Approximation of Digital Surfaces by a Hierarchical Set of Planar Patches ⋆

Jocelyn Meyron and Tristan Roussillon

Univ Lyon, INSA Lyon, CNRS, UCBL, LIRIS,
UMR5205, F-69622 Villeurbanne, France
{jocelyn.meyron,tristan.roussillon}@liris.cnrs.fr

Abstract. We show that the plane-probing algorithms introduced in Lachaud et al. (J. Math. Imaging Vis., 59, 1, 23–39, 2017), which compute the normal vector of a digital plane from a starting point and a set-membership predicate, are closely related to a three-dimensional generalization of the Euclidean algorithm. In addition, we show how to associate with the steps of these algorithms generalized substitutions, i.e., rules that replace square faces by unions of square faces, to build finite sets of elements that periodically generate digital planes. This work is a first step towards the incremental computation of a hierarchy of pieces of digital plane that locally fit a digital surface.

Keywords: Digital planes · Multi-dimensional continued-fraction algorithms · Generalized substitutions · Plane-probing algorithms.

1 Introduction

Digital geometry mainly deals with sets of discrete elements considered to be digitized versions of Euclidean objects. A digital surface may be seen as a mesh of unit square faces whose vertices have integer coordinates. A challenge is to decompose digital surfaces into patches, such as pieces of digital planes.

A digital plane has been analytically defined as a set of points with integer coordinates lying between two parallel planes. Given a finite point set, one can decide whether this set belongs to a digital plane or not in linear time using linear programming. For a review on digital planarity, see [7]. However, a linear programming solver does not help so much for the analysis of digital surfaces, because one does not know which point set should be tested to obtain patches that approximates the tangent plane of the underlying surface.

In order to cope with this problem, *plane-probing* algorithms have been developed [17, 18]. Their main feature is to decide on-the-fly how to probe a given point set and locally align a triangle with it. However, if probing for points in a sparse way is perfect for digital planes, it is not enough for non-convex parts where the triangles may jump over holes or stab the digital surface. Therefore, that approach also requires to associate pieces of digital planes to the triangles and check whether they fit the digital surface or not.

⋆ This work has been funded by PARADIS ANR-18-CE23-0007-01 research grant.

In this paper, we make a first step towards the incremental generation of a hierarchy of pieces of digital plane, which can be used during the execution of a plane-probing algorithm (see Fig. 1). To do that, we take advantage of the combinatorial properties of digital planes.

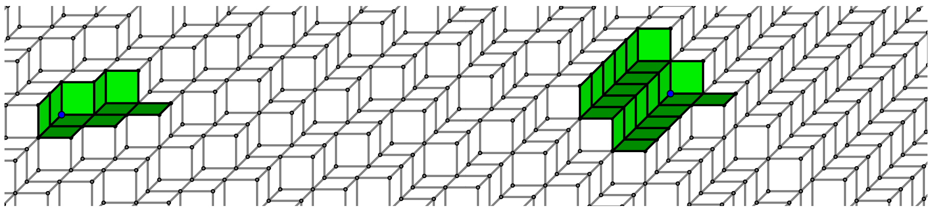


Fig. 1: Local approximation of a digital sphere of radius 63 by planar patches (in green) from two starting points (in blue). The implementation combines the plane-probing algorithm H [18] with the generation method of Section 3.

The particular case of digital lines has been studied for a long time in different contexts and has led to many applications. One key result is that digital lines are hierarchical point sets whose structure is exactly described by the continued fraction expansion of their slope and strongly relies on the Euclidean algorithm. For a survey on digital straightness, see [15].

There has been much effort done in order to find similar results in three dimensions despite the lack of a canonical algorithm and the diversity of existing generalizations of the Euclidean algorithm. Some combinatorial results, involving symmetries, piece exchanges and flips, have been stated thanks to an appropriate representation of digital planes [16]. However, most of other related works depend on a multi-dimensional generalization of the Euclidean algorithm such as *Brun*, e.g., [4], *Jacobi-Perron*, e.g., [6], *Fully substractive*, e.g. [3], or a mix of several of them [13]. Those algorithms have been used to generate digital planes from a normal vector. There are two different but closely related construction schemes in the literature. The first one is based on union and translation of point sets [5, 10, 13]. It has been used mostly to construct the thinnest digital plane that is connected [3, 5, 8, 9]. The second one is based on a description of standard digital planes as unions of square faces and uses rules that replace square faces by unions of square faces. Since the pioneer work of Arnoux and Ito [2], that formalism has been used for instance in [4–6, 11, 12]. Both construction schemes incrementally generate sets so that the current set will be included in the next one, but suffer from topological and geometrical limitations. This work is based on the second scheme that generates few elements in comparison with the first scheme.

In Section 2, we show how one can relate plane-probing algorithms to a three-dimensional generalization of the Euclidean algorithm. In Section 3, we introduce the generalized substitutions, describe our approach and examine several properties on the generated sets. Finally, the paper ends with some concluding remarks and perspectives.

2 Generalization of the Euclidean Algorithm

Let $\{\mathbf{e}_1, \ldots, \mathbf{e}_d\}$ be the canonical basis of $\mathbb{R}^d$. We denote $\mathbf{0}$ the origin and $\mathbf{1} = \sum_{i=1}^{d} \mathbf{e}_i$ the vector with all coordinates equal to 1 whatever the dimension d. We are interested below in vectors with coprime positive integer coordinates, i.e., in the set $\mathcal{V}^d := \{(v_1, \ldots, v_d) \mid \forall i, v_i \in \mathbb{N} \setminus \{0\} \text{ and } \gcd(v_1, \ldots, v_d) = 1\}$.

2.1 Three-dimensional Euclidean Algorithms

In its additive form, the Euclidean algorithm can simply be expressed as: "for a rational number represented by a pair of integers: subtract the smaller element to the larger one and repeat". It can be also expressed as a map $\Pi : \mathcal{V}^2 \to \mathcal{V}^2$ such that $\Pi(\mathbf{1}) = \mathbf{1}$ and

$$\text{for any } \left(\begin{smallmatrix} a \\ b \end{smallmatrix}\right) \in \mathcal{V}^2 \setminus \{\mathbf{1}\},\ \Pi\left(\left(\begin{smallmatrix} a \\ b \end{smallmatrix}\right)\right) = \begin{cases} \left(\begin{smallmatrix} 1 & -1 \\ 0 & 1 \end{smallmatrix}\right)\left(\begin{smallmatrix} a \\ b \end{smallmatrix}\right) \text{ if } a > b, \\ \left(\begin{smallmatrix} 1 & 0 \\ -1 & 1 \end{smallmatrix}\right)\left(\begin{smallmatrix} a \\ b \end{smallmatrix}\right) \text{ otherwise.} \end{cases}$$

Note that for all $\mathbf{v} \in \mathcal{V}^2$, there exists a non-negative integer N such that N successive applications of Π returns $\mathbf{1}$, i.e., $\Pi^N(\mathbf{v}) = \mathbf{1}$. In dimension 2, it is clear to decide which number has to be subtracted to the other, whereas in dimension 3, i.e., for triplets, it is not the case, hence the diversity of existing generalizations. Several existing algorithms stick to a convention, which may lead to ambiguities or to a null coordinate in case of ties:

- *Brun* consists in subtracting the second largest entry to the largest one; $(1, 2, 2)$ may lead to either $(1, 0, 2)$ or to $(1, 2, 0)$.
- *Farey* consists in subtracting the smallest entry to the second largest one; $(1, 1, 2)$ may lead to either $(0, 1, 2)$ or to $(1, 0, 2)$. In both cases, the numbers cannot be reduced further.
- *Selmer* consists in subtracting the smallest entry to the largest one, $(1, 2, 2)$ may lead to either $(1, 1, 2)$ or to $(1, 2, 1)$. While the numbers are not all equal, no zero appears.

