date. Missing is still a comprehensive and comparative performance evaluation of the DSS recognition algorithms suggested so far in the literature. A statistical analysis of measured time complexities would also be of interest. The random DSS generation algorithm of [52] may be used for input data.

The segmentation of a (closed) 8-curve into maximum-length DSS’s depends upon the starting point and orientation of the traversal. It would be of interest to analyze the possible variations of these segmentations.

Acknowledgments: The authors thank Filippo Mignosi for informing us about [59], a forthcoming book on the theory of words.

References

- [1] M. Morse and G. A. Hedlund. Symbolic dynamics II: Sturmian sequences. *Amer. J. Math.*, 61:1–42, 1940.
- [2] L. A. Santaló. Complemento a la nota: un teorema sobre conjuntos de paralelepípedos de artistas paralelas. *Publ. Inst. Mat. Univ. Nac. Litoral*, 2:49–60, 1940.
- [3] J. Loeb. Communication theory of transmission of simple drawings. In W. Jackson, editor, *Communication Theory*, pages 317–327. Butterworths, London, 1953.
- [4] H. Freeman. Techniques for the digital computer analysis of chain-encoded arbitrary plane curves. In *Proc. Natl. Elect. Conf.*, volume 17, pages 421–432, 1961.
- [5] J. Bresenham. An incremental algorithm for digital plotting. In *Proc. ACM Natl. Conf.*, 1963.
- [6] J. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems J.*, 4:25–30, 1965.
- [7] U. C. Montanari. A method for obtaining skeletons using a quasi-Euclidean distance. *J. ACM*, 15:600–624, 1968.
- [8] D. E. Knuth. *The Art of Computer Programming*, Vol. 2, pages 309, 316–332. Addison-Wesley, Reading, Mass., 1969.
- [9] H. Freeman. Boundary encoding and processing. In B. S. Lipkin and A. Rosenfeld, editors, *Picture Processing and Psychopictorics*, pages 241–263. Academic Press, New York, 1970.
- [10] G. U. Montanari. A note on minimal length polygonal approximation to a digitized contour. *Comm. ACM*, 13:41–47, 1970.
- [11] G. B. Reggiori. Digital computer transformations for irregular line drawings. Technical Report 403-22, New York University, April 1972.
- [12] E. M. Coven and G. A. Hedlund. Sequences with minimal block growth. *Math. Systems Theory*, 7:138–153, 1973.
- [13] R. Brons. Linguistic methods for description of a straight line on a grid. *Computer Graphics Image Processing*, 2:48–62, 1974.
- [14] A. Rosenfeld. Digital straight line segments. *IEEE Trans. Computers*, 23:1264–1269, 1974.
- [15] C. Arcelli and A. Massarotti. Regular arcs in digital contours. *Computer Graphics Image Processing*, 4:339–360, 1975.
- [16] J. Rothstein and C. Weiman. Parallel and sequential specification of a context sensitive language for straight line grids. *Computer Graphics Image Processing*, 5:106–124, 1976.

- [17] A. Fam and J. Sklansky. Cellularly straight images and the Hausdorff metric. In *Proc. Conf. on Pattern Recognition and Image Processing*, pages 242–247, 1977.
- [18] T. Pavlidis. *Structural Pattern Recognition*. Springer, New York, 1977.
- [19] J. Sklansky and V. Gonzalez. Fast polygonal approximation of digitized curves. *Pattern Recognition*, 12:327–331, 1980.
- [20] A. Hübler, R. Klette, and K. Voss. Determination of the convex hull of a finite set of planar points within linear time. *EIK*, 17:121–140, 1981.
- [21] E. Creutzburg, A. Hübler and V. Wedler. On-line Erkennung digitaler Geradensegmente in linearer Zeit. In *Proc. GEOBILD'82*, Wiss. Beiträge der FSU Jena, pages 48–65, 1982.
- [22] E. Creutzburg, A. Hübler and V. Wedler. On-line recognition of digital straight line segments. In *Proc. 2nd Intl. Conf. AI and Inf. Control Systems of Robots*, Smolenice, Czechoslovakia, pages 42–46, 1982.
- [23] G. Dettori. An on-line algorithm for polygonal approximation of digitized plane curves. In *Proc. Intl. Conf. on Pattern Recognition*, pages 739–741, 1982.
- [24] C. E. Kim. On cellular straight line segments. *Computer Graphics Image Processing*, 18:369–391, 1982.
- [25] A. Rosenfeld and C. E. Kim. How a digital computer can tell whether a line is straight. *American Mathematical Monthly*, 89:230–235, 1982.
- [26] L. D. Wu. On the chain code of a line. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 4:347–353, 1982.
- [27] S. Shlien. Segmentation of digital curves using linguistic techniques. *Computer Vision, Graphics, Image Processing*, 22:277–286, 1983.
- [28] T. A. Anderson and C. E. Kim. Representation of digital line segments and their preimages. In *Proc. Intl. Conf. on Pattern Recognition*, pages 501–504, 1984.
- [29] L. Dorst and R. P. W. Duin. Spirograph theory: A framework for calculations on digitized straight lines. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 6:632–639, 1984.
- [30] T. A. Anderson and C. E. Kim. Representation of digital line segments and their preimages. *Computer Vision, Graphics, Image Processing*, 30:279–288, 1985.
- [31] H.-U. Döhler and P. Zamperoni. Compact contour codes for convex binary patterns. *Signal Processing*, 8: 23–39, 1985.
- [32] S. H. Y. Hung. On the straightness of digital arcs. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 7:1264–1269, 1985.

