

Another definition for digital tangents

Thierry Monteil

CNRS – Université Montpellier 2
<http://www.lirmm.fr/~monteil>

Abstract. The aim of this short note is to describe the set of finite words that appear in the cutting sequences of a smooth curve to arbitrary small scale. This language strictly contains the factors of Sturmian words, and can be decided by a linear time algorithm.

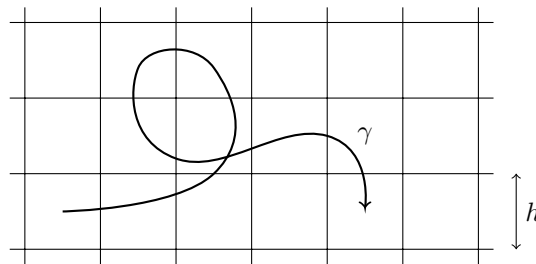
Keywords: Cutting sequence, symbolic coding, tangent estimation, multi-grid convergence, digital straight segment, Sturmian word.

1 Introduction

A *smooth curve* is a map γ from a compact interval I of the real line to the plane, which is C^∞ and such that $\|\gamma'(t)\| > 0$ for any $t \in I$ (this last property is called *regularity*). Any such curve can (and will be considered to) be arc-length reparametrised (*i.e.* $\forall t \in I, \|\gamma'(t)\| = 1$).

We can approximate such a curve by drawing a square grid of width (*mesh* or *resolution*) h on the plane, and look at the sequence of squares that the curve meets. For a generic position of the grid, the curve γ does not hit any corner and crosses the grid transversally, hence the curve passes from a square to a square that is located either *right*, *up*, *left* or *down* of it. We record this sequence of moves and define the *cutting sequence* of the curve γ with respect to this grid as a word w on the alphabet $\{r, u, l, d\}$ which tracks the lines of the grid crossed by the curve γ .

The following picture shows a curve γ with cutting sequence $rruuldrrrd$.



Note that since the grid can be translated, a given curve may have more than one cutting sequence for a given mesh h . Our knowledge of the curve from one of its cutting sequences increases when the mesh h decreases, and when the mesh approaches 0, the local patterns of the cutting sequence allow the digital geometers to define infinitesimal estimators (like tangents or curvature), for example

by recognising maximal straight segments appearing in the cutting sequence [4] [3].

Cutting sequences associated to straight segments are known to be exactly the *balanced words* (see Section 3.5), which are also the finite factors of Sturmian words. A problem is that some non-balanced words appear at arbitrary small scale even for very regular curves. For example, the word *rruu* appears in the cutting sequence of infinitely many Bresenham circles [5] but is not balanced. This “infinitesimal noise” has the following consequence: when the mesh tends to 0, the length of the maximal segments does not necessarily tend to infinity, causing some convergence problems for the estimators.

A solution might be found in another definition of what digital tangents are: in this paper, we consider all the finite words that appear in the cutting sequences of some smooth curve for arbitrary small scale (like *rruu* for the circle). More precisely, let $F(\gamma, G)$ denote the set of factors of the cutting sequence of the curve γ with respect to the square grid G (when the curve hits a corner, the cutting sequence is not defined and we set $F(\gamma, G) = \emptyset$). We define the *asymptotic language* of γ by

$$T(\gamma) = \limsup_{\text{mesh}(G) \rightarrow 0} F(\gamma, G) = \bigcap_{\varepsilon > 0} \bigcup_{\text{mesh}(G) \leq \varepsilon} F(\gamma, G).$$

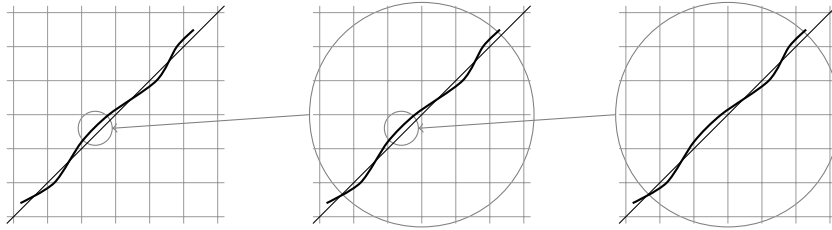
More generally, when X is a set of curves, let us denote by $T(X)$ the set $\bigcup_{\gamma \in X} T(\gamma)$. When X is the set of smooth curves, we simply denote $T(X)$ by T , and call its elements *tangent words*. The aim of this note is to characterise this language. Note that we are only interested in the language, not on the rate of convergence to it.