We generalize below that last algorithm with the following two definitions:

Definition 1. *Let $\mathcal{T}$ be the set of all permutations over $\{1, 2, 3\}$. For a permutation $\tau \in \mathcal{T}$, let $\mathbf{U}_{\tau(1),\tau(2),\tau(3)}$ be a 3×3 matrix having -1 at the intersection between the $\tau(1)$-th row and the $\tau(2)$-th column and the same entries as $\mathbf{I}_3$, i.e. the 3×3 identity matrix, elsewhere. Let $\mathcal{U}$ be the set $\{\mathbf{U}_{\tau(1),\tau(2),\tau(3)} \mid \tau \in \mathcal{T}\}$.*

$\mathbf{U}_{1,2,3}$	$\mathbf{U}_{1,3,2}$	$\mathbf{U}_{2,3,1}$	$\mathbf{U}_{2,1,3}$	$\mathbf{U}_{3,1,2}$	$\mathbf{U}_{3,2,1}$
$\begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$

Note that the elements of $\mathcal{U}$ are matrices with determinant 1.

Definition 2. *A three-dimensional Euclidean algorithm is a map* $\Pi : \mathcal{V}^3 \to \mathcal{V}^3$ *such that* $\Pi(\mathbf{1}) = \mathbf{1}$ *and for any* $\mathbf{w} \in \mathcal{V}^3 \setminus \{\mathbf{1}\}$*, there is a matrix* $\mathbf{U} \in \mathcal{U}$ *satisfying* $\Pi(\mathbf{w}) = \mathbf{U}\mathbf{w}$.

Note that Selmer is a three-dimensional Euclidean algorithm according to Definition 2.

The following proposition shows that the repeated application of Π always brings a vector $\mathbf{v} \in \mathcal{V}^3$ to $\mathbf{1}$, exactly as with the Euclidean algorithm in 2D.

Proposition 1. *Let* Π *be a three-dimensional Euclidean algorithm. For all* $\mathbf{v} \in \mathcal{V}^3$*, there exists a non-negative integer* N *such that* $\Pi^N(\mathbf{v}) = \mathbf{1}$ *and* $N \leq \|\mathbf{v}\|_1 - 3$.

Proof. For all $n \geq 0$ for which the coordinates of $\Pi^n(\mathbf{v})$ are not all equal, by definition of Π, there exists a matrix $\mathbf{U} \in \mathcal{U}$ such that $\Pi^{n+1}(\mathbf{v}) = \mathbf{U}\,\Pi^n(\mathbf{v}) \in \mathcal{V}^3$. Furthermore, we have $\|\,\Pi^{n+1}(\mathbf{v})\|_1 < \|\,\Pi^n(\mathbf{v})\|_1$ in that case.

If there is $N \geq 1$ for which the coordinates of $\Pi^N(\mathbf{v})$ are all equal, there is a strictly positive integer h such that $\Pi^N(\mathbf{v}) = h\mathbf{1}$. However, $\Pi^N(\mathbf{v}) = \mathbf{M}\mathbf{v}$, where $\mathbf{M}$ is a product of elementary matrices belonging to $\mathcal{U}$. It follows that $\mathbf{M}$ has determinant 1 and is therefore invertible. We have $\mathbf{M}\mathbf{v} = h\mathbf{1} \Leftrightarrow \mathbf{v} = h\mathbf{M}^{-1}\mathbf{1}$ and since the coordinates of $\mathbf{v}$ are coprime, we obtain $h = 1$ and $\Pi^N(\mathbf{v}) = \mathbf{1}$.

As a consequence, $(\|\,\Pi^n(\mathbf{v})\|_1)_{n=0,\dots,N}$ is a strictly decreasing integer sequence from $\|\mathbf{v}\|_1$ to $\|\,\Pi^N(\mathbf{v})\|_1 = 3$, which concludes the proof. □

We focus now on the following finite sequences:

Definition 3. *A sequence of matrices* $(\mathbf{U}_n)_{0 \leq n \leq N}$ *is* valid *iff* $\mathbf{U}_0 = \mathbf{I}_3$ *and every* $\mathbf{U}_n$*,* $n \in \{1, \dots, N\}$*, is in* $\mathcal{U}$*. In addition, a valid sequence* reduces *a vector* $\mathbf{v} \in \mathcal{V}^3$ *iff for all* $n \in \{0, \dots, N-1\}$*,* $\mathbf{U}_n \cdots \mathbf{U}_0\mathbf{v} \in \mathcal{V}^3 \setminus \{\mathbf{1}\}$ *and* $\mathbf{U}_N \cdots \mathbf{U}_0\mathbf{v} = \mathbf{1}$*. In that case, we set* $\mathbf{v}_n := \mathbf{U}_n \cdots \mathbf{U}_0\mathbf{v}$ *and* $\mathbf{a}_n := \mathbf{U}_0^{-1} \cdots \mathbf{U}_n^{-1}\mathbf{1}$.

Let $\langle \cdot, \cdot \rangle$ stand for the usual scalar product on $\mathbb{R}^3$. Note that for all $\mathbf{x} \in \mathbb{R}^3$, $\langle \mathbf{x}, \mathbf{e}_i \rangle$ is equal to the i-th coordinate of $\mathbf{x}$. The following proposition shows several properties of the above-defined sequences:

Proposition 2. *We have* $\mathbf{a}_N = \mathbf{v}_0$ *and for each* $n \in \{0, \dots, N\}$:

(i) for each $i \in \{1, 2, 3\}$*,* $\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T\mathbf{e}_i, \mathbf{v}_0 \rangle$ *is the* i*-th coordinate of* $\mathbf{v}_n$*,*
(ii) for each $i \in \{1, 2, 3\}$*,* $\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T\mathbf{e}_i, \mathbf{a}_n \rangle = 1$*,*
(iii) the differences $(\mathbf{U}_0^T \cdots \mathbf{U}_n^T(\mathbf{e}_2 - \mathbf{e}_1), \mathbf{U}_0^T \cdots \mathbf{U}_n^T(\mathbf{e}_3 - \mathbf{e}_2))$ *form a basis of the lattice* $\{\mathbf{x} \in \mathbb{Z}^3 \mid \langle \mathbf{x}, \mathbf{a}_n \rangle = 0\}$.