- [33] C. Ronse. A simple proof of Rosenfeld's characterization of digital straight line segments. *Pattern Recognition Letters*, 3:323–326, 1985.
- [34] R. Shoucri, R. Benesch, and S. Thomas. Note on the determination of a digital straight line from chain codes. *Computer Vision, Graphics, and Image Processing*, 29:133–139, 1985.
- [35] M. Lothaire (ed.). *Combinatorics on Words*, 2nd ed. Cambridge University Press, 1987.
- [36] E. Creutzburg, A. Hübler and O. Šýkora. Geometric methods for on-line recognition of digital straight segments. *Computers and Artificial Intelligence*, 7:253–276, 1988.
- [37] S. X. Li and M. H. Loew. Analysis and modeling of digitized straight-line segments. In *Proc. 9th Intl. Conf. on Pattern Recognition*, pages 294–296, 1988.
- [38] W. G. Kropatsch and H. Tockner. Detecting the straightness of digital curves in $O(N)$ steps. *Computer Vision, Graphics, and Image Processing*, 45:1–21, 1989.
- [39] A. Hübler. Diskrete Geometrie für die digitale Bildverarbeitung. Habil.-Schrift, Friedrich-Schiller-Universität Jena, 1989.
- [40] M. C. Irwin. Geometry of continued fractions. *Amer. Math. Monthly*, 96:696–703, 1989.
- [41] J. Koplowitz, M. Lindenbaum, and A. Bruckstein. The number of digital straight lines on an $N \times N$ grid. *IEEE Trans. Information Theory*, 36:192–197, 1990.
- [42] V. A. Kovalevsky. New definition and fast recognition of digital straight segments and arcs. In *Proc. Intl. Conf. on Pattern Recognition*, pages 31–34, 1990.
- [43] D. M. Acketa, J. D. Žunić. On the number of linear partitions of the (m, n) -grid. *Information Processing Letters*, 38:163–168, 1991.
- [44] A. M. Bruckstein. Self-similarity properties of digitized straight lines. *Contemp. Math.*, 119:1–20, 1991.
- [45] S. Chattopadhyay and P. P. Das. A new method of analysis for discrete straight lines. *Pattern Recognition Letters*, 12:747–755, 1991.
- [46] S. Dulucq and D. Gouyou-Beauchamps. Sur les facteurs des suites de Sturm. *Theoret. Comput. Sci.*, 71:381–400, 1991.
- [47] M. Lindenbaum and J. Koplowitz. A new parametrization of digital straight lines. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13:847–852, 1991.
- [48] F. Mignosi. On the number of factors of Sturmian words. *Theoret. Comput. Sci.*, 82:71–84, 1991.

- [49] J.-P. Reveillès. Géométrie discrète, calcul en nombres entiers et algorithmique. Thèse d'état, Univ. Louis Pasteur, Strasbourg, 1991.
- [50] A. W. M. Smeulders and L. Dorst. Decomposition of discrete curves into piecewise segments in linear time. *Contemporary Math.*, 119:169–195, 1991.
- [51] K. Voss. Coding of digital straight lines by continued fractions. *Comput. Artif. Intell.*, 10:75–80, 1991.
- [52] J. Berstel and M. Pocchiola. Random generation of finite Sturmian words. LIENS - 93 -8, Département de Mathématiques et d'Informatique, Ecole Normale Supérieure, LITP - Institute Blaise Pascal, 1993.
- [53] K. Voss. *Discrete Images, Objects, and Functions in Z^n* . Springer, Berlin, 1993.
- [54] I. Debled-Rennesson and J.-P. Reveillès. A linear algorithm for segmentation of digital curves. *Intl. J. Pattern Recognition Artificial Intelligence*, 9:635–662, 1995.
- [55] J. Yuan and C. Y. Suen. An optimal $O(n)$ algorithm for identifying line segments from a sequence of chain codes. *Pattern Recognition*, 28:635–646, 1995.
- [56] J. Françon, J.-M. Schramm, and M. Tajine. Recognizing arithmetic straight lines and planes. In *Proc. DGCI*, LNCS 1176, pages 141–150. Springer, Berlin, 1996.
- [57] R. Klette and B. Yip. The length of digital curves. *Machine Graphics & Vision*, 9:673–703, 2000 (extended version of: R. Klette, V. V. Kovalevsky, B. Yip. Length estimation of digital curves. In *Proc. Vision Geometry VIII*, SPIE 3811, pages 117–129).
- [58] R. Klette. Cell complexes through time. In *Proc. Vision Geometry IX*, SPIE 4117, pages 134–145, 2000.
- [59] M. Lothaire (ed.). *Algebraic Combinatorics on Words*. Cambridge University Press, to appear.