2 Analytic characterisation

Proposition 1. *A word w is tangent if, and only if, for any $\varepsilon > 0$, w is the cutting sequence of a curve γ which is ε -close (for the C^1 norm) to a straight segment (the grid is fixed).*

Proof. The *only if* part is straightforward. For example, the word 00011 cannot be tangent, since, when the mesh goes to zero, it corresponds to a point where the slope of the curve is both at most $1/2$ (because of the factor 000) and at least 1 (because of the factor 11).

For the *if* part, since the space C^∞ endowed with the distance defined by $d(\gamma, \delta) = \sum_{n \geq 0} 2^{-\min(1, \sup_{t \in I} \|\gamma^{(n)}(t) - \delta^{(n)}(t)\|)}$ is complete, we can build a convenient curve by accumulating arbitrarily small and flat copies of a curve whose cutting sequence is w (the n th copy of the curve should be flat enough so that its first n derivatives are very small, ensuring that we get a Cauchy sequence whose limit has the desired property).



□

3 Combinatorial description

For the sake of simplicity, let us first focus on curves going right and up, *i.e.* smooth curves such that both coordinates of $\gamma'(t)$ are positive for any t . Let us rename r and u by 0 and 1 respectively to stick to the usual notation.

3.1 Renormalisation (desubstitution)

Balanced words are known to have a hierarchical structure, where the morphisms $\sigma_0 = (0 \mapsto 0, 1 \mapsto 10)$ and $\sigma_1 = (0 \mapsto 01, 1 \mapsto 1)$ play a crucial role [8] [6]. The same renormalisation applies to tangent words.

Proposition 2. *Let w be a finite word over the alphabet $\{0, 1\}$. The following are equivalent:*

- w is tangent.
- $\sigma_0(w)$ is tangent.
- $\sigma_1(w)$ is tangent.

Proof (sketch). It suffices to notice that applying the substitution σ_1 (resp. σ_0) to the word w geometrically corresponds to applying the linear bi-uniformly continuous bijection given by the matrix $M_0 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ (resp. $M_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$). Such maps preserve tangency. A clear presentation of this argument can be read in [10]. □

Hence, given a finite word w , we can “desubstitute” it by

- removing one 0 per run of 0 if 11 does not appear in w , or
- removing one 1 per run of 1 if 00 does not appear in w .

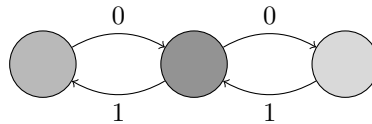
If we repeat this process as much as possible, we get a *derivated word* denoted by $d(w)$. The word w is balanced if, and only if, $d(w)$ is the empty word, and the derivation process is related to the continued fraction development of the slope of the associated straight segment.

3.2 Last step

To finish the description of tangent words going up and right, it suffices to describe the set of derivated words that are tangent words.

Let us say that a word w is *diagonal* if one of the following equivalent conditions hold:

- w is recognised by the following automaton with three states, which are all considered as initial and accepting:

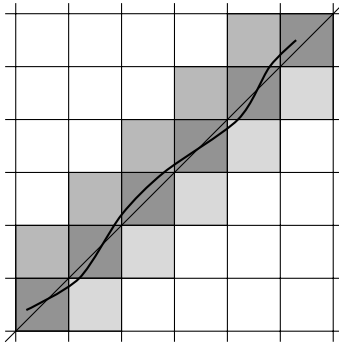


- the word w is in the language defined by the regular expression $(\varepsilon|0|1)(01|10)^*(\varepsilon|0|1)$,
- there exists an integer k such that for any prefix p of w , the difference between the number of occurrences of 0 in p and the number of occurrences of 1 in p is between k and $k + 2$, *i.e.*

$$\max\{|p|_0 - |p|_1 \mid p \text{ is a prefix of } w\} \leq \min\{|p|_0 - |p|_1 \mid p \text{ is a prefix of } w\} + 2,$$

where $|p|_i$ denotes the number of occurrences of the letter i in p .