Proof. By Definition 3, we have $\mathbf{v}_N = \mathbf{U}_N \cdots \mathbf{U}_0\mathbf{v}_0 = \mathbf{1}$, which is equivalent to $\mathbf{v}_0 = \mathbf{U}_0^{-1} \cdots \mathbf{U}_N^{-1}\mathbf{1} = \mathbf{a}_N$. For (i), $\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T\mathbf{e}_i, \mathbf{v}_0 \rangle = \langle \mathbf{e}_i, \mathbf{U}_n \cdots \mathbf{U}_0\mathbf{v}_0 \rangle = \langle \mathbf{e}_i, \mathbf{v}_n \rangle$. For (ii), $\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T\mathbf{e}_i, \mathbf{a}_n \rangle = \langle \mathbf{e}_i, \mathbf{U}_n \cdots \mathbf{U}_0\mathbf{a}_n \rangle = \langle \mathbf{e}_i, \mathbf{1} \rangle = 1$. Finally, for the last item, note that $\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T(\mathbf{e}_2 - \mathbf{e}_1), \mathbf{a}_n \rangle = 0$ and $\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T(\mathbf{e}_3 - \mathbf{e}_2), \mathbf{a}_n \rangle = 0$ by (ii). The fact that $(\mathbf{U}_0^T \cdots \mathbf{U}_n^T(\mathbf{e}_2 - \mathbf{e}_1), \mathbf{U}_0^T \cdots \mathbf{U}_n^T(\mathbf{e}_3 - \mathbf{e}_2))$ form a *basis* of the lattice $\{\mathbf{x} \in \mathbb{Z}^3 \mid \langle \mathbf{x}, \mathbf{a}_n \rangle = 0\}$ comes from the fact that the matrix $\mathbf{U}_0^T \cdots \mathbf{U}_n^T$ has determinant 1. □

Fig. 2 illustrates the action of the matrices $\mathbf{U}_N \cdots \mathbf{U}_0$ and $\mathbf{U}_0^{-1} \cdots \mathbf{U}_N^{-1}$ in (a), and $\mathbf{U}_0^T \cdots \mathbf{U}_N^T$ in (b). Note that for each $n \in \{0, \dots, N\}$, $\mathbf{U}_0^T \cdots \mathbf{U}_n^T$ transforms the basis $\{\mathbf{e}_i \mid i \in \{1,2,3\}\}$ to $\{\mathbf{U}_0^T \cdots \mathbf{U}_n^T \mathbf{e}_i \mid i \in \{1,2,3\}\}$. In other words, it deforms an orthonormal basis to a basis that is more and more aligned with the plane of normal $\mathbf{v} = \mathbf{v}_0$ because the quantities $\{\langle \mathbf{U}_0^T \cdots \mathbf{U}_n^T \mathbf{e}_i, \mathbf{v}_0 \rangle \mid i \in \{1,2,3\}\}$ are smaller and smaller by Proposition 2, item (i).

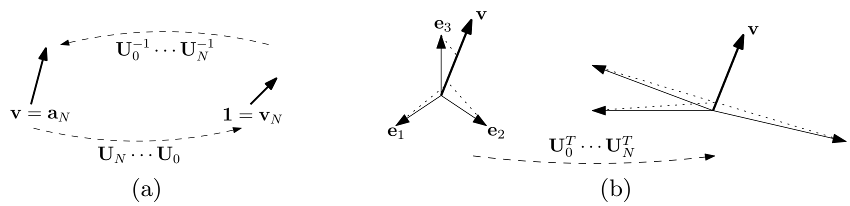


Fig. 2: Geometrical interpretation of the matrices $(\mathbf{U}_n)_{0 \le n \le N}$. In (b), we have implicitly represented the scalar projection of the basis vectors in the direction of $\mathbf{v}$ with the help of dotted lines. In the rightmost figure, the scalar products $\{\langle \mathbf{U}_0^T \cdots \mathbf{U}_N^T \mathbf{e}_i, \mathbf{v} \rangle \mid i \in \{1,2,3\}\}$ are equal to the coordinates of $\mathbf{v}_N$ and are thus all equal to 1.

2.2 Relation With Plane-Probing Algorithms

A *digital plane* is formally defined by a normal $\mathbf{v} \in \mathbb{Z}^3 \setminus \{\mathbf{0}\}$ and a position $\mu \in \mathbb{Z}$ as follows:

$$\mathcal{P}_{\mu,\mathbf{v}} := \{\mathbf{x} \in \mathbb{Z}^3 \mid \mu \le \mathbf{x} \cdot \mathbf{v} < \mu + \|\mathbf{v}\|_1\}. \tag{1}$$

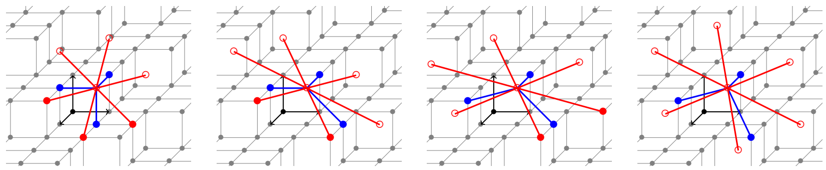
In what follows, we assume w.l.o.g. that $\mathbf{v} \in \mathcal{V}^3$ and $\mu = 0$. Given a digital plane $\mathcal{P} \in \{\mathcal{P}_{0,\mathbf{v}} \mid \mathbf{v} \in \mathcal{V}^3\}$ of unknown normal vector a *plane-probing algorithm* computes the normal vector $\mathbf{v}$ of $\mathcal{P}$ by sparsely probing it with the predicate "is $\mathbf{x}$ in $\mathcal{P}$?".

We now describe the plane-probing algorithm H introduced in [18]. The state of the algorithm is a basis of three vectors, which can be stored in a 3×3 matrix, denoted by the letter $\mathbf{B}$, with the index of the step as a subscript. At initialization, $\mathbf{B}_0$ is set to identity. In order to explain how $\mathbf{B}_{n+1}$ is computed from $\mathbf{B}_n$ at a step $n \ge 0$ of the algorithm, let us introduce the following set of differences (see Fig. 3 for an illustration):

$$\mathcal{D}_n := \{\mathbf{B}_n(-\mathbf{e}_{\tau(1)} + \mathbf{e}_{\tau(2)}) \mid \tau \in \mathcal{T}\}.$$

If $\mathcal{P} \cap \{\mathbf{1} + \mathbf{d} \mid \mathbf{d} \in \mathcal{D}_n\} = \emptyset$ (see Fig. 3 on the right), the algorithm halts. Otherwise, there exists a permutation $\tau \in \mathcal{T}$ such that:

1. $\mathbf{1} + \mathbf{B}_n(-\mathbf{e}_{\tau(1)} + \mathbf{e}_{\tau(2)}) \in \mathcal{P}$ (see Fig. 3),
2. the sphere passing by $\{\mathbf{1} + \mathbf{B}_n \mathbf{e}_i \mid i \in \{1,2,3\}\}$ and the point $\mathbf{1} + \mathbf{B}_n(-\mathbf{e}_{\tau(1)} + \mathbf{e}_{\tau(2)})$ does not include in its interior any other point of $\mathcal{P} \cap \{\mathbf{1} + \mathbf{d} \mid \mathbf{d} \in \mathcal{D}_n\}$.



$\mathbf{B}_0 = \mathbf{I}_3$	$\mathbf{B}_1 = \mathbf{B}_0(\mathbf{U}_{3,2,1})^T$	$\mathbf{B}_2 = \mathbf{B}_1(\mathbf{U}_{2,1,3})^T$	$\mathbf{B}_3 = \mathbf{B}_2(\mathbf{U}_{3,1,2})^T$
$\mathbf{v} = \mathbf{v}_0 = (1, 2, 4)$	$\mathbf{v}_1 = (1, 2, 2)$	$\mathbf{v}_2 = (1, 1, 2)$	$\mathbf{v}_3 = (1, 1, 1)$
$\mathbf{a}_0 = (1, 1, 1)$	$\mathbf{a}_1 = (1, 1, 2)$	$\mathbf{a}_2 = (1, 2, 3)$	$\mathbf{a}_3 = (1, 2, 4)$

Fig. 3: Execution of the algorithm H on a digital plane of normal vector $(1, 2, 4)$. For $n \in \{0, 1, 2, 3\}$, the column vectors of $\mathbf{B}_n$ (resp. elements of $\mathcal{D}_n$) are depicted with blue (resp. red) arrows that points to $\mathbf{1}$. The end points are depicted with disks (resp. circles) if they belong (resp. do not belong) to the digital plane.

From that permutation τ, we set $\mathbf{B}_{n+1} := \mathbf{B}_n(\mathbf{U}_{\tau(1),\tau(2),\tau(3)})^T$. In other words, the $\tau(1)$-th column of $\mathbf{B}_{n+1}$, i.e., $\mathbf{B}_{n+1}\mathbf{e}_{\tau(1)}$, is equal to the difference $\mathbf{B}_n(\mathbf{e}_{\tau(1)} - \mathbf{e}_{\tau(2)})$, while the other columns are identical in $\mathbf{B}_{n+1}$ and $\mathbf{B}_n$.

Note that there may be as much as three points in $\mathcal{P} \cap \{\mathbf{1} + \mathbf{d} \mid \mathbf{d} \in \mathcal{D}_n\}$ because $\mathbf{1}$ does not belong to $\mathcal{P}$ and $\mathcal{D}_n$ contains three distinct pairs of vectors of opposite sign. The in-sphere criterion generally provides a way of selecting one of those points and thus one elementary matrix. If several points are in a cospherical position, one can resort to a lexicographic order so that the algorithm is defined without any ambiguity.

That algorithm is very similar to a three-dimensional Euclidean algorithm, but does not exactly correspond to Definition 2, because two consecutive steps are not independent. However, it produces a sequence of matrices $(\mathbf{U})_{0 \le n \le N}$ such that $\mathbf{U}_0 = \mathbf{I}_3$ and $\mathbf{B}_n^T = \mathbf{B}_{n-1}^T \mathbf{U}_n$ for $n \in \{1, \dots, N\}$. That sequence is not only valid by construction, but also reduces the normal vector $\mathbf{v}$ (Definition 3):

- $\forall n \in \{0, \dots, N-1\}$, $\mathbf{U}_n \cdots \mathbf{U}_0 \mathbf{v} \in \mathcal{V}^3 \setminus \{\mathbf{1}\}$ by item 1 of [18, Lemma 1],
- $\mathbf{U}_N \cdots \mathbf{U}_0 \mathbf{v} = \mathbf{1}$ by [18, Theorem 2].

As a consequence, Proposition 2 also applies. For instance, $\mathbf{a}_N = \mathbf{v}$ [18, Corollary 4] means that the algorithm, which computes $\mathbf{a}_N$ only from a predicate "is $\mathbf{x}$ in $\mathcal{P}$?", is indeed able to retrieve the normal vector $\mathbf{v}$ of $\mathcal{P}$ (see also Fig. 3).

What makes the algorithm H very different from the classical three-dimensional Euclidean algorithms like Selmer is twofold: on one hand, it uses the past results for each new computation and on the other hand, it uses a geometrical criterion to decide which elementary matrix has to be applied. That is why less elongated patterns are obtained with that algorithm in the next section and for instance in Fig. 5. In addition, since it uses only a set-membership predicate, it can also be applied to arbitrary digital surfaces with slight changes as shown in Fig. 1 and [18, Section 5].

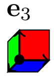
Lastly, note that an update step in the algorithm R, also introduced in [18], can be decomposed into elementary steps that are update steps in the algorithm H [17, Section 3.5]. Therefore, we have the same results for the algorithm R.

3 Pattern Generation with Generalized Substitutions

The goal of this section is to show how one can use a sequence of matrices that reduces a vector $\mathbf{v} \in \mathcal{V}^3$ to generate a piece of digital plane of normal $\mathbf{v}$.

The generation method is based on a description of standard digital planes as union of faces. For $\mathbf{x} \in \mathbb{Z}^3$ and $i \in \{1,2,3\}$, we define the *pointed face* of type i and origin $\mathbf{x}$ as the following subset of $\mathbb{R}^3$: $(\mathbf{x}, i^*) := \left\{\mathbf{x} + \sum_{j \neq i} \lambda_j \mathbf{e}_j, \lambda_j \in [0,1]\right\}$.

$\mathbf{e}_3$

$\mathbf{e}_1$  $\mathbf{e}_2$ shows $(\mathbf{0}, 1^*)$ in red, $(\mathbf{0}, 2^*)$ in green and $(\mathbf{0}, 3^*)$ in blue.

We will use the following notation for translations of faces: if $(\mathbf{x}, i^*)$ is a face and $\mathbf{y}$ is a vector, then $(\mathbf{x}, i^*) + \mathbf{y} := (\mathbf{x} + \mathbf{y}, i^*)$, which extends in a natural way to union of faces.

For a vector $\mathbf{v} \in \mathcal{V}^3$, a *stepped plane* is defined as an infinite set of pointed faces where each face $(\mathbf{x}, i^*)$ verifies $0 \leq \langle \mathbf{x}, \mathbf{v} \rangle < \langle \mathbf{e}_i, \mathbf{v} \rangle$ (see for instance [14, Definition 1.2.1]). That way, the points of a digital plane, as defined in the previous section, are the vertices of the faces of a stepped plane. By abuse of notation, we will use $\mathcal{P}_{0,\mathbf{v}}$ or simply $\mathcal{P}$ to denote both a digital plane and a stepped plane in the following.