For example, the word 0110100110 (which is not balanced) is diagonal:



Proposition 3. *A word w is tangent if, and only if $d(w)$ is diagonal.*

Proof (sketch). If $d(w)$ is not the empty word, then 00 and 11 are occurrences of $d(w)$, hence the slope of the tangent is 1 (the diagonal). The three gray diagonals correspond to the three states of the automaton defining the diagonal words. \square

3.3 Example

The word $w = 0111011011011101$ is tangent: it can be desubstituted as 011010110110, then 010001010, and finally $d(w) = 10011$, which is diagonal since it can be written as $(10)(01)1$.

3.4 Complexity

The *complexity* of a language L is the map that counts, for any integer n , the number of elements of L of length n . Since T contains the language defined by the regular expression $(01|10)^*$, the complexity of the language T has exponential growth, whereas the complexity of the language of balanced words has cubical growth (an explicit formula was given in [7]).

3.5 Balance

A binary word w is said to be *k-balanced* if for any two factors u and v of w of the same length, the number of 0 in u and the number of 0 in v differ by at most k . The 1-balanced words (also known as *balanced* words) correspond to the cutting sequences of discrete line segments.

We already saw that the balanced words form a strict subset of the tangent words, since 0011 is tangent but not balanced. Conversely, tangent words form a strict subset of 2-balanced words, since 00011 is 2-balanced but not tangent.

3.6 Algorithm

The combinatorial description provides a linear time algorithm that decides whether a word is tangent or not: concerning the renormalisation, we can accelerate the desubstitution procedure by removing a run equal to the length of the shortest inner run from any run of the non-isolated letter (including possible leading and trailing runs even if they have shorter length). Each such accelerated desubstitution reduces the size of the word by at least $2/3$, hence, if c denotes the complexity of the `derivate` algorithm that maps a word w to $d(w)$, we have:

$$c(n) \leq n + c((2/3)n) \leq n + (2/3)n + c((2/3)^2n) \leq \dots \leq n \sum_{k \geq 0} (2/3)^k \leq 3n .$$

The last step consist in deciding whether the obtained derivated word matches the regular expression $(\varepsilon|0|1)(01|10)^*(\varepsilon|0|1)$, this can be achieved in linear time as well. A basic implementation exists in the free open source mathematical software Sage: the `is_tangent()` method is being reviewed as ticket #9877 and can be tested at <http://www.sagenb.org/home/pub/2123/>.

Moreover, any existing digital straight segment recognition algorithm based on the hierarchical structure can easily be adapted to tangent words recognition, in particular, we can construct on-line linear-time algorithms for this purpose [1]. Also, digital straight segment recognition can easily be replaced by tangent word recognition in existing digital geometry algorithms, in particular for those dealing with curve segmentation.

4 Other classes of curves

Let us briefly study the asymptotic language of some other classes of curves. Each class is stable by the action of invertible linear maps on the plane, hence

the associated languages turn out to be stable by the renormalisation procedure described in subsection 3.1.

4.1 Continuous curves

Proposition 4. *If X denotes the set of continuous curves (defined on a closed interval), then all the words are admissible: $T(X) = \{0, 1\}^*$ (or $T(X) = \{r, u, l, d\}^*$ if we deal with curves going in all directions).*

Proof. Given a word w , it suffices to construct a curve $\gamma : [0, 1/2] \rightarrow \mathbb{R}^2$ whose cutting sequence is w , then to glue a smaller copy of it from $[1/2, 3/4]$ to \mathbb{R}^2 , then to glue another even smaller copy from $[3/4, 7/8]$ to \mathbb{R}^2 and so on to get a curve $[0, 1] \rightarrow \mathbb{R}^2$ that admits w in its cutting sequence at arbitrary small scales. \square

Note that the statistical properties of the finite words appearing in the asymptotic language of generic continuous functions have been studied (in a similar framework) in [9].