3.1 Substitutions and Generalized Substitutions

In this subsection, we first recall the definition of generalized substitutions and then show how to use them to generate a stepped plane of normal $\mathbf{v}$.

We consider a 3-letter alphabet $\mathcal{A} := \{1,2,3\}$. A word is an element of the free monoid $\mathcal{A}^\star$ generated by $\mathcal{A}$. The empty word is denoted by ϵ and the concatenation operation is denoted by $\cdot$ or is left implicit. A *substitution* σ over $\mathcal{A}$ is a non-erasing endomorphism of $\mathcal{A}^*$, completely defined by its image on the letters of $\mathcal{A}$ by the relation $\sigma(w \cdot w') = \sigma(w) \cdot \sigma(w')$. The *abelianization mapping* $l : \mathcal{A}^* \to \mathbb{N}^3$ is such that $l(w) = (|w|_1, |w|_2, |w|_3)$, where $|w|_i$ denotes the number of occurrences of the letter i in w. The *incidence matrix* $\mathbf{M}_\sigma$ of σ is the 3×3 matrix whose i-th column is $l(\sigma(i))$ for every $i \in \mathcal{A}$. We assume that all the substitutions we work with are unimodular, i.e., such that $\det(\mathbf{M}_\sigma) = \pm 1$.

Furthermore, we define the following set:

$$\mathcal{S}_\sigma^i := \{(s,j) \in \mathcal{A}^* \times \mathcal{A} \mid j \in \mathcal{A},\ i \cdot s \text{ is a suffix of } \sigma(j)\}. \quad (2)$$

To obtain $\mathcal{S}_\sigma^i$, one splits the words $\{\sigma(j) \mid j \in \mathcal{A}\}$ at each occurrence of the letter i. For each decomposition, we keep j as well as the suffix s located just after i.

Example 1. For σ: $1 \mapsto 1$, $2 \mapsto 21$, $3 \mapsto 32$, $\mathcal{S}_\sigma^2 = \{(1,2), (\epsilon, 3)\}$.

The *generalized substitution* of a pointed face $(\mathbf{x}, i^*)$ is [14, Definition 1.2.3]:

$$E_1^*(\sigma)(\mathbf{x}, i^*) := \bigcup_{(s,j) \in \mathcal{S}_\sigma^i} (\mathbf{M}_\sigma^{-1}(\mathbf{x} + l(s)), j^*). \quad (3)$$

Example 2. For σ: $1 \mapsto 1, 2 \mapsto 21, 3 \mapsto 32$, $E_1^*(\sigma)(\mathbf{0}, 2^*) = \{(\mathbf{M}_\sigma^{-1}\mathbf{e}_1, 2^*), (\mathbf{0}, 3^*)\}$.

We extend this definition to unions of faces: $E_1^*(\sigma)(\mathcal{F}_1 \cup \mathcal{F}_2) := E_1^*(\sigma)(\mathcal{F}_1) \cup E_1^*(\sigma)(\mathcal{F}_2)$.

This setting is just fine for our purpose, but note that generalized substitutions allow a more general setting by considering the free group – instead of the free monoid – generated by $\mathcal{A}$ (see for instance [1, 6, 12]).

One of the main results about generalized substitutions and stepped planes is that $E_1^*(\sigma)$ preserves stepped planes ([14, Proposition 1.2.4, item (3)] or [11, Theorem 1] for a proof). Indeed, for any substitution σ whose incidence matrix $\mathbf{M}_\sigma$ is unimodular,

$$E_1^*(\sigma)(\mathcal{P}_{0,\mathbf{v}}) = \mathcal{P}_{0,\mathbf{M}_\sigma^T\mathbf{v}}. \tag{4}$$

We focus now on specific finite sequences of substitutions defined below:

Definition 4. *A sequence of substitutions* $(\sigma_n)_{0\leq n\leq N}$ *is* admissible *iff the sequence* $\left((\mathbf{M}_{\sigma_n}^T)^{-1}\right)_{0\leq n\leq N}$ *reduces the vector* $\mathbf{M}_{\sigma_0}^T \dots \mathbf{M}_{\sigma_N}^T \mathbf{1}$ *(See Definition 3 to recall the notion of reduction). In that case, we also set* $\mathbf{a}_n := \mathbf{M}_{\sigma_0}^T \dots \mathbf{M}_{\sigma_n}^T \mathbf{1}$ *for all* $n \in \{0, \dots, N\}$ *in accordance with Definition 3 and the relation* $\mathbf{U}_n^{-1} = \mathbf{M}_{\sigma_n}^T$.

One can obtain 2^N admissible sequences of substitutions from any sequence of matrices that reduces a given vector. Indeed, since each matrix $\mathbf{U}$ of the sequence belongs to $\mathcal{U}$, there are exactly two substitutions σ such that $\mathbf{M}_\sigma = (\mathbf{U}^T)^{-1}$. For instance, the inverse $\left(\begin{smallmatrix}1&0&0\\0&1&0\\0&1&1\end{smallmatrix}\right)$ of the matrix $\left(\begin{smallmatrix}1&0&0\\0&1&0\\0&-1&1\end{smallmatrix}\right)$ is the incidence matrix of the two substitutions: $1 \mapsto 1, 2 \mapsto 23, 3 \mapsto 3$ and $1 \mapsto 1, 2 \mapsto 32, 3 \mapsto 3$. This choice has an impact on the geometry and topology of the generated patterns (see Section 3.3).

For two substitutions σ', σ'', we denote by $\circ$ their composition: $(\sigma' \circ \sigma'')(w) := \sigma'(\sigma''(w))$. Note that $(\mathbf{M}_{\sigma'\circ\sigma''})^T = \mathbf{M}_{\sigma''}^T \mathbf{M}_{\sigma'}^T$ and $E_1^*(\sigma') \circ E_1^*(\sigma'') = E_1^*(\sigma'' \circ \sigma')$ [14, Proposition 1.2.4, item (1)]. To save space, we set $\sigma_{i\cdots 0} := \sigma_i \circ \dots \circ \sigma_0$ for $1 \leq i \leq N$. The following theorem is a direct consequence of (4):

Theorem 1. *Let* $(\sigma_n)_{0\leq n\leq N}$ *be an admissible sequence of substitutions (see Definition 4). For all* $n \in \{0, \dots, N\}$, $E_1^*(\sigma_{n\cdots 0})(\mathcal{P}_{0,\mathbf{1}}) = \mathcal{P}_{0,\mathbf{a}_n}$

Proof. Applying (4) and using the definition of $\mathbf{a}_n$, we get:

$$E_1^*(\sigma_{n\cdots 0})(\mathcal{P}_{0,\mathbf{1}}) = \mathcal{P}_{0,\mathbf{M}_{\sigma_0}^T\cdots\mathbf{M}_{\sigma_n}^T\mathbf{1}} = \mathcal{P}_{0,\mathbf{a}_n} \qquad \square$$

3.2 Generation Method and Properties of the Patterns

In this section, we do not apply generalized substitutions on the whole stepped plane $\mathcal{P}_{0,\mathbf{1}}$ as in Theorem 1. We apply them only on the lower unit cube composed of the three pointed faces $(\mathbf{0}, 1^*)$, $(\mathbf{0}, 2^*)$ and $(\mathbf{0}, 3^*)$ because it periodically generates $\mathcal{P}_{0,\mathbf{1}}$ and is also included in any stepped plane of normal $\mathbf{v} \in \mathcal{V}^3$. The result is a finite set of pointed faces that we call *pattern*.