4.2 C^k curves

Proposition 5. *If X denotes the set of C^k regular curves, then $T(X) = T$ ($1 \leq k \leq \infty$).*

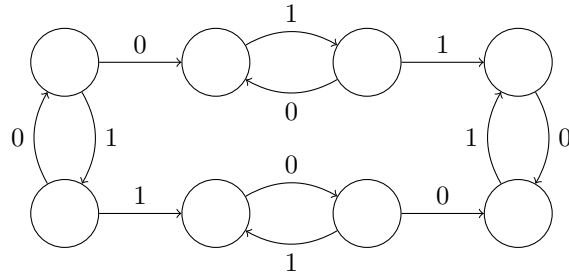
Proof. The *only if* part of the proof of Proposition 1 only uses the fact that the curve is C^1 . The C^∞ function built in the *if* part is in particular C^k for any $1 \leq k \leq \infty$. \square

Hence, the asymptotic language shows some stability with respect to the regularity of the curve. This is a hint to grasp higher order notions such as curvature: we will probably have to look one step further, like the mutual organisation of those words. Note that existing methods based on maximal digital straight segments were proven to be not rigorous [2].

4.3 Analytic curves

However, many tangent words won't appear for the more rigid class of analytic curves.

Proposition 6. *If X denotes the set of analytic regular curves, then a finite word w is in $T(X)$ if, and only if, $d(w)$ is recognised by the following automaton with eight states, which are all considered as initial and accepting:*



Proof (sketch). The tangent words oscillating at least twice should be removed. For example, the word 001100 cannot be in $T(X)$ since, if a corresponding analytic curve γ is denoted by (x, y) , then the derivative of $(x - y)$ has two close zeroes, but since the mesh can be arbitrarily small, we get an accumulation of such zeroes, which contradicts the analyticity of the map $(x - y)'$. \square

Note that those remaining words can all be found in $T(\mathcal{C})$, where \mathcal{C} denotes the set of planar circles (winding both clockwise and counterclockwise).

4.4 Smooth curves defined on an open interval

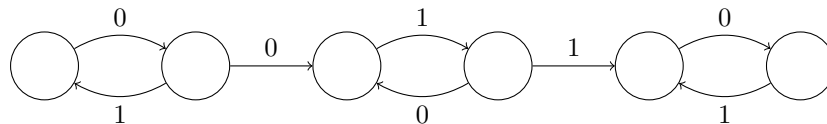
To get a uniform control on the tangents of the curve, we assumed the compactness of the interval on which the smooth curve is defined. Here is why this assumption was necessary.

Proposition 7. *If X denotes the set of regular smooth curves defined on an open interval, then $T(X) = \{0, 1\}^*$ (or $T(X) = \{r, u, l, d\}^*$ if we deal with curves going in all directions).*

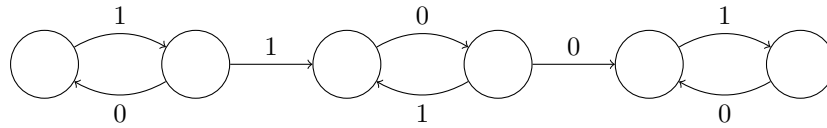
Proof. As in the proof of Proposition 4, we can accumulate any noise near a boundary of the interval, since there is no need to ensure derivability at the endpoint. \square

4.5 Smooth curves with nowhere zero curvature

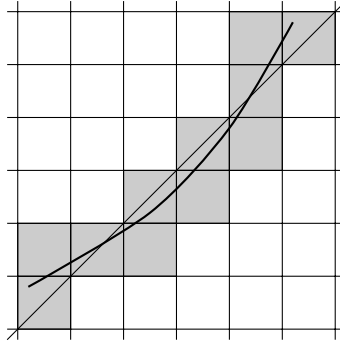
Proposition 8. *If X (resp. Y) denotes the set of smooth curves whose curvature is positive (resp. negative), then a finite word w is in $T(X)$ (resp. $T(Y)$) if, and only if, $d(w)$ is recognised by the following automaton with six states, which are all considered as initial and accepting:*



and respectively for smooth curves with negative curvature:



For example, the word 1001010110 appears in the cutting sequence of a smooth curve with positive curvature:



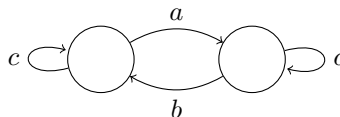
We can notice that the asymptotic language of analytic regular curves is the union of those two languages, which is the asymptotic language of smooth curves with nowhere zero curvature, showing again some stability with respect to the considered class of curves.