Definition 5 (Pattern). *Let* $(\sigma_n)_{0\leq n\leq N}$ *be an admissible sequence of substitutions (see Definition 4). Let* $\mathcal{W}_0$ *be the lower unit cube* $\cup_{i\in\mathcal{A}}(\mathbf{0}, i^*)$ *and for all* $n \in \{1, \dots, N\}$, *let* $\mathcal{W}_n$ *be the image of* $\mathcal{W}_0$ *by* $E_1^*(\sigma_{n\cdots 0})$, *i.e.,* $\mathcal{W}_n := E_1^*(\sigma_{n\cdots 0})(\mathcal{W}_0)$.

Fortunately, there exists a way to incrementally generate $\mathcal{W}_n$ from $\mathcal{W}_{n-1}$ in the manner of a union-translation scheme. This is the process we use in practice.

Theorem 2. *Let $(\sigma_n)_{0\leq n\leq N}$ be an admissible sequence of substitutions (see Definition 4). We have for all $n \in \{1,\ldots,N\}$ and for all $i \in \mathcal{A}$,*

$$E_1^*(\sigma_{n\cdots 0})(\mathbf{0}, i^*) = \bigcup_{(s,j)\in\mathcal{S}^i_{\sigma_n}} \left(\mathbf{M}_{\sigma_0}^{-1}\cdots\mathbf{M}_{\sigma_n}^{-1} l(s) + E_1^*(\sigma_{n-1\cdots 0})(\mathbf{0}, j^*)\right).$$

The proof, based on (2) and (3), is given in appendix.

As shown in Fig. 4, for $n \geq 1$, σ_n describes how the parts of $\mathcal{W}_n$ relate to the ones of $\mathcal{W}_{n-1}$. As an example, let us consider $E_1^*(\sigma_{3\cdots 1})(\mathbf{0}, 1^*)$, which is the red part of $\mathcal{W}_3$. It has been obtained as the union of two parts of $\mathcal{W}_2$: $E_1^*(\sigma_{2\cdots 1})(\mathbf{0}, 1^*)$ (in red) and $E_1^*(\sigma_{2\cdots 1})(\mathbf{0}, 3^*)$ (in blue), because letter 1 belongs to both $\sigma_3(1)$ and $\sigma_3(3)$ and is also in the last position, which means with no suffixes and thus no translations.

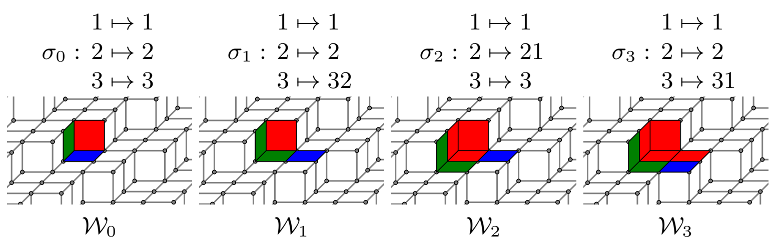


Fig. 4: The substitutions σ_1, σ_2 and σ_3 have been obtained from the reduction of $(1, 2, 4)$ using algorithm H (see also Fig. 3). The images of $(\mathbf{0}, 1^*)$, $(\mathbf{0}, 2^*)$ and $(\mathbf{0}, 3^*)$ by E_1^* are displayed respectively in red, green and blue.

The following theorem gathers several properties of patterns:

Theorem 3. *Let $(\sigma_n)_{0\leq n\leq N}$ be an admissible sequence of substitutions (see Definition 4). The following properties hold on the patterns defined in Definition 5:*

(i) $\forall n \in \{1,\ldots,N\}$, $\mathcal{W}_{n-1} \subset \mathcal{W}_n$ *and* $\forall n \in \{0,\ldots,N\}$, $\mathcal{W}_n \subset \mathcal{P}_{0,\mathbf{a}_N}$,
(ii) $\forall n \in \{1,\ldots,N\}$, $\mathcal{W}_n$ *periodically generates* $\mathcal{P}_{0,\mathbf{a}_n}$ *with period vectors:*

$$\mathbf{M}_{\sigma_0}^{-1}\cdots\mathbf{M}_{\sigma_n}^{-1}(\mathbf{e}_2 - \mathbf{e}_1) \text{ and } \mathbf{M}_{\sigma_0}^{-1}\cdots\mathbf{M}_{\sigma_n}^{-1}(\mathbf{e}_3 - \mathbf{e}_2),$$

(iii) $\forall n \in \{0,\ldots,N\}$, $\forall i \in \{1,2,3\}$, $\mathcal{W}_n$ *has* $\langle \mathbf{e}_i, \mathbf{a}_n\rangle$ *faces of type* i.

Proof. (i) Since $\mathcal{W}_0 \subset \mathcal{P}_{0,\mathbf{1}}$, Theorem 1 implies that $\mathcal{W}_N \subset \mathcal{P}_{0,\mathbf{a}_N}$. In addition, we have for all $n \in \{1,\ldots,N\}$:

$$\begin{aligned}
\mathcal{W}_n &= \cup_{i\in\mathcal{A}}\ E_1^*(\sigma_{n\cdots 0})(\mathbf{0}, i^*) && \text{(Definition 5)}\\
&= \cup_{i\in\mathcal{A}} \cup_{(s,j)\in\mathcal{S}^i_{\sigma_n}} \left(\mathbf{M}_{\sigma_0}^{-1}\cdots\mathbf{M}_{\sigma_n}^{-1} l(s) + E_1^*(\sigma_{n-1\cdots 0})(\mathbf{0}, j^*)\right) && \text{(Theorem 2)}\\
&\supset \cup_{j\in\mathcal{A}}\ E_1^*(\sigma_{n-1\cdots 0})(\mathbf{0}, j^*) = \mathcal{W}_{n-1} && \text{(Definition 5),}
\end{aligned}$$

where the inclusion comes from the trivial fact that for all $j \in \mathcal{A}$, $(\epsilon, j) \in \cup_{i\in\mathcal{A}} \mathcal{S}^i_{\sigma_n}$, i.e., the word $\sigma_n(j)$ ends with a letter $i \in \mathcal{A}$.

(ii) from (i), we have $\mathcal{W}_n \in \mathcal{P}_{0,\mathbf{a}_n}$ and from Proposition 2 (iii) (with the relation $\mathbf{U}_n^T = \mathbf{M}_{\sigma_n}^{-1}$), we have $\langle \mathbf{M}_{\sigma_0}^{-1} \cdots \mathbf{M}_{\sigma_n}^{-1}(\mathbf{e}_2 - \mathbf{e}_1), \mathbf{a}_n \rangle = 0$ and $\langle \mathbf{M}_{\sigma_0}^{-1} \cdots \mathbf{M}_{\sigma_n}^{-1}(\mathbf{e}_3 - \mathbf{e}_2), \mathbf{a}_n \rangle = 0$. Hence, for every $a, b \in \mathbb{Z}$, $\mathcal{W}_n + a\big(\mathbf{M}_{\sigma_0}^{-1} \cdots \mathbf{M}_{\sigma_n}^{-1}(\mathbf{e}_2 - \mathbf{e}_1)\big) + b\big(\mathbf{M}_{\sigma_0}^{-1} \cdots \mathbf{M}_{\sigma_n}^{-1}(\mathbf{e}_3 - \mathbf{e}_2)\big) \subset \mathcal{P}_{0,\mathbf{a}_n}$.

(iii) According to the definition of $E_1^\star$, equation (3), the number of faces of type j^0 in $E_1^*(\sigma)(\mathbf{0}, 1^\star)$ is equal to the number of pairs (s, j^0) in S^1_σ, which is equal to $|\sigma(j^0)|_1$, i.e., the number of occurences of 1 in $\sigma(j^0)$. More generally, the number of faces of type j^0 in $\cup_{i\in\mathcal{A}} E_1^*(\sigma)(\mathbf{0}, i^\star)$ is equal to $|\sigma(j^0)|_1 + |\sigma(j^0)|_2 + |\sigma(j^0)|_3$, i.e., the number of letters in $\sigma(j^0)$. Similarly, the number of faces of type i in $\mathcal{W}_n$ is equal to the number of letters in $(\sigma_{n\cdots 0})(i)$ and is equal to

$$\langle l(\sigma_{n\cdots 0}), \mathbf{1} \rangle = \langle \mathbf{M}_{\sigma_{n\cdots 0}} \mathbf{e}_i, \mathbf{1} \rangle = \langle \mathbf{e}_i, (\mathbf{M}_{\sigma_{n\cdots 0}})^T \mathbf{1} \rangle = \langle \mathbf{e}_i, \mathbf{M}_{\sigma_0}^T \cdots \mathbf{M}_{\sigma_n}^T \mathbf{1} \rangle = \langle \mathbf{e}_i, \mathbf{a}_n \rangle.$$

□

Theorem 3 shows that our method provides a hierarchical set of patterns, all included in a given stepped plane (i). The pattern of the highest level periodically generates the underlying stepped plane (ii) and is of minimal size (iii) because if one sum the normal vector of all its faces, we get exactly $\mathbf{a}_N$, i.e., the normal of the stepped plane, and one cannot expect to find a smaller pattern with the same normal. We discuss below two additional properties that we would like to have: shape compactness and connectivity.

3.3 Geometrical and Topological Issues

Our first remark is that the choice of the algorithm has a great impact on the shape of the pattern (see Fig. 5). In addition, the patterns generated using the algorithms H and R are much more compact. This is because the basis of the lattice $\{\mathbf{x} \in \mathbb{Z}^3 \mid \langle \mathbf{x}, \mathbf{a}_N \rangle = 0\}$ returned by the algorithm R (resp. algorithm H) is experimentally always (resp. almost always) reduced [18]. A first short-term perspective is to bound the distance of the pattern boundary to the origin in order to objectively compare the patterns generated by different algorithms. For the algorithms H and R, such bound may involve geometrical arguments based on the empty-circumsphere criterion.

Our second remark is about the connectivity of the patterns. There are no a priori guarantees that ensure that the patterns are vertex- or edge-connected. The connectivity of the last pattern is linked to the choice of substitutions, because one can associate two substitutions to one incidence matrix. Fig. 6 and Fig. 7 show that one can end up with patterns of different topology, depending on which substitutions are used. We have experimentally noticed that there is always a sequence of substitutions among the 2^N admissible sequences that keep the pattern edge-connected (see Fig. 6). A second short-term perspective is to prove that such a connecting sequence indeed always exists and to design an algorithm that finds it. Even if there is a way of generating edge-connected patterns from specific sets of substitutions (see for instance [6]), the literature currently lacks general results we could directly reuse in our setting.

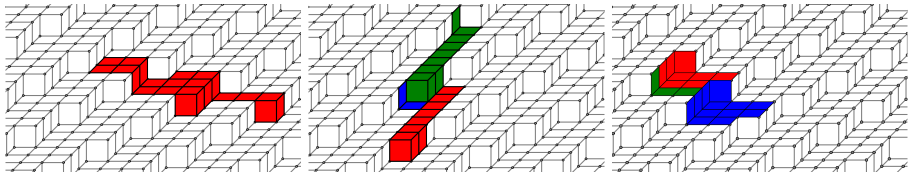
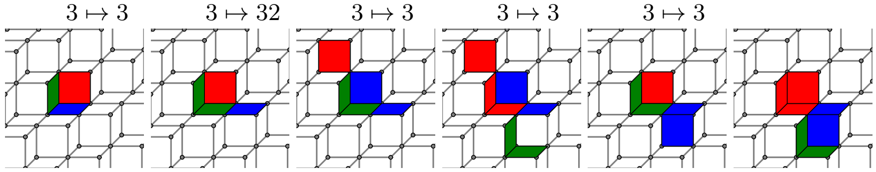


Fig. 5: Patterns of normal (2,6,15) generated by Brun, Selmer and H from left to right (same color convention as the one used in Fig. 4). In all cases, every substitution σ has been chosen so that for all $i \in \{1, 2, 3\}$, $\sigma(i)$ starts with i. Even if Brun does not correspond to Definition 2, we have included it for comparison, but keeping only the first red set because it ends on $(1, 0, 0)$ instead of $(1, 1, 1)$.

$$\sigma_0 : \begin{array}{l} 1 \mapsto 1 \\ 2 \mapsto 2 \\ 3 \mapsto 3 \end{array} \quad \sigma_1 : \begin{array}{l} 1 \mapsto 1 \\ 2 \mapsto 2 \\ 3 \mapsto 32 \end{array} \quad \sigma_2 : \begin{array}{l} 1 \mapsto 13 \\ 2 \mapsto 2 \\ 3 \mapsto 3 \end{array} \quad \sigma_3 : \begin{array}{l} 1 \mapsto 1 \\ 2 \mapsto 21 \\ 3 \mapsto 3 \end{array} \quad \sigma_2' : \begin{array}{l} 1 \mapsto 31 \\ 2 \mapsto 2 \\ 3 \mapsto 3 \end{array} \quad \sigma_3' = \sigma_3$$



$\mathcal{W}_0$ $\mathcal{W}_1$ $\mathcal{W}_2$ $\mathcal{W}_3$ $\mathcal{W}_2'$ $\mathcal{W}_3'$

Fig. 6: The substitutions have been obtained from the reduction of $(2, 2, 3)$ using the algorithm H (same color convention as the one used in Fig. 4). Using σ_2' instead of σ_2 can make the pattern edge-connected.