5 Curves going in all directions

Let us finish by the complete characterisation of tangent words without any assumption on the direction of the curve.

The four letters $\{r, u, d, l\}$ cannot all appear simultaneously in a tangent word. If only two letters appear in a tangent word and are consecutive for the cyclic order $r < u < l < d$, then, if we replace them by 0 and 1, we have the same characterisation as above.

If two non-consecutive letters a and b appear in a tangent word w , we are in the case where the tangent is vertical or horizontal, and this case is similar to the last step described before (no renormalisation must be done): there exists a letter c , different from a and b , such that w is recognised by the following automaton with two states, both initial and accepting:



That is, after removing the occurrences of c in w , the letters a and b are alternating. Equivalently the word w is in the language defined by the regular expression $c^*(\varepsilon|b)(c^*ac^*bc^*)^*(\varepsilon|a)c^*$.

6 Conclusion

We introduced and described the asymptotic language of smooth curves, *i.e.* the set of words that can survive in the cutting sequence of a smooth curve when the grid mesh goes to zero, and discussed some of its properties. We saw that those words are closely related to the tangents of the curve.

So, we can use them to change one of the most basic primitive for discrete geometry, that of discrete tangency: instead of using digital straight segment, we could use tangent words. One possible advantage is that the smallest length of maximal tangent words one can find in a smooth curve goes to infinity when the grid mesh goes to zero, whereas it may stay bounded for straight segments. What can be done with this?

In terms of complexity, most tangent words are not balanced. However, we can notice a kind of prevalence of balanced words among tangent words. For example, the intersection of the asymptotic languages of all closed smooth regular curves is the set of balanced words (those are the most “stable”). Hence, the study of probabilistic aspects of the occurrences of tangent words should be interesting.

References

1. Egbert Creutzburg, Albrecht Hübler, and Volker Wedler. On-line erkenung digitaler geradensegmente in linearer zeit. *Proceedings of GEO-BILD'82, Wiss. Beitrage der FSU Jena*, pages 48–65, 1982.
2. François de Vieilleville, Jacques-Olivier Lachaud, and Fabien Feschet. Convex digital polygons, maximal digital straight segments and convergence of discrete geometric estimators. *J. Math. Imaging Vision*, 27(2):139–156, 2007.
3. Simon Hermann and Reinhard Klette. A comparative study on 2d curvature estimators. *International Conference on Computing: Theory and Applications*, 0:584–589, 2007.
4. Reinhard Klette and Azriel Rosenfeld. Digital straightness—a review. *Discrete Appl. Math.*, 139(1-3):197–230, 2004.
5. Zenon Kulpa. On the properties of discrete circles, rings, and disks. *Computer Graphics and Image Processing*, 10(4):348 – 365, 1979.
6. M. Lothaire. *Algebraic combinatorics on words*, volume 90 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2002. Chapter 3, *Sturmian Words* (by Jean Berstel and Patrice Séébold).
7. Filippo Mignosi. On the number of factors of Sturmian words. *Theoret. Comput. Sci.*, 82(1, Algorithms Automat. Complexity Games):71–84, 1991.
8. N. Pytheas Fogg. *Substitutions in dynamics, arithmetics and combinatorics*, volume 1794 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 2002. Chapter 6, *Sturmian Sequences* (by Pierre Arnoux).
9. Cristobal Rojas and Serge Troubetzkoy. Coding Discretizations of Continuous Functions. Preprint available at <http://arxiv.org/abs/0907.2299>, 2009.
10. John Smillie and Corinna Ulcigrai. Symbolic coding for linear trajectories in the regular octagon. Preprint available at <http://arxiv.org/abs/0905.0871>, 2009.