4 Conclusion

We have introduced a three-dimensional version of the Euclidean algorithm that turns out to be closely related to plane-probing methods appearing in digital geometry. With the help of generalized substitutions, we have presented a way of generating hierarchical sets of pieces of digital planes. The patches of highest level periodically cover the underlying digital planes and are of limited size. We expect to obtain soon theoretical guarantees regarding the shape and connectivity of the generated patches. After having achieved this goal, we will use that generation method to improve the local analysis of digital surfaces using plane-probing.

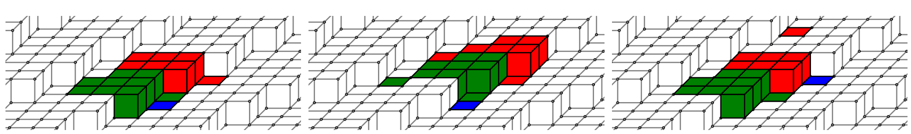


Fig. 7: Patterns of normal $(2, 5, 17)$ generated by the algorithm H. From left to right: connected pattern, not edge-connected, not vertex-connected.

References

1. Arnoux, P., Furukado, M., Harriss, E., Ito, S.: Algebraic numbers, free group automorphisms and substitutions on the plane. Trans. Amer. Math. Soc. **363**, 4651–4699 (2011)
2. Arnoux, P., Ito, S.: Pisot substitutions and Rauzy fractals. Bulletin of the Belgian Mathematical Society Simon Stevin **8**(2), 181–208 (2001)
3. Berthé, V., Domenjoud, É., Jamet, D., Provençal, X.: Fully Subtractive algorithm, tribonacci numeration and connectedness of discrete planes. Research Institute for Mathematical Sciences, Lecture note Kokyuroku Bessatu B **46**, 159–174 (2014)
4. Berthé, V., Fernique, T.: Brun expansions of stepped surfaces. Discret. Math. **311**(7), 521–543 (2011)
5. Berthé, V., Jamet, D., Jolivet, T., Provençal, X.: Critical connectedness of thin arithmetical discrete planes. In: Proc. DGCI. pp. 107–118 (2013)
6. Berthé, V., Lacasse, A., Paquin, G., Provençal, X.: A study of Jacobi-Perron boundary words for the generation of discrete planes. Theor. Comput. Sci. **502**, 118 – 142 (2013)
7. Brimkov, V., Coeurjolly, D., Klette, R.: Digital planarity – a review. Discret. Appl. Math. **155**(4), 468–495 (2007)
8. Domenjoud, E., Laboureix, B., Vuillon, L.: Facet connectedness of arithmetic discrete hyperplanes with non-zero shift. In: Proc. DGCI (2019)
9. Domenjoud, E., Provençal, X., Vuillon, L.: Facet connectedness of discrete hyperplanes with zero intercept: The general case. In: Proc. DGCI. pp. 1–12 (2014)
10. Domenjoud, E., Vuillon, L.: Geometric Palindromic Closure. Uniform Distribution Theory **7**(2), 109–140 (2012)
11. Fernique, T.: Multidimensional Sturmian Sequences and Generalized Substitutions. International Journal of Foundations of Computer Science **17**, 575–600 (2006)
12. Fernique, T.: Generation and recognition of digital planes using multi-dimensional continued fractions. Pattern Recognition **42**(10), 2229–2238 (2009)
13. Jamet, D., Lafrenière, N., Provençal, X.: Generation of digital planes using generalized continued-fractions algorithms. In: Proc. DGCI. pp. 45–56 (2016)
14. Jolivet, T.: Combinatorics of Pisot Substitutions. Phd thesis, Université Paris Diderot, University of Turku (2013)
15. Klette, R., Rosenfeld, A.: Digital straightness – a review. Discret. Appl. Math. **139**(1-3), 197–230 (2004)
16. Labbé, S., Reutenauer, C.: A d-dimensional extension of christoffel words. Discrete and Computational Geometry **54**(1), 152–181 (2015)
17. Lachaud, J.O., Meyron, J., Roussillon, T.: An optimized framework for plane-probing algorithms. J. Math. Imaging Vis. **62**, 718–736 (2020)
18. Lachaud, J.O., Provençal, X., Roussillon, T.: Two plane-probing algorithms for the computation of the normal vector to a digital plane. J. Math. Imaging Vis. **59**, 23–39 (2017)

A Proofs

Proof (of Theorem 2).

$$\begin{aligned}
E_1^*(\sigma_{n\cdots 0})(\mathbf{0},i^*) &= E_1^*(\sigma_{n-1\cdots 0})\big(E_1^*(\sigma_n)(\mathbf{0},i^*)\big)\\
&= E_1^*(\sigma_{n-1\cdots 0})\left(\bigcup_{(s,j)\in\mathcal{S}^i_{\sigma_n}}(\mathbf{M}_{\sigma_n}^{-1}l(s),j^*)\right)\\
&= \bigcup_{(s,j)\in\mathcal{S}^i_{\sigma_n}} E_1^*(\sigma_{n-1\cdots 0})\big(\mathbf{M}_{\sigma_n}^{-1}l(s),j^*)\big)\\
&= \bigcup_{(s,j)\in\mathcal{S}^i_{\sigma_n}} \big((\mathbf{M}_{\sigma_{n-1}}\cdots\mathbf{M}_{\sigma_0})^{-1}(\mathbf{M}_{\sigma_n})^{-1}l(s)+E_1^*(\sigma_{n-1\cdots 0})(\mathbf{0},j^*)\big)\\
&= \bigcup_{(s,j)\in\mathcal{S}^i_{\sigma_n}} \big((\mathbf{M}_{\sigma_n}\cdots\mathbf{M}_{\sigma_0})^{-1}l(s)+E_1^*(\sigma_{n-1\cdots 0})(\mathbf{0},j^*)\big).
\end{aligned}$$

The second to last line comes from

$$E_1^*(\sigma_{n-1\cdots 0})(\mathbf{x},i^*) = (\mathbf{M}_{\sigma_{n-1}}\cdots\mathbf{M}_{\sigma_0})^{-1}\mathbf{x}+E_1^*(\sigma_{n-1\cdots 0})(\mathbf{0},i^*),$$

since $(\mathbf{M}_{\sigma_{n-1}}\cdots\mathbf{M}_{\sigma_0})^{-1}$ does not depend on the union in the definition of E_1^*, equation (3) (see also [14, Proposition 1.2.4, item (2)]). □